

National Library of Canada

Bibliothèque nationale du Canada

Direction des acquisitions et

des services bibliographiques

Acquisitions and Bibliographic Services Branch

395 Wellington Street Ottawa, Ontario K1A 0N4 395, rue Wellington Ottawa (Ontario) K1A 0N4

Accellation Contracted and

Our fair - Notice references

NOTICE

AVIS

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments. La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canadä

_

INTERFACIAL TENSION IN POLYMER BLENDS:

MEASUREMENTS AND ANALYSIS

by

NICOLE RAYMONDE DEMARQUETTE

A Thesis Submited to the Faculty of Graduate Studies and Research in Partial Fulfillment of the Requirement for the Degree of Doctor of Philosophy

Department of Chemical Engineering McGill University Montreal, Canada

December 1993

Copyright © Nicole Raymonde Demarquette (1993)



National Library of Canada

Acquisitions and Bibliographic Services Branch Bibliothèque nationale du Canada

Direction des acquisitions et des services bibliographiques

395 Wellington Street Ottawa, Ontano K1A 0N4 395, rue Wellington Ottawa (Ontario) K1A 0N4

Your ble - Votie reference

Our hie Notre référence

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons. L'auteur a accordé une licence irrévocable et non exclusive à la Bibliothèque permettant nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission. L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-94610-5



To my Parents and Vocezinho,

<u>ABSTRACT</u>

Interfacial tension is one of the most important parameters that influence the morphology of polymer blends. However, few data are available regarding interfacial tension between polymers due to experimental difficulties.

Two devices, one based on the pendant drop method and another based on the spinning drop method, were constructed in this work to conduct an experimental study of interfacial tension between polymers. With both instruments, it was possible to view the drop in real time and to calculate the interfacial tension on-line. The following aspects were considered: special syringe design, the necking and capillary effects, in the pendant drop method, and sealing of the rotating tubes and density measurement in the spinning drop method. The two devices were shown to be reliable and complementary. A method to infer interfacial tension from transient measurements was suggested.

Interfacial tension was measured for different polymer pairs. The effects of temperature, molecular weight, bimodal blending and molecular weight distribution were evaluated. The effect of adding compatibilizers was also studied. Surface analysis of the interfaces was performed using E.S.CA. to explain the experimental results.

The experimental results were compared with the predictions of lattice theories and the square gradient theory. A strategy to evaluate the Flory-Huggins interaction parameter between polymers was developed. When this strategy was employed, theoretical predictions and experimental data showed good agreement for the influence of temperature and molecular weight on interfacial tension. However, the theory could not predict the effect of polydispersity on interfacial tension.

RESUME

La tension interfaciale est l'un des plus importants paramètres qui influence la morphologie des mélanges de polymères. Cependant, les valeurs de tension interfaciale reportées dans la litterature sont rares dû à des difficultés expérimentales.

Deux machines, l'une basée sur le principe de la goutte pendante et l'autre sur le principe de la goutte tournante ont été construites afin de conduire une étude expérimentale sur la tension interfaciale entre polymères. Avec les deux machines, il était possible de visualiser la goutte en temps réel et de calculer la tension interfaciale simultanément. Pour ces deux instruments, les aspects suivants furent abordés: la conception d'une seringue spéciale ainsi que les effets de goutte tombante et de capillarité pour l'appareil basé sur le principe de la goutte pendante et l'étanchéité des tubes et mesures de densité pour l'appareil basé sur le principe de la goutte tournante. Les deux procédés furent démontrés fiables et complémentaires. Une méthode pour obtenir la tension interfaciale à partir de mesures effectuées lors de l'état transient fut suggerée.

La tension interfaciale fut mesurée pour différentes paires de polymères. L'influence de la température, du poids moléculaire, et de la distribution moléculaire fut évaluée. La tension interfaciale pour des mélanges bimodaux fut aussi étudiée. Les différentes valeurs de tension interfaciale furent comparées, pour les différents systèmes, et une explication des phénomènes observés est présentée. L'influence d'agents compatibilisants fut étudiée. Les interfaces furent étudiées par E.S.C.A afin d'expliquer les résultats expérimentaux.

Les résultats expérimentaux furent comparés aux prédictions des théories de lattice et square gradient. Une stratégie, pour évaluer le paramètre d'interaction Flory-Huggins entre deux polymères fut développée. Quand cette stratégie fut utilisée, les prédictions théoriques et les données expérimentales concordèrent pour l'influence de la température et du poids moléculaire sur la tension interfaciale. Cependant, la théorie ne pouvait pas prédire l'effet de la polydispersité.

ACKNOWLEDGMENTS

Thank you, Merci, Obrigada, Danke schön, شكرا , Dhānyabäd, Arigato, Nihao, Gracias, Ενχαρισλω, Gracie, Dziekuje, CΠACUBA, مرسي, Vatlho'... I do not think that any of these words can really express my gratitude for all the persons who helped me throughout this work, without whom it would not have been possible.

First, I would like to thank **Professor Kamal**, my supervisor, for his patience, guidance, moral and technical support during those last three years. I hope he forgives me for spending so much money in the workshop and for sometimes being overexcited about an idea that wouldn't always work at the end.

I would also like to thank several Professors in the Chemical Engineering Department for the help they provided me during the last three years, **Professor Dealy** thank you for helping me during the seminar, **Professor Rey** for his judicious pieces of advice, **Professor Patterson** for his moral support before ANTEC conference, **Professor Munz** who loaded his suitcases with my thesis when coming back from Brazil, **Professor Cooper** for his help in Chemistry, **Professor Weber** for helping me in understanding drops and **Professor Vera** without whom I would never have found the courage to work some thermodynamic theories. Thank you (and Gracias !!!).

I would also like to express my sincere thanks to **Professor Axelrad**, from the Mechanical Engineering Department at McGill University, for his pieces of advice, **Professor Zoller**, from the Mechanical Engineering Department of the University of Colorado, for his helpful discussion on density measurements, **Professor Joseph**, from the Aerospace and Mechanical Engineering Department of the University of Minnesota for his helpful discussions, **Professor Koberstein**, From the Chemical Engineering Department of the University of the University of Connecticut for his helpful telephonic conversations, **Claude Verdier**, from the Institut de Mecanique of Grenoble for his helpful discussion on the spinning drop apparatus.

In the Chemical Engineering Department there are some people without whom life would be totally different, and that I would really like to thank. Starting with the women, there are the wonderful secretaries on the first floor: **Pat**, always running but always ready to help you and type all those letters for work permit, Medicare and..., **Anne**, always present when you need any information, Louise, Valerie, Jennifer, Claire and Barbara, always so nice. In the Chemical Engineering Department there is also: Jean Dumont, who always makes the life of students easier by having everything you might need or knowing where to get it (Merci Jean, l'affaire est ketchup maintenant!!!), the fantastic persons of the workshop and electronic workshop: Alain, the artist who made the spinning drop machine spin... (C'est Beau Alain, Ca a marché!!! It was noisy but it span. Un gros merci.), Charles, the "King" of the plungers and syringes who would hate to see me arriving in the workshop for some new plungers... (But Charles, it worked, no more capillary or necking effects), Walter, always ready to discuss far horizons, Mr. Krish, (thank you very much for you very good technical advises), Lou, who hurt himself twice working on my equipment, Mr Habib, (how many thermocouples did I get from you?), Frank, who probably still remembers my silly questions about electronics, Ed, always nice and suggesting analysis for you to do.

I would like also to thank Hydro-Quebec, NSERC and FCAR for their financial support.

Thank you to all the ones who left some footprints in the department, in particular all the people in the 5th floor of the polymer group: Yalda, so nice and always there to receive you, Marion, ("Elle est drole aujourd'hui ta goutte, je crois qu'elle dois avoir la varicelle!" Merci, Marion, et désolée pour le bruit... Embrasse bien Juliette et Anna pour moi), Katia, the one without whom the early mornings in the 5th floor would have been pretty lonely, Mazen, the person to know on the 5th floor, Imad, the "King" of C language, computers, (Oh by the way, when are you learning "D" language?), Furong, always nice, easy going even when because of you, his equipment gets all broken (By the way, how is it over there, upside down, in Australia?) John, any questions on polymer chemistry ask John!!! Probably he will know, Hamid, who helped me to prepare those blends which stink so much (How is Mr. Ali?) Marc, Laurent, Yoshito, (with Mazen, a good team!!!), Yoshio, Hideaki, Harry, Jean-Pierre, (Enarque et Astrophysicienne ca jette!!!) Hugues, Paul, Mike, Dib, Issam, Richard, Burke, Sassan, Tony, Savvas (hugs to little Georges), Martin, Meng, Bing.

Besides the fifth floor there are other persons in the department that I would like to thank, Antonio-Carlos, who loaded his handbag with some chapters of my thesis, Eric, who helped me so much although he was so busy writing his thesis, Hassan, (and what about the shear force?), Anne, always ready to have a small chat, (best hugs to Adele et Augustin), Bernard, Dr. Frydricovitch, Bing.

Also I would like to say un gros merci à André, from the electrical Engineering Department, (embrasse bien Cedric pour moi), Nandini, for helping me writing my programs.

Also thank you to Suzy Poulin, from ESCALAB at Ecole Polytechnique for performing my ESCA experiments, Giselle, in the Chemistry Department of McGill University who performed some of my G.P.C. experiments.

And most of all I would like to thank all the friends and persons who made my stay in Montreal so unforgettable and enjoyable, Mazen, for always having being so nice (Thank you for all the canastra games, the falafel, mansaf, all those good Arabic recipes), Vincent, one of the most knowledgeable person I know! and always so helpful, Earle, Liz, William, always so kind and always there to help and in case I needed someone to trust, Imad, who tried to teach me some Arabic, Chico and Norma, (e ai tudo bem. Vamos comer barros lucos este sabado.), Genevieve and Donald, who helped me to adapt to Montreal when I first arrived, the friends at Paprican, who helped me to come back to Montreal.

I would also like to thank all my family and in laws for being so helpful to me. Merci Parrain, tata Caty, Michel et Louis for having always been there when I needed to trust someone and I needed some help. Thank you also for having helped me all along my school life. Thank you also to my other aunts, Aline, Bernadette, Monique to their husbands and children for being always so nice and thoughtful when I was away from home. Obrigada Seu Milton (from whom I learned a very good expression "Gwen ta mao"), Dona Zilda for always being so nice and patient with me when I was writing my thesis in Brazil, Tania and Luiz for your niceness (et Luiz, c'est tout bijoux!), Silvia and Luciana, Dito and Daniel (the caro's tudos version U.S.A.), obrigada tambem Maria por tudo o que voce fez para mim.

Wou, Wou also to Active, Delurée, Grapete and Cherry, in what language can I express my thanks to you?

I would also like to thank my **Parents**, for always having been present when I needed anything, for moral support, for backing me up and for being so good teachers for life. Thank you for having supported my ideas, even if they were sometimes weird, thank you for having had so much patience raising me. Thanks, Merci.

Eventually, I would like to thank my husband, **Roberto Vocezinho**, for everything he did for me during those last four years, whether it was during my master or while I was doing this work, your help was so precious to me. Without you I would never have been able to do it. How many Sundays did you spend helping me when I was desperate because it was not working? How many times did you have to bear my bad mood and to try to put me back into a good mood? Thanks for being so patient, so nice and such a support for me, thanks also for all the pieces of advice at all levels. Obrigada.

TABLE OF CONTENTS

	Page
Abstract	i
Resume	ii
Acknowledgments	iii
Table of Contents	vii
List of Figures	xv
List of Tables	xxi

CHAPTER I. INTRODUCTION

1) INTRODUCTION	2
2) OBJECTIVES	4
3) ORGANIZATION OF THE THESIS	5
REFERENCES	6

1)	CHAPTER OUTLINE	8
2)	LITERATURE REVIEW	
	2.1. Introduction	8
	2.2. Static Methods	
	2.2.1. The Pendant Drop Method	9
	2.2.2. The Sessile Drop Method	11
	2.2.3. The Spinning Drop Method	12
	2.3. Dynamic Methods	
	2.3.1. The Breaking Thread Method	13

2.3.4. In	terfacial Tension Inferred from Rheological Measurements
2.4. Choice	of Methods
THE PEND	ANT DROP APPARATUS
3.1. Theory	
3.2. Evolutio	n of the Pendant Drop Method
3.3. Pendant	Drop Apparatus
3.3.1. G	eneral Description
3.3.2. T	he Optical System
3.3.3. T	he Experimental Cell
3.3.4. T	he Drop Insertion Device
3.3.5. T	he Measurement of Interfacial Tension
3,3,6, C	omputer Programs
a)	Edge Detection
b)) Smoothing of the Drop
C)	Shape Comparison
3.4. Experim	ental Procedures
3.4.1, Pi	reparation of the Polymeric Material
3.4.2. E	xperimental Procedure
3.5. Evaluation	on of the Apparatus
3.5.1. 0	ptical Distortion
3.5.2. C	ontrol of The Temperature
3.5.3. In	itluence of the Syringe
3.5.4. D	etermination of the Equilibrium State
3.5.5. R	eproducibility of the Experiments
3.5.6. E	rtor Analysis
a)) Error on the Determination of B
b) Error on the Determination of a
c)) Error on the Determination of τ
ď) Error on the Determination of $\Delta \rho$
e) Total Error on the determination of Interfacial Tension
3.5.7. C	omparison of Interfacial Tension Results

3.5.8. Drop Profile Analysis	35
3.6. Experiments with Polymers	
3.6.1. Evolution of the Pendant Drop with Time	
a) Typical Behavior	38
b) The Necking and Capillarity Effects	38
i) Necking Effect	41
ii) Capillary Effect	41
3.6.2. Experimental Results	44
3.6.3. Evolution of $\gamma(t)$	44
3.7. Conclusions - Pendant Drop Method	47
4) THE SPINNING DROP APPARATUS	47
4.1. Тhеогу	48
4.2. Spinning Drop Method Development	50
4.3. Apparatus	
4.3.1. General Description	51
4.3.2. The Experimental Assembly	52
4.3.2.1. The Motor	53
4.3.2.2. Oven and Controller	53
4.3.2.3. Shaft and Glass Tube	54
4.3.2.4. The Capacitance Probe	55
4.3.3. The Optical System	55
4.3.4. Measurement of Interfacial Tension	55
4.4. Experimental Procedures	
4.4.1. Samples	56
4.4.2. Loading Procedures	56
4.4.3. Determination of the Corrective Factor and Optical Enlargement	57
4.5. Results	
4.5.1. Determination of Optical Correction Factor and Density of the	
Matrix	58
4.5.1.1. Determination of Optical Correction Factor	58
4.5.1.2. Determination of Density with the Capacitance Probe	59
4.5.2. Evaluation of the Spinning Drop Apparatus	60

4.5.2.1. Determination of Experimental Error	60
4.5.2.2. Comparison of Data	60
4.5.3. Experiments with Polymers	
4.5.3.1. Evolution of Drop Profile	64
4.5.3.2. Spinning Drop Method and three Different Approaches	
a) Value at Equilibrium	64
b) Method of Patterson [1971]	64
c) Method of Joseph [1992]	65
4.5.3.3. Measurement of Interfacial Tension between Polymers	66
4.6. Conclusion - Spinning Drop Method	66
5) COMPARISON BETWEEN THE TWO APPARATUSES	66
6) SUMMARY	68
REFERENCES	70

1) INTRODUCTION	75
1.1 Chapter Outline	75
2) SELECTION AND PROPERTIES OF MATERIAL	. 76
2.1. Choice of Material	76
2.1.1 Pure Material	. 76
2.1.2 Blends	. 79
2.1.3 Commercial Resins	. 83
a) PP and EVOH	. 84
b) MAgPP and EVOH	84
2.2. Molecular Weight	85
2.3. Maleation Content	85
a) Fourier Transform Infrared Spectroscopy	86
b) Titration	. 87

2.4. Density
2.4.1. Density of Polypropylene
2.4.2. Density of Monodisperse and Polydisperse Polystyrene
2.4.3. Density of Blends of Polystyrene
2.4.4. Density of Polyethylene
2.4.5. Density of Ethylene Vinyl Alcohol
2.4.6. Density of Maleated Polypropylene
3) INFLUENCE OF TEMPERATURE
3.1. Introduction
3.2. Results and Discussion
4) INFLUENCE OF MOLECULAR WEIGHT
4.1. Introduction
4.2. Results and Discussion
5) INFLUENCE OF POLYDISPERSITY
5.1. Introduction
5.2. Influence of Blending: Results and Discussion
5.3. Effect of Polydispersity
a) PP/PS
b) PE/PS polymer Pair
6) EFFECT OF COMPATIBILIZERS
6.1. Background
6.1.1. Introduction
6.1.1.1. Addition of Copolymer
6.1.1.2. Functionalization
6.1.2. Influence of Compatibilizers on Blend Properties
6.1.3. Effect of Compatibilizers on Interfacial Tension
6.2. Experimental Conditions
6.3. Results
6.4. Influence of Other Additives Already Present in Commercial PP

7) SURFACE ANALYSIS OF INTERFACES	126
7.1. Objectives of Surface Analysis Studies	126
7.1.1. Analysis of Compatibilizer	126
7.1.2. Difference between Pure and Commercial PP	127
7.1.3. Observation of the Interfaces between Polymers	127
7.2. Surface Analysis of Polymers	127
7.2.1. Existing Methods for Surface Analysis	127
7.2.2. Electron Spectroscopy for Chemical Analysis (E.S.C.A)	128
7.2.2.1. Description of the Method	128
7.2.2.2. E.S.C.A. and Polymers	128
7.2.3. S.E.M.	129
7.3. E.S.C.A. Analysis	129
7.3.1. Experimental Procedures	129
7.3.2. Results and Discussion	130
7.3.2.1. Differences between Pure and Commercial PP	132
7.3.2.2. Influence of a Compatibilizer	133
i) Analysis of PP _{com} and MAgPP ₂	133
ii) Analysis of EVOH	135
7.4. S.E.M. ANALYSIS OF THE SAMPLES	137
8) CONCLUSIONS	141
8.1. Effect of Temperature	141
8.2. Effect of Molecular Weight and Molecular Weight Distribution	141
8.3. Effect of Compatibilizers	142
8.4. Surface Analysis	142
REFERENCES	145

CHAPTER VI. THEORETICAL CONSIDERATIONS

1) CHAPTER OUTLINE	149
2) THERMODYNAMICS OF INTERFACES	149
3) THEORIES OF POLYMERIC INTERFACES	151
3.1. Empirical Theories	151
3.2. Microscopic Theories of Polymeric Interfaces	153
3.2.1. Lattice Theories	153
3.2.2. Theories Based on Square Gradient	156
3.3. Effects of Compatibilizers	159
4) COMPARISON WITH THEORY	160
4.1. Evaluation of the Parameters	160
4.1.1. Evaluation of ρ_{o} , b and N	161
4.1.2. Evaluation of the Flory-Huggins Interaction Parameter	162
4.1.2.1. Definition	162
4.1.2.2. Validity of the Expression for χ	163
4.1.2.3. Values of χ used for this research	165
4.2. Comparisons for the PP/PS Polymer Pair	167
4.2.1. Temperature Influence	167
4.2.2. Molecular Weight Influence	171
4.2.3. Polydispersity Influence	176
4.3. Comparisons for the PE/PS Polymer Pair	177
4.3.1. Temperature Influence	177
4.3.2. Molecular Weight Influence	180
5) CONCLUSIONS	183
REFERENCES	185

CHAPTER VIII. CONCLUSION

1) CONCLUSIONS	189
2) RECOMMENDATIONS FOR FUTURE WORK	191
3) CONTRIBUTIONS TO KNOWLEDGE	192

APPENDIXES

A)	PROGRAMS FOR DROP ANALYSIS	A.1
B1)	INTERFACIAL TENSION DATA	B1.1
B2)	SOLUBILITY PARAMETER OF EVOH	B 2.1

LIST OF FIGURES

The Pendant Drop Geometry	10
The Spinning Drop Geometry	12
The Pendant Drop Apparatus	23
General View of the Pendant Drop Apparatus	24
Pendant Drop Apparatus: the Experimental Cell	26
Pendant Drop Apparatus: the Drop Insertion Device	27
a) Drop of Polystyrene in Polypropylene at 230°C injected with a	
Syringe with a small Capillary Diameter, Ø=0.8 mm	32
b) Drop of Polystyrene in Polypropylene at 230°C injected with a	
Syringe with a big Capillary Diameter, $\emptyset = 1.7 \text{ mm}$	32
a) Pendant Drop of Ethylene Vinyl Alcohol Copolymer in	
Polypropylene at 240°C	
- Digitized Image	36
b) Pendant Drop of Ethylene Vinyl Alcohol Copolymer in	
Polypropylene at 240°C	
- Edge Detection	36
c) Pendant Drop of Ethylene Vinyl Alcohol Copolymer in	
Polypropylene at 240°C	
- Smoothed Profile	37
d) Pendant Drop of Ethylene Vinyl Alcohol Copolymer in	
Polypropylene at 240°C	
- Superposition of the Experimental Profile (•) and the Theoretical	
Profile (-)	37
Evolution of one Drop of Ethylene Vinyl Alcohol Copolymer in	
Polypropylene at 223°C from the beginning of the Experiment to	
fourteen hours after	39
	The Pendant Drop Geometry The Spinning Drop Geometry The Pendant Drop Apparatus General View of the Pendant Drop Apparatus Pendant Drop Apparatus: the Experimental Cell Pendant Drop Apparatus: the Drop Insertion Device a) Drop of Polystyrene in Polypropylene at 230°C injected with a Syringe with a small Capillary Diameter, \emptyset =0.8 mm b) Drop of Polystyrene in Polypropylene at 230°C injected with a Syringe with a big Capillary Diameter, \emptyset =1.7 mm a) Pendant Drop of Ethylene Vinyl Alcohol Copolymer in Polypropylene at 240°C - Digitized Image b) Pendant Drop of Ethylene Vinyl Alcohol Copolymer in Polypropylene at 240°C - Edge Detection c) Pendant Drop of Ethylene Vinyl Alcohol Copolymer in Polypropylene at 240°C - Smoothed Profile d) Pendant Drop of Ethylene Vinyl Alcohol Copolymer in Polypropylene at 240°C - Superposition of the Experimental Profile (•) and the Theoretical Profile (-) Evolution of one Drop of Ethylene Vinyl Alcohol Copolymer in Polypropylene at 240°C - Superposition of the Experimental Profile (•) and the Theoretical Profile (-) Evolution of one Drop of Ethylene Vinyl Alcohol Copolymer in Polypropylene at 223°C from the beginning of the Experiment to fourteen hours after

2.10.:	Superposition of an E	xperimental Drop of E	thylene Vinyl Alcohol	
	Copolymer in Polypro	pylene at 223°C and t	he corresponding	
	theoretical drop. (•) re	epresent the Experiment	ntal Data and (-) the	
	Theoretical Data			40
	a) at $t = 0$			
	b) at $t = 2$ hours			
	c) at t = 6 hours			
	d) at t = 14 hours			
2.11.:	The "Necking Effect",	Evolution of a Drop of	of Polystyrene in	
	Polypropylene at a Te	mperature of 250°C		42
	a) at $t = 0$	c) at $t = 1.5$ hour	e) at t = 3 hours	
	b) at $t = 1$ hour	d) at $t = 2$ hours	f) at $t = 4$ hours	
2.12.:	The "Capillarity Effec	t", Evolution of a Drop	p of Polystyrene in	
	Polypropylene at a Te	mperature of 250°C	•••••••••••••••••••••••••••••••••••••••	43
	a) at t = 0	-c) at t = 2 hours	e) at $t = 4$ hours	
	b) at $t = 1.5$ hour	d) at t = 3 hours		
2.13.:	Interfacial Relaxation	Function for Polyprop	ylene in Ethylene Vinyl	
	Alcohol Copolymer at	t 223°C		45
2.14.:	General View of the S	Spinning Drop Apparat	rus	51
2,15.:	Sketch of the Spinning	g Drop Apparatus		52
2.16.:	Spinning Drop Appar	atus: the Shaft		54
2.17.:	Optical Correction Fa	ctor for Ethylene Viny	l Alcohol Copolymer	58
2.18.:	Density of Ethylene	Vinyl Alcohol Copolyn	ner	59
2.19.:	Tracings of a Spinning	g Drop of Ethylene Vi	nyl Alcohol Copolymer in	
	Polypropylene			
	a) at $t = 0$	c) at $t = 110$ minutes	e) at $t = 300$ minutes	
	b) at $t = 12$ minutes	d) at t = 210 minutes	••••••	62
2.20.:	Spinning Drops of Et	hylene Vinyl Alcohol (Copolymer in Polypropylene	
	Digitized Images			
	a) at t = 0	c) at $t = 110$ minutes	e) at t = 300 minutes	
	b) at $t = 12$ minutes	d) at t = 210 minutes		63
2.21.:	Evolution of the Inter	facial Tension of a Spi	inning Drop of Polypropylene	
	in Ethylene Vinyl Alc	ohol Copolymer at 202	2°C	64

2.22.:	: Evolution of the Radius of a Drop of Polypropylene in Ethylene Vinyl		
	Alcohol Copolymer at 202°C		65

3.1.:	Gel Permeation Chromatography Spectrum for PS ₃	80
3.2.:	Gel Permeation Chromatography Spectrum for PS _b	81
3.3.:	Gel Permeation Chromatography Spectrum for PS _c	82
3.4.:	Typical I.R. Spectrum of MAgPP	86
3.5.:	Interfacial Tension between Polypropylene and Ethylene Vinyl Alcohol	
	Copolymer as a Function of Temperature	93
3.6.:	Interfacial Tension between Polypropylene and Monodisperse	
	Polystyrene as a Function of Temperature	94
3.7.:	Interfacial Tension between Polypropylene and Polydisperse	
	Polystyrene as a Function of Temperature	95
3.8.:	Interfacial Tension between Monodisperse Polyethylene and	
	Monodisperse Polystyrene as a Function of Temperature	96
3.9.:	Interfacial Tension between Polypropylene and Monodisperse Polystyrene	
	as a Function of Molecular Weight of Polystyrene	100
3.10.:	Interfacial Tension between Monodisperse Polyethylene and	
	Monodisperse Polystyrene as a Function of Molecular Weight	
	of Polyethylene	101
3.11.:	Interfacial Tension between Polypropylene and Polystyrene and	
	Polypropylene and Polystyrene as a Function of $M_n^{-2/3}$ (M_n	
	being the Number Average Molecular Weight of Polystyrene or	
	Polyethylene respectively)	103
3.12.:	Interfacial Tension between a Bimodal Blend of Polystyrene (PS_2 and PS_4)	
	and Polypropylene as a Function of the Molar Percentage of PS_4	106
3.13.:	Interfacial Tension between a Bimodal Blend of Polystyrene (PS_3 and PS_5)	
	and Polypropylene as a Function of the Molar Percentage of PS ₅	107
	•	

3.14.:	Interfacial Tension between Monodisperse Polystyrene and Polypropylene
	and between Polydisperse Polystyrene and Polypropylene as a Function of
	Temperature
3.15.:	Interfacial Tension between Monodisperse Polyethylene and Monodisperse
	Polystyrene and between Polydisperse Polyethylene and Monodisperse
	Polystyrene
3.16.:	Addition of Block or Graft Copolymer to Immiscible Polymer Pair A-B
	[Paul (1978)]
3.17.:	In Situ Formation of Graft Copolymer as a Results of Reaction between
	OH and MAH groups
3.18.:	a) Reaction between MAH and N ₆
	b) Reaction between MAH and EVOH 117
3.19.:	Interfacial Tension between Polypropylene and Ethylene Vinyl Alcohol
	Copolymer as a Function of Maleation Level
3.20.:	Diffusion of Maleic Anhydride 123
3.21.:	X-ray Spectrum of Commercial Polypropylene 125
3.22.:	E.S.C.A. Spectrum of Ethylene Vinyl Alcohol Copolymer
3.23.:	S.E.M. Image of a Drop of Ethylene Vinyl Alcohol Copolymer in
	Commercial Polypropylene
3.24.:	S.E.M. Image of a Drop of Ethylene Vinyl Alcohol Copolymer in MAgPP ₂ 138
3.25.:	S.E.M. Image of the Interface between Ethylene Vinyl Alcohol and
	Commercial Polypropylene
3.26.:	S.E.M. Image of the Interface between Ethylene Vinyl Alcohol and MAgPP ₂ 139
3.27.:	S.E.M. Image of the Interface between Ethylene Vinyl Alcohol and MAgPP ₂
	at high optical enlargement

CHAPTER IV. THEORETICAL CONSIDERATIONS

 4.1.: Interfacial Tension between PS₅ and PP as a Function of Temperature: Comparison between the Predictions of the New Version of the Square Gradient Theory and Experimental Data - χ from Equation 4.40. 167

CHAPTER IV. THEORETICAL CONSIDERATIONS

4.2.:	Interfacial Tension between PS and PP as a Function of Temperature:	
	Comparison between the Predictions of the New Version of the Square	
	Gradient Theory and Experimental Data - χ from Table 4.2.	169
4.3.:	Interfacial Tension between PS, and PP as a Function of Temperature:	
	Comparison between the Predictions of the New Version of the Square	
	Gradient Theory, Helfand and Tagami, Helfand and Sapse, Helfand and	
	Bhattacharjee and Experimental Data - χ from Table 4.2.	170
4.4.:	Interfacial Tension between PS_3 and PP as a Function of Temperature:	
	Comparison between the Predictions of the New Version of the Square	
	Gradient Theory, Helfand and Tagami, Helfand and Sapse, Helfand and	
	Bhattacharjee and Experimental Data - χ from Table 4.2.	171
4.5.:	Interfacial Tension between PP and PS as a Function of Molecular	
	Weight of PS Comparison between the Predictions of the New Version	
	of the Square Gradient Theory and Experimental Data -	
	χ from Equation 4.40	172
4.6.:	Flory-Huggins Interaction Parameter between Polypropylene and	
	Polystyrene as a Function of Molecular Weight of Polystyrene	174
4.7.:	Interfacial Tension between PP and PS as a Function of Molecular	
	Weight of PS Comparison between the Predictions of the New	
	Version of the Square Gradient	
	Theory and Experimental Data - χ from Table 4.2	175
4.8.:	Interfacial Tension between PE and PP as a Function of Temperature:	
	Comparison between the Predictions of the New Version of the Square	
	Gradient Theory and Experimental Data - χ from Table 4.2	178
4.9.:	Interfacial Tension between PE_3 and PP as a Function of Temperature:	
	Comparison between the Predictions of the New Version of the Square	
	Gradient Theory, Helfand and Tagami, Helfand and Sapse, Helfand and	
	Bhattacharjee and Experimental Data - χ from Table 4.2.	179

CHAPTER IV. THEORETICAL CONSIDERATIONS

4.10.:	Interfacial Tension between PE_1 and PP as a Function of Temperature:
	Comparison between the Predictions of the New Version of the Square
	Gradient Theory, Helfand and Tagami, Helfand and Sapse, Helfand and
	Bhattacharjee and Experimental Data - χ from Table 4.2
4.11.:	Interfacial Tension between PE and PS as a Function of Molecular
	Weight of PE Comparison between the Predictions of the New Version
	of the Square Gradient Theory and Experimental Data - χ from Table 4.2., 181
4.12.:	Flory-Huggins Interaction Parameter between Polyethylene and Polystyrene
	as a Function of Molecular Weight of Polyethylene

LIST_OF TABLES

CHAPTER II. EXPERIMENTAL EQUIPMENT

Table 2.1. : Comparison between Measured and Reported Values	35
Table 2.2. : Interfacial Tension from Transient Measurements	. 46
Table 2.3. : Comparison between the Pendant and Spinning Drop Method	. 61

Table 3.1. : Pure Resins used in this Study	78
Table 3.2. : Blends of Polystyrene used in this Study	79
Table 3.3. : Commercial Resins used in this Study	83
Table 3.4. : Maleation Content of Maleated PP	84
Table 3.5. : Density of the Polymers used in this Study	88
Table 3.6. : Coefficients of Linear Regression of the Dependence of the Interfacial	
Tension on the Temperature $\gamma = a - bt$	97
Table 3.7. : Coefficients of Linear Regression of the Dependence of the Interfacial	
Tension on the Logarithm of the Molecular Weight $\gamma = a + b \log(M_n)$	104
Table 3.8. : Enhancement of PS/LDPE Blend Properties through Addition of Graft	
Copolymer	115
Table 3.9. : Interfacial Tension between MAgPP and EVOH	121
Table 3.10. : Interfacial Tension between Pure and Commercial Resins	.124
Table 3.11. : Results of Elemental Analysis	125
Table 3.12. : Elemental Analysis of the Samples	130
Table 3.13. : Multiplex Analysis of O for EVOH	133
Table 3.14. : C/O for MAgPP ₂	134
Table 3.15. : Multiplex Analysis of O for MAgPP ₂	134
Table 3.16. : Evolution of the C/O Ratio for EVOH	. 136
Table 3.17. : Multiplex Analysis of O for EVOH	136
Table 3.18. : Multiplex Analysis of C for EVOH	136

CHAPTER IV. THEORY

Table 4.1.: Values of the Number Monomer Density, Kuhn Statistical Segme	nt and
Solubility Parameter for the Polymers used in this Study	162
Table 4.2. : Interaction Parameter used in this Study	
Table 4.3. : Interfacial Tension Values between PP and PS_1 or PS_2 at 186°C	175



I

1. INTRODUCTION

Polymer blends have gained popularity in packaging applications, especially to replace multi-layer products. The growth in the use of polymer blends is mainly because they combine the properties of different components to result in a material with optimized mechanical strength, low permeability to water and to oxygen, optical properties etc. Other advantages of polymer blends include low capital cost, single step processing and adaptability to recycling of reground products.

When working with polymer blends, it is important to obtain at least partial compatibility between the components of the product. Polymer compatibility is an important factor in the processing of polymer blends since compatibility governs the adhesion and the condition of the interface and, therefore, the morphology and mechanical properties of the blend, i.e. the final characteristics of the blend. Interfacial tension is one of the key parameters that govern the compatibility between the components and the morphology of a polymer blend [Wu 1987, Van Oene 1972]. It is the single most accessible parameter that describes the thermodynamic state and structure of an interface.

Following Adamson [1967], interfacial tension can be defined as the reversible work required to create a unit of interfacial area at constant temperature, T, pressure, P, and number of molecules, n. In thermodynamic terms, interfacial tension may be identified as an increment in Gibbs free energy per unit increment in area [Wu 1982], and can be evaluated by the following expression:

$$\gamma = \left(\frac{\partial G}{\partial A}\right)_{T,P} \tag{1.1}$$

where G is the free energy of the system and A is the interfacial area.

Numerous studies have been carried out to determine the effect of interfacial tension on the properties of polymer blends [Wu 1987, Xanthos et al 1990...]. It has been shown that the final characteristics of polymer blends depend on the micro structure of the blend, which depends on the size of the dispersed phase and on the interfacial tension between the components of the blend.

Wu [1987] demonstrated that the dispersed-drop size, in a polymer blend consisting of a matrix and a dispersed phase, is directly proportional to the interfacial tension as follows:

$$\frac{G\eta_{m}d_{d}}{\gamma} = 4\left(\frac{\eta_{d}}{\eta_{m}}\right)^{z_{0.84}}$$
(1.2)

where G is the shear rate in the extruder, η_m is the viscosity of the matrix phase, η_d is the viscosity of the dispersed phase, γ is the interfacial tension and d_d the average diameter of the droplets in the dispersed phase.

Also interfacial tension has been shown to govern the adhesion between two phases [Wu 1982]. The work of adhesion between the different phases is given by:

$$W_{1} = \sigma_{1} + \sigma_{2} - \gamma \tag{1.3}$$

where W_{1} is the work of adhesion between the two phases, σ_{1} and σ_{2} are the values of the surface tension for the two components and γ is the interfacial tension between the two components.

Very limited data are available regarding the interfacial tension between pairs of polymers melts. Most of the methods used to measure interfacial tension are based on the shape of a drop of one polymer immersed into a second polymer called matrix until equilibrium is reached. The research has been limited because of some difficulties in the experimental techniques (inability to ascertain mechanical equilibrium, long equilibration times that would exceed the time for the melt to undergo degradation and others). Only few experimental studies are available relating to the influences of temperature and molecular weight on interfacial tension between polymers [Wu 1974, Escudie 1986, Anastasiadis 1988]. Moreover, no data have been reported so far regarding the effect of polydispersity on the interfacial tension of polymer melts.

Recent theories dealing with interfacial tension between polymers can be divided into two categories. The first are based on the lattice theory of Helfand and Tagami [1971-1972] and the second on the square gradient theory developed by Cahn and Hilliard [1958]. In their original form, both theories were unable to predict the effect of molecular weight and molecular weight dispersity on interfacial tension between polymers, because these theories were derived under the assumption that the polymers had an infinite molecular weight. Recently, the theories have been modified and it is now possible to evaluate theoretically the effect of molecular weight on interfacial tension between polymers using a new version the lattice theory [Helfand and Bhattacharjee 1989] and the effect of both molecular weight and polydispersity using a new version of the square gradient theory [Broseta et al 1990]. These theories have not been compared to experimental data yet.

Enhancement of blends of polypropylene (PP) and ethylene vinyl alcohol copolymer (EVOH) using maleic anhydride grafted polypropylene (MAgPP) as a compatibilizer has been studied by the Polymer Group at McGill University. The blends were tested for morphological structure and mechanical properties; it was suggested that the enhancement of the properties of the blends due to the addition of the compatibilizers was mainly due to the decrease of the interfacial tension between the components. It is therefore desirable to study the influence of compatibilisation on the interfacial tension between PP and EVOH.

2. OBJECTIVES

The objectives of this work were

a) To design, construct and operate an apparatus for measuring the interfacial tension between pure and commercial polymers. The apparatus should reduce or eliminate the problems associated with polymer degradation, irregular droplet shape, long equilibration times and inaccuracy of identifying the equilibrium profile.

b) To study experimentally the interfacial tension between polymer pairs that are commonly used in the polymer industry and to extend the range of data of interfacial tension between polymers.

c) To conduct an experimental and theoretical study of the influence of parameters such as temperature, molecular weight and polydispersity on interfacial tension.

d) To evaluate the influence of compatibilisation on the interfacial tension between polymers, in particular between PP and EVOH.

<u>3. ORGANIZATION OF THE THESIS</u>

The remaining part of this thesis consists of the following chapters. Chapter II describes the two instruments constructed for the determination of the interfacial tension between polymer melts. Chapter III presents and discusses the experimental results. Initially, the materials used in this study and their properties are described (molecular weight, density...). Then, the experimental data obtained regarding the influences of temperature, molecular weight and molecular weight dispersity and the effects of compatibilizers on the interfacial tension between polymers are presented. The results of the surface analysis of the interface are also reported in this Chapter. Chapter IV uses the results of Chapter III to evaluate available thermodynamic theories for the prediction of the interfacial tension between polymers. Chapter V includes a summary of the conclusions of the thesis and recommends the direction of some future work on polymer interfacial studies.

An extensive literature review was conducted during this research. This has been incorporated directly into the appropriate chapters, so that relevant results from the literature can be immediately compared to the present work. A list of the nomenclature is given at the end of the thesis.

REFERENCES

Adamson A.W. [1967]: "Physical Chemistry of Surfaces", 2nd Ed., Wiley (interscience), New York, (1967).

Anastasiadis S.H. [1988]: "Interfacial Tension of Immiscible Polymer Blends", Ph.D. Thesis, Princeton University, (1988).

Broseta D., Fredrickson G.H., Helfand E. and Leibler L. [1990]: Macromolecules, Vol 23, No 1, p 432, (1990).

Cahn J.W. and Hilliard J.E. [1958]: J. Chem. Phys., Vol 28, p258, (1958).

Escudie E. [1986] : Materials Chemistry and Physics, Vol 14, p 239, (1986).

Helfand E. and Tagami Y. [1971]: Polymer letters, Vol 9, p741, (1971).

Helfand E. and Tagami Y. [1972]: J.Chem. Phys., Vol 56, p 3592, (1972).

Helfand E. and Tagami Y. [1972]: J. Chem. Phys., Vol 57, p 1812, (1972).

Helfand E. and Bhattacharjee S.M. [1989]: J. Chem. Phys., Vol 91 (11), p 7200, (1989).

Van Oene H. [1972]: J. Colloid Interf. Sci., Vol 40, p448 (1972).

Wu S. [1974]: J. Macromol.Sci. - Revs Macromol. Chem., C10, p1 (1974).

Wu S. [1982] : "Polymer Interface and Adhesion, Marcell Dekker, Inc., NY, (1982).

Wu S. [1987] : Polymer Eng Sci., Vol 27(5), p335, (1987).

Xanthos M., Young M.W. and Biesenberger J.A. [1990]: Polym. Eng. Sci., Vol 30(6), p 355 (1990).



1. CHAPTER OUTLINE

In this chapter, a short review of interfacial tension measurement techniques is presented. The reasons for choosing the pendant drop and the spinning drop methods for measurement of interfacial tension between polymers are given. The interfacial tensiometer based on the pendant drop method is first described. The pendant drop apparatus is evaluated and tested. Following this, the spinning drop interfacial tensiometer is presented. The apparatus and experimental procedures are discussed and the spinning drop apparatus is evaluated. The main characteristics of the two instruments are compared. The last part presents the conclusions of this chapter.

2. LITERATURE REVIEW

2.1 Introduction

Some of the early interfacial tension measurements for polymeric materials were reported in 1969 [Wu 1969, Roe 1969]. Since then, different measurement techniques have been developed. Previous reviews of these techniques have been reported by Wu [1974, 1982]. Among the various available methods to measure interfacial tension, only a few are suitable for polymers because of their high viscosity.

In general, the equilibrium static methods are most commonly used. They involve the evaluation of a profile of either a sessile drop [Staicopulus 1962, 1963, 1967], or a spinning drop [Patterson et al. 1971, Elmendorp and Vos 1986] or a pendant drop [Roe 1969, Wu 1969, 1970]. Dynamic methods, based on the thread breaking method are also used to evaluate the interfacial tension between polymers [Chappelear 1964, Carriere and Cohen 1989]. Surface light scattering methods [Sauer 1987, Jon 1986] have been suggested recently and are now being further investigated. Grassmespacher and Meissner [1992] and Graebling [1991] developed a theory based on rheological measurements to infer the interfacial tension between components of a polymer blend. These methods are discussed in more detail in the next section.

Various other methods have been proposed for the measurement of surface and interfacial tension, among them the capillary rise technique [Shonhorn 1966, Edward 1968, Hartford 1969], Wilhelmy plaque [Dettre and Johnson, 1966], Du Nouy ring [Du Nouy 1919, Newman 1958, Shonhorn 1965], the drop weight methods [Padday 1969] and the maximum bubble pressure [Edwards 1968]. Unfortunately, these methods cannot be used to measure interfacial tension between polymers because of their high viscosity.

2.2. Static Methods

2.2.1. The Pendant Drop Method

The pendant drop method is probably the most convenient, versatile and popular method used to measure interfacial tension [Roe 1967, 1969]. It has been used extensively for the determination of surface and interfacial tension between polymers [Dettre 1966, Wu 1970, 1971, Roe 1967, 1969, 1970].

The pendant drop method involves the determination of the profile of a drop of one liquid suspended in another liquid. The profile of a drop of liquid suspended in another denser liquid at equilibrium is determined by the balance between gravity (or buoyancy force in this case) and surface forces. The equation of Bashforth and Adams [1892] which is based on Lapiace's equation [Bashforth and Adams, 1892], relates the drop profile to the interfacial tension through a nonlinear differential equation which is given below:

$$2 + B\frac{z}{a} = \frac{1}{\frac{R_1}{a}} + \frac{\sin \Phi}{\frac{x}{a}}$$
(2.1a)

$$B = \frac{a^2 \Delta \rho g}{\gamma}$$
(2.1b)

where $\Delta \rho$ is the difference between the densities of the two polymers in contact, g is the acceleration due to gravity, γ is the interfacial tension, a is the radius of curvature at the apex of the drop, x, z, Φ , are the coordinates defined as in Figure 2.1, and R₁ is the radius of curvature at the point with coordinates (x,z).



Figure 2.1 : The Pendant Drop Geometry
The pendant drop method has several advantages: it is an absolute method, the interface is not disturbed during measurements, the method is independent of the contact angle and the experimental set-up is simple. Recent improvements in the pendant drop method relate to computerization for data acquisition and analysis. This will be discussed in a later section.

On the other hand, the pendant drop method presents the following potential problems. It requires the knowledge of the density of the materials used; such information is scarcely reported for polymeric materials. Also, it is difficult to apply the method when the densities of the two phases differ by less than 5%, due to potential detachment and necking of the drop [Demarquette and Kamal 1993]. Other limitations relate to the very slow approach to equilibrium with polymeric systems and the requirement that the material with the smaller density has to be transparent.

2.2.2. The Sessile Drop Method

The sessile drop method is very similar to the pendant drop method. A drop of one liquid rests on a flat plate which surrounded by another liquid of different density. The shape of the drop is determined by the balance between gravity (or buoyancy forces) and surface forces. The Bashforth and Adams equation [1882] relates the drop profile to the interfacial tension. The form of the equation is very similar to the one for the pendant drop method except for a change in the sign of the gravitational term.

The sessile drop has an advantage over the pendant drop, since the detachment of the drop is eliminated. However, the time to reach equilibrium is much longer due to the larger areas of solid-liquid contact [Wu 1974].

The sessile drop method has been less widely employed than the pendant drop. It was used by Sakai [1965] to measure the surface tension of polyethylene (PE) and by Oda and Hata [1968] to measure interfacial tension between other polymer melts. Few methods to solve the Bashforth and Adams equation (in order to evaluate the interfacial tension from the shape of a sessile drop) have been proposed [Staicopulus 1962, 1963, 1967, Buttler and Bloom 1966, Maze and Burnet 1969, 1971].

Lau and Burns [1972] used a combination of pendant drop and sessile drop for measurement of surface tension of Polystyrene (PS). The advantage of this combined method is that no air bubble is entrapped when the sessile drop is formed.

2.2.3. The Spinning Drop Method

The deformation of a spinning drop of one liquid embedded in another liquid has been suggested as a method to measure interfacial tension many years ago [Vonnegut 1942].

The spinning drop method consists of inserting a drop of a lighter phase into a denser phase in a horizontal tube. The tube is rotated at a known speed about the horizontal axis. Under the centrifugal acceleration, the lighter drop is squeezed forming an elongated tube (see Figure 2.2). The final dimensions of this elongated drop are a function of the speed of the tube, the difference of density between the two phases and the interfacial tension between the two phases. Therefore, knowledge of the speed of the tube and the difference between the densities of the two phases allows one to infer interfacial tension.





At high speeds, when the length of the drop exceeds four times its diameter (see details in Section 4.1.), a simple relationship between the diameter of the drop, d, and the interfacial tension γ exists:

$$\gamma = \frac{\Delta \rho \omega^2 d^3}{32} \tag{2.2}$$

where $\Delta \rho$ is the density difference between the two phases and ω is the angular speed of the tube.

Patterson et al. [1971] were the first to use the spinning drop method to determine the interfacial tension between polymers melts. They determined the interfacial tension between polyisoburylene and poly(dimethylsiloxane). The method was further developed by Elmendorp [1986]. Verdier [1990] used the spinning drop method to determine the interfacial tension between PE and PS, polymethylmetacrylate (PMMA) and PS, and PMMA and PE in the temperature range from 200°C to 250°C. Joseph [1992] developed a theory to understand the relaxation of the spinning drop.

The spinning drop method is more complicated than the pendant or sessile drop method because of the rotating mechanism, as well as the measurement of a rotating drop profile. However, it has the advantage of requiring less complex calculations and less time for reaching mechanical equilibrium.

2.3. Dynamic Methods

2.3.1. The Breaking Thread Method

The breaking thread method involves a fluid thread embedded in another fluid and subjected to a perturbation of wavelength superior to the circumference of the thread. The perturbation grows with time according to the following equation:

$$\alpha = \alpha_{o} \exp(qt) \tag{2.3a}$$

where α and α_0 are the amplitudes at time t and t=0 respectively and q is the rate constant defined as

$$q = \frac{\gamma \Omega}{\eta_m d_o}$$
(2.3b)

where η_m is the matrix viscosity, γ is the interfacial tension, d_o is the fiber diameter and Ω is a function that depends on the wavelength and the ratio of matrix and fiber viscosity [Tomokita 1938].

To obtain the interfacial tension, the logarithm of α is plotted as a function of time. The rate constant is calculated from the slope of the graph and the interfacial tension is obtained from Equation 2.3b.

This method has the advantage of not requiring the knowledge of $\Delta \rho$. Also the interfacial tension between materials with the same density can be measured. However, the main difficulty is in the need to know the zero-shear stress viscosities of the polymers.

The method has been used to measure the interfacial tension between polymeric materials [Chappelar 1964, Elemans et Janssen 1990], but a probable error of 30% was reported, which makes this method inaccurate.

2.3.2 Short Fiber Retraction

When a short fiber of one fluid is embedded in another fluid the fiber contracts into a sphere [Lord Raleigh 1879, 1892, 1899]. This, in essence, describes, the short fiber retraction method. The retraction time t_r of the fiber can be expressed as:

$$t_{r} = \lambda \ln \{ [(L_{o} / R_{o})^{3} - 1] / [6(L_{o} / R_{o}) - 5] \}^{3}$$
(2.4a)

$$\lambda = \frac{6R_{a}\eta}{\gamma}$$
(2.4b)

14

where R_o and L_o are the initial radius and length of the cylindrical fiber respectively, η is the effective viscosity and γ is the interfacial tension.

Carriere and Cohen [1991] used this method to measure the interfacial tension between high molecular weight polycarbonate and polymethyl methacrylate resins but reported a possible error of 26%.

2.3.3. The Surface Light-Scattering Method

The method is based on the fact that capillary waves on an interface have a dispersive velocity distribution and an exponentially damped amplitude. In this method, an electro-capillary wave is generated on the surface, and an incident beam of light is scattered by the capillary waves at the interface. The power spectrum of the scattered light depends on the densities and viscosities of the two phases and also on the interfacial tension between them [Jon, 1986].

This method has the advantage of being a non-invasive method, since the interface is stationary and planar. It also allows the precise control of the concentration of the additives used with polymers. However, it is limited to low viscosity materials, because in the case of high viscosity, the spatial damping electro-capillary wave is too rapid to be detected.

This method has been used to determine the interfacial tension between solutions of polymers [Sauer, 1987].

