IMPLEMENTATION OF MULTIVARIATE ARTIFICIAL NEURAL NETWORKS
COUPLED WITH GENETIC ALGORITHMS FOR THE MULTI-OBJECTIVE
PROPERTY PREDICTION AND OPTIMIZATION OF EMULSION POLYMERS

A Thesis

presented to

the Faculty of California Polytechnic State University,

San Luis Obispo

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Polymers and Coatings Science

by

David Chisholm

June 2019

© 2019

David Chisholm

ALL RIGHTS RESERVED

COMMITTEE MEMBERSHIP

TITLE: Implementation of Multivariate Artificial
Neural Networks Coupled with Genetic
Algorithms for the Multi-Objective Property
Prediction and Optimization of Emulsion
Polymers

AUTHOR: David Chisholm

DATE SUBMITTED: June 2019

COMMITTEE CHAIR: Erik Sapper, Ph.D.
Assistant Professor of Chemistry and
Biochemistry

COMMITTEE MEMBER: Raymond Fernando, Ph.D.
Director, Western Coatings Technology Center

COMMITTEE MEMBER: Gregory Williams, Ph.D.
Section Manager, Behr Paint Company R&D

ABSTRACT

Implementation of Multivariate Artificial Neural Networks Coupled with Genetic Algorithms for the Multi-Objective Property Prediction and Optimization of Emulsion Polymers

David Chisholm

Machine learning has been gaining popularity over the past few decades as computers have become more advanced¹. On a fundamental level, machine learning consists of the use of computerized statistical methods to analyze data and discover trends that may not have been obvious or otherwise observable previously. These trends can then be used to make predictions on new data and explore entirely new design spaces. Methods vary from simple linear regression to highly complex neural networks, but the end goal is similar. The application of these methods to material property prediction and new material discovery has been of high interest as many researchers have begun using the structure-property relationships of materials in conjunction with computational modeling to discover new materials with novel chemical and physical properties²⁻⁸.

One such class of materials is that of emulsion polymers, which are heavily used in the coatings industry as they serve as the binder in many waterborne coating systems⁹⁻¹⁰. The great advantage of these materials is that they are synthesized in water at high solids (30-70%) and therefore are largely compliant with stringent environmental regulations. The chemistry of these polymers is highly variant, but the predominant chemistries include copolymers of styrene and acrylic monomers such as n-butyl acrylate or copolymers of only acrylic monomers. Due to the high degree of complexity and variability of these systems, modeling their behavior according to structure-property relationships is currently

impractical. Instead, this thesis will demonstrate the use of supervised machine learning methods in conjunction with genetic algorithms to predict and optimize emulsion polymer performance based on recipe composition. These emulsion polymers will also be evaluated for use in concrete coatings meant to be applied with minimal preparation work, i.e. no etching.

ACKNOWLEDGMENTS

I would like to acknowledge and thank my advisor, Dr. Erik Sapper for his support throughout my academic career at Cal Poly as well as during my thesis project work. He encouraged me to work independently and as a result I learned a great deal about programming and how to implement machine learning methods in polymer research.

In addition, I would also like to thank Behr Paint Company for allowing me the opportunity to conduct my Master's Thesis work in their facilities in Santa Ana, CA. The polymer group there was really excellent and I thank them for their support in sharing their materials and lab space. A special thanks to my supervisor at Behr, Dr. Greg Williams, for his exceptional support. His example has shaped how I conduct research, interpret results, and continue to seek improvement in all aspects. I am grateful to have had the opportunity to work with him for my Master's internship as well as previously during my undergraduate years at the University of California, Irvine.

TABLE OF CONTENTS

	Page
LIST OF TABLES	x.
LIST OF FIGURES	xii.
 CHAPTER	
1. INTRODUCTION	1
1.1. Machine Learning Methods	1
1.1.1. Regression Modeling	1
1.1.2. Classification Modeling	3
1.1.3. Artificial Neural Network Modeling	4
1.1.4. Genetic Algorithms	7
1.2. Emulsions	9
1.2.1. Surfactants	9
1.2.2. Emulsion Stabilization	12
1.2.3. Polymerizable Surfactants	16
1.3. Emulsion Polymers	19
1.3.1. Emulsion Polymerization	19
1.3.2. Copolymerization	21
1.4. Objectives	26
2. EXPERIMENTAL METHODS	27
2.1. Emulsion Polymer Synthesis	27
2.2. Characterization Methods	28
2.2.1. Quality Check (QC) Properties	28

2.2.2. Thermal Properties	28
2.3. Test Methods	30
2.3.1. Cross-Hatch Adhesion	30
2.3.2. Accelerated Dirt Pick-Up Resistance	31
2.3.3. Hot Tire Pick-Up Resistance	32
2.3.4. Pendulum Hardness	33
2.4. Computational Methods	34
2.4.1. Data Preparation and Processing	34
2.4.2. Artificial Neural Networks and Training	34
3. RESULTS AND DISCUSSION	37
3.1. Initial Resin Synthesis	37
3.2. Computer Modeling	44
3.2.1. Artificial Neural Network Performance	44
3.2.2. Genetic Algorithm Development	46
3.3. Neural Network and Genetic Algorithm Prediction Evaluation	49
3.3.1. Neural Network Prediction Evaluation	49
3.3.2. Genetic Algorithm Prediction Evaluation	51
3.4. Neural Network Improvements	54
3.4.1. Neural Network Retraining	54
3.4.2. Neural Network Prediction Validation	55
3.5. Graphical User Interface (GUI)	60
3.6. Resin Incorporation into Coatings	63
4. CONCLUSIONS	64

BIBLIOGRAPHY	65
--------------------	----

LIST OF TABLES

Table	Page
1. Hot Tire Pick-Up Resistance Rating Scale	33
2. List of Categorical and Continuous Input Variables	34
3. List of Categorical and Continuous Output Variables	35
4. Resin Recipes' Adjusted Factors for the First DOE, All Loadings are Weight Percentages Based on the Total Monomer Loading	38
5. Resin Property Data for Continuous Variables	39
6. Resin Property Data for Categorical Variables	40
7. Final Prediction Errors and Accuracies of the Artificial Neural Networks Trained on the First DOE Data Set	45
8. Polymer Recipes Used to Evaluate the Performance of the Neural Networks, All Loadings are with Respect to Total Monomer	49
9. Predicted and Measured Values for the Properties of the Validation Batches	50
10. Desired Performance Criteria for Each of the Runs of the Genetic Algorithm	51
11. Polymer Recipes Found by the Genetic Algorithm to Have the Highest Fitness Scores, All Loadings Shown are with Respect to Total Monomer ..	52
12. Predicted and Measured Values for the Properties of the Validation Batches with the Predicted Fitness Scores and the Actual Fitness Scores ..	53
13. Final Prediction Errors and Accuracies of the Neural Networks Trained Using the First DOE Data Set and the Validation Batches Data Set	54

14. Second Set of Polymer Recipes Used to Evaluate the Performance of the Neural networks, All Loadings are with Respect to Total Monomer	56
15. Variations in the Validation Batches Used to Test the Neural Networks’ Capabilities	57
16. Differences Between Predicted and Measured Values for the Properties of the Validation Batches	57
17. Average Error of Predictions Compared to the Average Model Errors	58
18. Final Prediction Errors and Accuracies of the Neural Networks Trained Using the First DOE Data Set and All Validation Batches	59
19. Adhesion Performance of 30% Volume Solids Paints Formulated at Various PVC Seven Days After Application to Unprepared Concrete	63

LIST OF FIGURES

Figure	Page
1. Simple Linear (Top) and Polynomial Regression (Bottom) Example Graphs	2
2. Conversion of a List of Categorical Variables to a List of Dummy Variables	3
3. General Structure of an Artificial Neural Network	5
4. Example of a Weight Vector Within a Neural Network	6
5. Schematic Representation of How a Genetic Algorithm Sequentially Determines the Correct Sequence of Genes	8
6. Oil-in-Water Emulsion Stabilized by Surfactant Molecules	9
7. Surfactant in Water Just Below (Left) and Above (Right) its CMC	10
8. Chemical Structure and Shape of Sodium Dodecyl Sulfate	11
9. Schematic Representation of the Potential Distribution as a Function of Distance from the Surface of a Charged Particle	13
10. Schematic Representation of DLVO Interactions; the Sum of the Attractive and Repulsive Potential Energy Curves Result in the Total Potential Energy Curve	14
11. Example Structure of a Non-Ionic Surfactant	15
12. Example Structure of a Surfactant That Would Exhibit Electrostatic and Steric Stabilization	16
13. Surfactant Leaching of a Painted Wall	17
14. Example Structure of a Polymerizable Surfactant	17

15. Common Acrylic Monomers Used in Emulsion Polymerization	20
16. Pictorial Representations of Heterogeneous and Homogeneous Nucleation in the First Stage of Emulsion Polymerization	21
17. Pictorial Representation of the Particle Growth Stage of Emulsion Polymerization	22
18. Relevant Reactions and Their Rates Needed to Determine the Monomer Reactivity Ratios of a Given Pair of Monomers	24
19. Summarization of Reactivity ratios of Terminal and Penultimate Models ..	25
20. Cross-Hatch Adhesion Test Substrate Appearance After Cutting the Film .	30
21. Cross-Hatch Adhesion Rating Scale from ASTM D3359-97	30
22. Hot Tire Pick-Up Resistance Press Set to 22mm with a Sample Inside	32
23. Example Form of a ReLU Activation Function	36
24. Chemical Structure of EDTA	42
25. Pictorial Representation of How k-Fold Cross-Validation Splits a Dataset of Six Observations into Three Folds and Configures the Data into Training and Test Sets	44
26. Continuous Neural Network Property Prediction Errors (Top) and Categorical Neural Networks Property Prediction Accuracies (Bottom) As a Function of Folds	45
27. Example of How Recipes Would be Scored by the Fitness Function According to Specified Criteria	47
28. Training Results for the Second Training of the Neural Networks Using the Data from the First DOE and the Validation Batches	55

29. Training Results for the Third Training of the Neural Networks Using All of the Collected Data from the First DOE and All Validation Batches	59
30. Home Screen of the GUI Used to Predict Polymer Properties and Recipes	60
31. Property Predictor Window (Left) with Input Recipe Components and Predictions Window (Right) with the Corresponding Properties	61
32. Recipe Predictor Window (Left) with Input Property Criteria and Predictions Window (Right) with the Best Recipe Found by the Algorithm	62

1. Introduction

1.1. Machine Learning Methods

1.1.1. Regression Modeling

Machine learning is a catch-all term used to describe the use of statistical techniques and methods on computers to model relationships between variables that are known and unknown, the two main types of machine learning being unsupervised and supervised. In unsupervised machine learning, the output variables are not known, with the goal being to model the structure present in a given set of data. Conversely, in supervised machine learning, the types of output variables and what they should be are known¹¹.

In all supervised machine learning schemes, the input and output variable sets that will be modeled are split into training sets and test sets. The size of the training and test sets determines how well the model can learn¹², with an 80-20 split being the common practice for most models. The model is first trained on the training input and provided output variables before its accuracy is tested against the test set, which is meant to mimic new data that has not been seen by the model¹³. The ultimate goal of this supervised learning scheme is to develop a function that best describes the relationship between the input variables and output variables. The type of function that is fit to the data can be varied with the simplest form being a straight line.

For simple linear regression¹⁴ using the least squares method, the aim is to find a straight line that minimizes S , the sum of the square of the distances between the predicted Y-value and the actual Y-value, defined as the residual (Δ), the equations for which is shown in equation 1 and 2, where the actual value is y and the predicted value is $\hat{y}$.

$$S = \sum_{i=1}^n \Delta_n^2 \quad (1)$$

$$\Delta = (y - \hat{y}) \quad (2)$$

A simplified example of a linear regression of a data set with only one input and one output variable can be seen in the top graph of **Figure 1**. While this example has only one input variable, the least squares regression method is not limited solely to systems that have only one input variable¹⁵. When there are multiple input variables, instead of having only one summation to consider, there is one summation for every input variable. The resultant sum of these sums is then what is minimized instead of the single summation.

