

Optimal Formulation of Complex Chemical Systems with a Genetic Algorithm

Morgan Theis¹³, Gianluca Gazzola¹³,
Michele Forlin²³, Irene Poli²³,
Martin M. Hanczyc¹³, Mark A. Bedau^{134*}

¹ProtoLife SRL

c/o Parco Vega (Porta dell' Innovazione), Via della Libertà 12, Marghera
Venezia 30175, Italia

²University Ca' Foscari, Venezia, Italia

³European Center for Living Technology

Ca' della Zorza, Dorsoduro 3859, Venezia 30125, Italia

⁴Reed College, 3203 SE Woodstock Blvd., Portland OR 97202, USA

*corresponding author: mark@protolife.net

Abstract

We demonstrate a method for optimizing desired functionality in real complex chemical systems, using a genetic algorithm. The chemical systems studied here are mixtures of amphiphiles, which spontaneously exhibit a complex variety of self-assembled molecular aggregations, and the property optimized is turbidity. We also experimentally resolve the fitness landscape in some hyper-planes through the space of possible amphiphile formulations, in order to assess the practicality of our optimization method. Our method shows clear and significant progress after testing only 1 % of the possible amphiphile formulations.

Introduction

It is often very difficult to construct a complex chemical formulation that achieves some desired functionality. A number of factors contribute to this difficulty, including unanticipated complications like chemical side reactions, and miscellaneous practical considerations such as the sheer number of independent possibilities that one can actually test in the lab. We here demonstrate a genetic algorithm or “GA” (Holland 1975, Goldberg 1989, Forrest 1993) for optimizing, or at least improving, the functionality of an arbitrary chemical system. Our GA shows a clear and significant improvement in target functionality, which was obtained after explicitly testing only 1% of the possible formulations. This work is a proof of principle for the ability to formulate or “program” complex chemical systems with functionality that is useful in science or industry.

The functionality optimized in our demonstration is the turbidity of amphiphile formulations. Turbidity is a convenient surrogate for other, more desirable functionality, such as internal vesicle volume, vesicle longevity in specific environments, membrane permeability to selected molecules, or longevity of specific chemical reactions encapsulated in vesicles.

A population of amphiphiles that assembles into vesicles, micelles, oil droplets and other structures is a tangible chemical system in the laboratory. A genetic algorithm is another, very different kind of complex system which is abstract and typically computer-controlled. We couple these two different kinds of complex

systems, marrying an abstract and autonomous machine learning method with a concrete chemical screening method to create complex chemical systems.

We are interested in optimizing the properties of amphiphilic systems that make vesicles for several reasons. First, these types of vesicles are good chemical models for cell membranes (e.g. Menger and Angelova 1998), which are fundamental components of all natural living systems. Also, vesicle morphology and functionality is variable and sensitive to parameters easily varied in the laboratory during their formation, such as chemical composition, pH, chemical environment, local temperature, agitation intensity and frequency, and time allowed for formation. Furthermore, vesicles have commercial applications in many areas, including drug delivery, medical imaging contrast agents, cosmetics, and food processing. Figure 1 shows two vesicle populations with different chemical composition (all other variables held constant), illustrating the variable morphology of the vesicles created in this work.

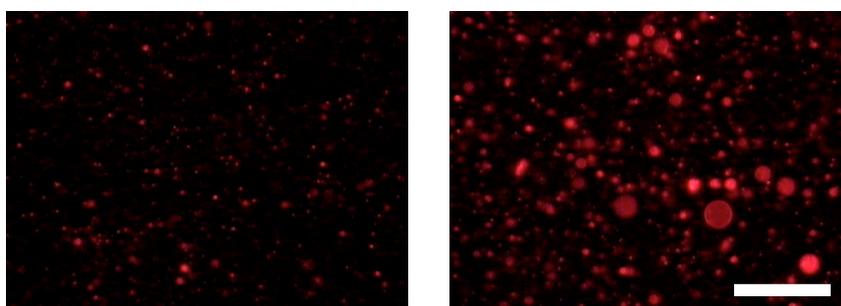


Figure 1. Two fluorescent micrographs of different vesicle populations produced in the experiment demonstrating the diversity of vesicle structures. Bar: 100 microns.

Turbidity was chosen as a target functionality to optimize because it is a straightforward optical property of a system and because high-throughput assays for turbidity can be easily achieved with a spectrophotometer. The turbidity of a vesicle solution also has some correlation to the size of vesicles contained therein (Pozharski et al. 2001), and giant vesicles are of special value in science and industry (Luisi and Walde 2000).

Here we optimize chemical functionality with a genetic algorithm, because genetic algorithms are familiar and straightforward. But the optimization process in our method could be achieved equally well with many other machine learning processes, such as neural network models that predict optimal formulations from previous experimental results (Bull 1999).