2.3.4. Interfacial Tension Inferred from Rheological Measurements

The interfacial tension between polymers affects the rheological characteristics of polymer blends. When blends of polymers are submitted to small amplitude oscillatory shear, they show elastic behavior at low frequency. This has been proven to be caused by the interfacial tension between phases of the blend. Grassmespacher and Meissner [1992] and Graebling and Muller [1991] developed a model to evaluate the influence of the interfacial tension on the storage and loss moduli. They showed that they could infer the

interfacial tension from the weighted relaxation spectrum of the blend. This new method has been tested successfully only for PS/PMMA blends (in the case of Grassmespacher and Meissner) and for PS/ poly(oxyethylene-diol) (POE-PO) blends (in the case of Graebling and Muller). It remains to be seen if this method can be applied to other blends. If so, it is of practical interest to study the influence of compatibilizers on polymer blends. Also, it is a very simple method. It should be pointed out, however, that rheological characteristics of blend could depend on the morphology (i.e. dispersed phase dimensions). Therefore, the results of the rheological measurements may not provide a unique value of interfacial tension.

2.4. Choice of Methods

In order to study the interfacial tension between polymer blends, two different methods were chosen: the pendant drop and the spinning drop. For polymer melts which do not undergo degradation, the pendant drop method can be used to obtain very accurate interfacial tension data. On the other hand, for polymer melts for which the degradation is important, the spinning drop method or accelerated spinning drop can be used (as described in Section 4). Also, with one or the other instrument, it is possible to measure the interfacial tension, even if one of the polymers is opaque. If this polymer constitutes the lighter phase, the spinning drop is used; whereas if it constitutes the heavier phase, the pendant drop is used. In that sense, the two methods are complementary.

<u>3. THE PENDANT DROP APPARATUS</u>

3.1. Theory

The theory of the pendant drop method is based on the balance between gravitational (or buoyancy) and surface forces. The mechanical equilibrium of a drop of one liquid suspended in another, from which it is separated by an interface, is given by a modified form of Laplace's equation, which includes the effect of gravity. Starting from Laplace's equation [Bashforth and Adams, 1882], one has:

$$\frac{1}{R_1} + \frac{1}{R_2} = \frac{\Delta P}{\gamma}$$
(2.5)

where R_1 is the radius of curvature in the plane of Figure 2-1, R_2 is the radius of curvature in a plane perpendicular to Figure 2.1, ΔP is the pressure difference across the curved interface and γ is the interfacial tension.

If Φ is the angle between the radius of curvature R_1 and the Z axis, and x is the abscissa, then the radius of curvature in the plane perpendicular to the plane of Figure 2.1, R_2 , is given by:



Figure 2.1 : The Pendant Drop Geometry

The pressure ΔP can be expressed as:

$$\Delta \mathbf{P} = \Delta \mathbf{P}_{\mathbf{a}} + \mathbf{g} \mathbf{z} \Delta \mathbf{p} \tag{2.7}$$

where $\Delta \rho$ is the density difference between the two fluids, g is the gravitational acceleration, and z is the ordinate along the Z-axis.

 ΔP_0 is determined when $\Phi = 0$, z=0, R₁ = R₂ = a, a being the radius of curvature at the apex of the drop. Thus, Equation 2.5 gives:

$$\Delta P_0 = \frac{2\gamma}{a} \tag{2.8}$$

Equation 2.5 can be written as:

$$\frac{\frac{1}{R_1}}{\frac{1}{a}} + \frac{\sin \Phi}{\frac{x}{a}} = B\frac{z}{a} + 2$$
(2.9)

where the dimensionless quantity B is given by:

$$B = \frac{a^2 g \Delta \rho}{\gamma}$$
(2.10)

Equation 2.9 constitutes the Bashforth and Adams [1882] equation. R_i and Φ can be obtained from geometric considerations and are given below:

$$R_{1} = \frac{ds}{d\Phi} = \frac{\left\{1 + \left(\frac{dz}{dx}\right)^{2}\right\}^{\frac{2}{2}}}{\frac{d^{2}z}{d^{2}x}}$$
(2.11)

$$\sin \Phi = \frac{\frac{dz}{dx}}{\{1 + (\frac{dz}{dx})^2\}^{\frac{1}{2}}}$$
(2.12)

where s is the curviligne coordinate.

In dimensionless form, the Bashforth and Adams equation can be written as:

$$\frac{\mathrm{d}\Phi}{\mathrm{d}S} = \frac{2}{\mathrm{B}} + Z - \frac{\sin\Phi}{\mathrm{X}} \tag{2.13a}$$

with

$$\frac{\mathrm{dX}}{\mathrm{dS}} = \cos\Phi \tag{2.13b}$$

$$\frac{dZ}{dS} = \sin \Phi$$
 (2.13c)

$$X(0) = Z(0) = \Phi(0) = 0$$
 (2.13d)

with the dimensionless parameters defined as follows:

$$X = xc^{\frac{1}{2}}$$
 (2.14a)

$$Z = zc^{\frac{5}{2}}$$
 (2.14b)

$$S = sc^{2}$$
(2.14c)

$$B = ac^{\overline{2}} = a(\frac{\Delta \rho g}{\gamma})$$
 (2.14d)

$$c = \left(\frac{\Delta \rho g}{\gamma}\right)^{\frac{1}{2}}$$
(2.14e)

The Bashforth and Adams equation is a nonlinear differential equation between x and z relating the drop profile to interfacial tension. In the Bashforth and Adams equation, R_1 , x, a and z appear as ratios of the radius of curvature at the apex. Therefore, for the same value of B, the drop keeps the same shape, irrespective of its volume [Buttler 1966].

3.2. Evolution of the Pendant Drop Method

In 1882, Bashforth and Adams [1882] derived the theoretical form of a sessile or pendant drop and calculated tables of drop contours. These tables can be used to determine the interfacial tension by fitting the experimentally measured drop contour to the theoretical curve. Photographs of the evolving drop could be taken as a function of time for comparison. However, this procedure is very tedious. To simplify this procedure, the following empirical relationship was proposed by Andreas [1938].

$$\gamma = \frac{gD_c^2 \Delta \rho}{H}$$
(2.15)

where γ is the interfacial tension, $\Delta \rho$ is the density difference, D_e is the equatorial diameter of the drop, H is a correction factor which is related to the shape factor of the pendant drop, S, defined as:

$$S = \frac{D_*}{D_e}$$
(2.16)

where D_{n} is the drop diameter measured horizontally at a distance D_{e} away from the vertex of the drop. Stauffer [1965] and Fordham [1948] obtained the values of H by solving the Bashforth and Adams equation. The above techniques have been discussed by Adamson [1967]. A more elaborate method was proposed by Roe et al [1967]. It involves a series of S values S_{n} , where:

$$S_n = \frac{D_n}{D_e}$$
(2.17)

where D_n is the horizontal drop diameter measured at a distance of $D_{ex}\frac{n}{10}$, (with n being an integer $1 \le n \le 10$) from the vertex of the drop.

Unfortunately, the above methods use only few measurements to define the entire shape of the drop, leading to imprecision in the comparison between the experimental profile and the numerical solution of the Bashforth and Adams equation. The error involved in using these methods was analyzed by Roe [1967]. Another problem encountered in using the above methods of comparison is in the determination of the instant at which the mechanical equilibrium of the drop is reached. As an example, Wu

[1974] recommends that the variation of $\frac{1}{H}$ in Equation 2.15 should be less than 0.5% over a period of fifteen minutes.

Recent progress in image analysis and data acquisition systems has made it possible to obtain a direct digitization of the drop image with the aid of a video frame grabber or digital camera [Girault 1982, 1984]. The digital signals are analyzed using different algorithms to determine the interfacial tension from the drop profile [Rotenberg 1983, Anastasiadis 1987]. Girault et al [1982] used a least square comparison procedure to analyze the profile of a pendant drop. The parameters B (Equation 2.10) and a (distance from the apex to the center of the drop) are first calculated using the polynomial approximation of Fordham's table. Then, using these values, the Bashforth and Adams equation is integrated. The digital and numerical profiles are compared, changing the values of B and a in order to optimize the final profile. The drawback of this algorithm is that the optimization is done on the two parameters separately. Also, the optimization procedure is not very effective (least square optimization).

In another work [1984] Girault et al use the Laplace equation at the inflection plane of the drop, located by a polynomial technique. However, this method has the disadvantages of requiring a drop long enough to have an inflection point and of requiring a perfectly symmetric drop. Also, it places emphasis on the points located between the capillary tube and the equator.

Huh and Reed [1983] developed an algorithm to compare theoretical and experimental profiles of the pendant drop, using a multiple regression routine. The applicability of their method was questioned in the cases of large and flat drops [Boyce and Al, 1984]. Also, the optimization involved only a and B, although an accurate optimization should be based on five parameters.

Rotenberg [1983] developed an algorithm to determine interfacial tension from the shape of the drop. The algorithm is based on the optimization of five parameters as

suggested by Huh and Reed [1983], and it could be used in the cases of large and flat drops. This algorithm has been used for the determination of interfacial tension between polymers. However, problems have been encountered in its use [Anastasiadis, 1987]. The program did not converge to reasonable solutions. Another drawback of this algorithm is the long computational time.

Anastasiadis et al [1987] developed an algorithm that uses the shape comparison routines of Siegel [1982a, 1982b]. These routines are resistant to outlying points that may be encountered in case of noisy digitized signals. The programs showed satisfactory performance in the evaluation of the interfacial tension of polymers. This algorithm is the one used in the present research for the analysis of the pendant drops. It is described in Section 3.6.

In the present work, an automatic digitization procedure was used. The drop was then analyzed using a shape comparison, as in the work of Anastasiadis [1987]. The digitization and drop shape analysis is rapid and requires less than one minute of computation time (using a 80486, 33 MHz computer) to infer the interfacial tension value from the image of the drop.

Cheng et al [1990] and Skinner et al [1989] developed an automatic procedure to measure the evolution of contact angles and interfacial tension (from drop shape analysis) as a function of time. After automatic digitization of the drop image they used the Rotenberg [1983] algorithm to infer the interfacial tension from their drop profile. Although the method of Cheng is similar to the one employed here, it is more complicated (it uses adaptive thresholding as described in Section 3.3.5.). This procedure has been successfully used to study protein solutions [Voigt et al 1991, Miller et al 1993].

3.3. Pendant Drop Apparatus

3.3.1. General Description

The apparatus developed for this research consists of a heated sample holder in which the pendant drop is formed, an optical system to capture the image of the drop, and a data acquisition system with a PC computer equipped with an Intel 80486 processor to compute the interfacial tension from the drop profile. It has been described in Demarquette and Kamal [1992]. Figure 2.3 shows the different parts of the apparatus. A general view of the apparatus is given in Figure 2.4.



Figure 2.3: The Pendant Drop Apparatus



Figure 2.4: General View of the Pendant Drop Apparatus

3.3.2. The Optical System

A xenon lamp is used for illumination of the drop (fiber optic light source from Klinger Inc.). The light is conducted by an optical fiber to the target area. The wavelength of the light source can be modified using filters (in cases when the drop and the surrounding medium have a refractive index difference lower than 1%). The lighting is uniform, allowing a global thresholding for digitization of the drop as described in Section 3.5.5.

The visualization system consists of a mono-zoom body (M4C from Klinger Inc.), a straight tube for video camera adaptation, an optical objective x 1.5 and a compact color CCD video camera (pulnix CCD camera). The mono-zoom body has a magnification factor from 1 to 4. The tube rectifies the image and has a zero angle between the observation plane and the optical axis.

The light source and video camera are assembled on a XYZ support. The whole system is mounted and aligned on a vibration proof table (T500 from Klinger inc.). It is very important to perform the experiments on a vibration-proof table (Patterson [1971]) to avoid the detachment of the pendant drop. Alignment is very important to avoid any distortion of the digitized image. A commercial stage micrometer (100x0.01 mm) square reticule (from Klinger Inc.) was used in order to verify if the image has any distortion and to measure the optical enlargement.

3.3.3. The Experimental Cell

The sample holder for the pendant drop consists of an electrically heated, hollow, cylindrical stainless steel chamber (Environmental chamber, modified unit 100-07-(15) from Rame-Hart), in which a hollow copper block is inserted (see Figure 2.5). A standard 10 mm path length (45 mm height) adsorption cell is placed in this sample holder and forms the sample cell. Two quartz windows, cut through the mid-plane of the cylinder, permit illuminatior, and viewing of the drop. A proportional temperature controller with a precision of $\pm 0.5^{\circ}$ C (proportional temperature controller with thermocouple 100-50 from Rame-Hart) is used to maintain the sample at a temperature up to 300° C.

It is very important to maintain a neutral gas atmosphere to avoid thermal degradation of the polymer. In this work, an argon atmosphere was used in the sample chamber. The outlet of the chamber has a valve that is closed after the initial air inside is purged and replaced by argon.



Figure 2.5: Pendant Drop Apparatus: the Experimental Cell

3.3.4. The Drop Insertion Device

The drop insertion device consists of a specially designed syringe system to avoid problems encountered by other researchers (such as the necking effect described in Section 3.5.3).

The syringe is heated at the same temperature as the sample chamber. The body of the syringe is held in place by a stainless steel support (elevated temperature syringe head 100-11/02-115 from Rame-Hart) which is also heated and its temperature is maintained by another temperature controller (proportional temperature controller 100-50 from Rame-hart).

The syringe consists of a stainless steel tube, 1.7 mm in diameter, welded inside the body of the syringe. Inside the capillary tube, one polymer is melted and pushed using a plunger into the second polymer that is already melted in the sample cell. The movement of the plunger is precisely controlled by a threaded screw mounted on the top of the plunger. In order to avoid problems such as capillary effects, the plunger has been specially developed. It has at its tip a small ring of resilient material (rylon) to seal the capillary tube, once the polymer has been extruded. Figure 2.6 shows the syringe used in the research. The capillary tube must be straight along its whole length in order to avoid any asymmetry of the drop. This effect of asymmetry can still be corrected if one finds the axis of symmetry when analyzing the drop. However, this is not a trivial task [Anastasiadis 1988]. The straightness of the tube was checked with a plumb bob. Another syringe with a capillary tube of 0.8 mm diameter was used to study the effect of the volume of the drop on the interfacial tension.



Figure 2.6: Pendant Drop Apparatus: the Drop Insertion Device

3.3.5. The Measurement of Interfacial Tension

Using the above described system (sample holder, syringe, light source, visualization system) an image of the drop was obtained and digitized by a frame grabber resident in a 80486 computer. The frame grabber had a resolution of 480x640 pixels. The different programs used in the calculation of the interfacial tension from this drop image are discussed below. Images of the drop can be taken at a frequency of 1 image per 10 seconds and processed on line for determination of interfacial tension. The digitization of the image is done by global thresholding. The uniform gray level background of the image of the drop (because of the choice of the illumination system) allows global thresholding that requires much less computation time than adaptive thresholding used by other researchers (Cheng [1990]). The whole process, including digitization and computations, requires less than one minute (computing time) for each image.

3.3.6. Computer Programs

A batch file presented in Appendix A1 gives the sequence of the programs used for the determination of the interfacial tension. The whole sequence of programs can be run in less than one minute (with a 486 microcomputer) giving an effective on-line computation of interfacial tension. The commands of the batch file are described in Appendix A1. Only the main programs are discussed bellow. The listings of all the programs written in C language can be found in Appendix A1. To write the programs, an image analysis library (SII from Coreco) was used.

The drop analysis consists first of a digitization of the drop, followed by an edge detection and a smoothing of the detected edge. Once smoothed, the experimental contour is compared to the solution of the Bashforth and Adams equation using a robust shape comparison. The value of the interfacial tension is estimated and the experimental profile and optimized solutions of the Bashforth and Adams equation are displayed on the monitor (for visual comparison of the experimental and optimized solutions).

a) Edge Detection

An edge detection program is used after the digitization of the image of the drop to obtain its contour. The algorithm used scans the image in a raster fashion to detect the contours of the objects. The program calculates the areas of the different objects and obtains the contour of the largest object which is the drop Subsequently, the contour of the syringe is eliminated from the image.

The contour of the drop is defined by the last pixel with a gray level higher than the threshold value. Some researchers (Girault [1984]) have shown that the interfacial tension value is affected by less than 1% if the contour of the drop is chosen based on the last black (above the threshold) instead of the first white (below the threshold) pixel.

b) Smoothing of the Drop

A smoothing program is needed due to the finite resolution of the frame grabber. The smoothing routine makes use of local polynomial regression methods. The smoothing is done piece wise (i.e., point by point replacement) along the whole profile of the drop. The method and algorithm were first used by Anastasiadis [1987]. The description and calculations of the smoothing program are given in Appendix A1.

c) Shape Comparison

This program makes a comparison of the shapes using the experimental points after smoothing and the theoretical points found by solving the Bashforth and Adams equation by a fourth order Runge-Kutta method. The Bashforth and Adams equation is first solved for a value of B (Equation 2.10) approximated by the empirical formula of Huh and Reed [1983] or given by the user. The empirical formula of Huh and Reed is based on the following equation:

$$B = \{\exp(-6.70905 + 15.30025S - 16.44090S^{2} + 9.92425S^{3} - 2.58504S^{4})\}^{\frac{1}{2}} (2.18)$$

where S is the ratio $\frac{D_e}{D_s}$, D_e is the equatorial diameter of the drop and D_s is the diameter measured horizontally at a distance D_e from the apex of the drop (c.f. Figure 2.1).

A robust shape comparison between the experimental and the theoretical profiles is then performed. The robust shape comparison consists of an optimization on five parameters: three parameters for alignment of the imaging system to the coordinate system of the dimensionless drop (an x translation, a z translation and a rotation), one parameter for the magnification factor of the drop and one parameter for the scaling factor B. The optimal value of B is then obtained, and the interfacial tension is calculated from the following relationship:

$$\gamma = \frac{ga^2 \Delta \rho}{B}$$
(2.19)

where B is the dimensionless quantity defined in Equation 2.10, $\Delta \rho$ is the density difference between the two fluids, g is the gravitational acceleration and a is the radius of curvature at the apex of the drop.

More details about the calculations used to perform the shape comparison are presented in Appendix A1. This method has been developed by Siegel [1982a, 1982b] for the shape comparison of skulls of primates. Anastasiadis [1986] used it for drop shape comparison to measure interfacial tension between polymers.

The robust method was preferred instead of the simple least-square method because, contrary to the latter, it is not strongly influenced by atypical data or occasional incorrect values [Siegel (1982)].

3.4. Experimental Procedures

3.4.1. Preparation of the Polymeric Material

Two ways of preparing the polymeric material were used, depending on whether pure laboratory samples or commercial polymers were used (see Chapter 3, Section 1 for the description of the materials). In the case of pure laboratory polymers, the specimens were melted in place (in the experimental cell for the lighter phase polymer and in the syringe for the heavier phase) in a vacuum oven and used immediately afterwards for experiments. Samples of commercial polymers (which are obtained in pellet forms) were first dried for 24 hours. The heavier phase was shaped by extrusion in a capillary rheometer with a die diameter of 0.8 mm. The polymer with the lowest density was compression molded and then cut to the dimensions of the experimental cell. After cutting, the samples were dried for 24 hours at 75°C in order to avoid bubbles during experiments and inserted into the cell or syringe.

3.4.2. Experimental Procedure

The syringe containing the heavy phase was assembled in the controlled temperature support. The syringe (with the sample) and the cell (with the matrix) were heated up to the desired temperature. The experiment started 15 minutes after thermal equilibrium was reached. The syringe was then pushed into the other melt (inside the sample cell) until the tip of the syringe reached the level of the optical path. The drop of the polymer with the heavier phase was then gently extruded into the matrix after 15 minutes. This interval before extrusion minimizes any possible temperature difference between the polymers.

3.5. Evaluation of the Apparatus

3.5.1. Optical Distortion

The image of the drop was verified for optical distortion using perfectly parallel lines of a stage micrometer. Different magnification factors of the camera were used. The optical distortion was less than 0.3% for any magnification. It was also verified that no lens effect existed when the sample was placed in the cell.

3.5.2. Control of the Temperature

The temperature was maintained constant by two temperature controllers within $\pm 0.5^{\circ}$ C. The temperature of the melt was measured immediately after every experiment, using a thermocouple. It was observed that the temperature stayed constant during the experiments. Also, the temperature was measured at different locations in the experimental cell and no temperature gradient was found.

3.5.3. Influence of the Syringe

Determination of the interfacial tension between PP and PS at 230°C was carried out using two capⁱllary tubes with different diameters (the tubes were described in Section 3.3.4). Figures 2.7a and 2.7b show drops of PS in PP injected with the two syringes. The two experiments gave the same interfacial tension within 3% (which is within the experimental error), showing that:

(i) The diameter of the capillary tube has no influence on the determination of the interfacial tension. This has been shown experimentally by Stauffer [1969] and theoretically by Patterson [1971];

(ii) The size of the drop has no influence on the determination of the interfacial tension. These conclusions are supported by the theoretical analysis of Bashforth and Adams.



a) Small capillary Diameter \emptyset = 0.8 mm b) Large Capillary Diameter \emptyset = 1.7 mm

Figure 2.7. Drop of Polystyrene in Polypropylene at 230°C

3.5.4. Determination of the Equilibrium State

During experiments, measurements were taken every fifteen minute, until the mechanical equilibrium of the drop was reached. This was determined when two consecutive measurements of interfacial tension from the drop profile varied by less than 2%. Thereafter, measurements were taken every ten minute for two hours; the data were averaged and reported as the equilibrium interfacial tension value. A method to determine the interfacial tension (at equilibrium) from transient data is presented in Section 3.6.3.

3.5.5. Reproducibility of the Experiments

The reproducibility of the experiments was found to be better than 4% as can be seen from the results presented in Chapter 3.

3.5.6 Error Analysis

Interfacial tension is given by:

$$\gamma = \frac{ga^2 \Delta \rho}{B} = \frac{ga_x^2 \tau^2 \Delta \rho}{B}$$
(2.20)

where γ is the interfacial tension, a is the distance from the apex of the drop to the center of the drop in cm, a_{τ} is the same distance in pixels, τ is the optical magnification factor, $\Delta \rho$ is the density difference between the two fluids, g is the gravitational constant and B is the parameter defined by Equation 2.10 corresponding to the best fit between the theoretical and experimental curve.

a) Error in the determination of B

The optimal value of B was obtained by the robust shape comparison. The step of optimization between two consecutive B values was 0.005. Three types of error in the estimation of B were evaluated by the program (see Appendix A1). The best value of B was obtained for the smallest error of each kind. For the three types of error, the values of B for the same measurement differ by less than 0.1%.

b) Error in the determination of a

The error in the determination of a was equivalent 1 pixel in 300, i.e., 0.3%.

c) Error in the determination of τ

The error in the determination of τ was the same as for a, i.e., 0.3%.

d) Error in the determination of $\Delta\rho$

The determination of the density of polymer melts is not an easy task. The densities of the polymer, in this work, were determined either by calculation using an Equation of State or measured with a capillary rheometer, or still using a capacitance probe device described in the spinning drop section. Depending on the method used, the error in the determination of density could vary from 0.5% to 2%. The methods used for each polymer are reported in Chapter 3.

e) Total error in the determination of interfacial tension

The experimental error in the determination of interfacial tension is the sum of the errors reported above plus the one related to the accuracy of the equilibrium state. The

maximum relative error,
$$\frac{\Delta \gamma}{\gamma}$$
 is:
 $\frac{\Delta \gamma}{\gamma} = 0.001 + 2 \times 0.003 + 2 \times 0.003 + 2 \times 0.02 = 0.073$ (2.21)

Therefore, the maximum experimental error involved in the measurement of interfacial tension is 7.3%. It should be considered that

(a) most of the uncertainty in the determination of the interfacial tension is due to the uncertainty in the determination of the polymer densities;

(b) the maximum experimental error obtained in this work is smaller than previously reported by other researchers [Sammler 1992].

3.5.7 Comparison of Interfacial Tension Results

Three systems were studied in order to test the validity of the method/apparatus:

(i) surface tension of glycerin (i.e. the interfacial tension between glycerin and air)

(ii) interfacial tension between n-hexane and water

(iii) interfacial tension between n-octane and water

The physical constants for these materials are well documented and the materials are readily available. Table 2.1 summarizes the experimental values found with the present method and the values reported in the literature.

Materials	Δρ(g/cm ³)	Values of the present work (dyne/cm)	Literature Values (dyne/cm) Perry's [1973] and Handbook of chemistry and physics [1983]
Glycerine/Air	1.17	62.2±0.2 at 23°C	63.4 at 20°C
n-Hexane/Water	0.34	49.6±1.0 at 22°C	51.1 at 20°C
n-Octane/Water	0.31	48.4±1.0 at 22°C	49.6 at 20°C

 Table 2.1 : Comparison between measured and reported values

The comparison of the measured and reported values of surface tension of glycerin and interfacial tension of n-hexane/water and n-octane/water indicates the accuracy of the experimental and computational systems (including the frame grabber). The differences observed between the experimental and reported values are probably due to the temperature differences between this work and available data.

The apparatus was evaluated for measurements between polymers. The interfacial tension data between polymers obtained with the pendant drop method were compared with the values obtained with the spinning drop device. These results will be discussed in Section 5.

3.5.8 Drop Profile Analysis

Figures 2.8a to 2.8d show the typical processing sequence of a drop using the system described previously. In this case, a drop of ethylene vinyl alcohol copolymer (EVOH) in polypropylene (PP) at 240°C is shown. Figure 2.8a shows the digitized image as it appears on the video monitor, once the image is digitized by the frame grabber. Figure 2.8b shows the same drop after the edge detection programs. Figure 2.8c shows the drop after the profile has been smoothed. Figure 2.8d gives the superposition of the

experimental data (points) and the theoretical profile (line) obtained using the Bashforth and Adams equation and the shape comparison routine. Only half of the drop is shown here since the drop is symmetrical. It should be noted that the scale on Figures 2.8b, 2.8c, and 2.8d is different from the one of 2.8a.



a) Digitized Image



b) Edge Detection



d) Superposition of the Experimental Profile (•) and the Theoretical Profile (-)

Figure 2.8: Pendant drop of EVOH in PP at 240°C

3.6 Experiments with Polymers

Three aspects of the experiments with polymers are examined here:

a) the evolution of a pendant drop of one polymer into another,

b) the interfacial tension results between polymers and

c) the evolution of the interfacial tension as a function of time

It should be noted that, in this section, an example of only one polymer pair is shown, the complete set of results is given in Chapter 3.

3.6.1. Evolution of the pendant drop with time

a) Typical Behavior

Figure 2.9 shows the evolution of a drop of ethylene vinyl alcohol copolymer (EVOH) in polypropylene (PP) at 223°C, at different times during the experiment. This is the typical evolution of a pendant drop of a polymer into another. Figure 2.10 gives the superposition of the theoretical and experimental drops at the beginning of the experiment, two hours, six hours and fourteen hours after starting the experiment. At the beginning, the experimental drop and theoretical profile do not compare very well. When the drop reaches equilibrium, the matching of the theoretical drop (obtained from the Bashforth and Adams equation) with the experimental data is very good.

b) The necking and capillarity effects

In some cases (for example in the case of experiments with polystyrene (PS) and polypropylene (PP)), experimental difficulties were observed. If the drop (polymer with the higher density) was smaller than a certain critical volume, the drop would retract into the syringe (capillary effect). If the drop (polymer with higher density) was larger than this critical volume, it would neck and detach.



Figure 2.9: Evolution of one Drop of EVOH in PP at 223°C from the Beginning of the Experiment to fourteen hours after



a) at t = 0





c) at t = 6 hours d) at t = 14 hours Figure 2.10: Superposition of an experimental drop of EVOH in PP at 223°C and the corresponding theoretical drop. (•) represent the experimental data and (-) the theoretical data

i) Necking Effect

Figure 2.11 shows the "necking phenomenon". It shows the evolution of a drop of PS_b in PP at 250°C: a) at the beginning of the experiment, b) after one hour, c) after one and a half hour, d) after two hours, e) after three hours and f) after four hours.

The phenomenon of necking has been observed for the same polymers by Escudie [1986] (PP with a molecular weight of 330,000 and PS with a molecular weight of 220,000). The necking effect suggests that there is a critical necking volume for drops that is particular to every polymer. This critical volume should not be surpassed, otherwise incorrect determination of the interfacial tension would occur as indicated above. Reported values of the interfacial tension using the pendant drop method by different researchers could therefore be questionable. The critical volume is likely to be related to the balance between gravity and surface forces.

ii) Capillary Effect

Figure 2.12 shows the evolution of a drop of PS_b in PP at a temperature of 250°C, for a case where the initial drop volume was smaller than the critical capillary volume: a) at the beginning of the experiment, b) after one hour and a half, c) after two hours, d) after three hours and e) four hours after starting the experiment.

When the drop was smaller than the critical volume, it was found that a fine film of polystyrene formed on the internal surface of the plunger. This indicates a capillarity effect. The syringe and plunger were redesigned in order to avoid this problem. The new design, with a resilient ring of material on the plunger was successful. It was possible to form small drops and reach mechanical equilibrium.



Figure 2.11: The "Necking Effect", Evolution of a Drop of PS in PP at a Temperature of 250°C.

a) at $t = 0$	d) at t = 2 hours
b) at t = 1 hour	e) at t = 3 hours
c) at t = 1.5 hour	f) at t = 4 hours



c) f)

Figure 2.12: The capillarity effect, evolution of a drop of PS in PP at a Temperature of 250°C.

a) at t = 0	d) at $t = 3$ hours
b) at t = 1.5 hour	e) at t = 4 hours
c) at t = 2 hours	

43

3.6.2 Experimental results

The interfacial tension between ethylene vinyl alcohol copolymer (EVOH) and polypropylene (PP) was measured for temperatures ranging from 210°C to 250°C. These polymers were commercial high molecular weight resins. The results are reported in Chapter 3. Also interfacial tension values for the polymer pairs polypropylene (PP) / polystyrene (PS) and polyethylene (PE) / polystyrene (PS) were measured. Effects of temperature, molecular weight, molecular weight distribution and blending were evaluated with the pendant drop method. The temperatures ranged from 130°C to 250°C. Experiments were also conducted to test the stability of the measured interfacial tension values after equilibrium. In these experiments, the interfacial tension was measured up to eight hours after the equilibrium was reached. The values of interfacial tension remained constant over the whole eight hour period to within $\pm 1\%$.

It was not possible to measure the interfacial tension between EVOH and maleated PP (except for MAgPP₂ see Chapter 3, Section 6). In one case, degradation took place before reaching mechanical equilibrium (for MAgPP₁). In the other cases, the polymer with the lower density (forming the matrix) was too dark. The pendant drop method requires a transparent matrix which was not the case for MAgPP_x with x>3.

3.6.3 Evolution of $\gamma(t)$

It was observed that the evolution of the interfacial tension with time $\gamma(t)$, obtained experimentally from the shape comparison, can be fitted by a negative exponential. For example, Figure 2.13 shows the evolution of $\gamma(t)$ for commercial PP and EVOH at 226°C. The open points represent the experimental values of $\gamma(t)$, and the continuous line represents the best fit (obtained by least square regression, the fitting was done using the mathematical tools of Sigma-plot Software, Jandel Scientific) corresponding to the following equation:

$$\gamma(t) - \gamma_{\infty} = (\gamma_{\alpha} - \gamma_{\infty})e^{-m_{1}(t-t_{\alpha})}$$
(2.22)

where γ_{∞} is the value of $\gamma(t)$ at infinite time, i.e., the value of interfacial tension. γ_{0} is the value of $\gamma(t)$ at t_{0} , and m_{1} is a constant.

Joseph [1992] studied the evolution of the radius of a spinning drop of one polymer into another. He demonstrated theoretically that the evolution of the radius of the spinning drop of one polymer into another can be fitted by:

$$R(t) - R_{\infty} = (R_{o} - R_{\infty})e^{-m(t-t_{o})}$$
(2.23)

where R(t) is the radius of the spinning drop as a function of time, R_{∞} is the radius at infinite time, i.e., the radius at equilibrium, R_{0} is the radius at t_{0} , and m is a parameter related to the relaxation times of the polymer. Fitting equation 2.23 to experimental data of R(t), Joseph could infer the value of R_{∞} at equilibrium and consequently the value of interfacial tension.



Figure 2.13: Interfacial Relaxation Function for EVOH in PP at 223°C

In this work, the same rational was applied to the experimental value of the interfacial tension obtained with the pendant drop. It was attempted to fit an exponential curve to $\gamma(t)$. This is shown in Figure 2.13. It can be seen that equation 2.22 represents well the evolution of the interfacial tension as a function of time for this work.

The interfacial tension, between PP and EVOH at a temperature of 223°C, obtained by the exponential fitting is 18.3 dyne/cm which is in agreement with the value calculated at equilibrium: 17.9 dyne/cm. Table 2.2 shows γ_{∞} (obtained by exponential fitting) for some polymer pairs as examples, as well as the interfacial tension obtained at equilibrium. It can be seen that good agreement is obtained between the two sets of data. Table 2.2 also shows the values of m_1 for the different polymer pairs and of the zero shear stress viscosity, η_{∞} , of the polymers used. It can be observed that the values of m are an increasing function of the zero shear stress viscosity.

Resins	γ at equilibrium (dyn/cm)	γ calculated with 2.22 (dyn/cm)	m _l (hour ⁻¹)	η, (Pa.s)	Time to reach equilibrium (hour)
PP/EVOH (216°) Matrix PP Drop EVOH	19.63±0.85	20.57	1.64	-	8
Matrix PP Drop EVOH	18.40±0.93	17.89	1.5	2.87 10 ⁴ 4.42 10 ³	6.8
Matrix PP Drop EVOH	16.43主0.40	16.63	1.06	•	4.75
Matrix PP Drop EVOH	14.83±0.20	15.71	1.08	1.16 10 ⁴ 3.73 10 ³	4
PP/PS (178°) Matrix PP Drop PS	11.02±0.4	11.34	1.99	5.22 10 ⁴ 7.85 10 ³	5
PP/PS (226°) Matrix PP Drop PS	8.46±0.45	7.86	1.3	1.53 10 ⁴ 6.75 10 ³	3
PP/PS _a (220°) Matrix PP Drop PS	6.78±0.25	6.20	0.3	· .	1.66

Table 2.2 : Interfacial	Tension	from Tr	ransient I	Measurements
-------------------------	---------	---------	------------	--------------
The values of the zero shear stress viscosity were obtained from dynamic measurements and analysis with Iris Software [1990] developed by Baumgaertel [1989].

It was observed that at lower temperatures, more time was required to reach equilibrium. Also, depending on the size of the drop, the time to reach equilibrium would vary (the smaller the drop the shorter the time to reach equilibrium). These two facts suggest that the time to reach equilibrium is a function of the viscoelastic properties of polymer melts.

4. THE SPINNING DROP APPARATUS

The pendant drop apparatus described above was used to measure the interfacial tension for different polymer pairs (PP/PS, PE/PS, PP/EVOH). Also, it was used to study the effect of temperature, molecular weight and molecular weight distribution on interfacial tension. However, it was not possible to measure the interfacial tension between maleated polypropylene and EVOH with the pendant drop apparatus and to study the effect of compatibilizers on interfacial tension for the following reasons:

i) In some cases, the polymer would degrade before equilibrium would take place (in the case of MAgPP₁ see Chapter 3, Section 6)

ii) In other cases, the resin with lower density was opaque (in the case of $MAgPP_x$ with x>2), making it difficult to perform an experiment with the pendant drop instrument.

Therefore, it was decided to build a spinning drop apparatus to determine the interfacial tension for the above systems.

4.1 Theory

An analysis of the physical basis for the spinning drop method is given below (following the work of Vonnegut, 1942 and Princen, 1967).

Consider the system consisting of a drop of one fluid 1 (density ρ_1) in a fluid 2 (density ρ_2) placed in a horizontal spinning tube. In the remaining discussion the drop (the lighter phase) will be fluid 1 and the matrix (the heavier phase) will be fluid 2. The tube is

rotated at a known speed about the horizontal axis. The drop elongates until the centrifugal and interfacial forces are balanced. The geometry of the drop satisfies the following equation:

$$\Delta \mathbf{P} = \gamma \mathbf{J} \tag{2.24}$$

where ΔP is the difference of pressure across the interface, γ is the interfacial tension and J is the total curvature of the interface.

The difference of pressure can be written as:

$$\Delta \mathbf{P} = \Delta \mathbf{P}_{o} - \frac{1}{2}\omega^{2}r^{2}\Delta\rho \qquad (2.25)$$

and the curvature as

$$J = \frac{1}{r} \frac{d}{dr} \frac{r}{(1 + (\frac{dr}{dx})^2)}$$
 (2.26)

where $\Delta \rho$ is the density difference between the two fluids, r is the cylindrical coordinate, and ΔP_o is the difference in pressure at r = 0:

$$\Delta P_{o} = \frac{2\gamma}{a} \tag{2.27}$$

where a is the radius of curvature at the origin.

Substituting Equations 2.25, 2.26 and 2.27 into Equation 2.24 yields, after some transformation :

$$\gamma = \frac{\omega^2 \Delta \rho}{4C} \tag{2.28}$$

where C is a constant to be calculated from:

$$L_{o} = \frac{\frac{4}{3}(Cr_{o}^{3} + 1)}{(Cr_{o}^{3})^{\frac{1}{3}}}$$
(2.29)

where L_{o} is the equilibrium length of the rotating drop and r_{o} is the initial drop radius.

If the speed is sufficiently high (if the length of the drop is at least equal to four times the diameter of the drop as derived by Vonnegut), Equation 2.28 can be reduced to:

$$\gamma = \frac{\omega^2 d^3 \Delta \rho}{32} \tag{2.30}$$

where d is the final diameter of the drop.

The effect of gravity can be neglected if the Froude number meets the following conditions [Than, 1988]:

$$\mathbf{F} = \frac{\omega^2 \mathbf{d}}{2\mathbf{g}} >> 1 \tag{2.31}$$

The effect of gravity tends to locate the drop at $\frac{gr}{\omega^2}$ (where g is the gravitational acceleration, r is the radius of the drop and ω is the rotational speed). At very high angular velocity, the displacement of the drop from the axis of rotation is very small (25µm at an angular velocity of 6000 rpm) and, therefore, it can be neglected.

4.2 Spinning Drop Method Development

Patterson et al [1971] were the first to use the spinning drop method to determine the interfacial tension between polymers. They determined the interfacial tension between polyisobutylene and poly(dimethylsiloxane). Steady state was not reached even after three hours for viscous systems (3000-5000 poise). The maximum speed reached was 6500 rpm. However, the researchers noticed that after an initial time varying from 25 to 480 seconds, depending on the viscosity of the liquids, log(L)=f(1/t) was linear (where L is the length of the drop). Extrapolating the line to 1/t = 0, they could infer the interfacial tension in a reasonable time to avoid thermal degradation.

Another approach to reduce the equilibration time was employed by Elmendorp and de Vos [1986]. They forced the equilibrium to occur by increasing the angular velocity at the beginning of the experiment. Then they reduced the speed to the level required for the measurement. Their apparatus could rotate at speeds up to 25,000 rpm. It was used successfully to measure interfacial tension between different polymer pairs.

Joseph et al [1992] developed a theory of exponential fitting to cope with the problem of long equilibration times for polymers. They proved that the relaxation of the polymeric system in the spinning tube follows an exponential decay. When performing an experiment, they measured the radius of the evolving drop at a fixed value of rotational velocity. The radius tends exponentially to an equilibrium value according to the following equation:

$$R(t) - R_{\omega} = (R_{o} - R_{\omega})e^{-m(t-t_{o})}$$
(2.23)

where R(t) is the maximum radius of the spinning drop as a function of time, R_o is the initial radius of the drop, R_{∞} is the equilibrium radius, t_o is the initial time at which the experiment started and m is a parameter related to the relaxation times of the system.

Joseph et al fit an exponential curve to the data to determine R_{∞} and m and thus they were able to infer the interfacial tension. They applied their theory to the results of Verdier [1990] who measured the interfacial tension between polyethylene (PE) and polystyrene (PS), and between PE and polymethylmetacrylate (PMMA), and between PS and PPMA at temperatures ranging from 200°C to 250°C. The conclusion was that the method of fitting introduced by Patterson and the method of exponential fitting led to near agreement. However, only few data were available for comparison.

In the present work, all three methods were used to calculate interfacial tension from spinning drop profiles of polymers, i.e.:

(i) increasing the speed of the spinning drop [Vonnegut, 1986],

(ii) the interpolation of Patterson [1971] and

(iii) the theoretical approach of Joseph [1992].

4.3 Apparatus

4.3.1 General Description

The apparatus consisted of three parts: an experimental assembly mounted on a vibration proof table, an optical system to capture the image of the drop, and a data acquisition system with a PC computer equipped with an Intel \$0486 processor to compute the interfacial tension from the drop profile. A general view of the apparatus is given in Figure 2.14.



Figure 2.14 General View of the Spinning Drop Apparatus

4.3.2 The experimental Assembly

A schematic representation of the experimental assembly is shown in Figure 2.15. It consists of a motor which rotates the shaft attached to a glass tube containing the polymers, an oven and a capacitance probe (to measure the volume expansion of the polymers). The motor and bearings are mounted on precision rods (for precise alignment) away from the oven to avoid heating. The special high speed ball bearings are located on cooling fins. The oven sits on a maronite support and does not have any contact with the linear rods. The different parts of the experimental cell are described below:



Figure 2.15: Sketch of the Spinning Drop Apparatus

4.3.2.1 The Motor

A brushless DC motor (from Nu-Tech Ind.) was used for rotating the shaft containing the glass tube with the polymers. The speed of rotation of the motor can be controlled through its power supply up to 40,000 rpm. The speed is displayed on a digital RPM readout. The speed of the motor was checked independently using an oscilloscope. No vibrations of the assembly were noticed when using the motor.

4.3.2.2. Oven and Controller

The oven consists of two parts, the top one being easily removable. A cylindrical hole runs through the center of the oven and allows the glass tube to rotate freely. The oven contains two windows (one on each side of the oven), allowing the viewing of the glass tube and polymer. A light source, located behind the windows, provides the illumination of the polymers. The temperature at the center of the oven is measured with a thermocouple inserted through the top. The oven can be heated up to 400°C. The temperature is maintaine.¹ constant with a P.I.D. (Proportional Integrator Derivator) controller (Model EPS-120-25-2 from Mellen Company) to within $\pm 0.5^{\circ}$ C. This oven has been specially manufactured for this project by Mellen company.

There is some discrepancy between the temperatures of the polymer and the oven, because of the poor thermal conductivity of the polymer. For each polymeric material, a temperature calibration was conducted. In order to determine the temperature of the polymer, a thermocouple was inserted in the polymer. This temperature was obtained while the tube was rotating and correlated with the oven temperature.

During an interfacial tension measurement, the calibrated temperature of the oven was used to infer the temperature of the polymer. The length of the thermocouple was varied to verify if there was a gradient of temperature of the polymer along the tube. Less than 0.5°C difference was observed along the whole length of the tube.

4.3.2.3. Shaft and Glass Tube

Figure 2.16 shows a sketch of the shaft in which the glass tube containing the polymer is inserted. The glass tube fits tightly into the shaft. Two "O" rings on each part of the shaft prevent the glass from breaking during heating and rotation. The glass tube (manufactured especially by Wilmad Glass company) is made of bore glass (borosillicate glass). The outside diameter is 1/2 inch and the outside diameter 3/8 inch. The glass tube is closed using one glass plunger in each side. One of the plungers is allowed to move as a result of the expansion of the polymer which reaches about 18%. At the end of this plunger a system with a capacitance probe was mounted to allow the determination of density as a function of temperature. The glass dilatation can be neglected in the calculation. Once the setpoint in temperature is reached, this plunger is locked in place to avoid movement during rotation and to prevent polymer and air leakage.



Figure 2.16: Spinning Drop Apparatus: the Shaft

4.3.2.4. The Capacitance Probe

A capacitance probe (HPC-375-A from Capacitec) was used to measure the melt expansion as a function of the temperature. The output voltage of the capacitance probe is directly proportional to the distance between the plunger and the capacitance probe. Since the weight and the initial volume of the polymer are known, it is possible to calculate the melt expansion as a function of temperature.

4.3.3 The optical system

The light source used for the spinning drop was the same one used for the pendant drop. The same camera was also used for the spinning drop apparatus with a TV zoom lens 17-102 mm instead of the macro-zoom lens used for the pendant drop. The light source and video camera were assembled on a XYZ support, as for the pendant drop. The whole system was also mounted and aligned on a vibration proof table. Three rods of stainless steal of different diameters were used to determine the distortion and the optical enlargement. The rods were viewed by the camera and digitized by the frame grabber. It was possible to calculate the optical enlargement knowing the real dimensions of the rods and the dimensions (pixels) of the rod digitized by the frame grabber. The edges of the rods were perfectly parallel. It was ascertained that they were still parallel after digitization.

4.3.4 Measurement of Interfacial Tension

Using the system described above, images of the spinning drop of polymers were obtained and digitized by the same frame grabber as in the case of the pendant drop. The dimensions of the spinning drop obtained with this system were used in Equation 2.30, together with data regarding the difference of densities of the two polymers and the rotational speed, in order to obtain the interfacial tension. The spinning drop was monitored during the experiment. When the dimensions of the drop remained constant for more than 1/2 hour, equilibrium was assumed to have been reached.

The dimensions of the drop were measured either using a cross-hair cursor that displays the position of the object on the monitor or, more precisely, with the edge detection program presented in Section 3.3.5.a.

As in the case of the pendant drop, the whole process of digitizing and calculation of interfacial tension required less than one minute per image. The evolution of the image was monitored on the screen of the computer and the interfacial tension was inferred from this image.

4.4 Experimental Procedures

4.4.1 Samples

The polymer with the highest density (forming the matrix (rod), in which a drop of a second polymer was inserted) was used to form two equal hemicylinders that were obtained by injection molding. A hole was drilled in one of the hemicylinders. The hole was filled with the polymer with the lower density to form the drop. The polymer forming the drop was compression molded and then machined to the proper dimensions. The excess of polymer in the hole was trimmed so that smooth contact was obtained between both surfaces of the two hemicylinders. The two hemicylinders containing the drop were inserted inside the glass tube. Prior to an experiment, the samples were dried for 12 hours at 70°C under a nitrogen atmosphere, in order to avoid air bubble during the experiment.

4.4.2 Loading Procedures

When the samples were ready to use, they were inserted in the glass tube between the two plungers. The tube was then mounted in place in the shaft of the apparatus. The sample was progressively heated under vacuum, to avoid the formation of any air bubble. Once the desired temperature was reached, the vacuum pump was disconnected and the experiment started.

4.4.3 Determination of the Correction Factor and Optical Enlargement

The dimensions of the drop had to be corrected for optical enlargement and for the combined refrective index of polymer 2 (matrix) and the glass.

Theories have been developed to evaluate the refractive index of the polymer 2glass assembly involving different optical quantities [Seeto 1977, 1982]. However, it was simpler to determine the correction factor experimentally as reported in earlier works [Manning 1976, Verdier 1990]. A rod was inserted in polymer 2 (matrix) inside the experimental tube and the diameter of the rod, d_{app} , was measured for different temperatures. The real diameter of the rod, d_{real} was known and the correction factor, n, could be calculated as follows:

$$n = \frac{d_{app}}{d_{real}}$$
(2.32)

The correction factor was measured as a function of temperature and used to determine interfacial tension. It should be noted that the index of refraction is independent of the size of the rod. This has also been observed by Verdier [1990].

4.5. Results

In this section, three different aspects of the experiments with polymers are discussed:

(i) the determination of the optical correction factor and density;

(ii) the evaluation of the apparatus; this includes a determination of the experimental error and a comparison of data;

(iii) experiments with polymers; this includes the study of the evolution of the spinning drop profile with time, the study of the evolution of the radius of the spinning drop and a summary of the experiments performed with the spinning drop method.

4.5.1 Determination of the Optical Correction Factor and Density of the Matrix

When using the spinning drop apparatus, prior to experiment, two measurements had to be performed, as described below.

4.5.1.1 Determination of the Optical Correction Factor

The correction factor was measured as described in Section 4.3. The correction factor was evaluated for a matrix of EVOH as a function of temperature; EVOH was used as the matrix for all the experiments. The thermal expansion of the rod used for measurement was negligible.

Figure 2.17. shows the measurements for the correction factor obtained in this research. The data points are represented by the dots. The correction factor for a matrix of EVOH decreases linearly as a function of temperature. Applying a least square regression to the experimental data, the following equation gives the best fit:

$$n = 1.58 - 8.47 \times 10^{-4} T$$
 $r^2 = 0.999$ (2.33)



where π is the correction factor and T is the temperature in degrees Celsius.

Figure 2.17: Optical Correction Factor for EVOH

4.5.1.2 Determination of Density with the Capacitance Probe

Figure 2.18 shows the density of ethylene vinyl alcohol copolymer as a function of temperature. The line represents the measurements obtained with the capacitance probe, and the data points represent the measurements obtained with the capillary rheometer. This graph shows very good agreement between the results obtained with the capacitance probe and the capillary rheometer.



Figure 2.18: Density of EVOH

4.5.2 Evaluation of the Spinning Drop Apparatus

4.5.2.1 Determination of Experimental Error

The total error in the determination of the interfacial tension is due to the combination of all the different experimental errors that are associated with the determination of the parameters used in the determination of interfacial tension. The maximum error associated with each of the measurements involved is given below:

- (a) temperature: 0.5%.
- (b) rotational speed: 0.02%.
- (c) density difference: 2%.
- (d) determination of the correction factor: 1%.

(e) approximation of equation 2.30 : 0.4% (when the length of the drop is equal to four times the maximum diameter of the drop). In practice, the length of the drop is always more than four times the maximum diameter.

Using Equation 2.30, the maximum relative error in the determination of the interfacial tension can be calculated as follows:

$$\frac{\Delta\gamma}{\gamma} = 0.005 + (0.00025)x^3 + 0.02 + (0.01)x^3 + (0.01)x^3 + 0.004 \approx 0.09$$
(2.34)

The estimated maximum error in the determination of the interfacial tension is therefore 9%.

4.5.2.2 Comparison of Data

a) Spinning Drop Measurements at Different Speeds

The interfacial tension between polypropylene and ethylene vinyl alcohol at a temperature of 202°C was measured with the spinning drop apparatus using two angular speeds. The interfacial tension results for the experiment performed at a speed of 16,000 rpm were the same as those obtained during the experiment performed at a speed of 20,000 rpm. The only difference was the time to reach equilibrium. It took 150 minutes to

reach equilibrium when the drop was rotated at 16,000 rpm and 100 minutes at 20,000 rpm.

b) Comparison with Pendant Drop Measurement

The interfacial tension between polypropylene and ethylene vinyl alcohol was measured at two temperatures with both the pendant drop apparatus and the spinning drop apparatus. Table 2.3 shows the results obtained with the two methods.

Temperature (^o C)	Interfacial Tension measured with pendant drop method	Interfacial Tension measured with spinning drop method
202	17.1±0.4	16.9±0.3
224	12.0±01	12.5±0.5

Table 2.3: Comparison between the Pendant and Spinning Drop Method

The results obtained with the pendant drop and the spinning drop assemblies are in good agreement. The differences between the values obtained for the interfacial tension are within the experimental error. The comparisons of the values of interfacial tension measured with the pendant drop and the spinning drop methods indicate the accuracy of the experimental and computational systems.