Figure 1. Simple Linear (Top) and Polynomial Regression (Bottom) Example Graphs

In addition to linear regression using straight lines, data can also be modeled using curved lines such as polynomials as shown in the bottom graph of **Figure 1**, where the data is modeled using a three degree polynomial. Though the line is curved, the concept remains the same: the distance between the predicted and actual values is minimized to give the best fitting line. While these methods are useful for modeling trends in output data that is numerical, they cannot model trends in data that is categorical.

1.1.2. Classification Modeling

Classification models differ from regression models in that the output variables they predict are categorical in nature and not numerical. These models tend to be more complex than regression models, as the categorical variables are represented by vectors of ones and zeros called dummy variables¹⁶. For instance, trying to predict whether or not a given polymer will be able to form a film at room temperature would be a classification problem and the schematic in **Figure 2** illustrates how a small list of outputs from that test with two unique values would be transformed into dummy variables.

Categorical Output Values	Encoded Dummy Variables	
Pass	1	0
Fail	0	1
Fail	0	1
Pass	1	0
Pass	1	0
Fail	0	1

Figure 2. Conversion of a List of Categorical Variables to a List of Dummy Variables

Once the data is properly encoded, the implementation of algorithms such as decision tree and support vector machine can be done, which essentially determine how to sort the data into the output categories based on the input variables.

1.1.3. Artificial Neural Network Modeling

Artificial neural networks (ANNs) are among the more advanced machine learning methods, even placed in their own class of machine learning named deep learning. This technique can be either supervised or unsupervised and these neural networks can take any number of categorical and continuous input variables and utilize them to predict either categorical¹⁷ outputs using classification or continuous¹⁸ outputs using regression, though both types of outputs cannot be modelled in the same neural network.

Neural networks are highly useful when the precise nature of the relationship between variables is either unknown or not desired to be known¹⁹, essentially acting as a black box that simply takes inputs and gives outputs. The relationships a neural network develops between variables can be modelled indirectly, however, by having the model predict outputs over a range of inputs incrementally and then plotting the results against the input variables in question.

Figure 3. General Structure of an Artificial Neural Network

The general structure of an artificial network, meant to mimic that of the human brain, is shown in **Figure 3**, where the model has a layer of input neurons, one or more hidden layers, and lastly an output layer. The number of hidden layers and the number of neurons, also known as nodes, in each of those layers is a matter of hyperparameter optimization in any given system. Too many hidden layers can result in overfitting²⁰, where the model memorizes the data set instead of generalizing and learning the correlations between the variables. The best way to prevent overfitting for any machine learning model is to collect more data in an effort to provide a high degree of variance within the dataset. For continuous variables, this means having data points that are relatively far apart, covering a larger area of possible values, whereas for categorical variables, this means collecting data for all of the possible categories to prevent biases from developing. A dataset for particle sizes of latexes that has high variance, for example, might have particle sizes ranging from 100nm to 1000nm. If the difference between the largest particle size and the smallest particle size was only 10nm, then the dataset would not have a high

variance and would therefore be difficult to model. If the variance of the model is too high without having a lot of data points to fill in the gaps between points, however, then local minima and maxima could be missed.

Artificial neural networks are trained in a similar fashion to that of simpler models: the data set is split into training and test sets, the model learns correlations based on the training set, and then the model's performance is tested on the test set. The major difference between neural networks and simpler models is how the training step is done and the procedure used to do this is known as the optimization algorithm²¹. There are several different optimization algorithms, with gradient descent²² being the simplest and therefore widely utilized. In this scheme, the goal is to minimize the loss function by adjusting the weights applied to the nodes within the network. This is accomplished in a process called backpropagation²³.

Figure 4. Example of a Weight Vector Within a Neural Network²³

In backpropagation, the error vectors of each weight are back propagated throughout the neural network starting from the output layer. The weights are then adjusted and the process is repeated until the loss function has been minimized. Once training has

been completed, the networks can then be evaluated using the test set and then used to predict new outcomes in the same manner as other machine learning methods.

The advantage of using artificial neural networks over other machine learning models is that they are highly accurate models that essentially encompass all of the other machine learning models. Neural networks tend to require more data than other models but are able to handle many input and output variables, both categorical and continuous, more efficiently than other models. The more input and output variables that are being modeled, however, the longer the training and optimization of the networks will take and the more computationally expensive the overall process will be.

1.1.4. Genetic Algorithms

Similar to the backpropagation algorithms used in training neural networks, genetic algorithms are tools used for the targeted optimization of specific variables with constraints based on the theories of natural selection and evolution²⁴. In most genetic algorithms, the first generation or set of genes is randomly selected from a larger set of genes making the parent. This parent set is then evaluated in a fitness function and if the fitness is less than the defined optimal fitness, the genes will be mutated. The way in which the mutation is done can be varied depending on the nature of the problem²⁵, but essentially one of the genes in the parent will be randomly selected and switched with another gene that was randomly selected from the gene pool. This second generation, or child, is then evaluated against the fitness function again and this process is done iteratively until an optimal fitness is reached as illustrated in **Figure 5**.

Figure 5. Schematic Representation of How a Genetic Algorithm Sequentially Determines the Correct Sequence of Genes

In the example above, both the order and identity of the letters matter and as such the first sequence has a fitness score of two and not three, even though it has two C's and a B like the target sequence. If only the identity of the letters mattered and not the order, then the fitness score of the first set of genes would have been four instead of two. Every genetic algorithm is different and how the genes are selected, mutated, and finally evaluated in the fitness function is highly important and determines how efficiently the algorithm will perform²⁶.

In this work, a customized genetic algorithm was written to be used in a reverse-engineering sense to determine an emulsion polymer recipe that meets desired performance attributes. The fitness function of the genetic algorithm uses predictions made by artificial neural networks that have been previously trained on emulsion polymer data. The validity of the networks' property predictions will be evaluated.

1.2. Emulsions

1.2.1. Surfactants

Emulsions are essentially dispersions of one phase of material in a larger continuous phase of another, i.e. hexane in water and as mentioned previously, the emulsions in this work consist of polymer particles dispersed in aqueous media. These systems differ greatly from solutions, as the phases in these mixtures are completely distinct from one another and are not homogeneously mixed as they are in solution. If desired, the two could be separated rather easily, whereas this is not the case with solutions. Depending on the hydrophilicity of the dispersed polymer particles, some amount of water will be able to enter the particles, however, the vast majority of the water remains outside of the particles²⁷. As the two phases do not want to mix with one another due to the polarity differences between them, emulsions are inherently unstable and will eventually separate into two different phases.

Figure 6. Oil-in-Water Emulsion Stabilized by Surfactant Molecules

The instability of these emulsions can be overcome, however, by the addition of surfactants, or surface active agents as shown in **Figure 6**. These molecules are amphiphilic

molecules having both hydrophilic and hydrophobic portions which have been shown to preferentially position themselves at interfaces between different phases of materials²⁸⁻³⁰. While at these interfaces, the free energy of the surfactants is minimized as both portions of the molecule have favorable interactions as they are surrounded by similar species. These surfactants are not locked in place, however, having the ability to move across the emulsified droplet surface and even migrate from one emulsion droplet to another neighboring droplet.

Figure 7. Surfactant in Water Just Below (Left) and Above (Right) its CMC

Surfactants do not form micelles immediately upon addition to a liquid, however, and some do not form micelles at all. In order for a surfactant to form micelles in a given system, its concentration in that system needs to be above its critical micellar concentration, defined as the concentration above which a surfactant will begin to form micelles³¹. When a surfactant is first added to a liquid, the molecules will migrate to the air-liquid interface and the interfaces between the container and the liquid as shown in **Figure 7**. As more surfactant is added, these interfaces eventually become completely saturated by the surfactant molecules, having no more room to accommodate additional surfactant

molecules. Adding additional surfactant above this point will result in micelle formation in the liquid phase in the form of spherical aggregates of surfactant molecules. For surfactants added to water, or another polar solvent, the hydrophobic tails will face toward one another, with the hydrophilic portion pointed out into the water phase. For a surfactant added to a non-polar solvent, the orientation of the surfactant would be reversed, having the hydrophilic portion facing inwards and the hydrophobic portion facing out into the solvent phase.

Conventional surfactants used for the stability of oil-in-water emulsions consist of a hydrophilic head attached to a hydrophobic tail, usually an alkyl chain. The chemical structure of surfactants can be highly variant as the major requirement for a molecule to act as a surfactant is for it to be amphiphilic, having both a hydrophilic portion and a lipophilic portion. Not all molecules that are amphiphilic will make good emulsifying agents in all systems, however, as the hydrophile-lipophile balance (HLB) value and the surfactant number (N_s) of each candidate need to be considered³².

The HLB value indicates the relative hydrophilicity and conversely the lipophilicity of a given surfactant whereas the surfactant number indicates what type of shape the surfactant molecule adopts. Sodium dodecyl sulfate, for instance, as shown in **Figure 8**, has a higher HLB value and it adopts a cone-like shape, making it better suited to stabilize spherical oil-in-water emulsions.

Figure 8. Chemical Structure and Shape of Sodium Dodecyl Sulfate

Surfactants that have higher HLB values tend to have higher critical micellar concentrations as these are more hydrophilic and therefore a higher concentration of the surfactant molecules can remain in the water phase before needing to collapse into micelles to lower their free energy. The higher the hydrophobicity of a given surfactant, the lower the concentration of surfactant needed to form micelles in water³³, whereas the reverse is true in non-polar solvents like hexane or xylene. Regardless of which surfactant is selected for any given system, the main modes of stabilization they can provide to an emulsion include electrostatic and steric stabilization.

1.2.2. Emulsion Stabilization

The use of ionic surfactants, like sodium dodecyl sulfate, would contribute to electrostatic stabilization³⁴ whereas non-ionic surfactants with long hydrophilic chains would contribute to steric stabilization³⁵. When the ionic surfactants are added to a system of dispersed polymer particles, the surfactants' hydrophobic tails are able to adsorb onto the polymer particle surface, adding a layer of charges to the particles, dubbed the Stern Layer³⁶. This layer of charges can either be positive or negative, depending on the surfactants used. In the case of polymer particles stabilized by sodium dodecyl sulfate, the charge would be negative due to the sulfate group's negative charge. In addition to this layer of charge, electrolytes in the system with opposite charges to the layer can associate to the Stern Layer, forming a second layer of charges, known as the Diffuse Layer. This layer is not as strongly associated to the particles, with its electric potential dropping off as the distance from the particles gets larger as shown in **Figure 9**. The sum of the Stern and

Diffuse Layers is known as the Electric Double Layer and this double layer gives the particles an overall effective charge otherwise known as the zeta potential.

Figure 9. Schematic Representation of the Potential Distribution as a Function of Distance from the Surface of a Charged Particle³⁷

When particles that have like charges approach one another there is an electrostatic repulsive force generated which keeps the particles apart, in accordance with DLVO Theory³⁸. There is an attractive force due to Hamaker attractions which gets larger as the particles get closer to one another, however, the electrostatic repulsive force is greater, resulting in a net stabilization of the particles. As shown in **Figure 10**, as the distance between charged particles gets smaller, the sum of the attractive and repulsive interactions results in the potential energy of the system going up, which is highly dis-favorable. This energy barrier is proportional to the magnitude of the zeta potential that the particles have and this prevents the agglomeration of particles in the system.

Figure 10. Schematic Representation of DLVO Interactions; the Sum of the Attractive and Repulsive Potential Energy Curves Result in the Total Potential Energy Curve

The stabilization of these particles can be reduced significantly, however, if ions from salts such as sodium chloride are added to the system³⁹. When these ions are added, the thickness of the electric double layer of the particles gets reduced, resulting in a decrease in the zeta potential of the particles. Since the particles do not have as high a charge as they had previously, the magnitude of the electrostatic force that prevents them from coming together is proportionally lowered, lowering the energy barrier and making it easier for the particles to agglomerate. The concentration of ions needed to completely destabilize a given emulsion or suspension of particles is defined as the critical coagulation concentration⁴⁰. At this concentration of ions, the zeta potential of the particles is essentially lowered to zero, resulting in rapid agglomeration of the particles due to their being no energy barrier and ultimately complete separation of the two phases. In addition to concentration of ions, the charge of those ions plays a large role as well⁴¹. Ions with higher charges, i.e. +2 or +3, have a much larger destabilizing effect at the same

concentration than ions with a charge of +1. In certain situations this is very useful, i.e. cleaning murky water to make it safe to drink⁴², however, this destabilization is not normally desired in the case of polymer emulsions.