We demonstrate a genetic algorithm optimizing the chemical properties of amphiphilic systems, but this method has much broader application. Genetic algorithms have been used to optimize other kinds of chemical functionalities, such as the ligand binding affinity of compounds synthesized from isocyanides, aldehydes, amines, and carboxylic acids (Weber et al. 1995), protease selectivity of hexapeptides (Singh et al. 1996), and mildness of shower gels to human skin (Patel 2004). This shows the broad scope of the methods demonstrated here. Our work differs from these earlier achievements in two primary respects: in our employment of high-throughput screening methods, and in our work with amphiphilic systems that exhibit complex self-assembling structures like vesicles.

Our method for optimizing chemical functionality should be contrasted with four similar but different methods: (i) optimizing a model or simulation of a chemical system, (ii) instantiating a model optimized by a GA, (iii) chemical screening, and

(iv) *in vitro* evolution. In earlier work we optimized the parameters of a dissipative particle dynamics simulation of self-assembling amphiphiles (Bedau et al. 2005). Although the optimization methods used in that work and here are essentially the same, the targets differ because only here do we have the goal to create real chemical formulations. This is the critical difference between our present method and optimizing merely a model or simulation of a chemical system.

Genetic algorithms have been used to optimize various kinds of actual physical systems, ranging from jet engines to autonomous adaptive agents (Forrest 1993; Lipson and Pollack 2000). The typical procedure is for the GA to optimize the parameters of some model or simulation of the physical system. Only those systems described by the optimal models are ever actually constructed. This procedure is not very helpful in the present context because a substantial “reality gap” afflicts models of the relevant chemical reaction networks. While there are impressive models of both very small populations of molecules displaying a wide range of realistic physical properties and larger populations of less-realistic molecules (Jiang et al. forthcoming), these models are ineffective for predicting the relevant properties of many chemical systems, including those studied here. In our method, the genetic algorithm is applied to a population of actual chemical systems realized in the laboratory, and this prevents any reality gap from arising. This method also ensures that the optimized chemical systems are reproducible, once they are found.

If one creates a library of chemical compounds that are candidates for some desired functionality (e.g., some pharmacological property), one can then often devise a way to screen the library for that functionality. Repeated screenings can further narrow down exactly which library compounds are optimal. When one works with real chemical systems, no reality gap arises. However, the best solution of this method will always pre-exist in the library. Our genetic algorithm, on the other hand, is not limited by a pre-existing library. Instead, by design, the genetic algorithm explores a large space of possible recipes by selectively sampling only a tiny fraction of that space. Our approach allows us to find the most fit chemical system without creating every possible permutation of similar systems.

In vitro or directed evolution (Ellington and Szostak 1990; Chapman and Szostak 1994; Rohatgi, Bartel, and Szostak 1996; Wright and Joyce 1997; Joyce 2004) selects polymers based on fitness measurement made on real chemical systems, and it uses mutagenic PCR to introduce possible solutions not present in the initial library of possibilities. The candidate solutions produced by *in vitro* evolution are governed by random mutagenesis, and therefore the permutation processes governing their evolution cannot be programmed by a GA. This is much the same for SELEX (systematic evolution of ligands by exponential enrichment) where specific DNA or RNA molecules are selected through incremental enrichment from a very large initial population (Tuerk and Gold 1990; Irvine, Tuerk, and Gold 1991). The limitations of this system are the same as for general chemical screening where the best solution must be already present in the initial population. By their design, both *in vitro* evolution and SELEX can be applied only to chemical functionality that is mitigated by replicable polymer systems. By contrast, our method in principle could be used to program (optimize) arbitrary chemical functionality.

In summary, the novelty of our chemical design method is based on the coupling of two important properties. First, the method finds a real, replicable chemical product. Second, it possesses flexibility both in the physical systems to which it is applicable and the search mechanism it uses to explore those systems.

Materials and Methods

A library of 16 amphiphiles was compiled. Cholesterol (CHOL), 1-Palmitoyl-2-Oleoyl-*sn*-Glycero-3-[Phospho-*rac*-(1-glycerol)] (Sodium Salt) (POPG); 1,2,3-Trioleoylglycerol (Triolein) (TRIO); 1,2-Diphytanoyl-*sn*-Glycero-3-Phosphocholine (MEPC); 1-Oleoyl-2-Hydroxy-*sn*-Glycero-3-Phosphocholine (LPC); 1,2-Dioleoyl-*sn*-Glycero-3-Phosphate (Monosodium Salt) (PA); Phytosphingosine (*Saccharomyces cerevisiae*) 4-Hydroxysphinganine (YPHY); 1,2-Dioleoyl-*sn*-Glycero-3-Phosphopropanol (Sodium Salt) (PPRO); 1,2-Di-O-Octadecenyl-*sn*-Glycero-3-Phosphocholine (DIPC); L- α -Phosphatidylcholine (Soy) (PC); 16:0 PG or 1,2-Dipalmitoyl-*sn*-Glycero-3-[Phospho-*rac*-(1-glycerol)] (Sodium Salt) (DPPG); 1,2-Dipalmitoyl-*sn*-Glycero-3-Phosphocholine (DPPC); 1,2-Dioleoyl-*sn*-Glycero-3-Phosphocholine (DOPC); (2S,3R,4E)-2-acylaminooctadec-4-ene-3-hydroxy-1-Phosphocholine (Egg, Chicken) (SPM) were purchased from Avanti Polar Lipids (Alabaster, AL, U.S.A.). DL-all-*rac*- α -Tocopherol ($\geq 97\%$) (VITE) and 1,1',2,2'-Tetramyristoyl Cardiolipin (Sodium Salt) (CARD) were purchased from Sigma-Aldrich (Milan, Italy). All amphiphiles were diluted to 2.5mM with chloroform (Sigma) and stored at -20°C in 4mL glass vials.