4.5.3 Experiments with Polymers

Only three aspects of the experiments performed with polymers are examined here. The complete set of experiments with polymers is given in Chapter 3 Section 6:

4.5.3.1 Evolution of Drop Profile

Figures 2.19a to 2.19e are tracings of a drop of ethylene vinyl alcohol in polypropylene at a temperature of 202°C. They show the evolution of the drop rotated at 16,000 rpm at: a) the beginning of the experiment, b) after 12 minutes, c) after 110 minutes, d) after 210 minutes and e) after 300 minutes. After 150 minutes the drop shape did not change any more with time, as can be seen in Figures 2.19d and 2.19e. Figures 2.20a to 2.20e show the same drops as they appear on the monitor after digitization.



Figure 2.19: Tracings of a Drop of PP in EVOH at 202°C



Figure 2.20 : Evolution of a Drop of PP in EVOH at 202°C - Digitized Images

4.5.3.2. Comparison Between Three Calculation Methods

Three approaches were used to calculate the interfacial tension from the profile analysis of the spinning drop as described in Section 4.2. The three approaches were: a) value at equilibrium; b) method of Patterson and c) method of Joseph. These methods have been discussed in Section 4.2. As an example, the results of using the three different approaches for the determination of the interfacial tension between pure PP and EVOH at 202^o are reported below.

a) Value at equilibrium

The value at equilibrium measured experimentally (at times greater than 150 min.) was 16.2 dyn/cm. For all the experiments performed in this work, mechanical equilibrium of the spinning drop was reached and the interfacial tension could therefore be inferred from the dimensions of the drop at equilibrium.

b) Method of Patterson [1971]

Figure 2.21 shows the evolution of the logarithm of the interfacial tension as a function of the inverse of time. At times greater than 18 minutes, the logarithm of the interfacial tension as a function of the inverse of time can be fitted by a straight line as observed by Patterson. The intercept of this line with the y-axis provides the value of the interfacial tension, which, for the experimental data of interfacial tension between EVOH and PP at 202°C, is equal to 16.9 dyn/cm. The best fit was obtained of the data was obtained performing a least square regression.





c) Method of Joseph [1992]

Figure 2.22 shows the evolution of the radius of the drop of EVOH in PP at 202°C. The data points are represented by the open circles. The line corresponds to the fitting (obtained by least-square regression, the fitting was done using the mathematical tools of Sigma-Plot Software) of Equation 2.31 with R=1.37 mm and m=0.07 hour. It yields an interfacial tension value of 16.7 dyn/cm.

The analysis of the evolution of the radius of the spinning drop with time was conducted using the empirical method suggested by Patterson [1971] and the theoretical treatment suggested by Joseph [1992]. Both methods provide interfacial tension values in good agreement with the one found at equilibrium. The differences observed between the results are within the experimental errors.



Figure 2.22: Evolution of the Radius of a Drop of PP in EVOH at 202°C

4.5.3.3 Measurement of Interfacial Tension between Polymers

The spinning drop apparatus was used to determine the interfacial tension for different polymer pairs. The results are presented in Chapter 3, Section 6. The following polymers and operating conditions were used:

a) Polypropylene and ethylene vinyl alcohol copolymer at 202° C and 224° C - in order to compare with experimental results obtained with the pendant drop apparatus and to validate both devices.

b) Polypropylene and ethylene vinyl alcohol copolymer at temperatures below 210°C At these temperatures, it was not possible to use the pendant drop apparatus to determine the interfacial tension due to long equilibration times.

c) Maleated polypropylene (MAgPP) and ethylene vinyl alcohol as a function of maleation content of polypropylene. The interfacial tension between these resins could not be measured with the pendant drop method. For MAgPP₁ (see Chapter 3, Section 6) degradation took place before mechanical equilibrium of the drop. The resins with maleation level higher than 0.15 wt % (MAgPP_x with x>2) were opaque. Since the MAgPP forms the matrix in the pendant drop method, interfacial tension measurements between MAgPP_x with x>2 with the pendant drop method were not possible.

5) COMPARISON BETWEEN THE TWO TECHNIQUES

The two methods presented above, the pendant drop and the spinning drop, were used to measure the interfacial tension between polymers. They proved to be reliable and the results of interfacial tension between polymers found with both machines compared very well within experimental error, as shown in Table 2.3.

The pendant drop method has the following advantages over the spinning drop for the determination of interfacial tension between polymer pairs:

(a) The solution of the Bashforth and Adams equation used with the pendant drop requires less assumptions than the solution of the equilibrium equation used for the spinning drop method (Equation 2.30).

(b) When the material with the higher density is opaque, it is not possible to use the spinning drop method.

(c) Less polymer is required to perform the pendant drop experiment.

The disadvantages of the pendant drop method when compared to the spinning drop method are outlined below.

(a) Equilibrium times are longer, which could prevent some measurements due to thermal degradation of the polymer.

(b) When the material with the lower density is opaque, it is not possible to use the pendant drop method.

(c) Problems such as the capillary or necking effects can be encountered.

6. SUMMARY

The main achievements reported in this Chapter are summarized below:

(a) Two static methods, the pendant drop and the spinning drop, were chosen among the different existing methods to measure interfacial tension of polymers. These two methods present the following advantages over other methods: precision, shorter equilibration time and reliability.

(b) Two devices, one based on the pendant drop and the other on the spinning drop to measure the interfacial tension between polymers, were designed, constructed and proven to be reliable. Algorithms were chosen and computer programs were written for digitization of the drops and calculations of interfacial tension.

(c) The special design and operating procedure employed with the pendant drop method have removed complications associated with capillarity and necking effects.

(d) A capacitance probe was employed for the determination of density as a function of temperature for polymers. The capacitance probe results compared well with capillary rheometer measurements.

(e) For both instruments, the possibility of viewing the drop in real time and calculation of interfacial tension on-line results in high accuracy in the determination of drop equilibrium and observation of eventual degradation.

(f) Measurements of interfacial tension between polymers performed with both instruments compared well. The two devices have proven to be complementary and useful for the determination of interfacial tension between the different polymers used in this study.

(g) The pendant drop method should be used when there are limited quantities of polymer and the denser polymer is opaque. The spinning drop should be used when thermal degradation could be a problem and when the lighter polymer is opaque.

(h) It is possible to infer the interfacial tension between polymers from transient measurements using an exponential fitting of the interfacial tension (pendant drop) or

radius (spinning drop). This procedure shortens the experimental time, which helps to avoid the thermal degradation of the polymers.

REFERENCES

Adamson A.W. [1967] : "Physical Chemistry of surfaces", 2nd ed., Wiley (Interscience), New York, pp 9-41, (1967).

Anastasiadis S.H., Chen J.K., Keberstein J. T., Sohn J.E. and Emerson J.A. [1986] Polym. Eng. Sci., Vol 26, p 1410, (1986).

Anastasiadis S.H., Chen J.K, Koberstein J.T., Siegel A.F., Sohn J.E. and Emerson J.A. [1987] : J. Colloid Interface Sci., Vol93, p169, (1987).

Anastasiadis S.H. [1988] : "Interfacial Tension of Immiscible Polymer Blends", Ph.D. Thesis, Princeton University, (1988).

Andreas J.M., Hauser E.A. and Tucker W.B. [1938] : J. Physical Chemistry, Vol 42, p 1001, (1938).

Bashforth S. and Addams J.C. [1882] : An attempt to test the Theory of Capillary Action, Cambridge University Press and Deighton, Bell and Co, London (1882).

Baumgaertel M. and Winter H.H. [1989] : Rheol Acta, Vol 28, p 511, (1989).

Baumgaertel M. and Winter H.H. [1990] : "Iris Software to Determine Relaxation Spectrum from Experiments" (1990).

Boyce J.F., Schürch S., Rotenberg Y. and Newman A.W. [1984] : Colloids and Surfaces, Vol 9, p307, (1984).

Butler J.N. and Bloom B.H. [1966] : Surface Science, Vol 4, p 1, (1966).

Carriere C.J., Cohen A. and Arends C.B. [1989] : J. of Rheol., Vol 33 (5), p 681, (1989).

Carriere C.J., Cohen A. [1991] : J. of Rheol., Vol 35 (2), p205, (1991).

Chappelear D.G. [1964] : Polymer Preprints, Vol 5, p 363, (1964).

Cheng P., Li D., Boruvka I., Rotenberg Y. and Newman A.W., [1990] : J. of Colloid and Surfaces, Vol 43, p 151, (1990).

Cohen A. and Carriere C.J. [1989] : Rheol Acta, Vol 28, p 223, (1989).

Demarquette N.R. and Kamal M.R. [1992] : Antec proceedings, p 1907, (1992).

Demarquette N.R. and Kamal M.R. [1993] : Interfacial tension of polymers using an on-line pendant drop apparatus, submitted to the journal of Polymer Engineering Sciences.

Dettre R.H. and Johnson R.E. Jr. [1966] : J. Colloid Interface Sci., Vol 21, p367, (1966).

Du Nouy L. [1919] : J. Gen. Physiol., Vol 1, p 521 (1919).

Edwards H. [1968] : J. Appl. Polym. Sci., Vol 12, p 2213 (1968).

Elemans P. and Janssen J. : J. Rheol, Vol 34, p 781, (1990).

Elmendorp J.J. and de Vos G. [1986] :Polym. Eng Sci., Vol 26, p 415, (1986).

Escudie E. [1986] : Material chemistry and Physics, Vol 14, p239, (1986).

Fordham S. [1948] : Proc. Roy. Soc. (London), A 194, Vol 1, (1948).

Girault H.H., Schiffrin D.J., Smith B.D.V., [1982] : J. Electroanal. Chem., Vol 137, p 207, (1982).

Girault H.H., Schiffrin D.J., Smith B.D.V., [1984] : J. Colloid Interface Sci., Vol 93, p 169, (1984).

Graebling D. and Muller R. [1991] : Colloids and Surfaces, Vol 55, p 89, (1991).

Grassmespacher H. and Meissner J. [1992] : J. Rheol., Vol 36(6), p 1127, (1992).

Hartford J.R.J., and White E.F.T. [1969] : Trans. J. Plastics Inst., Vol 53, (1969).

Handbook CRC of Physics and Chemistry [1983] : 4th Edition, R.C. Weast ed, C.R.C Press Boca Raton, FL, (1983).

Huh H C., Reed R.L. [1983] : J. Colloid Interface Sci., Vol 91, p 472, (1983).

Jon D.I. [1986] : "Measurements of low and ultralow interfacial tension by means of the blade, spinning drop, pendant drop and laser light scattering method", Ph.D Thesis, University of New York, (1986).

Joseph D.D., Arney M.S., Gillberg G., Hu H., Hultman D., Verdier C., Vinagre T.M. [1992] : J. Rheol., Vol 36(4), p 623, (1992).

Lau W. Y. and Burns C.M. [1972] : Surface Science, Vol 30, p 483, (1972).

Lord Rayleigh [1879] : Proc. R. Soc. London, Vol 29, p 71, (1879).

Lord Rayleigh [1892] : Phil Mag., Vol 34, p195, (1892).

Lord Rayleigh [1899] : Phil. Mag., Vol 48, p321, (1899).

Manning C.D. [1976] : M.S. Thesis, University of Minesota, (1976).

Maze C. and Burnet G. [1969] : Surface Science, Vol 13, p 451, (1969).

Maze C. and Burnet G. [1971] : Surface Science, Vol 24, p 335, (1971).

Miller R., Treppo S, Voigt A., Zingg W. and Newman A.W. [1993] : Colloids and Surfaces, Vol 69, p203 (1993).

Newman S.B. and Lee W.L. [1958] : J.Polym Sci., Part B, Vol 3, p 235, (1958).

Oda Y. and Hata T. [1968] : Preprints, 267, 17th Annual Meeting of the High Polymer Society of Japan (May 1968).

Oda Y. and Hata T. [1968] : Preprints, 267, Sixth Synposium on Adhesion and Adhesives, Osaha, Japan, p 69, (Jun 1968).

Padday J.F. [1969] : in Surface and Colloid Science, Vol 1, E. Matijevic Ed., John wiley & Sons, New York, (1969).

Patterson H.T., Hu K.H. and Grindstaff T.H. [1971] : J. Polym. Sci., Part C, Vol 34, p 31, (1971).

Perry R.H. and Chilton O.H. [1973] : "Chemical Engineers'Hndbook, McGraw Hill, New York, (1973).

Princen H.M., Zia I.Y.Z. and Mason S.G. [1967] : J.Colloid Interface Sci., Vol 23, p 99, (1967).

Roe R.J., Bachetta V.L. and Wong P.M.G. [1967] :J.Phys. Chem., Vol 71, p 4190, (1967).

Roe R.J. [1970]: J. Phys. Chem., Vol 72, p 2013, (1970).

Roe R.J. [1969] : J. Colloid Interface Sci., Vol31, p 228 (1969).

Rotenberg Y., Boruvka L. and Newman A.W. [1983] : J. Colloid Interface Sci., Vol 93, p169, (1983).

Sakai T., [1965] : Polymer, Vol 6, p 659, (1965).

Sammler R.L., Dion R.P., Carriere C.J. and Cohen A. [1992] : Rheol Acta, Vol 31, p 554, (1992).

Sauer B.B., Skarlupka R.J., Sano M. and Yu H. [1987] : Polymer preprints, Vol 28, p 20, (1987).

Schonhorn H. and Sharpe L.H. [1965] : J. Polym. Sci., Part B, Vol 3, p 235, (1965).

Schonhorn H., Ryan F.W., and Sharpe L.H. [1966] : J. Polym. Sci., A-2, Vol 4, p 538, (1966).

Seeto Y. [1977] : M.S. Thesis, University of Minnesota (1977).

Seeto Y., Scriven L.E. [1982] : Rev. Sci. Instr., Vol 53(11), p 1757, (1982).

Siegel A.F. [1982] : Biometrica, Vol 69, p 242, (1982).

Siegel A.F., Benson R.H. [1982] : Biometrics, Vol 38, p 341, (1982).

Skinner F.K., Rotenberg Y., and Newman A.W. [1989] : Journal of colloid and interface Science, Vol 130, $n^{0}1$, (1989).

Staicopulus D.N. [1962] : Journal Interface Sci., Vol 17, p 439, (1962).

Staicopulus D.N. [1963] : Journal Interface Sci., Vol 18, p 793, (1963).

Staicopulus D.N. [1967] : Journal Interface Sci., Vol 23, p 453, (1967).

Stauffer C.E. [1965] : J. Physical Chemistry, Vol 69, n 6, p 1933, (1965).

Than P., Preziosi L., Joseph D.D. and Arney [1988] : J.Colloid Interface Sci., Vol 124, p 552, (1988).

Tomokita S. [1935] : Proc. R. Soc., London Ser. A, Vol 150, p 322, (1937).

Verdier C. [1990] : Topics in the fluid mechanics of viscoelastic liquids, Ph.D. Thesis, University of Minesota, (1990).

Voigt A., Thiel O., Williams D., Policova Z., Zingg W. and Newman A.W. [1991] : Colloid and surfaces Vol 69, p203 (1993).

Vonnegut B. [1942] : Rev. Sci. Instrum., Vol 13, p 6, (1942).

Wu S. [1969] : J. Colloid Interface Sci., Vol 31, p153, (1969).
Wu S. [1970]: J. of Phys. Chem., Vol 74, p 632, (1970).
Wu S. [1971] : J. Polym. Sc., Part C, Vol 34, p19, (1971).
Wu S. [1974]: J. Macromol. Sci. - Revs. Macromol. Chem., C10, p1 (1974).
Wu S. [1982]: Polymer Interface and Adhesion, Marcell Dekker, Inc., NY, (1982).



1. INTRODUCTION

Data regarding interfacial tension between polymers are rather scarce. Research in this field has been limited because of some shortcomings of the experimental techniques (inability to ascertain equilibrium, long equilibration times that would exceed the time for the melt to undergo degradation and others) as described in Chapter 2. Most of the methods used to measure interfacial tension are based on the shape of a drop of one polymer immersed into a second polymer (the matrix) until equilibrium is reached.

Wu [1974,1982] summarized the interfacial tension data between polymers published up to me early 80's. Later, Escudie [1986] measured the interfacial tension between polypropylene (PP) and polystyrene (PS) with the pendant drop technique and studied the effect of temperature on the interfacial tension. Anastasiadis [1988] studied the effects of both temperature and molecular weight on the following polymer pairs: poly(dimethylsiloxane) / polystyrene (PDMS/PS), poly(methylmethacrylate) / polystyrene (PBDH/PS).

Experimental studies of the effect of polydispersity and/or bimodal blending on interfacial tension have not been reported yet. Some relevant theoretical aspects [Helfand et al (1989), Hariharan and Kumar (1990), Broseta et al (1990)] have been reported.

The work presented here aims at extending the range of temperatures and molecular weights employed in interfacial tension measurements. It also attempts to study the effects of bimodal blending and polydispersity on interfacial tension for both pure and commercial polymers. The effect of compatibilizers on interfacial tension is also evaluated.

<u>1.1 Chapter Outline</u>

This chapter presents all the experimental results obtained in this study. The results reported here relate to the study of the influence of temperature, molecular weight, bimodal blending, polydispersity and compatibilizers on interfacial tension. In order to explain some of the experimental observations, the results of surface analysis of the interfaces between dispersed and continuous phase are reported.

The experiments for the determination of the influence of temperature, molecular weight, bimodal blending and polydispersity on interfacial tension were conducted using the pendant drop apparatus described previously. The following polymer pairs were considered: polypropylene / polystyrene (PP_{pure}/PS), polyethylene / polystyrene (PE/PS), polypropylene / ethylene vinyl alcohol copolymer ($PP_{com}/EVOH$). The data are discussed and compared in light of previously published results. A further theoretical analysis of the data is given in Chapter 4.

Interfacial tension between maleate-grafted polypropylene (MAgPP) with different levels of maleation and ethylene vinyl alcohol copolymer (EVOH) was measured, in order to study the effect of maleation on the interfacial tension.

Finally, the results of the surface analysis of the different polymers/blends used in this work are reported. The surfaces of polypropylene (PP), maleated polypropylene (MAgPP) and ethylene vinyl alcohol copolymer (EVOH) were examined by Electron Spectroscopy for Chemical Analysis (E.S.C.A) and by Scanning Electron Microscope (S.E.M).

2. SELECTION AND PROPERTIES OF MATERIAL

2.1. Choice of Material

The choice of materials took the following factors into consideration: a) the utility of the selected polymer systems for fundamental studies and industrial applications; b) the absence of information on interfacial tension for the chosen polymer systems; c) the need for interfacial tension data in this laboratory for the development of laminar blends.

2.1.1. Pure Materials

Pure polystyrene (PS), polyethylene (PE) and polypropylene (PP_{pure}) resins were used in order to study the effects of temperature, molecular weight and polydispersity on interfacial tension. Table 3.1 lists the pure resins and their molecular weight characteristics. PS, PE and PP_{pure} were chosen because they are extensively used commercially. The interfacial tension was measured for PP_{pure}/PS and PE/PS pairs. The polymers used in this work had the following characteristics:

a) PP_{pure} : the same isotactic PP_{pure} was used for all the experiments. The molecular weight of the polymer was $M_w \approx 300,000$ and M_w/M_n was 5.54.

b) PS: monodisperse and polydisperse samples of PS were used in this study. The molecular weight of monodisperse PS varied from 938 to 380,000, in order to study the effect of molecular weight on interfacial tension over a wide range of molecular weights. A polydisperse sample of PS was also used to evaluate the influence of polydispersity on interfacial tension.

c) PE: the samples of PE used in this work were monodisperse and polydisperse with molecular weight ranging from 700 to 2,000.

Table 3.1 : Pure Resins used in this Study

Material	M _n	M _w /M _n	Supplier
Polypropylene	54 000	5 54	Polyssianses Inc.
r i pure	54,000	5,54	r orysciences, me.
Monodisperse			
polystyrene			ļ
PS,	938	1.09	Polymer Laboratories, Ltd
PS ₂	1,589	1.06	Polymer Laboratories, Ltd
PS ₃	4,755	1.05	Polymer Laboratories, Ltd
PS₄	19,417	1.03	Polysciences, Inc.
PS,	86,438	1.04	Polysciences, Inc.
PS ₆	380,000	1.04	Polysciences, Inc.
<u>Polydisperse</u> <u>polystyrene</u> PS	115,500	2.84	Polysciences, Inc.
Monodisperse polvethylene			
PE,	680	1.18	Polymer Laboratories, Ltd
PE,	1,050	1.20	Polymer Laboratories, Ltd
PE ₃	1,870	1.15	Polymer Laboratories, Ltd
Polydisperse polyethylene			-
PE,	770	1.64	Polysciences, Inc.
PE	1,050	1.24	Polysciences, Inc.
PE	1,950	1.56	Polysciences, Inc.
		I	

2.1.2 Blends

In order to study the interfacial tension of bimodal systems, monodisperse samples of pure PS were blended. The following blends were made:

a) PS_2 ($M_n = 1.589$) with PS_4 ($M_n = 19,417$) in 3/97, 8/92, 20/80, 50/50, 70/30 molar ratios (corresponding to 97/3, 92/8, 75/25, 50/50, 25/75 ratios by weight). b) PS_3 ($M_n=4.755$) and PS_5 ($M_n=86,438$) in 2/98, 5/95, 15/85, 50/50 molar ratios (corresponding to 95/5, 75/25, 50/50, 25/75 ratios by weight).

The interfacial tension between each of the above blends and PP_{pure} was measured exclusively with the pendant drop method, since the method requires small amounts of the polymer with the higher density (here the PS, which is available in small quantities). The blends were first obtained by mixing the polymer powders in a mortar, the mixed powders were then inserted in the syringe of the pendant drop system and melted in place.

Blends of polydisperse PS were also used. Table 3.2 presents the characteristics of these blends.

Material	M _n	M _w /M _n	Supplier
PS _a First Peak Second Peak	59,900 655	2.06 1.65	Polysciences, Inc.
PS _b First Peak Second Peak	66,100 695	1.91 1.63	Polysciences, Inc.
PS _e First Peak Second Peak	52,700 570	2.16 1.29	Polysciences, Inc.

Table 3.2: Blends of polystyrene used in this study

The molecular weight distributions of the above bimodal blends were determined in the Chemistry Department of McGill University. The two peaks for each polymers were integrated separately. Figures 3.1 to 3.3 show the GPC molecular weight spectra for these 3 bimodal blends of polydisperse polystyrene.



Figure 3.1: Gel Permeation Chromatography Spectrum for PS_



Figure 3.2: Gel Permeation Chromatography Spectrum for PS_b



LOG M

Figure 3.3: Gel Permeation Chromatography Spectrum for PS_c

.
2.1.3 Commercial Resins

Products of polypropylene (PP) and poly(ethylene-vinyl alcohol) EVOH are often used in packaging applications. EVOH is useful in packaging because it is resistant to permeation of oxygen and carbon dioxide. It is approved by the Food and Drug Administration (FDA) for food contact applications. Blends of PP and EVOH are immiscible and have very poor mechanical properties [Lohfink 1990]. It is possible to use a compatibilizer such as maleic anhydride to improve the miscibility of the blend components. Maleic anhydride can be grafted onto the backbone of polypropylene (to maleate PP), and it has been shown that this transformation improves the mechanical and structural properties of PP/EVOH blends [Lohfink 1990, Hozhabr 1991, Arghyris 1991]. It was therefore of special interest to study the interfacial tension between PP and EVOH and also to study the effect of the addition of maleic anhydride groups to the PP on interfacial tension between maleated PP (MAgPP) and EVOH. Table 3.3 shows the main characteristics of the polymers evaluated in this study.

Material	M _n	M _w /M _n	Supplier
Polypropylene (NPP 7200-AF)			Northern Petrochemical
PP _{com}	75,000	5.44	Company
Ethylene vinyl Alcohol Copolymer (EP-F101)			Usi Chemical
ÈEVOH	35,000	23.12	

Table 3.3 : Commercial Resins used in this Study

Commercial Resins contain a variety of additives and contaminants. Some of these are discussed below.

a) PP_{com} and EVOH

The PP_{con} (NPP 7200-AF) used in this study was a blow molding grade polypropylene. The EVOH (EP-F101) used in this study had 32 mol% ethylene content.

The above commercial resins contain in their compositions some additives such as antistatic agent, coupling agent and others (antioxydant, catalyst residual). The additives are incorporated into the pure polymers in order to avoid lint deposit during storage of the polymer (antistatic agent) or to enhance mechanical properties (coupling agents). The composition of these additives was determined by E.S.C.A (Electron Spectroscopy for Chemical Analysis) and X Ray. The additives were essentially SiO₂ and SiO₃ as shown in Section 7 of this chapter.

b) MAgPP

In order to study the effect of addition of maleic anhydride groups to the PP_{com} on interfacial tension between PP_{com} and EVOH, six maleated samples of PP were used. The level of maleation of PP_{com} varied from 0.067 to 0.262 wt % as reported in Table 3.4. Two commercially available maleated polypropylene samples were investigated (MAgPP₁ and MAgPP₂). They were obtained from Mitsui Petrochemical Company and Mitsubishi Petrochemical Company under the commercial names Admer and Modic, respectively. Maleated polypropylene with other levels of maleation was prepared by mixing a highly maleated polypropylene (Herkoprime G-201 from Himont) with a non-maleated polypropylene in different amounts to produce different maleation levels. The mixing of PP and maleated PP was conducted in a screw extruder. The extrudates were subsequently pelletized. The determination of the content of maleic anhydride grafted on the PP was determined by F.T.I.R (Fourier Transform Infra Red) or titration, as described in Section 2.3 of this chapter.

Material	Maleation Content (% wt)	Supplier
MAgPP,	0.067	Mitsui Petrochemical Company
MAgPP,	0.098	Mitsubishi Petrochemical Company
MAgPP	0.153	Prepared at McGill
MAgPP	0.168	Prepared at McGill
MAgPP	0.222	Prepared at McGill
MAgPP6	0.262	Prepared at McGill

Table 3.4 :	Maleation	Content of	Maleated	PP
-------------	-----------	-------------------	----------	----

2.2 Molecular Weight

The molecular weights of the monodisperse samples of PS and PE were provided by the suppliers and were confirmed using Gel Permeation Chromatography (G.P.C) at the Chemistry laboratories, McGill University.

The molecular weights of all the polydisperse samples and blends of PS were determined by G.P.C at the American Polymer Standard Corporation. The molecular weights of the resins used in this study were reported in Tables 3.2 and 3.3.

The following procedures were employed in the determination of the molecular weight distributions of the polymer by G.P.C:

(i) The PP samples were dissolved in trichlorobenzene (TCB) at 135°C; the column used was American GPC (10^6 + linear + 500Å).

(ii) The PS samples were dissolved in tetrahydrofuran (THF) at 30°C; the column used was American GPC ($10^6 + 500 \text{ \AA} + 100 \text{ \AA}$).

(iii) The PE samples were dissolved in TCB at 135°C and the column used was AM Gel Three linear.

(iv) The EVOH sample was dissolved in dimethyl formamide (DMF) at 50°C and the column was American GPC (Linear + 500Å).

In all cases, the G.P.C. columns were calibrated with standard polymers.

2.3. Maleation Content

Maleated polypropylene is modified polypropylene by grafting maleic anhydride on its backbone. The structure of maleic anhydride grafted polypropylene is given below:



Two methods were used to determine the maleation content of the polypropylene samples: Fourier Transform Infrared spectroscopy (F.T.I.R.) or titration. These methods are described below.

a) Fourier Transform Infrared spectroscopy

The I.R. spectrum of maleic anhydride is characterized by the presence of four important bands of absorption at 1840 cm⁻¹,1780 cm⁻¹,1050 cm⁻¹ and 885 cm⁻¹. The 1780 cm⁻¹ band is very strong and can be conveniently used for the quantitative determination of maleation content in a resin (Trivedi and Culbertson, [1982]). A typical I.R. spectrum is shown on Figure 3.4. The samples were preheated for 5 minutes prior to analysis to remove residual maleic anhydride. Duplicates were run for each of the blends.



Figure 3.4: Typical Spectrum of MAgPP

b) Titration

Maleic ant-ydride can be hydrolyzed to maleic acid. The maleic acid produced in hydrolysis can be titrated with potassium hydroxide to the two end points at pH = 3.7 and pH = 9.5 (Trivedi and Culbertson, [1982]).

Maleated polypropylene was dissolved in hot, refluxing water-saturated xylene. The solution was constantly stirred to ensure good homogeneity. The hot solution was titrated with potassium hydroxide, using thymol blue in dimethyl formamide as an indicator. The end point was detected by a change of color from yellow to blue. Excess of potassium hydroxide was added, and the solution was back-titrated with isopropanic HCl.

The maleation content of the resins used in this study was determined by either of the above two methods. It was verified that both methods gave the same results. The experimental data are presented in Table 3.4. The errors associated with these estimates are $\pm 10\%$.

2.4. Density

The density of the resins was needed for the determination of interfacial tension (cf Equations 2.2 and 2.1). The different methods used to determine the density of each polymer are presented below. All the density values used in this work are presented in Table 3.5.

2.4.1. Density of Polypropylene

The density of polypropylene was evaluated using the equation proposed by Zoller [1979] and given below:

$$V(P,T) = V(0,T)\{1 - C(T)\ln[1 + \frac{P}{B(T)}]\}$$
(3.1)

$$C = 0.0894$$
 (3.1.a)

$$B(T) = B_o \exp(-B_1 T)$$
(3.1.b)

(°C) <u>Material</u> <u>Polypropylene</u> <u>PP</u> <u>Mc nodisperse</u> <u>polystyres</u> <u>PS</u> 938 <u>PS</u> 1,589 <u>PS</u> 1,589 <u>PS</u> 4,755 <u>PS</u> 19,417 <u>PS</u> 86,438 <u>PS</u> 380,000	-			-	0.759	0.757	0.755													
Material Polypropylene PP <u>Micoodisperse</u> PS1 938 PS1 1,589 PS3 4,755 PS3 4,755 PS3 19,417 PS3 86,438 PS4 380,000	-	-		_	0.759	0.757	0.755													
PP M(codiuperte polystyrene PS, 938 PS, 1,589 PS, 1,589 PS, 4,755 PS, 19,417 PS, 86,438 PS, 380,000		-	-	-	0.759	0.757	0.755													
Mc oodinerte polystyrene PS1 938 PS1 1,589 PS3 4,755 PS3 19,417 PS5 86,438 PS3 380,000	-	-	-	-	0.739	0.757	0.122	0.14	0.761	0 740	0.747	A 747	0.744	0 743	6941	0 719	0716	0.713	0.731	0 7 70
PS, 1,589 PS, 1,589 PS, 1,589 PS, 4,755 PS, 19,417 PS, 86,438 PS, 380,000		-	-					Ų.751	0.751	0.749	0.747	0.747	U.744	0.742	V./41	0,738	0.735	0.752	0.731	0.749
PS, 938 PS, 1,589 PS, 4,755 PS, 19,417 PS, 86,438 PS, 380,000		-																		
PS, 1,589 PS, 4,755 PS, 19,417 PS, 86,438 PS, 380,000	-	-					0.952													
PS ₃ 4,755 PS ₄ 19,417 PS ₅ 86,438 PS ₅ 380,000	-	•-		-	+	-	0917	-	-	-	-	-	-	-	-	-	-	-	-	-
PS, 19,417 PS, 86,438 PS, 380,000	1 -		-	-	0.976	-	0.969	0 961	0.960	-	-	กจีรา	-	-	-	-	-	-	-	-
PS, 86,438 PS, 380,000	-	-	-	-	0.984	-	0.978	0.972	0.700	0.961	-	0.960	-	-	~	~	-	-	-	-
PS, 380,000		-	-	-		0.983	0.980	0.974	-	0.966	-	0.962	-	-	-	-	-	-	-	-
	1 -	-	-	-	-	0.705	0.700	0.771	-	0.700	-		-	-	0.953	0 951	-	0.943	-	0.938
Polydisperse	-	-	-	-	-	-	-	-	-	-	-	-	-	-			-		-	
polystyrene																				
PS 115,500					0.986		0.980	0.974		0.966			0.958		0.953					
Blends of polystyrene	-	-	-	-		-	••••	••••	-		-	-		-		-	-	-	-	-
PS.	_				0.986		0.980	0.974		0.964			0.959							
PS.	1.024	1.007	0.999	0.992	0.985	-	0.981	0.973	0.966		-	0.939		-	-	-	-	-	_	-
PS					0.984	-	0.979	0.970		-	-		-	-	-	-	-	-	_	-
Monodisperse	1 -		-	-		-			-	-	-	-	-	-	-	-	-	_		_
colyethylene																				
PE, 1,870	0.789	0.776	0.773	0.769	0.765	_	0.761		_	_	_		_	_	_	_	-		_	_
PE, 1,050	0.779	0.771	0.764	0.761	0.752	-	0,746	-	-	-	-	-	-	-	-	-	_	_	_	_
PE 680	0.772	0.769	0,761	0.759	0.749	_	0.739	_	-	-	-	_	-	-	_	_	_	_		_
Pohylisterne	1					-		-	-	-	-	-	-	-	-	-	-			
<u>colyrthylene</u>																				
PE, 1,950	0.789	0.776	0,773	0.769	0.765	-	0.761	_	-	_	-		_	_	-		-	_	-	-
PE 1,050	0.779	0.771	0.764	0.761	0.752	-	0.746	-	-	-	-	-	-	_	-	-	-	_	_	-
PE 770	0.772	0.769	0.761	0.759	0.749	-	0.739	-	-	-	-	-	_	_	-	-	-		-	
Ethylene Vinyl	1																			
Alcohol Copolymer																				
EVON								_			0.970		0. <u>961</u>	0.954			0.941		0 934	

Table 3.5 : Density of the polymers used in this study

$$B_o = 1520 \text{kg} / \text{m}^2 = 147 \text{latm}$$
 (3.1.c)

$$B_1 = 4.177 \times 10^{-3} C^{-1}$$
 (3.1.d)

where P is the pressure, T is the temperature in K and V is the specific volume.

Zoller [1979] showed that his experimental data and the equation of state give the same results to within 0.001 g/cm^3 .

The density of polypropylene was also measured in this laboratory with a mercury dilatometer. The experimental data did not differ by more than 3% from the data obtained using Zoller's equation.

2.4.2. Density of Monodisperse and Polydisperse Polystyrene

The values of the density of PS were taken from an empirical equation of state proposed by Fox and Flory [1950]. Their equation was obtained for molecular weight (M_v) between 3,000 an 85,000. The density values taken for the two samples with lower molecular weights in this work were extrapolated using that equation. Fox and Flory showed that between 55,000 and 85,000, there was no influence of molecular weight on the density. Ratzsh [1986] also showed that above M_n of 20,000, the influence of the molecular weight on density is almost negligible. Therefore, the equations of state taken for the density as a function of temperature were:

 $PS(M_w = 4,755)$

$$d = 1.09 - 0.00068T \tag{3.2}$$

PS(M_w=19,417)

$$d = 1.10 - 0.00069T$$
(3.3)

PS(Mw=86,438)

$$d = 1.10 - 0.00068T \tag{3.4}$$

PS(M_w=380,000)

$$d = 1.10 - 0.00068T \tag{3.5}$$

where d is the density in g/cm^3 and T is the temperature in $^{\circ}C$.

The densities of the polydisperse samples were taken to be the same as those of the monodisperse samples for molecular weights (M_n) above 60,000. Fox et al (1950) reported a difference between the prediction of the proposed equation and the experimental data of less than 0.03%

2.4.3. Density of Blends of Polystyrene

The densities of the polystyrene blends presented in Table 3.5. were measured with a calibrated gauge heated in a vacuum oven. The experimental error associated with this technique is $\pm 5\%$.

The density of the bimodal blends of PS with M_n (1,589 and 19,417) and (4,755 and 86,438) could not be determined experimentally due to the small available quantity of polymers. The density could not be calculated since no equation of state exists for bimodal polymers. The densities of the bimodal blends were determined just at 186°C because experiments with these materials were performed only at this temperature, (see Section 5 of this Chapter).

The value of the density chosen for the bimodal blends at 186° C was 0.980 g/cm^3 . This value is based on the density of the polydisperse polystyrene blends presented in Table 3.2 at 186° C. The density of the monodisperse blends should be at least as high as the one of the polydisperse blends, following the theoretical rule that the density increases with molecular weight. The density of polystyrene as a function of molecular weight levels off at a value of 0.981 g/cm^{-3} at 20,000, as shown by Ratsch [1986]. The use of either 0.979 or 0.981 results in a difference of interfacial tension of less than 0.5%, and therefore no significant error occurs if either value is used.

2.4.4. Density of Polyethylene

At temperatures above 130°C, the PE used in this study was in a liquid form. The density of the polyethylene was measured with a gauged vial of 10 ml heated in a vacuum oven. The volume expansion of the glass could be taken as negligible (it is equal to 93.5 x 10^{-7} K⁻¹).

2.4.5. Density of Ethylene Vinyl Alcohol

The density of EVOH was measured with the capacitance probe as described in Chapter 2. The error in the evaluation of the density by this method is less than 2%.

2.4.6. Density of Maleated Polypropylene

The density of the maleated polypropylene was assumed to the same as for polypropylene due to the small amount of maleic anhydride added to the resins.

<u>3. INFLUENCE OF TEMPERATURE</u>

3.1. Introduction

Interfacial tension is the single most accessible parameter that describes the thermodynamic state and structure of an interface; therefore it is important to study the influence of temperature on interfacial tension.

In general, interfacial tension is experimentally found to decrease with temperature. The temperature coefficient $\frac{\partial \gamma}{\partial T}$, for polymers (where γ is the interfacial tension and T is the temperature) has been found experimentally to be approximately 0.03 dyn.cm⁻¹.°C⁻¹ [Wu 1982]. This coefficient is smaller than for shorter molecules because of conformational restrictions in polymeric systems [Wu, 1969 and Roe, 1968].

3.2. Results and Discussion

The influence of temperature on the interfacial tension was evaluated for PP_{pure}/PS , PE/PS and $PP_{com}/EVOH$ polymer pairs. Different types of polystyrene and polyethylene with varying molecular weight and polydispersity were used. The resins used were described in Section 2.

The interfacial tension results for the different polymer pairs as a function of temperature are presented in Figures 3.5 to 3.8. The corresponding experimental data are reported in Appendix B. The main conclusions from these experiments are outlined below. A more extensive theoretical analysis is presented in Chapter 4.

a) Theoretically, interfacial tension between polymers should decrease with increasing temperature. An increase of temperature decreases the free energy of mixing at the interface, resulting in a decrease of the interfacial tension. For all the cases studied in this work, the interfacial tension decreased linearly with increasing temperature.

b) The coefficients of the linear least square regression of the data (obtained by fitting the interfacial tension as a function of temperature) for the different polymer pairs are reported in Table 3.6. The coefficients for the monodisperse samples are of the same order of magnitude as those reported by other researchers Wu [1974].

c) The coefficients in Table 3.6 show a stronger dependency of interfacial tension on temperature for polydisperse systems and for blends than for monodisperse polymers (for monodisperse systems coefficients in the range of 0.04 compared to 0.08 for polydisperse systems). Escudie [1986] studied the effect of temperature on interfacial tension between PP (M_w =235,000) and PS (M_w =220,000) "with very broad molecular weight distribution" and reported a temperature coefficient of 0.069 dyn/cm/K which is in the range of the temperature coefficients reported here for polydisperse PP_{pure}/PS systems.

d) Anastasiadis [1987] reported that the temperature coefficient decreases when the overall molecular weight decreases. In this study, no conclusion could be made in the case of the monodisperse system (PS/PE). However, the results of the interfacial tension between PP_{pure} and the bimodal blends of PS seem to be in agreement with the observation of Anastasiadis.







Figure 3.6: Interfacial Tension between PP_{pure} and Monodisperse Polystyrene as a Function of Temperature



Figure 3.7: Interfacial Tension between PP_{pure} and Polydisperse Polystyrene or Blends of Polystyrene as a Function of Temperature



Figure 3.8: Interfacial Tension between Monodisperse Polyethylene and Monodisperse Polystyrene as a Function of Temperature

e) The value of the coefficient for the $PP_{com}/EVOH$ polymer pair is more than twice the value of the highest coefficient for the other polymer pairs. This can be attributed to the large polydispersity of EVOH, to the effect of additives such as compatibilizers and anti-static agents in the commercial resins and also to the polar groups in EVOH.

f) The values of the interfacial tension of $PP_{com}/EVOH$ polymer pairs are much higher than that for PP_{pure}/PS . This is very likely because of the very high value of the Hildebrand solubility parameter of EVOH. The interfacial tension between two polymers is directly proportional to the difference between Hildebrand solubility parameters of the two polymers (see Chapter 4). The solubility parameters of PS, PE and PP are respectively 18.7, 16.4, and 17.8 $(J.cm^{-3})^{1/2}$ [Cowie 1991] and the solubility parameter of EVOH is 22.8 $(J.cm^{-3})^{1/2}$ as estimated from the sum of the various molar attraction constants tabulated by Small and Hoy [Cowie 1991] (See appendix B₂). Thus, the interfacial tension between PP_{com} and EVOH is expected to be significantly higher than the interfacial tension between PP_{com} and PS or PE and PS.

Polymer pairs	а	ь	2بر
Polypropylene & Monodisperse			
Polystyrene			
PP _{num} /PS ₃ 4,755	13.34	0.0430	0.983
PP_pup/PS_ 19,417	14.95	0.0471	0.998
PP _{pure} /PS ₅ 86,438	15.35	0.0456	0.989
PP _{pure} /PS ₆ 380,000	14.70	0.0374	0,991
Polypropylene & Polydisperse Polystyrene			
PP _{purc} /PS 115,500	23.81	0.0708	0.993
Polypropylene & Blends of Polystyrene			
PP _{pure} /PS _a	24.64	0.0825	0.992
PPpure/PSb	19.89	0.0740	0.996
PP'pure/PSc	13.74	0.0433	1.000
Monodisperse Polystyrene &			
Monodisperse Polyethylene			
PS 4,755 / PE, 1,870	11.31	0.0325	0.981
PS 4,755 / PE ₂ 1,050	12.00	0.0402	0,995
PS 4,755 / PE ₁ 680	11.99	0.0430	1.000
Monodisperse Polystyrene & Polydisperse Polyethylene			
PS 4,755 / PE, 770	16.84	0.0728	0.999
Polypropylene and Ethylene Vinyl Alcohol Copolymer			
PP _{com} / EVOH	121.12	0.2110	0.993

Table 3.6. : Linear Regression coefficients of $\gamma = a - bT$

4. INFLUENCE OF MOLECULAR WEIGHT

4.1. Introduction

Very few studies have been reported regarding the influence of molecular weight on the interfacial tension between polymers. An increase of the molecular weight of either one of the polymers involved in the interfacial tension measurement generally results in an increase of the interfacial tension between the polymers [Wu, 1974, 1978, 1982 and Anastasiadis 1986] except for cases where the end group of the chain is highly polar. In that case, the molecular weight dependence is reversed [Koberstein 1986, Paul 1991]. The increase of interfacial tension with molecular weight is probably due to an increase of the configurational restriction on the chains. This leads to a decrease of the entropy of mixing, thus to an increase of the Gibbs free energy of mixing, which results in an increase of the interfacial tension. A more detailed theoretical analysis of the effect of molecular weight on interfacial tension is presented in Chapter 4.

Several empirical relationships between interfacial tension and molecular weight have been proposed. Two of the most important equations are discussed below.

Gaines and coworkers [1975, 1978] studied the effect of molecular weight on the interfacial tension for the systems n-alkanes/perfluoroalkane($C_{12.5}F_{27}$), poly(dimethyl siloxanes) (PDMS)/($C_{12.5}F_{27}$) or C_8F_{18} and ethanes with poly(dimethyl siloxanes) and found that the interfacial tension is proportional to $M_n^{-\frac{2}{3}}$. A similar empirical relationship between surface tension and molecular weight has been used with polymers.

Anastasiadis [1988] proposed the following relationship for the influence of molecular weight on the interfacial tension, based on measurements involving poly(dimethylsiloxane) / polystyrene (PDMS / PS), poly(methylmetacrylate) / polystyrene (PPMA / PS), polybutadiene / polystyrene (PBDH / PS) polymer pairs:

$$\gamma = C_1 - C_2 M_n^{-2} \tag{3.6}$$

where γ is the interfacial tension and C₁, C₂ and z are adjustable constants. However, Anastasiadis concluded that the value of the exponent could not be quantified. This exponent is a function of the polydispersity of the system. For polydisperse systems, the exponent is close to 0.5 which implies a weaker dependence on molecular weight. He suggested that in a polydisperse system, there is a preferential migration of the low molecular weight species to the interface. For monodisperse systems, a relationship of the order of $M_n^{-\frac{2}{3}}$ was observed. For very high molecular weights, the dependence changed to M_n^{-1} . However, Anastasiadis suggested that further experimentation was needed to fully examine the value of z.

4.2. Results and Discussion

The effect of molecular weight on interfacial tension was studied for the following polymer pairs: a) pure PP (cf. Section 2 of this Chapter) with monodisperse polystyrene with molecular weights ranging from 900 to 400,000 and temperatures ranging from 178°C to 240°C; b) monodisperse PE with molecular weights ranging from 700 to 2100 with PS₃ at temperatures ranging from 130°C to 160°C.

Figures 3.9 and 3.10 show the effect of molecular weight of PS and PE respectively on interfacial tension. The main conclusions from these experiments are summarized below. A more detailed theoretical analysis of the influence of molecular weight on interfacial tension is presented in Chapter 4.

a) At any temperature, the interfacial tension increases for both polymer pairs as the molecular weight of one of the components increases, as expected thermodynamically. A similar trend has been reported in the literature [Wu 1969, Anastasiadis 1988] for a limited range of molecular weights.



Figure 3.9: Interfacial Tension between PP_{pure} and Monodisperse Polystyrene as a Function of the Molecular Weight of Polystyrene





b) The experimental data appear to suggest that the interfacial tension between PP_{pure} and PS starts to level off when the molecular weight of PS exceeds 30,000. This value of 30,000 corresponds to the value at which entanglements start to occur for PS (Welygan [1982]). Sauer and Dipaolo [1990] observed similar behavior (leveling off) for dependence of the surface tension of poly(dimethylsiloxane) (PDMS) on molecular weight.

c) Figure 3.11 shows a plot of interfacial tensions of PP_{pure} and PS (varying the molecular weight of PS) and of PE and PS (varying the molecular weight of PE) versus $M_n^{-\frac{2}{3}}$. According to Gaines and coworkers [1975, 1978], the resulting curves should be straight lines. The interfacial tension data for PE/PS appear to follow the $M_n^{-\frac{2}{3}}$ dependence. It is interesting to note that these data were obtained over a narrow range of molecular weights, which is also the case for the data of Gaines and Coworkers. On the other hand, the interfacial tension data for PP_{pure}/PS systems do not follow the $M_n^{-\frac{2}{3}}$ rule of Gaines and Coworkers, if the whole range of molecular weight (from 980 to 380,000) is considered. It is worth noting, however, that the PP_{pure}/PS data appear to follow the $M_n^{-\frac{2}{3}}$ rule over a narrow range of low molecular weight values.

d) The hard lines of Figure 3.9 and 3.10 represent the best fit to equation 3.6 (obtained by least-square regression, the fitting was done using the mathematical tools of Sigma-Plot)

$$\gamma = C_1 - C_2 M_n^{-2} \tag{3.6}$$

where γ is the interfacial tension and C₁, C₂ and z are adjustable constants.

Table 3.7. presents the coefficients C_1 , C_2 and z for the different polymer pairs at different temperatures. It can be seen that the coefficients z for PP/PS polymer pairs are much smaller than for the PE/PS polymer pairs. This is probably due to the broad range of molecular weight for the PP/PS polymer pair.



Figure 3.11: Interfacial Tension of PP_{pure} and PS (varying the molecular weight of PS) and of PE and PS (varying the molecular weight of PE) vs. M_n^{-2/5}

Polymer Pairs	С,	C,	2
PP/PS at 186°C	70.09	70.08	0.009
PP/PS at 213°C	74.00	74.86	0.008
PE/PS ₃ at 130°C	7.20	5444	1.35
PE/PS ₃ at 150°C	6.77	3216	1.20
PE/PS ₃ at 160°C	7.57	135	0.62

Table 3.7: Coefficients of Equation 3.6.

5. INFLUENCE OF POLYDISPERSITY AND BLENDING ON INTERFACIAL TENSION

5.1. Introduction

The polymers used for industrial applications are seldom monodisperse. It is therefore, useful to study the influence of polydispersity on the interfacial tension. No experimental study on the effect of polydispersity has been published to date. Only theoretical studies have been reported [Hariharan and Kumar (1990), Broseta et al. (1990)]. These theories are reviewed in Chapter 4.

In order to understand the effect of polydispersity on the interfacial tension between polymers, the interfacial tension between PP_{pure} and bimodal blends of PS was considered. The polydisperse PP_{pure} was the same as that used for the above study of temperature and molecular weight effects. The bimodal blends of PS are blends of PS₂ and PS₄ and of PS₃ and PS₅, as described in Section 2 of this Chapter.

5.2. Results and Discussion

The interfacial tension between PP_{pure} and bimodal blends of monodisperse PS was measured as a function of the molar concentration of the bimodal blends at a temperature of 186°C. Also, the interfacial tension between 3 polydisperse bimodal blends of PS and PP was measured as a function of temperature.

Figures 3.12 and 3.13 show the interfacial tension between the bimodal monodisperse blends of PS and PP_{pure} as a function of the molar composition of the blends for both bimodal blends studied (PS₂ (M_n =1,589) and PS₄ (M_n =19,417)); and (PS₃ (M_n =4,755) and PS₅ (M_n =86,438)). The open circles represent the experimental data. The full line represents the interfacial tension calculated as if the values of interfacial tension between the bimodal blends of PS and the PP_{pure} were weighted averages of the values of the interfacial tension between the unimodal PS and the PP_{pure}. Figure 3.7 presents the interfacial tension between the bimodal blends of polydisperse PS and PP_{pure} as a function of temperature.

It can be seen from Figures 3.12 and 3.13 that the interfacial tension between bimodal blends of monodisperse PS and PP is higher (up to 20%) than the weighted average of the interfacial tension between the unimodal PS and the PP_{pure}. The difference is increasing with increasing molar fraction of the material with the highest molecular weight until it reaches a maximum for a critical molar fraction of the material with the highest molecular weight. At concentrations higher than this critical concentration the difference between the interfacial tension for bimodal blends of PS and PP_{pure} and the weighted average of the interfacial tensions between the unimodal PS and PP_{pure} becomes smaller.



Figure 3.12 : Interfacial Tension between a Bimodal Blend of PS and PP_{pure} as a Function of Molar Fraction of PS (19,417)



Figure 3.13 : Interfacial Tension between a Bimodal Blend of PS and PP_{pure} as a Function of Molar Fraction of PS (86,438)

At each temperature, the interfacial tension between the bimodal or polydisperse PS and PP_{pure} is much higher than the weighted average of the interfacial tension between the unimodal PS and PP_{pure} . In this case, the molar proportion of the component with the highest molecular weight is very small (less than 5% in each case).