Figure 11. Example Structure of a Non-Ionic Surfactant

Non-ionic surfactants with water soluble chains of ethylene oxide units, such as the one shown in **Figure 11**, are able to avoid this ion-induced destabilization because they stabilize emulsions via an entropic mechanism. When the ethylene oxide chains are pointed out into the water phase, the chains have a large degree of conformational entropy, meaning the chains can take on many different conformations in the water phase without being hindered⁴³. If two particles that have these surfactants come close to one another, the total number of possible conformations that the ethylene oxide chains of the surfactants can adopt is lowered, resulting in a decrease in the system's entropy and therefore an increase in the system's free energy. Since this is dis-favorable, a force is generated which prevents the particles from agglomerating in a similar fashion to that of ionic surfactants. The ethylene oxide chain length plays a large role in this stabilization and the longer the ethylene oxide chains, the better the stabilization it will provide to a given emulsion⁴⁴.

Figure 12. Example Structure of a Surfactant that Would Exhibit Electrostatic and Steric Stabilization

A common practice is to use both non-ionic and ionic surfactants together in the same emulsion system to provide better stabilization⁴⁵ by utilizing both stabilization mechanisms. Surfactants are also commercially available with structures similar to the one shown in **Figure 12** that combine the two stabilization mechanisms into one molecule by having a hydrophobic group that is a chain of ethylene oxide units capped with an ionic group. This class of surfactants has the advantage of exhibiting both stabilization mechanisms in one molecule and although they are still susceptible to lessened stabilization if more electrolytes are added to the system, the added electrolyte ions are less likely to cause complete destabilization of the dispersion.

1.2.3. Polymerizable Surfactants

One potential drawback to using single molecule surfactants in these systems is that they can leach out of the final polymer films and can negatively affect the appearance and performance of the film. One such defect is known as surfactant leaching, in which surfactant molecules will migrate out of a coating's film when it gets wet, resulting in streaks on the coating's surface⁴⁶ as shown in **Figure 13**.

Figure 13. Surfactant Leaching of a Painted Wall⁴⁶

These issues can be overcome, however, through the use of surfactants that are polymerizable instead of conventional surfactants, an example structure of which is given in **Figure 14**. Polymerizable surfactants used for the stabilization of emulsion polymers are molecules that have reactive groups somewhere in the hydrophobic portion of the molecule that can participate in the radical polymerization reaction.

Figure 14. Example Structure of a Polymerizable Surfactant

Since these surfactants have unsaturated double bonds, they are able to react with the growing polymer chain radicals and as a result these surfactants will end up in the polymer backbone⁴⁷, assuming they have comparable reactivity to that of the monomers in the system. To better match the reactivity and ensure proper random distribution of the surfactant molecules in the polymer chain, the reactive group in the surfactant should match

that of monomers. For example, a polymerizable surfactant meant to stabilize a vinyl polymer emulsion should have a vinyl group. If the emulsion is acrylate-based then the reactive group should be an acrylate. Being covalently linked to the polymer chain, these molecules provide stabilization to the emulsified polymer chains in water just like a traditional surfactant without the undesired ability to leach out once the film has been cast.

1.3. Emulsion Polymers

1.3.1. Emulsion Polymerization

Emulsion polymers can be synthesized in a number of different ways with free-radical emulsion polymerization being the predominant method. This method's mechanism was outlined by Harkins⁴⁸ in the 1940s and its full mechanism is still the subject of some debate, though it's widely accepted that the reaction proceeds through three major stages: nucleation, particle growth, and finally the consumption of monomer. The vast majority of emulsion polymerizations are conducted in water, with monomer being the dispersed phase, as their end applications, i.e., paint resins, require them to be water-based systems. The major components needed for these emulsion polymerizations include monomer, surfactant, and a water-soluble initiator. Other components can be added to aid in processing, such as defoamer to reduce foam formation or chain transfer agents for molecular weight control⁴⁹⁻⁵⁰, but they are not necessarily required.

Surfactant molecules stabilize the monomer droplets in water. A variety of surfactant chemistries can be used, with nearly all having higher HLB values. The water-soluble initiator, as the name implies, is responsible for initiating the polymerization reaction by generating radicals which can react with monomer present in the system. The most common initiators used in emulsion polymerization are thermal initiators, which are molecules which break apart upon heating to generate the radicals. In order for the monomers to react with the initiator radicals the monomers need to have unsaturated double bonds. The most common monomers used for emulsion polymerizations of paint resin polymers are acrylate monomers such as those shown in **Figure 15**. Vinyl monomers can

also be used, but monomers that react with water should be avoided, as they can destabilize the emulsion or even prevent it from forming at all.

Figure 15. Common Acrylic Monomers Used in Emulsion Polymerization

The first stage of free-radical emulsion polymerization, particle nucleation, can begin by either homogeneous nucleation or heterogeneous nucleation. In homogeneous nucleation, initiator radicals in the water phase react with the small amount of monomer molecules that are present⁵¹. Once the growing chain becomes too large to be soluble in the water phase, surfactant molecules will then surround the polymer chain, forming a small micelle which can then continue growing in the particle growth stage. In heterogeneous nucleation, monomer molecules present in either the monomer droplets or the water phase diffuse into small surfactant micelles that do not have any monomer. Initiator radicals can then enter these smaller monomer micelles and react with the monomer present, forming a growing polymer chain. These micelles will then continue growing in the particle growth stage of the polymerization. While the path to get there is

different, the end result of both mechanisms is the same for both types of nucleation, illustrated in **Figure 16**.

HETEROGENEOUS NUCLEATION

HOMOGENEOUS NUCLEATION

Figure 16. Pictorial Representations of Heterogeneous and Homogeneous Nucleation in the First Stage of Emulsion Polymerization

Once all of the surfactant molecules in the system are in micelles and there is none left to form new micelles, the second stage, particle growth, begins. As depicted in **Figure 17**, during this stage, the total number of particles within the system is fixed, meaning no new particles are formed, they simply grow larger. The most accepted mechanism for this stage states that monomer molecules diffuse out of the monomer droplets into the water phase and then into the growing polymer chain micelles, subsequently reacting⁵².

Figure 17. Pictorial Representation of the Particle Growth Stage of Emulsion Polymerization

As the polymer particles increase in size, the monomer droplet micelles get depleted and the surfactant molecules that were stabilizing those monomer droplets migrate into the polymer particle micelles. The final stage of the polymerization begins when all of the monomer droplet micelles have been depleted and all surfactant molecules have migrated to polymer particles.

At the start of the final stage, the majority of the remaining monomer molecules are present inside the polymer particles with a small portion in the water phase. The monomers inside the polymer particles get consumed by the growing polymer chains while the monomers that are in the water phase have to diffuse into the particles before they can be consumed as well. Radical scavengers, known as chasers, can be added at this stage to ensure higher conversion of monomers, but they are not necessarily needed in every system.

1.3.2. Copolymerization

Nearly all emulsion polymers are copolymers, meaning their composition includes more than one monomer species. Meeting desired performance attributes for coatings applications using only one monomer is highly unlikely and therefore multiple monomers are incorporated to adjust the polymers' chemical and physical properties such as the glass transition temperature (T_g). An emulsion polymer intended for use in a coating to be applied at room temperature, for example, would need to have a glass transition temperature in the range of 0-5°C to minimize the amount of coalescent needed for the paint to form a cohesive film. The T_g of any copolymer can be calculated using the Fox Equation, equation 3, where T_g is the glass transition temperature of the copolymer, W_1 and W_2 are the weight fractions of each of the monomers, and T_{g1} and T_{g2} are the glass transition temperatures of the homopolymers of the respective monomers.

$$\frac{1}{T_g} = \frac{W_1}{T_{g1}} + \frac{W_2}{T_{g2}} \quad (3)$$

The equation above can be applied to any number of monomers and monomer compositions only if the resulting copolymer is a random distribution of the monomers used to make it. If the polymer does not have a random composition, then the equation above is not valid. In order to ensure the correct polymeric composition, the monomer reactivity ratios need to be taken into account, whose values are dependent on the rates of each of the reactions shown in **Figure 18**.

Figure 18. Relevant Reactions and Their Rates Needed to Determine the Monomer Reactivity Ratios of a Given Pair of Monomers

The monomer reactivity ratio⁵³ is defined as the ratio of the rate of monomer one (M1) addition to a monomer one radical (M1*) divided by the rate of monomer two (M2) addition to a monomer one radical (M1*). If a monomer reacts more often with the radical analogue of itself than that of the other monomer radical, then it would have a high reactivity ratio. This would not be an ideal scenario for a random distribution of monomers in a copolymer, however, as the monomers have a preference over which radical they add to. In order for a copolymer to have a random composition, there needs to be little to no preference in monomer addition.

These equations are based on the terminal model and this model ignores the influence of neighboring monomers on the reactivity of the radical, stating that the reactivity of a radical is independent of what that radical is attached to⁵⁴. This is an approximation used to simplify the calculations involved in the kinetics, but in reality the neighboring monomers do have an effect and other models have been put forth which attempt to capture the impacts that neighboring monomers have such as the penultimate model.

Growing Chain	Monomer	Rate Constant	New Structure	Reactivity Ratios
Terminal				
-1	1	k_{11}	-11	$(r_1) = k_{11} / k_{12}$
-2	1	k_{21}	-21	
-1	2	k_{12}	-12	$(r_2) = k_{21} / k_{22}$
-2	2	k_{22}	-22	
Penultimate				
-1-1	1	k_{111}	-111	$r_1 = k_{111} / k_{112}$
-2-1	1	k_{211}	-211	
-2-2	1	k_{221}	-221	$r_1' = k_{211} / k_{212}$
-1-2	1	k_{121}	-121	
-1-1	2	k_{112}	-112	$r_2 = k_{222} / k_{221}$
-2-1	2	k_{212}	-212	
-2-2	2	k_{222}	-222	$r_2' = k_{122} / k_{121}$
-1-2	2	k_{122}	-122	

Figure 19. Summarization of Reactivity Ratios of Terminal and Penultimate Models⁵⁴

While the terminal model assumes that the reactivity of a radical is independent of its neighbor, in the penultimate model, the monomer next to the radical is also considered. This results in a total of eight possible propagation reactions and four reactivity ratios in a binary copolymerization system, instead of four propagation reactions and two reactivity ratios. A summarization of the terminal and penultimate models can be seen in **Figure 19**.

1.4. Objectives

The overall objective of this work was to develop machine learning models to predict and optimize the properties of emulsion polymers meant for application in concrete coatings. These coatings are meant to be applied to concrete with minimal prep, i.e., no etching or priming.

2. Experimental Methods

2.1. Emulsion Polymer Synthesis

All of the polymers in this study were synthesized using a proprietary starve-fed emulsion polymerization procedure, allowing for controlled particle growth as well as lower polydispersity of particle sizes⁵⁵. Sometimes referred to as semi-continuous batch polymerization, this technique involves the simultaneous slow feeding of pre-emulsified monomer in water and initiator solution into a reaction vessel containing a smaller solution of surfactant and water. Pre-emulsions were prepared one day prior to batch making to test for stability. In order for a pre-emulsion to be considered stable, it needs to be stable for the duration of the pre-emulsion's feeding time during synthesis, which for these batches is four hours. Separation of the emulsified layers overnight was considered acceptable as long as the layers could be easily re-emulsified through mixing.

2.2. Characterization Methods

2.2.1. Quality Check (QC) Properties

Polymer physical properties were evaluated including: density, particle size, weight percent solids, and pH. Density was determined using a pycnometer and pH was measured using a two-point calibrated pH probe. Polymer pH was measured before and after correction with ammonia to a range of 8.5-9. Weight percent solids for the polymers was measured using a solids analyzer. Wet samples were put into a tray and an initial mass was taken. The samples were then heated from 60°C to 150°C and held at 150°C until the mass of the tray no longer changed. The percent solids, then, were expressed as a percentage of final mass divided by initial mass. Particle size was measured via dynamic light scattering using a Nano-S Zetasizer from Malvern with a scattering angle of 173°.