Stocks of amphiphiles in chloroform were removed from the freezer and set on a heat block (60°C) for 30 minutes, ensuring that all solutions were free of precipitates. Glass sleeves (400uL capacity, Chemtek Analytica) were inserted into an empty 96-position rack and the full rack was placed on a heating block for 96-well plates (VWR, Milan, Italy). Amphiphiles in chloroform were added into the appropriate glass sleeves according to a computer-generated well map (see *Computer Methods*), and each plate contained three replicates of each recipe. For example, Well A5 and A9 contain the same recipe as Well A1.

The rack of glass sleeves containing amphiphile mixtures was heated for 10 minutes before being transferred to a vacuum chamber where it was desiccated for 30 minutes to remove all remaining chloroform. After drying, 400ul of 50mM sucrose at room temperature was added to each sleeve. The rack was covered with a flexible lid that contained individual plugs for each sleeve and then vortex agitated for 10 seconds and placed on the heat block (60°C) for 60 minutes. The plate was briefly removed and vortex agitated again every 20 minutes, including at the end of the heating process.

After removing the lid, 300ul of solution from each well was transferred to a corresponding well in a new clear plastic 96 well plate for analysis. This step was performed at room temperature. The plastic plate was then placed on the heat block (60°C) for 10 minutes before being transferred to the spectrophotometer (Perkin Elmer, Wallac Victor³ Multilabel Counter) maintained at 25°C, for turbidity analysis.

Analysis Methods

Turbidity readings were taken using a wavelength of 405nm with a 0.1 second reading time. The measurement was taken at a standard height of 8mm above the bottom of the wells. The turbidity readings were used to calculate fitness measurements and perform the genetic operations necessary to produce the next generation of recipes.

Four samples with diverse turbidity values were chosen from each plate for analysis by microscopy. A 10ul sample from a well was placed onto a standard

microscope slide, 0.6ul of 100uM Rhodamine 6G fluorescent dye (Sigma) was added and briefly mixed with a pipette tip. Samples were visualized using fluorescence microscopy (Specialty Microscopes with a Panasonic GP-KR222 camera and BTV Pro software). The pictures were taken with a 25x objective lens.

Computer Methods

A genetic algorithm (GA) implemented on a personal computer generated all amphiphile recipes used in this study. A recipe g consists of a number of “genes” $g = (g_1, g_2, \dots, g_{16})$, one for each of the 16 amphiphiles in our library, where g_i is the number of volume units of the i^{th} amphiphile. A single volume unit is 20ul, and the total volume in a given recipe is always 100ul, so each recipe g contains between one and five $g_i > 0$. A generation of the GA was a set of 30 recipes that are produced and tested together in the laboratory in multiwell plates. A single well contains a single recipe. No recipe can occur more than once in a given generation, and no recipe can occur in more than one generation. Each plate contained three replicates of each recipe, in addition to some controls. The turbidity readings of already completed generations were used to generate the recipes for subsequent generations.

Given the turbidity measurements of the replicates of a recipe, the recipe’s fitness was calculated as simply the mean turbidity minus its standard deviation. By subtracting the standard deviation of measured turbidity of the three replicates of each recipe from the mean of those measurements, recipes with wildly varying turbidity measurements have proportionally lower fitness. This fitness criterion puts a premium on the reproducibility of high turbidity.

The initial generation of recipes was randomly generated, with the constraint that cholesterol cannot exceed 40% of a recipe. After the fitness of each recipe in a given generation has been measured and recorded, the next generation of recipes (new plate of experiments) was created by the GA as follows:

- (1) Twelve parents are samples without replacement from the pool of all previous recipes (experiments) by fitness proportional squared selection, according to the following probability:

$$p_j = \frac{F_j^2}{\sum_{i=1}^n F_i^2}$$

where j is any recipe, p_j is the probability of choosing recipe j , and F_j is the fitness of recipe j , as described above.

