Computational Molecular Design Using Tabu Search

By

Joseph Hacker

Submitted to the graduate degree program in Chemical and Petroleum Engineering and the Graduate Faculty of the University of Kansas in partial fulfillment of the requirements for the degree of Master of Science.

_____

Chairperson Dr. Kyle Camarda

_____

Dr. Stevin Gehrke

_____

Dr. Sarah Kieweg

Date Defended: June 28, 2011

The Dissertation Committee for Joseph Hacker

certifies that this is the approved version of the following dissertation:

Computational Molecular Design Using Tabu Search

_____

Chairperson Dr. Kyle Camarda

Date approved: April 17, 2012

Abstract

The focus of this project is the use of computational molecular design (CMD) in the design of novel crosslinked polymers. A design example was completed for a dimethacrylate as part of a comonomer used in dental restoration, with the goal to create a dental adhesive with a longer clinical lifetime than those already on the market.

The CMD methodology begins with the calculation of molecular descriptors that describe the crosslinked polymer structure. Connectivity index are used as the primary set of descriptors, and have been used successfully in other CMD projects. Quantitative structure property relationships (QSPRs) were developed relating the structural descriptors to the experimentally collected property data. Models were chosen using Mallows' *Cp* with correlation coefficient significance. Desirable target property values were chosen which lead to an improved clinical lifetime. Structural constraints were defined to increase stability and ease of synthesis. The Tabu Search optimization algorithm was used to design polymers with desirable properties. Finally, a prediction interval was calculated for each candidate to represent the possible error in the predicted properties.

The described methodology provides a list of candidate monomers with predicted properties near the desired target values, which are selected such that the adhesives will show improved properties relative to the standard HEMA/BisGMA formulation. The methodology can be easily altered to allow for additional property calculations and structural constraints. This methodology can also be used for molecular design projects beyond crosslinked polymers.

Acknowledgements

I would like to thank Dr. Kyle Camarda for his contribution to this work, and for being an increasingly supportive advisor. I would also like to thank all of the friends that I have made here. You've helped me grow into a better person these past years, and I will miss you greatly. Finally I would like to thank my wife, for supporting me during my long hours, and for pretending to listen to me whenever I rattled on about my work.

Table of Contents

List of Figures

List of Tables

Chapter 1.

Introduction

1.1 Motivation

The motivation for this research begins with the choice between dental resin composites and dental amalgam. Fillings in the anterior teeth almost exclusively use resin composites, as well as most posterior depending on the market. Resin composites have many advantages over amalgam including improved aesthetics and lower environmental impact. Amalgam is still being used in posterior fillings because it is difficult to apply resin composites where it is harder to stay dry, and because amalgam has a significantly lower failure rate than resin composites. The failure rate of resin composites is more than 50% greater than that of amalgam after 8 years (Collins, 1998)

Current research seeks to develop dental resin composites with improved longevity and a lower failure rate (Spencer, 2010). Much of recent research employs a trial-and-error approach: small changes are made to an established molecule, the molecule is synthesized, and its properties are tested in hopes that it is superior to the established molecule (Park, 2007; Edgar, *et al* 1999). This is an expensive and time-consuming process. With this method one could try to improve a few properties, for example by understanding the effect of rotational freedom on glass transition temperature (Bicerano,

2002), but it can be difficult to predict how this change will affect other properties. This method may cause some properties to improve while other properties deteriorate.

A more effective method of designing new materials is the use of computational molecular design. With a computational molecular design method, the values of many different properties are estimated, and the molecule is changed so that these property values are optimized simultaneously. Nearly any optimization method that can solve for a nonlinear objective function can be used, such as genetic algorithms (Konig, 1999), ant colony optimization (Korb, 2006), or Tabu search (Eslick, 2008). This solves the backwards design problem, which is to design a molecule with a set of desired properties. This is much more difficult than the forward design problem, which is predicting the properties of a known molecule (Gani, 1993). The solution of the reverse design problem was named one of the grand challenges in the computational needs in the chemical industry (Edgar, 1999).

Little attention has been given to error analysis in computational molecular design (Roughton, 2011). When developing QSPRs, there is experimental error and error from the QSPR not fitting the data perfectly. This error propagates through the design process. When the properties of the designed molecules are calculated the actual value of the property is most likely within a range of values, known as a prediction interval (Wasserman, 2004). Previous research in CMD only reports a single value as their result, while a ranged value may be more appropriate. This work uses a ranged value for predicted properties.

1.2 Optimization Procedure

The reverse design problem begins with the development of quantitative structure property relationships (QSPRs), which are statistically derived models that relate the molecule's structure to its properties. Property data is collected experimentally for the type of material being designed. Because polymer property data is often dependent on processing conditions (Eslick, 2009) property data published in the literature may not be consistent. In this work a set of consistent experiments was designed to collect important property data for a set of methacrylate polymers, such as glass transition temperature, viscosity, and storage modulus.

The experimental property data is then correlated with molecular descriptors of the polymer. In the past, group contribution methods have been used extensively to predict the properties of polymers and other materials. A major problem with using group contribution methods for polymers is that they miss some information by not taking into account the internal structure of the repeat units. The use of topological indices has been shown to be very effective in describing polymers (Camarda, 1999). In this work, Randić's molecular connectivity indices are employed as structural descriptors. These numerical values contain information about the bonds and oxidation state of each atom in the polymer repeat unit by examining the paths of the hydrogen suppressed molecular graph of the polymer (Randić, 1975).

Once the molecular descriptors and property data are collected, QSPRs are then created. Experimental data is exported to statistical software which creates a list of potential QSPRs with the highest correlation coefficient for each size (number of variables), leaving the user to create criterion for choosing which QSPR is superior. This is not always straightforward, as adding more descriptors will always raise the correlation coefficient. Adding too many descriptors will lower the statistical significance of the coefficients, leading to more error, or uncertainty, when using the correlation to design a new molecule. This work aims to create a criterion for QSPR selection using correlation coefficient, statistical significance, Mallows' *Cp*, and number of coefficients.

Then the optimization problem is formulated using target properties to create the objective function, and structural constraints. An optimization method is used to find a molecule which minimizes the objective function, resulting in a molecule with properties close to the targets. In this project we use the Tabu Search algorithm because it has been shown to handle the polymer design optimization problem effectively, and it allows the use of non-linear objective functions and QSPRs.

1.3 Research Goals

The goal of this project is to develop a method of computer-aided molecular design for crosslinked polymers. The method includes the development of quantitative structure property relationships (QSPRs), the formulation of the design problem, and the use of the Tabu Search optimization method to design crosslinked polymers. Additional analysis of the error from the QSPRs were done in order to calculate a confidence interval for the

calculated properties of the designed molecules, something which is frequently overlooked in many other studies (Roughton, 2011). A design example was completed for crosslinked methacrylate dental polymers, but the procedure will work for many other types of molecules (Lin, 2004; McLeese, 2010).

1.4 Thesis Overview

Background information is provided to the reader in Chapter 2. Included is background on the experiments that were done to collect property data, on molecular descriptors, and on the methods behind QSPR development, the field of molecular design, and optimization.

In the development of QSPRs, choosing the list of prospective molecular descriptors is an important step. The list of molecular descriptors studied, how they were calculated, and why they were chosen are given in Chapter 3.

The QSPRs that were developed during this research are provided in Chapter 4. This section describes how each QSPR was chosen, how their validity was tested, and how the prediction interval was calculated.

Once the QSPRs are developed, the optimization problem is then formulated. Details of how target properties and additional structural constraints were used to develop the objective function are given in Chapter 5. It will then explain how Tabu Search is used to

solve this optimization problem, and explains the advantages that Tabu Search has over other optimization methods for problems like this.

Multiple examples were performed with different sets of target properties to test the validity of our Tabu Search algorithm. Explanations of these examples, as well as a list of candidate monomers, are given in Chapter 6.

Conclusions and recommendations for future projects are provided in Chapter 7.

In the appendices, a more thorough explanation of experimental procedures is given. In QSPR development, experimental consistency is important. If the reader wishes to add to the experimental data provided in this research, it would be advised that they follow the experimental procedures provided here for consistency. The appendices also provide a manual for the Polymer Designer program designed by Eslick (Eslick, 2008) which was used extensively in this project. This manual should be considered an addendum to Eslick (2008), as this manual only explains how Polymer Designer can be modified in order to solve other design problems involving polymers or other molecules.

Chapter 2.

Background

This Chapter provides background about the experiments performed to collect property data, the QSPRs that were created to predict these properties for the entire space of methacrylate monomers, and the computational molecular design framework which utilizes these QSPRs to design a monomer which minimizes the objective function.

2.1 Experimental Background

This section provides background to the experiments done for property data collection, as well as background in the synthesis of the composite resins which are being studied in this project. Experiments were performed to collect property data for percent solubility, percent water sorption, storage modulus, rubbery modulus, and viscosity. These properties were chosen as they can describe the behavior of the resin both before and after polymerization, and can be used to represent clinical lifetime of the resin. Data was collected experimentally, rather than through literature research, to improve consistency of results. For example, the value of the recorded glass transition temperature can be very different depending on how it is measured (Bicerano, 1996). This would make it

impossible for a QSPR, depending on only the structure of the molecule, to accurately predict the measured property.

2.1.1 Sample Preparation

Dental resin composites are composed of monomers or comonomers and a photoinitiator, such as camphorquinone (CQ). The most common monomers used in dental resin composites are 2,2-Bis[4-(2-hydroxy-3-methacryloxypropoxy) phenyl]-Propane (BisGMA), ethoxylated bisphenol A glycol dimethacrylate (BisEMA), and urethane dimethacrylate (UDMA) (Sideridou, 2001). Other methacrylates, such as 2-hydroxyethyl methacrylate (HEMA) can be added to change certain properties of the final resin (Collins, 1998).

The resins are polymerized through light curing. A common photoinitiator system is the use of CQ as a photosensitizer, and N,N-dimethylaminoethyl methacrylate (DMAEMA) as a reducing agent (Sideridou, 2001). The photoinitiator system used in this study is CQ as a photosensitizer, ethyl 4-N,N-dimethylaminobenzoate (EDMAB) as a reducing agent, and the hydrophilic iodonium salt 2,6-dichlorophenol-Indophenol (DPIHP). This system gives a larger degree of polymerization than the standard photoinitiator system when the resin is polymerized in the presence of water. (Fouassier, 1993; Ye, 2009).

Dental resins are polymerized through the use of a curing light at the appropriate wavelength. The photosensitizer absorbs photons of a certain frequency range, exciting

the molecule to an activated triplet state. The most common photosensitizer, CQ, absorbs photons at 468 nanometers, or blue light (Lovell, 2001). Once in the triplet state, the photosensitizer reacts with the reducing agent to form an aminoalkyl free radical. An aminoalkyl free radical breaks the methyl-vinyl double bond group in the methacrylate to start the chain initiation for the chain growth polymerization. Because BisGMA and many of the other monomers used in the making of dental resins are dimethacrylates, crosslinking occurs (Cook, 1992). The purpose of the iodonium salt is to act as the reducing agent in the hydrophilic regions for resins cured in water, which the hydrophobic amino reducing agent cannot reach (Ye, 2009).

Resin samples for experimental testing are prepared by curing the resin in a mold so that the polymer sample will be either a beam, rod, or a film, depending on what properties are being determined (Sideridou, 2008; Podgorski, 2010). In the experiments performed in this study the beam samples were cured in rectangular glass beams with dimensions of 1mm x 1mm x 15mm. Samples for mechanical testing were formed as round glass beams with dimensions of 1mm x 15mm.

2.1.2 Storage Modulus and Rubbery Modulus

Storage modulus is a measurement of energy storage capability. The rubbery modulus is the storage modulus at temperatures higher than the glass transition temperature, when the resin is rubbery. A high storage modulus correlates to a high tensile strength (Bosze, 2006), so a dental resin composite with large storage and rubbery modulus is desired.

Storage modulus is measured using dynamic mechanical analysis (DMA), a technique widely used to study the viscoelastic behavior of polymers (Brostow, 2010; Deshayes, 2011; Ge, 2010). A sinusoidal stress is applied at a constant frequency, and the resulting strain is measured. For viscoelastic materials, there will be a phase difference between stress and strain. This gives the following equations for strain and stress:

$$\varepsilon = \varepsilon_0 \sin(t\omega)$$

$$\sigma = \sigma_0 \sin(t\omega + \delta)$$

where ω is the frequency of the strain, t is time, and δ is the phase lag between stress and strain in radians . For purely elastic materials there is no phase difference, so delta is zero. For purely viscous materials, delta would be 90 degrees (Meyers, 1999). The property *tan(δ)* can be used as a measure of how viscous a material is, with a value of zero being purely elastic and a value of one being purely viscous (Ferry, 1980).

The dynamic modulus is the ratio of stress to strain. The dynamic modulus can be divided into real and imaginary parts such that

$$E = E' + iE''$$

$$E' = \frac{\sigma_0}{\varepsilon_0} \cos(\delta)$$

$$E'' = \frac{\sigma_0}{\varepsilon_0} \sin(\delta)$$

where *E* is the dynamic modulus, *E'* is defined as the storage modulus, and *E''* is defined as the loss modulus. The storage modulus is a measurement of energy storage, as opposed

to the loss modulus which is a measurement of energy dissipation due to viscous forces (Menard, 1999).

In this study, the storage modulus was measured at 37^{o}C to simulate oral conditions, and the rubbery modulus was measured at 175^{o}C, well above the glass transition temperature for the systems being studied. The strain frequency was 1 Hz for both the storage and rubbery modulus.

2.1.3 Water Sorption and Solubility

Water sorption is a measure of how much water the resin absorbs. The presence of water in the polymer network may lower mechanical properties by acting as a plasticizer, or by interfering with hydrogen bonding between monomers (Park, 2009). A resin with high solubility is of concern as the leaching of molecules to the surroundings can cause the composite to break down over time. Thus resin composites of low water sorption and solubility are desired. The American Dental Association requires that water sorption be less than or equal to 40 μg per cubic millimeter, and the solubility be less than or equal to 7.5 μg per cubic millimeter (ADA, 2003).

The ADA has a standardized test for determining water sorption and solubility. The initial mass of a disk-shaped resin sample is measured (m_1). The sample is soaked in water for seven days at 37^{o}C to simulate oral conditions, and the saturated mass is measured (m_2). The sample is then dried in a desiccator at 37^{o}C and the mass is recorded again (m_3). The solubility is calculated as

$$W_{SU} = \frac{m_1 - m_3}{V}$$

and

$$W_{SP} = \frac{m_2 - m_3}{V}$$

is water sorption (ADA, 2003; Dhanpal, 2009).

Some studies instead weight the water sorption and solubility equations with initial mass instead of volume (Sideridou, 2004; Park, 2009). This study does the same, which is not an issue as the HEMA/BisGMA control sample passes the ADA standardized test, and finding a resin with superior properties to the control will result in a resin that will also pass the standardized test (Malacarne, 2006; Park, 2007).

2.1.4 Glass Transition Temperature

The glass transition temperature is the temperature where amorphous polymers transition between being hard and brittle to being soft and pliable. Above the glass transition temperature, thermal energy is high enough that long polymer chains can move around each other in random micro-Brownian motion, making the polymer appear rubbery. Below the glass transition temperature the polymer chains can only make short-range motions, making the resin appear hard (Fried, 2003). A dental adhesive resin near its glass transition temperature would be pliable and the dental restoration would not be secure. Dental adhesive resins with a glass transition temperature significantly higher than body temperature are desired.

The glass transition temperature of the resin can be measured using differential scanning calorimetry (DSC). A sample is placed in a temperature controlled chamber with a standard, and the temperature is slowly increased. The DSC measures the rate of energy needed to slowly raise the sample's temperature. From this data the heat capacity as a function of temperature can be calculated (Dean, 1995).

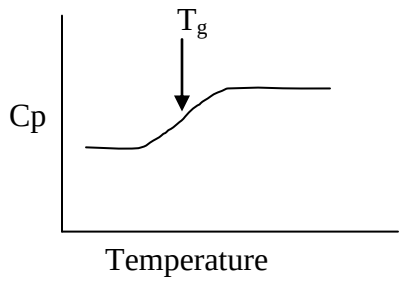


Figure 2.1- Glass transition temperature

During the glass phase transition the heat capacity increases as a second order transition; a continuous transition with no latent heat (USM, 2005). The glass transition temperature can be read from the DSC results as the median temperature where this heat capacity change is occurring (O'Neill, 1964). Experimental data can be found in the appendices.

2.1.5 Viscosity

The viscosity of the unreacted resin affects how well a composite can bond to the tooth surface. If the viscosity is too high, the composite does not bond to the tooth surface well, which leaves room for increased levels of bacteria to collect within the gap, causing

decay. During polymerization, some parts of the resin solidify before others. Polymerization shrinkage occurs, and the parts of the resin which are bonded to the surface will move away, leaving a gap. If the resin has a low viscosity, the still liquid resin can then flow into these gaps before polymerizing, decreasing the gap size (Spencer, 2010). Dental resin composites with viscosities that are lower than the standard are desired.

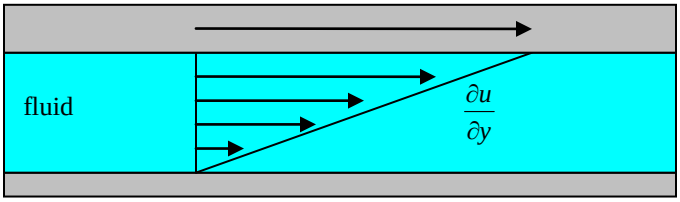


Figure 2.2 - Velocity gradient for a cone and plate viscometer

Viscosity is commonly measured using a cone and plate viscometer. A thin layer of resin is placed between a flat plate and a cone at a very shallow angle. As the cone rotates, the viscosity of the resin causes resistance to the rotation. The force that the viscometer applies to rotate the cone is converted to torque by dividing the force by the area of the plate (Barnes, 1993). For straight, parallel, uniform flow, the viscosity is proportional to torque using the equation

$$\frac{F}{A} \equiv \tau = \mu \frac{\partial u}{\partial y}$$

where u is the rotational velocity, and y is the position is the axial direction. For a Newtonian fluid, the velocity gradient in the axial direction is constant, so it can be calculated by dividing the rotational velocity of the cone by the thickness of the resin layer. The purpose of using a cone and plate geometry rather than two flat plates is that

using a cone keeps the velocity gradient roughly constant in the radial direction (Barnes, 1993). In this study, the viscosity was measured at a range of shear rates to confirm that the resins are Newtonian fluids. Experimental data can be found in the appendices.

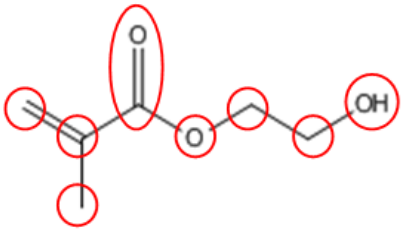
2.2 Molecular Descriptors

In order to design a model linking molecular structure to physical and chemical properties of interest, a numerical representation of a molecule's 2-D structure is required. Molecular descriptors provide a way to describe the structure of a molecule mathematically. Examples of simple molecular descriptors are molecular weight or number of rings. This section provides background for molecular descriptors and how they are calculated.

The group contribution method is a technique used to predict properties of molecules. Group contribution uses the idea that number and type of functional groups in a molecule is proportional to many physical properties. Group contribution has been used in polymer design (Satyanarayana, 2008) and in the UNIFAC method to calculate activity coefficients for equilibrium (Fredenslund, 1975).

The Joback method (Joback, 1987) uses group contribution to predict eleven properties of small organic molecules. The Joback method uses a very simple method of group assignment, making it useful for users with limited experience in chemistry. Figure 2.3 gives an example of calculating the boiling point with the Joback method.

$$T_b = 198 + \sum G_i$$



Group	T_b
$=CH_2$	18.18
=C<	24.14
CH_3	23.58
>C=O	76.75
CH_2	22.88
-O- (nonring)	22.42
OH (alcohol)	92.88

$$T_b = 198 + 18.18 + 24.14 + 23.58 + 76.75 + 2 \times 22.88 + 22.42 + 92.88 \cong 303.71K$$

Figure 2.3 - Calculating the boiling point using the Joback method (Joback, 1987)

Marrero and Gani (2001) expanded on the Joback method and other simple group-contribution methods for property prediction. The Marrero/Gani group contribution considers three levels of molecular groups. In the first group the entire molecule is described similarly to the Joback method. Some properties of small organic molecules only need to be described using the first group. The second group is used to better describe polyfunctional compounds and differentiate between isomers. The third group is used to better describe polycyclic compounds (Marrero, 2002). The second and third groups do not need to describe the entire molecule, and can overlap. The Marrero/Gani group contribution method has shown to be more accurate than the other simpler group contribution methods (Marrero, 2001).