2.2.2. Thermal Properties

Differential scanning calorimetry (DSC) was conducted on the emulsion polymers using a DSC 214 Polyma® from Netzsch to determine their glass transition temperatures. Drawdowns of the polymers were made using a standard 3-mil drawdown bar on Leneta release charts and allowed to dry for at least one day before testing. Squares were cut from the dried films and tested in the DSC. To mitigate the effects of water and erase the sample thermal history⁵⁶⁻⁵⁷, all samples were annealed at 105°C for two minutes before the testing temperature range was applied. A typical temperature range for the testing interval was -20°C to 50°C, as the theoretical glass transition temperatures of the polymers ranged between -5°C and 10°C.

The minimum film formation temperature of the emulsion polymers were determined using an MFFT Bar – 90 from Gardco. Wet samples were applied to the bar using a 75 micron drawdown cube and the resulted films were evaluated for cracking. The MFFT was determined to be the temperature at which no cracks are observed in the dried film on the bar. Samples were evaluated with the bar set to the temperature range 0-18°C.

2.3. Test Methods

2.3.1. Cross-Hatch Adhesion

Cross-hatch adhesion was conducted according to a modified version of ASTM D3359-97⁵⁸. The emulsion polymers were applied as-is using a natural spread rate, onto a block of unprepared, smooth concrete using a foam brush. Adhesion was then evaluated on these films in the following manner, at times of one day and seven days after the films were applied.

Figure 20. Cross-Hatch Adhesion Test Substrate Appearance after Cutting the Film

Two sets of squares were cut into the films using a razor blade and a 3-mm cutting guide as shown in **Figure 20**. The set of squares on the left was for dry adhesion and the one on the right was for wet adhesion. For wet adhesion, a square piece of a paper towel was cut and placed onto the set of squares on the right. The towel was saturated with water and left on the squares for ten minutes. The paper towel was then removed and the excess water was dabbed off using a paper towel and the squares were allowed to sit for an additional ten minutes.

Following this, pieces of adhesion tape were stuck to both sets of squares and force was applied to ensure the tape was indeed stuck to all of the individual squares. The tape was then removed in a swift upward motion. Both wet and dry adhesion were rated based on how much of the coating was removed according to the scale shown in **Figure 21**.

Surface of cross-cut area from which flaking has occurred. (Example for 6 parallel cuts)	None					Greater than 65%
Classification	5	4	3	2	1	0
<p>5: The edges of the cuts are completely smooth; none of the squares of the lattice is detached.</p> <p>4: Small flakes of the coating are detached at intersections; less than 5 % of the area is affected.</p> <p>3: Small flakes of the coating are detached along edges and at intersections of cuts. The area affected is 5 to 15 % of the lattice.</p> <p>2: The coating has flaked along the edges and on parts of the squares. The area affected is 15 to 35 % of the lattice.</p> <p>1: The coating has flaked along the edges of cuts in large ribbons and whole squares have detached. The area affected is 35 to 65 % of the lattice.</p> <p>0: Flaking and detachment worse than Grade 1.</p>						

Figure 21. Cross-Hatch Adhesion Rating Scale from ASTM D3359-97⁵⁸

2.3.2. Accelerated Dirt Pick-Up Resistance

The application of the resins for accelerated dirt-pick up resistance (ADPUR) was conducted in the same manner as that of adhesion. Resins were applied to an unprepared block of smooth concrete via foam brush. Seven days after application, synthetic black dirt was placed on top of the films and let sit for two hours. After two hours, the dirt was removed from the block by tilting the blocks so that the dirt ran off. Any remaining standing dirt on the films was removed by blowing air lightly over the samples. Each of the sample's

performance was rated on a 0-5 scale, with a score of zero being very dirty and a score of five having very little to no dirt.

2.3.3. Hot Tire Pick-Up Resistance

Hot tire pick-up resistance of the resins was measured one day and seven days after application onto 3x3" concrete tiles. The tiles were split in two after being applied in the same manner as the samples for adhesion and accelerated dirt-pick up resistance. A rectangular piece of a tire was dipped in water and then placed on top of the concrete tile. The two were then placed into a press which was then depressed to 21mm, measured from the tip of the top vertical bar to the line indicated on the bottom vertical bar, shown in **Figure 22**.

Figure 22. Hot Tire Pick-Up Resistance Press Set to 21mm with a Sample Inside

The whole apparatus was then placed in a 65°C oven for one and a half hours. Following this, the apparatus was removed from the oven and the press was unscrewed to remove the tile and tire. Samples were rated based on how difficult it was to remove the tire from the surface of the coating as well as the condition of the coating after tire removal in accordance with the scale shown in **Table 1**.

Table 1. Hot Tire Pick-Up Resistance Rating Scale

Surface Condition	Rating Scale
No sticking, no imprint, no delamination	10
No sticking, slight imprint, no delamination	9
Slight sticking, slight imprint, no delamination	8
Moderate to slight sticking, slight imprint, no delamination	7
Moderate sticking, slight imprint, no delamination	6
Moderate sticking, moderate imprint, no delamination	5
Moderate sticking, slight imprint, slight delamination	4
Moderate sticking, moderate imprint, slight delamination	3
Severe sticking, moderate imprint, slight delamination	2
Severe sticking, severe imprint, slight delamination	1
Severe sticking, severe imprint, delamination	0

2.3.4. Pendulum Hardness

Resins were drawn-down on aluminum panels to a wet film thickness of 3-mils using a draw-down bar. The pendulum hardness of the films was measured as swings using a TQC Pendulum hardness tester from Gardco set to 6°.

2.4. Computational methods

2.4.1. Data Preparation and Processing

Before the neural networks could be trained on the data set, the input and output variables needed to be scaled and encoded. Recipe variables such as surfactant and adhesion promoter and their respective chemistries were encoded as dummy variables and loading levels were scaled to a 100 scale so that no values were greater than one. This was done to improve the efficiency of the neural networks in predictions as well as training. A complete list of both continuous and categorical input variables is listed in **Table 2**.

Table 2. List of Categorical and Continuous Input Variables

Variable	Variable Type
Surfactant Chemistry	Categorical
Surfactant Loading	Continuous
Adhesion Promoter Chemistry	Categorical
Adhesion Promoter Loading	Continuous
Bulk Monomer 1 Loading	Continuous
Bulk Monomer 2 Loading	Continuous
Acid Monomer Loading	Continuous
Theoretical T_g	Continuous
Theoretical Weight Percent Solids	Continuous

2.4.2. Artificial Neural Networks and Training

Two different types of artificial neural networks were developed to model the properties of the emulsion polymers: one for continuous properties and one for categorical properties. A list of the continuous and categorical outputs and their respective units is provided in **Table 3**. Dry and wet adhesion, dirt pick-up resistance and hot tire pick-up resistance were all evaluated seven days after application to the substrate.

Table 3. List of categorical and continuous output variables

Property	Property Type	Property Units/Scales
Percent Recovered	Continuous	%
Actual T _g	Continuous	Kelvin
MFFT	Continuous	Kelvin
Particle Size	Continuous	Nanometers
Weight % Solids	Continuous	%
Koenig Hardness	Continuous	Swings
Pre-emulsion Stability	Categorical	Pass/Fail
Dry Concrete Adhesion	Categorical	0-5, 5 best
Wet Concrete Adhesion	Categorical	0-5, 5 best
Dirt Pick-up Resistance	Categorical	0-5, 5 best
Hot Tire Pick-up Resistance	Categorical	0-10, 10 best

All modeling and computational work was conducted in Python, using Keras and TensorFlow as the main packages for the neural networks. The continuous properties were all modeled using a single neural network while separate classification networks were trained for each of the categorical properties resulting in a total of six neural networks. All of the neural networks were trained using a modified k-fold strategy in which the data set was split k times into training and test folds and then the models were trained and evaluated sequentially on each of the folds in the data. Optimal model hyperparameters including activation functions, number of hidden layers, number of nodes etc. were determined using a combination of trial and error and grid searching to minimize the amount of error in the model predictions.

For nearly all of the networks, the activation function applied to the weight vectors from the input layer to the hidden layer was ReLU⁵⁹. ReLU, or rectified linear units, is a rectifier function that is zero for all values less than zero and positive for all values greater than zero, with the general form shown in **Figure 23**.

Figure 23. Example Form of a ReLU Activation Function⁵⁹

The activation function that was applied to the weight vectors going from the hidden layer to the output layer varied depending on the nature of the outputs being modeled. For categorical outputs with more than two categories, such as adhesion, the SoftMax activation function was used while the Sigmoid activation function was used for the network modeling pre-emulsion stability.

3. Results and Discussion

3.1. Initial Resin Synthesis

As with all neural networks and machine learning models, a data set is needed with which to train and evaluate the model. In order to generate a data set in a systematic way, a standard design of experiments, or DOE⁶⁰, was conducted by varying adhesion promoter chemistry and their loadings as well as surfactant chemistry. All of the surfactants used in this DOE were at 2% loading based on the total monomer amount and were polymerizable; the variations between them are the end group. All polymers were formulated to have a theoretical glass transition temperature around 5°C and had an additional 1% of a conventional non-polymerizable anionic surfactant. The full list of variations in recipes for the first DOE is presented in **Table 4** and their resultant properties are listed in **Tables 5** and **6**.

Table 4. Resin Recipes' Adjusted Factors for the First DOE, All Loadings are Weight Percentages Based on the Total Monomer Loading

Resin	Adhesion Promoter General Chemistry	Promoter Loading	Surfactant General Chemistry
1	None	0%	Sulfate
2	None	0%	Non-ionic
3	None	0%	Phosphate
4	Phosphate	1%	Sulfate
5	Phosphate	1%	Non-ionic
6	Phosphate	1%	Phosphate
7	Phosphate	2%	Sulfate
8	Phosphate	2%	Non-ionic
9	Phosphate	2%	Phosphate
10	Ureido	0.5%	Sulfate
11	Ureido	0.5%	Non-ionic
12	Ureido	0.5%	Phosphate
13	Ureido	1%	Sulfate
14	Ureido	1%	Non-ionic
15	Ureido	1%	Phosphate
16	Alkoxysilane	1%	Sulfate
17	Alkoxysilane	1%	Non-ionic
18	Alkoxysilane	1%	Phosphate
19	Alkoxysilane	2%	Sulfate
20	Alkoxysilane	2%	Non-ionic
21	Alkoxysilane	2%	Phosphate
22	Ureido	1%	Sulfate
23	Ureido	1%	Non-ionic
24	Ureido	1%	Phosphate
25	Ureido	2%	Sulfate
26	Ureido	2%	Non-ionic
27	Ureido	2%	Phosphate

Batches 10-15 all had lower ureido loadings than intended due to an error in formulation. The monomer was assumed to be 100% active, however, it turned out to be only 50% active and therefore six additional batches were conducted to account for this at the correct loading levels.