- (2) Each parent with probability one creates one child by random mutation. One volume unit (20ul) is removed from an ingredient bin that had a positive value in the parent recipe and one volume unit of another (randomly chosen from the remaining 15) ingredient is substituted to create the child recipe.
- (3) Parents were paired in the order chosen (1 with 2, 3 with 4, etc), and these pairs with probability one each produced two “crossed” children, according to the following algorithm:
 - (a) Randomly choose a number, x , from 1-4 and assign it to the first parent. This will be the number of volume units (identified as “chips”, and indexed with a given amphiphile) taken from the first parent. The remaining y chips (where $y = 5 - x$) are chosen from the second parent.
 - (b) Randomly choose x chips from the first parent and y chips from the second parent. That chip combination becomes the first crossed child.

(c) The remaining chips from the two parents become the second crossed child.

(4) Six randomly generated recipes (created as described for the initial population) were added to each generation, in order to visit unexplored regions of recipe space.

Therefore all generations contained 30 new children (12 mutants, 12 crosses, and 6 random recipes). The only exception is generation 4, which contains twice as many children, as a test of the laboratory feasibility of doubling population size.

Our GA differs from typical GAs in one important respect: at every generation all previously tested recipes are potentially available as parents. Every potential new mutant and crossed recipe was screened against previously tested recipes, with new recipes generated to replace any duplicates. Thus, the fitness distribution across the population in a given generation of the GA is measured across the entire set of recipes that have been evaluated so far. In other words, the population size grows each generation by the number of new recipes tested (30).

Once an entire new generation was created, a well map describing each recipe in each well of a 96-well plate was produced, for use when creating the new generation of experiments in the laboratory.

Results

We used a genetic algorithm to optimize turbidity of amphiphile combinations. Each amphiphile formulation included up to 5 volume units drawn from a library of 16 possible amphiphiles (see Materials and Methods), for a total of $16 \cdot 1 + 5 \text{ choose } 5 = (16-1+5)!/5!(16-1)! = 15,504$ possible recipes that constitute a simplex. We ran our genetic algorithm for a total of five generations. Each generation included three replicates of 30 recipes, except that in the fourth generation we doubled the number of recipes, as a test of the practicality of processing larger generations. So our GA tested a total of 180 recipes, or 1.16% of the possible recipes.

Figure 2 shows a bar plot of all recipes made in Generation 1 (y-axis) and their corresponding turbidity (y-axis). The recipes were sorted according to level of fitness. As the figure shows, the turbidity levels of the initial population of recipes generated at random are variable with a few recipes at high levels.

The error bars in Figure 2 indicate a standard deviation of three replicates above and below the mean. So the error bars provide a visual representation of the variability among recipe replicates. Since fitness is calculated as average turbidity minus the standard deviation, single recipes that produced variable levels of turbidity were penalized. It is clear that while the noise of a recipe was sometimes enough to significantly alter its fitness (away from the average turbidity), these cases were rare, and the high turbidity of the most fit recipes is apparent. This assured us that the noise in our system was acceptable and that our fitness measurement was not over-penalizing high turbidity recipes due to variability.

A single recipe can produce different levels of turbidity due to random fluctuations in the conditions during the self-assembly process. However, the most variable readings observed were due to the presence of large precipitates that formed in the preparation. These precipitates varied not only in abundance but also in their position on the bottom of the 96-well plate, which lead to highly variable turbidity values. Recipes that produced such inconsistencies were penalized by the fitness function in our GA.

Figure 3 shows the fitness distribution of all recipes over five generations, ordered first by generation and then by fitness within a generation. The maximum fitness in each generation tended to increase over time, rising from 0.41 in the initial generation to 0.60 in the next two generations, and then rising again to 0.72 in the fourth generation. The average fitness of each generation also tended to increase over time, with mean fitness per generation going from 0.12 in generation 1 to 0.22 in generation 2, falling a bit to 0.19 in generation 3, and then in the final two generations rising again to 0.21 and 0.23. As expected, recipes created by mutation (blue, Figures 3 and 4) and cross-over (green) from high fitness recipes were consistently more fit than those that were randomly generated (purple).

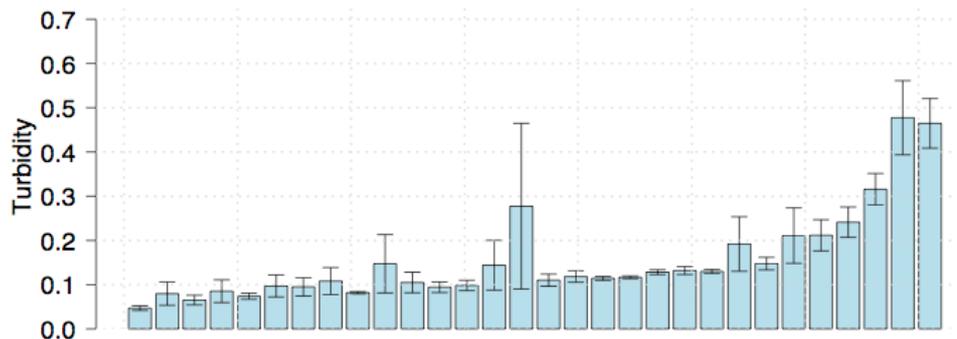


Figure 2. Turbidity of all of the recipes in generation 1, sorted by fitness (turbidity – standard deviation). The height of each bar is the mean of three replicates of a recipe. One standard deviation is shown above and one below each bar. A recipe’s fitness is the height of the lower error bar. Note that some recipes have substantial error bars, but in most cases the standard deviation is less than 20% percent of the mean turbidity value.