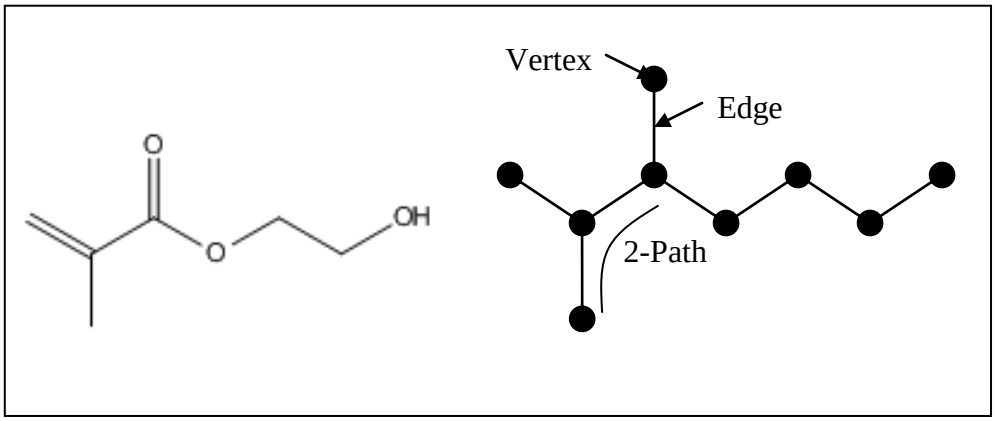


Figure 2.4 - The molecular graph for HEMA

Many molecular descriptors are found by examining the molecular graph, where each vertex represents an atom and each edge represents a bond. When calculating descriptors for organic molecules, the hydrogen molecules are often excluded in the molecular graph, because the number of hydrogen atoms is implied through valency. This is called a hydrogen suppressed graph (Bicerano, 2002; Eslick, 2009). Molecular descriptors that are found using the molecular graph are called structural descriptors.

A structural descriptor similar to the group contribution that has been used in molecular design is the Signature descriptor (Weis, 2010). The Signature descriptor describes the local neighborhood of a molecule starting from a root atom. The Signature extends outward from the root atom and records the atomic bonds present. The number of steps outward is equal to the predefined height, *h*. This is repeated for all the atoms and summed to give the molecular Signature,

$$^{h}\sigma = \sum_{\chi \in V} {}^{h}\sigma(\chi)$$

where ${}^{h}\sigma$ is the Signature descriptor of height h, χ is an atom in the molecule, and the set V is all the atoms present in the molecule (Brown, 2006). A height-0 Signature would be a list of the atoms present in the molecule.

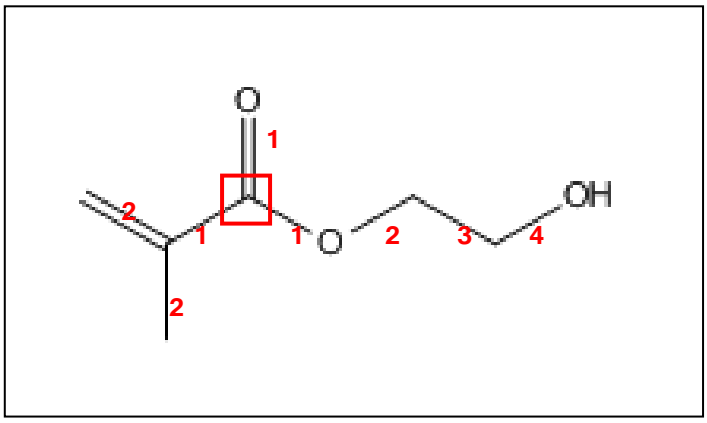


Figure 2.5– Bonds present in different heights of the Signature descriptor from a root carbon atom. Carbon-hydrogen bonds are not being represented.

Figure 2.5 shows one step in finding the Signature descriptor for HEMA. The height-1 atomic Signature for the root carbon atom would be [C]([C],=[O],O). This describes the identity of the root atom, the atoms which the root atom is bonded to, and the types of bonds. Computing this for the entire molecule gives a table of the atomic Signatures present with the number of times it occurs. For example, the height-1 atomic Signature [H](C) occurs seven times in HEMA. Similar to the group contribution method, the number of times an atomic Signature occurs can be correlated with the desired properties. The Signature descriptor has been used to design solvents (Weis, 2009), and polymers (Brown, 2006).

Another set of structural descriptors which have been used in polymer design are Randic's connectivity indices (Randic, 1975). Connectivity index contain information about the amount of branching in the molecule and the oxidation states of the non-hydrogen atoms by examining the paths of the molecular graph. Bicerano used zeroth-order and first-order connectivity index to correlate a large number of physical properties for straight-chain polymers (Bicerano, 2002). Raman and Maranas were the first to use connectivity index for product design (Raman, 1998). Connectivity index have been used successfully in the design of alkenes (Nelson, 2001) ionic liquids (McLeese, 2010) and polymers (Camarda, 1999; Eslick, 2009). This research uses connectivity index as its primary set of descriptors.

The simple and valence connectivity index are calculated from the simple atomic connectivity index and the atomic valency connectivity index. The simple atomic connectivity index, δ, is equal to the number of non-hydrogen atoms boned to a given basic group, which is also the vertex degree for the vertex in the hydrogen-suppressed molecular graph. The atomic valency connectivity index is found using

$$\delta^{v} = \frac{Z^{v} - N_{H}}{Z - Z^{v} - 1}$$

where Z^{v} is the number of valence electrons around the atom, Z is the total number of electrons around the atom, and N_H is the number of hydrogen atoms bonded to the atom (Bicerano, 2002). The *n*th order simple and valence molecular connectivity index are given by

$$^{n}\chi = \sum_{k \in \text{n-length paths}} \frac{1}{\sqrt{\prod_{i \in \text{atoms in k}} \delta_i}}$$

$$^{n}\chi^{v} = \sum_{k \in \text{n-length paths}} \frac{1}{\sqrt{\prod_{i \in \text{atoms in k}} \delta_i^{v}}}$$

where *k* is all of the paths of length *n*. In graph theory, a path is a sequence of vertices where the next vertex is always adjacent to the previous vertex. Two vertices are adjacent if there is a bond connecting them. The path length is equal to the number of edges in the path, so a zeroth-order connectivity index only examines the individual atoms and can be computed using the following equations (Bicerano, 2002).

$$^{0}\chi = \sum_{i \in \text{basicgroup}} \frac{1}{\sqrt{\delta_i}}$$

$$^{0}\chi^{v} = \sum_{i \in \text{basicgroup}} \frac{1}{\sqrt{\delta_i^{v}}}$$

Table 2.1- Simple and valence atomic connectivity index for basic groups used in this research

	δ	δ^v		δ	δ^v
C	4	4	C=	3	4
CH	3	3	O=	1	6
CH_2	2	2	O	2	6
CH_3	1	1			

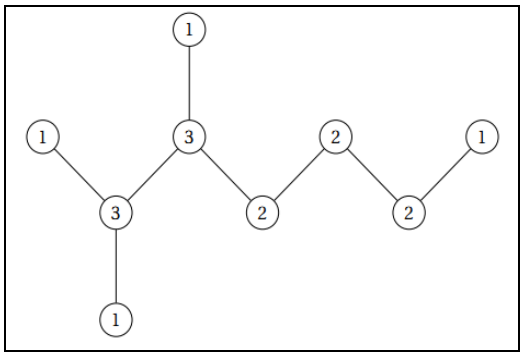


Figure 2.6 - The molecular graph of HEMA with the simple atomic connectivity index for each vertex

Figure 3 shows the molecular graph of HEMA with the simple atomic connectivity index for each vertex shown. The zeroth-order simple connectivity index would be equal to

$$^{0}\chi = \frac{1}{\sqrt{1}} + \frac{1}{\sqrt{3}} + \frac{1}{\sqrt{1}} + \frac{1}{\sqrt{3}} + \frac{1}{\sqrt{1}} + \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{1}} = 7.28$$

after summing over each atom. The first-order simple connectivity index would be equal to

$$^{1}\chi = \frac{1}{\sqrt{1\times 3}} + \frac{1}{\sqrt{1\times 3}} + \frac{1}{\sqrt{3\times 3}} + \frac{1}{\sqrt{1\times 3}} + \frac{1}{\sqrt{2\times 3}} + \frac{1}{\sqrt{2\times 2}} + \frac{1}{\sqrt{2\times 2}} + \frac{1}{\sqrt{2\times 1}} = 4.18$$

after summing each 1-path, or edge. The connectivity index is an extrinsic property so it is a function of the molecular weight of the molecule. A scaled, or intrinsic, connectivity index, ξ, can be found by dividing by the number of paths (Bicerano, 2002). Both intrinsic and extrinsic connectivity indices are used in this project.

This project studies crosslinked polymers. The degree of crosslinking has a great effect on the polymer properties, and many descriptors do not account for crosslinking. Bicerano correlated the change of glass transition temperature to crosslink density in crosslinked polymers (Bicerano, 1996). Researchers have shown crosslinking affects the

polymers' properties (Matsui, 1999; Manu, 2009), but little research has been done in the property prediction for randomly crosslinked copolymers (Eslick, 2009).

The Polymer Designer program used in this research uses a novel method to account for crosslinking (Eslick, 2008). The monomer concentration and degree of polymerization are predetermined, and a large random copolymer is randomly generated. The polymer is divided into an inner core and an outer buffer. Crosslinked polymer networks are generally treated as being infinite, but polymer graphs need to be finite. This means the chain has to be cut. The core and buffer technique separates the core from the chain cut by putting a buffer region of monomer groups in between. The descriptors are calculated based on the molecules in the core, with some buffer molecules being used depending on the type of descriptor being calculated. The size of the core and buffer region can be adjusted depending on the project. A larger core gives more consistent descriptor calculations, as there is randomness in the placement of monomers and crosslinks. A larger buffer region further reduces the effect of chain cuts. However, larger polymer graphs can be very computationally expensive, especially during CMD when thousands of candidate monomers might be generated (Eslick, 2008).

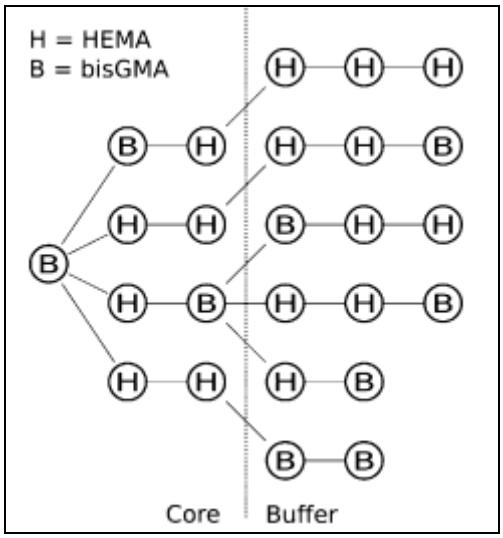


Figure 2.7 - The core and buffer region for a polymer graph. (Eslick, 2008)

Many other molecular descriptors exist which can be used to describe polymers. Todeschini and Consonni provided a comprehensive list of molecular descriptors which could be useful for this work in this project (Todeschini, 2000). The list of molecular descriptors, and the methodology of how they are calculated within the CMD framework, is provided in Section 3.

2.3 QSPR Development

This section describes the techniques used to develop and analyze the QSPRs used in this project.

2.3.1 Model Creation

This project uses multiple linear regression in the development of QSPRs (Draper, 1966). Non-linear QSPRs were correlated through manipulating the response and predictor variables such that linear regression could still be used. For example, the natural log of the response variable can be taken, or the response variable could be multiplied by a predictor variable before linear regression is done.

2.3.2 Model Selection

Choosing between models of different sizes (number of descriptors) is an issue in QSPR development. Model choice involves finding a balance between bias and variance. Choosing too few descriptors leads to high bias, or underfitting. Bias is the difference between the predicted value and observed value. Choosing too many descriptors leads to high variance, or overfitting. Variance is a measure of how sensitive the model is to the original data. A model with high variance won't be able to predict the properties of molecules that are outside of the original data (Bullinaria, 2010). There are numerous methods that try to find the proper balance. This section describes some of these methods.

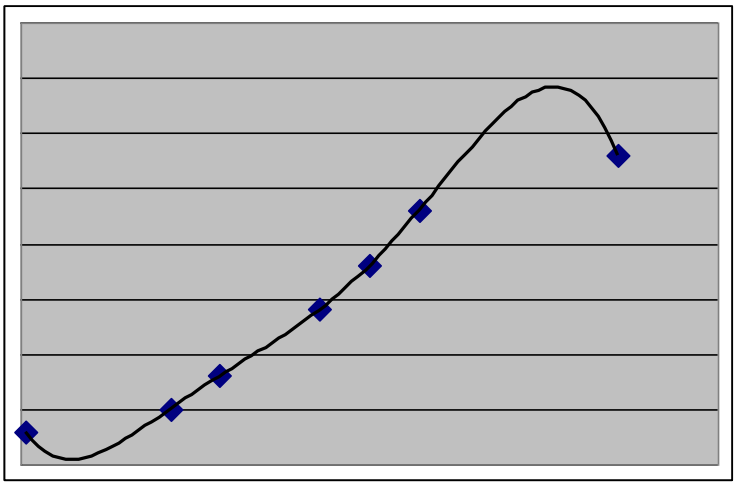


Figure 2.8 - A model with no bias but high variance

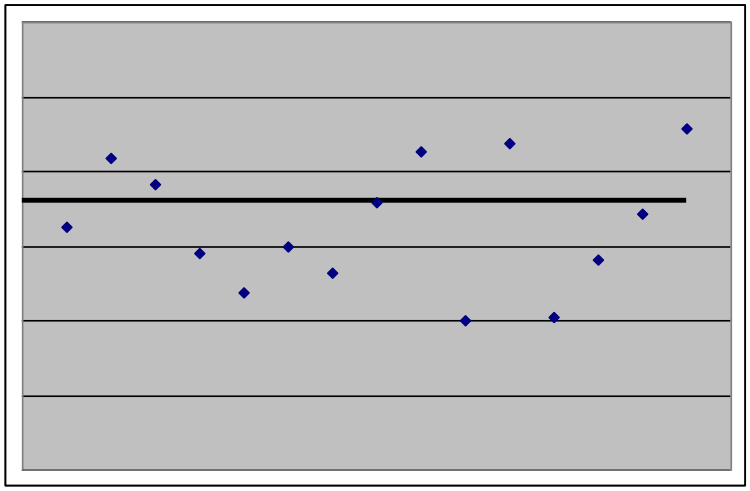


Figure 2.9 - A model with no variance and high bias

The coefficient of determination, r^2, can be viewed as a model selection technique. The coefficient of determination is defined as

$$r^2 = 1 - \frac{\sum_i \left(Y_i - Y_{p,i}\right)^2}{\sum_i \left(Y_i - \bar{Y}\right)^2}$$

where Y is the observed value, Y_p is the predicted value, and $\overline{Y}$ is the average observed value (Draper, 1966). The problem with using r^2 for model selection is that it only takes bias into account. Adding more descriptors will always increase r^2, which leads to overfitting and high variance. However, the r^2 value can be used to determine what the best model is of a specific size.

A method for comparing models of different sizes is Mallows' *Cp* (Mallows, 1973). Mallows' *Cp* addresses the problem of overfitting by putting a price on adding more descriptors. For a model with P descriptors chosen from a pool of k descriptors, *Cp* is equal to

$$Cp = \frac{\sum_{i=1}^{N}\left(Y_i - Y_{p,i}\right)^2}{\sum_{i=1}^{N}\left(Y_i - Y_{p,i}\right)^2}_{p=k} - N + 2P$$

where Y is the true value of the property, Y_p is the predicted value, and N is the number of data points (Wasserman, 2004). This equation could be thought of as

$$Cp = \text{Error} + \text{Complexity of Model}.$$

Models with values of *Cp* roughly equal to P are ideal, lowering variance while not dramatically increasing bias (Mallows, 1973).

Another method for comparing models is k-fold cross-validation. Cross-validation is used to assess how well a model will be able to describe outside data points, or data that was not used to develop the model. The data is first randomly divided into k groups of roughly equal size. For each group k, the model is reevaluated leaving out the data points

in *k*. Then the new model is used to predict the data points in *k*, and the error is used to calculate the cross-validation coefficient Q^2 (Wasserman, 2004). The value Q^2 has an upper bound of r^2. Values of Q^2 close to r^2 means the model has little variance, because changing the initial data set does not affect the overall error.

The cross-validation coefficient is calculated using the predicted residual sum of squares (PRESS) equal to

$$PRESS = \sum_{k=1}^{K} \sum_{i \in k} \left(Y_i - Y_{p,i} \right)^2$$

where *k* is a test set, *Y* is the observed value, and Y_p is the predicted value. The value of Q^2 is then equal to

$$Q^2 = 1 - \frac{PRESS}{\sum_{k=1}^{K} \sum_{i \in k} \left(Y_i - \overline{Y} \right)^2}$$

where $\overline{Y}$ is the average observed value with the *k*th set omitted (Picard, 1984). Because the groups of *k* are randomly selected the *k*-fold cross-validation should be repeated numerous times to find an average. The following graph shows how the randomness in *k*-group selection can increase error. If the data points circled were selected to be in the same group the value of *PRESS* would be very high. A widely used variant is the Leave-one-out cross-validation, where *k* is equal to the number of data points (Picard, 1984). Leave-one-out is computationally expensive because of the number of different models that need to be created. However, leave-one-out does not need to be repeated because it eliminates the randomness of *k*-fold cross-validation.

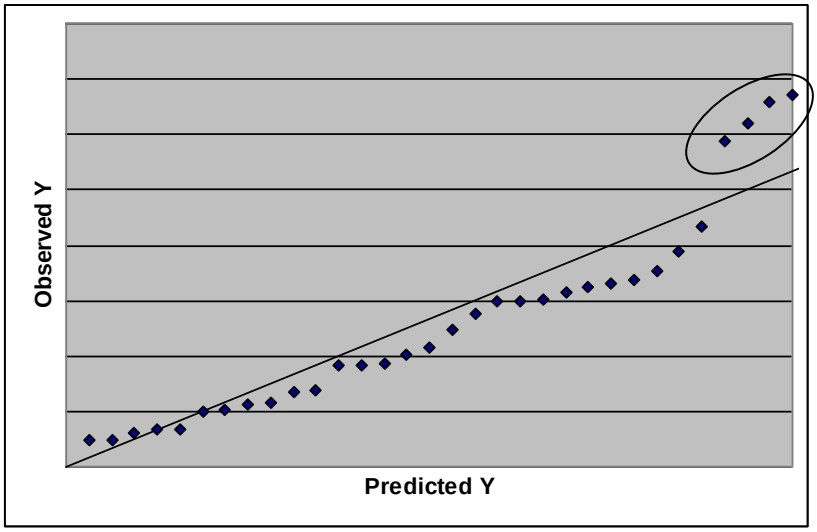


Figure 2.10 - *k*-fold cross-validation should be repeated numerous times to find an average

The significance value of each correlation coefficient can also be calculated. The *p*-value is the probability that one can obtain will get similar or better correlation results if there is no relationship between the predictor and response variable. Generally, if the *p*-value is less than 0.05 or 0.01 then the coefficient is significant (Wasserman, 2004). Models that pass the criteria for the correlation coefficient, Mallows' *Cp*, and cross-validation may still have coefficients that are not statistically significant.