3.1.1. Resin Physical Properties and Performance

Table 5. Resin Property Data for Continuous Variables

Resin	% Recovered	Actual T _g (K)	MFFT (K)	PS (nm)	Wt % Solids	Hardness (swings)
1	86.4	284.8	278.2	136	49.14	15
2	87.6	284.4	279.8	157	51.19	14
3	90.2	285.3	283.7	143	48.89	28
4	87.4	291.8	281.2	149	49.30	19
5	85.8	290.9	281.8	152	51.39	16
6	92.2	292.3	282.6	135	47.85	21
7	92.4	286.6	280.4	137	50.01	17
8	89.2	287.4	280.8	142	48.83	13
9	93.2	289.0	281.6	140	48.74	20
10	88.0	290.6	281.4	155	49.88	16
11	88.1	287.8	281.0	156	49.98	13
12	90.7	289.6	282.6	160	49.96	23
13	82.6	285.6	281.6	137	52.96	19
14	91.6	286.9	281.6	218	49.84	15
15	90.0	288.3	284.0	140	49.50	19
16	88.0	283.6	281.0	142	49.95	20
17	89.1	288.4	281.4	163	50.71	17
18	90.4	288.8	281.2	129	49.74	23
19	90.7	292.7	282.0	140	48.73	22
20	90.2	291.3	282.0	139	48.92	17
21	91.7	292.8	282.0	134	48.02	24
22	86.4	290.0	279.4	147	48.88	16
23	88.4	296.3	279.4	188	48.23	15
24	88.6	294.3	279.4	152	48.87	22
25	73.1	290.9	281.6	191	53.05	17
26	79.4	289.1	281.6	203	49.84	13
27	0.0	NA	NA	NA	NA	NA

Table 6. Resin Property Data for Categorical Variables

Resin	Pre-Emulsion Stability	Dry Adhesion	Wet Adhesion	DPUR	HTPUR
1	Stable	5	4	0	1
2	Stable	5	4	0	2
3	Stable	4	4	1	2
4	Stable	5	5	0	0
5	Stable	5	5	0	1
6	Stable	5	5	2	0
7	Stable	5	2	0	1
8	Stable	4	2	0	0
9	Stable	5	5	2	0
10	Stable	5	5	0	0
11	Stable	5	5	0	1
12	Stable	5	5	1	0
13	Stable	4	4	0	5
14	Stable	4	2	0	1
15	Stable	2	1	1	0
16	Stable	5	5	0	6
17	Stable	5	5	0	3
18	Stable	5	5	1	8
19	Stable	5	5	0	2
20	Stable	5	4	0	0
21	Stable	5	5	0	6
22	Stable	5	5	0	7
23	Stable	5	5	0	6
24	Stable	5	5	2	2
25	Stable	5	5	0	1
26	Stable	5	5	0	1
27	Unstable	NA	NA	NA	NA

The only batch in the first DOE that was unstable was Batch 27, which had 2% ureido adhesion promoter and 2% phosphate surfactant. Batch 27 was stable for the initial four hours, but separated into two layers overnight and would not re-emulsify; therefore, the batch was not run and no other data was recorded for it in the table above. If the batch were to be run and the pre-emulsion separated during feeding, then the composition of the final polymers in the latex particles would not be uniform, and therefore not comparable to the other polymers in the study. The instability of the batch was likely due to higher

hydrophilicity of the surfactant. Though it has nearly the same chemistry and structure as the other surfactants, it does have higher solubility in water and as such would not be expected to be as efficient as the other two surfactants at stabilizing emulsions⁶¹. Other batches within this DOE also separated overnight, but upon mixing re-emulsified without issue.

Initially, adhesion was tested on both etched and unetched concrete, however, all samples were able to pass both wet and dry adhesion on etched concrete. For most concrete coatings to adhere to concrete substrates, the concrete needs to be etched with an acid solution such as hydrochloric acid in order to provide a sufficiently rough surface. As the roughness of the concrete's surface increases, the adhesion of the coating proportionally improves⁶² as there is more surface area available for the polymer to interact with the concrete. In addition to providing roughness, etching also increases the porosity of the concrete, allowing for better penetration of the coating into the substrate, also improving the adhesion performance. Since there was no variation in performance observed between samples on etched concrete, it was not pursued, instead opting for adhesion to unetched concrete. Both the dry and wet adhesion ratings recorded in **Table 6**, then, are adhesion to unprepared, smooth concrete.

The lack of etching makes the samples here truly stand out, as adhesion to bare concrete is notoriously difficult. Surprisingly, even the negative control polymers, batches without any adhesion promoters, adhered quite well to the concrete, all receiving scores of at least 4 for both wet and dry adhesion. This could be due to the presence of carboxylic acid monomer in the polymer. Molecules with carboxylic acid groups such as

ethylenediaminetetraacetic acid (EDTA), shown in **Figure 24**, have been shown to chelate to calcium when deprotonated⁶³⁻⁶⁸.

Figure 24. Chemical Structure of EDTA

As calcium chloride is used in the manufacture of concrete⁶⁹, it is logical to infer that the carboxylic acid groups in the polymer would be able to chelate to the calcium ions in the concrete, providing some degree of adhesion through those ionic interactions. The phosphate adhesion promoter is meant to do the same, able to chelate to many different types of metals and thereby provide improved adhesion to inorganic substrates. So it stands to reason that the carboxylic acid functional groups would be able to do the same but to a lesser degree as there is only one negative charge to interact with the substrate while the phosphate group has two.

None of the samples exhibited good dirt pick-up resistance, while some exhibited decent hot tire pick-up resistance. In general, polymers will either have good dirt pick-up resistance and bad hot tire pick-up resistance or the reverse. The reason for this is because dirt pick-up resistance is more a function of hardness while hot tire pick-up resistance is more a function of flexibility. The higher the glass transition temperature of a polymer, the better the dirt pick-up resistance will be, however, the hot tire pick-up resistance will be

weaker because the film will not be able to flex with the tire and therefore will easily delaminate from the substrate when the tire is removed. One way to overcome this is to lower the surface energy of the polymer, which would decrease the attractive forces between the surfaces of the tire and the polymer film⁷⁰, thereby reducing the likelihood of the polymer sticking to the tire at all. The monomers that would be used to do this, i.e. fluorocarbons, are highly expensive, however, and tend to be more toxic and difficult to work with, limiting their potential implementation.

3.2. Computer Modeling

3.2.1. Artificial Neural Network Performance

Once all of the data had been collected for the first set of resins, the first set of artificial neural networks were trained using a modified k-fold cross-validation method. In normal k-fold cross-validation, a set of data is split into folds and then these folds are used in different combinations to train and evaluate models to test the skill of an overall model as illustrated in **Figure 25**. In this process, a new model is trained and evaluated at each configuration of the data and then discarded.

Figure 25. Pictorial Representation of How k-Fold Cross-Validation Splits a Dataset of Six Observations into Three Folds and Configures the Data into Training and Test Sets

The method used to train the models was similar to cross-validation, however, instead of training new models in each iteration, the same model was trained iteratively on each of the configurations of the data, ideally lowering the error in predictions with each configuration. The final error and accuracies of the neural networks' predictions and graphs of the prediction errors and accuracies as a function of folds are displayed in **Table 7** and **Figure 26** respectively.

Table 7. Final Prediction Errors and Accuracies of the Artificial Neural Networks Trained on the First DOE Data Set

Property	Error & Accuracy of Models' Predictions
% Recovered	± 2
Actual T_g (K)	± 3
MFFT (K)	± 2
Particle Size (nm)	± 7
Wt % Solids	± 1
Hardness (swings)	± 2
Dry Adhesion	100%
Wet Adhesion	100%
Dirt Pick-Up Resistance	100%
Hot Tire Pick-Up Resistance	100%

Figure 26. Continuous Neural Network Property Prediction Errors (Top) and Categorical Neural Networks Property Prediction Accuracies (Bottom) as a Function of Folds

Although the errors in predictions for the continuous variables are low and the accuracies of the classifications of categorical variables are high, one major issue with this first data set is that it is not variant in terms of adhesion and dirt pick-up resistance and is therefore biased⁷¹. Almost all of the resins had good adhesion and all of the resins had poor dirt pick-up resistance. After being trained on this skewed data set, the models will likely end up predicting that all future polymer recipes will result in good adhesion and poor dirt pick-up resistance. While this may be the case within the design space of the DOE, it is not likely and this is most definitely not the case for all possible polymer recipes. There are bound to be polymers that have terrible adhesion and polymers that have exceptional dirt pick-up resistance, but the models cannot recognize this as they were not trained on data that reflects this.

3.2.2. Genetic Algorithm Development

After training, the neural networks were then encoded into a genetic algorithm to be used to discover polymer recipes to match specified performance attributes. The gene set that the algorithm pulled from was a list of recipe components including monomer loadings, surfactant, surfactant loading, adhesion promoter, etc. These components were then encoded using the same scaling and categories as the data that the models were previously trained on to ensure proper predictions. Once encoded, the recipe was then passed to a custom function that employed the neural networks to predict the performance values for that recipe which were combined into a vector and passed to the fitness function.

The fitness function then compared the values in the predicted vector to those in the target vector. Each of the properties were evaluated and given a binary score of either one, if the value met the specified criteria, or zero if the value did not meet the specified criteria. These scores were stored in a vector and the sum of that vector was defined as the fitness score for that recipe. If the fitness score was less than the defined optimal fitness, then the algorithm would mutate the recipe, switching one of the components' values with a different value, and then the performance values for that recipe would be determined and subsequently evaluated in the fitness function again. This process was done iteratively until an optimal fitness score was reached. An example of how recipes' performance values would be scored by the fitness function is provided in **Figure 27**. The first recipe receives a score of two out of ten as only two of the properties meet the fitness criteria while the second recipe receives a score of five out of ten because five of its properties match the fitness criteria defined in the second table.

Property	Target Values	Fitness Criteria	Recipe 1 Predicted Properties	Fitness 1	Recipe 2 Predicted Properties	Fitness 2
% Recovered (%)	90	± 2 or $\geq$ Target	85	0	92	1
Actual T_g (K)	278.2	± 1	288.2	0	279.2	1
MFFT (K)	273.2	± 1	278.2	0	276.2	0
PS (nm)	100	± 5	136	0	110	0
Wt % Solids (%)	50	± 2	48.75	1	49.2	1
Hardness (swings)	20	± 1	19	1	30	0
Dry Adhesion	5	$\geq$ Target	4	0	5	1
Wet Adhesion	5	$\geq$ Target	4	0	4	0
DPUR	5	$\geq$ Target	0	0	2	0
HTPUR	10	$\geq$ Target	6	0	8	0
Total Fitness	10		2		4	

Figure 27. Example of How Recipes Would be Scored by the Fitness Function According to Specified Criteria

In order to make the algorithm more customizable and applicable by an end user, the option to prefer specific properties was introduced. For instance, if one wanted to make a polymer that had a particle size of 150nm but did not care what the other performance values were, then they could specify that particle size was to be preferred above all other properties. When the algorithm then evaluates the recipe's performance values in the fitness function, only the particle size would be considered, meaning the optimal fitness for the perfect recipe would be one. In addition to property preferences, the ability to adjust the tolerances in the fitness function was also added to the algorithm. This feature applies more to the precision of the predictions, meaning if one wanted to predict a recipe for a polymer that had a particle size of 150nm plus or minus 10nm versus plus or minus 20nm. One could also simply specify that the particle size needed to be between 100nm and 200nm, instead of inputting a specific value.

3.3. Neural Network and Genetic Algorithm Prediction Evaluation

3.3.1. Neural Network Prediction Evaluation

In order to test the validity of the models' prediction accuracies and their ability to extrapolate, polymers were synthesized that had recipes intentionally outside of the current design space of the models. While surfactant loading and theoretical glass transition temperature were held constant in the first DOE, these recipe components were deliberately varied in the second round of resins. The full list of recipes for the second set of resins are shown in **Table 8** and the predicted properties and experimentally determined properties are shown in **Table 9**.

Table 8. Polymer Recipes Used to Evaluate the Performance of the Neural Networks, All Loadings are with Respect to Total Monomer

	Validation 1	Validation 2	Validation 3
Surfactant Chemistry	Non-ionic	Sulfate	Non-ionic
Surfactant Loading	1.0%	1.0%	5.0%
Adhesion Promoter Chemistry	Alkoxysilane	Alkoxysilane	Alkoxysilane
Adhesion Promoter Loading	0.5%	0.5%	1.0%
Bulk Monomer 1 Loading	52.274%	55.8%	51.898%
Bulk Monomer 2 Loading	46.226%	42.7%	46.102%
Acid Monomer Loading	1.0%	1.0%	1.0%
Theoretical T_g	273.15	268.15	273.15
Theoretical % Solids	51.61%	51.61%	52.61%

Table 9. Predicted and Measured Values for the Properties of the Validation Batches

	Validation 1		Validation 2		Validation 3	
	Predicted Values	Measured Values	Predicted Values	Measured Values	Predicted Values	Measured Values
% Recovered (%)	86.0	87.7	86.59	85.6	87.04	83.9
Actual T _g (K)	283.21	283.35	281.45	283.15	286.23	277.95
MFFT (K)	277.57	273.15	274.81	273.15	278.99	273.15
Particle Size (nm)	153	156	139	138	151	153
Wt % Solids (%)	48.27	51.18	48.89	48.44	47.89	50.89
Hardness (swings)	19	6	17	3	20	5
Dry Adhesion	5	1	5	0	5	0
Wet Adhesion	4	0	5	0	4	0
DPUR	0	0	0	0	0	0
HTPUR	8	7	8	9	8	8

As expected, the neural networks predicted that all of these batches would have good adhesion while in reality they all had very poor adhesion to the concrete. Since the models were trained on recipes that only had good adhesion, the models inferred that all recipes would have good adhesion, though this is clearly not the case. In addition, the models predicted the hardness would be much higher than the hardness ended up being. Here again, the recipes that the models were trained on all had higher hardness values due to their glass transition temperatures being higher, resulting in another bias. In general, hardness has been shown to increase as the glass transition temperature of the polymer increases⁷²⁻⁷³. Varying the glass transition temperature in these batches was needful, as the correlation between the glass transition temperature and the hardness was not established in the network, this correlation can now be learned when the models are retrained with this data.