The experimental observations mentioned above are in disagreement with the theoretical treatment of Broseta et al [1990] which predicts that the interfacial tension between a binary blend and another polymer is equal to the mean average of the interfacial tension between the components of the blend and the other polymer. Yet, the difference between our experimental data and the theory is unlikely to be due to experimental error or the mixing process since the experiments were repeated with different blends.

A possible explanation of the behavior observed experimentally could be the migration of the small molecules of the blends (components of the lower molecular weight) to the interior leaving the interface mainly occupied by the molecules with the higher molecular weight. Thus, the smaller molecules would not make the expected contribution to reducing the interfacial tension.

More experiments should be performed at higher temperature, to see if the same phenomena occur.

5.3. Effect of Polydispersity

The effect of polydispersity on interfacial tension was studied using two different polymer pairs: a) PP_{nur}/PS and b) PE/PS

a) PP_{pure}/PS

Figure 3.14 shows the interfacial tension between PP_{pure} and different samples of PS as a function of temperature. The data for two monodisperse (M_n =380,000 and M_n =86,438) and one polydisperse (M_n =115,500, M_w/M_n =2.84) samples of PS are presented. The data of Escudie [1986] are also plotted in this graph (the PS used by that author had a molecular weight of 220,000). However, the author does not specify the index of polydispersity.



Figure 3.14: Interfacial Tension between PP_{pure} and Monodisperse PS, PP_{pure} and Polydisperse PS as a Function of Temperature

The interfacial tension between polydisperse PS and PP_{pure} is higher than the interfacial tension between monodisperse sample of PS and PP even when the molecular weight of the monodisperse sample (380,000) is higher than the molecular weight of the polydisperse sample (115,000). The interfacial tension values between the polydisperse PS and PP_{pure} (for both the data of this work and the data of Escudie) show a stronger dependence on the temperature than the interfacial tension between the monodisperse PS and PP_{pure} .

b) PE/PS polymer pair

Figure 3.15 shows the interfacial tension between PE and PS as a function of temperature for one monodisperse sample of PE and one polydisperse PE of the same molecular weight.



Figure 3.15: Interfacial Tension between Monodisperse PE and PS Polydisperse PE and PS as a Function of Temperature

The interfacial tension between polydisperse PE and PS showed a stronger dependence on temperature than the interfacial tension between monodisperse PE and PS. At temperatures below 436 K, (intersection of the two linear regressions) the interfacial tension between polydisperse PE and PS is higher than the interfacial tension between monodisperse PE and PS. At temperatures higher than 436 K, the interfacial tension between polydisperse PE and PS is higher than the interfacial tension between polydisperse PE and PS is higher than the interfacial tension between polydisperse PE and PS is higher than the interfacial tension between polydisperse PE and PS.

A possible explanation of the above observations could be that, at low temperatures, the small molecules of a polydisperse system tend to stay in the interior and that, as the temperature increases, the smaller molecules tend to migrate to the interface to act as surfactants. Thus, it would appear that the theoretical prediction of Broseta [1990], that there is a migration of the shorter chain to the interface leading to a decrease of the interfacial tension, is only valid at high temperatures.

6. EFFECT OF COMPATIBILIZER

This section presents the interfacial tension results obtained when compatibilizers and additives (that are normally used in commercial polymers) were mixed with pure polymers.

Interfacial tension between maleate-grafted polypropylene (MAgPP) with different levels of maleation and ethylene vinyl alcohol copolymer (EVOH) was measured at two temperatures (218°C and 232°C). The effect of maleation on the interfacial tension between MAgPP and EVOH was studied. Interfacial tension values between pure PP and EVOH and commercial PP and EVOH were obtained under the same conditions. The difference obtained in the values of the interfacial tension for those two polymer pairs is discussed.

6.1 Background

6.1.1 Introduction

The last decade has seen a large increase in the use of blends of polymers that are normally not compatible. These blends combine the properties of different polymers to obtain a material with optimized properties.

When two polymers are in contact, it is important that the interface be small enough for the material to be considered macroscopically homogeneous; the interface should also be strong enough to assimilate stresses and strains without disruption of the morphology. The polymers that constitute the blend should be at least partially miscible, in order to result in a finer morphology that produces better mechanical properties. This partial miscibility is directly dependent on the interfacial region between the components of the blend. Good compatibility is achieved when the interfacial tension between the components of the blends is low. However, partial miscibility is seldom achieved in polymer blends. Thus, it becomes necessary to enhance the compatibility between the components of the blend. It should be noted that the compatibilisation of polymer blends does not induce thermodynamic miscibility of the polymers, which would cause the polymers to exist in a single homogeneous phase [Coran and Patel 1983].

There are several methods to improve the miscibility of polymers (use of compatibilizers, mechanical treatment...) in a blend [Utracki 1989]. Two of the most popular methods to achieve better compatibility involve the addition of a block (or graft copolymer) to a blend and the functionalization of one of the components of the blend.

6.1.1.1. Addition of Copolymer

The presence of a block or graft copolymer can lower the interfacial tension between two polymers, thus increasing their miscibility. If a third component (a block copolymer) with segments capable of specific interaction or chemical reaction with the blend constituents is added to the blend, this third component tends to concentrate at the interface and act as an emulsifier as shown in Figure 3.16.



Figure 3.16 : Addition of block or graft copolymer to immiscible polymer pair A-B [Paul 1978]

6.1.1.2 Functionalization

It is possible to modify a polymer by grafting an additive on the backbone chain. This modified polymer will possibly form a better blend with a second polymer that originally was not compatible with the unmodified first polymer. This is called functionalization. Specific interactions or chemical reactions probably occur at the interface and a copolymer is formed in situ, acting then in the same manner as a separately added copolymer. Figure 3.17. shows an example of compatibilisation by functionalization.



Figure 3.17 : In situ Formation of graft copolymer as a result of reaction between OH and MAH groups

It has been shown by several researchers [see references quoted in 6.1.2] that the addition of a compatibilizer (either block or graft copolymer or functionalization of one of the polymers in the blend) improves the mechanical properties of the blend. Unfortunately, the compatibilizers and/or functionalised polymers are expensive. Also, in some cases, some properties such as permeability deteriorate at high compatibilizers levels. It is therefore of interest to maximize the efficiency of these additives. Thus, it is very important to understand the influence of the compatibilizers on interfacial tension between polymers.

6.1.2 Influence of Compatibilizers on Blend Properties

Barentsen [1970,1973,1974], Heikens [1977] and Locke [1973] studied the enhancement of properties of polystyrene (PS) / low density polyethylene (LDPE) blends through the addition of a graft copolymer. They reported an improvement in the mechanical properties of the blend and a finer morphology of the same as the

concentration of the graft copolymer was increased. The improvement of the properties is shown in Table 3.8.

Blend PS / LDPE /Graft	Yield Stress (kg/cm ²)	Elongation at Yield (%)	Size of Disperse Phase (µm)
75/25 / 0.00	170	2.3	2.60
75 / 23.5 / 1.25	204	3.2	1.48
75 / 17.5 / 7.50	262	2.8	0.83

 Table 3.8 : Enhancement of PS/LDPE blend properties through addition of graft copolymer (Locke 1973)

Fayt and Jerome [1981,1982,1986,1987] studied the compatibility between low density polyethylene (LDPE) and polystyrene (PS). They showed that the compatibility between PE and PS could be improved by adding as little as 2% by weight of poly(butadiene-b-styrene) (b stands for block). This amount was sufficient to reach the optimum improvement in physico-mechanical properties of the PS/LDPE blend. They also showed that a further increase in the compatibilizer concentration did not improve further the properties of the blend. Fayt and Jerome also studied the composition of the interface using a marked block copolymer (with OsO_4). They studied the morphology of the blend with T.E.M (Transmission Electron Microscopy) and were able to observe that the block copolymer to the interface as follows: due to the incompatibility between the two polymers, the only place for the copolymer to be is at the interface, where each of the two blocks of the copolymer is compatible to one of the two homopolymers forming the blend.

Fayt and co-workers [1986] also reported results for PS / polymethyl metacrylate (PMMA) modified by poly(styrene-b-isoprene-b-methyl metacrylate). In that case, at least 5% of additive was needed to obtain a finer morphology.

Ouhadi et al [1986] studied the effect of adding poly(methyl metacrylate-b- α methyl styrene) to polyvinylidene fluoride (PVF₂) / poly(α -methyl styrene) (P α MS) blends. The conclusions obtained were similar to those reported above. They showed that

a small addition (2%) of block copolymer greatly reduces the particle size of the dispersion, improves the interfacial adhesion and the mechanical properties of the blend. A further increase of the additive concentration to 12% improves only slightly the properties, whereas a further increase of the concentration did not show any improvement.

Yamamoto [1990] developed new techniques to prepare block and graft copolymers by using special types of organic peroxides. He studied the effect of adding these copolymers to incompatible polyblends and reported, for more than 20 polymer pairs, improvement of mechanical properties and a finer morphology.

Numerous studies have been conducted on the effect of maleation of one polymer to improve the compatibility of the components of the blend. Maleation of one polymer consists in grafting maleic anhydride (MAH) on the backbone of one of the polymers in the blend. Ide and Hasegawa [1974] studied the effect of maleating polypropylene (PP) in a Nylon 6 (N₆)/PP blend. They showed that the mechanical properties of the system maleated polypropylene (MAgPP)/N₆ improved substantially. For example, the yield strength increased from 3300 lbf/in² to 5500 lbf/in² and the elongation at break from 5 to 28 % when the concentration of MAH grafted on the backbone of PP increased from 0 to 3.6 %.

Hozhabr [1991] showed that maleation of polypropylene improves the mechanical properties of blends of polypropylene / poly(ethylene-vinyl alcohol) copolymer. He showed that when the concentration of MAH was increased from 0 to 0.2 %wt, the blends acquired a finer morphology. He also showed that there was a substantial improvement in the mechanical properties of the blends when the concentration of MAH was increased up to 0.2 %wt. No further improvements were observed if the concentration was further increased. The impact strength increased from 330 N/mm for a MAH level of 0.07% to above 500 N/mm at 0.2% and decreased to about 440 N/mm as the concentration of MAH was further increased.

Several other authors used grafting with maleic anhydride for compatibilization, among them Spector [1990] who used maleated polystyrene to improve the properties of PS/EVOH blends, Kim [1991] who studied the morphological changes of polyethylene/nylon-6 blends and Carrot [1991] who used it to improve the properties of polyamide-6 with ABS. Maleic anhydride has also been used to improve adhesion capacity of polymers on metal. More details can be found in Shultz et al [1989]. In the above cases, it is believed that the enhancement of the blend properties is due to a chemical reaction of MAH with Nylon 6(N_6 in the case of Ide) or with EVOH (in the other cases) at the interface, as shown in Figures 3.18a and 3.18b







Figure 3.18b : reaction between MAH and EVOH

Xanthos et al [1990] studied PP / polyethylene teraphtalate (PET) blends and evaluated the effect of functionalizing the polypropylene with reactive acrylic acid. They observed a finer morphology with PP-g-AA (polypropylene grafted with acrylic acid) than obtained with PP.

Other additives were used to promote the properties of blends. Favis and Willis [1988,1990] investigated the influence of the addition of an ionomer (Surlyn 9020 from Dupont) on the morphology of polyolefin/polyamide blends. They showed that the addition of only 0.5 wt % of an ionomer to the blend would reduce the dimensions of the disperse polyamide by a factor of three. However, a further increase of the concentration of compatibilizer did not lead to further improvement.

It can be seen from all the above studies that a small concentration (about 0.5% wt) of compatibilizing additive can improve the mechanical properties of the blend and produce a finer morphology. However, when the concentration of additive is further increased, no additional improvement in the morphology or of the mechanical properties is observed.

6.1.3 Effect of Compatibilizers on Interfacial Tension

The use of block (or graft) copolymer or the compatibilisation of one of the components in polymer blends has been shown to modify the morphology and improve the mechanical properties of blends. The mechanisms responsible for the improvement of these properties in multi-phase systems are not clearly understood yet. Improved knowledge of these mechanisms requires more detailed information about the interface region. It is, therefore, of interest to study the effect of compatibilizers on interfacial tension between polymers. Unfortunately, only few studies have been conducted to understand the effect of compatibilizers on interfacial tension.

Anastasiadis [1987] showed that by adding as little as 1.29 wt % blend copolymer poly(styrene-b-1,2 butadiene) (P(S-b-B)) to a polystyrene/polybutadiene system, it is possible to reduce the interfacial tension by around 40%. When the copolymer concentration is further increased, no decrease of the interfacial tension was found. This, according to Anastasiadis, was an indication of a saturation of the interface with the copolymer.
Gaillard and coworkers [1980,1982] studied the effect of adding a block copolymer poly(styrene-b-butadiene) to a polystyrene-polybutadiene-styrene ternary system. The interfacial tension first decreased with increasing the amount of copolymer and then leveled off.

Patterson et al [1971] studied the effect of functionalizing poly(dimethyl siloxane) (PDMS) on the interfacial tension between PDMS and poly(oxyethylene-b-oxypolypropylene) (POE-PO). The interfacial tension between PDMS and POE-PO is reduced by 63% with the addition of 10% carboxyl groups to the alkyl side chains of the PDMS molecules. However, doubling the number of carboxyl groups did not produce further reduction the interfacial tension.

Wu [1987] reported a 40 times reduction of the interfacial tension between nylon and ethylene-propylene (EP) rubber using modified EP. However, he did not measure the interfacial tension directly but inferred it from the size of the dispersed drop in the blend.

Xanthos et al [1990] indicated that the interfacial tension between polypropylene (PP) and polyethylene terephtalate (PET) was reduced four times when the PET was functionalized with acrylic acid. However, the interfacial tension values reported were determined using empirical relations relating the interfacial tension to the surface tensions of the components [Wu 1971,1974].

Chen et al [1991] measured the interfacial tension between different polymer pairs before and after functionalization using the pendant drop method. For all the cases studied by these authors, the interfacial tension was shown to decrease when one of the polymer was functionalised.

Many researchers have studied the improvement of blend properties involving polypropylene by maleation of the PP. The influence of maleation on the interfacial tension of blends involving polypropylene has not been reported yet. The results presented in the present work show the influence of maleation of polypropylene (PP) on the interfacial tension between PP_{com} and ethylene vinyl alcohol copolymer (EVOH) as a function of the weight percentage of the maleic anhydride grafted on the polypropylene; the data are given below.

6.2 Experimental Conditions

The interfacial tension values between maleated commercial polypropylene (MAgPP) with different levels of maleation and ethylene vinyl alcohol copolymer (EVOH) were measured at a temperature of 218°C. The level of maleation varied from 0.067 wt % to 0.262 wt %, as shown in Table 3.4. Interfacial tension between MAgPP₂ and EVOH was also measured at a temperature of 240°C.

The interfacial tension between MAgPP and EVOH was determined with both the pendant drop and the spinning drop instruments described in Chapter 2, for maleation levels zero and 0.09 wt % (PP and MAgPP₂). For the other levels of maleation, the interfacial tension between MAgPP and EVOH was determined with the spinning drop instrument. It was not possible to use the pendant drop device because: i) thermal degradation of MAgPP₁ took place before equilibrium was reached, and

ii) the maleated polypropylene was brown for $MAgPP_x$ with x>2.

When performing the experiments with the spinning drop devices, it was necessary to increase the rotation speed of the tube at the beginning of the experiment, for MAgPP, with x>2, in order for the equilibrium of the drop to be obtained before thermal degradation took place. The measurements of interfacial tension were carried out at a speed of 16,000 rpm, and it was necessary to use speeds up to 20,000 rpm at the beginning of the experiment.

6.3 Results

The values of the interfacial tension between MAgPP and EVOH are presented in Table 3.9. The two techniques gave comparable results, as can be seen for EVOH/PP and EVOH/MAgPP₂ at 218°C.

Figure 3.19 shows the interfacial tension between MAgPP and EVOH as a function of the level of maleation. The open points represent the experimental data and the continuous line the best fit obtained by linear regression.

Material	Maleation Content (% wt)	T (°C)	Interfacial Tension (dyn/cm) ¹	Interfacial Tension (dyn/cm) ²
EVOH/PP	0	218	18.6±0.4	18.4±0.9
EVOH/PP	0	232	-	14.8±0.1
EVOH/				
MAgPP	0.067	218	17.9±0.2	-
EVOH/			(ĺ
MAgPP ₂	0.098	218	16.7±0.2	16.3±0.4
EVOH/				
MAgPP ₂	0.098	232	-	12.0±0.2
EVOH/				
MAgPP ₃	0.153	218	13.1±0.2	-
EVOH/				
MAgPP	0.168	218	11.5±0.2	-
EVOH/				
MAgPP,	0.222	218	12.3±0.1	-
EVOH/				
MAgPP ₆	0.262	218	11.9 ± 0.3	-

Table 3.9: Interfacial tension between MAgPP and EVOH.

1: Measured with the spinning drop apparatus

2: Measured with the pendant drop apparatus

The interfacial tension decreases as a function of temperature as seen with EVOH/PP and EVOH/MAgPP₂. The interfacial tension between MAgPP and EVOH at 218°C decreased linearly with increasing % wt of MAH up to 0.17 % wt. The equation describing the best fit of the data obtained by least square regression is given below:

$$\gamma = 19.4 - 0.92C$$
 $r^2 = 0.94$ (3.8)
T = 218° C

where γ is the interfacial tension and C is the maleation content (% wt).



Figure 3.19: Interfacial Tension between MAgPP and EVOH as a Function of Maleation Level

The maximum decrease of interfacial tension was found when 0.17 % wt of maleic anhydride was added to the polypropylene. When the maleation content of PP was higher than 0.17 % wt the decrease of interfacial tension leveled off. This suggests a saturation of the maleic anhydride. The corresponding decrease in interfacial tension was about 35%. This decrease in the interfacial tension was of the same order of magnitude as that observed by Anastasiadis [1988] for systems of PS/PBD and copolystyrene/polybutadiene as copolymer.

The results shown above are in agreement with the results reported by Hozhabr [1991] for the morphological and mechanical properties of MAgPP and EVOH blends. He reported a finer dispersion for blends of MAgPP and EVOH than for blends of PP and

EVOH for concentrations of MAH up to 0.2 % wt. Further increase in the maleation concentration did not result in further improvement of the blend.

It has been suggested previously by other researchers that a chemical reaction between MAgPP and EVOH occurs at the interface, as shown in Figure 3.18.b. In order to verify such hypothesis and the eventual migration of the maleic anhydride group towards the interface, analysis of the interface between MAgPP and EVOH was conducted. This is the subject of the next Section.

The maleic anhydride groups appear to act as surfactants. The maleic anhydride groups probably migrate towards the polar material (EVOH), as shown schematically, in Figure 3.20. The interfacial free energy between MAgPP and EVOH is then reduced.



Figure 3.20 : Migration of Maleic Anhydride Group towards Interface

The interfacial tension between EVOH and PP decreased by 11.4 % at 218°C when 0.098 % wt of maleic anhydride was grafted on the polypropyiene and by 18.9 % at 232°C, indicating a faster migration of the maleic anhydride towards the interface or a more efficient reaction at higher temperatures, as expected.

6.4. Influence of other Additive, already present in Commercial PP

The interfacial tension between pure PP and EVOH was compared with the interfacial tension between commercial PP and EVOH at 218°C and 232°C. This comparison was conducted in order to study the influence on the interfacial tension of other additives already present in the commercial PP. Table 3.10 shows the values of the interfacial tension for this polymer pair and commercial PP and EVOH at the two temperatures.

	218°C	232°C
Pure PP/EVOH	16.30±0.35	10 70±0.40
Commercial PP/EVOH	18.40±0.93	14.83±0.02

Table 3.10 : Interfacial Tension between Pure and Commercial Resins

It can be seen that, for both temperatures, the interfacial tension between commercial PP and EVOH is greater than the interfacial tension between pure PP and EVOH. This difference is too large to be due to the small differences in the molecular weight or polydispersity of the two samples. However, it may be attributed to the presence of additives such as anti-static agents, coupling agents and others in the commercial polypropylene.

Different methods of analysis were used to determine the differences in composition between pure PP and commercial PP.

a) Elemental analysis was performed on both the commercial and pure PP to verify the presence of those additives. The method was not accurate enough to detect any difference between the two resins. The results of the analysis are summarized in Table 3.11.

	% C	% H	% Residue
Pure PP	85.75	14.08	0.17
Commercial PP	85.56	14.33	0.11

Table 3.11 : Results of Elementai Analysis

b) X ray analysis was performed on both pure and commercial samples. Figure 3.21 shows the spectrum of the commercial PP. It can be seen that besides carbon (and hydrogen which cannot be detected by this method), it contains aluminum, silicon and chlorine. Both the aluminum and chlorine are contaminants from the sample holder and the glue to mount the sample. The spectrum of pure PP did not show the presence of silicon.



Figure 3.21 : X-ray Spectrum of Commercial PP

c) Both samples were analyzed by Electron Spectroscopy for Chemical Analysis (E.S.C.A). It was observed that pure PP contained only carbon and hydrogen, whereas the commercial PP contained silicon and oxygen in the form of SiO_x or SiO_2 is commonly used as antistatic agent. The detailed results of this analysis are reported in the next section. Neither chlorine nor aluminum was detected in this analysis. In this test, silver glue was used to hold the sample

On the basis of the above analysis, it seems that the commercial polypropylene contained SiO_x , which probably contributed to increasing the interfacial tension between PP and EVOH.

7. SURFACE ANALYSIS OF INTERFACES

7.1 Objectives of Surface Analysis Studies

7.1.1 Analysis of Compatibilizer

It has been shown in the previous section that the interfacial tension between polypropylene (PP) and ethylene vinyl alcohol copolymer (EVOH) could be reduced by 35% if 0.17% wt maleic anhydride acid was grafted on the backbone of the PP. It was suggested that this decrease of interfacial tension was due to a chemical reaction between the maleic anhydride group and EVOH at the interface. In order to verify this hypothesis, it was decided to conduct surface analysis of PP_{com} , MAgPP and EVOH before and after they were in contact.

One of the best methods to analyze polymer surfaces, in order to study the effect of composition, is the Electron Spectroscopy for Chemical Analysis (E.S.C.A). Some of the advantages of this technique are outlined below:

i) It can provide information on the elemental compositions of the surfaces and of the chemical state of the elements over a depth of a few Angstrom.

ii) It is one of the few non-destructive methods for analyzing polymer surfaces.

7.1.2. Difference between Pure and Commercial PP

It was reported in the previous section that the interfacial tension values between commercial PP and EVOH and between pure PP and EVOH differed by 10% to 27%, according to the temperature used for the experiment. X-ray analysis was then performed to identify chemical differences between the two samples; the presence of silicon was detected, as mentioned earlier.

In order to understand the influence of silicon on the interfacial tension, E.S.C.A. analysis of the pure and commercial samples of PP was performed, before and after contacting EVOH. The two resins were also studied using a Scanning Electron Microscope.

7.1.3. Observation of the Interfaces between Polymers

The interfacial tension data reported in this study can vary from 3 to 18 dyn/cm depending on the polymer pair used. It was therefore of interest to see if any difference (morphological or physical) could be observed in the interfacial regions. In order to do this, the samples were studied using a Scanning Electron Microscope (S.E.M).

<u>7.2. Surface Analysis of Polymers</u>

7.2.1. Existing Methods for Surface Analysis

Several techniques have been developed and employed to perform surface analysis such as Fourier Infrared Spectroscopy (F.T.I.R), Auger Electron Spectroscopy (A.E.S), Secondary Ion Mass Spectroscopy (S.I.M.S), Electron Spectroscopy for Chemical Analysis (E.S.C.A), Ion Scattering Spectroscopy (I.S.S) and others. A good review of these methods can be found in [Clark 1977].

The only non-destructive proven methods which can be used for polymer surface analysis are:

i) F.T.I.R. which averages the composition of the polymers from the surface down to 200-3000 nm. ii) E.S.C.A.: the depth of observation is 1-2 nm using this method.

For this work, the use E.S.C.A. was prefered, for its sensitivity, its quantitative potential and non distructive nature. It was not possible to use F.T.I.R. because the sample would need to be a film.

7.2.2. Electron Spectroscopy for Chemical Analysis (E.S.C.A)

7.2.2.1. Description of the Method

A good review of E.S.C.A can be found in Briggs [1983]. During analysis by E.S.C.A, the surface to be analyzed is bombarded with a mono-energetic beam of X-rays. The photons of the X-rays interact with the molecules of the surface, and electrons are ejected. The electrons are collected on special grids where their binding energy will be determined. The binding energy is specific to each element and to the chemical bonds presents on the surface. The analysis of the energy spectrum provides a method of conducting chemical analysis.

7.2.2.2. E.S.C.A. and Polymers

E.S.C.A is an excellent tool for polymer surface analysis [Clark 1973-1977]. Polymer surfaces are known to be relatively stable when bombarded by X-rays during E.S.C.A. analysis. However, because the X-ray gun in some instances becomes very warm, thermal degradation can be observed and yellowing of the sample can occur. Care should be taken to avoid exposing the sample for extended periods of times. When analyzed by E.S.C.A, polymeric materials can charge electrically (because they are good insulating material). This results in a shift of the emission peaks which should be taken into consideration when performing the analysis of polymers.

7.2.3. S.E.M.

Numerous morphological studies of polymer blends have been conducted using S.E.M [Fayt et al 1981, 1982, 1986, 1987, Lohfink 1990, Arghyris 1991, Hozhabr 1991 and others]. The instrument can provide enlargements up to 200,000 times. Some care should be taken to avoid thermal degradation using S.E.M; long exposure times and high voltage should be avoided.

7.3. E.S.C.A. Analysis

7.3.1. Experimental Procedures

Four different polymers were studied by E.S.C.A: pure PP (PP_{pure}), commercial PP (PP_{com}), MAgPP₂, EVOH. The four polymers were analyzed before an experiment for the determination of interfacial tension and after contacting the other complementary polymer (PP_{nure} /EVOH, PP_{com} /EVOH, MAgPP₂/EVOH) in the pendant drop apparatus.

The preparation of the polymers for E.S.C.A. analysis was as follows:

a) Polymers before an interfacial tension determination. the polymers were analyzed by E.S.C.A. as received, the PP_{com} , the MAgPP₂, the EVOH were in pellet form and the PP_{pure} in powder form and were dried in a vacuum oven for 24 hours at 70°C prior to experiment.

b) Polymers after an interfacial tension determination at 232°C. In the experiment for interfacial tension determination, the EVOH was always the drop inserted in a matrix of either PP_{pure} , PP_{com} or MAgPP₂. After the experiment, the sample was cooled and the drop was removed from the matrix, cut and analyzed. The matrix was also cut and analyzed. The analyses were conducted in the interface region. All the composition results discussed in this section refer to the interface region.

The malcated resin studied with E.S.C.A. was $MAgPP_2$, because it was the only one that could be used with the pendant drop method. The geometry of the spinning drop, which was used for the other level of maleation was not adequate for E.S.C.A. measurements. The surface analysis was done at the Escalab of Ecole Polytechnique de Montreal in the Physics Department.

The polymeric samples were glued on the sample holder with a silver glue in order to avoid contamination by silicon. It has been shown that, very often, silicon present in the glue to mount the samples migrates to the surface of the samples [Clark 1977]. The X-ray source used was a 12 kV source of MgBE 20 mA. The vacuum used was 9 10⁻⁹ Torr.

7.3.2. Results and Discussion

Composition analysis was conducted first. A typical composition spectrum obtained for EVOH is shown in Figure 3.22. The main peaks are identified. Table 3.12 gives the atomic percentages of the elements contained in the different polymers.

	С	0	Si	N	S	F
PP pure	100	-	-	-	-	-
PP com	93.1	6.4	0,5	-	-	-
MAgPP,	94.5	2.8	2.5	-	0,2	-
EVOH	74.9	21.1	0.7	1,2	2.1	-
PP pure /						
EVOH	92.8	4.8	1.6	-	0.8	
PP com /	89.9	6.3	3.2	-	0.6	-
ЕVOH	91.4	5.5	2.4	-	0.7	
MAgPP ₂	87.2	7.1	5.3	-	0.4	-
/ EVOH	86.7	6	7.3		-	
EVOH /	86.9	8.6	4.6	-	-	-
PP pure	88	12	-	-		
EVOH /	75.1	20.2	3.9	-	0.3	0.4
PP com	72.7	22.3	2.5	-	2.5	_
EVOH /	77	18.9	2.1	-	2.0	-
MAgPP,	73.7	18.6	5.3	1.1	1.3	-

Table 3.12. : Elemental Analysis of the Samples (in %)

In Table 3.12, the first polymer listed in the first column is the one analyzed. For example, in the EVOH/PP_{com} case, the polymer analyzed is EVOH which has been in contact with PP_{com} . Two different samples were analyzed for each of the polymers used in the interfacial tension determination experiments, as shown in the Table.



Figure 3.22: E.S.C.A. Spectrum of Ethylene Vinyl Alcohol Copolymer

It can be seen from Table 3.12, that there is good reproducibility in terms of compositions of the duplicate samples. The analysis of the polymeric samples studied here showed the presence of three elemental impurities: N, F and S. The first two elements N and F were present in very small quantities in 1 or 2 samples. The presence of S seems to be due to a contamination of one of the two batches used for the analysis (the first batch was analyzed in October 1992 and the other one in December 1992). The multiplex analysis of the elements C, O and Si did not show any bonding with N, F or S. Therefore, these three elements (N, F, and S) were considered as impurities. For all the samples studied here, the analysis of the silicon multiplex shows that the silicon is in SiO₂ form.

7.3.2.1. Differences between Pure and Commercial PP

It has been observed that the interfacial tension value between PP_{pure} and EVOH was lower than the interfacial tension between PP_{com} and EVOH. It was therefore of interest to study the difference in surface composition between PP_{pure} and PP_{com} before and after contact with EVOH.

Table 3.12 shows the difference in composition between the pure and commercial polypropylene (PP). The pure PP contains just carbon and hydrogen, whereas the commercial PP contains also oxygen and silicone, which could be an anti-static or a nucleating agent or a residue of the catalyst used for polymerization. The multiplex of O and C showed that 95% of the O present was in the form of SiO_x and that 5% of the O was linked to carbon.

It can be seen from Table 3.12 that the elemental composition of the PP_{com} before and after contacting EVOH is the same. The elemental analysis of PP_{pure} after contacting EVOH shows the presence of oxygen that did not exist in the PP_{pure} before contacting EVOH. Also, it can be seen from Table 3.12 that EVOH after contacting PP_{pure} contains less oxygen than before contacting. This could be interpreted as a loss of oxygen from the EVOH to the PP_{pure} . In order to further investigate this possibility, it was decided to perform multiplex analysis of the O and C elements, for EVOH and PP_{pure} before and after contacting each other.

Table 3.13, shows a comparison between the multiplex of O of EVOH before and after contact with PP pure or PP commercial. The numbers in Table 3.13 indicate the percentage of the type of bonding involving O. Σ represents the atomic percentage of O as presented in Table 3.12 (in other words, Σ =21.1 means that O represents 21.1% of the total components of the sample).

Emission Energy (eV)	529.9 C-O	533.6 OH
Polymer		<u>=C</u>
ΕVOH (Σ=21,1)		100
EVOH after contact with PP commercial (Σ=22.3)	5	95
EVOH after contact with PP pure (Σ =18)	25	75

Table 3.13 : Multiplex analysis of O(1s) for EVOH

It can be seen from the results that the number of OH bonds decreases =C

when EVOH is in contact with PP_{pure} . The multiplex analysis of C for EVOH corroborated this conclusion. The multiplex analysis of C and O for PP_{pure} after contacting EVOH showed that the totality of the oxygen was bonded to carbon either as C-OH or C-O-C.

The results presented here indicate a loss of oxygen from the EVOH to the PP_{pure} . Either there was a contamination of EVOH to PP_{pure} or this suggests that a chemical reaction between EVOH and PP_{pure} occurred at the interface and that after separation, the O remained attached to PP_{pure} . This chemical bonding is likely to be responsible for the lower value of the interfacial tension between EVOH and PP_{pure} when compared to EVOH and PP_{even} .

7.3.2.2. Influence of the Compatibilizer

i) Analysis of PP_{com} and MAgPP₂

The results presented in Table 3.12 do not show any composition differences of the commercial polypropylene before and after contacting with EVOH. This is an indication that no chemical reaction occurred between PP_{com} and EVOH.

Table 3.14 presents the results of the analysis of C/O ratio in the $MAgPP_2$ before and after contacting EVOH.

Table 3.14 : C/O for MAgPP₂

C/O before contacting EVOH	C/O after contacting EVOH	
33.8	12.3	

It can be seen that the C/O ratio decreased after contacting EVOH. This suggests either a migration of the maleic anhydride groups towards the interface or a gain of O from EVOH and most likely a combination of the two. It should be noticed that the resins contained similar amounts of silicon. It was decided to further investigate the O bonds to verify if there is a chemical reaction occurring at the interface.

Table 3.15 shows the results of the multiplex analysis of O for MAgPP₂ before and after contacting EVOH. As in the case of Table 3.13, the numbers refer to a percentage of the type of bonds involving O. Σ represents the atomic percentage of O as presented in Table 3.12.

Emission Energy (eV) Polymers	529.7 C-O	531.5 SiO _x	532.7 SiO ₂	533,8 OII =C
MAgPP ₂ bcfore contact with EVOH (Σ=2.8)	9	23	68	-
MAgPP ₂ after contact with EVOH (Σ=7.1)	-	-	94	6

Table 3.15 : Multiplex Analysis of O(1s) for MAgPP₂

It seems that, for both samples, most of the oxygen is linked to Si. The remaining oxygen is incorporated in C-O bonds before the sample was in contact with the EVOH and in a OH, once the sample has been in contact with EVOH. The analysis of the carbon $=\dot{C}$

multiplex corroborated this conclusions. It was observed that, before contacting, no C-OH was present and that after the contact, C bonds in the form of C-OH were detected.

The above two results confirm the possibility of a chemical reaction as shown below. This reaction involves a decreases of the C-O bonds and an increase in the OH bonds. $=\overset{1}{C}$



As mentioned before (last section), this chemical reaction is likely to be responsible for the decrease of the interfacial tension when comparing PP_{com} and $MAgPP_2$ with EVOH. As expected, chemical bonds are formed between the two polymers and the interfacial tension decreases.

ii) Analysis of EVOH

The analysis of EVOH is a complement of the analysis reported above. The same conclusions as for PP_{com} and $MAgPP_2$ should be reached for EVOH. Table 3.16 shows the analysis of the C/O ratio for the EVOH before and after contacting PP_{com} or $MAgPP_2$.

Polymers	C/0	
EVOH	3,55	
EVOH after contact with PP commercial	3.72	
	3.26	
EVOH after contact with MAgPP ₂	4.07	
	3.96	

Table 3.16.: Evolution of the C/O ratio for EVOH

Tables 3.17 and 3.18 show the results of the analysis of the multiplex for O and C of EVOH before and after contact with PP or $MAgPP_2$.

Emission Energy	529.9	533.6
Polymer		1 =C
EVOH(Σ=21,1)		100
EVOH after contact with PP		
commercial (S=22.3)	5	95
EVOH after contact with		
MAgPP ₂ (Σ=18.9)	100	-

Table 3.17 : Multiplex Analysis of O for EVOII

Table 3.18 : Multiplex Analysis of C for EVOH

Emission Energy	285,00 C-C	286,6 C-OH or	287.9 [°] C=0
Polymer		<u> </u>	
ΕVOH(Σ=74.9)	26	52	22
EVOH after contact with PP commercial (Σ=72.7)	67	29	-
EVOH after contact with MAgPP ₂ (Σ=77)	82	18	-

It can be seen from both Tables that the number of carbons chemically bonded to oxygen are less after contact with $MAgPP_2$ which can be due to a loss of oxygen to $MAgPP_2$. The multiplex analysis of C for EVOH shows that the number of C-OH bonds decreases. This indicates that the chemical reaction depicted above probably occurs.

7.4. S.E.M. Analysis of the Samples

Interface regions between $PP_{cont}/EVOH$, MAgPP/EVOH and PP_{pure}/PS were observed with a scanning electron microscope. The regions observed were the interfaces between a pendant drop and the matrix obtained when performing an interfacial tension measurement.

Pendant drops of one polymer in another were formed; the mechanical equilibrium of the drop was reached, and the sample was left to cool to ambient temperature. Then the polymers were cut at the region of the interface. The slabs of polymer to be observed were then mounted on S.E.M. sample holders. The samples were then coated with colloid graphite and placed in a sputtering device to be coated with an alloy of Pd-Au. This coating was needed in order to avoid charging of the polymer samples, since they are non-conductive. The samples were observed with an accelerating voltage of 15 and 20 kV and at a working distance of 39 mm.

Figures 3.23 and 3.24 show a drop of EVOH in PP commercial and a drop of EVOH in MAgPP₂, respectively, after cooling from 223°C to ambient temperature. It seems from these photographs that the interface of EVOH with MAgPP₂ is less "visible" than the interface between EVOH and PP. Figures 3.25 and 3.26 show the interface between the same polymers and it seems that there is a gap of about 1 μ m between the EVOH and the PP whereas the EVOH and the MAgPP₂ seem to be in very close contact. These observations were uniform all around the drop. Unfortunately, it was not possible to see the interface between EVOH and MAgPP₂ at higher optical enlargement because of the lack of resolution of the microscope (see Figure 3.27). However, there is an indication of better adhesion between EVOH and MAgPP₂ than between EVOH and PP. Therefore, it appears, as expected, that a lower value of interfacial tension between two polymers leads to better adhesion between the resins. In general, these findings tend to support the chemical reaction hypothesis proposed above between EVOH and MAgPP₂.



Figure 3.23: S.E.M. Image of a Drop of EVOH in PP_{cont}



Figure 3.24: S.E.M. Image of a Drop of EVOH in MAgPP₂



Figure 3.25: S.E.M. Image of a Drop of EVOH in PP_{con}



Figure 3.26: S.E.M. Image of a Drop of EVOH in MAgPP₂



Figure 3.27: S.E.M. Image of a Drop of EVOH in MAgPP₂

8. CONCLUSIONS

The main conclusions of this chapter are outlined below

8.1. Effect of Temperature

a) The interfacial tension between polymers decreased linearly as a function of temperature for every polymer pair studied in this work. The temperature coefficient $\partial \gamma / \partial T$ was between 0.003 and 0.008 dyn.cm⁻¹.°C⁻¹ for the PP_{pure}/PS and PE/PS polymer pairs.

b) No systematic dependence of the temperature coefficient on the molecular weight could be established on the basis of experimental data.

c) The polydisperse systems or bimodal systems showed a stronger dependence on temperature than the monodisperse systems.

8.2. Effect of Molecular Weight and Polydispersity

d) The interfacial tension between polymers increased with increasing molecular weight of one of the polymers for the two polymer pairs studied in this work: PP_{pure}/PS and PE/PS.

e) The interfacial tension between polymers did not always follow the $M_n^{-\frac{2}{3}}$ dependence over all the range of molecular weights considered.

f) The interfacial tension follows a power law of the molecular weight, if the molecular weights are low and the range of molecular weight is narrow, in such a case, the power is close to -2/3.

g) The interfacial tension between PP_{pure} and PS appears to level off at higher molecular weight of PS.

h) The interfacial tension at a temperature of 459 K of bimodal blends of PS and PP_{pure} was higher than the weighted molar average of the interfacial tension between the equivalent unimodal PS and PP_{pure} .

i) The interfacial tension between polydisperse PS and PP_{pure} was higher than the interfacial tension between monodisperse PS and PP_{pure} for similar molecular weight for all temperature studied.

j) The interfacial tension between polydisperse PE and PS was higher than the interfacial tension between monodisperse PE and PS at temperature lower than 436 K and lower at temperatures above 436 K.

8.3. Effect of Compatibilizers

k) The interfacial tension between PP_{com} and EVOH was shown to decrease linearly with increasing concentration of maleic anhydride grafted on the backbone of the polypropylene molecules for concentrations up to 0.17% wt. A 0.17 % wt concentration of maleic anhydride grafted on the backbone of PP_{com} decreases the interfacial tension between PP_{com} and EVOH by 35%. When higher concentrations of maleic anhydride graft were used, no further decrease of the interfacial tension was observed.

1) The effect of grafting maleic anhydride on the backbone of PP_{com} on the interfacial tension between PP_{com} and EVOH was shown to be greater at higher temperatures.

8.4. Surface Analysis

m) The interfacial tension between pure PP_{pure} and EVOH was shown to be lower (30%) than the interfacial tension between commercial PP_{com} and EVOH. X-ray analysis and E.S.C.A results have shown that the commercial polypropylene contains silicon in the form of SiO_x. This is probably the reason for the increase of the interfacial tension when using commercial samples.

n) The silicon found in the different polymers was always in the form of SiO_x , SiO_2 .

o) The only difference in the elemental analysis of the pure and commercial polypropylene is the presence of silicon in the form of SiO_x in the commercial sample.

p) The elemental analysis of PP_{pure} after contacting EVOH showed the presence of oxygen that did not exist in the PP_{pure} before contacting EVOH. On the other hand, the EVOH after contacting PP_{pure} contained less oxygen than before contacting PP_{pure} . The two results suggest that either a loss of the oxygen of the EVOH to PP_{pure} or a chemical reaction between the EVOH and the PP_{pure} occurred. This chemical bonding of oxygen is likely to be the reason for the lower interfacial tension between EVOH and PP_{pure} when compared to EVOH and PP_{com} .

q) The composition of the PP_{com} before and after contacting EVOH did not change; the composition of EVOH before and after contacting PP_{com} did not change either.

r) The surface of $MAgPP_2$ after contacting EVOH was richer in oxygen than before contacting, suggesting that either a migration of the maleic anhydride to the interface or a gain of oxygen of $MAgPP_2$ from EVOH, and most probably a combination of both occurred.

s) The oxygen present in the $MAgPP_2$, not bonded to silicon, was in the form of C-O before contacting EVOH and in the form of OH after contacting EVOH. The number of

≠Ċ

OH bonds in EVOH decreases when EVOH is in contact with $MAgPP_2$. =C

The two former results suggest the occurrence of a chemical reaction at the interface as shown in Figure 3.24. This chemical bonding was likely to be the reason for the lower value of the interfacial tension between EVOH and MAgPP₂, when compared to EVOH and PP_{com} .

t) The scanning electron microscope studies showed better adhesion between the maleated polypropylene and the ethylene vinyl alcohol copolymer than between the polypropylene and the ethylene vinyl alcohol copolymer.

REFERENCES

Anastasiadis S.H., Chen J.K., Koberstein J.T., Sohn J.E., and Emerson J.A. [1986] : Polym. Eng. Sci., Vol 26, p 1410, (1986).

Anastasiadis S.H., Chen J.K., Koberstein J.T., Siegel A.F., Sohn J.E. and Emerson J.A. [1987] : J. Colloid Interface Sci., Vol 119, p55 (1987).

Anastasiadis S.H. [1988] : "Interfacial Tension of Immiscible Polymer Blends": PhD. Thesis, Princeton University (1988).

Arghyris L. [1991] : Master Engineering Dissertation, Chemical Engineering Department, McGill University, Montreal, Quebec (1991).

Barentsen W.M. and Heikens D. [1970] : J. Mater. Technol., Vol 1, p 49, (1970).

Barentsen W.M. and Heikens D. [1973] : Polymer, Vol 14, p 579, (1973).

Barentsen W.M., Heikens D. and Piet P. [1974] : Polymer, Vol 15, p 119, (1974).

Briggs D. and Seah M.P. [1983] : "Practical Surface Analysis by Auger and X-ray Photoelectron Spectroscopy" John wiley and Sons, Chichester-New York-Brisbane-Toronto-Singapore, (1983).

Broseta D., Fredrickson G.H., Helfand E. and Leibler L [1990] : Macromolecules, Vol 23, n 1, p 432, (1990).

Carrot C., Guillet J. and May J.F. [1991] : Plastics, Rubber and Composites Processing and Applications, Vol 16, p 61, (1991).

Chen C.C. and White J.L. [1991] : Antec Proceedings, p 699, (1991).

Clark D.T. [1973] : "Chemical applications of E.S.C.A."in Electron Emission Spectroscopy, ed. W. dekeyser, D. Reidel Publishing Co., Dortrecht, Holland (1973).

Clark D.T. [1975] : "Advances in Polymer Friction and Wear", Los Angeles, March 1974, ed L.H.lee, Vol 5A, Plenm Press, New York, (1975).

Clark D.T. [1976] : in "Structural Studies of Macromolecules by Spectroscopics Methods", Chapter 9, ed. K. Ivin, Wiley, London (1976).

Clark D.T. [1977] : "E.S.C.A. applied to polymers" in Advances in Polymer Science (Ed H.J. Cantow et al.) Vol 24, Springer Verlag, Berlin-Heidelberg. (1977).

Coran A.Y. and Patel R. [1983] : Rubber Chem. Technol., Vol 56, p 1045, (1983).

Cowie J.M.G. [1991] : "Polymers: Chemistry and physics of Modern Materials", 2nd Edition, Ed Blackie, USA Chapman and Hall, New York, chapt 8, (1991).

Escudie E., Graciaa A. and Lachaise J. [1986] : Materials Chemistry and Physics, Vol 14, p 239, (1986).

Favis B.D. and Willis J.M. [1990] : Proceedings of "Compalloy Europe '90" p 35, (1990).

Fayt R. and Jerome R. [1981] : Journal of Polymer Science, Polymer Letter Ed, Vol 19, p 79, (1981).

Fayt R. and Jerome R. [1981] : Journal of Polymer Science, Polymer Physics Ed, Vol 19, p 1269, (1981).

Fayt R. and Jerome R. [1982] : Journal of Polymer Science, Polymer Physics Ed, Vol 20, p 2209, (1982).

Fayt R. and Jerome R. and Tessie [1986] : Journal of Polymer Science, Polymer Letter, Vol 24, p25, (1986).

Fayt R. and Jerome R. and Tessie [1986] : in Polyblends - '86, NRCC/IMRI Polymers Symposium Series, L.A. Utracki, Ed., Montreal, Canada, (1986).

Fayt R. and Jerome R. [1987] : Polymer Engineering Science, Vol 27, No5, p 328, (1987).

Fox T.G. Jr. and Flory P.J. [1950] : J. Appl. Phys., Vol 21, p581, [1950]

Gaillard P., Ossenbach-Sauter M. and Riess G. [1980] : Makromol. Chem., Rapid Commun., Vol 1, p 771, (1980).

Gaillard P., Ossenbach-Sauter M. and Riess G. [1982] : in Polymer Compatibility and Incompatibility: Principes and Practice, K. Solc, Ed., MMI Symposium Series, Vol 2, Michigan Molecular Institute Press, Harwood Academic Publishers, New York, (1982).

Gaines G.L.Jr. and G.L. Gaines [1975] : J. Colloid Interface Sci., Vol 50, p 272, (1975).

Hariharan and Kumar [1990] : Macromolecules, Vol 23, p 3584, (1990).

Heikens D. and Barentsen W.M. [1977] : Polymer, Vol 18, p 70, (1977).

Helfand E., Somendra M., Bhattacharjee and Fredrickson G.H. [1989] : J.Chem. Phys., Vol 91 (11), p7200, (1989).

Ide F. and Hasegawa A. [1974] : Journal of Applied Polymer Science, Vol 18, p 963, (1974).

Hozhabr S. [1991] : Master Engineering Dissertation, Chemical Engineering Department, McGill University, Montreal, Quebec, [1991] Enhancement of interfacial adhesion in polymer blends of polypropylene and poly(ethylene vinyl alcohol copolymer).

Kim B.K., Park S.Y. and Park S.J. [1991] : Eur. Polym. J., Vol 27, No 4/5, p 349, (1991).

Koberstein J. [1986] : Report N^o ARO 22886.3 MS US Army Research Office Report "Characterisation of Polymeric Surfaces and Interfaces" (1986).

Legrand D.G. and Gaines G.L.Jr. [1978] : J. Colloid Interface Sci., Vol 63, p 394, (1978).

Lohfink G.R. [1990] : PhD Dissertation, Chemical Engineering Department, McGill University, Montreal, Quebec [1990]. Morphology and permeability in extruded polypropylene/ethylene vinyl-alcohol copolmer blends.

Locke C.E. and Paul D.R. [1973] : Journal of Applied Polymer Science, Vol 17, p 2597, (1973).

Locke C.E. and Paul D.R. [1973] : Journal of Applied Polymer Science, Vol 17, p 2791, (1973).

Molau G.E. [1965] : Journal of Polymer Science., A3, p 4235, (1965).

Ouhadi T., Fayt R., Jerome R. and Teyssie Ph. [1986] : Polym. Commun., Vol 27, p 212, (1986).

Ouhadi T., Fayt R., Jerome R. and Teyssie Ph. [1986] : Journal of Polymer Science, Polymer Physics, Vol 24, p 973, (1986).

Ouhadi T., Fayt R., Jerome R. and Teyssie Ph. [1986] : Journal of Applied Polymer Science, Vol 32, p 5647, (1986).

Patterson H.T., Hu K.H. and Grindstaff T.H. [1971] : J. Polym. Sci., Part C, Vol 34, p31 (1971).

Paul D.R. [1978] : in "Polymer Blends", Vol 2, pp 35-62, D.R. Paul and Seymour Newman, eds., Academic Press, New York, (1978).

Paul [1991] : "Advances in high Performance Polymer Alloys" May 21-23, Ann Arbor, Michigan, U.S.A. (1991).

Ratzsch M.T., Regener E. and Wohlfarth Ch. [1986] : Acta Polymerica, Vol 37, p 441, [1986].

Roe R.-J. [1968] : J. Phys. Chem., Vol 72, p2013, (1968).

Sauer B.B. and Dipaolo N.V. [1991] : J. of colloid and Interface Sci., Vol 144, n 2, p 527, (1991).

Schultz J., Lavielle L., Carre A. and Comien P. [1989] : J. Mat. Sci., Vol 24, p 4363, (1989).

Spector R.J. and Mc Carty S.P. [1990] : Polymer Preprints, Vol 31, Nº1, p 560, (1990). Trivedi B.C. and Culbertson B.M.[1982] : "Maleic Anhydride", 1st edn, Plenum Press, New York [1982].

Utracki L.A. [1989] : in "Polymer Alloys and Blends, Thermodynamics and Rheology" Chapt 2, Hanser Verlag, Munich, Vienna, New york (1989).

Welygan D.G. and Burns C.M. [1982] : J. Adhesion, Vol 14, p129. (1982).

Willis J.M. and Favis B. [1988] : Polymer Engineering and Science, Vol 28, N°21, p1415 (1988).

Wu S. [1971] : Journal of Polymer Science, Part C, Vol 34, p 19 (1971).

Wu S. [1974] : Journal of Macromolecular Science, Part C10, Vol 1, Marcel Dekker, New York, (1974).