2.3.3 Error Analysis

A concept within statistical analysis which has not been used extensively in molecular design is the prediction interval (Roughton, 2011). The prediction interval is similar to a confidence interval, but for predicted values. The prediction interval depends on the error in the original model, and on how different the predictor variables for the new observation are compared to the original variables. If the candidate molecule is very

similar to the molecules used to develop the QSPR, the prediction interval will be smaller. The prediction interval is equal to

$$\pm t_{\alpha/2,n-(k+1)} \sqrt{\hat{\sigma}^2 \left(1 + x'_p \left(X'X\right)^{-1} x_p\right)}$$

where t is the critical value of the t-distribution at the desired confidence level and degrees of freedom, $\hat{\sigma}^2$ is the mean square error, x_p is an array of descriptors for the new observation used in the model, and X is the matrix of descriptors of previously observed data points (each row is a different observation, each column is a different descriptor) (ReliaSoft, 2008).

After the molecular design algorithm finds a solution, the prediction interval can be calculated for each property. The results can be presented as a range in which the property lies in, instead of a single value.

2.4 Molecular Design and Formulating the Design Problem

This section provides an overview of molecular design, molecular design techniques, and the formulation of the design problem. Computational molecular design is the use of an optimization method to design a molecule or set of molecules which fit a set of desired properties (Gani, 1998). CMD can be used to greatly decrease the resources used in product design compared to the trial-and-error approach. Using CMD, a list of candidate molecules is created which should have the desired properties, making the experimental synthesis more efficient (Lin, 2005).

CMD requires the solution of both the forward and backward design problem. The forward design problem is the prediction of the molecule's properties based on its structure. The backwards design problem is finding a molecule which fits a set of desired properties (Edgar, 1999).

The forward design problem is usually solved through the use of either group contribution-additivity models or through quantitative structure property relationships (QSPRs). Group contribution has been widely used in molecular design, and uses the properties of atoms or groups to predict the properties of the entire molecule (Gani, 1991; Marrero, 2001; Friedler, 1998; Constaninou, 1994; Karunanithi, 2005). A major problem with the use of group contribution to describe polymers is that it does not take into account the order of the monomer repeat units (Camarda, 1999). More recently, the use of QSPRs with topological index as structural descriptors has been used successfully to describe polymers and other molecules (Camarda, 1999; Raman, 1998; Visco, 2002). QSPRs are developed by regressing property data versus structural descriptors, such as the Wiener Index, Randić's molecular connectivity index, or simple descriptors like molecular weight, to form an empirical model.

Once the forward design problem has been solved, the backwards design problem needs to be formulated. The objective function defines the set of target properties, and has the non-linear general form

$$f = \sum_{i \in \text{properties}} s_i \left(\frac{P_{i,\text{target}} - P_{i,\text{predicted}}}{P_{i,\text{target}}} \right)^2$$

where $P_{i,\text{predicted}}$ is the value of property *i* predicted by the QSPRs, $P_{i,\text{target}}$ is the desired value of property *i*, and s_i is a scaling factor used to adjust the importance of each property (Eslick, 2009). As the predicted properties approach the target values, the objective function approaches zero, so the objective function should be minimized. A disadvantage to this form of the objective function is that properties can not be minimized or maximized. However, this is not an issue as QSPRs should not be used to predict properties outside of the range of data used to formulate them (Eslick, 2008). The objective function can be written in other forms, perhaps in linear or convex forms to simplify the solution method. This is needed for some deterministic optimization techniques. This is not necessary in this project, as the Tabu Search algorithm can solve non-linear, non-convex problems.

Beyond the objective function, the design problem also has constraints. One constraint that must always be present in molecular design is that the molecule has to be feasible; the valency of each atom is satisfied, and the molecular structure is connected. Other structural constraints can be present, such as the exclusion of unstable peroxide groups, or a minimum and maximum molecular weight. Candidate molecules need to be checked for feasibility before the objective function for that molecule is calculated. If a molecule is infeasible it should be rejected immediately. In this project, most of the constraints are implied in the search algorithm; candidate molecules are changed such that an infeasible solution can not be produced. This is described further in Section 5.2.

Constraints can also be accounted for by using the penalty method. The penalty method can be used to convert a constrained optimization problem into an unconstrained optimization problem, simplifying the solution while still giving the same solutions. This is done by adding a penalty term to the objective function (Viswanathan, 1990). When the constraint is not violated the penalty term is equal to zero, and when the constraint is violated the penalty term becomes an arbitrarily large value so that any infeasible solutions will not be picked as the best solution. In this project, the constraint of having no peroxide groups present was accounted for using the penalty method. A penalty term was added to the objective function, counting the number of peroxide groups present and adding a thousand to the objective function for each. Good objective functions in this project are less than one, so a molecule with a peroxide group present will never be presented as a candidate molecule. This technique is described further in Section 5.2.

The design problem can be solved using either deterministic or stochastic search algorithms. A deterministic method aims to find a global minimum to the objective function, and does this by determining what the next candidate solution is by examining the current solution. It acts predictably, so that with the same initial solution the algorithm will always take the same route to the same final solution (Horst, 1996). A simple example of a deterministic method for this type of combinatorial optimization problem is Branch-and-Bound. Deterministic methods have been successfully used to solve molecular design problems previously (Sahinidis, 2004; Maranas, 1996). A stochastic method uses random elements in the algorithm, and aims to find good near-optimal solutions, which will not necessarily be the global optimum. Deterministic

methods have many problems when solving large design problems. Finding a global optimum can be prohibitively computationally expensive (Lin, 2005). Also, the QSPRs have limited accuracy, so there is no guarantee that the global optimal solution will actually be superior to the near-optimal solutions that a stochastic method would provide. Multiple runs of a stochastic method will result in a list of different near-optimal solutions. This allows the use of other criteria, such as cost or ease of synthesis, to help rank the final candidate molecules.

An example of a stochastic method that has been used in molecular design is the genetic algorithm. Genetic algorithms have been used to design linear polymers (Venkatasubramanian, 1994), model proteins (Konig, 1999), and are used extensively outside of molecular design (Jeon, 2010; Layric, 2005). Genetic algorithms mimic natural evolution by allowing the best known solutions to breed with each other, resulting in offspring solutions which should have solutions superior to the parent solutions. Candidate solutions need to be described in strings, called chromosomes. At each generation the most fit solutions are stochastically selected to breed, being combined and possibly introducing mutations, creating a new generation of solutions. The least fit solutions are abandoned, mimicking natural selection (Banzhaf, 1998; Goldberg, 1989). This is repeated until a satisfactory solution is found.

Another stochastic method which has been used more recently for molecular design is the Tabu Search algorithm. Tabu Search has been used to design catalysts (Lin, 2005), crosslinked polymers (Eslick, 2009), has been used to solve the traveling salesman

problem (Knox, 1994), as well as many other applications. The Tabu Search algorithm relies on a memory of previously visited solutions to avoid revisiting areas of the solution space that have already been explored.

Tabu Search starts with an initial solution. At each iteration, the algorithm can make a specified number of moves away from the current solution. These moves correspond to changing atoms or groups in the molecule. Solutions that can be reached within this specified number of moves make up the neighbors of the current solution. These moves are stochastically chosen, and a subset of neighbors are evaluated. After a possible solution is evaluated it is added to the Tabu list, and solutions on the Tabu list will not be revisited. The neighbor with the lowest objective function is chosen as the new current solution, and the next iteration begins. The inclusion of the Tabu list guarantees that previous solutions will not be revisited, which could occur if it is a local minima, saving calculation time. The Tabu list also encourages searching in more diverse areas (Eslick, 2008). The algorithm is continued until a stop criteria is reached, possibly after a set number of non-improving iterations.

The length of the Tabu list is limited to reduce computation and memory usage, and to allow solutions to be revisited if the search is proceeding in a different direction (de Werra, 1989).

Many additions can be made to the basic Tabu Search algorithm. One is the use of long-term memory to store a list of good previous solutions, highlighting areas of the solution

space that might have better solutions that have not been found. The algorithm can then revisit these areas. This is called intensification (Glover, 1990). Local intensification can be used by limiting the number of moves the algorithm makes when already at a good solution. This forces the algorithm to look more thoroughly around areas where a near-optimal solution may exist. Diversification can be used by rerunning the algorithm at a different starting point, allowing the algorithm to explore parts of the solution space that have not been evaluated (Glover, 1990).

There are many adjustable parameters in Tabu Search, such as the length of the Tabu list, the number of moves, and the size of the subset of neighbors being evaluated. The value of these parameters can make a substantial difference to the quality of solutions found, or the computation time needed to find the solutions. The optimal values of these parameters depend on the size and type of design problem being solved. For example, in this project using a larger possible step size of 8 improved the average objective function significantly compared to a smaller step size of 2.

The rest of this thesis describes how this specific project was implemented. The following Chapter describes the molecular descriptors used and how they were calculated.

Chapter 3.

Calculating Descriptors

Methods used in calculating the molecular descriptors in Polymer Designer are described in this section. The main focus in this project was the use of connectivity index, which require a path finding algorithm. Connectivity index have been successfully used to create QSPRs for polymer systems before (Bicerano, 2002). Polymer Designer uses subgraph isomorphism to find chemical substructures within the monomer or polymer, which can be used in group contribution techniques (Eslick, 2009). Methods for calculating 100% crosslink density, number of rotational degrees of freedom, and molecular weight are also discussed in this Chapter.

3.1 Group Contribution and Subgraph Isomorphism

The subgraph isomorphism algorithm (Ullmann, 1976) is used to identify the molecular substructures for the group contribution method. It is also used in other descriptor calculations to find functional groups, such as number of vinyl groups for calculating crosslink density (Eslick, 2009).

A subgraph is a graph that is contained within a larger graph. Two graphs *G* and *H* are isomorphic if you can apply a bijection to the vertex sets

$$f : V(G) \rightarrow V(H)$$

such that an edge connecting vertices *u* and *v* in *G* exists only if an edge connecting vertices *f(u)* and *f(v)* exists. More generally, if the only difference between two graphs are the names of the vertices and spatial placement, then they are isomorphic (West, 2001). Figure 3.1 shows two isomorphic graphs, to show that it is not immediately obvious when two graphs are isomorphic.

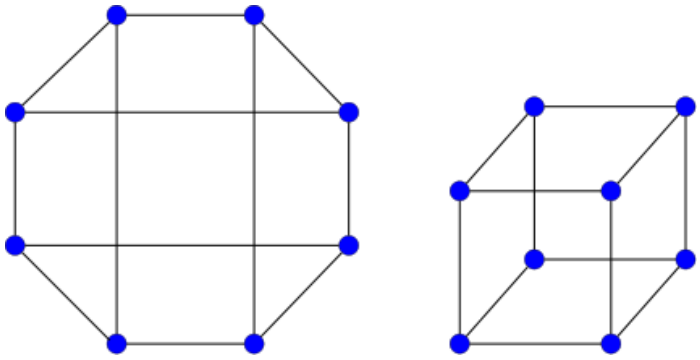


Figure 3.1 - Different representations of a cube graph (Aspnes, 2010)

The subgraph isomorphism algorithm is used to find how many subgraphs exist of a certain functional group or group contribution substructures within the monomer graph. Finding double-bonded oxygens in the molecular graph of HEMA is used as an example. These graphs are shown in Figure 3.2 and Figure 3.3.

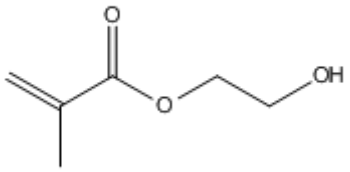


Figure 3.2 – HEMA Graph

Figure 3.3 - Double-bonded oxygen subgraph. The atom labeled as '1' is a dummy atom

The original implementation of the algorithm by Ullman (1976) was not made with molecular graphs in mind. Atoms and bond types had to be added so that the hydroxide or double-bonded carbon would not be found by the algorithm.

In group contribution, first-order groups can not overlap. The algorithm was modified so that vertices can be labeled as already being within a subgraph so that atoms will not be included in more than one substructure. For second and third-order groups, this is not necessary as they can overlap (Marrero, 2001).

3.1.1 Connectivity Indices and Path Finding

The calculation of connectivity indices uses a path finding algorithm. For example, the third order connectivity index (χ^3) needs a list of all paths of length three. The path finding algorithm used in this project is a breadth first search (West, 2001). The

algorithm builds a path tree starting from a root vertex. The algorithm records all vertices that are one edge from the root, or all the paths of length one. The algorithm continues by finding the vertices adjacent to each of these vertices, and so on. A vertex is not counted twice if a cycle exists. The following figure gives an example of a path tree where each number is the ID of the atom. The path finding algorithm is repeated for all atoms as the root vertex. This will find each path twice; backwards and forwards. This was fixed by only allowing paths where the ID number of the head vertex is larger than that of the tail vertex (Eslick, 2009).

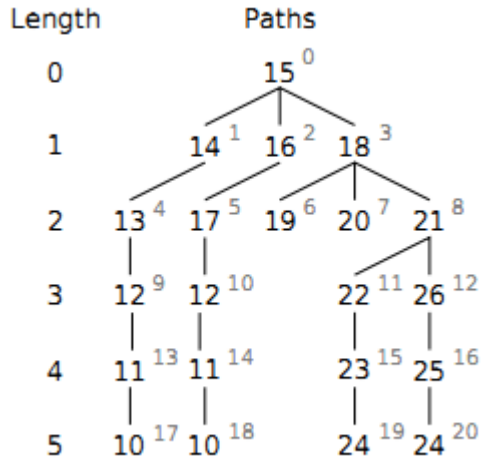


Figure 3.4 - Path tree from the breadth first search (Eslick, 2009)

The simple (δ) and valency (δ^v) atomic connectivity index need to be calculated. The atomic connectivity index were pre-calculated for each type of atom needed in this project. The algorithm looks at the atom's hybridization and number of implied hydrogen atoms and assigns atomic connectivity index using an if-then-else statement. The connectivity indices are then calculated using the following equations (Bicerano, 2002).

$$ {}^{n}\chi = \sum_{k \in \text{n-length paths}} \frac{1}{\sqrt{\prod_{i \in \text{atoms in k}} \delta_i}} $$

$$ {}^{n}\chi^{v} = \sum_{k \in \text{n-length paths}} \frac{1}{\sqrt{\prod_{i \in \text{atoms in k}} \delta_i^{v}}} $$

The connectivity indices are size-dependent descriptors, or extrinsic. Some properties may correlate better with a size-independent, or intrinsic, descriptor, so a weighted connectivity index is calculated using the following equation

$$ \xi^{n} = \frac{\chi^{n}}{N} $$

where N is the number of non-hydrogen atoms.

The connectivity index can be calculated for either the single monomer or a representative piece of the polymer. Some of the paths will extend into the buffer region. When this occurs, only a fraction of the path's value should be added to the connectivity index. This is done using the equation

$$ {}^{n}\chi = \sum_{k \in \text{n-length paths}} \frac{n_{core}}{n} \frac{1}{\sqrt{\prod_{i \in \text{atoms in k}} \delta_i}} $$

where n_{core} is the number of atoms in the path that are in the core region (Eslick, 2008).

A mole average connectivity index can also be easily calculated. The connectivity index for HEMA and BisGMA are pre-calculated. The path finding algorithm only has to be used on the test monomer, instead of the entire crosslinked polymer. When Tabu Search

is being employed, possibly thousands of large polymer graphs have to be created. This can be very computationally expensive. Correlations using less computationally expensive descriptors should be chosen if they perform as well as those with the more expensive descriptors.

3.1.2 100% Crosslink Density

The 100% crosslink density is the maximum number or crosslinks per repeat unit if the monomers are randomly crosslinked. It is found using the following equation

$$CD_{100} = \sum_i x_i \left(n_{v,i} - 1\right)$$

where n_v is the number of vinyl groups of monomer i, and x_i is the mole fraction of monomer i (Eslick, 2009). This is the crosslink density if every double bond in a vinyl group is broken and become part of the backbone. This is unlikely to occur physically, though processing conditions can be altered and candidate monomers can be chosen to increase degree of polymerization.

The number of vinyl groups can be found using the subgraph isomorphism algorithm described previously. However, the number of vinyl groups is normally prespecified prior to the design phase. This limits the size of the design space, and therefore reduces computation time.

3.1.3 Molecular Weight

The molecular weights for each atom are stored in a database. Since the molecular graphs stored are hydrogen suppressed, the number of hydrogen atoms needs to be calculated. This is done by examining the hybridization of each atom and its vertex degree to see if any hydrogen atoms are bonded to that atom. The atomic weights are then summed.

3.1.4 Rotational Degrees of Freedom

The number of rotational degrees of freedom, N_{rot}, is used in the correlation for the glass transition temperature. Bicerano (1996) found that the glass transition temperature for randomly crosslinked polymers correlated well with how flexible the monomer is. For the polymers used in this project, N_{rot} is equal to the number of single bonds not in a ring plus the number of vinyl groups. The subgraph isomorphism algorithm is used to count the number of single bonds. Bonds in a ring can be labeled as being aromatic so that they are not counted as single bonds by the algorithm.

With the experimental data collected and the molecular descriptors calculated the QSPRs can be correlated. Chapter 4 summarizes the correlation results.

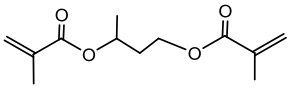
Chapter 4.

Development of QSPRs

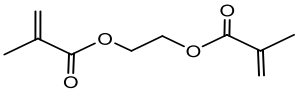
This section provides a summary of the QSPRs that were developed for this project. Also described is how each model was chosen over other prospective models.

4.1.1 Physical and Chemical Properties

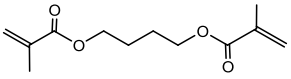
Properties of a set of methacrylate polymers were collected experimentally. A set of fifteen methacrylate test monomers were tested at a range of concentrations. The dental polymers were made from a mixture of the test monomer, the methacrylate HEMA, and the methacrylate BisGMA. HEMA and BisGMA are commonly used in dental polymers (Ye, 2009). The concentrations tested were 25, 35, 45, and 55 weight percent test monomer, each time with 45 weight percent HEMA and the balance BisGMA. Figures 4.1 and 4.2 show the test monomers used in this project.



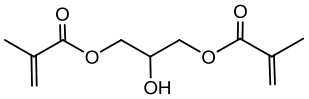
1,3-Butanediol dimethacrylate



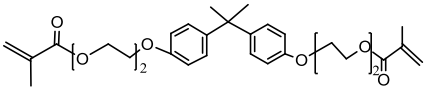
Ethylene glycol dimethacrylate



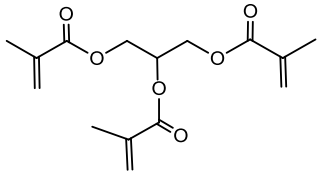
1,4-Butanediol dimethacrylate



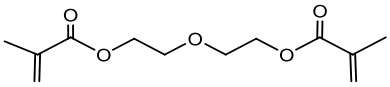
1,3-Glycerol dimethacrylate



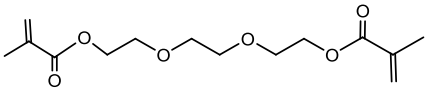
Bisphenol A ethoxylated dimethacrylate



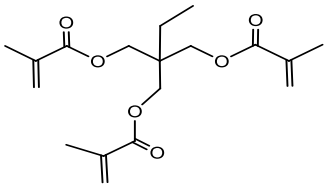
Glyceryl trimethacrylate



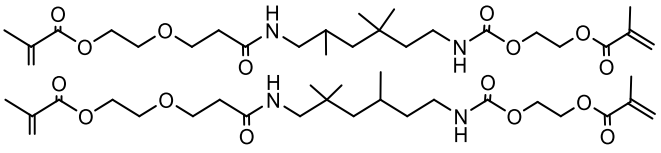
Diethyleneglycol dimethacrylate



Triethylene glycol dimethacrylate

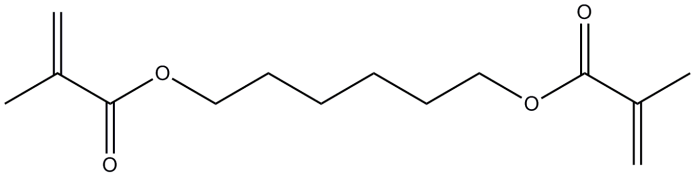


Trimethylolpropane trimethacrylate

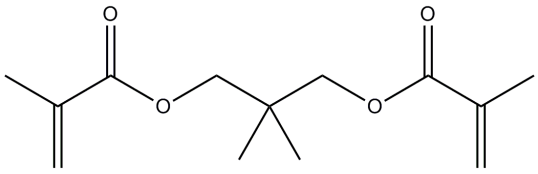


Urethane dimethacrylate

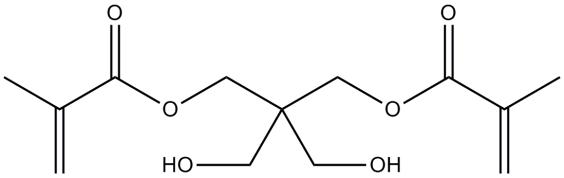
Figure 4.1 – Test monomers.