Rather surprisingly, the measured particle size values were very close to the predicted values, lying well within the error of predictions shown previously in **Table 7**. Even though particle size is definitely related to the amount of surfactant present in the

emulsion⁷⁴⁻⁷⁵, the two being inversely proportional, these results indicate that surfactant loading is not necessarily the biggest factor and is certainly not the only one. If surfactant concentration was the sole factor responsible for determining the final particle size of the emulsion polymer, then the predictions would not have been very accurate as the models were not trained on data with varying surfactant loadings.

3.3.2. Genetic Algorithm Prediction Evaluation

To test the genetic algorithm, it was run three separate times to determine recipes for emulsion polymers to meet the performance specified in **Table 10**. The second and third runs had the same criteria in the hopes of demonstrating the ability to get to the same performance with different polymer recipes.

Table 10. Desired Performance Criteria for Each of the Runs of the Genetic Algorithm

	Run 1 Desired Performance	Run 2 Desired Performance	Run 3 Desired Performance
% Recovered (%)	≥ 85	≥ 85	≥ 85
Actual T_g (K)	288.15 ± 1	288.15 ± 1	288.15 ± 1
MFFT (K)	273.15 ± 1	279.15 ± 1	279.15 ± 1
Particle Size (nm)	150 ± 5	140 ± 5	140 ± 5
Wt % Solids (%)	50 ± 2	50 ± 2	50 ± 2
Hardness (swings)	≥ 20	≥ 15	≥ 15
Dry Adhesion	≥ 4	≥ 4	≥ 4
Wet Adhesion	≥ 4	≥ 4	≥ 4
DPUR	≥ 1	≥ 1	≥ 1
HTPUR	≥ 6	≥ 6	≥ 6

In the event that no recipe was found which met the specified criteria and to prevent the algorithm from running indefinitely, a time limit and a limit to the number of predictions were imposed on the algorithm for each of the runs. The maximum number of

predictions allowed per run was set to 10000 and the time limit was set for three hours. When either of these limits were reached, the algorithm would simply stop, having stored all of the recipes and their predicted performance values in a data frame. The recipes could then be sorted and sifted through to determine which recipes would give the most apt results.

Unfortunately, none of the runs of the genetic algorithm resulted in an optimal fitness score of ten; instead, the recipe with the highest fitness score in each of the runs was the one that was synthesized, whose recipes are shown in **Table 11**. The predicted and measured properties for these batches are shown in **Table 12**.

Table 11. Polymer Recipes Found by the Genetic Algorithm to Have the Highest Fitness Scores, All Loadings Shown are with Respect to Total Monomer

	Recipe 1	Recipe 2	Recipe 3
Surfactant Chemistry	Phosphate	Phosphate	Sulfate
Surfactant Loading	1.46%	1.97%	1.13%
Adhesion Promoter Chemistry	Ureido	Phosphate	Alkoxysilane
Adhesion Promoter Loading	0.19%	0.25%	0.37%
Bulk Monomer 1 Loading	54.94%	44.89%	54.74%
Bulk Monomer 2 Loading	43.87%	53.86%	43.89%
Acid Monomer Loading	1.0%	1.0%	1.0%
Theoretical T_g	269.86	284.66	269.77
Theoretical % Solids	51.84%	52.10%	51.68%

Table 12. Predicted and Measured Values for the Properties of the Validation Batches with the Predicted Fitness Scores and the Actual Fitness Scores

	Recipe 1		Recipe 2		Recipe 3	
	Predicted Values	Measured Values	Predicted Values	Measured Values	Predicted Values	Measured Values
% Recovered (%)	87.38%	92.3%	88.36%	92.7%	86.39%	87.8%
Actual T _g (K)	282.97	276.35	281.65	300.35	281.1	277.65
MFFT (K)	278.54	273.15	276.57	288.15	276.79	273.15
Particle Size (nm)	147.28	144.73	137.42	133.17	140.06	132
Wt % Solids (%)	48.26%	47.04%	48.24%	50.2%	48.91%	50.4%
Hardness (swings)	25	4	24	41	18	4
Dry Adhesion	4	0	5	3	5	0
Wet Adhesion	4	0	4	1	4	0
DPUR	1	0	2	1	0	0
HTPUR	6	2	6	6	3	6
FITNESS	8	3	8	4	7	3

Unlike the previous validation recipes, there are greater differences observed between the predicted and measured values for these batches. Some properties, such as particle size, are predicted quite well, having errors lying within the prediction errors of the neural networks; however, the differences between predicted and measured adhesion and hardness are not even close for most of the batches. This demonstrates the neural networks inability to extrapolate correctly as all of these recipes are outside of the design space in which the neural networks were trained. The data for these batches along with the previous validation batches were added to the original data set and all of the neural networks were subsequently retrained to enlarge upon the design space that they model.

3.4. Neural Network Improvements

3.4.1. Neural Network Retraining

In order to improve upon the neural networks' predictive capabilities, they were retrained with the second set of resins included in the dataset using the modified k-fold method again. In this training, however, the networks were trained using the leave-one-out method⁷⁶. In this method, the data is split such that only one data point is used to test the model. For example, if the entire data set is 100 data points, then the data set would be split into 99 training points and 1 test point. This dataset would be split into 100 folds, meaning every data point would have the opportunity to be a test point. The results of this training method are shown in **Table 13** and in the graphs in **Figure 28**. Their prediction errors and classification accuracies were similar to that of the previous models, but the models are expected to be better able to recognize the fact that not every recipe will result in a polymer with good adhesion or high hardness.

Table 13. Final Prediction Errors and Accuracies of the Neural Networks Trained Using the First DOE Data Set and the Validation Batches Data Set

Property	Error & Accuracy of Models' Predictions
% Recovered	± 2
Actual T_g (K)	± 3
MFFT (K)	± 3
Particle Size (nm)	± 7
Wt % Solids	± 3
Hardness (swings)	± 2
Dry Adhesion	100%
Wet Adhesion	100%
Dirt Pick-Up Resistance	100%
Hot Tire Pick-Up Resistance	100%
Pre-Emulsion Stability	100%

Figure 28. Training Results for the Second Training of the Neural Networks Using the Data from the First DOE and the Validation Batches

3.4.2. Neural Network Prediction Validation

A second set of validation batches, whose recipes are in **Table 14**, were prepared to again test the models' performance and to further elucidate the impact of surfactant concentration on particle size.

Table 14. Second Set of Polymer Recipes Used to Evaluate the Performance of the Neural Networks, All Loadings are with Respect to Total Monomer

	Validation 4	Validation 5	Validation 6
Surfactant Chemistry	Phosphate	Phosphate	Phosphate
Surfactant Loading	2.0%	5.0%	10.0%
Adhesion Promoter Chemistry	Alkoxysilane	Alkoxysilane	Alkoxysilane
Adhesion Promoter Loading	1.0%	1.0%	1.0%
Bulk Monomer 1 Loading	48.49%	48.47%	48.44%
Bulk Monomer 2 Loading	49.47%	49.42%	49.33%
Acid Monomer Loading	1.0%	1.0%	1.0%
Theoretical T_g	278.2	278.2	278.2
Theoretical % Solids	50.00%	51.11%	51.11%

When the pre-emulsions for the batches were prepared, only Validation 4 was able to form a stable emulsion. As the unstable batch in the previous DOE had been largely ignored, these batches prompted the development of another neural network, responsible for predicting pre-emulsion stability. Since 5% of the phosphate surfactant was already unstable, it was assumed that any amount above this would result in an unstable emulsion. At these higher loadings of surfactant, the adhesion promoter would not be expected to offer any kind of stabilization to the emulsion, in fact in most cases adding adhesion promoter actually destabilized the emulsions slightly as they are more hydrophilic in nature than the other monomers.

Accordingly, a dataset was generated for unstable batches that included all of the adhesion promoters in combination with 4% to 11.5% of the phosphate surfactant in 0.5% increments at a fixed monomer combination for a total of 144 simulated data points. The monomer combination was fixed as different combinations of monomers will have varying hydrophobicities and as such the level of phosphate surfactant at which those emulsions

will be unstable will likely be variant. This simulated data and the data for the previous batches were then used to train the pre-emulsion stability network.

To test the models' performance again, twelve more batches were synthesized with the variations shown in **Table 15**. Adhesion promoter loading and glass transition temperature were held constant at 1% and 2°C respectively for all batches. The errors between the predicted values and the measured values and how their averages compare to the model average errors are shown in **Tables 16** and **17** respectively.

Table 15. Variations in the Validation Batches Used to Test the Neural Networks' Capabilities

	Adhesion Promoter Chemistry	Surfactant Chemistry	Surfactant Loading
Validation 1	Alkoxysilane	Sulfate	1%
Validation 2	Alkoxysilane	Sulfate	3%
Validation 3	Alkoxysilane	Sulfate	6%
Validation 4	Alkoxysilane	Non-ionic	1%
Validation 5	Alkoxysilane	Non-ionic	3%
Validation 6	Alkoxysilane	Non-ionic	6%
Validation 7	Ureido	Non-ionic	1%
Validation 8	Ureido	Non-ionic	3%
Validation 9	Ureido	Non-ionic	6%
Validation 10	Ureido	Sulfate	1%
Validation 11	Ureido	Sulfate	3%
Validation 12	Ureido	Sulfate	6%

Table 16. Differences Between Predicted and Measured Values for the Properties of the Validation Batches

	R1 Error	R2 Error	R3 Error	R4 Error	R5 Error	R6 Error	R7 Error	R8 Error	R9 Error	R10 Error	R11 Error	R12 Error
% Recovered	-2.97	-3.43	-8.77	-5.78	-5.70	-7.40	-3.81	-5.88	-10.94	-7.35	-7.51	-16.23
Actual T _g	2.30	0.57	-2.99	3.48	1.79	-2.70	6.66	10.96	4.05	5.10	2.66	-2.86
MFFT	2.44	1.40	-0.88	-1.39	-2.38	-2.77	1.41	2.17	-0.91	2.56	1.12	-1.74
Particle Size	15.14	16.07	20.43	2.65	-1.18	-5.45	11.12	11.42	14.92	0.21	22.16	38.65
Wt% Solids	-1.76	-3.41	-9.81	2.01	2.94	2.14	0.65	2.43	1.15	-0.87	-1.82	-2.31
Hardness	4	7	4	1	0	-1	-2	4	2	6	7	5
Dry Adhesion	1	0	0	0	0	0	0	0	0	0	0	0
Wet Adhesion	0	0	0	0	0	0	0	0	0	0	0	0
DPUR	0	0	0	0	0	0	0	0	0	0	0	0
HTPUR	-1	-1	-1	4	-1	0	0	-1	2	-1	1	-5
PE Stability	0	0	-1	0	0	0	0	0	-1	0	0	-1

Table 17. Average Error of Predictions Compared to the Average Model Errors

	Model Error	Average Prediction Error
% Recovered	± 2%	7.15
Actual T _g	± 3K	3.84
MFFT	± 3K	1.76
Particle Size	± 7nm	13.28
Wt% Solids	± 3%	2.61
Hardness	± 2	3.58
Dry Adhesion	± 1	0.08
Wet Adhesion	± 1	0.00
DPUR	± 1	0.00
HTPUR	± 1	1.50
PE Stability	± 1	0.25

As seen in **Table 17**, on average, most of the error in predictions for the polymer properties are quite close to the error of the networks, though they are individually variant. Though the networks were eventually able to classify the training data without error, it is highly unlikely that the models will be 100% accurate all of the time and to account for this, the errors for these predictions were set to plus or minus one. The predictions for percent recovered for every batch was quite conservative, having each of the measured results being higher than the predicted values. Not surprisingly, the property with the most variation was that of particle size as the measurement of particle size for all of these batches

was conducted using dynamic light scattering and this method has been shown to have a large degree of error compared to other methods⁷⁷⁻⁷⁹.