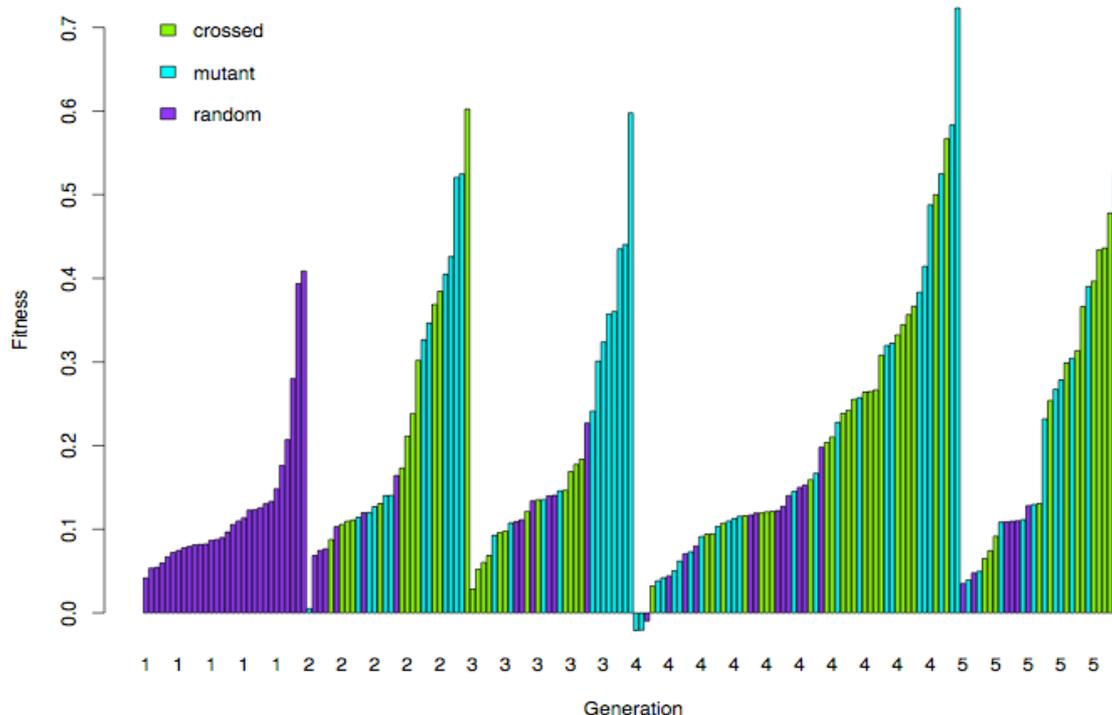


Figure 3. Fitness values of every recipe tested. All recipes were grouped by generation and then ordered by fitness. Randomly

generated recipes are shown in purple, mutant recipes are shown in blue, and cross-over recipes are shown in green. Generation 4 contains twice as many recipes in order to test the practicality of processing larger populations. Some recipes have a negative fitness value because the standard deviation of their replicates exceeded their mean. Note that, because any previously tested recipe can potentially parent recipes in any given generation, the current population fitness is not reflected merely by the fitness of the most recent generation of recipes tested.

Examination of the fitness distribution of the recipes created by mutation and crossing gives a clear indication of the success of the GA. By comparing the distribution of fitness values across all the recipes in our search space, which is approximated by the fitness distribution of a random sample of recipes (Figure 4, purple distribution in front) with the fitness distribution of the recipes produced by mutation and crossing (Figure 4, turquoise and green distribution in back), it is readily seen that the GA spent its time preferentially testing relatively high fitness recipes. Fully 43% of the mutant and crossed recipes from the GA have a fitness value that is within the top 10% of the recipes chosen at random. Therefore, the GA is very successful at identifying untested recipes with relatively high fitness.