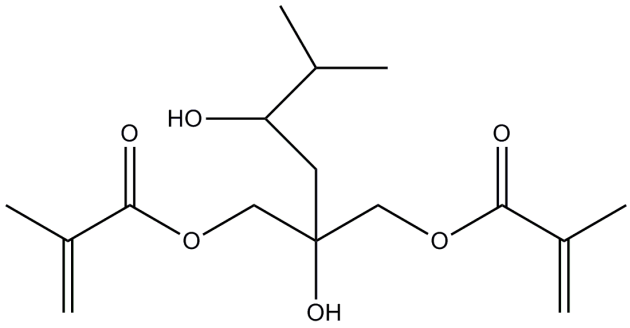
1,6-hexanediol dimethacrylate



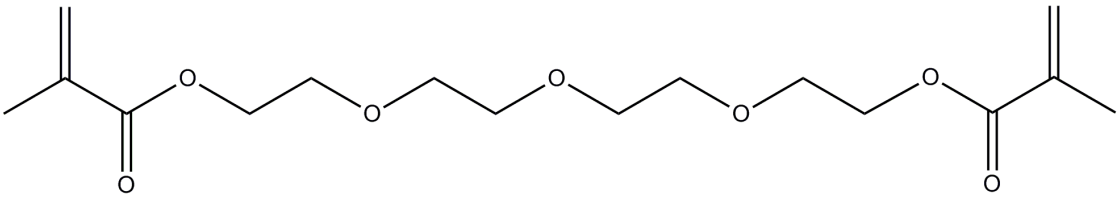
Neopentyl glycol dimethacrylate



Pentraerythritol dimethacrylate



Pentaerythritol trimethacrylate



Tetraethylene glycol dimethacrylate

Figure 4.2 – Additional test monomers.

Correlations were made for viscosity, storage modulus, rubbery modulus, percent water sorption, percent water solubility, and glass transition temperature. The storage modulus, rubbery modulus, and glass transition temperature would be heightened in an ideal dental polymer, while water sorption and solubility for the polymerized material would be lowered (Park, 2009; Fried, 2003; Bosze, 2006). The range in property data is limited, and extrapolating results outside the initial data set would result in large errors. Because of this, the target values for these properties were set to near the high or low end of the experimental values. Low viscosity resins are desired (Spencer, 2010), but choosing a viscosity value too low may make the resin difficult to handle or collect on the surface of the tooth. A median value of viscosity was chosen.

4.1.2 Model Selection and Statistical Analysis

The R statistics program (R, 2007) was used to create the correlations using multiple linear regression. The descriptor selection package, LEAPS, examines all combinations of descriptors up to a certain size using a branch-and-bound method (Lumley, 2004). LEAPS provides the best subset of descriptors provided for the prediction of the property, along with a value of Mallows' *Cp* and r^2 for each model.

The choice of model size is first determined using Mallows' *Cp*. The first model examined is the one with the smallest Mallows' *Cp*. The purpose of minimizing Mallows' *Cp* is to lower variance while not increasing bias too much. However, no single model

selection technique is perfect, and sometimes Mallows' *Cp* is too sensitive towards increasing bias. If a much smaller model gives a good r^2 value, then it may be best to ignore Mallows' *Cp* and choose the smaller model. The statistical significance of each descriptor is then calculated. If any descriptor does not pass the 5% level of significance the model is rejected and the next best model is examined.

Once the final model is chosen for each property the confidence interval is calculated for the observations used in making the model. This gives another view of how accurate each individual QSPR is.

The descriptors used in these correlations are summarized in Table 4.1.

Table 4.1 – Molecular descriptors used in creating correlations.

Molecular Descriptors	
χ_{avg}^{n}	Average nth-order simple connectivity index
$\chi_{avg}^{v,n}$	Average nth-order valence connectivity index
χ_{x}^{n}	Nth-order simple connectivity index of test monomer
$\chi_{x}^{v,n}$	Nth-order valence connectivity index of test monomer
ξ_{avg}^{n}	Average weighted nth-order simple connectivity index
$\xi_{avg}^{v,n}$	Average weighted nth-order valence connectivity index
ξ_{x}^{n}	Nth-order weighted simple connectivity index of test monomer
$\xi_{x}^{v,n}$	Nth-order weighted valence connectivity index of test monomer
CD_{100}	Crosslink density of fully crosslinked polymer
MW_{avg}	Mole average molecular weight of comonomer

MW_x	Molecular weight of test monomer
N_{rot}	Number of rotational degrees of freedom

4.1.3 Viscosity

The values of Mallows' *Cp* and r^2 for the viscosity correlations are given in Table 4.1. This is an example of Mallows' *Cp* being sensitive to increasing bias. The fifteen descriptor model had low significance of some coefficients. The five descriptor model was chosen because of its high significance and adequate r^2.

Table 4.2 – Statistical results for viscosity prediction models. Red highlighted cells represent a model which was rejected. The green highlighted cells represent the selected model.

Viscosity Model		
#	Mallows' Cp	R^2
1	936	0.68
2	654	0.78
3	480	0.83
4	297	0.89
5	144	0.94
6	105	0.96
7	96.4	0.96
8	72.8	0.97
9	65.0	0.97
10	57.0	0.97
11	42.9	0.98
12	39.5	0.98
13	29.1	0.99
14	19.9	0.99
15	16.6	0.99
16	17.7	0.99

Multiple linear regression gives the following model.

$$\mu = (.119) + (-0.0935)\chi_x^1 + (0.101)\chi_{avg}^2 + (-0.217)\chi_{avg}^{v,0} + (0.246)\chi_{avg}^{v,1} + (0.00354)MW_x$$

Figure 4.3 shows the predicted viscosity versus the experimental viscosity and includes the 95% confidence intervals. The 45 degree line does not represent the model, and is only shown to aid the reader. Data points on the 45 degree line represent data points where the experimental value is exactly equal to the predicted value. Ideally the 95% confidence values would overlap the 45 degree line for all points. The points that do not overlap could be due to additional experimental error or the models could not adequately describe that particular monomer.

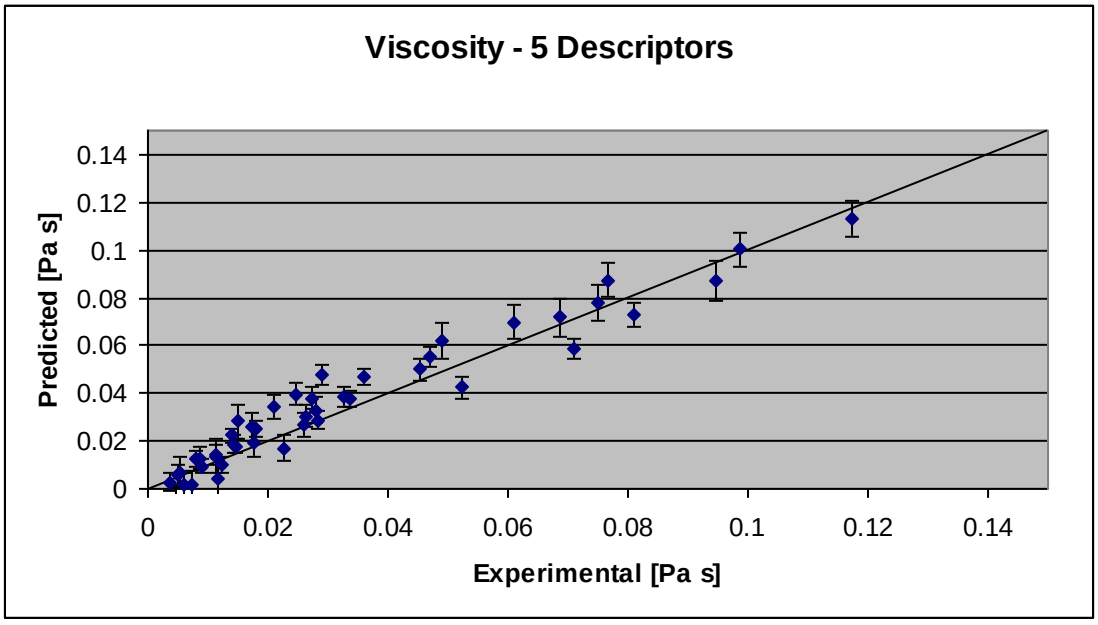


Figure 4.3 – Confidence interval for viscosity

4.1.4 Percent Water Sorption

The values of Mallows' *Cp* and r^2 for the water sorption correlations are given in Table 4.2.

Table 4.3 - Statistical results for water sorption prediction models. Red highlighted cells represent a model which was rejected. The green highlighted cells represent the selected model.

Water Sorption Model		
#	Mallows' *Cp*	r^2
1	1494.06	0.13
2	1146.81	0.33
3	882.98	0.48
4	459.77	0.72
5	348.82	0.78
6	264.25	0.83
7	233.13	0.85
8	197.92	0.87
9	162.58	0.89
10	107.49	0.93
11	60.45	0.95
12	16.19	0.98
13	12.9	0.98
14	11.64	0.98
15	12.03	0.99

The fourteen descriptor model had low significance for many of its descriptors. The ten descriptor model was selected because of its high significance and adequate r^2.

$$W_{SP} = 21180 + (94.82)\chi^{0}_{avg} + (76.41)\chi^{1}_{avg} + (-224.36)\chi^{2}_{avg} + (-80.82)\chi^{v,0}_{avg} + (-23.85)\chi^{v,1}_{avg}$$

$$+ (159.66)\chi^{v,2}_{avg} + (-11766.54)\xi^{0}_{avg} + (-26125.48)\xi^{1}_{avg} + (1447.66)\xi^{v,0}_{avg} + (-2630.15)\xi^{v,2}_{avg}$$

Figure 4.4 shows the predicted water sorption versus the experimental water sorption and includes the 95% confidence intervals. The 95% confidence intervals do not overlap the 45 degree line for only a few of the data points.

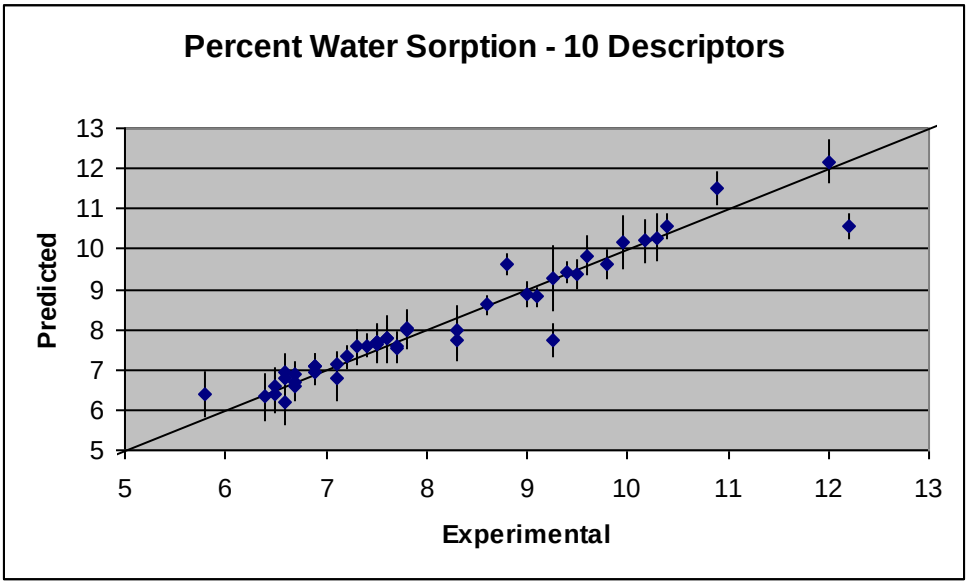


Figure 4.4 - 95% Confidence interval for percent water sorption

4.1.5 Glass Transition Temperature

The degree of crosslinking greatly affects the glass transition temperature. Crosslinking restricts the movement of polymer chains, raising the amount of thermal energy needed for Brownian motion to occur (Fried, 2003). Multiple linear regression of the glass

transition temperature data proved difficult, and research in the literature suggested a non-linear correlation would be necessary (Bicerano, 1996; Schneider, 1999; Bicerano, 2002).

Bicerano (1996) gives a correlation between crosslinked and uncrosslinked glass transition temperature for randomly crosslinked high polymers.

$$T_g(n) = \left(1 + \frac{5}{n \cdot N_{rot}}\right) T_g(\infty)$$

In this correlation, *n* is the molecular weight in between crosslinks, which is the reciprocal of our definition of crosslink density. The number of rotational degrees of freedom, N_{rot}, can be defined for all types of polymers (Bicerano, 1996). However for our purposes, with these monomers, it is simply equal to the number of single bonds that are not in a cycle, plus the number of vinyl groups for crosslinking. Since the actual crosslink density is a function of processing conditions, the 100% crosslink density was used in this expression.

$$T_g = \left(1 + \frac{5CD_{100}}{N_{rot}}\right) T_g(\infty)$$

A nonlinear transformation of the glass transition temperature experimental data was performed to create a nonlinear model using multiple linear regression. The resulting correlation replaced the $T_g(\infty)$ term in the QSPR.

The values of Mallows' *Cp* and r^2 for the glass transition temperature are given in Table 4.3. The ten descriptor model was chosen because it had the lowest Mallows' *Cp* value, good significance for each parameter, and had an adequate r^2.

Table 4.4 - Statistical results for glass transition temperature prediction models. The green highlighted cells represent the selected model.

Glass Transition		
#	Mallows' *Cp*	r^2
1	64.5	0.48
2	35.8	0.64
3	23.2	0.71
4	13.6	0.77
5	9.9	0.80
6	8.4	0.82
7	2.9	0.86
8	2.1	0.87
9	1.2	0.89
10	-0.2	0.90
11	0.7	0.91

Multiple linear regression gave the following model.

$$T_g = \left(1 + \frac{5CD_{100}}{N_{rot}}\right)\left((-389.6) + (-38.2)\chi_x^2 + (26.9)\chi_x^3 + (21.2)\chi_x^{v,0} + (-32.6)\chi_x^{v,3}\right.$$

$$\left. + (245.6)\chi_{avg}^0 + (-115.3)\chi_{avg}^3 + (-9.0)MW_{wted} + (-662.3)\chi_{avg}^{v,0} + (-2.1)MW_{avg} + (-190.9)CD_{100}\right)$$

Figure 4.5 shows the predicted glass transition temperature versus the experimental glass transition temperature and includes the 95% confidence intervals.

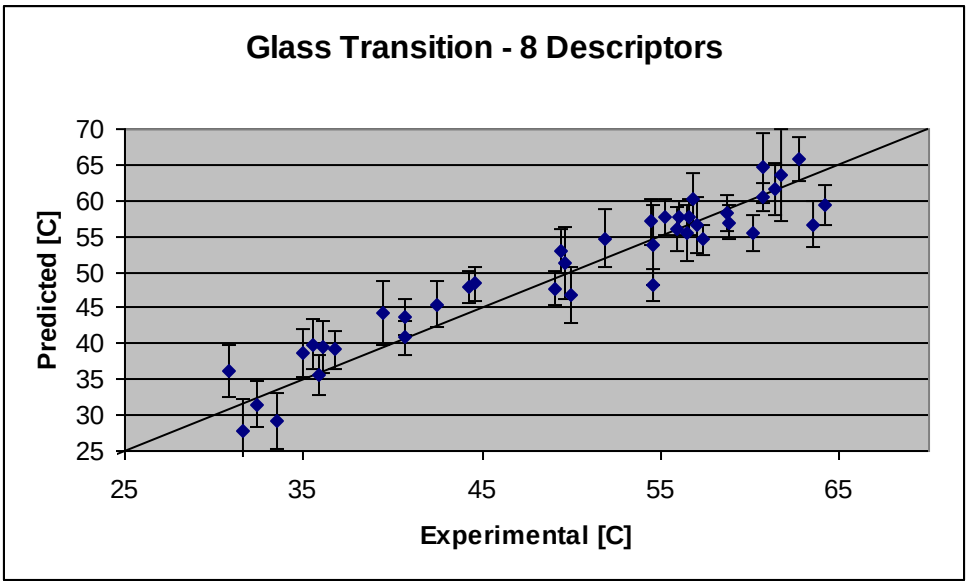


Figure 4.5 - 95% confidence interval for glass transition temperature

4.1.6 Storage Modulus

When examining models for the storage modulus the intercept tended to not pass the 5% significance level. A model without an intercept was found. The values of Mallows' *Cp* and r^2 for the storage modulus correlations are given in Table 4.4. The four descriptor model was chosen because it has the lowest Mallows' *Cp* value and had high significance.

Table 4.5 - Statistical results for storage modulus prediction models. The green highlighted cells represent the selected model.

Storage Modulus		
#	Mallows' *Cp*	r^2
2	3.08	0.45
3	0.96	0.63
4	-0.11	0.70
5	1.58	0.75

Regression gave the following model.

$$E' = (490.05)\chi_x^2 + (-367.81)\chi_x^{v,0} + (388859.72)\frac{CD_{100}}{MW_{avg}} + (292.17)N_{rot}$$

Figure 4.6 shows the predicted storage modulus versus the experimental storage modulus and includes the 95% confidence intervals. Less experimental data was collected for storage modulus than other properties.

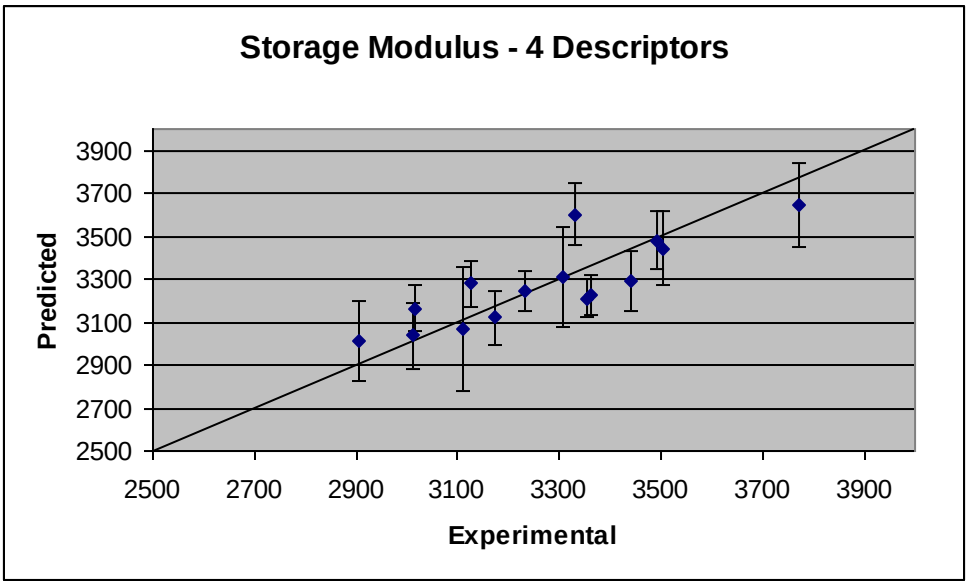


Figure 4.6 - 95% confidence interval for storage modulus

4.1.7 Rubbery Modulus

The same as with the storage modulus, when examining models for the rubbery modulus the intercept tended to not pass the 5% significance level. A model without an intercept was found. The values of Mallows' *Cp* and r^2 for the rubbery modulus correlations are given in Table 4.5. The three descriptor model was chosen because it had the lowest Mallows' *Cp*, good significance, and adequate r^2.

Table 4.6 - Statistical results for rubbery modulus prediction models. The green highlighted cells represent the selected model.

Rubbery Modulus		
#	Mallows' *Cp*	r^2
2	5.3	0.83

3	1.59	0.88
4	2.89	0.91
5	4.01	0.91

Multiple linear regression gives the following correlation.

$$E_r = (110.27)CD_{100} + (67.75)\chi^1_{avg} + (-4.624)MW_{avg}$$

Figure 4.7 shows the predicted rubbery modulus versus the experimental rubbery modulus and includes the 95% confidence intervals. Less experimental data was collected for rubbery modulus than other properties. All of the confidence intervals overlap the 45 degree line for this model, which is ideal.