Having collected even more data, the networks were re-trained, including all data from previous batches. The results from this training are shown in **Figure 29** and **Table 18**.

Figure 29. Training Results for the Third Training of the Neural Networks Using All of the Collected Data from the First DOE and All Validation Batches

Table 18. Final Prediction Errors and Accuracies of the Neural Networks Trained Using the First DOE Data Set and All Validation Batches

Property	Error & Accuracy of Models' Predictions
% Recovered	± 2
Actual T_g (K)	± 3
MFFT (K)	± 3
Particle Size (nm)	± 4
Wt % Solids	± 3
Hardness (swings)	± 2
Dry Adhesion	100%
Wet Adhesion	100%
Dirt Pick-Up Resistance	100%
Hot Tire Pick-Up Resistance	100%

3.5. Graphical User Interface (GUI)

The latest networks were then encoded into a graphical user interface along with the genetic algorithm to afford users the ability to use these prediction tools simply and efficiently. The home screen of the GUI is displayed in **Figure 30**. From this window, the user can choose to either use the property predictor or the recipe predictor.

Figure 30. Home Screen of the GUI Used to Predict Polymer Properties and Recipes

The Property Predictor window, shown in **Figure 31**, predicts polymer properties based on the inputs provided by the user using the neural networks developed previously. If these inputs are not typed correctly, however, an error message will be raised alerting the user to fix the issue. For example, the sum of the monomer loading fields and the adhesion promoter loading field needs to equal 100. If this condition is not met, clicking the Predict button will result in an error message and the prediction will not proceed until corrected. The Total Monomer Check button was added so that the user could verify that the total monomer loading is indeed equal to 100 before trying to predict the properties. Clicking this button will give a pop-up message displaying the current sum of the monomer loadings. Once all recipe components are input correctly and the Predict button is clicked, a new window will pop up with the predicted properties of the given recipe. These predictions can then be exported to an excel file using the Export Results button in the Predictions window.

Figure 31. Property Predictor Window (Left) with Input Recipe Components and Predictions Window (Right) with the Corresponding Properties

The Recipe Predictor window, shown in **Figure 32**, employs the genetic algorithm to predict a polymer recipe to match the property criteria provided by the user. This GUI provides various functionality to the user allowing them to adjust the tolerance of the predictions, select which properties are to be preferred, and define how many iterations the algorithm is allowed to run and how long the algorithm has to complete the task.

Figure 32. Recipe Predictor Window (Left) with Input Property Criteria and Predictions Window (Right) with the Best Recipe Found by the Algorithm

As seen in **Figure 32**, the best recipe that the algorithm found after 1000 recipe iterations only had a fitness score of five, even though the optimal fitness score was seven. If the algorithm was allowed to go through more than 1000 iterations, then it may have found a recipe that met all of property criteria. The time limit and iteration limits were established to prevent the algorithm from running indefinitely in the event that it is unable to find the perfect recipe. Since the algorithm stores all of the recipes that it generates along

with their predicted properties, the user can export all of this data to determine if there were a set of recipes that would be worth trying.

3.6. Resin Incorporation into Coatings

All of the resins from the first DOE, 27 total, were incorporated at 19% volume solids into 28PVC coating formulations with TiO_2 and other fillers for a total of 32% volume solids. The dry and wet adhesion of these paints were evaluated on unprepared smooth concrete. Although many of the polymers had perfect adhesion when applied to concrete neat, none of the paints in this study had any adhesion to the concrete. To elucidate the cause of this failure in adhesion, a ladder study of PVC was conducted using one of the resins, the results for which are shown in **Table 19**.

Table 19. Adhesion Performance of 30% Volume Solids Paints Formulated at Various PVC Seven Days after Application to Unprepared Concrete

PVC	Dry Adhesion	Wet Adhesion
0	5	5
5	0	0
10	0	0
15	0	0
20	0	0
25	0	0
30	0	0

As seen in **Table 19**, adding any amount of pigment to the resin resulted in complete loss of adhesion. Since the functional groups that were added to the resins are intended to bind to inorganic materials, it is possible that they would bind to the pigments.

In doing so, the number of available binding sites per chain would be lowered, resulting in a loss in the benefit of the adhesion promoter. While this limits the use of these resins in pigmented coatings, they could be used as clear primers.

4. Conclusions

The use of artificial neural networks and genetic algorithms to predict and optimize emulsion polymer performance based on its recipe has been demonstrated. A graphical user interface was developed to facilitate efficient use of the neural networks and the genetic algorithm for end users. Although the models were not able to predict all of the properties perfectly, they were able to do so with a fair degree of accuracy. The final data set used was still biased, however, favoring good adhesion and poor dirt pick-up resistance, and would require additional polymer data to eliminate this bias. As these models are able to be continually trained and optimized, their predictive capabilities can be readily improved through the expansion of the design space and preparation of more emulsion polymers.

Since the adhesion performance of these resins is lost when formulated with pigments, their application in pigmented coatings are limited. They could, however, be used as clear primers for other coatings to applied to. This would still eliminate the need for etching and drastically reduce the amount of preparation typically required of concrete coatings.

Bibliography

- (1) Butler, K. T.; Davies, D. W.; Cartwright, H.; Isayev, O.; Walsh, A. Machine Learning for Molecular and Materials Science. *Nature* **2018**, *559*, 547-555.
- (2) Cao, L.; Zhu, P.; Zhao, Y.; Zhao, J. Using Machine Learning and Quantum Chemistry Descriptors to Predict the Toxicity of Ionic Liquids. *J. Haz. Mat.* **2018**, *352*, 17-26.
- (3) Gao, H.; Struble, T. J.; Coley, C. W.; Wang, Y.; Green, W. H.; Jensen, K. F. Using Machine Learning to Predict Suitable Conditions for Organic Reactions. *ACS Central Sci.* **2018**, *4*, 1465-1476.
- (4) Ahneman, D. T.; Estrada, J. G.; Lin, S.; Dreger, S. D.; Doyle, A. G. Predicting Reaction Performance in C-N Cross-Coupling Using Machine Learning. *Science* **2018**, *360*, 186-190.
- (5) Nandy, A.; Duan, C.; Janet, J. P.; Gugler, S.; Kulik, H. J. Strategies and Software for Machine Learning Accelerated Discovery in Transition Metal Chemistry. *Ind. Eng. Chem Res.* **2018**, *57*, 13973-13986.
- (6) Venkatrama, V.; Alsberg, B. K. Quantitative Structure-Property Relationship Modeling of Thermal Decomposition Temperatures of Ionic Liquids. *J. Mol. Liq.* **2016**, *223*, 60-67.
- (7) Venkatrama, V.; Evjen, S.; Knuutila, H. K.; Fiksdahl, A.; Alsberg, B. K. Predicting Ionic Liquid Melting Points Using Machine Learning. *J. Mol. Liq.* **2018**, *264*, 318-326.
- (8) Kang, X.; Qian, J.; Deng, J.; Latif, U.; Zhao, Y. Novel Molecular Descriptors for Prediction of H₂S Solubility in Ionic Liquids. *J. Mol. Liq.* **2018**, *265*, 756-764.
- (9) Tigh, R. S.; Evren, V. Synthesis and Characterization of Pure Poly(acrylate) Latexes. *Prog. Org. Coat.* **2005**, *52*, 144-150.
- (10) Heiskanen, N.; Jamsa, S.; Paajanen, L.; Koskimies, A. Synthesis and Performance of Alkyd-Acrylic Hybrid Binders. *Prog. Org. Coat.* **2010**, *67*, 329-338.
- (11) Supervised vs. Unsupervised Learning
<https://towardsdatascience.com/supervised-vs-unsupervised-learning-14f68e32ea8d> (accessed April 30, 2019)
- (12) Roy, P. P.; Leonard, J. T.; Roy, K. Exploring the Impact of Size of Training Sets for the Development of Predictive QSAR Models. *Chem. Intel. Lab. Sys.* **2008**, *90*, 31-42.

- (13) How to Develop a Machine Learning Model from Scratch
<https://towardsdatascience.com/machine-learning-general-process-8f1b510bd8af>
 (accessed May 4, 2019)
- (14) Introductory Business Statistics with Interactive Spreadsheets
<https://opentextbc.ca/introductorybusinessstatistics/chapter/regression-basics-2/>
 (accessed April 30, 2019)
- (15) Breiman, L.; Friedman J. H. Predicting Multivariate Responses in Multiple Linear Regression. *J. R. Statist. Soc.: Stat. Meth. B* **1997**, *59*, 3-54.
- (16) Smarter Ways to Encode Categorical Data for Machine Learning.
<https://towardsdatascience.com/smarter-ways-to-encode-categorical-data-for-machine-learning-part-1-of-3-6dca2f71b159> (accessed April 30, 2019)
- (17) Dreiseitl, S.; Ohno-Machado, L. Logistic Regression and Artificial Neural Network Classification Models: A Methodology Review. *J. Biomed. Info.* **2002**, *35*, 352-359.
- (18) Tosum, E.; Aydin, K.; Bilgill, M. Comparison of Linear Regression and Artificial Neural Network Model of a Diesel Engine Fueled with Biodiesel-Alcohol Mixtures. *Alex. Eng. J.* **2016**, *55*, 3081-3089.
- (19) Mjalli, F. S.; Al-Asheh, S.; Alfadala, H. E. Use of Artificial Neural Networks Black-Box Modeling for the Prediction of Wastewater Treatment Plants Performance. *J. Env. Man.* **2007**, *83*, 329-338.
- (20) Preventing Deep Neural Networks from Overfitting
<https://towardsdatascience.com/preventing-deep-neural-network-from-overfitting-953458db800a> (accessed May 21, 2019)
- (21) 5 Algorithms to Train a Neural Network
https://www.neuraldesigner.com/blog/5_algorithms_to_train_a_neural_network
 (accessed May 2, 2019)
- (22) Bottou, L. Large-Scale Machine Learning with Stochastic Gradient Descent. *Proceedings of COMPSTAT 2010*, 177-186.
- (23) How the Backpropagation Algorithm Works
<http://neuralnetworksanddeeplearning.com/chap2.html> (accessed May 2, 2019)
- (24) Introduction to Genetic Algorithms – Including Example Code.
<https://towardsdatascience.com/introduction-to-genetic-algorithms-including-example-code-e396e98d8bf3> (accessed May 1, 2019)