Wu S. [1978] : In "Polymer Blends", D.R. Paul, and S. Newman, Eds., Academic Press, New york, (1978).

Wu S. [1982] : "Polymer Interface and Adhesion"Marcell Dekker, Inc., NY, (1982).

Wu S. [1987] : Polymer Engineering and Sciences, Vol 27, No 5, p 335, (1987).

Xanthos M., Young M.W. and Biesenberger J.A. [1990] : Polymer Engineering and Science, Vol 30, No6, p 355, (1990).

Yamamoto T. [1990] : Proceedings of Compalloy Europe '90, p 55 (1990).

Zoller P. [1950] J. of Pol. Appl. Sci., Vol 23, pp1057-1061, [1950].

THEORETICAL CONSIDERATIONS

•

1. CHAPTER_OUTLINE

In this chapter, the theoretical aspects of interfacial tension are reviewed. The results obtained in the present work and presented in Chapter 3 are compared with the lattice theories of Helfand et al [1971, 1972, 1975, 1976, 1990] and with the square gradient theories [Cahn and Hilliard 1958, Broseta et al 1990] for the influence of temperature, molecular weight and molecular weight dispersity. It was not possible to compare the interfacial tension results for the influence of compatibilisation with theoretical predictions because no theory has been developed so far to evaluate the effect of functionalization on interfacial tension. In order to estimate theoretically the interfacial tension parameter, χ , between the two polymers. A strategy is proposed to evaluate χ for the polymer pairs used in this study.

In this chapter, the thermodynamics of polymer blends and interfaces are first reviewed, then the existing theories for the prediction of interfacial tension are presented. Subsequently, the results of the present work are compared with the theoretical predictions of Helfand and Tagami lattice model [1990] and with the theoretical prediction of the square gradient theory [Broseta et al 1990]. The last part presents the conclusions of this chapter.

2. THERMODYNAMICS OF INTERFACES

Interfacial tension can be defined as the energy required to produce a unit of interfacial area at constant temperature, T, pressure, P, and number of molecules, n. The thermodynamic treatment of the interfacial tension, following the work of Adamson [1967] is discussed below.

According to the first law of thermodynamics, when a unit of interfacial area is produced, the change of energy dE of the system is:

$$d\mathbf{E} = \delta \mathbf{q} - \delta \mathbf{w} \tag{4.1}$$

where δw is the work done by the system and δq is the heat absorbed by the system.

The work done by the system can be divided into two parts:

$$\delta w = PdV + \delta w_{nonPdV} \qquad (4.2)$$

where P is the pressure, dV is the change of volume, and δW_{pompdV} is the work not associated with change of volume.

The second law of thermodynamics for reversible processes gives:

$$\delta q_{rev} = T dS \tag{4.3}$$

where T is the temperature and S is the entropy of the system.

Therefore,

$$dE_{rev} = TdS - PdV - \delta w_{nonPdV}$$
(4.4)

An expression of the Gibbs free energy of mixing is given by:

$$G = H - TS = E + PV - TS \tag{4.5}$$

The differential of the Gibbs free energy is given by:

$$dG = dE + PdV + VdP - TdS - SdT$$
(4.6)

Substituting Equation 4.4 into 4.6 gives:

$$dG_{rev} = TdS - PdV - \delta w_{nonPdV} + PdV + VdP - TdS + SdT$$
(4.7)

At constant temperature and pressure Equation 4.7 becomes:

$$dG = -\delta w_{\text{nonPdV}} \tag{4.8}$$

 $\delta w_{normally}$ can be identified as $-\gamma dA$, and accordingly:

$$\gamma = \left(\frac{\partial G}{\partial A}\right)_{T,P} \tag{4.9}$$

Equation 4.9 defines the interfacial tension as an increment in Gibbs free energy per unit increment in interfacial area. Therefore, in order to derive a theoretical expression for the interfacial tension, one has estimate the Gibbs free energy.

3. THEORIES OF POLYMERIC INTERFACES

3.1. Empirical Theories

One of the earliest theories to evaluate interfacial tension is the rule of Antonoff [1942] which states that the interfacial tension between two materials is the difference between the two surface tensions of the two materials. This empirical relationship is not valid for polymers [Wu 1974].

Later, it was proposed [Girefalco and Good 1957] that the interfacial tension between two materials could be expressed as follows:

$$\gamma = \sigma_1 + \sigma_2 - w_s \tag{4.10}$$

where w_{1} is the work of adhesion between the two phases and σ_{1} and σ_{2} are the surface tension values for the two components.

Good and Girefalco [1958,1960,1964,1970] started from Equation 4.10 to derive the following expression for the interfacial tension:

$$\gamma = \sigma_1 + \sigma_2 - 2\phi_{GG} (\sigma_1 \sigma_2)^{0.5}$$
(4.11)

$$\phi_{GG} = \frac{W_a}{(W_{c1}W_{c2})^{0.5}}$$
(4.12)

where W_{c1} and W_{c2} represent the work of cohesion for the two components, and ϕ_{GG} is the Good and Girefalco interaction parameter which can be evaluated in terms of the molecular constants of the individual phases. Unfortunately, this approach to calculate interfacial tension is limited by the paucity of information on the values of ϕ_{GG} . Wu [1969,1970] tabulated some values of ϕ_{GG} for polymers pairs based on the measured values of interfacial and surface tension. They range between 0.5 and 1.2 [Wu 1974].

An alternative theory that has been proposed is the fractional polarity theory. [Wu 1971,1973], where the intermolecular energies are assumed to consist of additive non-polar (σ^{d}) and polar (σ^{p}) parts:

$$\sigma = \sigma^d + \sigma^p \tag{4.13}$$

The interfacial tension between two phases with similar polarities (which is the case for polymers) is then expressed as:

$$\gamma = \sigma_1 + \sigma_2 - \left(\frac{4\sigma_1^d \sigma_2^d}{\sigma_1^d + \sigma_2^d}\right) - \left(\frac{4\sigma_1^p \sigma_2^p}{\sigma_1^p + \sigma_2^p}\right)$$
(4.14)

or as:

$$\gamma = \sigma_1 + \sigma_2 - 2(\sigma_1^{d} \sigma_2^{d})^{0.5} - 2(\sigma_1^{p} \sigma_2^{p})^{0.5}$$
(4.15)

where 1 and 2 represent the two polymers.

Wu [1970] studied the influence of temperature on interfacial tension. He showed that the temperature coefficient of the interfacial tension can be obtained from the temperature coefficients of the surface tension (after differentiation of equation 4.11) and assuming that ϕ is independent of temperature. Good agreement was found with experiment [Wu 1969,1970,1971,1973]. However, there is no relationship between the perturbation occurring at the surface of the polymer and at the interface between two polymers. Thus, the above relationships relating interfacial tension to surface tension have to be considered as empirical.

3.2. Microscopic Theories of Polymeric Interfaces

3.2.1. Lattice Theory

Several lattice theories have been proposed for polymer-polymer interfaces [Roe 1975, Helfand 1975-1976, Weber 1976 and Helfand and Tagami 1971-1972]. Unfortunately, most of them cannot be verified experimentally, because of the lack of knowledge of the required lattice parameters.

Helfand and Tagami [1971, 1972] introduced a theory to analyze the interface between polymers A and B. This theory is indistinguishable from the other lattice theories, but it has the advantage of not requiring the knowledge of lattice parameters. It is based on a self-consistent mean field, which determines the configurational statistics of the polymer chains in the interfacial region. Also it assumes that, in the interfacial region, the interface energetic forces (determined essentially by polymer A / polymer B segmental interaction parameter χ [Flory (1941, 1953)]) tend to drive the A and B molecules apart. This separation, however, must be achieved in such a way as to prevent a gap from opening between the polymer phases. With these assumptions, they determine a self consistent mean field which determines the configuration of the polymer chain at the interface. The polymers satisfy a diffusion equation which is solved asymptotically for an infinite degree of polymerization and low degree of compressibility. This treatment yields the following expression for the density profiles of the polymers at the interface (the complete derivation of the theory can be found in Helfand and Tagami [1972]):

$$\frac{\rho_{\lambda}(\mathbf{x},\boldsymbol{\chi})}{\rho_{o}} = \frac{\varepsilon^{2}}{1+\varepsilon^{2}}$$
(4.16)

$$\frac{\rho_{\rm B}(\mathbf{x},\boldsymbol{\chi})}{\rho_{\rm o}} = \frac{1}{1+\varepsilon^2} \tag{4.17}$$

$$\varepsilon = \exp\left[\frac{(6\chi)^{\frac{1}{2}}x}{b}\right]$$
(4.18)

with.

where ρ_A and ρ_B are the densities at the interface of molecules A and B, ρ_A is the monomer density, x is the distance from the plane of the interface, χ is the Flory-Huggins interaction parameter, b is the effective length of the monomer units. From the above density profiles, Helfand and Tagami inferred a measure of the effective interfacial thickness:

$$a_{1} = \frac{\rho_{0}}{\left(\frac{d\rho_{a}}{dx}\right)_{x=0}} = \frac{2b}{\left(6\chi\right)^{\frac{1}{2}}}$$
(4.19)

Using for the interfacial tension the expression given in Equation 4.20 and the density profile they obtained:

$$\gamma = \frac{kT}{\rho_o} \int_0^1 d\lambda \int_{-\infty}^{\infty} \chi \rho_A(x, \lambda \chi) \rho_B(x, \lambda \chi) dx$$
(4.20)

which yields:

$$\gamma = \left(\frac{\chi}{6}\right)^{\frac{1}{2}} b\rho_{o} kT \tag{4.21}$$

where k is the Boltzman constant, T is the temperature, the values of b and ρ_{μ} at the interface are approximated by the geometrical means of the values of these constants for the two polymers.

To derive the above equations for the interfacial tension, Helfand and Tagami had to make the following restrictive assumptions:

(i) the polymer chains have an infinite molecular weight;

(ii) the systems are symmetric; they calculated the monomer density and the effective length as the geometric average of these constants for the two polymers; and

(iii) the systems were considered to be incompressible (κ the compressibility is considered to be about 5 x 10⁻¹¹ cm²/dyne).
The interfacial tension, as calculated by the theory, was originally compared [Helfand and Tagami (1971)] to the experimental values of interfacial tension of Roe [1969] and Wu [1970] for three polymer pairs at 150°C: polystyrene / poly(methyl metacrylate) (PS/PMMA), PMMA / poly-n-butyl methacrylate (PnBMA) and PnBMA / polyvinyl acetate (PVA). The experimental and theoretical results were in good agreement, but only three data points at only one temperature were compared.

Helfand and Sapse [1975] extended the theory for non-symmetric systems and obtained:

$$\gamma = kT\alpha^{\frac{1}{2}} \left[\frac{\beta_A + \beta_B}{2} + \frac{1}{6} \frac{(\beta_A - \beta_B)^2}{\beta_A + \beta_B} \right]$$
(4.22)

where α is the mixing parameter $(\alpha = \chi(\rho_{0A}\rho_{0B})^{\frac{1}{2}})$ and $\beta_i^2 = \frac{1}{6}\rho_{0i}b_i^2$.

The theory was compared to experimental data and the agreement was reasonable with a tendency of the theory to give lower values than the experiments.

Later, Tagami [1980] extended the theory to the case of compressible nonsymmetric systems, but the resulting equations are too complicated and the results do not differ significatively from those predicted by Helfand and Sapse.

Anastasiadis [1986] compared his experimental data for the influence of temperature on the interfacial tension between polystyrene and polybutadiene with the theoretical values predicted by Equations 4.21 and 4.22. He showed that the theory predicted the correct order of magnitude for the interfacial tension but not the correct trend. The theoretical value of the interfacial tension was shown to increase with an increase in temperature. The author suggested that the failure of the theory was due to the determination of the Flory-Huggins interaction parameter. For the determination of this parameter he used the relationship between the Flory-Huggins interaction parameter and the Hildebrand solubility parameter derived by Flory [1953]:

$$\chi = \frac{(\delta_{\rm A} - \delta_{\rm B})^2}{\rho_{\rm o} kT} \tag{4.23}$$

155

where χ is the Flory-Huggins interaction parameter, δ_{λ} and δ_{μ} are the Hildebrand solubility parameters for the two polymers, ρ_{α} is the monomer density defined as the geometric average of the monomer densities for the two polymers, k is the Boltzman constant, and T in the temperature. Anastasiadis claims that a proper temperature and molecular weight dependence of the Flory Huggins interaction parameter could lead to good agreement with the theory. However, he was not able to compare his experimental data in respect to molecular weight, since the theories assume infinite molecular weight.

Helfand and Bhattacharjee [1989] proposed a theory that calculates the correction to the concentration profile and the interfacial tension for a phase binary system when the molecular weight is large but finite. They modified the original model of Helfand and Tagami [1970-1971] by considering the behavior of chain ends at the interface and evaluating its influence on the entropy of mixing. The resulting equation for the interfacial tension is given below:

$$\gamma = \gamma_{o} \left(1 - \frac{2\log 2}{\chi N} \right) \tag{4.24}$$

where γ is the interfacial tension, γ_o is value of the interfacial tension as calculated in Equation 4.21, χ is the Flory-Huggins interaction parameter and N is the degree of polymerization. The predictions of this theory have not been compared to experimental data so far.

3.2.2. Theories based on Square Gradient

These theories were first introduced by Cahn and Hilliard (1958). In this approach, the interfacial tension is related to the Gibbs free energy of mixing. The local free energy, g, is a function of the local composition and the local composition derivatives, because g depends on the local composition as well as on the composition of the immediate environment. In the case of one-dimensional composition gradient and a flat interface, G, the total free energy, can be written as:

$$G = \int_{-\infty}^{\infty} (g_o(\phi) + \kappa (\frac{d\phi}{dx})^2) dx \qquad (4.25)$$

where the free energy is assumed to be a function of the local composition, ϕ , and its derivatives, g_{ρ} is the first term of a Taylor series expansion of g (the local free energy), i.e. the free energy density of a uniform system of composition ϕ , and x is the distance from the interface. κ is given by:

$$\kappa = -\left(\frac{\partial^2 g}{\partial \phi \partial \nabla^2 \phi}\right)_0 + \left(\frac{\partial^2 g}{\partial |\nabla \phi|^2}\right)_0 \tag{4.26}$$

The interfacial tension is the difference per interface unit area between the actual free energy of the system and that which it would have if the properties of the phases were homogeneous throughout. Thus, the interfacial tension, γ , is given by:

$$\gamma = \int_{-\infty}^{\infty} [\Delta g(\phi) + \kappa (\frac{d\phi}{dx})^2] dx \qquad (4.27)$$

where $\Delta g(\phi)$ is the free energy density of the uniform system of composition ϕ with respect to a standard state of an equilibrium mixture of the two phases, A and B, without the interface given by:

$$\Delta g(\phi) = \Delta g_{o}(\phi) - [n_{A} \Delta \mu_{A}(\phi_{e}) + n_{B} \Delta \mu_{B}(\phi_{e})]$$
(4.28)

where n_A and n_B are the number densities of molecules of type A and B, $\Delta \mu_A$ and $\Delta \mu_B$ are the changes in the chemical potentials of A and B, and ϕ_e is equal to the composition of either ϕ_A and ϕ_B of the two phases A and B at equilibrium.

At equilibrium, the composition variation will be such that the integral in Equation 4.27 is a minimum. Therefore, using Euler equation, we obtain:

$$\Delta g(\phi) = \kappa \left(\frac{d\phi}{dx}\right)^2 \tag{4.29}$$

The final expression for interfacial tension is:

$$\gamma = \int_{\phi_{A}}^{\phi_{B}} 2[\kappa \Delta g(\phi)]^{\frac{1}{2}} d\phi \qquad (4.30)$$

The theory has been widely used to predict the surface tension of liquids [Yang 1976, Carey 1978] and polymer melts [Poser 1979, Sanchez 1983, Dee 1991] and to predict the interfacial tension between polymers [Poser and Sanchez 1981].

The theory has been further developed during the last fifteen years in order to evaluate the interfacial tension between polymer melts. Poser and Sanchez [1981] reviewed the theory in conjunction with the compressible lattice fluid model of Sanchez and Lacombe [1976-1978]. Anastasiadis [1988] used the generalized square gradient approach in conjunction with the Flory-Huggins theory of the free energy of mixing. Anastasiadis evaluated numerically the theoretical expression of the interfacial tension and compared his experimental results to the theory and saw good agreement between his experimental data and the theory. However, the theory seemed to be only valid at high molecular weight. He suggested that the discrepancy was probably due to the difficulty in evaluating the Flory-Huggins free energy (evaluation of the Flory Huggins interaction parameter) or possibly to structural deficiency of the square gradient theory itself.

Broseta et al [1990] extended the work of Anastasiadis and co-workers [1988] and provided analytical expressions for the interfacial tension for finite molecular weight polymers. They also investigated the effects of polydispersity on interfacial tension between polymers. After integration, they obtained, for the interfacial tension between two polymers with finite molecular weight, the following expression:

$$\gamma = \gamma_{\infty} [1 - \frac{\pi^2}{12} (\frac{1}{\omega_{A}} + \frac{1}{\omega_{B}}) + \dots]$$
(4.31)

where γ is the interfacial tension, γ_{∞} is the interfacial tension as calculated in Equation 4.21, and ω_A and ω_B are the incompatibility degrees defined as $\omega_A = \chi_{AB} N_A$ and $\omega_B = \chi_{AB} N_B$, with χ_{AB} being the Flory-Huggins interaction parameter and N_A and N_B being the degrees of polymerization of polymers A and B, respectively.

In the same publication, Broseta et al [1990] derived an expression for the interfacial tension of bimodal blends. For large incompatibilities but for $\Delta\omega$ small ($\Delta\omega$ being the difference of compatibility between the two polymers), they obtained:

$$\gamma = \gamma_{\infty} \left[1 - \frac{\pi^2}{6\omega_n} + \dots \right]$$
(4.32)

where

$$\omega_{n} = \frac{1}{\left(\frac{\mathbf{x}_{o}}{\omega_{1}} + \frac{(1 - \mathbf{x}_{o})}{\omega_{2}}\right)}$$
(4.33)

where ω_1 , ω_2 are the incompatibility parameters of the two fractions of the polymer, and x_0 is the volume fraction of monomers belonging to small chains.

Equations 4.32 and 4.33 show that the interfacial tension is lowered by the presence of small chains, which play the role of a surfactant. The interfacial tension is therefore predicted to be lower for polydisperse systems than for monodisperse systems. Similar behavior has been predicted when considering the surface tension [Hariharan 1990].

The results of the above theoretical approaches have not been compared to experiments so far. The experimental results of this work are compared to the predictions of the square gradient theories in Section 4.

3.3. Effects of Compatibilizers

Very little work has been done to evaluate the effect of compatibilizers on interfacial tension. The only attempt that has been reported so far is the work of Noolandi and coworkers [1980-1982,1984] and Leibler [1982] who studied the effect of the addition of copolymers on interfacial tension. Hong and Noolandi [1980-1981] first derived a theory to evaluate the polymer density profile at the interface for a system of two polymers and a solvent. Then, they used statistical thermodynamics to describe the mixture polymer A / polymer B / copolymer AB [Hong and Noolandi 1982]. They evaluated the relative importance of the contributions of thermodynamic quantities, such as entropy and enthalpy of mixing of the two immiscible polymers and copolymer, that influence the interfacial tension reduction [1984]. An extension of this theory was proposed by Vilgis and Noolandi [1990]. They evaluated the interfacial tension reduction and characterized the interfacial tension profile for a blend containing homopolymer A, homopolymer B and an arbitrary copolymer CXY.

Up to now no theoretical work on the influence of functionalization on interfacial tension has been reported.

4. COMPARISON WITH THEORY

The experimental results were compared with the new development of the square gradient theories (Equations 4.31 and 4.32). It will be shown later in this chapter that both theories (Helfand's lattice theory and the new development of the square gradient theory) although conceptually different, provide essentially the same results. In both theories, there is a correction factor $\frac{1}{\omega}$ to account for the finite molecular weight, where ω is the incompatibility degree as defined in Equation 4.32. The only difference between the results given by the two theories is the replacement of the factor $\frac{\pi^2}{12}$ of Equation 4.31 by log 2 (Equation 4.26), because of a slightly different treatment of the entropy of mixing.

4.1. Evaluation of the Parameters

The different expressions used for the determination of the interfacial tension (Equations 4.21, 4.31 and 4.32) require the knowledge of the number monomer density,

 ρ_{α} , the Kuhn statistical segment length, b, the degree of polymerization, N, and the Flory-Huggins interaction parameter χ (or α).

4.1.1. Evaluation of ρ_o , b and N

The number monomer density, ρ_o , is calculated using the density data presented in Chapter 3 and the geometric average as suggested by Helfand and Tagami [1970-1971]:

$$\rho_{o} = \left(\frac{\rho_{A}}{m_{A}}\frac{\rho_{B}}{m_{B}}\right)^{\frac{1}{2}}$$
(4.34)

where ρ_A and ρ_B are the densities of the polymers A and B respectively, and m_A and m_B are the molecular masses of the repeating units of A and B, respectively.

The Kuhn statistical segment lengths are estimated using the characteristic ratios $([\langle r_o^2 \rangle / MW]^{\frac{1}{2}})$, which can be found in Brandrupt [1990], and the relation:

$$b_i = m_i [\langle r_o^2 \rangle / MW]^{\frac{1}{2}}$$
 (4.35)

The values of m_i , $([< r_o^2 > /MW]^{\frac{1}{2}})$, b_i used in this study are reported in Table 4.1.

The values of N_i are the degrees of polymerization of the polymers. They are given by:

$$N_i = \frac{M_n}{m_i}$$
(4.36)

where M_n is the number average molecular weight and m_i is the molecular mass of the repeating unit.

4.1.2. Evaluation of the Flory-Huggins Interaction Parameter

The determination of the Flory-Huggins interaction parameter is essential for the theoretical evaluation of the interfacial tension using the new version of the square gradient theory. Three aspects of the interaction parameter are reviewed here:

i) the definition of the parameter

ii) the determination of the parameter

iii) the values taken for this research

Polymer	m, (g/mol)	$[\langle r_{\bullet}^{2} \rangle / MW]^{\frac{1}{2}}$ (nm) \blacklozenge	b _i (nm)	δ (J.cm ⁻³) ^{1/2} Solubility Parameter
Polystyrene				from 17.4
PS	104	0.070	0.71	to 19.0 #
Polyethylene				from 15.8
PE	28	0.107	0.57	to 17.1 #
Polypropylene				from 16.8
PP	42	0.075	0.49	to 18.8 🗕

Table 4.1. : Values of the Number Monomer Density andKuhn Statistical Segment for the Polymer used in this Study

♠: Values obtained from Brandrupt [1990]

+: Values obtained from Van Krevelen [1976]

4.1.2.1. Definition

The Flory-Huggins interaction parameter, χ , was introduced in the lattice model of Flory-Huggins. The Flory-Huggins model was originally developed to evaluate the Gibbs free energy of mixing for polymer/solvent systems [Flory 1941] and later generalized to polymer systems [Flory 1953].

According to the Flory-Huggins theory, the expression for the free energy of mixing of homopolymer A and homopolymer B is:

$$\Delta G_m = \Delta H_m - T \Delta S_m \qquad (4.37)$$

$$\Delta H_{m} = \frac{RTV}{\upsilon_{A}} \chi_{AB} \phi_{A} \phi_{B} \qquad (4.38)$$

$$\Delta S_{m} = \frac{-RV}{\upsilon_{A}} \left[\frac{(\phi_{A} - \ln \phi_{A})}{N_{A}} + \frac{(\phi_{B} - \ln \phi_{B})}{N_{B}} \right]$$
(4.39)

where ΔG_m , ΔH_m and ΔS_m are, respectively, the Gibbs free energy, the enthalpy and the entropy of mixing, N_A and N_B are the degrees of polymerization for the A and B polymers, ϕ_A and ϕ_B are the volume fractions of polymers A and B, χ_{AB} is the Flory-Huggins interaction parameter, V is the total volume of the system and υ_A is the segmental volume.

It was suggested that the Flory-Huggins interaction parameter may be related to the solubility parameter as follows:

$$\chi_{AB} = \frac{(\delta_A - \delta_B)^2}{\rho_o kT}$$
(4.40)

where δ_1 and δ_2 are the Hildebrand solubility parameters of the two polymers, ρ_0 is the monomer density as calculated from Equation 4.34, k is the Boltzman constant, and T is absolute temperature.

4.1.2.2. Validity of the Expression for χ

Few experimental determinations of the Flory-Huggins interaction parameter have been reported to date, and, in particular, no data have been published for the polymer pairs studied in the present work. The scarcity of data forces the use of Equation 4.40 to determine the χ parameter necessary for the evaluation of the interfacial tension with the theories presented previously (Helfand and Bhattacharjee's lattice theory and the new version of the Square gradient). In Equation 4.40, the Hildebrand solubility parameter, δ , is needed. Although the Hildebrand solubility parameter should vary with temperature, there is no relationship reported yet between δ and temperature. Therefore, the parameter has to be assumed temperature independent. This could result in an error in the determination of the Flory-Huggins interaction parameter. Also, the values of the Hildebrand solubility parameter are reported with relatively wide range of variability (see Table 4.1: as an example, values of the Hildebrand solubility parameter for polypropylene vary from 16.8 to 18.8 (J.cm⁻³)^{1/2} [Van Krevelen 1976]) causing uncertainty in the value of the χ parameter.

The validity of Equation 4.40 was examined for the few published data of Flory-Huggins interaction parameter of polymer pairs. Ronca et al [1985] measured the interaction parameter for polystyrene (PS) / polybutadiene (PBD) polymer pairs. They found that the interaction parameter could be expressed as:

$$\chi = -0.2024 + \frac{116}{T} \tag{4.41}$$

where T is the temperature in Celsius.

The Flory-Huggins interaction parameter was calculated using Equation 4.40. ρ_{a} was calculated using Equation 4.34 and the density as determined in Chapter 3 for PS, and as determined by Anastasiadis [1988] for PBD. The Hildebrand solubility parameters, δ , used were respectively 16.6 (J.cm⁻³)^{1/2} for PBD and 17.9 (J.cm⁻³)^{1/2} for PS. With the above data, the calculated value of the Flory-Huggins interaction parameter is 0.044 (Equation 4.40) whereas the measured experimental value is 0.95 [Ronca 1985]. It can be seen that there is more than one order of magnitude difference between the two values.

Somani [1983] measured the quantity $\chi \rho_o kT$ for LDPE / PS polymer pairs. He reported a molecular weight dependence for this quantity, which, according to Equation 4.40 is equal $(\delta_{LDPE} - \delta_{PS})^2$ with δ_{LDPE} and δ_{PS} being the solubility parameters of LDPE and PS, respectively. This is contrary to what is expected from Equation 4.40, since no influence of molecular weight on the Hildebrand parameter has been reported. The difference between the solubility parameters is equal to 1.4 (J.cm⁻³)^{1/2}, taking the values of Table 4.1, whereas Somani reported values of $\chi \rho_o kT$ ranging from 7 to 11(J.cm⁻³)^{1/2}, depending on the polystyrene molecular weight.

Dependence of the Flory-Huggins interaction parameter on molecular weight has also been reported by other researchers [Kaddour 1987, Narasimhan 1989]. This is not taken into account when evaluating χ with Equation 4.40. The only molecular weight dependence of χ , as calculated by Equation 4.40, is the influence on the density.

From the few examples mentioned above, it can be seen that the Flory-Huggins interaction parameter cannot be estimated accurately using equation 4.40 because the Hildebrand solubility parameter is known only at one temperature and for a specific molecular weight.

It has been shown that χ should, in fact, be expressed as the sum of an enthalpic and an entropic contribution [Flory 1970, Huggins 1971, Jelenic 1984] in order to take into account the non-figurational entropy of mixing. An expression of χ can then be calculated as:

$$\chi(\mathbf{T},\mathbf{M}_{\mathrm{p}}) = \frac{\chi_{\mathrm{H}}}{\mathrm{T}} + \chi_{\mathrm{S}}$$
(4.42)

where $\chi_{\rm H}$ is the enthalpic contribution to the interaction parameter, $\chi_{\rm s}$ is the entropic contribution, and T is the temperature. This functional form of the χ parameter has been successfully used to describe phase diagrams of polymer/polymer systems [Rounds 1970], and interfacial tension between polymers [Anastasiadis 1988].

It has also been shown that the Flory-Huggins interaction parameter is probably a function of the molecular weight of the polymers [Burns and Kim 1988, Kaddour 1987]. This dependence should be taken into account in the interaction parameter values used for the calculation of the interfacial tension.

The Flory-Huggins interaction parameter for polydisperse systems has been shown to be the same as the one for monodisperse polymers [Narasimhan et al 1986] at the same value of M_n .

4.1.2.3. Values of χ used for this Research

Regardless of the theory used for the evaluation of the interfacial tension (Helfand's lattice theories or the square gradient theories), it is necessary to estimate the Flory-Huggins interaction parameter, χ . It has been shown that χ cannot be calculated from Equation 4.40. The most likely expression for χ should be Equation 4.40. Due to the lack of experimental values for $\chi_{\rm H}$ and $\chi_{\rm S}$, a strategy had to be developed for the evaluation of these parameters. It was assumed that the modified square gradient theory is correct. The interfacial tension was experimentally measured at two different temperatures. With these two interfacial tension values, the values of $\chi_{\rm H}$ and $\chi_{\rm S}$ were estimated using Equations 4.31 and 4.42. The values of $\chi_{\rm H}$ and $\chi_{\rm S}$ were then used to determine the dependency of χ on temperature, i.e. the theory was checked for other temperatures.

The temperatures used for the PP/PS polymer pairs were 208°C and 213°C except for PP/PS₆ for which the temperatures taken were 226°C and 228°C. The temperatures used for the PE/PS polymer pairs were 130°C and 150°C.

Table 4.2 shows the values of χ_{H} and χ_{s} determined using the strategy described above for the different polymer pairs studied in this work.

Polymer pair	χ _н	χ _s
PP/PS3	563	-1.03
PP/PS₄	804	-1.52
PP/PS ₃	949	-1.79
PP/PS ₆	1,007	-1.82
PE ₁ /PS ₃	538	-1.04
PE ₂ /PS ₃	452	-0.829
PE ₃ /PS ₃	452	-0.828

Table 4.2. : Interaction Parameter used in this Study

4.2. Comparisons for the PP/PS Polymer Pair

4.2.1. Temperature Influence

The influence of temperature on interfacial tension between PP and PS was studied theoretically using the updated version of the square gradient theory in conjunction with the Flory-Huggins parameter defined by Equation 4.40. The molecular weight of the polystyrene was 86,436. The values of the Hildebrand solubility parameter for both polymers were either $\delta_{pp} = 17.8 \ (J.cm^{-3})^{1/2}$ and $\delta_{ps} = 18.7 \ (J.cm^{-3})^{1/2}$ or $\delta_{pp} = 17.3 \ (J.cm^{-3})^{1/2}$ and $\delta_{ps} = 22.5 \ (J.cm^{-3})^{1/2}$ [Polymer Handbook 1989]. The theoretical predictions were compared with the experimental results. This is shown in Figure 4.1.



Figure 4.1: Interfacial Tension between PP and PS_5 as a Function of Temperature: Comparison between the Predictions of the New Development of the Square Gradient Theory and the Experimental data - χ from Equation 4.40

It can be seen that, depending on the values of the Hildebrand solubility parameter chosen, the interfacial tension values predicted using the theory vary by almost an order of magnitude. Also, using the new version of the square gradient theory in conjunction with the Flory-Huggins parameter defined by Equation 4.40, the correct trend is not observed. The interfacial tension increases with increasing temperature. This is opposite to what has been observed experimentally. The same observations could be made for all PP/PS polymer pairs with different molecular weight of PS. Anastasiadis [1987] showed the same trend comparing his experimental values with theoretical predictions for hydrogenated polybutadiene / polystyrene polymer pairs.

In a second approach, the influence of temperature on interfacial tension between PP and PS was studied theoretically, using the updated version of the new square gradient theory in conjunction with the Flory-Huggins parameter obtained using the strategy mentioned in section 4.1.2.3. Figure 4.2 shows the comparison between the predictions of the square gradient theory (Equation 4.32) and the experimental data of interfacial tension for polypropylene (PP) / polystyrene (PS) polymer pair.

Figure 4.2 shows very good agreement between the prediction of the theory and the experimental data, suggesting that the new version of the square gradient theory is valid for predicting the interfacial tension between PP and PS, as long as a correct expression for the Flory-Huggins interaction parameter is used.

In order to compare the new version of the square gradient theory with the other theories mentioned previously, the interfacial tension between PP and PS for PS_5 (M_w =380,000) and PS_3 (M_w =4,755) was calculated as a function of temperature using the following theories:

i) Helfand and Tagami: this theory assumes symmetry of both polymers and infinite molecular weight.

ii) Helfand and Sapse: this theory is a correction of the theory of Helfand and Tagami for non-symmetric systems.

iii) Helfand and Bhattacharjee: this theory is a correction of the theory of Helfand and Tagami for systems with finite molecular weight.

iv) The new version of the square gradient theory: this theory introduces a correction in the classical square gradient theory in order to consider polymer with finite molecular weight.



Figure 4.2: Interfacial Tension between PP and PS as a Function of Temperature: Comparison between the Predictions of the New Development of the Square Gradient Theory and the Experimental data - χ from Table 4.2

The results of interfacial tension between PP and PS₅ (M_w =380,000) and PP and PS₃ (M_w =4,755) are presented in Figures 4.3 and 4.4, respectively:

All the above theories require the use of the Flory-Huggins interaction parameter, χ . The values of χ were determined using the strategy described Section 4.1.2.3.

The following conclusions can be made on the basis of the results shown in Figures 4.3 and 4.4.

i) With the Flory-Huggins interaction parameter calculated as the sum of an enthalpic and an entropic part, the four theories predict the correct trend of the influence of temperature on the interfacial tension.

ii) For both polymer pairs presented here, the differences between the predictions of Helfand and Tagami (Equation 4.21) and Helfand and Sapse (Equation 4.22) are very small, suggesting that the system PP/PS is relatively symmetric.

iii) The correction in the theory for use with polymers having finite molecular weight is more important in the case of the lower molecular weight as expected.



Figure 4.3: Interfacial Tension between PP and PS₅ as a Function of Temperature: Comparison between the Predictions of the New Development of the Square Gradient Theory, Helfand and Tagami, Helfand and Sapse, Helfand and Bhatthacharjee and the Experimental data - χ from Table 4.2



Figure 4.4: Interfacial Tension between PP and PS₃ as a Function of Temperature: Comparison between the Predictions of the New Development of the Square Gradient Theory, Helfand and Tagami, Helfand and Sapse, Helfand and Bhatthacharjee and the Experimental data - χ from Table 4.2

4.2.2. Molecular Weight Influence

The influence of molecular weight on interfacial tension between PP and PS was theoretically studied with the new version of the square gradient theory. The only other theory capable of predicting the influence of molecular weight is the theory developed by Helfand and Bhattacharjee [1990] which results in a similar expression for the interfacial tension as the new development of the square gradient theory: Since the new development of the square gradient theory can also predict the effect of polydispersity, it was decided to use this theory to study the influence of both molecular weight and polydispersity. The effect of molecular weight on the interfacial tension between PP and PS, as a function of the molecular weight of PS, was evaluated at a temperature of 186°C.

In a first approach, the influence of molecular weight on interfacial tension between PP and PS at 186°C was theoretically studied with the new version of the square gradient theory in conjunction with the Flory Huggins parameter defined by equation 4.38 $(\chi = f(\Delta \delta))$. The values of the Hildebrand solubility parameter for both polymers were either $\delta_{pp} = 17.8$ (J.cm⁻³)^{1/2} and $\delta_{ps} = 18.7$ (J.cm⁻³)^{1/2} or $\delta_{pp} = 17.3$ (J.cm⁻³)^{1/2} and $\delta_{ps} =$ 22.5 (J.cm⁻³)^{1/2} [Polymer Handbook (1989)]. The theoretical predictions were compared with the experimental results. This is shown in Figure 4.5.



Figure 4.5: Interfacial Tension between PP and PS as a Function of Molecular Weight of PS: Comparison between the Predictions of the New Development of the Square Gradient Theory - χ from Equation 4.40

It can be seen that the theory predicts qualitatively reasonably well the effect of molecular weight on interfacial tension between PP and PS. However, for both sets of the Hildebrand solubility parameters, the theory underestimates the interfacial tension. The theory predicts that the interfacial tension between PP and PS_1 is negative, implying miscibility of the two polymers, which was not supported experimentally, since two phases were always present during the experiments.

In a second approach, the influence of molecular weight on interfacial tension (between PP and PS at 186°C) was theoretically studied with the new version of the square gradient theory in conjunction with the Flory Huggins parameter calculated with the strategy described in Section 4.1.2.3. In this comparison, six different polystyrene (PS₁ to PS₆) polymers were used with molecular weights ranging from 938 to 380,000. The Flory-Huggins interaction parameters for PP and PS₃, PS₄, PS₅, PS₆ (the four polystyrenes with the highest molecular weight) were calculated using the results of the strategy mentioned above. The same strategy could not be used to determine the Flory-Huggins interaction parameter between PS₁ (M_n = 938) and PP and PS₂ (M_n = 1,589) and PP, because only one interfacial tension value was measured for PS₁/PP and PS₂/PP polymer pairs and two values are needed for this determination. Another method was therefore developed to determine the Flory-Huggins interaction parameter between PS₂ and PP at 186°C.

It has been suggested that the Flory-Huggins interaction parameter depends on the molecular weight of polymers [Kaddour 1987, Kim and Burns 1989] according to the following relation:

$$\chi(T,M_{n}) = aM_{n}^{b} \qquad (4.43)$$

where χ is the Flory-Huggins interaction parameter, M_n is the number average molecular weight and a and b are constants.

The Flory-Huggins interaction parameter between PP and PS was plotted as a function of the molecular weight of polystyrene for the four PS polymers with the highest molecular weight and fitted to Equation 4.43. This is shown in Figure 4.6. The values of a and b were respectively 0.05684 and 0.1443. Using this equation, it was possible to evaluate the Flory-Huggins interaction parameter for PS₁ and for PS₂ at 186°C. With

these values of χ , the interfacial tension values between PP and PS₁ and PP and PS₂ at 186°C were calculated using the new version of the square gradient theory (Equation 4.31).



Figure 4.6: Flory-Huggins Interaction Parameter between PP and PS as a function of Molecular Weight of PS

Figure 4.7. shows the interfacial tension between PP and PS as a function of molecular weight of PS. It can be seen that there is good agreement between the theory and the experimental data. The experimental values of interfacial tension between PP and PS_1 and PP and PS_2 are in good agreement with the theoretical predictions of the new square gradient theory (Equation 4.31), as can be seen in Table 4.3, suggesting that the new square gradient theory is valid, as long as the correct value of the Flory-Huggins interaction parameter is used.

	Experimental Results	Theoretical Predictions	
Polymer Pair			
PP/PS ₁	4.26	4.29	
PP/PS,	4.64	4.78	

Table 4.3. : Interfacial	tension values	between PP a	nd PS,	or PS, at	186°C



Figure 4.7: Interfacial Tension between PP and PS as a Function of Molecular Weight of PS: Comparison between the Predictions of the New Development of the Square Gradient Theory - χ from Table 4.2

4.2.3. Polydispersity Influence

The only available theory to predict the effect of polydispersity on interfacial tension is the new development of the square gradient theory. To evaluate the effect of polydispersity on interfacial tension, Broseta et al [1990] considered a bimodal blend and derived an expression of the interfacial tension as a function of the incompatibility parameter and volume fraction of the two fractions of the polymer. The interfacial tension between a bimodal blend and another polymer was then calculated to be:

$$\gamma = \gamma_{\infty} [1 - \frac{\pi^2}{6\omega_n} + \dots]$$
(4.32)

where

$$\omega_{n} = \frac{1}{\left(\frac{x_{o}}{\omega_{1}} + \frac{(1 - x_{o})}{\omega_{2}}\right)}$$
(4.33)

where ω_1 and ω_2 are the incompatibility parameters of the two fractions of the polymer and x_0 is the volume fraction of small chain monomers.

Using Equations 4.30 and 4.31, Broseta et al [1990] concluded that the interfacial tension of polydisperse systems is lowered by the presence of small chains migrating to the interface. This was not observed by Goldblatt [1988] who studied the surface composition of polystyrene compared to the bulk by S.I.M.S (Second Ion Mass Spectroscopy).

In Chapter 3, experimental data regarding the interfacial tension between PP and bimodal blends of PS were reported. These results indicated that the presence of small chains increases the interfacial tension, in contrast to the theoretical predictions just mentioned above.

The discrepancies between the experimental and theoretical results could be explained as follows. In the new version of the square gradient theory, it is assumed that the incompatibility parameters of the two polymer fractions should be similar in value (5, 10 would be considered close, although no specific range has been defined by Broseta et al [1990]). This is not the case for the bimodal blends used in the present work as can be seen in Table 4.4.

Table 4.4 : Incompatibility Parameter of the Polymer Fractionsof the Bimodal Blends at 186°C

Polymer Blends	Fraction with lower M _n	Fraction with higher Mn		
PS ₂ 1,589 / PS ₄ 19,417	2.52	43.83		
PS3 4,755 / PS5 86,438	9.01	231.28		

The assumption of incompressibility used by Broseta et al for the development of the theory is probably not observed for the polymer used. Also, it has been observed experimentally that the temperature has a stronger influence on the interfacial tension for polydisperse systems than for monodisperse systems. The theory does not take this factor into consideration.

4.3. Comparison for the PE/PS Polymer Pair

4.3.1. Temperature Influence

The influence of temperature on interfacial tension between PE and PS was first studied theoretically, using the updated version of the square gradient theory. The comparison of the theoretical predictions of the new version of the square gradient theory with experiments for the PE/PS polymer pair supported the observations made for the PP/PS polymer pair.

When the new version of the square gradient theory was used in conjunction with the Flory-Huggins interaction parameter defined by Equation 4.40 ($\chi = f(\Delta \delta)$), the theoretical results were in relatively good quantitative agreement with the experimental results. However, the theory predicted that the interfacial tension should increase with increasing temperature, which is the opposite of what has been observed experimentally. When the new version of the square gradient theory was used in conjunction with the Flory-Huggins interaction parameter calculated using the strategy presented in 4.1.2.3, the theoretical predictions were in very good agreement with experimental observations. This is shown in Figure 4.8.



Figure 4.8: Interfacial Tension between PE and PP as a Function of Temperature Comparison between the Predictions of the New Version of the Square Gradient Theory and Experimental Data - χ from Table 4.2

The new version of the square gradient theory was also compared with the other theories [Helfand and Tagami 1970, Helfand and Sapse 1975, Helfand and Bhattacharjee 1990] for the PE/PS system. The Flory-Huggins interaction parameter, χ , used in conjunction with these theories was determined using the strategy described in section

178

4.1.2.3. As in the case of PP/PS polymer pairs, the four theories predict the correct influence of the temperature, when the Flory-Huggins interaction parameter is calculated as the sum of an enthalpic and an entropic part. For both PE/PS polymer pairs presented here, the differences between the predictions of Helfand and Tagami (Equation 4.21) and Helfand and Sapse (Equation 4.22) are larger than in the case of the PP/PS polymer pair, indicating that the system is less symmetric than the PE/PS polymer pair. The results of the comparison for PE₃/PP and PE₁/PP polymer pairs are shown in Figure 4.9 and 4.10, respectively.



Figure 4.9: Interfacial Tension between PE_3 and PP as a Function of Temperature Comparison between the Predictions of the New Version of the Square Gradient Theory, Helfand and Tagami, Helfand and Sapse, Helfand and Bhattacharjee and Experimental Data - χ from Table 4.2



Figure 4.10: Interfacial Tension between PE₁ and PP as a Function of Temperature Comparison between the Predictions of the New Version of the Square Gradient Theory, Helfand and Tagami, Helfand and Sapse, Helfand and Bhattacharjee and Experimental Data - x from Table 4.2

4.3.2. Molecular Weight Influence

The effect of molecular weight of PE on the interfacial tension between PE and PS was evaluated at a temperature of 160°C, using the new version of the square gradient. Three different molecular weights of PE were used ranging from 680 to 1,870. The theoretical predictions were compared to the experimental results.

The results of this comparison supported the conclusions obtained for the PP/PS polymer pair.

When the theory was used in conjunction with the Flory-Huggins interaction parameter as calculated by Equation 4.40 ($\chi = f(\Delta \delta)$), qualitative agreement between the theory and experiment was obtained. However, with the values of the Hildebrand solubility parameter chosen, the interfacial tension between PE and PS was calculated to be negative. On the other hand, when the theory was used in conjunction with the Flory-Huggins interaction parameter as evaluated by the strategy presented in section 4.1.2.3, good agreement between the theoretical predictions and the experiments was observed. The results of the comparison are shown in Figure 4.11.



Figure 4.11: Interfacial Tension between PE and PS as a function of the Molecular Weight of PE - Comparison between the Predictions of the New version of the Square Gradient Theory and Experimental Data - χ from Table 4.2

The Flory-Huggins interaction parameter was plotted as a function of the molecular weight of polyethylene. The data could still be fitted by Equation 4.43. The values of a and b were, respectively, 0.1282 an 0.0700. Figure 4.12 shows the Flory-Huggins interaction parameter as a function of molecular weight.



Figure 4.12: Flory-Huggins Interaction Parameter between PP and PS as a Function of Molecular Weight of PE

5. CONCLUSIONS

The main conclusions of this Chapter are outlined below.

a) The comparisons of the predictions of the lattice theories and square gradient theories for interfacial tension with the experimental values of the interfacial tension for PP/PS and PE/PS polymer pairs indicated that the expression usually employed to correlate the Flory-Huggins interaction parameter to the Hildebrand solubility parameter is inadequate (Equation 4.38). If this expression for the Flory-Huggins interaction parameter is used in the new version of the square gradient theory, the calculated interfacial tension value increases with increasing temperature, contrary to experimental observations.

b) In order to compare the theoretical predictions of the Helfand and Bhattacharjee and Square gradient models to the experimental data of interfacial tension, it was necessary to evaluate the Flory-Huggins interaction parameter for the polymer pairs studied. A strategy was developed. The Flory-Huggins interaction parameter was assumed to be of the form:

$$\chi(T, M_n) = \frac{\chi_H}{T} + \chi_s \qquad (4.42)$$

where $\chi_{\rm H}$ is the enthalpic contribution to the interaction parameter, $\chi_{\rm s}$ is the entropic contribution, and T is the temperature. The values of $\chi_{\rm H}$ and $\chi_{\rm s}$ for the PP/PS and PE/PS polymer pair studied here were calculated using two interfacial tension values determined experimentally at two different temperatures and assuming the theory of the square gradient to be correct.

c) The predictions of the new version of the square gradient theory for the influence of temperature on interfacial tension were shown to be in good agreement with experimental data for PP/PS and PE/PS polymer pairs, as long as the correct expression for the Flory Huggins interaction parameter was used. The Flory-Huggins interaction parameter was expressed as the sum of an enthalpic and an entropic part.

d) It was shown that the system PP/PS was relatively symmetric since the interfacial tension evaluated theoretically for symmetric system (Equation 4.21) was the same as the

one evaluated for non symmetric systems (Equation 4.26). On the other hand, the system PE/PS was shown to be relatively non symmetric, since there were larger differences between the predictions of the two equations.

e) The Flory-Huggins interaction parameter was shown to be a function of the molecular weight for the two polymer pairs studied here (PP/PS and PE/PS). The dependence could be expressed as:

$$\chi(T, M_n) = a M_n^{b} \tag{4.43}$$

where χ is the Flory-Huggins interaction parameter, M_n is the number average molecular weight, and a and b are constants.

f) The predictions of the new version of the square gradient theory for molecular weight influence on interfacial tension were shown to be in good agreement with experimental data for PP/PS and PE/PS polymer pairs, as long as the correct expression for the Flory Huggins interaction parameter was used.

g) The experimental results for the interfacial tension between bimodal blends of PS and PP were compared with the new version of the square gradient theory. The theory predicts that the interfacial tension between a bimodal blend and another polymer is lowered by the presence of the small chains. The experimental results presented in Chapter 3 showed that the interfacial tension was increased by the presence of smaller chains. Possible reasons for the discrepancy between the experiments and theory were presented.

REFERENCES

Adamson A.W. [1967]: "Physical Chemistry of Surfaces", 2nd Ed., Wiley (interscience), New York, (1967).

Anastasiadis S.H., Chen J.K., Koberstein J.T., Sohn J.E. and Emerson J.A. [1986]: Polym. Eng. Sci., Vol 26, p 1410 (1986).

Anastasiadis S.H., Chen J.K., Koberstein J.T., Siegel A.F., Sohn J.E. and Emerson J.A. [1987]: J. Colloid Interface Sci., Vol 93, p169, (1987).

Anastasiadis S.H. [1988]: "Interfacial Tension of Immiscible Polymer Blends", Ph.D. Thesis, Princeton University, (1988).

Anastasiadis S.H., Gancarz I. and Koberstein J.T. [1988]: Macromolecules, Vol 21, p 2980, (1988).

Antonoff G., Chanin M., and Hecht M. [1942]: J. Phys. Chem., Vol 46, p 492 (1942).

Antonoff G. [1942]: J. Phys. Chem., Vol 46, p 497 (1942)

Antonow G. [1907]: J. Chem. Phys., Vol 5, p 372, (1907)

Brangdrup J., Immergut E.H., [1990]: "Polymer Handbook", John Wiley and Sons, New York, (1990).

Broseta D., Fredrickson G.H., Helfand E. and Leibler L. [1990]: Macromolecules, Vol 23, No 1, p , (1990).

Burns C.M., Kim W.N. [1988]: Polymer Engineering and Science, Vol 28, No 21, p 1362, (1988).

Cahn J.W., Hilliard J.E., [1958] : J. Chem. Phys., Vol28, p258, (1958).

Carey B.S., Scriven L.E. and Davis H.T. [1978]: J. Chem. Phys., Vol 69, p 5040, (1978).

Dee G.T. and Sauer B.B. [1991]: Journal of Colloid Science, Vol 152, No 1, p 85, (1991).

Flory P.J. [1941]: J. Chem. Phys., Vol 9, p 660, (1941).

Flory P.J. [1953]: " Principles of Polymer Chemistry"Cornell University Press, Ithaca, N.Y., (1953).

Flory P.J., Orwoll R.A. and Vrij A. [1964]: J. Amer. Chem. Soc., Vol 86, p 3515, (1964).

Girefalco L.A. and Good R.J., [1957]: J. Phys. Chem., Vol 61, p 904, (1957).

Goldblatt D.D., Scilla G.J. and Park J.M. [1988]: Journal of Applied Polymer Science, Vol 35, p2075, (1988).

Good R.J., Girefalco L.A. and Krauss G. [1958]: J. Phys. Chem., Vol 62, p 1418, (1958).

Good R.J. and Girefalco L.A. [1960]: J. Phys. Chem., Vol 64, p561, (1960).