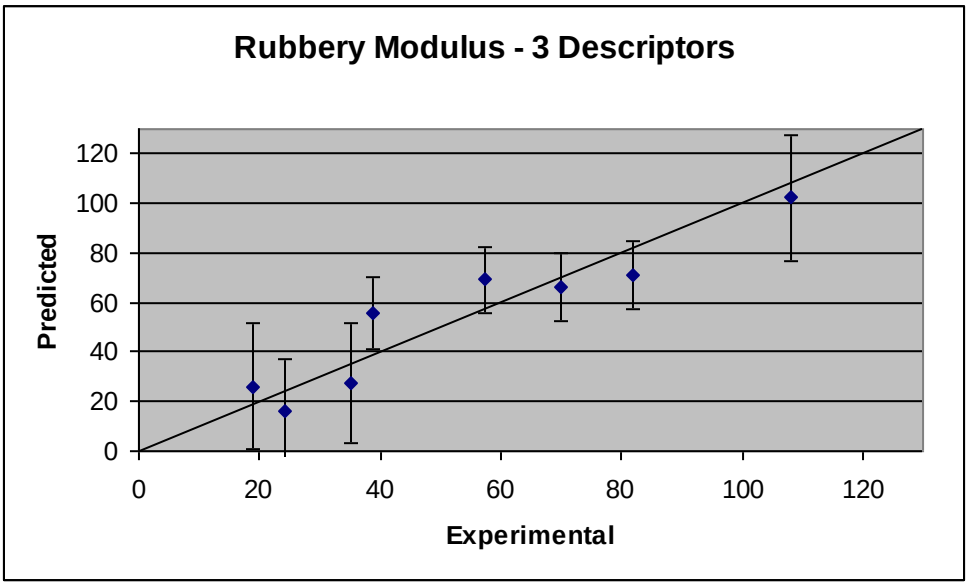


Figure 4.7 - 95% confidence interval for rubbery modulus

4.1.8 Solubility

The values of Mallows' *Cp* and r^2 for the percent solubility correlations are given in Table 4.6. The seven descriptor model was chosen because it had the lowest Mallows' *Cp*, passed the 5% significance level, and had an adequate r^2.

Table 4.7 - Statistical results for solubility prediction models. The green highlighted cells represent the selected model.

Percent Solubility		
#	Mallows' *Cp*	r^2
1	43.95	.094
2	18.83	.541
3	10.96	.704
4	7.81	.789
5	9.26	.798
6	5.98	.885
7	4.19	.947
8	6.02	.950

Multiple linear regression gives the following correlation.

$$W_{SU} = (12.666) + (24.307)\chi^{0}_{avg} + (-54.80)\chi^{3}_{avg} + (48.20)\chi^{v,3}_{avg} + (0.294)\chi^{v,0}_{avg} + (-77.14)MW_x + (0.294)\chi^{v,0}_{avg} + (-77.14)MW_x + (-14.41)N_{aro} + (-6.94)\chi^{v,0}_{x}$$

Figure 4.8 shows the predicted solubility versus the experimental solubility and includes the 95% confidence intervals. All of the 95% confidence intervals overlap the 45 degree line, which is ideal.

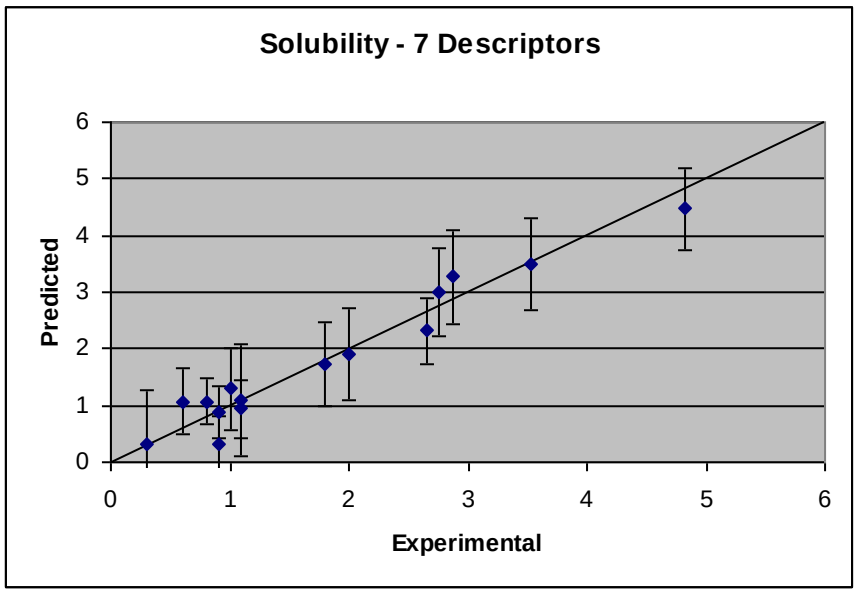


Figure 4.8 - 95% confidence interval for percent solubility

4.1.9 Summary

Table 4.8 summarizes the correlation results. Most of the QSPRs have strong correlations greater than or approaching a correlation coefficient of 0.90. The correlation for storage modulus can be improved through collecting more experimental data. Also other types of descriptors can be considered beyond connectivity index. Few QSPRs have been correlated for crosslinked methacrylates, and were correlated with a smaller set of experimental data (Eslick, 2009).

With the experimental data correlated the backwards design problem can be solved. Section 5 describes how the molecular design problem was formulated and solved in this project.

Table 4.8 - Summary of QSPR results

Property	Number of Descriptors	R^2
Glass Transition Temperature	8	0.91
Percent Water Sorption	10	0.93
Percent Solubility	7	0.95
Storage Modulus	4	0.70
Rubbery Modulus	3	0.88
Viscosity	5	0.94

Chapter 5.

Molecular Design

This section describes the design problem formulation and implementation of molecular design using the Tabu Search algorithm. Section 5.1 gives a description of how the problem would be solved using any type of CMD, while Section 5.2 gives details on how the problem was solved using Tabu Search.

5.1.1 Problem Formulation

This project seeks to design a methacrylate monomer for the use in dental resin composites. The goal is to find a monomer that will lead to resin composites that are more durable that those currently on the market. Target values for important physical properties were selected that would give an increased lifespan of the composite.

Table 5.1- Target property values

Property	Target Value
Storage Modulus [MPa]	3500
Rubbery Modulus [MPa]	40
Water Sorption [%]	6
Viscosity [Pa s]	0.1
Glass Transition Temperature [C]	74

Target values could only be chosen that are within the range of the experimental values used in the development of the QSPRs. Large values of storage and rubbery modulus were chosen because the value correlates with a high tensile strength (Bosze, 2006). Water sorption was minimized because the absorption of water can lower the mechanical properties of the monomer (Park, 2009). A median viscosity was chosen which was lower than the standard. Lower viscosity values allow the resin to bond more tightly to the tooth while the resin is curing (Spencer, 2010).

The properties were measured experimentally at concentrations of 55, 45, 35, and 25 weight percent test monomer, each time with 45 weight percent HEMA and the balance BisGMA. The CMD methodology was applied at each of these concentrations. It would be expected that one candidate monomer may perform well at one concentration but poorly at another. Future versions of the program could allow the evaluation of each candidate monomer at every concentration, but this added complexity could make the

optimization computationally expensive. Candidate monomers for each concentration are presented in Section 6.

The candidate monomers are built as combinations of a set of different functional groups. Each monomer is represented as an oligomer molecular graph in the program. The functional groups were chosen by considering all groups in the monomers used to make the QSPRs.

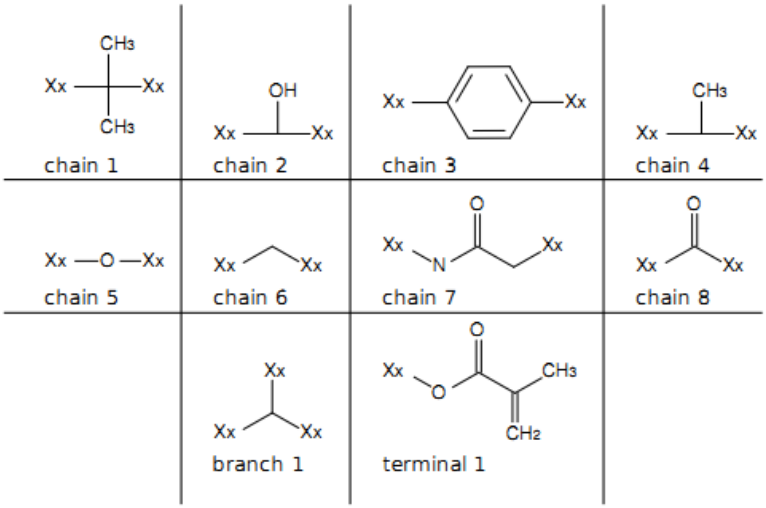


Figure 5.1 - Functional groups. Xx represent dummy atoms (Eslick, 2008).

Each candidate monomer was forced to have two methacrylate groups. Peroxide groups were not allowed, to avoid unstable molecules. Further stability criterion can be added to future versions of the CMD method. Feasibility criteria also have to be met; valency must be satisfied, and the molecular structure must be connected.

The goal of this project is to design the molecular structure of a polymer with desired properties. The objective function has the form

$$f = \sum_{i \in \text{properties}} s_i \left(\frac{P_{i,\text{target}} - P_{i,\text{predicted}}}{P_{i,\text{target}}} \right)^2$$

where f is the objective function, $P_{i,target}$ is the target value for property i, $P_{i,predicted}$ is the predicted property, and s_i is a weighting factor. Weighting factors of 1 were used for each property because no data was available regarding the amount each property affected the lifespan of the dental polymer. The objective function is zero when the predicted properties match the target values.

The Tabu Search algorithm is used to find a solution that minimizes the objective function. The following section describes how Tabu Search is implemented.

5.1.2 Tabu Search

Figure 5.2 describes the Tabu Search algorithm. All monomers considered during the optimization phase are created from predefined functional groups. The monomers are represented as an oligomer molecular graph. Each functional group is represented by a vertex, and bonds are represented as edges. Two types of initial solution were evaluated: in some cases, the structure of BisGMA was used to find similar solutions to that structure, while in other cases randomly generated polymer structures were used to explore different parts of the solution space.

During a Tabu Search solution, a set of valid moves are made during each iteration. The valid moves in this project are deletion of functional groups (then adding bonds connecting neighboring functional groups), addition of functional groups in the chain, or changing one functional group for another. These moves were chosen randomly. Feasibility criteria did not need to be included explicitly in the program, as they were implied in the set of legal moves. Each functional group is a segment of a polymer chain with a single bond on each end, so any valid move will not make the molecule infeasible. Methacrylate groups are not changed by the algorithm.

At each iteration, a list of neighbors to the current solution is made. A neighbor is any molecule that is within a set number of moves from the current solution. The number of possible moves chosen was eight in order to overcome the valleys that contain local minima, and explore other parts of the solution space. The most efficient step size may be different for each problem. When the objective function value for the current solution is less than one, the number of moves is set to one. This is a type of local intensification, and is used to focus on areas of the solution space where a good solution may exist. The objective function of each neighbor is examined, and the best non-Tabu solution is chosen to be the next solution. The previous solution is then added to the Tabu list.

The Tabu list is a list of previous solutions that neighbor molecules are compared to. If a neighbor molecule is too similar to any molecule on the Tabu list it is labeled as Tabu and will not be selected as a new solution. The usefulness of the Tabu list lies in the ability to avoid revisiting previous solutions, or to keep the algorithm from being stuck in

a local minima. If all neighbors are labeled as Tabu, then the best neighbor is chosen. If this occurs too frequently the Tabu criteria are too strict and are relaxed.

Molecules were said to be too similar if all connectivity indices used in this project (${}^0\chi, {}^1\chi, {}^2\chi, {}^3\chi, {}^0\chi^v, {}^1\chi^v, {}^2\chi^v, {}^3\chi^v$) lie within 15% of the previous solution range. For example, if the range of observed values for ${}^0\chi$ is 11.1-28.7, then if the values of ${}^0\chi$ are within 2.6 of each other the molecules are too similar. Even if a molecule is labeled as Tabu, it will still be chosen if its objective function is better than the best solution found so far. This is a type of aspiration criteria.

At each iteration, the objective function is calculated for each neighbor solution. The general form of the objective function is used, with the addition of a penalty function. The penalty function is used to avoid unstable solutions that contain peroxide groups. The number of peroxide groups is set as a descriptor variable, calculated using the subgraph isomorphism algorithm described in Section 3. A penalty function is added to the objective function so that 1000 is added to the objective for each peroxide group present. Good objective function values in this project are less than one, so no solution with a peroxide group will be presented as a candidate.

The algorithm continues until a stop criterion is met. In this project, this limit is set to 400 non-improving iterations. Numerous test runs of the algorithm showed that optimal solutions were rarely found after more than 400 non-improving iterations, and were frequently found before 200. Once the stop criterion is reached, the program reports the

best known solution. The following Chapter summarizes the results found from using this procedure.

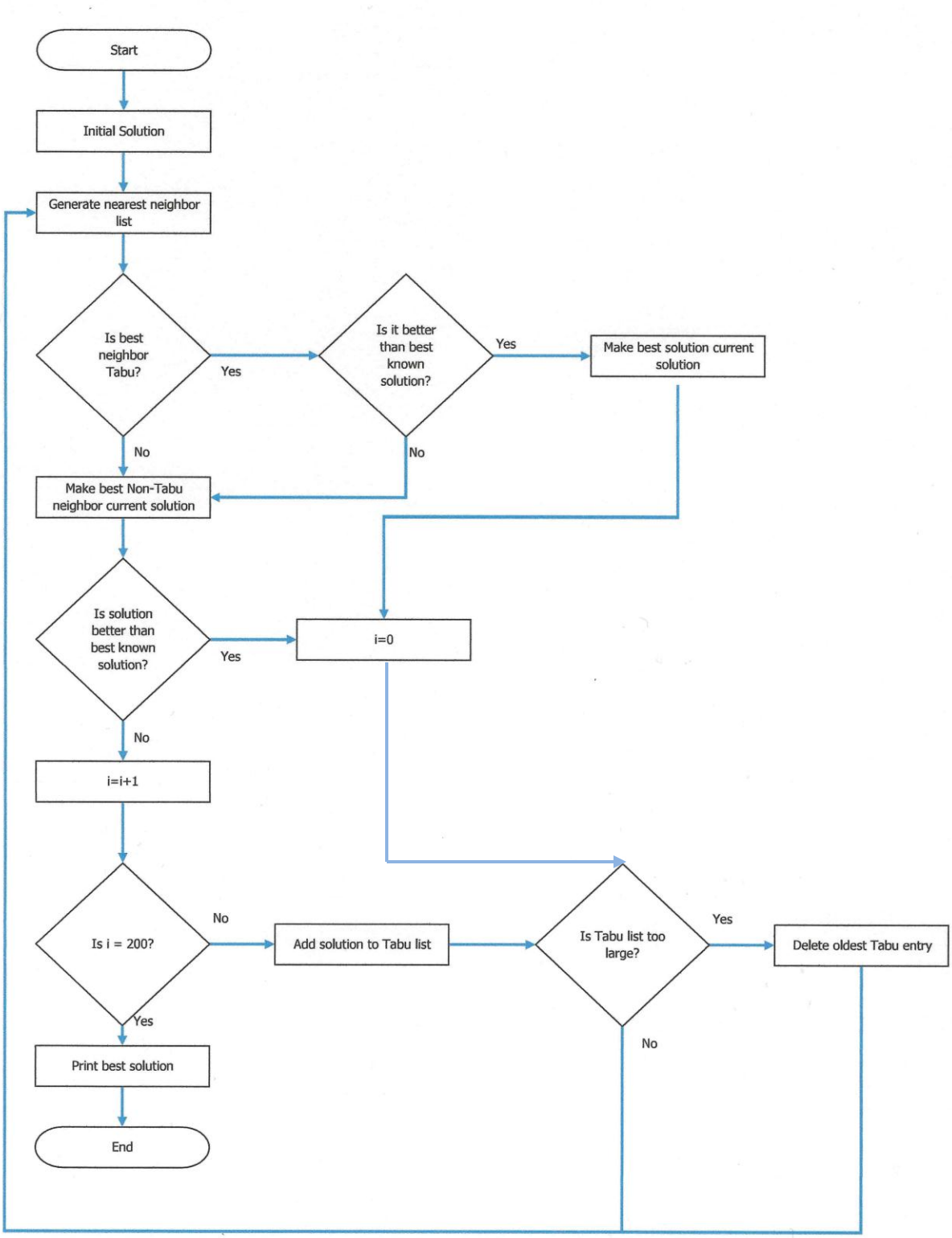


Figure 5.2 - Tabu Search flowchart

Chapter 6.

Results

This section summarizes the results found after completing the Tabu Search optimization procedure described previously. The target values are summarized in Table 5.1, which would yield a dental polymer with improved clinical lifetime to those currently on the market. Included are the overall results for the algorithm, as well as the candidate monomers. The algorithm was run at concentrations of 25, 35, 45, and 55 weight percent of the test monomer, each time with 45 weight percent HEMA, and the balance BisGMA. There were two different starting points: BisGMA, and a randomly generated monomer. The search was terminated after 400 non-improving iterations and took less than a minute to complete for each run.

6.1 Tabu Results

The Tabu Search algorithm was applied multiple times at 25 weight percent test monomer for the improved dental polymer case study in order to judge the overall effectiveness of the algorithm. The objective function and number of iterations were recorded and an average objective function was found.

The Tabu Search algorithm uses a stochastic parameter to define its search direction. One run of the algorithm may be able to find a good solution very quickly, while the next run may only look at an area of the solution space with no good solution and not be able to escape that region. Introducing additional heuristics to the algorithm such as diversification, described in Chapter 2.4, to the algorithm decreases the chances that any single run will give a poor result.

The average objective function and number of iterations for this example are given in the following table. An average objective function of 0.056 shows that any single run of the algorithm would likely give a reasonable result. Some adjustable parameters, such as the stop criteria or the step sizes, were changed to try to improve these results. Increasing the number of non-improving iterations lowered the average objective function and its standard deviation, as expected. However, these changes only lowered the average objective function because more iterations were available to escape the parts of the solution space corresponding to molecules with properties far from the target values. This did not increase the frequency or quality of the very best results, which tended to be found very quickly. Increasing the number of non-improving iterations greatly increases the run time while not greatly improving the quality of the top tiered results. The number of non-improving iterations was limited to 400. This lends itself to the idea that the Tabu Search algorithm may work best when run in parallel, running fewer iterations but in many different parts of the solution space simultaneously. This is addressed further in Chapter 7.

Table 6.1 - Average Tabu Search results for dental polymer case study. The numbers in parenthesis are standard deviations.

Average Tabu Search Results	
Objective Function	0.056 (0.03)
Iterations	660 (220)

6.2 Candidate Monomers

This section describes the candidate monomers for a dental polymer with improved clinical lifetime found at each concentration: 25, 35, 45, and 55 weight percent candidate monomer, given in Figure 6.1 through Figure 6.10. Each polymer also contained 45 weight percent HEMA, with the balance BisGMA. The objective functions and predicted property values are summarized in Table 6.2 through Table 6.9.

Currently, ease of synthesis is not being considered in the formulation. Also, the only consideration for stability is the prohibition of peroxide groups. Further restrictions can be added in future versions of the program to help make stable, easily synthesizable monomers.

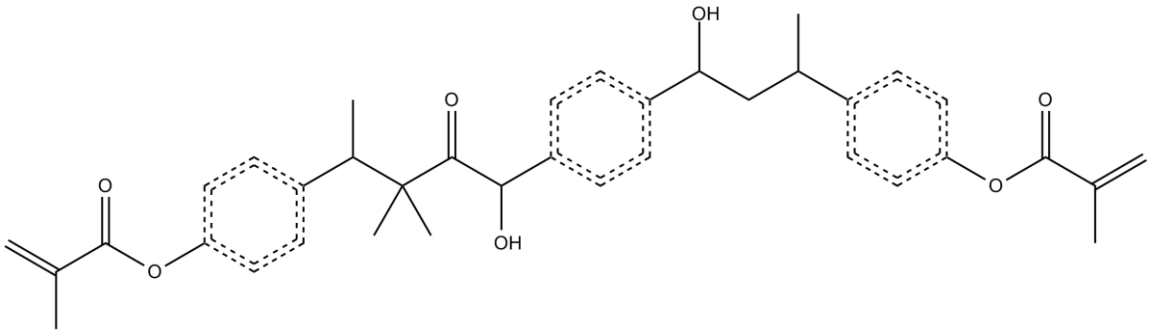


Figure 6.1 - Candidate monomer 25.1. Concentration of 25 weight percent.