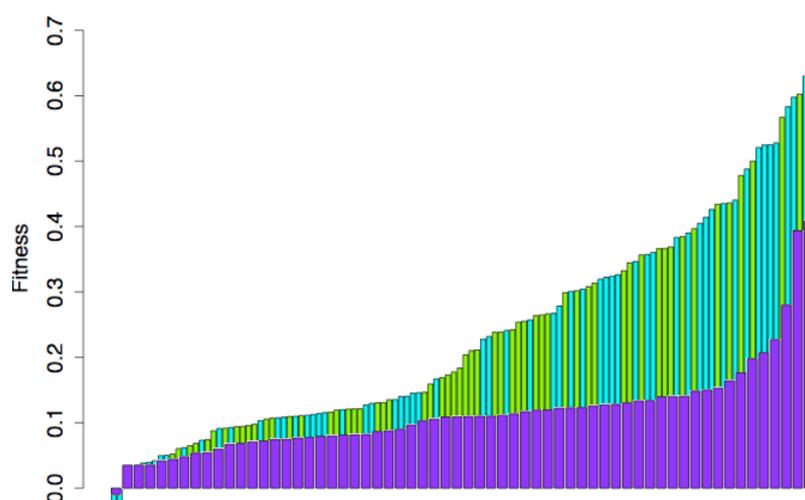


Figure 4. Distribution of fitness values in two subsets of recipes tested by the GA in this study (see Figure 3). The distribution shown in front (purple) is a random sample of recipe space; as such, it approximates the distribution of fitness values in the entire recipe space. The distribution shown behind the random sample contains those recipes that the GA created by mutation (turquoise) or crossing (green). Comparison of the two distributions shows that the GA preferentially searched the high-fitness regions of recipe space.

The highest fitness recipe found was composed of a 1:1:1:1:1 ratio of CARD, TRIO, PA, LPG, and DPPG (Figure 6, discussed below). To understand the chemical basis for its success and the success of other highly fit recipes, we first analyzed the distribution of individual ingredients in the search space. For each amphiphile, a sorted fitness graph was made containing all recipes created over the course of the five generations containing that amphiphile. Figure 5 shows two of these graphs, for

the ingredients TRIO and VITE. These graphs were chosen to represent the variability of the effects of ingredients on the fitness of recipes. Every recipe that contained VITE had low fitness (<0.3), indicating that it could have a universally detrimental effect on the fitness of recipes. Correspondingly, this graph contains very few bars since the GA quickly learned this pattern and steered recipes away from those containing VITE. The graph of TRIO recipes, however, shows a very broad distribution. TRIO was present in the most fit recipe, but it was also present in many medium and low fitness recipes. In fact, roughly 30% of recipes containing TRIO had fitness <0.2 . This shows that TRIO does not have a universally positive effect on turbidity, in the presence of other ingredients. The same pattern of recipe distributions with variable fitness was observed for almost all ingredients.