Good R.J. [1964]: Adv. Chem. Ser., Vol 43, p 74, (1964).

Good R.J. and Elbing E. [1970]: Ind. Eng. Chem., Vol 62, p 54 (1970).

Hariharan A., Kumar S.K. and Russell T. P. [1990]: Macromolecules, Vol 23, p 3584, (1990).

Helfand E. and Tagami Y. [1971]: Polymer letters, Vol 9, p741, (1971).

Helfand E. and Tagami Y. [1972]: J.Chem. Phys., Vol 56, p 3592, (1972).

Helfand E. and Tagami Y. [1972]: J. Chem. Phys., Vol 57, p 1812, (1972).

Helfand E. and Sapse A.M. [1975]: J. Chem. Phys, Vol 62, p1327, (1975).

Helfand E. [1975]: J. Chem. Phys., Vol 63, p 2192 (1975).

Helfand E. [1976]: Macromolecules, Vol 9, p 311, (1976).

Helfand E. and Bhattacharjee S.M. [1990]: J. Chem. Phys., Vol 91 (11), p 7200, (1989).

Hong K.M. and Noolandi J. [1981]: Macromolecules, Vol 14, p 72 (1981).

Huggins M.L. [1941]: J. Chem. Phys., Vol 9, p 440, (1941).

Huggins M.L. [1976]: J.Phys. Chem., Vol 80, p 1317, (1976).

Jelenic J., Kirste R.G., Oberthur R.C, Schmitt-Strecker S., Schmitt B.J., [1984]: Makrom. Chem., Vol 185, p 129 (1984).

Kaddour L.O., Anasagasti M.S. and Strazielle C. [1987]: Makromol. Chem., Vol 188 (9), p 2223, (1987).

Lacombe R.H. and Sanchez I.C. [1976]: J. Phys. Chem., Vol 80, p 2568 (1976).

Leibler L. [1982]: Macromolecules, Vol 18, p657, (1985).

Narasimha V., Huang R.Y.M., Burns C.M. [1986]: J. Polym. Sci., Polym. Symp., 74 (in honor Charles G. Overberger His 65th Birthday), p265, (1986).

Narasimha V., Huang R.Y.M., Burns C.M. [1989]: Journal of Applied Polymer Science, Vol 37, p 1909, (1989).

Noolandi J. and Hong K.M. [1980]: Ferroelectrics, Vol 30, p 117, (1980).

Noolandi J. and Hong K.M. [1982]: Macromolecules, Vol 15, p482, (1982).

Noolandi J. [1984]: Polym. Eng. Sci., Vol 24, p70, (1984).

Noolandi J. and Hong K.M. [1984]: Macromolecules, Vol 17, p1531, (1984).

Poser C.I. and Sanchez I.C.[1979]: J. Colloid Interface Sci., Vol 69, p539 (1979).

Poser C.I. and Sanchez I.C. [1981]: Macromolecules, Vol 14, p 361, (1981).

Roe R.J. [1969]: J. Colloid Interface Sci., Vol 31, p228, (1969).

Roe R.J. [1975]: J. Chem Phys., Vol 62, p 490, (1975).

Ronca and Russell T.P. [1985]: Macromolecules, Vol 18, p 665 (1985).

Rounds N.A. [1970]: PhD. Dissertation, University of Akron, Akron, OH, (1970).

Sanchez [1983]: Ann. Rev. Mater. Sci., Vol 13, p387 (1983).

Sanchez I.C. and Lacombe R.H. [1976]: J. Phys. Chem., Vol 80, p2352, (1976).

Sanchez I.C. and Lacombe R.H. [1977]: J. Polym. Sci., Polym Lett., Vol 15, p71 (1977).

Sanchez I.C. and Lacombe R.H. [1978]: Macromolecules, Vol 11, p1145 (1978).

Somani R.H. [1983]: "Measurement of Miscibility and Interaction in Polymer Blends", PhD Thesis, University of Connecticut, (1983).

Tagami Y. [1980]: J. Chem. Phys. Vol73, p 5354, (1980).

Tagami Y. [1980]: Ferroelectrics, Vol 30, p115 (1980)

Van Krevelen D.W. and Hoftyzer P.J. [1976]: "Properties of Polymers, Their Estimation and Correlation with Chemical Structure", Elsevier Scientific Publishing Company, Amsterdamn Oxford, New York, (1976).

Vilgis T.A., Noolandi J. [1990]: Macromolecules, Vol 23, p 2941, (1990)

Weber T.A. and Helfand E. [1976]: Macromolecules, Vol 9, p311, (1976).

Wu S. [1969]: J. Colloid Interface Sci., Vol 31, p153, (1969).

Wu S. [1970]: J. of Phys. Chem, Vol74, p632, (1970).

Wu S. [1971]: J. Polym. Sc., Part C, Vol 34, p 19, (1971).

Wu S.[1973]: J. Adhes., Vol 5, p 59, (1973).

Wu [1974]: J. Macromol.Sci. - Revs Macromol. Chem., C10, p1 (1974).

Yang A.J.M., Fleming P.D., III, and Gibbs J.H. [1976]: J. Chem. Phys., Vol 64, p3732, (1976).



1. CONCLUSIONS

The conclusions have been presented at the end of each chapter of the thesis. A brief summary of these conclusions is presented below.

a) Two instruments to measure interfacial tension between polymers, one based on the pendant drop method and the other on the spinning drop method, were constructed and shown to be reliable. It was possible to view the spinning and pendant drops and to calculate the interfacial tension on-line. A specially designed syringe effectively eliminated problems associated with "capillary" and "necking" effects, usually encountered with the pendant drop method. A capacitance probe attached to the spinning drop apparatus, made it possible to determine the density of polymers as a function of temperature.

b) The interfacial tension between polymers may be inferred from transient measurements using an exponential fit of the interfacial tension (pendant drop) or radius of the drop (spinning drop).

c) The interfacial tension between polymers decreased linearly as a function of temperature for every polymer pair used in this work. The temperature coefficient, $(\frac{\partial \gamma}{\partial T})$, for the dependence of the interfacial tension, γ , on temperature, T, varied between 0.003

and 0.008 dyn.cm⁻¹.^OC⁻¹. The polydisperse systems and bimodal systems showed a higher dependence on temperature than monodisperse systems.

d) The interfacial tension followed a power law of the molecular weight for the polymer pairs used in this work. The interfacial tension did not always show a $M_n^{-\frac{2}{3}}$ dependence, as observed by other researchers. This was probably due to the wider range of molecular weights employed in this study. When a narrow range of molecular weights was studied, a $M_n^{-\frac{2}{3}}$ dependence was found.

e) The interfacial tension (at 459 K) between polypropylene and bimodal blends of polystyrene was higher than the weighted molar average of the interfacial tension between the unimodal polystyrene and polypropylene.

f) It was shown that the interfacial tension between polydisperse polystyrene and polypropylene was higher than the interfacial tension between monodisperse polystyrene and polypropylene for similar molecular weight. Similar results were obtained for the polyethylene/polystyrene polymer pair at temperatures lower than 436 K.

g) The interfacial tension between polypropylene and ethylene vinyl alcohol copolymer decreased with increasing concentration of maleic anhydride grafted on the backbone of polypropylene for concentrations up to 0.17% wt. If higher concentration of maleic anhydride was used, no further decrease of the interfacial tension was observed. Surface analysis using E.S.C.A. suggested the occurrence of a chemical reaction at the interface between the MAH groups of the maleated polypropylene and the OH groups of the ethylene vinyl alcohol copolymer. Better adhesion was observed with S.E.M. between maleated polypropylene and ethylene vinyl alcohol copolymer.

h) The interfacial tension between pure polypropylene and ethylene vinyl alcohol copolymer was shown to be lower by 30% than the interfacial tension between commercial polypropylene and ethylene vinyl alcohol. X-ray analysis and E.S.C.A. results have shown that the commercial polypropylene contains silicon in the form of SiOx.

i) The expression correlating the Flory-Huggins interaction parameter to the Hildebrand solubility parameter was inadequate for evaluating the Flory-Huggins parameter since the Hildebrand solubility parameter is known at only one temperature. Also, χ should be expressed as the sum of an enthalpic (χ_H) and an entropic (χ_S) part. A strategy to calculate the Flory-Huggins interaction parameter for the polymer pairs used in this work was developed. The values of χ_H and χ_S for the polypropylene / polystyrene and polyethylene / polystyrene polymer pairs used in this work were calculated. The Flory-Huggins interaction parameter was shown to follow a power law dependence on the molecular weight of the polymer.

j) The experimental results presented in this work were shown to be in good agreement with the predictions of the new version of the square gradient theory (developed by Broseta et al) for the influence of temperature and molecular weight on interfacial tension, as long as a suitable expression for the Flory-Huggins interaction parameter was used. However, the theory could not predict the effect of polydispersity.
2. RECOMMENDATIONS FOR FUTURE WORK

Some important areas for future research are listed below:

a) More fundamental studies of the transient behavior during the determination of interfacial tension with the pendant drop method: more fundamental understanding of the apparent exponential variation of the transient interfacial tension values and modeling of the evolution of the drop shape.

b) The study of interfacial tension for bimodal blends and polydisperse systems should be extended to various blends with different compositions at various temperatures.

c) The study of the influence of compatibilisation should be extended to different temperatures.

d) The surface analysis of the interface should be extended to different concentrations of maleic anhydride grafted on the backbone of polypropylene for eventual quantitative correlation with interfacial tension. Also, analysis combining Second Ion Mass Spectroscopy (S.I.M.S.) and Electron Spectroscopy for Chemical Analysis (E.S.C.A.) could help to determine the composition of the sample, as a function of distance from the interface.

e) The Flory-Huggins interaction parameter for the polymer pairs used in this work should be determined experimentally.

f) The new version of the square gradient theory should be modified, with respect to its predictions of interfacial tension for polydisperse systems.

g) Theories capable of predicting the effect of compatibilisation on interfacial tension between polymers should be developed.

3. CONTRIBUTIONS TO KNOWLEDGE

a) A special syringe to eliminate "necking" and "capillary" effects usually encountered in measurement of interfacial tension of polymer melts with the pendant drop apparatus was designed, tested and found effective.

b) It was shown to be possible to infer the interfacial tension between polymer melts from transient measurements using an exponential fit of the transient interfacial tension data, for both the pendant drop method and the spinning drop method.

c) Interfacial tension values were measured for polymer pairs that have not been previously reported. In particular, the interfacial tension values between polypropylene and polystyrene, polyptyrene, polyptyrene, polypropylene and ethylene vinyl alcohol copolymer and maleated polypropylene and ethylene vinyl alcohol copolymer were obtained. The measurements covered a wider range of temperatures and molecular weights than reported before.

d) It was confirmed that the interfacial tension decreases as a function of temperature in a linear fashion, for the polymer pairs studied. However, the temperature coefficient of interfacial tension depends on polydispersity.

e) It was shown that the interfacial tension increases with increasing molecular weight for the polymer pairs used in this study. It was demonstrated experimentally that the $M_n^{-\frac{2}{3}}$ (M_n number average molecular weight) dependence of the interfacial tension is not always observed.

f) Experimental values of interfacial tension for bimodal blends and polydisperse systems were presented.

g) The use of maleic anhydride grafted on the backbone of polypropylene as compatibilizer was shown to decrease the interfacial tension between polypropylene and ethylene vinyl alcohol copolymer. It was shown, using surface analysis, that this decrease in the interfacial tension values was probably due to a chemical reaction occurring at the interface. i) The new version of the square gradient theory and lattice theory were compared to experimental data. The predictions of the theory were found to be in agreement with experimental data for temperature and molecular weight, when a strategy proposed in this work is employed to calculate the Flory-Huggins interaction parameter. The Flory-Huggins interaction parameter used in the theory should be a function of temperature and of molecular weight.



NOMENCLATURE

- A : Interfacial Area
- C : Constant
- D_e : Diameter of the Pendant Drop at the Equator
- D_n : Diameter of the Pendant Drop located at D_ex1/n of the Equator of the Pendant
 Drop (n Integer)
- D_s : Diameter of the Pendant Drop ameasured horizontally at a distance D_e from the Apex of the Drop
- E : Energy of Mixing
- EP : Ethylene Propylene
- E.S.C.A: Electron Spectroscopy for Chemical Analysis

EVOH : Ethylene Vinyl Alcohol Copolymer

- F : Froude Number
- G : Shear Rate in an Extruder
- G : Gibbs Free Energy
- H : Correction factor for the Pendant Drop
- H : Enthalpy
- I : Index of Polydispersity
- J : Total Curvature of the Interface of a Spinning Drop
- LDPE : Low Density Polyethylene
- L_o : Initial Length of Fiber

MAgPP: Maleated Polypropylene

MAgPP₁: Maleated Polypropylene with 0.067 Maleation Content

MAgPP₂: Maleated Polypropylene with 0.098 Maleation Content

MAgPP₃: Maleated Polypropylene with 0.153 Maleation Content

MAgPP₄: Maleated Polypropylene with 0.168 Maleation Content

MAgPP₅: Maleated Polypropylene with 0.222 Maleation Content

MAgPP₆: Maleated Polypropylene with 0.262 Maleation Content

- MAH : Maleic Anhydride
- M_n : Number Average Molecular Weight
- M. : Viscosity Average Molecular Weight
- M_w : Weight Average Molecular Weight
- N : Degree of Polymerization
- N₆ : Nylon six
- P : Pressure

PaMS : poly(a-methyl Styrene)

PBDH : Polybutadiene

PDMS : Polydymethyle Siloxane

- PE : Polyethylene
- PE, : Polydisperse Polyethylene with a number average molecular weight of 770
- PE_b : Polydisperse Polyethylene with a number average molecular weight of 1,050
- PE_c : Polydisperse Polyethylene with a number average molecular weight of 1,950
- PE₁ : Monodisperse Polyethylene with a number average molecular weight of 680
- PE₂ : Monodisperse Polyethylene with a number average molecular weight of 1,050
- PE₃ : Monodisperse Polyethylene with a number average molecular weight of 1,950
- PET : Polyethylene Teraphtalate
- PP-g-AA: Polypropylene grafted with Acrylic Acid

PMMA: Polymethyl Metacrylate

- PP : Polypropylene
- PPpure : Pure Polypropylene
- PP_{com} : Commercial Polypropylene
- PS : Polydisperse Polystyrene
- PS_a : Blend of Polystyrene (M_n=59,900 and M_n=655)
- PS_b : Blend of Polystyrene (M_n=66,100 and M_n=695)
- PS_c : Blend of Polystyrene (M_n =52,700 and M_n =570)
- PS_1 : Monodisperse polystyrene with a number average molecular weight equal to 938
- PS₂ : Monodisperse polystyrene with a number average molecular weight equal to 1,589
- PS₃ : Monodisperse polystyrene with a number average molecular weight equal to 4,755
- PS₄ : Monodisperse polystyrene with a number average molecular weight equal to 19,417
- PS₅ : Monodisperse polystyrene with a number average molecular weight equal to 86,438
- PS₆ : Monodisperse polystyrene with a number average molecular weight equal to 380,000
- PVF₂ : Polyvinylidene Fluoride
- R : Gaz Constant
- R_o : Initial Radius of the Fiber, or of the Spinning Drop
- R₁ : Radius of Curvature of the Pendant Drop
- R₂ : Other Radius of Curvature in the Pendant Drop

- R_w : Radius of the Spinning Drop at Infinite Time
- R(t) : Radius of the Spinning Drop as a Function of Time
- S : Dimensionless s Coordinate
- S : Entropy
- S.E.M.: Scanning Electro Microscope
- S.I.M.S: Secondary Ion Mass Spectroscopy
- T : Temperature in Kelvin
- X : Dimensionless x Coordinate
- V(T,P): Specific Volume
- Y : Dimensionless y Coordinate
- W_a : Work of Adhesion
- W_c : Work of Cohesion
- Z : Dimensionless z Coordinate
- a : Radius of Curvature at the Apex
- b : Effective Length of the Monomer Unit
- d : Diameter of the Spinning Drop
- d_{app} : Apparent Diameter of the Spinning Drop
- d_d : Average Diameter of the Dispersed Phase in a Polymer Blend
- d_{real} : Real Diameter of the Spinning Drop
- g : Gravitational Constant
- g. : First term of a Taylor serie Development of the Gibbs Free Energy of Mixing
- m : Relaxation Time
- m₁ : Constant
- m_A : Molar Mass of Polymer A
- n : Corrective Factor and Optical Enlargment of the Spinning Drop
- q : Rate Constant in the Breaking thread Method
- r : Cylindrical Coordinate
- r. : Initial Spinning Drop Radius
- $< r_o^2 > /MW$: Characteristic Ratio
- s : Curvilign Coordinate
- t : Time
- t_o : Time at the Beginning
- x : x coordinate in the system (O,X,Y,Z)
- x_o : Volume Fraction of the Monomer with the lowest Molecular Weight
- z : z coordinate in the system (O,X,Y,Z)

ΔG_{m}	: Gibbs Free Energy of Mixing			
ΔH_m	: Enthalpy of Mixing			
ΔS_m	: Entropy of Mixing			
ΔΡ	: Difference of Pressure across the Interface of a Pendant Drop			
ΔP _o	: Difference of Pressure across the Interface of a Spinning Drop at $r=0$			
Δμ	: Change of Chemical Potential			
Δρ	: Density Difference			
Φ	: Angular Coordinate			
$\Phi_{_{GG}}$: Good and Girefalco Interaction Parameter			
α	: Amplitude of the Oscillation for the Breaking Thread Method			
α,	: Amplitude of the Oscillation for the Breaking Thread Method at $t = 0$			
δ	: Solubility Parameter			
δq	: Heat Absorbed by the Thermodynamical System			
δw	: Thermodynamical Work			
δw _{norm}	: Work Associated to the Change of Volume			
Ŷ	: Interfacial Tension			
ንሞ	: Value of the Interfacial Tension at Infinite Time for a Pendant or Spinning Drop			
γ(t)	: Interfacial Tension as a Function of Time			
λ	: Time Scale			
χ	: Flory-Huggins Interaction Parameter			
XH	: Enthalpic Contribution of the Flory-Huggins Interaction Parameter			
χ _s	: Entropic Contribution of the Flory-Huggins Interaction Parameter			
η_d	: Viscosity of the Dispersed Phase in a Polymer Blend			
η"	: Viscosity of the Matrix in a Polymer blend			
η	: Effective Viscosity			
ν	: Segmental Volume			
ф	: Local Composition			
ρ	: Density			
ρι	: Density of the Spinning Drop			
ρ_2	: Density of the Fluid Surrounding the Spinning Drop			
ρ	: Monomer Density			
σ	: Surface Tension			
σ_{d}	: Non Polar Part of the Interfacial Tension			
σ_{p}	: Polar Part of the Interfacial Tension			
-	· Ontical Magnification Factor			

ω : Rotational Speed

 ω_A : Incompatibility Degree of Polymer A

.



APPENDIX A

This Appendix presents the programs used for the determination of interfacial tension with the pendant drop apparatus and the determination of the interfacial tension with the spinning drop apparatus.

1. PENDANT DROP APPARATUS

A batch file called *savenick.bat* calls the different programs useful for the determination of the interfacial tension from a pendant drop experiment. It requires the knowledge of three arguments:

1) The number of gray level for the digitization of the image (0 corresponds to 16, 1 to 64 and 2 to 256).

2) The name of the file in which all the transformed profile after transformation by the different programs will be stored with different extensions.

3) The threshold value for the edge detection. This value ranges from 0 to 255.

The listing of the batch file is given at the end of this appendix. Below the programs called by the batch file are described one by one. They are written in "C language" and need the functions of the S.I.I (Standard Image Interface) library. The listings of all the programs are also given at the end of this appendix.

1.1. od20 and odvga

od20 and odvga are two executable programs provided by Coreco Inc. with the frame grabber and the S.I.I library. They load the Oculus Driver (the frame grabber) and the VGA driver. The arguments of the programs correspond to specifications of the equipment.

1.2. od20grab

od20grab is also a program provided by Coreco Inc. with the S.I.I library. It digitizes the image of the drop. It can digitize an image in 16, 64 gray levels or 256 colours. The calibration for brightness and contrast is done automatically: the program performs calculations and sets the brightness and contrast to maximize the image quality. The programs is called with three arguments:

%1	0:	16 levels of gray
	1:	64 levels of gray
	2:	256 colors
-с		if an automatic calibration is required
%2 second argument in the bat		second argument in the batch file; here name of the tiff file.

<u>1.3 tiff</u>

tiff is again provided by Coreco with the SII library. The programs stores, loads and displays the header of the tiff file obtained after digitization done by *odgrab*. Here, it displays on the screen the digitized image of the drop.

The program is called with two arguments:

-t	0:	Saves a tiff file
	1:	Loads a tiff file
	2:	Displays the header of a tiff file
%2		Name of the tiff file. (second argument of the batch file)

1.4 edge-det

edge_det makes the edge detection of the pendant drop. The program extracts binary objects contours and their area from the image. The algorithm is based on the location and extraction of the binary contour (cntr) of the object. The algorithm scans the image in a raster fashion until it reaches a 'false' (pixel with a gray level above the threshold value) to 'true' (pixel with a gray level below the threshold value) transition not already marked. A marked transition indicates a previously extracted cntr. The algorithm extracts then the contour, by following the "edge" of the object until it returns to the starting position. Therefore, the edge of all the objects are detected. The program then calculated the area of all the objects detected and stores the chain code of the biggest object which is the drop.

<u>1.5. put in a</u>

put_in_a retrieves the edge file of the drop generated by *edge_det* and converts the chain code of the points in an ASCII format and stores the edge file of the drop. The edge detection is then displayed on the screen.

1.6. sorter

sorter sorts the experimental points of the drop in an order that is needed to do the shape comparison. The user is prompted to enter the y coordinate at which he wants the profile to start (in order to remove the syringe).

The points of the drop profile are sorted in the arrow direction (cf. Figure A.1). The median A of all the points is found and B and C are located 28 above A. The value 28 is chosen arbitrarily in function of the resolution of the screen. For the portion of the drop that is above B the first points from the left side are kept, for the points between B and C the first points from the bottom are kept and for the points above C the first points from the right are kept.



Figure A1 : Pendant Drop

1.7 smoother

A smoothing program is needed because of the finite resolution of the frame grabber.

The smoothing program is done piece wise (i.e. point by point replacement) along the whole profile of the drop. The program requires three parameters: the order of the polynomial for smoothing (quadratic or cubic), npow, the number of data points in the local target group, ns, the number of overall smoothing cycles, ntimes. The program starts by identifying ns contiguous points along the drop profile. The program iterates the smoothing loop by operation on the first ns points. The angle ω made by those ns points with the horizontal axis is determined by a linear least square regression. The middle point is rotated and translated according to the transformation below:

$$\begin{cases} x' = (x - x_m) \cos \omega + (y - y_m) \sin \omega & (A.1a) \\ y' = (y - y_m) \cos \omega - (x - x_m) \sin \omega & (A.1b) \end{cases}$$

where (x',y') are the coordinates of the points in the second coordinate system

(x,y) are the coordinates of the points in the first coordinate system

 (x_m, y_m) are the coordinates of the middle point in the first coordinate system.

The points are then smoother in the new coordinate system. After smoothing the coordinates of the midpoint in the first coordinate system are given by:

$$\left[xs_{m} = xs_{m}'\cos\omega - ys_{m}'\sin\omega + x_{m}\right]$$
(A.2a)

$$ys_{m} = xs_{m}'\sin\omega + ys_{m}'\cos\omega + y_{m}$$
(A.2b)

where (xs_m, ys_m) are the coordinates of the midpoint after smoothing in the first coordinate system

 (xs_m', ys_m') are the coordinates of the midpoint after smoothing in the second coordinate system,

 (x_m, y_m) are the coordinates of the midpoint before smoothing in the first coordinate system.

Each iteration produces the smoothed value for the midpoint of the target group. Therefore, the first ns/2 and last ns/2 points are not smoothed.

This program is based on the work of Anastasiadis [1988].

1.8 Shape Comparison

After a description of the main program, three aspects of the drop shape comparison program are discussed, a) the algorithm, b) the choice of the points for drop shape comparison, c) the type of error for evaluation of the drop shape comparison

1.8.1. Description of the program

The program *shape* makes a shape comparison between the experimental points after smoothing and the theoretical points found by solving Bashforth and Adams equation by a fourth order Runge-Kutta method.

Bashforth and Adams is first solved for a value of B (see Chapter 2) approximated either by the empirical formula of Huh and Reed [1971] or by the user. The empirical formula of Huh and Reed is given by:

$$B = [exp(-6.70905 + 15.30025S - 16.44709S^{2} + 9.92425S^{3} - 2.585035S^{4})]^{\frac{1}{2}}$$
(A.3)

where S is the ratio $\frac{D_x}{D_e}$, where D_e is the maximum diameter of the drop and D_s is the diameter of the drop located at D_e from the apex (see Figure 2.1). B is the dimensionless quantity defined as:

$$B = \frac{\Delta \rho g a^2}{\gamma}$$
(A.4)

A.5

where $\Delta \rho$ is the difference of the density between the two polymers, g is the gravitational constant taken as 980.41 cm/s², a is the distance from the apex to the center of the drop, γ is the interfacial tension.

A robust shape comparison between the experimental and the theoretical profile is then done. Three kinds of error are evaluated (they are discussed in 2.8.4). Then, the value of B is incremented and the comparison is performed again. For every kind error the optimal value of B is found.

The interfacial tension is then obtained from:

$$\gamma = \frac{\Delta \rho g a^2}{\gamma}$$
(A.4)

where $\Delta \rho$ is the difference of density between the two polymers, g is the gravitational constant, a the distance of the apex from the center of the drop and γ is the interfacial tension.

The algorithm of the program is mainly based on Anastasiadis'work.

1.8.2 Robust Shape Comparison

To compare n homologous points of two shapes (x_i, y_i) and (u_i, v_i) one set of points has to be transformed to correspond to the other one. The points to be transformed are rotated through θ , scaled through τ and translated by (α, β) as follows:

$$\begin{bmatrix} \mathbf{u}_{i} \\ \mathbf{v}_{i} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} + \tau \begin{bmatrix} \cos\theta & -\sin\theta \mathbf{u}_{i} \\ \sin\theta & \cos\theta \mathbf{v}_{i} \end{bmatrix}$$
(A.5)

Then α , β , τ and θ have to be chosen for the best fit. In the present program the optimization is done on five parameters since B has also to be optimized. The four parameters are evaluated separately and then the parameter B for the best fit is evaluated.

The optimization of the four scaling parameters is done in a similar way, as an example the optimization of τ is explained below:

For each pair of homologous points i and j, a value τ_{ij} that is the scale factor can be defined as follows:

$$\tau_{ij} = \frac{\{(\mathbf{x}_j - \mathbf{x}_i)^2 - (\mathbf{y}_j - \mathbf{y}_j)^2\}}{\{(\mathbf{u}_i - \mathbf{u}_i)^2 - (\mathbf{v}_j - \mathbf{v}_j)^2\}}$$
(A.6)

where (x_i, y_i) and (x_j, y_j) are the coordinates of the two points in the first shape and (u_i, v_i) and (u_j, v_j) are the coordinates of the homologous points in the second shape.

Comparing the n homologou: points 1/2n(n-1) values of τ_{ij} can be evaluated. They are combined taking the doubly repeated median; the scale estimate is then equal to:

$$\tau = \operatorname{med}(\operatorname{med}\tau_{ij}) \tag{A.7}$$

The rotation factor is evaluated similarly as:

$$\theta = \operatorname{med}(\operatorname{med}_{ij})$$
(A.8)

And the translation factor are taken as the simple medians:

$$\alpha = med\{x_i - \tau(u_i \cos\theta - v_i \sin\theta)\}$$
(A.9a)

$$\left\{\beta = \text{med}\left\{y_{i} - \tau(u_{i}\sin\theta + v_{i}\cos\theta)\right\}$$
(A.9b)

The robust shape comparison method was first used by A.F. Siegel [1982] to compare skulls of primates. A listing of the program written in FORTRAN can be found in his book [1982].

1.8.3. Choosing of the Points

To perform the shape comparison it is essential to choose points from the theoretical and the experimental profile that can be compared (in other words it is important to compare bananas with bananas and not with coconuts). In this program, the position of the points in reference to the apex is taken as a reference point.

To determine the apex it is necessary to determine first the axis of symmetry of the drop. If the drop is not aligned with the camera, the axis of symmetry is determined by statistical analysis (the algorithm of this routine can be found in Press [1990]). The intersection of the axis of symmetry and the bottom of the drop determines the apex of the drop.

Then, the slopes of all the line segments from the selected points to the apex for the experimental profile are calculated. The theoretical points are then chosen in order to match the slopes of all line segments connecting apex and chosen points of the experimental profile.

1.8.4 Types of Errors

To evaluate the shape comparison analysis three types of error were calculated:

$$SR = \frac{1}{F} \left[\frac{1}{N} \sum_{i} \left[(u_i - x_i)^2 + (v_j - y_j)^2 \right]^{\frac{1}{2}} \right]$$
(A.10)

SRR =
$$\frac{1}{F} \left[\frac{1}{N} \sum_{i} \left[(u_i - x_i)^2 + (v_i - y_i)^2 \right] \right]^{\frac{1}{2}}$$
 (A.11)

TENSE =
$$\frac{1}{F} \left[\frac{1}{N-1} \sum_{i} \left\{ \left[(u_i - x_i) - (u_{i-1} - x_{i-1}) \right]^2 + \left[(v_i - y_i) - (v_{i-1} - y_{i-1}) \right]^2 \right\} \right]^{\frac{1}{2}}$$
 (A.12)

with

$$F = \left[\sum_{i} (u_{i}^{2} + x_{i}^{2}) - \frac{1}{2}N\left[\sum_{i} (u_{i} + x_{i})\right]^{2} + \sum_{i} (v_{i}^{2} + y_{i}^{2}) - \frac{1}{2}N\left[\sum_{i} (v_{i} + y_{i})\right]^{2}\right]^{\frac{1}{2}} (A.13)$$

where N is the number of points compared, (u_i, v_i) and (x_i, y_i) are the coordinates of the two sets of points to be compared.

SR represents a simple root mean square residual, SRR is defined so that it is more resistant to outlying points, TENSE measures the "stress" required to "deform" the drop.

1.9 theoreti

Once the best value of B is determined *theoreti* prompts the user to enter it and displays the experimental profile and half of the theoretical profile superposed.

1.10 display

Display is a program that can display up to ten drops superposed.

2. SPINNING DROP APPARATUS

The digitization of the spinning drop image is done by the same programs as the digitization of the pendant drop. The measurement of the diameter of the drop is done either manually with a program called *curs* or with a program called *spinning*.

Both programs are discussed briefly below:

2.1. curs

curs is a programs provided by Coreco with the S.I.I library. A cross hair cursor can be manipulated. The position and level of gray of the pixel the cursor points can be indicated.

2.2 spinning

This program detects the edge of the spinning drop, the distance between the two straight lines of the drop is calculated and the interfacial tension is calculated from:

$$\gamma = \frac{\Delta \rho \omega^2 d^3}{32}$$
(A.14)

where $\Delta \rho$ is the difference of the densities of the two polymers, ω is the angular velocity of the spinning drop apparatus and d is the diameter of the drop.

REFERENCES

Anastasiadis S.H. [1988] : "Interfacial Tension of Immiscible Polymer Blends", Ph.D. Thesis, Princeton University, (1988).

Huh H.C., Reed R.L. [1983] : J. Colloid Interface Sci., Vol 91, p472, (1983).

Press [1990] : "Numerical Recipes in C: The art of Scientific Computing", New York, Cambridge, University Press, (1990).

Siegel A.F. [1982] : Biometrica, Vol 69, p 242, (1982).

Siegel A.F., Launer R.L. [1982]: "Modern Data Analysis", New York, Acadamic Press (1982).

```
(last modified: Mon Aug 28 14:27 1991) al/od20grab.c
                                                                                                                                                                                   (fast modified; Mon Aug 26 14:27 1991) at/od20grab.c
 #include Catdia.h>
#include Catl.h>
                                                                                                                                   main( argc, argv)
                                                                                                                                   int argc;
char sargv[];
 #include (ctype,h)
#include (string,h)
#include (ctype,h)
#include (string,h)
                                                                                                                                        int mis, retu;
Int mis, retu;
Static char stitle = "SII --> 0028 Grab Utility, v:1.88";
                                                                                                                                       printf("Coreco Inc, Xm\n\n", title);
  VERSION 1.1
                                                                                                                                        If argc > 0) argv[0] + HUL: /= Strip off program name #/
 HISTORY
1.8 March 17th, 1989 by D.T.
                                                                                                                                        /* display information on how to use this program */
if(getarg( argc, argv, "?", NULL))
                                                                                                                                           1.1 Hey 38, 1383, Geg
Start with argv[8]
Allows MSLL option buffer.
 SS
NAME: GETARG
 DECLARATION:
 static int getargi args, argv, switch, option)
 int argc;
char sargv[];
char semitch;
char seption;
                                                                                                                                        /= Initialize SII environment =/
mil_init[ argc, argv);
if[ (ets = getminer()))
 INDUTE
                                                                                                                                            τ
printfi"ese FATAL_ERROR: XB ese\n", silerrmsg[ sts)); /e Print error usi
if[sts == INCOCV)
       argo, argv: standard ARGC and ARGV command line arguments
                              pointer to the desired mutch character 
whe NEL pointer to get the remaining command line
        mitch
                                                                                                                                                printf("An OOx driver is associated with each device\n");
printf("Tou must load 0028 and 00VGA drivers before calling this progra
                              argument
                              pointer to an array that will contain the atring
associated to a switch.
MULL supported.
                                                                                                                                            exit(1);
       option;
                                                                                                                                       retv = do_application1 argc, argv);
 Outputs
                                                                                                                                       if( ists + getsilerr()) printf("\nSIIER; Xs\n", silerrmg( sts));
else printf("Successful\n");
                              if option is NULL, no output,
if switch is a valid switch, it contains the associated
      option:
                              if multi-
if multi-
processed yet
                                                                                                                                       si1_end();
1f( retv) = exit( 1);
else exit(0);
 Return Value:
                                                                                                                                    int do_application( argc, argv)
                                                                                                                                   Int argc;
that sargv[];
       -1 if e.k.
B The desired switch is not on the command line or there is no other
command line argument to process.
                                                                                                                                       int mode, save, calibration;
LUT molut;
 Description:
                                                                                                                                       IV etw_box;
char_buf[10];
char_filename
      Search for a given switch or the next argument on command line.
If found the associated string is returned and the switch is replaced
by a MLL pointer in argv.
                                                                                                                                                          ernes,
23
we/
static int getargi argc, argv, sw, option)
int argc;
char asrgv(2;
char sugv(2;
char sugtion;
d
                                                                                                                                        /s Open Image Vindows s/
if( Get_iw()) return( -1);
                                                                                                                                       /s get mode (default = 0) s/
if(_getarg(_argc, argv, "p", buf))
                                                                                                                                             sscanft buf, "%d", &mode);
lf{ mode 1= 0 && mode 1= 1 && mode 1= 2) mode = 0;
      4
1nt 1;
                                                                                                                                       /* Check if automatic calibration */
if(getarg( argc, argv, "c", NALI) calibration * 1;
else calibration * 8;
      for( 1 = 0; 1 < argc; ++1)
           /= Not a NEL pointer =/
if[_argv[1])
                                                                                                                                       /* Check if filename given to save image #/
iff getarg( argc, argv, MRL, filename)) save * 1;
also save * 0;
                ift ......
                     .
/= Is it a switch 7 s/
if{ sarge[1] as '-' ]] earge[1] as '/')
                                                                                                                                        /* Allocate Output LUT */
if(_lolut + lut_alloc( ied, LUT_OUT)) ** NULL)
                         / Is it the wanted switch 7 e/
ift tolower( argv[1][1]) += egu)
                                                                                                                                            {
printf( "Nemory Allocatin Error\n");
return( -1);
                             iff option t= NLL) strcpy( option, &argv[1][2]);
argv[1] = NLL;
returnt =1);
                                                                                                                                        /s Load 16 Gray Levels Butput LUT e/
lut_slop( olut, 8, 8, 15, getiutcol( olut));
lut_sel( olut);
                        >
               /n get remaining command line argument e/
else
                                                                                                                                       /= Calibrate in monochrome mode for continuous grab e/
if( calibration)
if( calibration)
                   ;
iff option := MULL; stropy( option, argv[1]);
argv[1] = MULL;
return' -t);
                                                                                                                                                printf("Not Enough Hemory to Perform Calibration\n");
return( =t);
                                                                                                                                       /a Continuous Grab: 328x200, 15 Gray Levels a/
              >
     returnt 81:
                                                                                                                                            iw_grab( twpcm, 1);
tw_copy( twdcm, twpcm);
 static int calib( int);
                                                                                                                                       while( tkbhit());
                                                                                                                                        /s Calibrate image if color image e/
if( calibration && mode == 2)
if( calib( mode))
 #1fndef min
 Stofine min(a,b) 
Sendif
                                           (((a) < (b)) 7 (a) : (b))
 adefine COLOR_PROTECTED &
                                                                                                                                                printf("Not Enough Hemory to Perform Calibration\n");
                                                                                                                                                 return( -1);
IV stud;
IV studen;
IV stugen;
IV stuge;
IV stuge;
IV stuge;
IV stuge;
                                  /e Display IV e/
/e IDSDEP or smaller IV for display in Continuous Grab e/
/e IDSDEP or grabbing in Continuous Grab e/
/e IG Gray Levels IV, veriable resolution e/
/e EG Gray Levels IV, veriable resolution e/
/e EGIOT IV, veriable resolution e/
                                                                                                                                        /= Grab a test image with the desired Pixel Size =/
                                                                                                                                            case to
                                                                                                                                                is_grab! tep4, 11; /s Grab a 15 Gray Lovet image s/
is_copy( ted, teg4); /s Display with the VGA driver a/
break;
 static int Get_1w();
static void Free_1w();
                                                                                                                                            Came 1:
              0021GRAB
                                                                                                                                                ut_slop( olut, 0, 0, 63, getlutcol( olut));
lut_sol( olut); /s Lond a 54 Gray Level LUT e/
is_grab( isg6, 1); /s Grab a 54 Gray Levels image a/
 Names
 60/
                                                                                                           (4 columns)
                                                                                                                                                                                                                                             4 columnal
```

```
(last modified: Mon Aug 26 14:27 1991) at/od20gmb.c
             to copy( ted, teg6); /e Display with the VGA driver e/
/= Clear Output LUT e/
metiutpage( clut, LUT_ECS);
twt_flat( clut, 8);
 fund i f
                 /= Display with the VGA driver e/
iw_grab( iwg12, 1);
                 /m Copy the image from the Oc-28 to the VGA board through
a 12-bits to 8-bits conversion a/
im_colacale(imd, img(2);
                 /e Load the 256 colors LUT e/
lut_sel( elut);
            breek:
        •
    if( save)
         iw_box = ia_wize( iwd);
iw_mave( iw_box, filename);
iw_free( iw_box);
    lut_free( olut);
    Free_1w();
return( #);
                         /* Deallocate ALL IVs */
 /**/
/**
Name:
           Get_14
Free_14
Declaration:
int Get_is()
void Free_is()
Description:
    Get_iw opens TWs from Oculus-20 and VGA drivers.
    Free_iw frees all IVs allocated by Get_iu.
88/
 int_Get_iu()
    {
int pixmi2;
IV mim;
int org;
    /s Keep the Origin of stdiw s/
org = getorg( stdiw);
    ind = index = NLL;
ing6 = ing6 = ing12 = NLL;
    ind = stdim; /= use standard in for display s/
    /* Create window for displaying continuous grab m/
iff (iwdcm = iw child( iwd, 0, 0, min(getxien( iwd), 320), min(getyien( iwd),
return( -1);
    /e----- Open Oculus-28 Acquisition IVs --
while( (iw = iw_open( O_RD | O_ACD, arg)) 1= NLL)
        pixsiz = getpixsiz( iw);
switch( pixsiz)
             case 4:
1f(_getylen(_1w) <= 200)
                     else
1wg4 = 1w;
break;
             case 6:
iwg6 = iw;
break;
             case 12:
19912 = 19;
break;
            $
    petmiierr();
return( #);
void Free_iu()
    /s Free IV related to the Display s/
is_free( isdcm);
    /e Free IV related to the Acquisition e/
im_free( imp();
im_free( imp();
im_free( imp();
 static int calib( mode)
 int a
   int a, 1;
int index;
leng ehisto, ehisto_col;
int range, offset;
```

4 columna

(het modified: Mon Aug 26 14:27 1991) at\od20gmb.c tong manvalue = 0L, minvalue; int highlim = 0, lowlim = 0; printf("Calibrating...please wait.\n"); 1f1 mode == 2) /m colar mode e/ (metadcrange(img12, 255); setadcoffset(img12, 8); im_grab(img12, 1); /e make a color histogram s/ if([histo_col = [long e] siis[loc([unsigned] 4856 e sizeof[long]]] == b return[-1]; 1=_h1stol_1ug12, h1sto_col, 4896); n = 168; 1f(histo = llong =) siislloct (unsigned) n = sizeof(long))) == MULLI { alifree((char e) histo); return(-1); /s make a 160 ontry histogram (index ontry + (38% a R) + (68% aG) + (18% for(i = 0; $i \in A$; histoli+=1 + 0 kL); for(i = 0; $i \in \{4360; i + 1\}$ index = { 1 & 8x8887} + Get (1 & 8x8878) >> 4} + 3et (4 & 8x8788) >> himtofindex2 ++ himto_col[1]; stiffeet (char a) histo_cot); else /* monochrone histogram s/ /s grab with 64-gray levels s/ setsdorangel img6, 255); setsdorffset(img6, 8); - 64 n = 62;)#_grabl 10g6, 1);)f(l histo = {long =} #lialloc((unaigned) n = mizeof(long))] == MLL) feturn(=1); im_histo(log6, histo, n); } for(i = 0; $i \in n$; $i \neq i$) If histo(i) > maxvalue) maxvalue = histo(i); minvalue = manvalue/100L; i = 0; /= find points = 1% of maxvalue =/ while(1 < n, histo[1++] < minvalue); lowlime = = 1; 1 = n = 1; while(1 > -1, histo[1--] < minvalue); highlim = 1 + 1; offset = (255elowlim)/(n ~ 1); range = (255el highlim = lowlim))/(n = 1); milfreet (char e) histo); /s set new range and offset s/ lft mode == 2) netadcranget img12, range); setadcoffsett img12, offset); etse setadcrange(iwg6, range); setadcoffsett iwg6, offset); return(0);

14 columnal

(last modified: Mon Oct 22 02:20 1990) 187,0

```
Finclude (atdia.h)
Finclude (atdia.h)
Finclude (atdia.h)
Finclude (atl.h)
Finclude (tiff.h)
Finclude "getarg.h"
                                                                          /s Required by all SII applications o/
/s Required only for checktif() o/
/s Define the getarg function s/
  /s Prototypes that can be exported to another application s/
void checktiff char =);
  int maint int args, char exarge)
          int error;
int action;
cher filenamel88
cher option[28];
                                                                                                  /* Error Code from SII */
                                               w(88):
           static char buf(1824);
          printfi"Coreco Inc., SII Demonstration: TIFF file <--> IV Utility, v: Xs\n\n"
ergv[8] = M&L; /# Strip off program name #/
          /# Check if -7 was selected by the user #/
/# If so, then print description of program to stdout #/
/# And exit to DOS #/
figetargi argc, argv, "7", MULL);
                 printf("Description:\n");
printf("\n");
printf(" This program allows you to:\n");
printf(" - Save aldiw to a TiFF file\n");
printf(" or - Load a TiFF file to aldow.\n");
printf(" or - Displey a TiFF file Header to atdout.\n");
printf("\n");
printf("\n");
printf("\n");
printf("\setted before the to atdout.\n");
printf("\
                   Printf("Description:\n");
                                                                                                           Description\n\n");
Select Operation\n");
0: Load Tiff File to stdiw\n");
1: Seve stdiw to Tiff File\n");
2: Display Tiff File Hender to stdout\n");
                   printff"
                  printf("\n");
printf("\n");
printf("\n");
printf("\n");
printf("\n");
exit([];
}
                                                                                                   /* Exit to DOS #/
         mii_init( argc, argv);
error = getmtierr();
if( error)
                                                                                                  /= Initializes SII and binds to driver(s) =/
/= Gets error status =/
/= Exits on fatal errors =/
                   printf("see FATAL_ERROR: X3 ===\n", silerrmag( error]); /= Print error u
lf( error == IMODEV)
                           printf("An OOX driver is associated with each Coreco device\n");
printf("You must load at least one OOX driver before calling this progr
                  າດາ້ານ:
          iff getarg( argc, argv, "T", option)) action = atoi( option);
if ( igetarg( argc, argv, MLL, filename))
                  printf("No filename specified.\n");
printf("You must specify the name of the tiff file (without the extension
                                      àoi
                   .....
                   witt t):
          ewitch( action)
                   CASE Up
                          ae e:
_printf("Retrieving image from disk..");
iw_load( stdiw, filename);
_ut_sel( getolut(stdiw)); /# Load Physical lut with lut retrieved f
                            iff_terror = getuiterr()) t= 8)
                                    printf("\nSIIERR: Xa\n", silerreag( error));
if( error == IIO)
                                             prntifferr( buf, 1024);
printf( "Extended Information:\nxx", buf);
                           else printf("Successful\n");
break;
                   case 1;
printf("Saving image to disk..");
iw_save( stdiw, filename);
                           if( (error = getailerr()) t= 8)
                                   printf("\nSIIERR: %s\n", siserrang( error});
                           eise printf("Successful\n");
break;
                   case 2:
                          checktif( filename);
break;
                  >
         sil_end();
return( 0);
  /••
Description:
Send to stdout the current tags of the tiff file.
(To debug incompatible tiff files)
es/
disfine Readfile( ptr, size, cnt, fp)
disfine AbsolutoSeek( fp, offast)
disfine SciativoSeek( fp, offast)
disfine TUTAL TAGS 65
                                                                                                                                               freed( (cher e) (PTr), (size), (c
fmeek( (fp), (offmet), 0)
fmeek( (fp), (offmet), 1)
 Adefine FIRST_TAG NewSubfileType
void checklif( filename) 
char efilename;
```

static char stagname[] + { "NeeSubflieType", "NeeSubfileType", "SubfileType", "ImageVidth", "ImageLength", "Bitm"erSample", "Compression", "141 "Compression", "NonStandardTiff", "MonStandardTiff", "Photometric", "Threeholding", "CellVidh", "CellVidh", "FillOrder", NonStandardTiff NonStandardTiff" "NonStandardTiff", "DocumentMame", "ImageOvecription", "Make", "Model", "StripOffacts", "StripOffests", "Oriontation", "NonStandardTiff", "NonStandardTiff", "Samplesfsrfixel", "RowsferStrip", "StripByteCounts", "HinSampleValue", "MaxGampleValue", "Xtespolution", 'Ylesolution" "PlanarConfiguratic "Pagetase" "XPosition", "YPosition", "Freedfisters", "Freedfisters", "Freedfisters", "GraySemponadunt", "GraySemponadunt", "GraySemponadunt", "GraySemponadunt", "GraySemponadunt", "NonStandardTiff", "NonStandardTiff", "PlanarConfiguration", NonStandardTiff ColorResponseUnit Colories DonseCur "ColofferponseCurv "NonStanderdTiff", "NonStanderdTiff", "NonStanderdTiff", "Software", "DateTime", "MonStanderdTiff" "NonStandardTiff" "NonStandardTiff" "NonStandardTiff" "NonStandardTiff" "NonStandardTiff" "NonStandardT)ff" "NonStandardT)ff" "NonStandardTiff" "Artist", "HostComputer", "Predictor", "VhitePoint", "PrimaryChroma", "ColorMap" ÷ static char usizename() = { "Byte", "ASCII", "Short", "Long". "Kational" 1: TIFF_HEADER head; TIFF_TAG tag; Int 1, count; FILE efp; char fname[80]; stropy(fname, filename); stroat(fname, ",tif"); /s Open file s/ ifi (fp = fopen(fname, "rb")) == MRL) return; /s Get Tiff header s/ ReadFile(ichar s) Ehead, sizeof(TIFF_MEADER), 1, fp); /= Check Header =/ if(head.ByteOrder 1= 0x6343) printf("Tifferr: TiffHotorola\n"); alse printf("TiffVersion: Xid,Xid\n", head.Version/18, head.VersionX18); /s Got the tags s/ AbsoluteSeek(fp, had.OffsetOfIFD); ReadFile((char s) &count, sizeof(int), 1, fp); printf("X20s X25 X25 X154/N1", "Tag Name", "Size", "Count", "Offset"); for(1 * 8; 1 < count; 1**) { { int index; Read lie(Etag, sizeof(TIFF_TAG), 1, fp); index = teg.type - FIRST_TAG; if(index < 0 [1 index > TOTA_TAGS] sprintf(fname, "NonStandardT)ff(x else stropy(fname, tegname[index]); printf("X215 X56 X51d X5d\n", fname, sizenamel tag.Sizel, tag.Length, tag.Offset.u[0]); els frintf("X29s XBs X5id X5id\n", fname, sizename(tag.Size), tag.Length, tag.Offset.1); > ¥