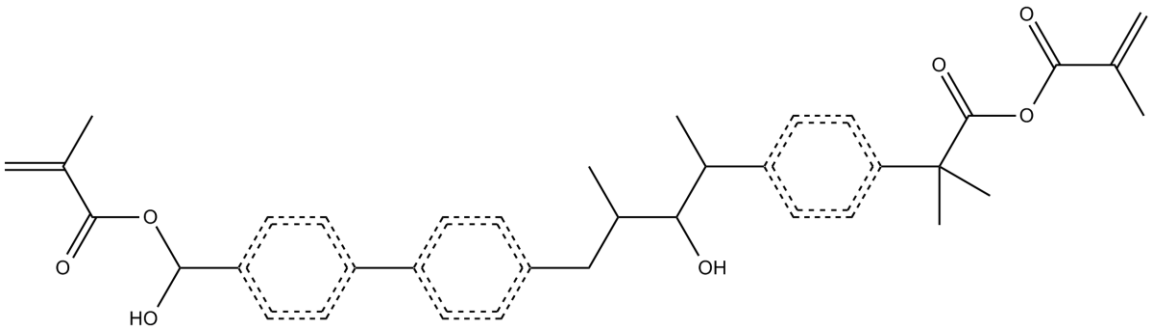


Figure 6.2 - Candidate monomer 25.2. Concentration of 25 weight percent.

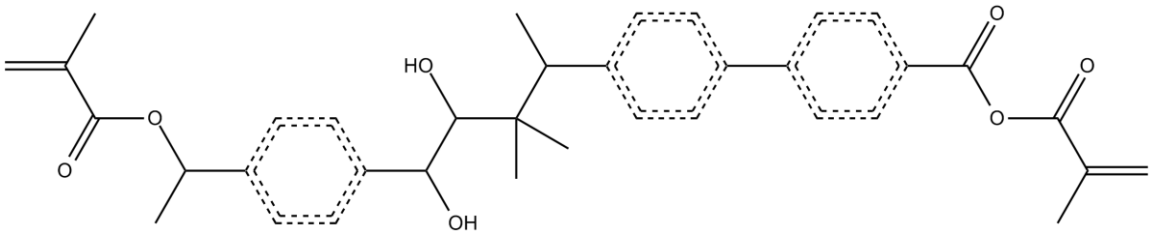


Figure 6.3 - Candidate monomer 25.3. Concentration of 25 weight percent.

Table 6.2 - Objective functions for candidate monomers at 25 weight percent

25 Weight Percent Candidate Monomers				
Name	Objective	Molecular Weight	Number of Iterations	Starting Point
BisGMA Control	1.07	513	-	-
25.1	0.012	599	860	BisGMA
25.2	0.023	599	785	BisGMA
25.3	0.023	585	617	Random Monomer

Table 6.3 - Predicted properties for candidate monomers at 25 weight percent

25 Weight Percent Candidate Monomers					
Name	Storage Modulus	Rubbery Modulus	Water Sorption	Viscosity [Pa s]	Glass Transition Temperature

	[MPa]	[MPa]	[%]		[C]
Target	3500	40	6	0.1	74
BisGMA Control	3306	30.5	7.5	0.197	68.9
25.1	3485	41.3	5.6	0.107	70.9
25.2	3491	41.3	5.3	0.107	70.9
25.3	3510	37.4	6.1	0.110	67.2

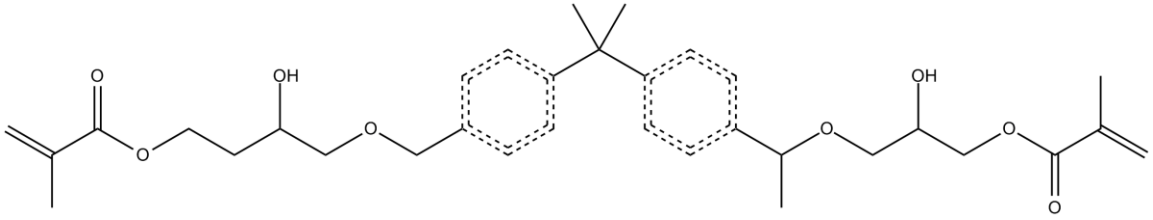


Figure 6.4 - Candidate monomer 35.1. Concentration of 35 weight percent.

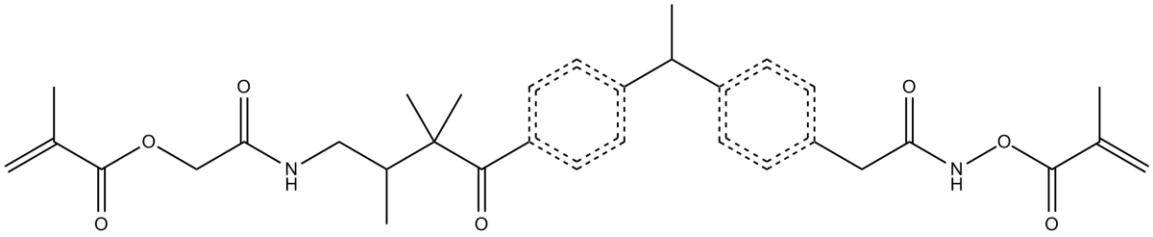


Figure 6.5 – Candidate monomer 35.2. Concentration of 35 weight percent.

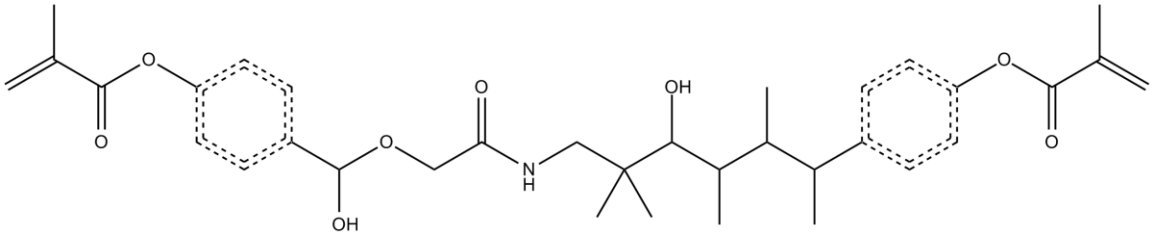


Figure 6.6 – Candidate monomer 35.3. Concentration of 35 weight percent.

Table 6.4 - Objective functions for candidate monomers at 35 weight percent

35 Weight Percent Candidate Monomers

Name	Objective	Molecular Weight	Number of Iterations	Starting Point
BisGMA Control	1.07	513	-	-
35.1	0.045	569	402	BisGMA
35.2	0.062	577	885	BisGMA
35.3	0.039	596	1153	BisGMA

Table 6.5 – Predicted properties for candidate monomers at 35 weight percent

35 Weight Percent Candidate Monomers					
Name	Storage Modulus [MPa]	Rubbery Modulus [MPa]	Water Sorption [%]	Viscosity [Pa s]	Glass Transition Temperature [C]
Target	3500	40	6	0.1	74
BisGMA Control	3306	30.5	7.5	0.197	68.9
35.1	3297	33.4	6.0	0.090	78.1
35.2	3407	37.4	4.8	0.096	82.2
35.3	3410	38.6	5.5	0.097	86.8

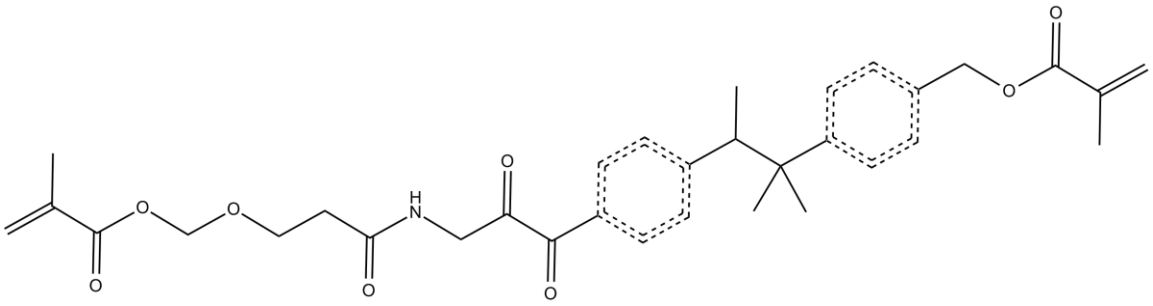


Figure 6.7 - Candidate monomer 45.1. Concentration of 45 weight percent.

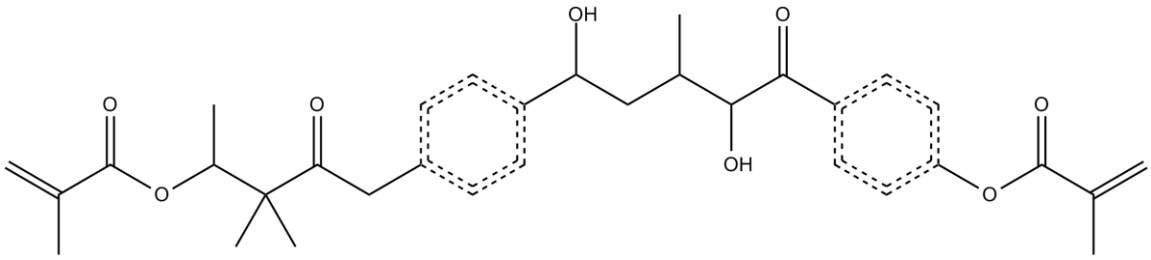


Figure 6.8 - Candidate monomer 45.2. Concentration of 45 weight percent.

Table 6.6 - Objective functions for candidate monomers at 45 weight percent

45 Weight Percent Candidate Monomers				
Name	Objective	Molecular Weight	Number of Iterations	Starting Point
BisGMA Control	1.07	513	-	-
45.1	0.068	578	930	BisGMA
45.2	0.074	565	681	BisGMA

Table 6.7 – Predicted properties for candidate monomers at 45 weight percent

45 Weight Percent Candidate Monomers					
Name	Storage Modulus [MPa]	Rubbery Modulus [MPa]	Water Sorption [%]	Viscosity [Pa s]	Glass Transition Temperature [C]
Target	3500	40	6	0.1	74
BisGMA Control	3306	30.5	7.5	0.197	68.9
45.1	3364	45.9	6.0	0.119	80.7
45.2	3535	32.4	7.0	0.109	82.2

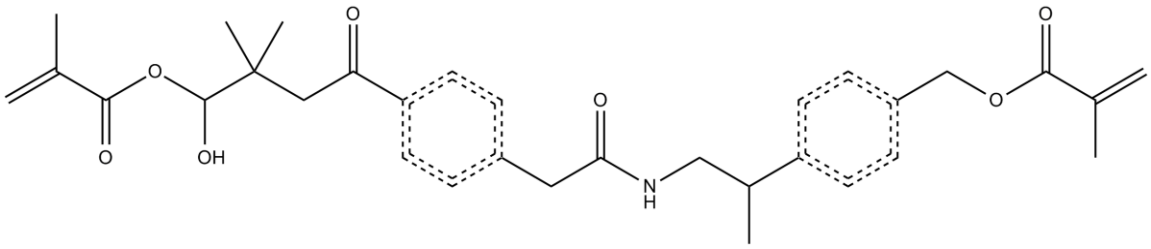


Figure 6.9 - Candidate monomer 55.1. Concentration of 55 weight percent.

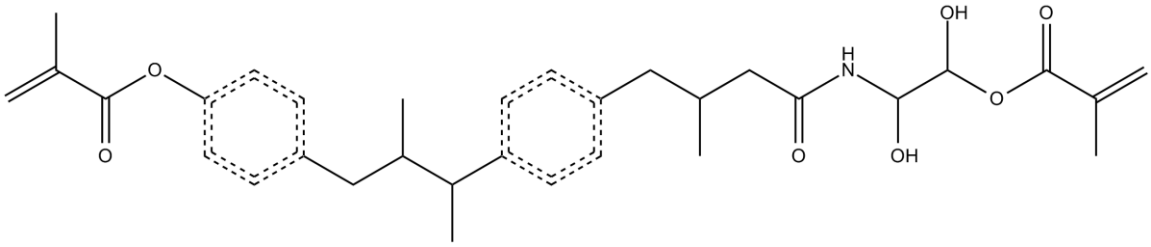


Figure 6.10 - Candidate monomer 55.2. Concentration of 55 weight percent.

Table 6.8 - Objective functions for candidate monomers at 55 weight percent

55 Weight Percent Candidate Monomers				
Name	Objective	Molecular Weight	Number of Iterations	Starting Point
BisGMA Control	1.07	513	-	-
55.1	0.043	550	656	BisGMA
55.2	0.028	552	980	BisGMA

Table 6.9 – Predicted properties for candidate monomers at 55 weight percent

55 Weight Percent Candidate Monomers					
Name	Storage Modulus [MPa]	Rubbery Modulus [MPa]	Water Sorption [%]	Viscosity [Pa s]	Glass Transition Temperature [C]
Target	3500	40	6	0.1	74
BisGMA Control	3306	30.5	7.5	0.197	68.9

55.1	3534	34.2	5.2	0.100	68.6
55.2	3445	35.3	5.9	0.093	67.5

An interesting trend for the 25 weight percent monomers is that good solutions tended to have three aromatic ring groups, oftentimes bonded directly together. This may make the monomer hard to synthesize, or unstable in some cases. Restrictions can be added to the algorithm to only allow two or zero aromatic rings if it is decided that other configurations are undesirable or infeasible. It is also interesting that candidate monomers 25.1 and 25.2 are very similar; they are actually made from the same functional groups. Monomers similar to 25.1 and 25.2 should be explored if these two monomers can not be synthesized.

The candidate monomers at 35 weight percent are more similar to the other monomers used to make the correlations, especially candidate 35.1. Candidate 35.1 should be stable and synthesizable, as it is nearly symmetric.

A literature search was made to find molecules similar to the candidate monomers. According to SciFinder's molecule database none of the candidate molecules had been synthesized previously, but there were some molecules very similar to candidate 35.1. Candidate 35.1 has the molecular formula $C_{33}H_{44}O_8$, while the similar monomers are $C_{35}H_{48}O_{10}$. Both of these monomers were patented for use in soft contact lenses.

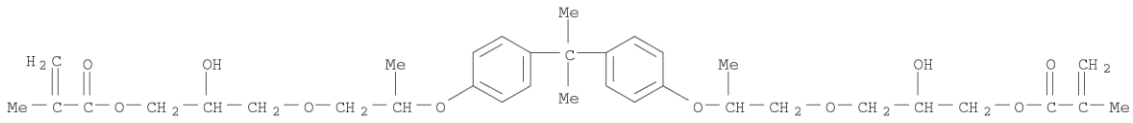


Figure 6.11 - Molecule similar to Candidate 35.1 (Hiroo, 1982)

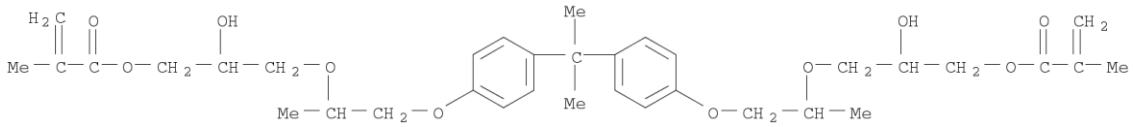


Figure 6.12 - Molecule similar to Candidate 35.1 (Kiyoshi, 1995)

The property values for these molecules were calculated. The correlations gave unrealistic negative values for water sorption. This is because the water sorption correlation is sensitive to the size of the molecule, and these molecules are larger than any molecule previously considered. The other properties gave realistic values. The objective functions were calculated excluding water sorption, with weighting factors to correct for only using four properties instead of five.

Table 6.10 - Predicted property values for monomer found by Hiroo, et al (1982)

Predicted Property Values for Monomer found by Hiroo, et al					
Weight Percent Monomer	Weighted Objective Function	Storage Modulus [MPa]	Rubbery Modulus [MPa]	Viscosity [Pa s]	Glass Transition Temperature [C]
25	0.29	3324	42.6	0.140	93.4
35	0.41	3364	52.7	0.138	95.8
45	0.69	3405	62.5	0.136	98.5
55	1.12	3448	72.1	0.133	101.4

Table 6.11 - Predicted property values for monomer found by Kiyoshi, et al (1995)

Predicted Property Values for Monomer found by Kiyoshi, et al					
Weight Percent Monomer	Weighted Objective Function	Storage Modulus [MPa]	Rubbery Modulus [MPa]	Viscosity [Pa s]	Glass Transition Temperature [C]
25	0.28	3312	42.6	0.139	93.3
35	0.40	3353	52.7	0.136	95.7
45	0.67	3394	62.5	0.134	98.3
55	1.10	3436	72.1	0.131	101.2

At 25 weight percent, both of these monomers show slight improvement over the HEMA/BisGMA control group. Using the Tabu Search algorithm to design a monomer, and then finding similar molecules which already exist, could be a valid strategy if it turns out to be difficult to include stability and ease of synthesis in the algorithm.

The results show that the algorithm can provide candidate monomers with good objective functions at any of the concentrations tested. The following section examines the error associated with these objective function and property values.

6.3 Prediction Intervals

The 95% prediction interval was calculated for each property. The calculation of a prediction interval, or error calculation, has seldom been calculated in molecular design (Roughton, 2011). The prediction interval was defined in Section 2.3.3.

The prediction interval is found using the following equation,

$$PI = \pm t_{\alpha/2,n-(k+1)} \sqrt{\hat{\sigma}^2 \left(1 + x'_p \left(X'X\right)^{-1} x_p\right)}$$

where t is the critical value of the t-distribution at the desired confidence level and degrees of freedom, $\hat{\sigma}^2$ is the mean square error, x_p is an array of descriptors for the new observation used in the model, and X is the matrix of descriptors of previously observed data points. The prediction interval is a function of the bias of the original correlation, and how different the descriptors of the candidate molecule are to the descriptors used to make the correlation. For example, the correlation for viscosity includes molecular weight, and the range of molecular weight used to make the correlation is 198-540 g/mol. If the molecular weight of the candidate molecular is much larger than 540 g/mol, there will be more error. A large prediction interval may show that the correlation is unsuited to describe that molecule. Even if the descriptors match perfectly, there is still the error associated with the original correlation, which is equal to the t-value multiplied by the mean error, $\hat{\sigma}$. This is reported as the minimal error. The following tables summarize the prediction intervals for each property and candidate monomer. Figure 6.13 gives a visual representation of the distribution of the calculated property within the prediction interval for one of the candidate monomers.

The prediction interval overlaps the target value in each case. This overlap shows that the global optimum of the design problem may actually perform worse than some local optima that the Tabu Search algorithm finds. The 68% prediction interval for rubbery modulus is reported. This was because of the large error in the correlation, which is due to having too few experimental data points, limiting the number of descriptors that could be used.

The prediction intervals were sensitive to the molecular weight or size of the candidate monomers. Most of the monomers tested to build the correlations were smaller than the candidate monomers. Correlations could be updated to include more experimental data for monomers that are larger than BisGMA.