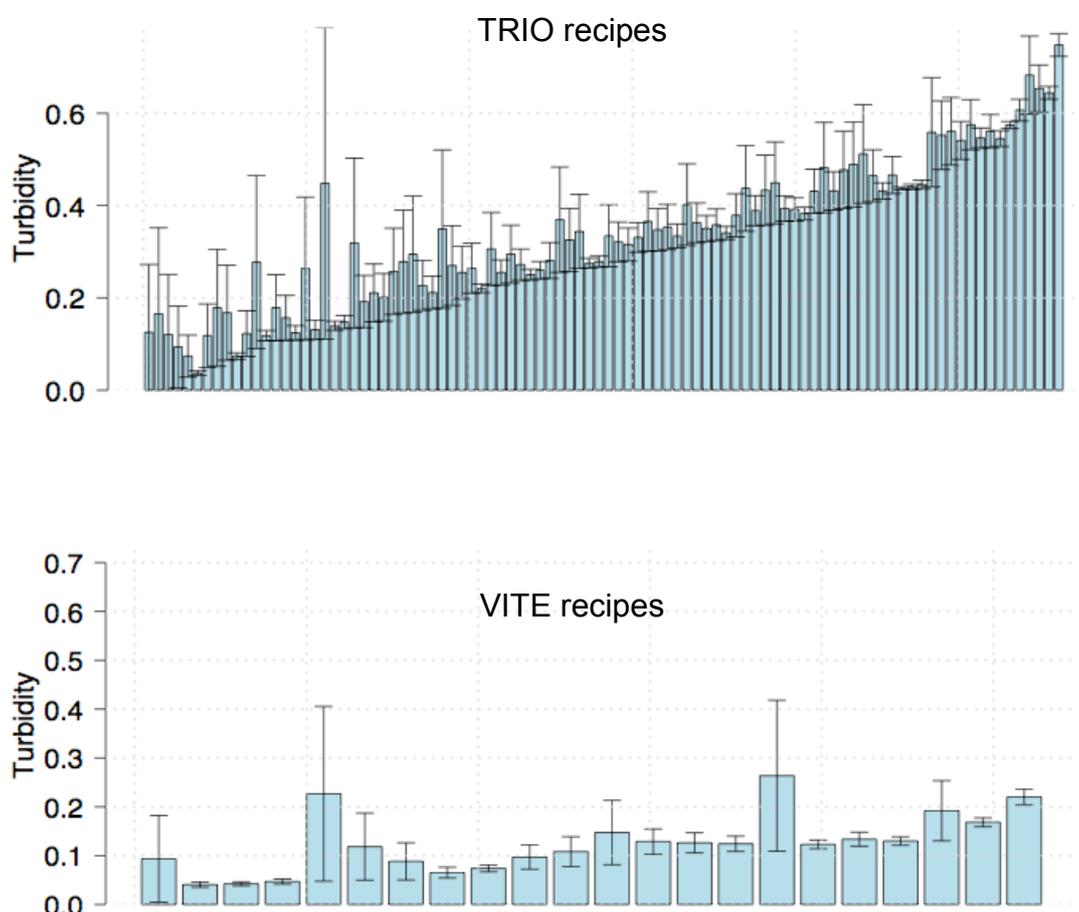


Figure 5. All recipes tested by the GA containing a given ingredient, sorted by fitness. Above: all created recipes that contained the ingredient TRIO, and their corresponding fitness. Below: an analogous distribution for VITE. Note that the fitness of the highest fitness VITE recipes is much lower than that for TRIO, which caused the GA to create many more TRIO recipes (recipes with TRIO were much more likely to be chosen as parents). Note also that even though TRIO appears in many very high turbidity recipes, it also appears in a large number of medium and low fitness recipes. This shows how the contribution of TRIO to fitness depends heavily on the other amphiphiles with which it interacts.

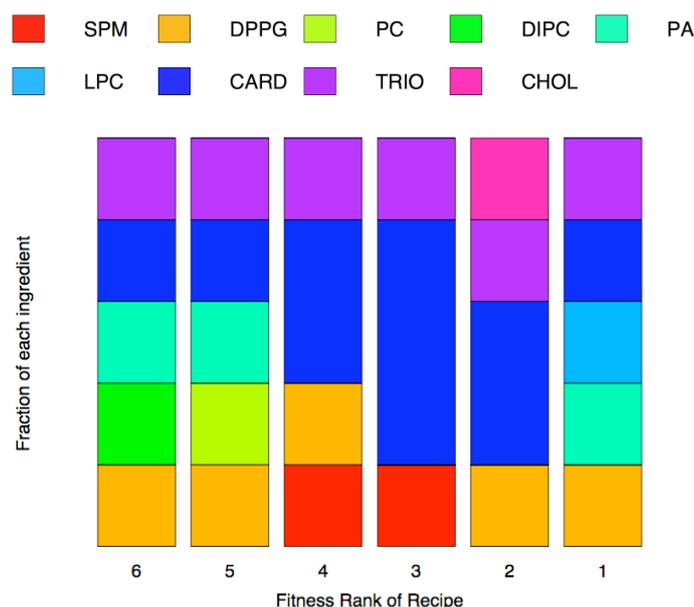


Figure 6. The ingredients in the six most fit recipes, in ascending order left to right. These constitute the top 3% of all recipes. The formulation of a recipe is represented as a bar indicating the relative proportion of each ingredient. Notice that CARD and TRIO appear in each of the recipes. Notice also that half of the recipes contain the maximum number (5) of different ingredients, and none contains fewer than three.

There are some discernable patterns that help explain observed fitness values. All of the six most fit recipes contained CARD and TRIO and none of them contained VITE (Figure 6). Lipids such as these with three or more chains may contribute more significantly to the turbidity value by adding more surface area to self-assembled structures making them larger. For CARD, perhaps the membrane that contains this amphiphile are more stable and thus represented in a higher proportion of existing vesicles as the time of analysis. Indeed, CARD has been found to interact with other amphiphiles within bilayers to affect both apparent area compressibility modulus and lysis tension of the membrane perhaps explaining why membranes that contain CARD are more stable (Nichols-Smith et. Al. 2004). On the other hand, it has been proposed that TRIO can act as a spacer between polar headgroups of other amphiphiles resulting in an increase in the effective surface area per lipid molecule (Prades et. Al 2003). A specific interaction between CARD and TRIO has not been described.

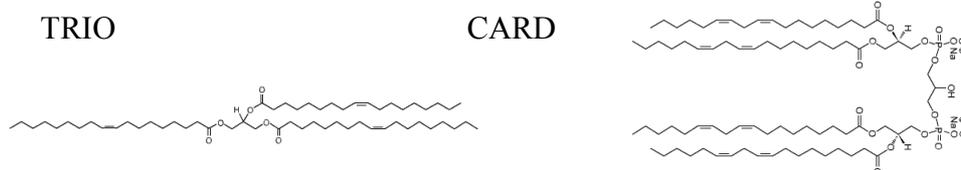


Figure 6 also shows that most of the most fit recipes contain close to the maximum number of different ingredients possible suggesting that contributions to high fitness values typically depend on interactions among different amphiphiles.

The dominance of CARD and TRIO was confirmed and further elucidated by a statistical analysis of the data from all generations, using neural net modeling and also using a third order polynomial fit (Forlin et al. 2006).