÷

fast modified: Mon Oct 22 02:20 1990) Wite

fclose(fp); }

(3 columna) 3

•

(fast modified: Mon Sep 16 17:21 1991) s:\edge_detc (last modified: Mon Sep 18 17:21 1991) alledge deta option I- DISPEAT; /s Option to display features s/ Program edge_det version 1 Programed by Ricole E. Demarquette under the supervision of Professor Zamal September E. 1331 getargi argc, argv, "d", stri) 68 68 68 68 68 69 69 69 141 /s Set display mode #/ (option |+ DISPONTE; /= /= Option to display CATTE =/ if(strchr(str, 's')) option (+ SINCLESTEP; if(strchr(str, 'c')) color + 1; /= User select color monitor = User select color monitor #/ Edge_det extracts binary objects contours and their area from the image. The algorithm is based on the location and extraction of the binary contour (cntr) of the object. The algorithm acans the image in a raster fashion unit it resches a 'faise' to 'true' transition not eiready marked. A marked transition indicates a previously extracted cotr. The algorithm extracts the contour, by following the edge of the object until it returns to the starting position. This program extracts the same of each object detected and stores the chain code of the bigget object which is the drop. ift getargt argc, argv, "T", NULLIE /a Set timer flag e/ /w Disable all options if timer req timer = 1; option = 0; ... getargi args, argv, "g", str)) 111 /s Get threshold from user s/ sscanft str. "Xd", &thrs); finclude (atdia,h) finclude (atdlib,h) finclude (atl,h) finclude (at,h) finclude "chrona,h" finclude "getarg,h" if(getarg(argc, argv, "n", strl) /e Required by sil SII applications e/ /e Required for wait function e/ /e Required to get processing timer e/ /e Define the getarg function e/ binmode + biniut[stoid stril; /s Set new binmode #/ /s Get file name s/ ift igetarg (argc,argv,MLL,string)) trintf ("No file name specified.\n"); printf("Tou must specify the name of the tiff file (without the extension a)i_end(); ewit(1); static void loadcolorf LUT elut); static void loadmonol LUT elut); static int do_cntrf IV elw, unsigned featmask, unsigned option, DOTOP etseq); adefine HACODT adefine HSCOLOR /s Size of DOTOP vector used to store the chain /s Humber of cotors for pseudo coloring e/ /s Hust be a power of 2 e/ 4436 /estretation of the different functions. This block does the initialization of the different functions. contrask + cntr_first(stdiw); /s Get the cntrask value s/ cntr_end(stdiw); /s fust follow cntr_first s/ /# Option flags for do_cntr() =/ #define DISPCKTE = #x8881 #define DISPFEAT = #x8882 #define SIMGLESTEP = #x8886 /s Display cntrs over the image s/ /s Display features to stdout s/ /s Single step between each object s/ /s Check for errors s/ If Cotrask (1) printf("Not enough memory\n");
si1_end(); /= bits used by the cntr_s implementation (dev) /= bits used for coloring =/ static int cotreask = 0; static int coloreesk = 0; mil_end(
exit(1);
} colormask = (MSCOLOR-1) = (cntrmask+1); /= Compute the color mask =/ setbinthrs(stdiw, binmode, thrs); /= Select the binary threshold =/ Hain program starts. -----/e Erase mil bits used to mark contours e/ /e This can be done when performing acquisition or binarization e/ /e This include colormask if DISPCHTE requested e/ is_moki stdie, getmaxgri(stdie) & "(cotrmmak | (option & DISPCHTE ? colormas maint int args, char /= Error Code from SII e/ /= Error Code from SII e/ /= Facture mask e/ /= Getion flag for do_cntr e/ /= Color monitor flag e/ /= timer flag e/ /= Gray level for binarization e/ int error: char str[20]; crum wtric(0); unsigned featmask = 8; unsigned option = 8; int color = 8; int timer = 8; GLLV_thrs = 120; DOTOP =tmeq; int : if(option & DISPONTE) /s Check if ontr display requested {
 LuT stut;
 Lut = lut_slipc(stdtw, LUT_GUT);
 tf(color) loadcolor(tut);
 else loadscolo(lut);
 lut_set(lut);
 lut_free(lut); /e Allocate an output LUT e/ /e Load lut with pseudo colors e/ /e Load lut with highlight gray lev /e Load physicat LUT e/ /e Free lut handle e/ totor «tang; int 1; FIL[endge; int binmode = BIN_NOTBIT; int montr = 0; static int binlut[6] = /= Default binarization mode =/ /= Number of cntrs extracted =/ /= this table converts value from user to BIM_= if(timer) TRIG_CHROND; maxdot + MADDOT; if(Itseq + (DOTOP +) malloc(sizeof(DOTOP) = maxdot)} == NULL) return(B); ncntr = do_cntr(sidiu, festmask, option,tseq); (BIN_WHITE, BIN_BLACK, BIN_CRAY, BIN_BIT, BIN_HOTGRAY, BIN_HOTGRAY, ift timer) unsigned time; /* Processing time (mm) */ time * CHRDMO; /* Gets processing time (mm) */ PRHCHRDMO(stdiw, time); /* Frints processing time */ int mudot: char string[40]; argv[8] = NULL; /= Strip off program name #/ printf("Number of contour: Xd\n", nontr); /s Check if -? was selected by the user s/ /s And exit to DOS s/ if(_getargt argc, argv, "?", NULL)) sil_end(); getsistr): getsustr; edge=fopen(string,"wt"); for (i=0;fs=q[1];i=+) fprint(edge,"2d",(int)fs=q[1]); fclose(edge); free((char =) fs=q); return(0); (printf("Usage: cntreat [=d7,.] [=f7,.] [=g7] [=n7] [=t]\n"); printf("\n"); printf("Switch Default Description\n\n"); printf("d7. None Display contour extracted\n"); printf(" c1 Calor monitor (default is mon printf(" c1 Calor monitor (default is mon Description(\n'); Display contour extracted\n"); c: Color monitor (default is monochrome)\n") s: Step one object at a time (ESC to abort)\ Extract selected features from each object\n printfc printf("-f7... None Extract selected features from each 0 a Area\m"); [maxgrl+1]/2 Select gray level for threshold\n"); S Binarization mode\n"); A pixel is part of a cntr if:\n"); 0; igrl > threshold\n"); 1; igrl < threshold\n"); 2; igrl = threshold\n"); 3; igrl & threshold\n"); 4; igrl = threshold\n"); 5; igrl & threshold\n"); 5; igrl & threshold\n"); 6; igrl & threshold\n"); 7; igrl & threshold\n"; 7; igrl None printf(" printf("-g? printf("-n? /accesses to a contract all the cotra from side and extracts their area as then, it selects the cotr with the biggest area (which is the drop) and as stores its chain code. printfi" ... printf(41 0210771 Input: printf(" printf(" printf(" printf("-t printf("\n"); IV that is searched for catrs binary attributes must be defined correctly 191 -11 41 41 auttil): option: Processing Options (can be Ofed): DISPFEAT: Each feature selected using feataask is printed to stdout for each cntr. A new line is generated for each cntr. A beader line is printed before the table of cntrs. /second distributes the use of the sil library and checks if the drivers as are loaded. .. stignit(argc, argv); /> Initializes SII and binds to driver(s) e/ error = getsiler(); /> Gets error status e/ if(error) /> Exits on fatat errors e/ DISPORTE Each ontr is draw to stdiw using then entry is grave to stole using a different gray levels are used. If a pseudo color Output LUT is loaded, then each entr will be displayed with a different color. The function uses only the colormask bits of the pixel to draw. Other bits remain unchanged. 88 99 vintf("ease FATAL_ERROR: %% ease\n", %))errmag(error)); /* Print error u
if(error ** INCOEV) ... printf("An OOX driver is associated with each Corece device\n"); printf("You must load at least one OOX driver before calling this progr Assuming that pixsiz is 8, and entransk is 1, we have the following: 88 89 ex(1(1); 65 65 69 thrs = (getmangrit stdiw) + 13/2; /s Gets the threshold value s/ * * * d d d c 40 40 draw x: don't care bit d: bit used to draw the catr or centroid c: bit used to mark catra (internal to SII) 40 90 90 90 90 ift getargt argc, argv, "f", atr)) /s Set featmask by parsing the -f a iff strchr(str, 'a')) featmask |= F_AREA; (4 columns)

```
(as modified. Mon Sep 16 17:21 1991) at edge detc
                                                                            (fast modified: Mon Sep 16 17:21 1991) at/edge_det.c
                  teturn
ee The Chain code of the drop
second 
                                                                                                                                                                                                       /44/
                                                                                                                                                                                                        Description:
                                                         /s Pointer to the Cntr representation s/
/s Paxiaum size of seq s/
/s Jize of current cntr stored in seqs/
/s Feature structure s/
Humber of Cntrs s/
      -----
        Int se
                                                                                                                                                                                                            This function loads the LUT with a faine coint table.
      Int 11.0;
CHIRFEAT feat;
Int nCATF = 8;
Long Int areamax;
                                                                                                                                                                                                              Assuming that Pixel2 is 8, and Cotreash is 1, we have the following
                                                                                                                                                                                                                         * * * 1 3 1 3 6
      /s The seq buffer allocated only if necessary s/
/s The seq buffer stores the representation of the current cntr s/
/s The seq buffer can be drawn using is_dredots s/
if( option & DISPCKTR)
                                                                                                                                                                                                                                                              coler

    Don't care bit
    c: cntr reserved bit
    color: Select the index in color table to display pixel in color
    f color is 0, then no coloring occurs.
    Otherwise, select entry 1 in color palette.

               naxdot = MAXDOT;
tf( (seq = (DOTOP =) malloc( sizeof(DOTOP) = maxdot)) == MULL) return( 8);
       -130
                                                                                                                                                                                                       economic static void londcolor( LUT stut)
              maxdot = 0;
                                                                                                                                                                                                              int 1, j. colrange, index, color,
static unsigned egapailNBCOLOR3 =
       /m Print header of feature table m/
if( option & DISPFEAT)
                                                                                                                                                                                                                     iaii'
                                                                                                                                                                                                                                            /s DeepStue
/s Tellow
/s Ditve
/s Blue
/s Cysn
/s Purple
                                                                                                                                                                                                                                                                             (not used) a/
                                                                                                                                                                                                                     8x8ff
8x268,
8x768,
8xff8,
              f fontmask & F_AREA)
printf("\n");
                                                                             printf(" Area ");
                                                                                                                                                                                                                      81085
                                                                                                                                                                                                                                            /s Purple s/
/s Aque s/
/s Eed s/
/s CrassGreen s/
/s CrassGreen s/
/s Rogents s/
/s Soft Blue s/
/s Soft Blue s/
/s Trink s/
/s Creen s/
/s Dark Red s/
        if( cntr_first( iw) >= 8) /= Check if initialization successful =/
                                                                                                                                                                                                                      Exafo,
Exafo,
                                                                                                                                                                                                                     8x881
8x651
8x851
8x164
8x170
9x121
               ACOMMANNS-
               /w Get cntrs until all stdiw scanned m/
while: (n = cntr_next( iw, seq, maxdot, Efect, fectmask)) > #)
                      nGhtr++:
                                                                             /s Increment number of cntrm s/
                       if( option & DISPFEAT) /s Display value of features selected s/
                                                                                                                                                                                                                      axofo)
                                                                                                                                                                                                                      0x509.
                              ift featmask & F_AREA) printf("XGid ", feat.area);
                                                                                                                                                                                                                       3 -
                             printf("\n");
                                                                                                                                                                                                               /# egapal format: 0x0bgr (each color component uses 4 bits) #/
                      iff option & DISPONTE) /* Display cntr to stdiw */
                                                                                                                                                                                                               colrange = getlutcol( lut) = 1;
                                                                                                                                                                                                                                                                                                         /s Get the scale for fut colors =/
                              int color:
                                                                                                                                                                                                               setlutpage( tut, LUT_RED);
for( 1 + 1; 1 < NSCOLOR; 1++);</pre>
                                                                                                                                                                                                                                                                                                         /s Select red page of color lut s/
/s For each color in palette s/
                                                                                                     /e color for dreeted e/
                              color = "colormask;
color &= getmaxgrl( iw);
                                                                                                     /# Not of colormask #/
/# Limit to maxgrl #/
                                                                                                                                                                                                                       setops( iv, CP_AND); /= AND pixels of colormask =/
setgril( iv, color); /= this clear all bits of colormask =/
iv_dredots( iv, rg_x(iv), rg_y(iv), seq, n);
color = (ncht x.KGCLOR)=1; /= Select drewing color =/
color == Cntrmask + 1; /= Shift to colormask position inside
                                                                                                                                                                                                                              iff (j & colormask) == index) /s Check if index matched =/
lut_singlet lut, j, color); /s Write the lut entry e/
                                                                                                                                                                                                                            1
                                                                                                       /s Or each pixel e/
/s with a pixel mask that falls inside
                              setope( iw, GP_OR);
setgril( iw, color);
                                                                                                                                                                                                                       $
                                                                                                                                                                                                                 settutpage( lut, LUT_GREEN);
fort i + 1; i < NSCOLOR; i++}</pre>
                                                                                                                                                                                                                                                                                                           /e Select the green page of lut e/
/e for each color in patette e/
 {
    color = (egapal[1] >> 4) & Bx8f; /= Extract the green component =/
    color = color = colrange / 16; /= Scale color according to lut =/
    index = ; = (cntrmask = 1); /= Shift to color bit field =/
    for(j=0; j <= getlutind( lut); j=>) /= Scan each entry of lut =/

 See This block chooses the biggest area and stores the chain code in a edge file
sector of the sector of the biggest area and stores the chain code in a edge file
of (abs(feat.sree))areanax)
                            areanaxofeat.area;
printf("Number of contours=Xi AreaMax=Xid\n", ncntr, areanax);
strncpytiseq.seq.anaxdot=1);
                                                                                                                                                                                                                              if( () & colormask) == index) /= Check if index matched
iut_single( iut, ), color); /= Write the lut entry =/
                                                                                                                                                                                                                              ъ
                         $
                                                                                                                                                                                                                       $
                             te_drudots( ie, rg_x(ie), rg_y(ie), seq, n);
}
                                                                                                                                                                                                                 setiutpage( lut, LUT_BLUE);
for( 1 + 1; 1 < NBCOLOR; 1++)</pre>
                                                                                                                                                                                                                                                                                                             /s Select the blue page of lut e/
/s for each color in paiette e/
                                                                                                                                                                                                                       (
color = [egapal[1] >> 8) & 0x0f; /= Extract the blue component =/
color = color = colrange / 16; /= Scale color according to lut =/
index = ; = (cntrmask = 1); /= Shift to color bit field =/
fortj=8; ] <= getlutind[ lut); ]=+1 /= Scan each entry of lut =/
                     •
        catr_end( tw):
                                                                                /e Close catr extraction e/
        if( maxdot > 8)
free( (char =) seq);
                                                                              /s Free memory atlocated s/
                                                                                                                                                                                                                               if( () & colormask) == index) /= Check if index metched =/
lut_single( lut, ), color); /= Write the lut entry =/
        return( montr);
                                                                                                                                                                                                                              ¥
                                                                                 /s Esturn number of ontr extracted s/
                                                                                                                                                                                                                     •
                                                                                                                                                                                                               ÷
  It columnal
 Description:
        This function loads the LUT with a highlighting momochrome table INSCOLOR gra
        Assuming that pixel2 is 8, and entrask is 1, we have the following;
                     x x x 1 1 1 1 c
                                                             Incho
                                      Don't care bit
                 c: cntr reserved bit
calar: Select the index in color table to display pixel in monochrome.
If color is 0, then no highlighting occurs.
Otherwise, highlight.
 erresservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesservesserve
          int 1, j, index, color:
         setlutpage( lut, LUT_RCB); /s Vrite 3 planes of lut e/
for( 1 = 1; 1 < NECDLOR; i++) /s for each evaluable index e/
{
    color = getlutcol( lut); /s Get Maximum intensity evaluable e/
    index = i = (cntremant + 1); /s Shift to color bit field e/
    for{j=0; j <= getlutind( lut); j++) /s Scan each entry of lut e/</pre>
                         if( (j & colormask) == index) /= Check if index matched =/
lut_mingle( lut, j, color); /= Write the lut entry e/
                         $
                                                                                                                                                                      (4 columna)
```



dast modified: 3st Feb 29 19:33 1992) at/put in a.c

A.18

Gast modified; Sat Feb 29 19:33 1992) at/put_in_ac

```
Bast modified: Sal Feb 29 19:34 19923 at/sorter1.c
                                                                                                                                                                                                                                                                                            Sast modified: Sat Feb 29 19:34 1992) at\eorter1.e
 /storestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestatestonestate
                                                                                                                                                                                                                  THE FIRST POINT FROM THE RIGHT EDGE OF SCHEEN INWARDS VILL REPRESENT
THE EDGE. IN THESE PORTIONS, MORIZONTAL LINES ARE REJECTED.
                                                                                                                                                                                                         84
88
                                                                                                                                                                                                              HANTERENELENCY/2;
Hore the Higher Arbitrarly
Hote that this hay prove to be too large for shall
DROPS or too shall for large drops every
 88
80
          PROGRAM BOTTOR, C SORTS THE EXPERIMENTAL POINTS IN AN ORDER WHICH IS RELOED
To run the program shorther and the program shape. It also emaile the user
to get fild of the stringe.
                                                                                                                                                                                                          pointenumber_of_points;
pointenumber_of_points;
pointe=pointb=pointc=1;
lastyl=lasty;=lastxl==1;
                                                                                                                                                                                                          for(point=1;pointC+number_of_points;point++)
 #include (graphics.h)
#include (stdio.h)
#include (stdio.h)
#include (stdio.h)
#define MPDINT 5000
#define MPDINT 3000
#define MPDINT 1000
                                                                                                                                                                                                             if[dy[point](-mody && dy[point])top && dx[point](-modx]
                                                                                                                                                                                                                      if(dy(point))+Lasty1)
                                                                                                                                                                                                                               ytl[pointa]=dy[point];
ktl[pointa]=dx[point];
lasty1=ytl[pointa];
temps=ktl[pointa];
  void main(int argc, char ==argv)
                                                                                                                                                                                                                               pointa++:
FILE wdrop, eBorted, eresult;

fint number_of_Doints, 1, Aulp, Alp;

int lenc, rexo, medy, medx, erased_points;

int top, test, teaps, tempc;

int end, lowy, 11, entered;

int end, Novy, 11, entered;

int tribrolWT13, xt2(WPOINT33, xt10WPOINT23;

int xt1(WPOINT31, xt2(WPOINT33, yt3(NPOINT33;

int tasty1, lasty2, lastx1;

int points, points, point;

char string[23].ch;

int device=DETECT, mode;
                                                                                                                                                                                                                      .....
                                                                                                                                                                                                                                if(dx[point](tempa)
                                                                                                                                                                                                                                       pointa
                                                                                                                                                                                                                                       yti[pointa]=dy[point];
xti[pointa]=dx[point];
tempa=xti[pointa];
pointa++;
                                                                                                                                                                                                                                  ъ
                                                                                                                                                                                                                           >
                                                                                                                                                                                                                   .
nulp=points=1:
   if(dy(paint))medy)
                                                                                                                                                                                                                        ifids[point]!+[astal]
                                                                                                                                                                                                                               yt2[pointb]=dy[point];
xt2[pointb]=dx[point];
lasta:=xt2[pointb];
tempb=yt2[pointb];
pointb**;
  ctracr();
if ((drop + fopen(argv(12,"rt")) ++ NUL)
           printf("UNABLE TO OPEN DROP FILE");
exit(1);
   1#
          ((sorted=fopes(argv(2),"wt")) == MULL)
                                                                                                                                                                                                                       elsa
         printf("UNABLE TO OPEN SORTED FILE");
exit(1);
                                                                                                                                                                                                                                 if(dy[point])tempbl
                                                                                                                                                                                                                                     paintb--;
yt2[paintb]=dy[paint];
xt2[paintb]=dx[paint];
tempb=yt2[paintb];
paintb++;
   if ((result+fopen(argv[2],"wt")) == NULL)
PrintforUNABLE TO OPEN RESULT FILE");
exit(1);
                                                                                                                                                                                                                                 3
                                                                                                                                                                                                                             3
                                                                                                                                                                                                                     nip=nuip=pointb=1;
                                                                                                                                                                                                                     if(dy[paint]<=eedy && dy[paint]>top && dx[paint]>=eedx)
                                                                                                                                                                                                                            if(dy[point]!=(asty2)
                                                                                                                                                                                                                                   yt3[pointc]=dy[point];
xt3[pointc]=dx[point];
lasty2=yt3[pointc];
tempc=xt3[pointc];
pointc++;
            if (fscanf(drop,"%d %", &dy[point], &dx[point]) ++ [OF)
            point++:
        )
    number_of_points=point=2;
                                                                                                                                                                                                                            else
                                                                                                                                                                                                                                     if(dx[point])tempc)
   /as THIS BLOCK INITIALIZES THE GAMPHICS SO THAT THE PROFILE CAN BE DISPLAYED

• GRAPHICALLY ON THE SERVEN.
                                                                                                                                                                                                                                            peinto
                                                                                                                                                                                                                                            yt3[pointc]=dy[point];
xt3[pointc]=dx[point];
tempc=xt3[pointc];
pointc=+;
              ...
    initgraph(&device,&mode,MLLL);
if ((test-graphresult()) 1=0)
                                                                                                                                                                                                                                      5
          v
printf("\n\UMABLE TD INITIALIZE GRAPHICS");
printf("\nGRAPHEISULT= Xd", test);
exit(1);
                                                                                                                                                                                                                              >
                                                                                                                                                                                                                      ¥
                                                                                                                                                                                                                       number_of_points=nlp=pointc=1;
    cleardevice():
   ** NOW THE POINTS THAT DEFINE THE EDGE ARE STORED IN THE ARRAYS X() AND ** y() IN CORRECT DRDER.
                                                                                                                                                                                                                           topestoi(argv[4]);
                                                                                                                                                                                                                  ้มถึงโตษริงไปไ
    y[1]=y11[1];
                                                                                                                                                                                                               for (1=nu(p+1;1<=n1p;1++)
    ermsed_points=0;
lexxx=53;
rexx=lamy=0;
for [point=1; paint<=number_of_points; ++point)
                                                                                                                                                                                                                  x[i]=xt2[1-nulp];
y[1]=yt2[1-nulp];
                                                                                                                                                                                                                for(i+nip+t;i<=number_of_points;i++)
                                                                                                                                                                                                               x[1]=xt2[1=n1p];
y[1]=yt3[1=n1p];
          ïŧ
              if (dy[point] < top)
erased_points++;
             else
K
                                                                                                                                                                                                               for (1+1; 1<+number_of_points; ++1)
                        lexxmintlexx.dx[point]);
rexxmax(rexx.dx[point]);
lowymax(lowy.dy[point]);
                                                                                                                                                                                                                       fprintf(sorted,"Zd Zd\n",y[13,x[13);
putplie1(x[13,y[13,RED);
delay(18);
                   •
         >
                                                                                                                                                                                                             ;
closegraph();
printf("mulp-ad nlp-ad \n", nulp.nlp);
fprintf(result, "Xd Xd", nulp.nlp);
delay(2000);
fclome(drop);
fclome(drop);
fclome(sorted);
            THE POINTS ARE SORTED, made is the NEDIAN OF ALL THE POINTS
mady is the value of above the Lonest value of Y. For All Parts of the
PROFILE That are Loner Than mady. The FIRST POINT ENCOUNTERED FROM
BOTTOM UP VILL REPRESENT THE EDGE SO THAT VERTICAL LINES ARE NOT
RECORDED. FOR THE PORTION OF THE DROP BEFORE ITO THE LEFT OF THIS BOTTOM
PORTION! THE FIRST POINT ENCOUNTERED FROM THE LEFT EDGE OF THE SCREEN
VILL REPRESENT THE EDGE. FOR THE PORTION TO THE RIGHT OF THE BOTTOM,
    ....
                                                                                                                                                                           C CON
```

C conumnel

```
(last modified: Tue Sep 17 10:56 1991) at\smoother.c
                                            (ast modified: Tue Sep 17 10:56 1991) a:\amoother.c
                                                                                                                                                                      atime TIMES
 844/
44
49
40
40
40
40
40
40
40
40
40
40
                           ather.c
                      programmed by Nandini Das Gupta
under the supervision of Professor Eanal
8 july 1991
                                                                                                                         printf("\nworking....cycle number Zd",nt);
for (ic=lstart; iG <= lend; ic+i
                                                                                                                              for (jc = 1; jc<-ns; jc++)
                                                                                                                                   ic = ic + jc = lstart;
stjcl = aslicl;
     PROCRAM MMONTHER, CASES FOR TWO FILENAMES, DHE CONTAINING THE SORTED PDINTS OF A PROFILE, AND ONE WHERE THE POINTS OF THE PROFILE, ONCE SHOOTHED, WILL BE WRITTEN.
                                                                                                                                   y[jc] = my[ic];
/* END OF jc LOOP #/
                                                                                                                              > /* END OF jc LOOP #/
angl*angins);
angi * -1.8 # ang2;
xmnift * wfistart1;
ysnift * yfistart1;
for(itr+1;itr(+ns;itr+1))
     THE USER INFUTS THE PARAFETERS OF THE SHOOTNING: THE NUMBER OF POINTS USED IN EACH LINEAR RECRESSION INNI, THE ONDER OF THE POLYNORTHAL USED INPON, AND THE NUMBER OF TIMES THE SHOOTNING IS REFEATED INTIMO.
THE PROGRAM IS BASED ON AND POSTLY TRANSLATED FROM FORTRAN TO C
LANGUAGE FROM THE WORK OF SPRIDS ANASTASIADIS, PRINCETON
UNIVERSITY.
                                                                                                                                 ٠
                                                                                                                                   y[itr] = y[itr] = yshift;
x[itr] = x[itr] = xshift;
                                                                                                                              #[ltr] = #Lttr; = #mit
} /= END OF itr LOOP=/
rotate(ns,angl);
pesmo(ns,npoe,npoet);
 extern unsigned _stklen =48880;

rinclude (graphics.h>

rinclude (statis.h>

rinclude (conto.h>

rinclude (conto.h>

rinclude (stati.h.h>

rinclude (stati.h.h>
                                                                                                                              ylistert) = clil+cl2lex[istert];
                                                                                                                            y(istar) = clist(istart);
costcoaleng2;;
wint=sin(ang2);
x[ic] = x[latert]=cost = y[istart]=sint + xshift;
y[ic] = y[istart]=cost + x[istart]=sint + yshift;
> /s ENO Gr ic LOOP = y
for (i=istart; 1<=(end; i=)</pre>
float x[100],y[100];
float c[4],a[4][5];
                                                                                                                         {
    sy[1] = y1[1];
    ax[1] = x1[1];
    /* END OF 1 LOOP =/
    /* END OF 1 LOOP =/

 10(4)
 maintint args, char everyw)
                                                                                                                  void rotate();
double ang[);
void pusmo[];
                                                                                                                    THE SHOOTHED POINTS ARE WRITTEN TO THE FILE
                                                                                                                  for (3+1; 1Chtot; 1++)
   FILE exorted, esecothed;
   flt( esorted, esocothed;
int point, i;
int end, entered_test;
int int is(1000].jv(1000],
fleet ax(1000], my(1000], my(1000], y1(1000);
char str[00], my(1000], my(1000], y1(1000);
int istart,lend, ic_jc, itr,nt,ntime,lc;
float xmhift, ymhift;
double angt, ang2;
int ms,ntor,npoen,npoet;
int device=DETECT,mode;
                                                                                                                          fprintf(smoothed,"%s %s\n",sy[1],ax[1]);
                                                                                                                 THE GRAPHICS ARE INITIALIZED AND THE ORIGINAL PROFILE IS DISPLAYED
IN WITE WILLE THE SHOOTHED PROFILE IS DISPLAYED IN GREEN
                                                                                                                     initgraph(ldevice.lmode.HLL);
if ((test-graphresult()) f=0)
{
                                                                                                                          printf("error: graphresult =Xd",test);
fclose(secothed);
 fclose(sorted);
   THE TWO FILES ARE OPENED. THE CONTENTS OF THE SORTED FILE ARE
READ AND CONVERTED TO FLOATING POINT VALUES FROM INTEGERS.
                                                                                                                           QK11(1);
                                                                                                                     cleardevice();
printf["DEIGINAL DEDP PROFILE IN WHITE");
printf["\nSHOOTHED DEOP IN GREEN");
   ciner();
printf("DO YOU WART TO SHOOTH THE PROFILE (Y/H)7; --);
                                                                                                                     rewindisorted):
   gets(str);
if ((str[0]++'N')))((str[0]++'n'))
                                                                                                                     rewindiamonthed):
     x1110)
                                                                                                                     for(point+1; pointCentot; point++)
   if (isorted = fopen(argv[1],"rt")) == HUL)
                                                                                                                          putpixel(ix(point], iy(point], WITE);
        printf("CANNOT OPEN SORTED FILE");
exit(1);
                                                                                                                       ł
                                                                                                                     for (point+1; pointC=ntot; point++)
   if (isooothed + fopen(argv[2],"wt")) ++ HJLL1
                                                                                                                          ix[point]=ax[point];
iy[point]=ay[point];
putpixel(ix[point], iy[point],GREEN);
        printfe"CANNOT OPEN SHOOTHED FILE"1;
        exit(1);
                                                                                                                     delay(100001:
   rewind(sorted);
   point + 1;
                                                                                                                 end-0;
while (tend)
                                                                                                                    if (fscenfiserted,"%d %d", &sylpoint), &sklpoint)) ** EOF)
        end + 1;
ay[point]=iy[point];
ax[point]=ix[point];
                                                                                                                     fclose(smoothed);
                                                                                                                     closegraph[];
        point++;
                                                                                                                  ntat - point - 2;
                                                                                                                     "FUNCTION ANGL) APPLIES LEAST SQUARE REGRESSION TO A SET OF POINTS TO FIND THEIR AVERAGE SLOPE.
 THE USER IS PROMPTED TO ENTER THE PARAMETERS OF THE SMOOTHING
                                                                                                                      WARNING: IT DOES NOT HANDLE CASES WHERE THE AVERAGE SLOPE IS ZERO
I.E., WHEN DENORS.
                             clrscr();
print("PLEASE ENTER THE FOLLOWING PARAMETERS:\n\n");
print("MUMBER OF POINTS USED IN PIECEWISE SHOOTHING: ");
entered=8;
entered=8;
                                                                                                                     TRANSLATED FROM FORTRAM SUBROUTINE OF SPIROS ANASTASIADIS, "LREGK"
                                                                                                                 89
                                                                                                                 **********
                                                                                                                  double ang(ns)
                                                                                                                  Int na;
   while Lientered)
                                                                                                                    double sx, sy, sxy, sx2;
       printf("ORDER OF SPOOTNING POLYNOMIAL (2 OR 3):

1f ( ((npowsta)(gets(str))) > 1) &L (npow < 5))

entered=1;
                                                                        -1:
                                                                                                                    18T 1:
                                                                                                                    double dhum,deno;
                                                                                                                    $X*$y*$Xy*$X2+0.0;
for (1=1; 1C+n$; 1++)
   printf("NUMBER OF SHOOTHING ROUTINE CYCLES: ");
ntime-atoi(gets(stri);
                                                                                                                       ٠
                                                                                                                         BX = BX + x[1];
By = By + y[1];
BX2 + BX2 + x[1]ex[1];
BXy = BXy + x[1]ey[1];
 THE SHOOTHING IS DONE ATLASS TIMES. EACH TIME, THE HIDPOINT OF A POLYMONIAL FIT OF AS POINTS IS TAKEN FOR EACH POINT IN THE PROFILE. THE SHORTHING IS DONE BY THE FUNCTION PUSHO. THE POINTS ARE FIRST ROTATED AND SHIFTED.
                                                                                                                    dnus - hersky - skasy;
deno - hersk2 - skask;
1f (deno(3.8e-30)
             npout-npow + 1;
lutart-ns/2 + 1;
lend + atot - ns/2;
                                                                                                                         printf("deno=0");
                                                                                                                          exit(1):
   for intel; at (* atime; at++) /www THE SHOOTHING IS REPEATED
                                                                                                                    return(atan2(dnum,deno));
```

```
void rotate(ne,angl)
int ne;
double angl;
                                                                                                                                void exchange(npow,npow1,k)
int npow,npow1,k;
                                                                                                                                  int pivot_row, pivot_column.i,j,itemp;
float pivot, temp;
   float exeliene), eyeliene);
   Int 1;
double mint.comf;
                                                                                                                               pivot_row-pivot_columnes;
pivot=fabsia(s)[s];
for {1=1; i<=ncow; i==]
   for (j+1, j(-hpow; j++)
       i
axn[i]=x[i]=comt - y[i]=mint;
ayn[i] + x[i]=mint + y[i]=comt;
x[i] = axn[i];
y[i] = ayn[i];
                                                                                                                                           if (fabs(s[i][j]) > pivot)
                                                                                                                                                 pivo1_row=1;
pivo1_column=j;
pivo1+fabsta[i][j]];
      1
                                                                                                                                        $
    FUNCTION PROMO ...
AMASTASIADIS: "...FITS A SET OF NO FOINTS TO A POLYNOMIAL
CONTAINING MOV TERNS, THE MIDDLE POINT OF THE SET OF NO FOINTS
IS THEM REFLACED BY THE CORRESPONDING POINT FROM THE LEAST
SQUARE FIT, FOR NOMENCLATURE AND THEORY, REFER TO 'INTRODUCTORY
COMPUTER METHODS AND NUMERICAL ANALISIS' BY R.N. FENNINGTON,
PUBLISHED BY COLLIER-MACHILLAN LIMITED, LONDON"
if [pivot_roe > k]
                                                                                                                                      for (j+1; j(+noow1; j++)
....
temp = a(pivot_rpull);
a(pivot_roul())=a(k)());
a(k)())=temp;
....
     ADAPTED FROM FORTRAN FUNCTION "PUSHO" BY ANASTASIADIS
ň
                                                                                                                                         •
                                                                                        ______
                                                                                                                                   .
void pusmo(ns,npow.npowi)
int ns, npow, npowi;
                                                                                                                                 if (pivot_column > k)
                                                                                                                                       for (1+1; 1(+npow; 1++)
 void eliminate();
float f(50](4), b(4);
                                                                                                                                            temp+a[1][ptvot_column];
a[1][ptvot_column] + a[1][k];
a[1][k] + Temp;
  int 1m,ka, ja, ib;
  for (is=1; isCons; is++)
                                                                                                                                          .
       f[ia][1]+1,8;
far (1b=2; 1b<= npow; 1b++)
{
f[ia][1b] = f[ia][1b-1]=x[ia];
                                                                                                                                       itemp+id[pivat_column];
id[pivat_column] + id[H];
id[k]+itemp;
                                                                                                                                    3
           5
     Þ
  for (im=1; in<-npow; im++)
        for (ka+1; ka(+1a; ka++)
              a[ka](ia] = 0.0;
for (ja=1; ja<= ns; ja==)
a[ka](ia] = a[ka](ia]=f[ja][ia]=f[ja][ka];
a[ia](ka] = a[ka](ia];
           >
     $
  for (kas1; kaConpou; ka++)
        b(ka) = 0.0;
for (js=1; js <= ns; js==)
b[ka]=b[ka] = y[js]=f[js][ka];
  eliminate(npow_npow1,b);
           _____
 ~
 98
98
      FUNCTION eliminate SOLVES THE MATRIX EQUATION acts by USING
GAUSSIAN ELIMINATION AND USING THE LARGEST ELEMENT AS THE PIVOT.
REFER TO 'INTRODUCTORY COMPUTER METHODS AND RUMERICAL ANLAYSIS'
BY R.M. PENNINGTON.
 41
61
62
63
                       ______
 void eliminate(npow,npow1,b)
int npow,npow1;
float b[4];
    int i, k, j, kk;
void exchange();
  far (i=1; i<=npow; i++)
       a[i]Cnpow13=b[i];
id[1]=1;
    $
 for (k=1; k<-npow; k++)
       if (k to nooe) exchange(nooe, nooet, k);
if (a[k][k]==0)
             printf(" NO UNIQUE SOLUTION ");
exit(1);
          ۰.
        1k=k+1;
for (j=kk; j(=npow1; j↔)
              a[k][j]=a[k][j]/a[k][k];
for (i=1; i<=npow; i++)
                  Ĩł.
                    if (k = 1) al12[j]=al1][j]=al1][k]=alk][j];
          /sseas HEXT J sas/
/sseasa HEXT L sas/
    3
  for (1=1; i<-npow; i++)
       for (j=1; j<=npow; j++)
              if (id(j) == 1) c(i)=a(j)(npow1);
           >
                                                                                                           (4 columna)
```

(fast modified: Tue Sep 17 10:56 1991) at/smoother.c

20722079660365626266294309996664636696666666666666666666656555

SUBROUTINE rotate() ROTATES THE SET OF POINTS (X12, Y13) COUNTERCLOCEVISELY BY AN ANGLE and

TRANSLATED FROM THE FORTRAN SUBROUTINE "ROTATE" BY ANASTASIADIS.

. /10

...

...

(last modified; Tue Sep 17 10:36 1991) at\emoother.c

es FUNCTION suchange Exchanges Boys and Dolumns of The Matrix a such on that the findt Elerent is always the Largest Elerent in the Matrix es Again, REFER TO PENNINGTON.

A.21

(4 columna)

(fast modified: Thu Sep 19 17:02 1991) at/shaps.c	Bast modified; Thu Sep 19 17:02 1991) z:\shape.c
/**	PIINTIC INTHE TOTAL AURELE OF POINTS IN THE EXPERIMENTAL PROFILE IS Xd",
es Bhape.c es version 1 es programmed by Nicole Desarguette and Nandini Das Gupta es under the supervision of Prof. R.E.Eamai	ntati; printf"-number of points for fitting: "1; noaintatai(printf); nataonintata();
ee 3 august 1331 ee	np2+np1+1;
 PROGRAM SHADD- C MARES A SHAPE COMPARAISON BETWEEN THE EXPERIMENTAL POINTS AF SHOOTHING AND THE THEORETICAL POINTS FOUND BY SOLVING BASHFORTH AND ADAMS EQUATION BY THE FOURTH ONDER RUNGE-CUITA RETHOD. THE THEORETICAL PROFILE IS FIRST SOLVED FOR A VALUE OF B APPROXIMATED CITHER BY THE CHPIEICAL FORTLA OF HEM AND REED OR BY THE FIRST VALUE ENTERED BY TH USAR, A ROBUST SHAPE COMPARISON BETWEEN THE EXPERIMENTAL AND THE THEORETICAL. 	/+
PEOFILES IS THEN DONG. THREE EINOS OF FREDES ARE EVALUATED. THEN, THE VALUE of DF BIS INCREMENTED AND THE COMPARISON IS PERFORMED AGAIN. THEN FOR EVERY EI of ENRORS THE OPTIMAL B IS FOLMO.	/storestering and the for Example of the formation of the
extern unsigned_Balen + 40000; #Include (graphics.h>	}yossysino21 - de; for (1=1;{ (1<=1=klisklisk[1](yds) }; 1==); µds1=ax[1]=(yds=ax[1])/(ay[1=1]=ay[1])=(ax[1=1]=ax[1]);
Anclude (float,h) Anclude (atdib,h) Anclude (atdib,h) Anclude (atdib,h) Anclude (atdib,h) Anclude (anth,h)	<pre>subsidex(1); for (1)=ext; (((<=ntOT) &L (ay(1)>yds)); 1==); xdsr=ax(1)=(yds=ay(1))/(ay(1=1)=ay(1))=(ax(1=1)=ax(1)); xdsr=ax(1)=(yds=ay(1))/(ay(1=1)=ay(1))=(ax(1=1)=ax(1)); xdsr=ax(1=1); asstroid===xds1; asstroid===xds1;</pre>
FILE soutput_file, eresult; int text,text,lex; float s,b,c,d; float sr(1802,sr(1902,tense(1802);/sANRATS CONTAINING THE DIFFERENT EINO OF EREO float sr(1802,sr(1902,tense(1802);/sANRATS CONTAINING THE DIFFERENT EINO OF EREO float sr(1802,sr(1902,tense(1802);/sANRATS CONTAINING THE DIFFERENT EINO OF EREO float sr(1802,sr(1902,tense(1802);/sANRATS CONTAINING THE DIFFERENT EINO OF THE float sr(1802,sr(1902);/s(1802);/sANRATS CONTAINING THE DIFFERENT EINO OF THE float sr(1802,sr(1902);/s(1802);/s(bola-spit -6.72385-(15.3882 +(-16.4473+(9.52425-2.58583#)+#)+#3+##3 bola-spit (bola); /************************************
float wt[2000],yt[2000]; /wARRAYS CONTAINING THE POINTS OF THE THEORETS	printf("\nbeta-%a", beta): if ((beta-24,6) [] (beta-(%, 125))
main[int argc, char ==argv}	printy["\nCF HUH AND REED IS NOT VALID");
float symmetry(); vold menu(); vold extrame(); vold extrame(); vold extrame(); vold ext(); vold ext();	Deta_starteptad.ut; Deta_edubetad.B1; step=C.B085;
void pr(); void coord(); FILE esmonthed.sprofile; Char string(20).ch; unsigned short end.end1.end2.check; int number_of_profiles; int number_of_profiles; int number_of_profiles; int number_of_profiles;	<pre>/************************************</pre>
EXPERIMENTAL PROFILES/ /sequent:NUMBER OF POINTS FO COMPARISON, UP TO 1086/	<pre>istep=6; ys[]=ay(istep]; xs[]=au(istep];</pre>
int melect,promrite.j; int mulp.nip.nol.no2.ii.m.nnl.istep.jm.nn2.nn3.nn4;/anulp. nip ARE DEFINED IN T float deyde.s.ds.wdsl.wdsr.step; float deyde.s.ds.wdsl.wdsr.step; float deyde.s.ds.wdsl.wdsr.step; dslDAFETER AT de FEON THE chast ext of The Concern The	for (1=2; 1<=nn1; 1==) { 1stop=&=(nutp=&l=(1=1)/nn; ys(1)==y(1stop); ==tillast(stop); ==tillast(stop);
float beta, beta_start, beta_end; float beta, beta_start, beta_end; float bt(set); float xs(100]; float xs(10); float x) nn2=nn1+1; nn3=(nosint=5)/2; far (1=nn2; 1<=nn3; 1==)
Int bets in s.j.jj.jj.k.mod2.msta.ns; float ay[2000] ax[2000]; float center; float center; float center; ASICE of The AXIS of STM ASICE of The AXIS of STM	<pre>istop+(n1p-nulp)/de(1-nn1) + nulp; ys(1)=ey(istop); xs(1)=ex(istop); }</pre>
float controld; /controld/iNIDUE IN THE dee short j1,j2,icon; float slope,sp[180]; /sAREAY OF THE SLOPES OF THE POINTS VITH THE APEXs/	nni=nn]=1; for (==nni; 1<=npoint; 1==) { istepanip=(ntot=5=nip)/nne(1=nn]}; ys[1]=ay(sitep); xs[1]=ax(sitep];
/endered the store of the store	/material states and a state state and a state s
<pre>Printf("EREOR OPENING FILE CONTAINING PROFILE"); exit(1);</pre>	/
} 1f {coutput_file=fopen[argv[2],"ut"})==HULL) { printff=ERKOR OFENING CUTPUT FILE="};	ee HEEE THE SLOPE OF EACH POINT RELATIVE TO THE APEX IS CALCULATED AND em STORED IN THE ARRAY TOPIL encourselessestementeressestementeressestementeressestementeressestementeres for (j=1; j <enpoint; j++)<="" td=""></enpoint;>
} } If {(result-fopen(argv[3],"rt"))++KLL}	if (xs[j])center) sp[j]=(ys[np2]-ys[j])/(xs[j]-center); else
<pre>printf("ERROR OPENING RESULT FILE"); exit(1); }</pre>	<pre>mp[j]+(ys[np2)-ys[j])/(conter-ms[j]); }</pre>
fscanf(result,"%d %d",&nutp,&nlp); restnd(smoothed);	Associated and the same of b and the step size
pointes; endes; while (lend) {	<pre>/ etseastereseaste transformationseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseasteres transformationseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseasteres transformationseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseasteres transformationseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseasteres transformationseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseasteres transformationseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseasteres transformationseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseasteres transformationseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseasteres transformationseastereseastereseastereseastereseastereseasteres transformationestereseastereseastereseastereseastereseastereseastereseastereseastereseastereseastereseasteresea transformation</pre>
<pre>rr v=scar(mmourmo, me xer,say(point),sax(point))+=(DF) end=1; point++; } ntotepoint=2;</pre>	<pre>print(T_VS(AKIAW 0: -); beta_startatof(potistring)); print(TEMDING 8: -); beta_endestof(potistring)); print(TSTP 51Z: -); stmp=atof(gets(string));</pre>
ANTICIDATION EXTERNO FINOS THE EXTERNE FOINTS OF THE PROFILE.	bt[1]=bets_start;
<pre>v run, 10m symmetry ranks inc control to the second s</pre>	Another state and the state of
ee THE USER INFUTS THE MURBER OF POINTS FOR THE FITTING (8 columns)	
	A.22

```
*****
                                                                                                                                                                                                                                                         START THE COLDEN SECTION
                                                                                                                                                                                                                                                                   tcount=interpolatelas.sy.nulp.nip.ntot);
cright=d.Sise(pright=pleft) = pleft;
cleft=d.Sise(pright=pleft) = pleft;
eleft=ercal(cleft,icount);
eright=ercal(cleft,icount);
for (1=1; (i1<=50) & ((pright=pleft))=tissit), i++)</pre>
          NOW THE POINTS ON THE THEORETICAL PHOFILE THAT ARE TO BE COMPARED TO THE
CORRESPONDING POINTS ON THE EXTERIENTAL PROFILE ARE LOCATED. THE
THEORETICAL PROFILE IS STORED IN THE AREATS XT[] AND YT[] AND THE CHOOSEN
POINTS ARE IN XIS] AND YT[].
...
...
 ••
            nsta=2;
mod2=0;
                                                                                                                                                                                                                                                                      if (erightceleft)
             mod2+0;
for (العادي: العادي) (=no1; (ا++)
                                                                                                                                                                                                                                                                                pleft=cleft;
cleft=cright;
eleft=eright;
cright=bila=(pright=pleft)=pleft;
eright=erical(cright,icount);
                       j1=np2-jj;
j2=np2+jj;
iCOn=8;
                       iui---;
ijj=1;
while ( (jjj=+1) ); (mod2x2) )
                                                                                                                                                                                                                                                                          )
1f (eright--eleft)
                                    end1=end2=0;
for (k=nata; ( (k<=1000) && (lend1) && (tend2)); k++)
<
                                                                                                                                                                                                                                                                                   pright*center+8.86;
plaft*center+8.85;
cright=0.618*(pright*plaft)*plaft;
claft=8.332*(pright*plaft)*plaft;
elaft*errcal(claft,icount);
eright*errcal(cright,icount);
                                                 mlope=yt[x]/xt[k];
if (1con(8)
                                                       {
    1f (slope>-sp[j2]) end1=1;
                                                else if (icon)0}
                                                                                                                                                                                                                                                                          st
                                                                                                                                                                                                                                                                                  ٠
                                                                                                                                                                                                                                                                                ٠
                                                                                                                                                                                                                                                                                   pright=cright;
cright=cleft;
eright=eleft;
cleft=0.323=ipright=pleft1+pleft;
eleft=ercal(cleft,icount);
                                                             if (slope >+sp[j1]) end2+1;
                                                 else if (icones0)
                                                             if (slope>sp[j1]) end2=1;
else
                                                                                                                                                                                                                                                                 > /same NEXT I some /
                                                                   4
                                                                        if (slope)-sp[j2]) end1+1;
                                                                                                                                                                                                                                                            center=(pright=pleft)/2.8;
return(center);
                                               > /ess %conset ss/
} /sessfor k sess/
                                                                                                                                                                                                                                                           1f (end2)
                                                       xts[j1]=xt[k];
yts[j1]=yt[k];
icon=~1;
                                            else if (end))
                                                       xts[j2]=-xt[k];
yts[j2]=yt[k];
icon=1;
                                                 3
                                                                                                                                                                                                                                                            void permoti;
float xx(20), yy(20);
float ext, exr,mp;
int 1,j,jjn,jn,jj,jj1,jn,11;
                                           mod2++;
nsta-k;
                                           JJJ**;
/ese END WHILE ====/
/ese NEXT JJ ====/
                                    }
                      ъ
                                                                                                                                                                                                                                                             xts[np2]=vts[np2]=0.0:
                                                                                                                                                                                                                                                            extems[1];
for (1=2; iC+nulp; 1++)
 A THE VALUE OF 5 IS WRITTEN TO THE OUTPUT FILE. THE FUNCTION IS

OR CALLARTES THE ROBUST FIT
                                                                                                                                                                                                                                                                   ł
                                                                                                                                                                                                                                                                        If (axtild=ext)
            rs(npoint,xts,yts,xs); /sTNE ERRORS/
pr(npoint,i,xs,ys); /sTNE ERRORS/
  ...
                                                                                                                                                                                                                                                                                    exteax[1];
                                                                                                                                                                                                                                                                                    text=1;
                                                                                                                                                                                                                                                                              ¥
               htfiellebtlil + step;
                                                                                                                                                                                                                                                                  3
                                                                                                                                                                                                                                                            /second and a second and a second and a second a
                                                                                                                                                                                                                                                                    ۲
                                                                                                                                                                                                                                                                         if (ax[1]>+exr)
                                                                                                                                                                                                                                                                               4
                                                                                                                                                                                                                                                                             uxr+axt
lexr+1;
}
                                                                                                                                                                                                                                                                                     exreax[1];
   /assessmentersEND OF MAIN PROGRAM _ sessessessess/
                                                                                                                                                                                                                                                               FUNCTION symmetry() "LOCATES THE AXIS OF SYMPETRY OF A DROP PROFILE
ASSUMING THE PROFILE IS VERTICAL, BY THE METHOD OF COLDEN SECTIONS.
pright, pieft - POSITION OF RIGHT AND LEFT MARGIN RESPECTIVELY
cright, cleft - POSITION OF RIGHT OR LEFT GOLDEN SECTION
eright, eleft - ERROR AT cright OR cleft." -AMASTASIADIS
                                                                                                                                                                                                                                                                        THE APEX IS FOUND
                                                                                                                                                                                                                                                              ap-ay[nulp];
far (1=nulp; 1C+nlp; 1++)
  ۲
                                                                                                                                                                                                                                                                          1f_(ay[1])+ap)
            TRANSLATED FROM FORTRAW FROM SUBROUTINE "SYMPETRY" BY AMASTASIADIS.
C VERSION RETURNS A FLOAT VALLE, CONTON, WICH IS THE CENTROID OF
THE PROFILE. THE CONTENTS IN THIS FUNCTION ARE ALSO TAKEN FROM
THE AMASTASIADIS SUBROUTINE.
                                                                                                                                                                                                                                                                                    ap+ay[1];
lex=1;
                                                                                                                                                                                                                                                                               $
                                                                                                                                                                                                                                                                  ¥
  float symmetry(ax,ay,nulp,nip,nic)
float ax(1000),ay(1000);
                                                                                                                                                                                                                                                            5
                                                                                                                                                                                                                                                            /second second s
  int interpolate();
float errcal();
  int i,icount;
float pright,pleft,tlimit,cright,cleft,eright,eleft;
float center;
                                                                                                                                                                                                                                                              (
lat 1;
float err,errof;
                               error=0,0;
for (1=1; 1<=1count; 1==)
           "LOCATE THE APPROXIMATE POSITION OF THE AXIS OF SYMETRY AND DEFINE THE
REGION OVER WHICH COLDEN SECTION IS APPLIED"
```