- (25) Srinivas, M.; Patnaik, L. M. Adaptive Probabilities of Crossover and Mutation in Genetic Algorithms. *IEEE Trans. Sys., Man Cyb.* **1994**, *24*, 656-667.
- (26) Filho, J. L.; Treleaven, P. C.; Alippi, C. Genetic-Algorithm Programming Environments. *Computer* **1994**, *27*, 28-43.
- (27) Tripathi, A.; Wei, C.; Tauer, K. Swelling of Latex Particles – Towards a Solution of the Riddle. *Col. Poly. Sci.* **2017**, *295*, 189-196.
- (28) Manne, S.; Gaub, H. E. Molecular Organization of Surfactants at Solid-Liquid Interfaces. *Science* **1995**, *270*, 1480-1482.
- (29) Zhang, R.; Somasundaran, P. Advances in Adsorption of Surfactants and Their Mixtures at Solid/Solution Interfaces. *Adv. Col. Inter. Sci.* **2006**, *123*, 213-229.
- (30) Adsorption of Anionic, Cationic and Nonionic Surfactants on Carbonate Rock in Presence of ZrO₂ Nanoparticles. *Physics Procedia* **2011**, *22*, 63-67.
- (31) Dominguez, A.; Fernandez, A.; Gonzalez, N.; Iglesias, E.; Montenegro, L. Determination of Critical Micelle Concentration of Some Surfactants by Three Techniques. *J. Chem. Educ.* **1997**, *74*, 1227-1231.
- (32) Guo, X.; Rong, Z.; Ying, X. Calculation of Hydrophile-Lipophile Balance for Polyethoxylated Surfactants by Group Contribution Method. *J. Col. Inter. Sci.* **2006**, *298*, 441-450.
- (33) Lin, I. J. Hydrophile-Lipophile Balance (hlb) of Fluorocarbon Surfactants and its Relation to the Critical Micelle Concentration (cmc). *J. Phys. Chem.* **1972**, *76*, 2019-2023.
- (34) Landfester, K.; Bechthold, N.; Tiarks, F.; Antonietti, M. Formulation and Stability Mechanisms of Polymerizable Miniemulsions. *Macromolecules* **1999**, *32*, 5222-5228.
- (35) Florence, A. T.; Madsen, F.; Puisieux, F. Emulsion Stabilization by Non-Ionic Surfactants: The Relevance of Surfactant Cloud Point. *J. Pharm. Pharmac.* **1975**, *27*, 385-394.
- (36) Brown, M. A.; Goel, A.; Abbas, Z. Effect of Electrolyte Concentration on the Stern Layer Thickness at a Charged Interface. *Angew. Chem. Int. Ed.* **2016**, *55*, 3790 –3794.
- (37) Us'yarov O. G. The Electrical Double Layer of Micelles in Ionic Surfactant Solutions in the Presence of a Background Electrolyte. *Col. Journal*, **2007**, *69*, 95-102.

- (38) Croll, S. DLVO Theory Applied to TiO₂ Pigments and Other Materials in Latex Paints. *Prog. Org. Coat.* **2002**, *44*, 131-146.
- (39) Lawrence, H. A.; Egon, M. Stability of Colloidal Silica: I. Effect of Simple Electrolytes. *J. Col. Inter. Sci.* **1969**, *31*, 287-296.
- (40) Hsu, J. P.; Yu, H. Y.; Tseng, S. Critical Coagulation Concentration of a Salt-Free Colloidal Dispersion. *J. Phys. Chem. B.* **2006**, *110*, 7600-7604.
- (41) Moncho-Jorda, A.; Martinez-Lopez, F.; Hidalgo-Alvarez, R. The Effect of the Salt Concentration and Counterion Valence on the Aggregation of Latex Particles at the Air/Water Interface. *J. Col. Inter. Sci.* **2002**, *249*, 405-411.
- (42) Faust, S. D.; Aly, O. M. *Chemistry of Water Treatment, Second Edition*; CRC Press, 1998.
- (43) Lindman, B.; Medronho, B.; Karlstrom, G. Clouding of Nonionic Surfactants. *Cur. Opin, Col. Inter. Sci.* **2016**, *22*, 23-29.
- (44) Cheng, H.; Zhang, H.; Liu, X.; Lin, M.; Qin, Z.; Fang, Y. Effect of Polyoxyethylene Chain Length on the Physicochemical Properties of N,N-Dimethyl-N-dodecyl Polyoxyethylene Amine Oxide Hybrid Surfactants (C₁₂EO_nAO, with n = 1-4). *J. Surf. Deterg.* **2015**, *18*, 487-493.
- (45) Chern, C. S.; Lin, S. Y.; Chen L. J.; Wu, S. C. Emulsion Polymerization of Styrene Stabilized by Mixed Anionic and Nonionic Surfactants. *Polymer* **1997**, *38*, 1977-1984.
- (46) Let Me Tell You About “Surfactant Leaching”
<http://www.joshuaosenstock.com/arr-haus/let-me-tell-you-about-surfactant-leaching> (accessed April 30, 2019)
- (47) Mekki, S.; Saidi-Besbes, S.; Elaissari, A.; Valour, J. P.; Derdour, A. Novel Polymerizable Surfactants: Synthesis and Application in Emulsion Polymerization of Styrene. *Polymer Journal* **2010**, *42*, 401-405.
- (48) Harkins, W. D. A General Theory of the Mechanism of Emulsion Polymerization. *J. Am. Chem. Soc.* **1947**, *69*, 1428-1444.
- (49) Nomura, M.; Minamino, Y.; Fujita, K. The Role of Chain Transfer Agents in the Emulsion Polymerization of Styrene. *J. Pol. Sci.: Pol. Chem. Ed.* **1982**, *20*, 1261-1270.

- (50) Tjiam, C.; Gomes, V. G. Optimal Operating Strategies for Emulsion Polymerization with Chain Transfer Agent. *Ind. Eng. Chem. Res.* **2014**, *53*, 7526-7537.
- (51) Hansen, F. K.; Ugelstad, J. Particle Nucleation in Emulsion Polymerization. I. A Theory for Homogeneous Nucleation. *J. Pol. Sci.: Pol. Chem. Ed.* **1978**, *16*, 1953-1978.
- (52) Vanderhoff, J. W. Mechanism of Emulsion Polymerization. *J. Pol. Sci.: Pol. Sym. Ban.* **1985**, *72*, 161-198.
- (53) Chujo, R.; Ubara, H.; Nishioka, A. Determination of Monomer Reactivity Ratios in Copolymerization from a Single Sample and Its Application to the Acrylonitrile-Methyl Methacrylate System. *Polymer Journal* **1972**, *3*, 670-674.
- (54) Berger, M.; Kuntz, I. The Distinction between Terminal and Penultimate Copolymerization Models. *J. Pol. Sci.* **1964**, *2*, 1687-1698.
- (55) Aguilar, J.; Rabelero, M.; Nuno-Donlucas S. M.; Mendizabal, E.; Martinez-Richa, A.; Lopex, R. G.; Arellano, M.; Puig, J.E. Narrow Size-Distribution Poly(methyl methacrylate) Nanoparticles Made by Semicontinuous Heterophase Polymerization. *J. App. Pol. Sci.* **2011**, *119*, 1827-1834.
- (56) Arnal, M. L.; Balsamo, V.; Ronca, G.; Sanchez, A.; Muller, A. J.; Canizales, E.; Urbina de Navarro, C. Applications of Successive Self-Nucleation and Annealing (SSA) to Polymer Characterization. *J. Therm. Anal. Cal.* **2000**, *59*, 451-470.
- (57) Nealy, D. L.; Davis, T. G.; Kibler, C. J. Thermal History and Melting Behavior of Poly(ethylene terephthalate). *J. Poly. Sci.* **1970**, *8*, 2141-2151.
- (58) ASTM D3359-97 Standard Test Methods for Measuring Adhesion by Tape Test
- (59) A Practical Guide to ReLU <https://medium.com/tinymind/a-practical-guide-to-relu-b83ca804f1f7> (accessed May 7, 2019)
- (60) What is Design of Experiments (DOE)? <https://asq.org/quality-resources/design-of-experiments> (accessed May 6, 2019)
- (61) Mohamed, A. I. A.; Sultan, A. S.; Hussein, I. A.; Al-Muntasheri, G. A. Influence of Surfactant Structure on the Stability of Water-in-Oil Emulsions under High-Temperature High Salinity Conditions. *J. Chemistry* **2017**, *2017*, 1-11.
- (62) Ramakrishna, S. N.; Clasohm, L. Y.; Rao, A.; Spencer, N. D. Controlling Adhesion Force by Means of Nanoscale Surface Roughness. *Langmuir*, **2011**, *27*, 9972-9978.

- (63) Arena, G.; Musumeci, S.; Purrello, R. Calcium- and Magnesium-EDTA Complexes, Stability Constants and Their Dependence on Temperature and Ionic Strength. *Thermo. Acta.* **1983**, *61*, 129-138.
- (64) Gardiner, J. Complexation of Trace Metals by Ethylenediaminetetraacetic Acid (EDTA) in Natural Waters. *Water Research* **1976**, *10*, 507-514.
- (65) Westin, K. J.; Rasmuson, A. C. Nucleation of Calcium Carbonate in Presence of Citric Acid, DTPA, EDTA, and Pyromellitic Acid. *J. Col. Inter. Sci.* **2005**, *282*, 370-379.
- (66) Tunay, O.; Kabdasli, I.; Tasli, R. Pretreatment of Complexed Metal Wastewaters. *Water Sci. Tech.* **1994**, *29*, 265-274.
- (67) Hering, J. G.; Morel, F. M. M. Humic Acid Complexation of Calcium and Copper. *Envir. Sci. Tech.* **1988**, *22*, 1234-1237.
- (68) Christensen, T.; Gooden, D. M.; Kung, J. E.; Toone, E. J. Additivity and the Physical Basis of Multivalency Effects: A Thermodynamic Investigation of the Calcium EDTA Interaction. *J. Am. Chem. Soc.* **2003**, *125*, 7357-7366.
- (69) Shideler, J. J. Calcium Chloride in Concrete, *ACI Journal*, **1952**, 537-560.
- (70) Al-Assi, M.; Kassem, E. Evaluation of Adhesion and Hysteresis Friction of Rubber-Pavement System. *Appl. Sci.* **2017**, *7*, 1029-1042.
- (71) Bird, T. J.; Bates, A. E.; Lefcheck, J. S.; Hill, N. A.; Thomson, R. J.; Edgar, G. J.; Stuart-Smith, R. D.; Wotherspoon, S.; Krkosek, M.; Stuart-Smith, J. F.; Pecl, G. T.; Barrett, N.; Frusher, S. Statistical Solutions for Error and Bias in Global Citizen Science Datasets. *Biol. Cons.* **2014**, *173*, 144-154.
- (72) Mehravar, S.; Ballard, N.; Tomovska, R.; Asua, J. M. The Influence of Macromolecular Structure and Composition on Mechanical Properties of Films Cast from Solvent-Free Polyurethane/Acrylic Hybrid Dispersions. *Macro. Mat. Eng.* **2019**, *1900155*, 1-10.
- (73) Kukanja, D.; Golob, J.; Zupancic-Valant, A.; Krajnc, M. The Structure and Properties of Acrylic-Polyurethane Hybrid Emulsions and Comparison with Physical Blends. *J. Appl. Poly. Sci.* **2000**, *78*, 67-80.
- (74) Douglas, S. J.; Illum, L.; Davis, S. S. Particle Size and Size Distribution of Poly(butyl 2-cyanoacrylate) Nanoparticles. II. Influence of Stabilizers. *J. Col. Inter. Sci.* **1985**, *103*, 154-163.

- (75) Qian, C.; McClements, D. J. Formation of Nanoemulsions Stabilized by Model Food-Grade Emulsifiers Using High-Pressure Homogenization: Factors Affecting Particle Size. *Food Hyd.* **2011**, *25*, 1000-1008.
- (76) Cross-Validation in Machine Learning <https://towardsdatascience.com/cross-validation-in-machine-learning-72924a69872f> (accessed May 30, 2019)
- (77) Anderson, W.; Kozak, D.; Coleman, V. A.; Jamting, A. K.; Trau, M. A Comparative Study of Submicron Particle Sizing Platforms: Accuracy, Precision and Resolution Analysis of Polydisperse Particle Size Distributions. *J. Col. Inter. Sci.* **2013**, *405*, 322-330.
- (78) McComiskey, K. P. M.; Tajber, L. Comparison of Particle Size Methodology and Assessment of Nanoparticle Tracking Analysis (NTA) as a Tool for Live Monitoring of Crystallization Pathways. *Eur. J. Pharm. Biopharm.* **2018**, *130*, 314-326.
- (79) Hou, J.; Ci, H.; Wang, P.; Wang C.; Lv, B.; Miao, L.; You, G. Nanoparticle Tracking Analysis Versus Dynamic Light Scattering: Case Study on the Effect of Ca²⁺ and Alginate on the Aggregation of Cerium Oxide Nanoparticles. *J. Haz. Mat.* **2018**, *360*, 319-328.