Table 6.12 - Prediction interval for glass transition temperature

Glass Transition Temperature – 95% Prediction Interval				
Name	Predicted Value	Confidence Interval	Percent of Target Value	Crosses Target Value?
Minimal Error	-	8.1	11	-
25.1	70.9	19.1	26	Yes
25.2	70.9	15.5	21	Yes
25.3	67.2	9.6	13	Yes
35.1	78.1	20.9	28	Yes
35.2	82.2	44.6	60	Yes
35.3	86.8	32.5	44	Yes
45.1	80.7	51.7	70	Yes
45.2	82.2	45.8	62	Yes
55.1	68.6	58.7	79	Yes
55.2	67.5	49.8	67	Yes

Table 6.13 - Prediction interval for viscosity

Viscosity – 95% Prediction Interval				
Name	Predicted Value	Confidence Interval	Percent of Target Value	Crosses Target Value?
Minimal Error	-	0.016	16	-
25.1	0.107	0.045	45	Yes
25.2	0.107	0.046	46	Yes
25.3	0.110	0.018	18	Yes
35.1	0.090	0.037	37	Yes
35.2	0.096	0.017	17	Yes
35.3	0.097	0.044	44	Yes
45.1	0.119	0.020	20	Yes
45.2	0.109	0.026	26	Yes
55.1	0.100	0.022	22	Yes
55.2	0.093	0.049	49	Yes

Table 6.14 - Prediction interval for percent water sorption

Water Sorption – 95% Prediction Interval				
Name	Predicted Value	Confidence Interval	Percent of Target Value	Crosses Target Value?
Minimal Error	-	1.0	17	-
25.1	5.6	1.6	27	Yes
25.2	5.3	1.6	27	Yes
25.3	6.1	1.7	28	Yes
35.1	6.0	1.5	25	Yes
35.2	4.8	2.5	42	Yes
35.3	5.5	1.6	27	Yes
45.1	6.0	2.3	38	Yes
45.2	7.0	2.0	33	Yes
55.1	5.2	2.0	33	Yes
55.2	5.9	1.4	23	Yes

Table 6.15 - Prediction interval for storage modulus

Storage Modulus – 95% Prediction Interval				
Name	Predicted Value	Confidence Interval	Percent of Target Value	Crosses Target Value?
Minimal Error	-	310	9	-
25.1	3485	1337	38	Yes
25.2	3491	1337	38	Yes
25.3	3510	1390	40	Yes
35.1	3297	1360	39	Yes
35.2	3407	1398	40	Yes
35.3	3410	1360	39	Yes
45.1	3364	1370	39	Yes
45.2	3535	1358	39	Yes
55.1	3534	1379	39	Yes
55.2	3445	1329	38	Yes

Table 6.16 - Prediction interval for rubbery modulus

Rubbery Modulus – 68% Prediction Interval				
Name	Predicted Value	Confidence Interval	Percent of Target Value	Crosses Target Value?
Minimal Error	-	13.7	34	-
25.1	41.3	19.2	48	Yes
25.2	41.3	19.2	48	Yes
25.3	37.4	18.6	45	Yes
35.1	33.4	19.1	46	Yes
35.2	37.4	19.6	48	Yes
35.3	38.6	20.2	49	Yes
45.1	45.9	18.8	51	Yes
45.2	32.4	18.7	42	Yes
55.1	34.2	18.9	47	Yes
55.2	35.3	18.7	49	Yes

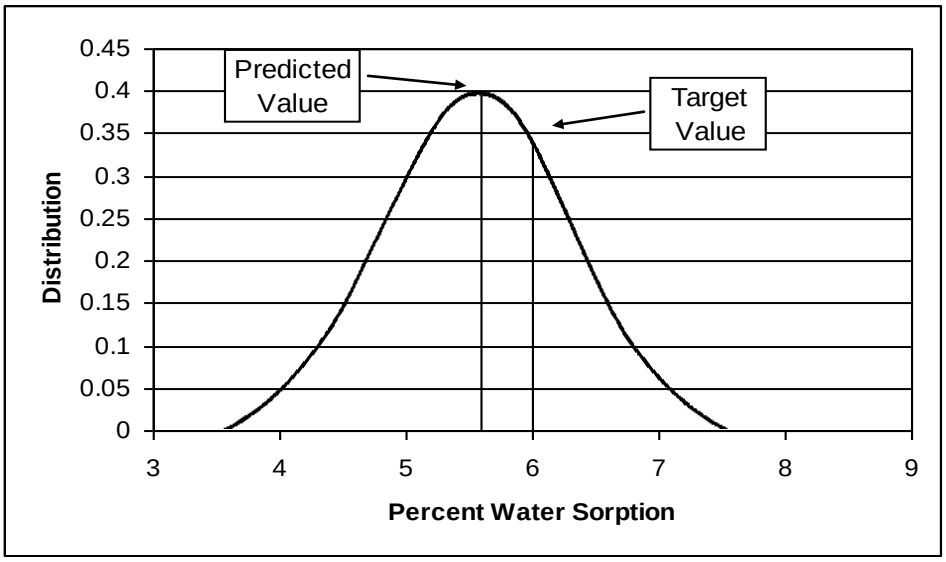


Figure 6.13 - Normal distribution for percent water sorption for Candidate 25.1

6.4 Summary

These results show that this methodology can be used to design crosslinked polymers using the Tabu Search algorithm. The prediction intervals found were as small as 13% of the target value. This is an acceptable range when one considers that finding candidate monomers which are improved compared to the standard resin is more important than finding a resin with a specific property value. The algorithm can provide a long list of candidate monomers which can be examined by experimental chemists to be considered for synthesis. Polymer Designer has a flexible framework that can be changed to add more restrictions to create candidates that are more easily synthesizable.

The addition of stability criterion decreased the chances of finding an unusable solution. When the algorithm was run without any stability criteria, candidate solutions with peroxide groups often appear.

More property data should be gathered to create more accurate correlations that will give predicted values with less error.

The next Chapter gives overall conclusions and recommendations for this project.

Chapter 7.

Conclusions and Recommendations

The previous Chapter provided a list of candidate molecules with predicted property values superior to that of the standard HEMA/BisGMA composite. The results show that this methodology can be used to design molecules with specified target properties. It also shows that this methodology is capable of handling the complexity that comes from crosslinking. Further restrictions or more accurate correlations can easily be added to create more suitable molecules. This methodology can be used by other projects to design different types of molecules.

Currently the Polymer Designer program is being edited to be used in other projects. It provides a flexible framework that can be changed to work with different molecules and types of functional groups.

A set of criteria for choosing the overall best QSPR models was created, but can be improved. The goal of any model selection technique is to give a correlation with low error. In this project we found that the prediction interval is dependent on the correlation error, the number of descriptors, and even the type of descriptors used. A more computationally extensive method could be employed in which the prediction interval for

a subset of molecules found, the correlation and with the smallest prediction interval is chosen. It may even be best to complete CMD with many different correlations to choose the best models.

Prediction intervals have rarely been considered in molecular design projects (Roughton, 2011). This method of error analysis can be used in a number of different ways to improve the project. The value of the prediction interval can be calculated as part of the Tabu Search algorithm. The objective function can be changed to take the prediction interval into account, favoring candidates both with favorable properties and smaller prediction intervals. A possible form of the objective function would be

$$f = \sum_{i \in \text{properties}} s_i \left[\left(\frac{P_{i,\text{target}} - P_{i,\text{predicted}}}{P_{i,\text{target}}} \right)^2 + a_i \left(\frac{P_{i,\text{target}} - P_{i,\text{predicted}\downarrow}}{P_{i,\text{target}}} \right)^2 + a_i \left(\frac{P_{i,\text{target}} - P_{i,\text{predicted}\uparrow}}{P_{i,\text{target}}} \right)^2 \right]$$

where $P_{\downarrow}$ is the lower bound of the prediction interval, $P_{\uparrow}$ is the upper bound for the prediction interval, and a_i is a weighting factor for the prediction interval values. A further restriction can be made to only allow candidates where the prediction interval overlaps the target property value.

The target property values and weighting functions should be examined carefully. The general effect that these properties have on the longevity was found, but a better understanding could give more exact values. The optimal property value could lie outside of the range of the experimental data. If that is the case, more molecules should be tested and new correlations should be created.

The primary descriptors used in this project were connectivity indices. There are thousands of structural descriptors that can be used in CMD. Signature descriptors (Weis, 2010) and Kier shape indices (Kier, 1987) have been used to design polymers. The algorithms already being used by Polymer Designer allow the calculation of many of these descriptors within the Tabu Search algorithm. Additional structural descriptors should be studied. A partially theoretical model could be built by studying chemically how different functional groups affect certain properties. This would give guidelines to which types of structural descriptors would more likely be able to model these properties.

Besides the experimental properties, other factors could possibly be related to structural descriptors. During the experimental testing phase, many monomers could not be included in this study because they would not dissolve into HEMA at the concentrations being tested. If solubility in HEMA could be predicted, this would save disnificant experimental effort and resources which would have been spent synthesizing a candidate monomer that is not feasible.

Additional stability criterion can be applied to the algorithm. Fink and Reymond (2007) applied a filter of rejected functional groups when creating a database of feasible stable organic molecules. Many of these functional groups can not be made with the chain groups used in this project. Only a part of Fink and Reymond's filter would need to be added to the Tabu Search algorithm. Restrictions can be added using the penalty method and the subgraph isomorphism algorithm described in Chapter 3.1. In addition, criterion for ease of synthesis could be added in a similar manner.

A literature search to find molecules similar to the candidate molecules was performed, described in Section 6.2. The property values at the tested resin concentrations were predicted for these similar molecules. This method could find an existing molecule suitable for use as part of a dental polymer which has never been considered before. The CMD results would provide a way of narrowing the search, as searching through all available monomers would be infeasible.

Overall the project shows that the Tabu Search algorithm is robust enough for the design of crosslinked polymers. The procedure outlined provided a list of candidate monomers that could show improvement to the standard dental composite resin on the market today. The examination of the correlation error through the prediction interval shows the error that is likely present in many other molecular design projects, suggesting that future projects should include error propagation during the design phase.

References

American Dental Association, "Resin-Based Composites," *Journal of American Dental Association,* **134**, 510-512 (2003).

Aspnes, James, "Notes on Graph Theory," Retrieved from http://pine.cs.yale.edu/pinewiki/GraphTheory (2010).

Banzhaf, Wolfgang; Nordin, Peter; Keller, Robert; Francone, Frank, *Genetic Programming – An Introduction,* Morgan Kaufmann, San Francisco, CA (1998).

Barnes, Howard; Hutton, John; Walters, Kenneth, *An Introduction to Rheology,* Elsevier (1993).

Bicerano, J., *Prediction of Polymer Properties,* 3rd edition, Marcel Dekker, Inc. (2002).

Bicerano, J., R. L. Sammler, C. J. Carrier, and J. T. Seitz, "Correlation between glass transition temperature and chain structure for randomly crosslinked high polymers," *Journal of Polymer Science*, **34**, 2247 (1996).

Bosze, E.J.; Alawar, A.; Berschger, O.; Tsai, Yun-I; Nutt, S.R, "High-Temperature Strength and Storage Modulus in Unidirectional Hybrid Composites," *Composites Science and Technology,* **66**, 1963-1969, (2006).

Brostow, Witold; Datashvili, Tea; Geodakyan, James; Lou, Jesse, "Thermal and mechanical properties of EPDM/PP + thermal shock-resistant ceramic composites," *Journal of Materials Science,* **46**, 2445 (2011).

Brown, Michael; Martin, Shawn; Rintoul, Mark; Faulon, Jean-Loup, "Designing Novel Polymers with Targeted Properties Using the Signature Molecular Descriptor," *Journal of Chemical Information and Modeling*, **46**, 2, 826 (2006).

Bullinaria, John A, "Bias and Variance, Under-Fitting and Over-fitting," University of Birmingham, Retrived from http://www.cs.bham.ac.uk/~jxb/INC/l9.pdf (2010).

Camarda, K. V. and C. D. Maranas, "Optimization in polymer design using connectivity indices," *Industrial Engineering and Chemistry Research*, **38**, 1884 (1999).

Collins, C. J., R. W. Bryant, and K. L. V. Hodge, "A clinical evaluation of posterior composite resin restorations: 8-year findings," *Journal of Dentistry*, **26**, 311–317 (1998).

Constaninou, L; Gani, R, "New Group Contribution Method for Estimating Properties of Pure Compounds," *American Institute of Chemical Engineering Journal*, **40,** 10, 1697 (1994).

Cook, WD, "Photopolymerization Kinetics of Dimethacrylates using the Camphorquinone Amine Initiator System," *Polymer*, **33,** 3, 600 (1992).

De Werra, D; Hertz, A, "Tabu Search Techniques – A Tutorial and an Application to Neural Networks," *OR Spektrum*, **11**, 3, 131 (1989).

Dean, John, *The Analytical Chemistry Handbook*. McGraw Hill. P 15.1-15.5 (1995).

Department of Polymer Science, The University of Southern Mississippi. "The Glass Transition," http://pslc.ws/macrog/tg.htm. (2005).

Deshayes, Gaelle; Delcourt, Cecile; Verbruggen, Ingrid; Trouillet-Fonti, Lise; Touraud, Franck; Fleury, Etienne; Degee, Philippe; Destarac, Mathias; Willem, Rudolph; Dubois,

Philippe, "Novel Polyesteramide-Based Di- and Triblock Copolymers: From Thermo-Mechanical Properties to Hydrolytic Degradation," *European Polymer Journal*, **7,** 1, P 98-110 (2011).

Dhanpal, P; Yiu, C.K.Y.; King, N.M.; Tay, F.R.; Hiraishi, N, "Effect of Temperature on Water Sorption and Solubility of Dental Adhesive Resins," *Journal of Dentistry*, **37**, P 122-132 (2009).

Draper, N. R. and H. Smith, *Applied Regression Analysis*, John Wiley and Sons, New York (1966).

Edgar, TF; Dixon, DA; Reklaitis, GV, "Vision 2020: Computational Needs of the Chemical Industry," National Research Council Chemical Sciences Roundtable, Washington DC (1999).

Eslick, John, "Molecular Design of Crosslinked Copolymers," doctoral dissertation, University of Kansas, Lawrence, Kansas (2008).

Eslick, John; Ye, Q.; Park, J.; Topp, E.M.; Spencer, P.; Camarda, Kyle, "A Computational Molecular Design Framework for Crosslinked Polymer Networks," *Computers and Chemical Engineering*, **33,** 954 (2009).

Ferry, J.D., *Viscoelastic Properties of Polymers*, Wiley, (1980).

Fink, Tobias; Reymond, Jean-Louis, "Virtual Exploration of the Chemical Universe up to 11 Atoms of C,N,O,F: Assembly of 26.4 Million Structures (110.9 Million Stereoisomers) and Analysis for New Ring Systems, Stereochemistry, Physicochemical Properties, Compound Classes, and Drug Discovery," *Journal of Chemical Information and Modeling*, **47**, 342 (2007).

Fouassier, JP, "A New 3-Component System in Visible Laser-Light Photoinduced Polymerization," *Journal of Imaging Science and Technology*, **37,** 2, 208 (1993).

Fredenslund, Aage, "Group-Contribution Estimation of Activity-Coefficients in Nonideal Liquid-Mixtures," *American Institute of Chemical Engineers Journal,* **21,** 6, 1086 (1975).

Fried, Joel, *Polymer Science and Technology*, Prentice-Hall, 2nd Edition, 154-158 (2003).

Friedler, F; Fan, L.T.; Katotai, L; Dallos, A, "A Combinatorial Approach for Generating Candidate Molecules with Desired Properties Based on Group Contribution," *Computers and Chemical Engineering*, **22,** 6, 809 (1998).

Gani, R; Fredenslund, A, "Computer-aided Molecular Design with Specific Property Constraints," *Fluid Phase Equilibrium,* **44,** 7262 (1993).

Gani, R; Nielson, B; Fredenslund, A, "A Group Contribution Approach to Computer-Aided Molecular Design," *American Institute of Chemical Engineers Journal,* **37,** 9, 1318 (1991).

Gardiner, D.J. *Practical Raman Spectroscopy*. Springer. 1989.

Ge, Z.Y.; Tao, Z.Q.; Li, G.; Ding, J.P.; Fan, L.; Yang, S.Y, "Synthesis and Properties of Novel Fluorinated Epoxy Resins," *Journal of Applied Polymer Science,* **120,** 1, 148-155 (2010).

Glover, F, "Tabu Search: A Tutorial," *Interfaces,* **20,** 74 (1990).

Goldberg, David E, *Genetic Algorithms in Search, Optimization and Machine Learning*, Kluwer Academic Publishers, Boston, MA (1989).

Hiroo, S; Kenichi, T, “Synthetic Resin Lens,” Patent Abstracts of Japan, Publication number 57-104901 (1982).

Horst, R; Tuy, H, *Global Optimization: Deterministic Approaches*, Springer (1996).

Jeon, So-Yeong; Kim, Yong-Hyuk, “A Genetic Approach to Analyze Algorithm Performance Based on the Worst-Case Instances,” *Journal of Software Engineering & Applications,* **3**, 767 (2010).

Joback, KG; Reid, RC, “Estimation of Pure-Component Properties from Group-Contributions,” *Chemical Engineering Communications,* **57,** 233 (1987).

Karunanithi, Arunprakash; Achenie, Luke; Gani, Rafiqul, “A Computer-Aided Molecular Design Framework for Crystallization Solvent Design,” *Chemical Engineering Science,* **61,** 1247 (2006).

Kier, L.B., “Index of Molecular Shape from Chemical Graphs,” *Medicinal Research Reviews,* Issue 7, 417 (1987).

Kiyoshi, I; Shinji, N; Yasuyoshi, K., “Contact Lens,” Patent Abstracts of Japan. Publication number 07-168139 (1995).

Knox, J., “Tabu Search Performance on the Symmetrical Traveling Salesman Problem,” *Computers and Operations Research,* **21,** 8, 867 (1994).

Konig, Rainer; Dandekar, Thomas, “Refined Genetic Algorithm Simulations to Model Proteins,” *Journal of Molecular Modeling,* **5,** 317 (1999).

Layric, Vasile; Iancu, Petricia; Plesu, Valentin, “Genetic Algorithm Optimization of Water Consumption and Wastewater Network Topology,” *Journal of Cleaner Production, 13,* 15, 1405 (2005).

Lovell, Lale; Newman, Sheldon; Donaldson, Matthew; Bowman, Christopher, "The Effect of Light Intensity on Double Bond Conversion and Flexural Strength of a Model, Unfilled Dental Resin," *Dental Materials,* **19,** 458 (2003).

Lumley, T, "The LEAPS Package," Retrieved from http://www.cran.r-project.org/doc/packages/leaps.pdf (2004).

Mallows, C.L., "Some Comments on Cp," *Technometrics,* **15,** 4, 661 (1973).

Manu, SK; Varghese, TL; Mathew, S; Ninan, KN, "Studies on Structure Property Correlation of Cross-Linked Glycidyl Azide Polymer," *Journal of Applied Polymer Science,* **114,** 3360 (2009).

Marrero, Jorge. Gani, Rafiqul, "Group-Contribution Based Estimation of Pure Component Properties," *Fluid Phase Equilibria,* **183,** 183 (2001).

Marrero, Jorge; Gani, Rafiqul, "Group-Contribution Based Estimation of Octanol/Water Partition Coefficient and Aqueous Solubility," *Industrial and Engineering Chemistry Research,* **41,** 6623 (2002).

Matsui, T; Miwa, Y, "Detection of a New Crosslinking and Properties of Liquid Polysulfide Polymer," *Journal of Applied Polymer Science,* **71,** 1, 59 (1999).

Menard, Kevin, *Dynamic Mechanical Analysis: A Practical Introduction,* CRC Press (1991).

Meyers, Marc; Chawla, Kumar, *Mechanical Behavior of Materials,* Prentice Hall (1999).

Nelson, Steven; Seybold, Paul, "Molecular Structure-Property Relationships for Alkenes," *Journal of Molecular Graphics and Modelling,* **20**, 1, 36 (2001).

Park, J., Q. Ye, E. M. Topp, E. L. Kostoryz, Y. Wang, S. L. Kieweg, and P. Spencer, "Preparation and properties of novel dentin adhesives with esterase resistance," *Journal of Applied Polymer Science,* 107, 3588 (2007).

Picard, Richard; Cook, Dennis, "Cross-Validation of Regression Models," *Journal of the American Statistical Association,* **79,** 387, 575 (1984).

Podgorski, Maciej, "Synthesis and Characterization of Novel Dimethacrylates of Different Chain Lengths as Possible Dental Resins," *Dental Materials,* **26,** 188-194 (2010).

R Development Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria, URL http://www.R-project.org (2007).

Raman, V.S.; Maranas, C.D., "Optimization in Product Design with Properties Correlated with Topological Index," *Computers and Chemical Engineering,* **22,** 6, 747 (1998).

Randic, M, "Characterization of Molecular Branching," *Journal of the American Chemical Society,* **97,** 23, 6609 (1975).