(last modified: Thu Sep 19 17:02 1991) at/shape.c

(fast modified: Thu Sep 19 17;02 1991) a:\shape.c

Sast modified; Thu Sep 19 17:02 1991) at/shape.c (last modified; Thu Sep 19 17:02 1991) at\shape.c 58-500-505 50/2012 C+(501-505 50/2012) (c)(502-505 50/2012) (c)(502-505 50/2012) (c)(502-502 50/2012) (c) errezi[1]exr[1]-2.0ecenter: returnlerror/11.Belcountil: ANTICAL AND A CIVES THE ROBUST FIT BETWEEN TWO SHAPES. AN FUNCTION COOLD FUTS TRANSFORMED COORDINATES INTO U.V FOR SHAPE 2 void rain, sta, yta, sa, ya) void coord(n,uu,vv)
int n;
float uu[1888],vv[1888]; ficat_sts[100],yts[100],xs[100],ys[108]; float fund(); wold (st); wold coord(); int 1; for [1=1; 1<=n; 1==] float thete, the, CT, dT, aT, bT, CTT; float thete, the, CT, dT, aT, bT, CTT; float xT, yT, u1, v1, ut[108], vT[106]; 4
u(1)=a=Czuu(1)=Cdevv(1);
v(1)=D=Cdeuu(1)=Cdevv(1);
/=escos printf("1=Xd v=Xe u=Xe",1,u(1),v(1));escosses/ la(n,xta,yta,xa,ya); ÷ /toronactorona nactoronactoronactoronactoronactoronactoronactoronactoronactoronactoronactoronactoronactoronactoronactoronactor for (1=1; 1<=n; 1==) float fmed(n,a) float a[1088]; int n; j1+0; for (j+1; j(+n; j++) • if m-ji t int m,1,k,11,j; float at; j1++; ut[j1]=sqrt((pow(#s[j]=xs[1),2)+pow(ys[j]=ys[1],2))/ f pow(u(j]=u[1],2)+pow(v(j]=v(1],2)); man/2+1; for (1+1; 1(+m; 1++) ۲ > 8+1; 11+1+1; > vt[i]=fmed(n-1,ut); rho-fmed(n,vt); /4014-10141104 THETA } if (n+2=(n/2)) return((a[n]+a[n-1])/2); else -- GET ROTATION THE' for (1-1; 1<-n; 1++) return(a(ml); j1+8; for (j=1; j<=n; j++) { L 11 (11-)) A FUNCTION INTERPOLATE INTERPOLATES THE LEFT SIDE OF A DROP PROFILE 4 ji++; xi=xs[j]=xs[+]; yi=ys[j]=xs[+]; ui=u[j]=u[1]; vi=u[j]=u[1]; ut[j]]=atan2(ui=y1=v1=x1,u1=x1=v1=y1); int interpolate(x1,y1,nwlp,nlp,ntot)
int nulp,nlp,ntot;
float x1[1884],y1[1860]; int ists, tend, 1, thigh, ilow, 1, icount: > ista=nulp=30; iend=30; i=ista; icount=1; ihigh=hlp; while (1>=iend) vt[i]=fmed(n-1.ut); 5 theta-fmed(n,vt); ct-rhoscos(thets) dt+rhossin(theta); for (1+1; 1<=n; 1++) ut[i]+xs[1]-ct+u[1]+dt+v[1]; vt[1]+ys[1]-dt+u[1]+ct+v[1]; printf("DIVISION BY # IN INTERPOLATE"); #x11(1); atefmed(n.ut): bt=fmed(n,ut);
at=at=ct=a=dt=b;
b=bt=dt=a=ct=b;
a=at;
a=at; \$ ctt-ct+c-dt+d: icountdectededtec; return()count): coord(n,xts,yts); rho-sqrt(c=c+ded);
theta=atan2(d,c); /second second s /bissessing espectangessingsessingsessingsessingsessingsessingsessingsessingsessingsessingsessingsessingsessingsessingsessing output the set of void lain,xta,yta,xa,ya) int n; float xta[100],yts[100],xa[100],ys[100]; {
 int n1,i;
 float p1=3,1415;
 float x1(1000]_z2(1000]_z3(1000]_bb_hh,k11,k12,k13,k14; /=z1=PHI, z2=X, z3=Z=/
 float x21,k22,k23,k24,k31,k32,k33,k34; float an, sx, sy, su, sv, sux, suy, suu, svx, svy, svv, ss; int 1: void coord(); 21(1)=22(1)=23(1)=0.0; Ex-By-SU-SV-SUX+SUY-SUU-S-X+SVy+SVV+8,0; for {1=1; 1<=n; 1++} hheh/2.8; bb=2.8/b; n1=n-1; for (i=1; i<=n1; i==) 4 Burekvos(1); ByrByrys(1); BursWyrS(1); BursWyrS(1); BursWyrYfs(1); BursBuryrTs(1)eys(1); BursBuryrTs(1)eys(1); BursBuryrfs(1)eys(1); ByrsWyryrTs(1)eys(1); BwrsWyryrts(1)eys(1); BR-BR-MB(1); if (i=1) k1=1.8/b; if (i=1) k1=bbc3i(1=sin(-z1(1)+2spi)/z2(1); k2=cos(z1(1)+p1); k3=sin(z1(1)+p1); k12=bbc3i(1)+bsk31=sin(-z1(1)+2spi+bsk11)/(z2(1)+bsk21); k22=cos(z1(1)+p1+bsk11); k12=sin(z1(1)+p1+bsk11); k13=bbc3i(1)+bsk32=sin(-z1(1)+2spi+bsk12)/(z2(1)+bsk22);

(last modified: Thu Sep 19 17:02 1991) at/shape.c (fast modified: Thu Sep 19 17:02 1991) at shape c k23=ccs(z1(1)=p1+hek12); k33=sin(z1(1)=p1+hek12); k14=bb+z3(1)=hek33=sin(=z1(1)=2=p1+hek13)/(z2(1)=hek23); k24=css(z1(1)=p1+hek13); z1(1=1)=z1(1)=h/6.se(k1=2,sk12=2,sk13=k13=k14); z2(1=1)=z2(1)=h/6.se(k1=2,sk12=2,sk13=k13=k14); z3(1=1)=z2(1)=h/6.se(k1=2,sk12=2,sk13=k13=k14); break; Case 2: index-sinisi26(number_of_profiles,srr); break; Case 3: index-sinisi26(number_of_profiles,tense);) printfi"\n PLEASE ENTER THE DIFFERENCE IN DENSITIES: "); delta_rho=ato(igetaistringi); Tenaton=gedeita_rho=(upex=centroid)=(apex=centroid)/(bt[index]=bt[index]); print("\nOPTIMAL VALUE Of b: Xa",bf(index)); print("\nIMCX.g: Xd",index); print("\nIMCX.g: Xd",index); print("\nIMCX.g: gradual; print("\nimcConclusion of AFEX: Xa",center); print("\nimcTonclusion; Xa",anama; print("\nimcTonclusion; Xa",anama; print("\nimcTonclusion; Xa",anama; print("\nimcTonclusion; Xa",anama; print("\nimcTonclusion; Xa",anama; print("\nimcTonclusion; Xa", Xa", Xa", print("\nimcTonclusion; Xa", Xa", Xa", print("\nimcTonclusion; Xa", Xa", Xa", Xa", entereded; inf (lich-getChe(j)=="q"; j] (che="q")) entered=1; } for (1=1; 1<-n; 1++) xt[1]=22[1]; yt[1]=23[1]; 3 vold pr(n, index, xa, ya) int n,index;
float xs[108],ys[100]; int i: float factor,sx,sox,sy,syy,rx,ry,r,rx1,ry1; int 1, index; float low; sr[index]=srr[index]=tense[index]=sx=sy=sxx=sy=0,8; far (i=1; i<=n; i=>) lowsa[1]; index=1; for (1=2; iC+number_of_profiles; 1++) if (a[i](low) Ł Bx=Bx=u[1]+um[1]; Bx=Bx=u[1]+um[1]; By=By=v[1]+ym[1]; By=By=v[1]+ym[1]; Tx=u[1]+um[1]; Ty=v[1]+um[1]; Ty=v[1]+um[1]; Ty=v[1]+um[1]; Br[Index]=Br[Index]+r; louna[1]; returnLindex); rx1=u[i=1]=xx5[i=1]; ry1=v[i=1]=ys[i=1]; Tense[index1=tense[index1=pow(rx=rx1,2) =pow(ry=ry1,2); \$ > factor=sqrt(sxx-pow(sx,2)/(2.0en) + syy-pow(sy,2)/(2.0en)); tense[index]=(sqrt(tense[index]/(1.0e(n-1)))/factor; srfindex]=(srfindex]/(1.0en)/factor; srrfindex]=(sqrt(srfindex]/(1.0en))/factor; f FLE efite_name; char string[20]; unsigned short finished,entered; int i,choice; > finished=8; white (ffinished) /enumeration DISPLAYS HAIN FENU OFICOUS enumerations and the set of the set clrscrij; printf("\nCAN WRITE THE FOLLOWING FILES;"); printf("\nA1 SR FUNCTION"); printf("\nA SRR FUNCTION"); printf("\nA THESE FUNCTION"); printf("\n\n HNTER & TO BETURN TO MAIN MEML"); printf("\n\n\n ENTER & TO BETURN TO MAIN MEML"); finished=1; if ((choice=atoi(gets(string)))==4) finished=1; if (ifinished) { void optimal(); void file_write(); char string[28], ch; int entered,choice,choosing; ŧ printf("\n\nPLEASE ENTER NAME OF THE FILE TO BE WRITTEN: "); enterod=0; while (ienterod) choosing=1; while (choosing) ł cirect). if ((file_name=fopen(getsistring),"wt")) == HLL)
 print("\n_cannot_open_file, please reenter: "); clrscr(); print("\nsesse MENJ OFTIONS esses\n"); print("\nsesse MENJ OFTIONS esses\n"); print("\n2 WRITE ERECK FILES"); print("\n3 QUIT PROGRAT'); print("\n1\n\n OFTION KUMBER; "); enterde-choica=0; while (lentered) if ((choice=atoi(gets(string))) >0) && (choice(4)) entered=1; gmttch (choice) } else enteredel; switch (choice) ł case 3: for ()=1; 1(=number_@f_profiles; 1++)
 fprintf(file_name,"Xe Xe\n",Tenme()],bf()]); fprint(file default: finished=1; }/essentChea/ }/see white not finishedees/ }/see white not finished eas/ /see function file_write esse/ > • \$ /nen /according to the second secon {
int minimize();
int entered, choice, index;
float delta_tho, tension;
char_string[28]_ch;
float g=388.665; cirecr(). CITECT(); print("DOFTINIZATION OF B BY MINIMIZATION OF WHICH OF THE FOLLOVING :"); print("\n 1 SE"); print("\n 3 TENET); enteremodels; enteremodels; while (tentered) printf("\n\n\n WICK TYPE OF [#204? "); if ((choice=atoi(gets(string))))) && (choice=(4)) entered=1; a columnal mitch (choice) 1 case 1: index-minimize(number_of_profiles,ar); A.25 (8 columna)

Bast modified; Thu Sep 19 17:03 1991) at/(theoret.c	fast modified: Thu Sep 19 17:03 1991) at/theoret,	
/**************************************	******	rp2=rp1+1;	
en Theoretical.c			
es version 1 es programmed by Nicola Demarquette		/annenuenenenenenenenenenenenenenenenenen	
es under the supervision of Frof, N.R. Kanaj as 12 August 1931		**************************************	
BR BR - BROCEAN Sharesting MARTS THE SULEY COMPLETION FOR A FILEW & THEFT BU		xslnp21+center;	
* USER AND DISPLAY THE EXPERIMENTAL AND THEORETICAL CURVES TOGETHER.	1111		
* ## * ################################			
extern unsigned, sthien +45588		IN THE POINTS FOR THE FITTING ARE CHOSEN HERE AND STORED IN THE ARRAY	
finalude farables b			
#include (float.h)		nntennet;	
dinclude (stdl)h> dinclude (stdl)h>		1atap=6;	
#include <conto.h> #include <aeth.h></aeth.h></conto.h>		ys[1]=ay[1stop]; zs[1]=ay[1stop];	
fif noutout file presult:		for (1=2; 1<=nn1; 1++)	
Int ieur, ieul, ieu; /#INDEX OF THE HOST EIGHT EXTREME, HOST	un,	istep-6+(nulp=6)=(1-1)/nn;	
floet wr(500)_wi(Se0); /www.hat.cs USED IN symmetrye/		ymilianyitatoo; xmilianxitatop;	
float sr[100],err[100],tonse[100];/sARRAYS CONTAINING THE DIFFERENT EIND (float u[100]],v[1001]; /sUSED IN coord TO RETRIEVE THE POINTS (OF E210) nn2=nn1+1;	
FLOAT ATTINGT ATTINGT ATTINGT ATTINGT ATTING THE POINTS OF THE TO	KORETI	na]=(npoint+5)/2; for (1=nn2: 1<=na]: 1++)	
		(internation-with)//afi-mails a suine	
		ys[1]-ay[1step];	
main(int argc, char eeargv) {		xmt1]+axt1mtep1; }	
ficat symmetry(); vold extreme();		no4-nn3+1;	
void theoretical_profile(); /eTHE FUNCTIONS ARE DESCRIBED WHEN DEFI	NCD+/	far (1+nn4; 1<+npaint; 1++)	
vold pr();		istep-nlp+(ntot-5-nlp)/nne(1-nn3);	
vold coord(); FILE ===monthed.=profile;		ys[1]=ay[1stop]; xs[1]=ax[1stop];	
cher string[20], ch; unsigned short and codt and check;			
Int point, ntot, apoint; /entot: TOTAL MUBER OF	F POINT		
JonpaintinGREE OF PO	INTS FO	AN THE POINTS CHOSEN FOR FITTING ARE WRITTEN TO THE OUTPUT FILE	
Int ij.test.ijj:	•••/	for (1=1;1(=npo)nt;1++)	
int mulp.nlp.nlp.np1,np2,it.nn.nn1,istep.jn.nn2,nn3,nn6;/anulp, nlp ARE DEFIN float vtn[188] vtn[188]	ED IN T En theo	fprintf(output_file,"%e %e\n",ym[i],xm[i]);	
FOR SHAPE COMPARISON	Hi /		
float xs[108], ys[108]; /sARRAYS OF THE CHOOSE	EN EXPE	HERE THE SLOPE OF EACH POINT RELATIVE TO THE APEX IS CALCULATED AND	
int beta_in,i,j,jj,jj,k,mod2.nsta.nu;	m/	44 - SIVELD IN THE ANALY SPLI.	
ficat av[1001],ax[100]); /nARRAYS OF THE EXTER: ficat center: /ncenter:#ETURNED_BY	INENTAL FUNCTIO	for (j=1; j<=npoint; j==)	
ABSICE OF THE AXIS I	OF SYNEL	if (ms(j))center)	
figet centroid; /acentroid; /acentroid; /acentroid; /acentroid;	THE day	else	
inhort j1,j2,100n; float slope,mp[100]; /sARRAY OF THE SLOPES	OF THE	sp[j]+(ys[mp2]-ys[j])/(center=xs[j]); }	
Int devicesDETECT mode	X4/ -		
float w: float auftably auftably.			
Int nick;			
/======================================		printt("\ntwick B FLCASCI "); beta=atof(gets(string));	
I SHOULD FURTHE SHOUTHED AND OUTPUT FILES AND READS THE X AND Y COORD I SHOUTHED FOINTS INTO THE BY AND BE ARRAYS RESPECTIVELY, NOTE THAT I	EACH	if((profile=fopen(argv[3),"et"))==NRL)	
- LINE CONTAINS THE Y FOLLOWED BY THE X OF ONE POINT.		print("ERROR OPENING OUTPUT FILE");	
cirscr():			
getatatring;		/>>>===	
if((string(0)++'N'))) exit(0);		e THIS & IS THE STEP SIZE USED FOR SOLVING THE LAPLACE EQUATIONS BY an ATH ORDER RINGA KUTTA METHOD. BEFORE THE THEORETICAL PROFILE IS	
1f ((smoothed+fopen(arg=[t],"rt"))+=HLL)		. CALCULATED BY THE FUNCTION theoretical profile THE VARIABLES & AND a	
PRINTE ("ERROR OPENING FILE CONTAINING PROFILE");			
•••••••••••••••••••••••••••••••••••••••		1 = 14.3100014-0.403//1000.0; theoretical_profile(1000.0;	
if (Coutput_file=fopen(argv[2],"et"))==HLL)			
printf("EREOR OFENING OUTPUT FILE"); avit();			
)) ··································		- CORRESPONDING POINTS ON THE EXPERIMENTAL PROFILE ARE LOCATED. THE	
ramingianootneg; nulpenipe8;		I DE THEURETICAL PROFILE IS STORED IN THE ARRAYS XTEL AND YEEL AND THE CHOOSEN	
if ((result=fopen(argv(4),"rt"))==KEL)		**************************************	
printf("ERROR OFENING RESULT FILE");		mpd2e0; for filel, lifenst, lies)	
THEANTERNET, "AD AD", ANUID, ANUD:		j1*npZ-jj; j2*np2*jj;	
pointel; endel;		iconet;	
while (lend)		unite ((jjj==1) 11 (mod222))	
if (fscanf(smoothed,"%e %e", Say[point], Sax[point])++{DF}		endt send2+0;	
point++;		TOP (Remotal) ((RC01000) 66 (10001) 65 (10002)); R++)	
ntat-point-2;		sloperytikj/xt[k]; if (lcon(8)	
		if (sippe)=un[i2]] end(=1:	
/bitesterstersterstersterstersterstersterst		alas if firmha)	
an FUNCTION symmetry FINDS THE CENTROID.			
} erespectivesessessessessessessessessessessessesse	17 (Slope >=mp[j1]) end2=1;		
contor-symmetry(ax,ay,nulp,nlp,ntot);		else 1f (icon==0)	
A		if (stope>up(j1)) end2=1;	
as THE USER INFUTS THE MERER OF POINTS FOR THE FITTING			
POINTS ("INTHE TOTAL RUBBLE OF POINTS IN THE EXPERIMENTAL PROFILE IS MAT	148/	if (stope>=sp(j2)) end(=t; }	
ATOTI: artard(m).efferte menere of anture for stirring. mi-		/ /ss icons=8 se/	
epoint-stoi(gets(string));			
npt=npaint/2:			
Gast modified: Thu Sep 19 17:03 1091) a:\theoret.c

xts[j1]=xt[k]; yts[j1]=yt[k]; icon==1; wise if (end)) if (eright==eleft) xts[j2]=-xt[k]; yts[j2]=yt[k]; icon=1; pright-Conter+8.46; plaft-Conter+8.45; cright=8.515+[pright-plaft]+plaft; claft=8.382+(pright-plaft)+plaft; alaft=#frcal(claft_icount); \$ mod2++; nsta+k; eright+errcal(cright_icount); JJJ+: /see EHD WHILE =see/ /see HEXT JJ = see/ if_(eright>eleft) ÷ pright+cright; cright+cleft; xts[np2]=yts[np2]=8.8; eright=eleft; cleft=0.382=(pright=pleft)=pleft; eleft=errcal(cleft,icount); THE VALUE OF & IS WRITTEN TO THE OUTPUT FILE. THE FUNCTION FR CALCULATES THE ROOUST FIT > /sess HEXT I sess/ conters(pright=pleft)/2.8; return(conter); } rainpoint, its, yts, xa, ys); /aTte ROBIST SWATE CONFALISON primpoint, its, ys); /aTte ROBIST SWATE CONFALISON coord(1880, xt, yt); /aTte THEORETICAL PROFILE confid:1880, xt, yt); /aTte THEORETICAL PROFILE rs(npoint,xts,yts,xs,ys);
pr(npoint,1,xs,ys);
coord(1898,xt,yt);
nicks8;
for (1j=1; 1j<=1808; 1j++)</pre> / THE CIDM EXTREME FINDS THE EXTERT POINTS OF THE PROFILE. •• INF AND INNI ARE THE FOINT RUPERS OF THE EXTERT EXTERT AND LEFT •• POINTS, ESPECTIVELY, WHILE ON AND ONL ARE THE EXTERT VALUES (X) •• COORDINATE, (** IS THE FOINT RUPERS OF THE APEX, WHILE OP IS THE •• VALUE (Y COORDINATE) ł. fprintf(profile,"%e %e\n",v[ij],u[ij]); n1c#++ ax[1j]+u[ij]: ny[ij]=v[ij]: void extreme(ax,ay,nutp,ntp,ntot) int nutp,nip,ntot; float ax[1080],ay[1080]; /MERODALISATION CONTRACTOR AND THE EXPERIMENTAL PROFILE IS DISPLATED THE GRAPHICS ARE INITIALIZED AND THE EXPERIMENTAL PROFILE IS DISPLATED IN GREEN WHILE THE THEORETICAL PROFILE IS DISPLATED IN MACENTA. Additional and the second and the secon (void pummo(); float xx(20), yy(20); float ext, ext,ep; int 1.j.jjn.jn.jj.jjt.jnn.i1; printf("error: graphresult=%d",test);
fclose(smoothed);
fclose(profile); ANTER LEFT EXTREME POINT IS FOLMO exit(1); } cleardevice(); printf("EXPERIMENTAL DROP IN GREEN"); printf("INTHEORETICAL DROP IN MAGENTA"); rewind(monthed); rewind(profile); ext+ax[1];
for {1+2; 1(+nulp; 1++) ۲. if (axtilc+axt) exteast13; 1exte1; for(ijj=1;ijj<=ntot;ijj++) putpixel(ax[ijj],ay[ijj],GREEN);
} for(ij=1; ij(=ntck;ij++) THE RIGHT EXTREME POINT IS FOLDO putpixel(nxtij],nytij],MAGENTA); ---exreax[nlp]; for (1=nlp; 1(=ntot; 1==) delay(5088); fcloseall(); fclosegraph(); ۲ 1f (as[1]>+***) ¢ exress(1): 1027=1; \$ Assessment of a second THE APEX IS FOUND -----44 84 84 84 84 86 p=ay[nulp]; or (1+nulp; 1<=nlp; 1+=) for (1f_tay[1]>+#01 TRANSLATED FROM FORTRAM FROM SUBROUTINE "SYMPETRY" BY AMASTASIADIS. C VERSION RETURNS A FLOAT VALUE, CONTON, WICH IS THE CENTROID OF THE PROFILE. THE COMPENTS IN THIS FUNCTION ARE ALSO TAKEN FROM THE AMASTASIADIS SUBROUTINE. DESCRIPTIONS at symmetry(ax,ay,mulp,nip,nic) wild all a stat. ap+ay[1]; lex=1; 48 49 49 49 float symmetry(ax,ay,nulp int nulp,nlp,ntot; float ax(1000),ay(1000); 3 /setential of the setential of the setential of the setential of the set of t int interpolate(); float errcal(); int i.icount: float center; int 1; float err,errof; error+8.8; for (1=1; 1<=1count; 1++) "LOCATE THE APPROXIMATE POSITION OF THE AXIS OF SYNETRY AND DEFINE THE REGION OVER WICH COLDEN SECTION IS APPLIED" err=xl[1]+xr[1]-2.%=center; error=error=err; center=(ax[iexl]+ax[iexr])/2.0; return(error/(1.0e)count)); pright=center+8.85; pleft=center=8.85; Tlimit=8.8881; AN FUNCTIN IN GIVES THE ROBUST FIT BETVEEN THO SHAPES. START THE COLDEN SECTION void ra(n,xts,yts,xs,ys) int n; float xts[100]_yts[100],xs[100],ys[100]; { float fmed(); void LB(); void coord(); lnt i,j1,j; if (erightCaleft)

(fast modified: Thu Sep 19 17:03 1991) al/theoretic

¢

1f (end2)





(7 celumna)



/**************	***************************************
din din	play.c
ee pro	gramed by Hicole Denarquette and Handow Compila
ee 25	ur the supervision of from the Kenel July 1931
44	
en COLORS AT THE S	C EMABLES THE USER TO DISPLAY 18 CURVES IN 18 DIFFERENT AME TIME. ************************************
<pre>#include (graphics. #include (stdie,h)</pre>	h>
Finclude (conto.h) Finclude (atditb.h) Finclude (asth.h)	
amin() (
FILE samothed;	ber of colete 1
ther string[10]18	0), mtr1(2);
1nt 1y[3000] 1x[3	ase), test, number, colour [18];
. THIS BLOCK INIT	TALIZES THE GRAPHICS TO DISPLAY THE PROFILES.
initgraph(Ldevice,L if ((test-graphres)	ande,NULL); 17(3) /= 8)
printf("SORRY, C printf("GRAPHES exit(1);	XM'T INITIALIZE GRAPHICS:"); LET = Xd",teet);
cteerdevicet);	
printfe"HOV MANY GR	APHS DO YOU VISH DISPLAYED? -);
number=stol(gets[st for {1=1;i<=number; {	ring(1)); **)
gets(string[1]);	THE THE MARE OF THE FILE CONTAINING POINTS: ");
colour(1) = ato)(CDLCUR7 =); gets(str1));
} cleardevice();	
- THIS BLOCK OF	CONSTRUCTORS AND
for(je);jCenumber;e	eji
	Anteringiji, il standili Anterini
closegreph();	FEN F1621 3;
enit(1);	
rewind(monthed):	
paint=1; end=0;	
/	********
on THIS BLOCK READS GROUNDERSEERSEERSEERSEERSEERSEERSEERSEERSEERS	DC FILES.
If (fscanf(smoot	hed,"%e %e",&y[point],&x[point]) == EOF)
point++;	
	nalat_7.
for (1=1; 1<=numbe	po m; r_of_points; 1++)
1#[1]+#[1];	
) }	
THIS BLOCK DISP	LAYS THE GRAPHICS.
{	nt; huisenst calcutilt.
delay(19);	urs'shibaruri'coron.fil;
} delay(10000);	
closegraph(); fcloseall(); }	

(1 column) 1

.

```
Sast modified: Mon Oct 22 02:20 19901 Curs.c
                                                                                                                                                                                                                                                                                                                                                                                                                                                                       (fast modified: Mon Oct 22 02:20 1990) ours.c
dinclude (atdia.h)
dinclude (atdia.h)
dinclude (atdia.h)
dinclude (ata.h)
dinclude (key,h)
dinclude "chrono.h"
dinclude "getarg.h"
                                                                                                                                                                                                                                                                                                                                Cross-Hair with Position Display near Cursor on Image\n\
Cross-Hair with Position Display at current position on image\n";
static unsigned cursor = 1; /s Current display function =/
                                                                                                                                                                                                                                                                                                                       \frac{2}{3}
                                                                                            /s Required by all SII applications =/ /s Define IA_s symbols s/ /s Define Key symbols s/ /s Required to get processing timer s/ /s Define the getarg function s/
                                                                                                                                                                                                                                                                                                                                    printf("Coreco Inc., SII Demonstration: Cursor Utility v: Xa\n\n", SII_RILEAS
argv[8] = MULL; /# Strip off program name #/
  Sifndef max
Sdefine max(a,b)
Sendif
                                                                                            /# Define max macro if not defined #/
{((a) > (b)) 7 (a) ; (b))
                                                                                                                                                                                                                                                                                                                                   /s Check if -? was selected by the user e/
/s if so, then print description of program to stdout e/
/s And estit to DOS e/
if(getarg(argc, argv, =?~, NLL))
  /* Prototypes specific to CURS.C */
static void menu( void);
static void dispit [V eim, int x8, int y8, CRLVL gri8, int x1, int y1, CRLVL gri
static void dispit [V eim, int x8, int y8, CRLVL gri8, int x1, int y1, CRLVL gri
                                                                                                                                                                                                                                                                                                                                            {
    printf("Description:\n");
    printf("n");
    printf("n");
    printf(" This program allows you to:\n");
    printf(" -Manipulate a cross-hair Gursor.\n");
    printf(" -Annotate the ladge at current cursor location.\n");
    printf(" -Control the drawing style for annotation.\n");
    printf(" -Faint a closed area.\n");

  /s Prototypes that can be exported to another application s/
vold ask_uvall char eprompt, unsigned vain, unsigned vmax, unsigned vdefault, un
vold gpartel IV atua;
                                                                                                                                                                                                                                                                                                                                            printf(" -- #aint's closed sres, (n');
printf("\n");
printf("\n");
printf(" This program is interactive and a senu that describes\n");
printf("\n");
prinf("\n");
printf("\n");
printf("\n");
printf("\n");
pri
  /68
  Description:
           Frints to atdout a menu of the function keys availables.
  static void menu()
                                                                                                                                                                                                                                                                                                                                               mit(1);
                                                                                                                                                                                                                                                                                                                                                                                                                                                /* Exit to 203 */
         printf("Careco Inc., SII Demonstration: Cursor Utility v: Xm\n", SII_R(LEASE)
printf("\n");
printf(" \F1: Annotate lange at current position\n");
printf(" F2: Select Cursor Type\n");
printf(" F3: Change Graphic Attributes for Annotation and Painting\n");
printf(" F3: Change Graphic Attributes for Annotation and Painting\n");
printf(" F3: Start Painting at current position\n");
printf(" F3: Start Painting at current position\n");
printf(" T3: Current Carsho, Carriette Current position\n");
printf(" F3: Start Painting at current position\n");
printf(" F3: Start Painting at current position\n");
printf("Cursor Special Keys\n");
printf(" PgDn: Howe to top\n");
printf("Crtl <-: Howe to left-hand side\n");
printf(" Tcl ->: Howe to center\n");
printf(" Home: Howe to center\n");
printf(" Nome: Howe to center\n");
printf(" Home: Howe to center\n");
            printf("Coreco Inc., SII Demonstration: Cursor Utility vi Xa\n", SII_R(LEASE)
                                                                                                                                                                                                                                                                                                                                    mit_init( argc, argv);
error = getsiierr();
if(_error)
                                                                                                                                                                                                                                                                                                                                                                                                                                                /= Initializes 311 and binds to driver(s) =/
/= Gets error status =/
/= Exits on fatal errors =/
                                                                                                                                                                                                                                                                                                                                              printf("eas Xs eas\n", Silerrang( error)); /# Print error using default
if[ error == INODEV)
                                                                                                                                                                                                                                                                                                                                                         printff"An OOX driver is associated with each device\n");
printff"rou must load at least one OOX driver before calling this progr
                                                                                                                                                                                                                                                                                                                                              ແນ້(1);
                                                                                                                                                                                                                                                                                                                                   setimpaget stdim, IV_GREEN);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 /* Work an green page 1f RGB #/
                                                                                                                                                                                                                                                                                                                                  /# Allocate a working im e/
/# This im holds the graphic attributes for annotation e/
/# This allows changing the graphic attributes without affecting the e/
/# Operation of the cross-hair cursor e/
/# stdim and im cover the same area of the frame buffer memory e/
im = im_childt stdim, B, B, getxien(stdim), getylen(stdim));
  ∕•=
 Prototype:
                                                                                                                                                                                                                                                                                                                                   x = (getxmax(iw) + getxmin(iw))/2;
y = (getymax(iw) + getymin(iw))/2;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 /* Place CURSE to center of is */
void dispxt iw, x0, y0, gr10, x1, y1, gr11, mode)

IV m1w; IV where drawing occurs

int x0, y0; Position of CUR30

int x0, y1; Position of CUR31 (not used)

int mode; Indicates which cursor is active CUR30 and/or CUR31

GRLVL gr10; Gray level of pixel at CUR31 (not used)

GRLVL gr11; Gray level of pixel at CUR31 (not used)
                                                                                                                                                                                                                                                                                                                                   menu();
while[ istop]
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 /* Display menu to uper e/
/* Frocess commands until uper escape
                                                                                                                                                                                                                                                                                                                                              (
int key;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   /* Key code */
                                                                                                                                                                                                                                                                                                                                              /s Gives control to interactive function s/
/s Interactive function returns if user press an undefined key s/
/s This key is interpreted at the caller's level s/
 Description:
The dispx functions are examples of a display function (DISPECT).
The display function is called by la_dx, and presents information to the monitor
This display function is called 2 times per cursar position:
The first time, it displays the information.
The second time, it erases the information.
                                                                                                                                                                                                                                                                                                                                              key = is_dat atdiw,
                                                                                                                                                                                                                                                                                                                                                                                            stdiw,
Ex, Sy,
HRLL, HRLL,
Emode,
defenouse,
dispfct[ curser],
ie_drwcurs);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 /m Current position of curmor e/
/m Cull31 im not used e/
/m Current mode e/
/m Use default mouse function s/
/m Use current display function from t
/m Dimplay cursor (declared in IA.H) =
 Each implementation is described in the cursor_def string near main().
                                                                                                                                                                                                                                                                                                                                               if( cursor += 1)
printf("$25s\r", "");
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   /* If cursor 1 is used #/
/* Erase last text printed with dimpt
static void displt IV eiw, int x0, int y0, GRLVL grie, int x1, int y1, GRLVL gri
                                                                                                                                                                                                                                                                                                                                              switch( key)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   /* Process special keys a/
           Static int visible - 0:
                                                                                                                             /* This flag will toggle with each call to disp
                                                                                                                                                                                                                                                                                                                                                          CASE F1:
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   /P Writes text at current position =/
           iff tvisible)
                                                                                                                              /# First time, print the information #/
                                                                                                                                                                                                                                                                                                                                                                    Ther str[80];
printf("Enter text 7 ");
getst str);
Tw_text( tw, x, y, str);
                     visible = 1; /= Toggle flag e/
printf("(X4d: X4d): X5u\r", x0, y0, grie);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  /* Ask for text to erite e/
/* Get a string e/
/* Write string at current curbor posi
/* Use to instead of stdiw e/
          else
                     visible = 0;
                                                                                                                       /# Second time, nothing to erase, toggle flag #/
                                                                                                                                                                                                                                                                                                                                                                    bfeak:
          ¥
                                                                                                                                                                                                                                                                                                                                                     case F2;
ssk_uval( cursor_def,
e, H6_CURSOR,
cursor,
&cursor);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 /= Select cursor e/
/= Prompt with list of display functio
/= Eange for cursor e/
/= Current cursor in the default e/
/= Address of cursor e/
  /s disp2 is defdisp (See SII REF) s/
/e position for disp3 e/
static int xt = 0;
static int yt = 0;
                                                                                                                              /* Position where information is displayed #/
                                                                                                                                                                                                                                                                                                                                                                     1ft cursor ++ 3)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  /* for disp1, set writing position a/ /* to current position a/
static void disp3( IV ers, int x9, int y8, GRLVL gris, int x1, int y1, GRLVL gri
                                                                                                                                                                                                                                                                                                                                                                               xt x;
            char str[68].
                                                                                                                                                                                                                                                                                                                                                                               yt + y;
           sprintf( str, "(Xd, Xd): Xd", x8, y8, gri8); /= Built the string s/
[W_text( iw, Xt, yt, str); /= Frint the string at (xt, //)
                                                                                                                                                                                                                                                                                                                                                                    break:
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  /* Changes Graphic attributes of iw e/
/* These attributes apply to iw_tert()
                                                                                                                                                                                                                                                                                                                                                         case F1-
                                                                                                                                                                                                                                                                                                                                                                    gpatt( 1w);
break;
  int_main( int argc, char swargv)
         (
int error;
IV eim;
int stop = 0;
int x, y;
int mode = CLR30;
GRLVL edge = 0;
                                                                                                                            /> Error Code from SII s/
/> Secondary handle to stdiw s/
/> flag to stop processing functions from user
/> Current cursor position relative to stdiw s/
/> current mode s/
/> Edge gray level for painting s/
                                                                                                                                                                                                                                                                                                                                                        Cabe A_F3;
gpatt( stdiw);
break;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 /s Changes Graphic attributes of curse
/* These attributes apply to is_dx, di
                                                                                                                                                                                                                                                                                                                                                         case FSI
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  /= Start Painting at current position
                                                                                                                                                                                                                                                                                                                                                                 #define HB_CURSOR 4
    static DISFECT display functions =/w Table of display functions =/
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 /s Prospt for edge s/
/s Range for edge grey level s/
/s Default value s/
/s Pointer to edge s/
                     MELL,
dispi,
defdisp,
                                                                                                                                                                                                                                                                                                                                                                   TRIG CHEONO;

10 paintt im, x, y, edge);

11m = CHEONO;

PREMEMONO( midim, time);

error = getatierr();

if( error)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 /* Trigger timer m/
/* Start painting at current position
/* Cat processing time m/
/* Frint processing time m/
/* Gats error statum m/
/* If error eccurs m/
/* Frint error using default error mes
                        d1203,
            static char ecursor_def +
                                                                                                                             /= Description of each display function a/
FUTTC Char Scarbary of the state of the stat
```

A.31 Cicolumnal 2

```
Sail modified; Mon Oct 22 02:20 1990) outjue
                                                   printf("eee Xm eee\n", silerrangt erroral;
                                          break,
                               case IA_QUIT;
stop = 1;
break;
                                                                                                                                         /= Special memory from is_dx() =/
/= Stop main loop =/
                      iff istop) menutly
        intop)
}
su_free( tw),
ali_end();
return( $;
}
                                                                                                                                         /* Display menu again #/
                                                                                                                                         /s Telease iw handle s/
/s Ends SII s/
   /==
  Synopeis:
  void ask_uval( prompt, min, max, def, vel)
Cher sprompt; question asked to user
unsigned vann; minimum value
unsigned vany; maximum value
unsigned value; default value
unsigned svalue; pointer to destination value
  Description:
 This function asks the user for a parameter int the following format:
"Enter (prompt) [(min)..(max)] {(def)} 7 "
  The value entered is checked against the valid range min <= val <= max.
Default value is obtained by pressing (enter).
   88/
   void ask_uvalt char eprompt, unsigned vain, unsigned vaax, unsigned vdefault, un
            int key;
           elie
t
                    t
ungetc: (char) key, stdin);
ifi scanf( "Xd", vslue) i= 1) svalue = vdefault;
getchar(); /s Gets return from user s/
         if avatue < vain [] avatue > vmax] avalue = vdefautt;
printf("\n");
  /**/
  Synopsis:
 void gpatt( iw)
IV eiw; Image Vindow that will be modified.
  Description:
  This function asks for each of the image window graphic attributes, taking values from the keyboard using ask_uval().
an/
void gpatti IV nin)
          unsigned ops, t, sclip, dir, zoosx, zoosy;
GRLML grts, gril;
GRLVC gri0, gri1;
ask_uval( "gray level for background", 8, getmaxgrl(iw), getgrl0(iw), &grl0);
ask_uval( "gray level for foreground drawing", 8, getmaxgrl(iw), getgrl1(iw),
ask_uval( "operation mode\n\
8: &IFLACE(n)
1: XOE(n)
2: AMO\n\
3: QP \n\
4: AIN(n)
5: HAC(n)
6: AOO(n)
7: AOO with saturation to MAXGRL(n)
6: AOO(n)
7: AOO with saturation to MAXGRL(n)
8: ASS of difference(n", 8, 8, getopaile), Eops);
ask_uval( "transparency mode(nb: Off (draw background)\n1: On (don't draw bac
ask_uval( "transparency mode(nb: Off (draw background)\n1: On (don't draw bac
ask_uval( "transparency mode(nb: Off (draw background)\n1: On (don't draw bac
ask_uval( "transparency mode(nb: Off (draw background)\n1: On (don't draw bac
ask_uval( "transparency mode(nb: Off (draw background)\n1: On (don't draw bac
ask_uval( "transparency mode(nb: Off (draw background)\n1: On (don't draw bac
ask_uval( "transparency mode(nb: Off (draw background)\n1: On (don't draw bac
ask_uval( "transparency mode(nb: Off (draw background)\n1: On (don't draw bac
ask_uval( "transparency mode(nb: Off (draw background)\n1: On (don't draw bac
ask_uval( "transparency mode(nb: Off (draw background)\n1: On (don't draw bac
ask_uval( "transparency mode(nb: Off (draw background)\n1: On (don't draw bac
ask_uval( "transparency mode(nb: Off (draw background)\n1: On (don't draw bac
ask_uval( "transparency mode(nb: Off)
0: Vertical, turned to the north\n\
0: Vertical, turned to the north\n\
0: Vertical, turned to the south\n\
0: Vertical, turned, the tert triing zoom', 1,
           u_gpatt( iw, grl8, grl1, (int) opu, (int) t, (int) aclip, (int) dir, (int) Z
metzoomx( iw, zoomx);
```

```
S columna 3
```



.... PROCEAN spinning. C calculates the interfacial tension from the spinning drop profile _____ Finclude (graphics,h) Finclude (stata,h) Finclude (conta,h) Finclude (statib,h) Adefine NPOINT 3003 Adefine NPOINT 3003 void masniint args, char eeargv: FILE edrop Fill adros: int and point.number_of_boints.test.entered.entrado.entre.left.right.niddle; int dkiNFOINT1.dv[NFOINT1.knigh[NFOINT2].ynigh[NFOINT2].wlow(NFOINT2].ylow(NFO int point.joonth.ingh.low.t.j; float sum_som.WIGH.tLOV.diameter; float delta_rho.omega.m.taw.gamma.diametre_reel.speed; char string[20]: int device=OETLCT, mode; ADDITUDE AND SO THE FILE CONTAINING THE DROP PROFILE ARE IMPUTTED OF BY THE USER CITECT(); if ((drop + fopen(argv[1],"rt")) == NRL1 ₹ PRINTFUTURABLE TO OPEN DROP FILE"); exit(1); 3 THE POINTS OF THE PROFILE ARE READ FROM THE DROP FILE. end=0; point=1; rewind(drop): while (tend) if (fscanfidrop,"%d %d", &dy[point], &dx[point]] == EOF) endel; point++; number of points-point-2; - THIS BLOCK INITIALIZES THE CAMPLES SO THAT THE PROFILE CAN BE DISPLATED - CRAMICALLY ON THE SCREEN. initgraph(idevice, inde, HLL); if ((test-graphresult()) 1+0) printf("\n\UMABLE TO INITIALIZE CRAPHICS");
printf("\nGRAPHRESILT= xd", test);
exit(1); cleardevice(): far (paint=0; paint <= number_af_paints ; ++paint)
 putpixel(dx(paint),dy(paint),VM(TE);</pre> / THIS BLOCK DISPLAY A CAID AND THE USER IS PROMPTED TO CUT THE ROUNDED • EDGES OF THE DROP TO LEAVE JUST THE STRAIGHT LINES TO CALCULATE THE • DIAMETER OF THE DROP Cleardewice(); gotoxy(1,1); printf("\n TO GET THE INTERFACIAL TENSION FROM THE DROP PROFILE"); printf("\n' TO HAVE TO GIT THE ROUMOED EDGES."); printf("\n' AND YOU ARE PROFITED TO ENTER TWO VALUES;'); printf("\n' THE VALUE OF X CORRESPONDING TO THE RIGHT ROUMOED EDGE"); printf("\n AND THE VALUE OF X CORRESPONDING TO THE RIGHT ROUMOED EDGE"); setcolor(8102); lime(1,50,623,50); lime(4,150,623,100); lime(4,150,623,100); lime(4,250,623,260); lime(4,250,633,250); lime(4,250,633,250); lime(4,250,633,250); lime(5,250,633,250); lime(5,250,633,450); lime(5,250,635,450); setcolor(SLUE)

line(608.8,608,473);

(last modified: Sun Oct 11 03.01 1992) 4:\spinning.c

/second second s

::

enfered=0: entres); entrado=0; PRINTER THAT PLEASE ENTER & COORDINATE WHERE TO CUT THE LEFT ROUNDED LOGE -1. while itentered! left+atoricetsistring));)ffileft>+31441eftC+6391) eise printri-OUT OF RANGE, PLEASE REENTER: "1; while Lientrei right=atoi(cets(string)); if((right)=0)&L(right(=633)) entre=1; entreet; else printfimout of RANGE, PLEASE REEKTER: "); . HERE THE USER IS PROMPTED TO ENTER THE Y MIDDLE OF THE DROP PRINTER THE PLEASE ENTER Y COORDINATE OF THE MIDDLE OF THE DROP: - 31 while (tentrado) l Biddleestail(etsistring)); if({Riddle>=03&&{Riddle<=473}} eise printfordut OF RANGE, PLEASE REENTER: "1: pointhel; pointlel; for (point+1;pointC+number_of_points;point++) if (dx[point])[eft #1 dx[point](right #1 dy[point](middle) xhigh(pointh)=dx[point]; yhigh(pointh)=dy(point]; pointh==; if id=[point]>teft #1_dx[point]<right #1_dy[point]>middle) xiow(pointl)+dx[point]; yiow(pointl)+dy[point]; point[++; 3 high-pointh-1; low-pointl-1; sum+0; 100-0 THICHOLS TLOVED: for cist; iC+high; i++) sumasum-yhigh[1]; 1 for (jet;)<=low;j++) som=som+ylow(j); THICH-sun/high; TLOV-son/low: dianeteretLOV-THICH; printfith THE DROP DIARETER IS XP", diameter); printf("\n\n PLEASE ENTER DELTA_RHQ IN g/cmcube; ");
delta_rho=atof(getsistring)); printf("\n\n PLEASE ENTER OMEGA IN rpm; "); onega=atofigets(string)); printf("\n\n PLEASE ENTER n: "); n=atof(gets(string)); printf("\n\n PLEASE ENTER OFFICAL ENLARGMENT IN pixels/cm; "); taumatof(gutsistring)); diametre_reeiwdiameter/(netau); speed=(omegas2=3,14116)/68; gamma=deita_rhosspeed=speedediametre_reeisdiametre_reei; printf("\n THE INTERFACIAL TENSION in dyn/cm + Xf", gammal;

C COLUMNS

C coiuir

APPENDIX B₁

This Appendix includes the experimental results of interfacial tension between immiscible polymer pairs as discussed in Chapter 4.

The results reported are divided into four categories:

a) polypropylene / ethylene vinyl alcohol copolymer

b) polypropylene / polystyrene

c) polyethylene / polystyrene

d) polypropylene / blends of monodisperse polystyrene

For all the results presented here the uncertainty in interfacial tension measurement represents the reproducibility of the experiment.

Temperature (°C) (K) Materials	202 475	211 484	218 491	224 497	232 505	241 514
<u>Commercial</u> <u>Polypropylene &</u> <u>Ethylene Vinyl Alcohol</u> PP _c /EVOH	-	19.63±0.85	18.40±0.93	16.43±0.40	14.83±0.02	13.40±0.10
Pure Polypropylene & Ethylene Vynil Alcohol PP _p / EVOH	17.00±0.10		-	12.25±0.25	-	-

Table B1.1 : Interfacial tension between PP and EVOH

Temperature (^O C) (K)	178 451	182 455	186 459	196 469	203 476	208 481	213 486	218 491	226 499	228 501	240 513	248 521
Materials												
Polypropylene & Monodisperse polystyrene												
ዮ° _p /ዮS ₁ 938	-	-	4.26±0.50	-	-	-	-	-	-	-		-
PPp/PS2 1,589	-	-	4.64±0.08	-	-	-	~	-	-	**	-	+
PP _p /PS ₃ 4,755	5.78±0.29	-	5.23±0.05	5.01 ± 0.42	4.5±0,29	-	4.26±0.05	-	-	-	-	-
PP _p /PS ₄ 19,417	6.61±0.30	-	6.11±0.20	5.75±0.33	-	5.15±0.29	4.92±0.30	-	-	-	-	-
PP _P /PS ₅ 86,438	-	7.06±0.30	6.83±0.07	6.46±0.45	-	6.00±0.38	5.53±0.30	-	-	-	-	-
PP _p /PS ₆ 380,000	-	-	-	-	-	-	-	-	6.30±0.11	6.15±0.20	5.67±0.12	5.47±0.20
Polypropylene & Polydisperse polystyrene		/										
PPp/PS 115,500	11.04±.34	-	10.80±.24	10.10±.19	9.25±0.29	-	-	8.46±0.06	7.74±0.17	-	-	-
Polypropylene & Blends of Polystyrene			·									
₽₽ _₽ /₽S _a	9.84±.29	-	9.36±.01	8.65±.15	-	7.25±.17	-	6.78±.22	-	-	-	-
₽₽ _₽ /₽S _b	6.84±.08	-	6.04±.03	5.41±.12	4.81±.10	-	4.23±.05	-	-	-	-	
· PPp/PSc	6.03±.20	-	5.66±.12	5.25±.03	-	-	-	-	-	-	-	-
)												

Table B1.2 : Interfacial tension between polystyrene and polypropylene

B1.3

APPENDIX B, EXPERIMENTAL DATA

.

Temperature (⁰ C) (K) Materials	130 403	150 423	160 433	169 442	178 451	186 459
Monodisperse Polystyreye & Monodisperse Polyethylene						
PS3 4,755 / PE3 1,870	7.00±0.20	6.41±0.21	6.25±0.12	6.00±0.30	5.43±0.33	5.16±0.08
PS ₃ 4,755 / PE ₂ 1,050	6.70±0.19	6.06±0.22	5.58±0.05	5.33±0.09	4.79±0.10	4.47±0.10
PS ₃ 4,755 / PE ₁ 680	6.40±0.20	5.54±0.30	5.11±0.42	-	-	-
<u>Monodisperse Polystyrene &</u> Polydisperse Polyethylene						
PS ₃ 4,755 / PE _c 1,950	-	-	-	-	-	5.44±0.08
PS ₃ 4,755 / PE _b 1,050	-	-	-	-	-	4.58±0.40
PS ₃ 4,755 / PE ₄ 770	-	-	5.20±0.13	4.49±0.08	3.91±0.08	3 28±0.08

Table B1.3 : Interfacial tension between polystyrene and polyethylene

B1.4

Table B1.4 : Interfacial tension between blends of PS and PP at a temperature of 186° C as a function of composition for the blend PS_2/PS_4

Percentage of Polymer with Higher Molecular Weight (%)	0/100	3/97	8/92	20/80	50/50	70/30	100/0
Blends		_					
PS ₂ 1,589/PS ₄ 19,417	4.64±0.08	6.14±0.20	7.70±0.34	9.15±0.25	9.8±0.70	8.40±0.4	6.11±0.20

Table B1.5 : Interfacial tension between blends of PS and PP at a temperature of 186°C as a function of composition for the blend PS₃/PS₅

Percentage of Polymer with Higher Molecular Weight (%)	0/100	2/98	5/95	15/85	50/50	100/0
Blends						
PS ₃ 4,755/PS ₅ 86,438	5.23±0.05	7.01±0.19	8.60±0.18	9.23±0.49	9.20±0.20	6.83±0.07

APPENDIX B₂

This Appendix gives an estimation of the solubility parameter for ethylene vinyl alcohol copolymer (EVOH) from the group molar attraction constants of Small [1953] and Hoy [1970].

According to Small [1953] and Hoy [1970] the solubility parameter for a polymer, δ , can be estimated as:

$$\delta = (\sum F)\rho/M_{\circ}$$
(B2.1)

where F are the various molar attraction constant for the different chemical groups, ρ is the polymer density and M_o is the molar mass of the repeat unit of the polymer.

The chemical formula of ethylene vinyl alcohol copolymer is:

The ethylene vinyl alcohol copolymer used in this study has 32% of alcohol, i.e. q=0.32 and p=0.38.

The density and molar mass for ethylene vinyl alcohol copolymer are:

$$\rho = 1.15 \text{ g/cm}^3$$

M_o=33.12 g

The contributions of the Hoy and Small values are

Group	F
CH ₂	0.68x2x269+0.32x269
CH I OH	462x0.32
	Σ=656

and thus the estimation of the dilution parameter for ethylene vinyl alcohol copolymer is:

$$\delta = 656 \times 1.15/33.12 = 22.8 (Jcm^{-3})^{\frac{1}{2}}$$
 (B2.2)

REFERENCES

Hoy K.L. [1970] : J. Paint Technol. Vol 42, p 76, (1970).

Small P.A. [1953] : J. Appl. Chem., Vol 3, p71, (1953).