ReliaSoft Corporation, "Experiment Design and Analysis Reference," Retrieved from http://www.weibull.com/DOEWeb/experiment_design_and_analysis_reference.htm. (2008).

Roughton, Brock; Topp, E.M.; Camarda, K.V., "Use of Glass Transitions in Carbohydrate Excipient Design for Lyophilized Protein Formulations," Manuscript submitted for publication (2011).

Satyanarayana, Kavitha; Abildskov, Jens; Gani, Rafiqul, "Computer-aided Polymer Design using Group Contribution Plus Property Models," *Computers and Chemical Engineering,* **33**, 5, 1004 (2008).

Schneider, A, "The Meaning of the Glass Temperature of Random Copolymers and Miscible Polymer Blends," *Journal of Thermal Analysis and Calorimetry,* **56,** 983 (1999).

Sideridou, I; Tserki, V.; Papanastasiou, G., "Effect of Chemical Structure on Degree of Conversion in Light-Cured Dimethacrylate-Based Dental Resins," *Biomaterials,* **23,** 1819-1829 (2002).

Sideridou, Irini; Karabela, Maria; Spyroudi, Crysa, "Dynamic Mechanical Analysis of a Hybrid and a Nanohybrid Light-Cured Dental Resin Composite," *Journal of Biomaterials Science,* **20,** 1797-1808 (2009).

Spencer, Paulette; Ye, Qiang; Park, Jonggu; Topp, Elizabeth M.; Misra, Anil; Marangos, Orestes; Wang, Yong; Bohaty, Brenda; Singh, Viraj; Sene, Fabio; Eslick, John; Camarda, Kyle; Katz, J. Lawrence, "Adhesive/Dentin Interface: The Weak Link in the Composite Restoration," *Annals of Biomedical Engineering,* **38,** 1989-2003 (2010).

Ullmann, J.R., "An Algorithm for Subgraph Isomorphism," *Journal of the Association of Computing Machinery***, 23,** 1, 31 (1976).

Venkatasubramanian, V; Chan, K; Caruthers, J.M., "Computer Aided Molecular Design Using Genetic Algorithms," *Computers and Chemical Engineering,* **18,** 833 (1994).

Visco, Donald Jr; Pophale, Ramdas; Rintoul, Mark; Faulon, Jean-Loup, "Developing a Methodology for an Inverse Quantitative Structure-Activity Relationship Using the Signature Molecular Descriptor," *Journal of Molecular Graphics and Modelling*, 20, 429 (2002).

Viswanathan, J., "A Combined Penalty-Function and Outer-Approximation Method for MINLP Optimization," *Computers and Chemical Engineering,* **14,** 7, 769 (1990).

Wasserman, Larry, *All of Statistics: A Concise Course in Statistical Inference,* Springer-Verlag, New York (2004).

Weis, Derick; Visco, Donald, "Computer-Aided Molecular Design Using the Signature Molecular Descriptor: Application to Solvent Selection," *Computers and Chemical Engineering,* **34,** 1018 (2010).

Ye, Qiang; Park, Jonggu; Topp, Elizabeth; Spencer, Paulette, "Effect of Photo-Initiators on the In Vitro Performance of a Dentin Adhesive Exposed to Simulated Oral Environment," *Dental Materials,* **25,** 452-458 (2009).

A. Appendix

A) Polymer Designer Handbook

This appendix gives instructions on how Polymer Designer (PD) can be used for molecular design using Tabu Search. It should be in addition to the Polymer Designer Manual given in the appendix of Eslick's thesis (Eslick, 2008). Eslick's thesis gives instructions on how to use the Polymer Designer program as it stands, but does not give details on how the code can be edited so that it can be customized to work for other research projects.

The purpose of this handbook is to describe how Tabu Search is done in PD, and show how the code can be edited to introduce new descriptors, use group contribution expressions, adjust Tabu Search parameters, and other steps necessary to change PD to be used in other research projects.

i) Running Tabu Search

This section describes how Tabu Search (TS) is run after all coding is complete. The algorithm has to have a starting point. This is done by creating a polymer structure which is made of the chain, terminal, and branching groups that will be used to build the candidate molecules. First, these groups must be chosen. This is done in the Graph Operations window, under the Graph tab in the graph editor toolbox. Pressing the add

monomer button gives you the list of monomers in the database. The monomer list shows the name of each group, the number of connectors, and the state vector. The state vector represents which state the monomer is in. For example, whether or not a monomer's methacrylate group is part of a chain or not. In most cases a group has only one state vector.

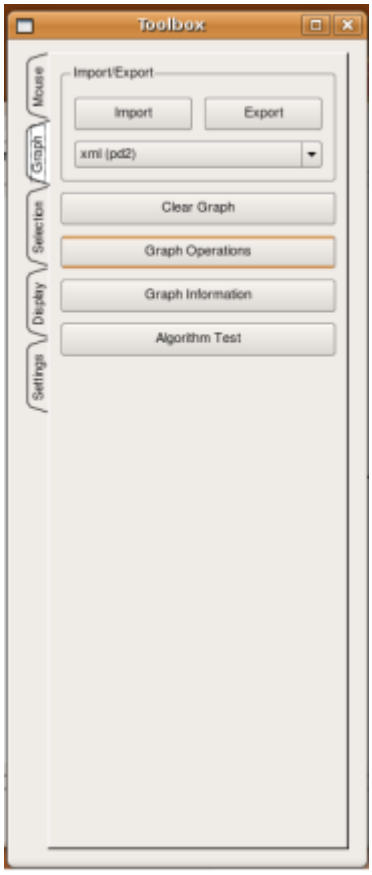


Figure A.1 - Graph editor toolbox

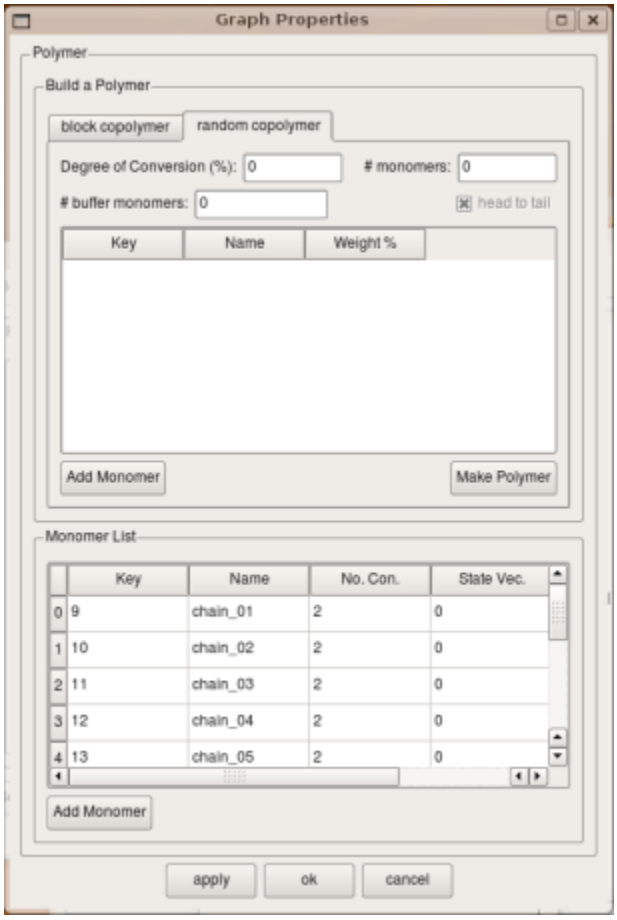


Figure A.2 - Graph Operations

After this the starting polymer is drawn. Each end of the polymer must have a terminal, a group with only one connector. The amount of branching has to be defined in the starting polymer, because PD's TS algorithm currently can not change branching groups; it will only change groups with two connectors. The properties of each edge then need to be edited. When a chain group is being replaced the program needs to know how to connect the newly inserted group. The edge properties need to be edited such that one monomer has connector label 1, and the other has connector label 2.

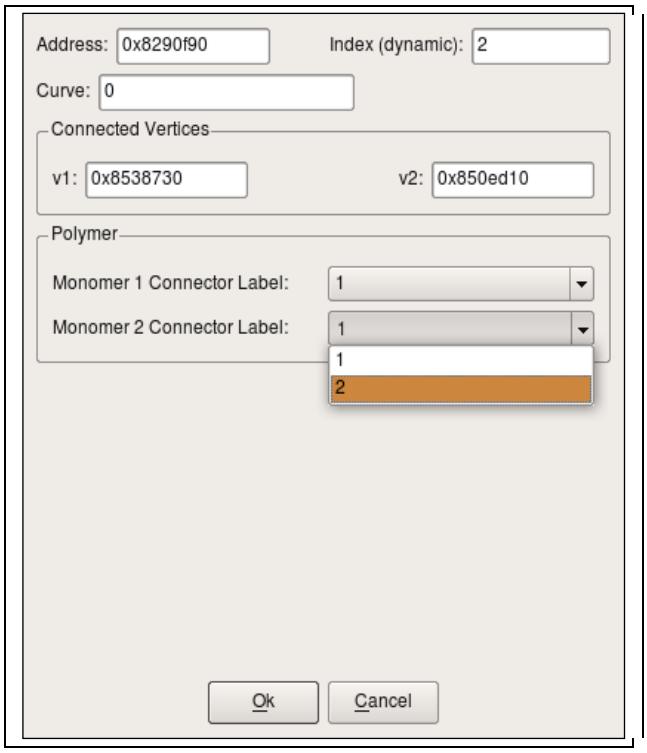


Figure A.3 - Editing the connector labels of an edge

Once this is done, Tabu Search can be run. The "Tabu Test" algorithm is found by pressing the algorithm test button under the Graph tab of the graph properties toolbox. The Tabu results are printed to the terminal, including the best objective function value found so far, and the objective value of the current solution. After the stop criteria has been met, currently set at 200 non-improving iterations, the search results are printed to the terminal.

```
non-improving iterations: 198 best objective: 0.0648537 current objective: 0.114271
non-improving iterations: 199 best objective: 0.0648537 current objective: 0.0954685
non-improving iterations: 200 best objective: 0.0648537 current objective: 0.102552
-------------TABU SEARCH RESULT----------------
Initial objective function: 0.136446
Best objective function: 0.0648537

Degree of conversion: 77.2424
Crosslink density: 0.163711
Crosslink density (DC = 100%): 0.222685
Tensile strength: 86.5399
Glass transision temperature: 118.082
Initial polymerization rate: 140.174
Molecular weight: 616.796
```

Figure A.4 - Tabu Search results

The candidate monomer is saved as an xml file in the directory that the PD program is run from, which is by default in /pd2/. The file is named tabuSol.xml. The file will be overwritten when the TS algorithm is run again, so it should be backed up immediately. The file can be opened by PD by importing it as a monomer structure.

ii) Adjusting Tabu Search Parameters

Termination Criterion

/src/tabu/pd2_tabusearch.cpp

Currently the termination criteria is set to stop after 200 iterations where an improved solution is not found. The program runs in a *while* loop. The current objective value is compared to the best objective value and if the current solution is better then the number of non-improving iterations is reset to zero, otherwise the value goes up by one.

Different terminal criteria can be used, such as reaching a certain objective value. This can be done by setting a new variable equal to the best objective function using the line

Bestobjectivevariable = best_sol->obj;

Local Intensification and Step Size

/src/tabu/dp_tabu_01.cpp

This code describes how chains are inserted, deleted, or swapped. One step is one of these actions, and the step size is the number of these actions that is made at each iteration. By default, the minimum step size is one, and the maximum step size is two.

The program first examines the size of the monomer. If it is at the maximum or minimum size, then groups can not be added or deleted respectively. Currently the minimum number of groups is three, two terminals and one chain group in between, and the maximum is thirty. The program then randomly selects how many groups will be added, deleted, or swapped, up to the maximum step size.

The step sizes can be edited to either increase the neighbor size, or for local intensification. For example, an IF statement can be added so that when a good solution is found the step size is decreased to search the surrounding solution space more thoroughly.

```
  nOpMin = 1;
  nOpMax = 1;
  maxVert = 30;

  currentobj= sol->obj;

  if((currentobj>1)) nOpMax=8;


  if( ((sol_data_t*)(newSol->sol))->mol_gen.numVertices() >=
maxVert) op = 1;
  else if(((sol_data_t*)(newSol->sol))->mol_gen.numVertices() <= 3
) op = 0;
  else op = pd2_tabuSearch::randomIntUniform(0,1);
  //select a number to delete or insert
  nOp = pd2_tabuSearch::randomIntUniform(0,nOpMax);
  ((sol_data_t*)(newSol->sol))->mol_gen = ((sol_data_t*)(sol-
>sol))->mol_gen;
  for(i=0; i<nOp; ++i){
    if(op==0) ((sol_data_t*)(newSol->sol))-
>mol_gen.constructMoleculeType1_InsertGroup();
    else ((sol_data_t*)(newSol->sol))-
>mol_gen.constructMoleculeType1_DeleteGroup();
  }
  //select a number of replacements
  nrMax = nOpMax - nOp;
  if(nOp < nOpMin) nrMin = nOpMin - nOp;
  else nrMin = 0;

  if( nrMax == 0 ) nr = 0;
  else if( nrMax == nrMin) nr = nOpMax;
  else nr = pd2_tabuSearch::randomIntUniform(nrMin,nrMax);
```

Tabu List

/src/tabu/dp_tabu_01.cpp

This code describes how the program checks to see if a solution is taboo or not. The program compares the molecular weight and weighted connectivity index, χ_i of the current solution to the molecules stored in the Tabu list, and if they are different enough the current solution is not labeled as Tabu.

This starts by defining the ranges of χ_i. This is the difference between the largest χ_i and the smallest χ_i for the molecules tested to make the QSPRs. The default values are for the

monomers being used when PD was first written. These should be updated for different projects to increase accuracy. However, because these are weighted values the default values may work for other projects. The value of *tabu_xi_close* and *tabu_mw_close* decide how different the current solution needs to be to not be taboo. These values can be adjusted to make it harder or easier for a solution to be labeled taboo.

iii) Adding Variables

Property Variables

Because property values are saved to the solution data, they need to be defined in multiple locations and need to be defined as public variables.

/src/tabu/dp_tabu_01.cpp

Property variables first need to be defined as local variables of the type double.

After the property values and objective function is calculated the property value needs to be saved to the solution data.

At the end of the document the data from the local solution needs to be saved to the public solution data.

/src/tabu/dp_tabu_01.hpp

The variables also need to be defined as public variables.

Descriptor Variables

Other descriptors can be calculated within the Tabu Search algorithm. If a variable does not need to be printed out with the solution, then it only needs to be defined as a local variable of type double. If the variable does need to be printed out, then it should be treated as a property variable.

iv) Calling Bicerano Connectivity Index

To save computation time, only descriptors being used in the QSPRs should be calculated. Extra connectivity index need to be defined as local double variables, and then the connectivity index are called using the following lines of code. The connectivity index can be calculated for the backbone, side chains, and crosslinked atoms if needed. The commands getBiceranoChi gets the unweighted connectivity index, and getBiceranoXi gets the weighted connectivity index.

v) Solution Printout

The data that is printed in the terminal after the Tabu Search algorithm completes can be edited.

/src/tabu/pd2_tabusearch.cpp

This part of the code prints the objective function value for the starting monomer and the objective function value for the best solution.

/src/tabu/dp_tabu_01.cpp

This part of the code prints the values of public variables before saving the best solution to an XML file named ‘tabuSol.xml’.

vi) Editing Objective Function and Property Calculations

/src/tabu/dp_tabu_01.cpp

The objective function is written in the following form.

In most cases, the weighting factor s_i will be equal to one. Penalty functions can be added to the objective function.

The property calculations are currently linear, but can be written in non-linear forms.

vii) Group Contribution

Polymer Designer can use group contribution techniques in addition to QSPRs. This is done by describing the groups, adding them to the group library, and having the TS algorithm count the number of those groups present for property calculations.

Counting Groups from the Group Library

The group library stores many different functional groups that can be used in group contribution. The default groups can be viewed in the folder */pd2/groups* saved as XML files that can be imported into a monomer structure. The default group library is also summarized in Appendix B of Eslick's thesis (Eslick, 2008).

These groups are called using their unique integer identifier (UID). The number of functional groups can be counted and set as a new local variable.

Adding to the Group Library

The default groups in the group library should not be changed because some methods depend on these unique groups, but more groups can be added. The first step is making a monomer structure of the functional group. Then the group is exported as an XML file. This provides the XML code describing the functional group.

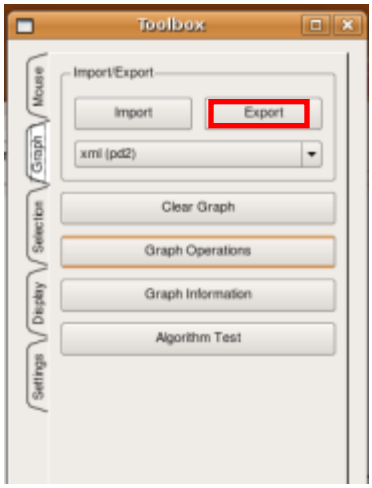


Figure A.5 - Exporting a monomer structure to an XML file

/src/files/groupLibrary.xml

The XML code can then be copied to the groupLibrary file, adding tags defining the UID for the new group.

viii) Editing Atomic Data

/src/files/atomic_data.xml

This file allows the addition of more elements than what's available by default. Atoms of different valencies can be added to allow for ionic materials.

ix) Replacing Groups During Tabu Search

/src/chem/polymer.cpp

The section that describes how groups are replaced during Tabu Search is in the *polymer.cpp* file.

The function starts by choosing an integer *vi* to represent a vertices in the graph. Each vertices in the polymer graph is either a chain, terminal, or branching monomer group. The *mon* variable chooses from the monomer list a random chain group (a monomer with a degree of 2). Then the function verifies that the randomly selected *vi* monomer is a chain group by looking at its degree. If a terminal or branching group was selected then the function terminates and no chain group is changed. Then the index for the vertex *vi* is edited to represent the group being replaced.

Replacing Terminal Groups during Tabu Search

The default is that only chain groups are changed during Tabu Search. The code can be edited so that terminal groups are replaced instead.

/src/chem/polymer.cpp

The way the *mon* variable is found needs to be edited. Instead a terminal group needs to be selected. Also the degree of vertex *vi* needs to be equal to 1.

/src/tabu/dp_tabu_01.cpp

Terminal groups being added or contracted can't happen, so a setting has to be changed so that the program will not try to do this.

If you do not wish to change too much of the code, then you can simply change the code so that *nOp* will always be equal to zero. The variable *nOp* is the number of chain groups to be deleted or added. If this is set to zero, then the only step changes that will take place is switching groups.

Possible Improvements

It would be possible to write the code so that the groups that can be changed are both chain and terminal. The ReplaceGroup() function could be changed so that it checks to see what the vertex degree of *vi* is then either swap with a terminal or chain. The functions InsertGroup() and DeleteGroup() should work as is. The problem is that the function terminates if the vertex degree of *vi* is not 2, but then still counts it as a step change. It there are too many terminal and branching groups then this will happen too often and the iteration count will be too innaccurate. Instead of the function terminating, the code could be written so that it enteres a For loop until it finds a *vi* with a degree of 2.

B) Nomenclature

HEMA	2-hydroxyethyl methacrylate
BisGMA	bisphenol A diglycidylether methacrylate
CD_{100}	Crosslink density of fully crosslinked polymer
Xx	Dummy atom
MW_{avg}	Mole average molecular weight of comonomer
MW_x	Molecular weight of test monomer
N_{rot}	Number of rotational degrees of freedom
MW_{wted}	Weight average molecular weight of comonomer
χ^{n}_{avg}	Average nth-order simple connectivity index
$\chi^{v,n}_{avg}$	Average nth-order valence connectivity index
χ^{n}_{x}	Nth-order simple connectivity index of test monomer
$\chi^{v,n}_{x}$	Nth-order valence connectivity index of test monomer
ξ^{n}_{avg}	Average weighted nth-order simple connectivity index
$\xi^{v,n}_{avg}$	Average weighted nth-order valence connectivity index
ξ^{n}_{x}	Nth-order weighted simple connectivity index of test monomer
$\xi^{v,n}_{x}$	Nth-order weighted valence connectivity index of test monomer
δ^{n}	Nth-order simple atomic connectivity index
$\delta^{v,n}$	Nth-order simple atomic connectivity index