To better understand the topology of the function that assigns fitness to recipes, we measured turbidity in a variety of “fitness slice” experiments. A fitness slice shows the fitness of recipes in a hyperplane through the space of possible recipes, revealing a small piece of the fitness landscape. The fitness slices shown here are triangular planes through our 16 dimensional simplex recipe space. Figure 7a shows the turbidity of recipes formed by systematically varying the relative proportions of SPM, PC, and TRIO. For example, the recipe in the corner of Figure 7a labeled “SPM max” consists of 100% SPM; analogously for the PC-max and TRIO-max corners. The recipe half way between PC-max and SPM-max consists of 50% each of PC and SPM; analogously for the TRIO-PC and TRIO-SPM edges of this fitness slice. In similar fashion, the recipes in Figure 7b consist of various proportions of SPM, DOPC, and CHOL, but in this case each recipe also contained 10% CARD. The recipe in the CHOL-max corner consists of 10% CARD and 90% CHOL; the same for the SPM- and DOPC-max corners. The recipe in the middle of the floor consists of 10% CARD, 30% SPM, 30% DOPC, and 30% CHOL. We did not test replicates of recipes in fitness slices, because the neighboring recipes in a fitness slice are nearly replicates.

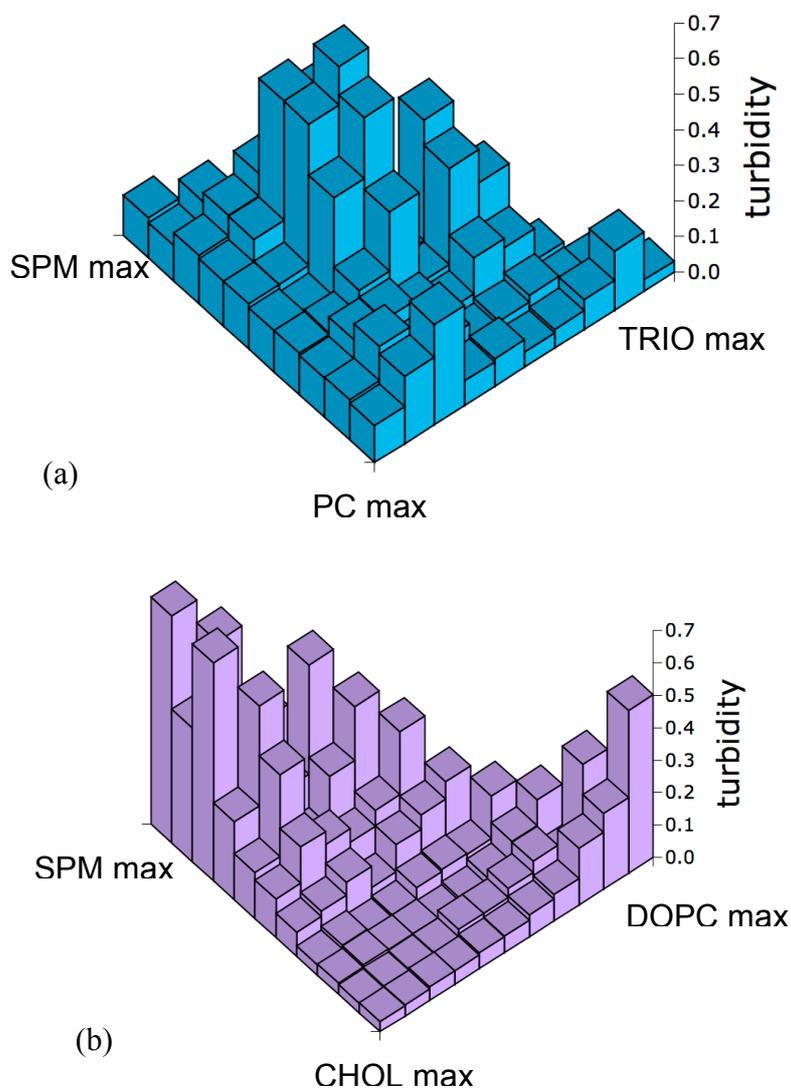


Figure 7. Turbidity measured in different slices of the recipe space. All recipes consist of recipes with various proportions of different amphiphiles, stepping in 10% increments. Notice that (a) indicates a fitness mound in

recipes containing roughly equal proportions of SPM, PC, and TRIO. The saddle visible in (b) indicates that recipes with a preponderance of SPM or (to a less extent) DOPC are fairly turbid, while recipes consisting mostly of CHOL are not. Note that in the left corner of the fitness slice (b), the recipe 90% SPM + 10% CARD shows a high fitness, while the similar recipe in the left corner of (a) with 100% SPM has much lower fitness. This illustrates how sensitive turbidity is to the exact composition of the amphiphile recipes, and might be evidence of a phase transition.

The fitness landscape revealed in Figure 7 shows clear topographical structure: (a) a mound and (b) a saddle. This is the kind of structure that a genetic algorithm can exploit. It also shows significant surface variation in adjacent sites, presumably mainly from experimental noise. Reduction of experimental noise could presumably be improved as much as desired by increasing the number of replicates of each recipe, thus increasing the testing in the lab. The coarse graining of the fitness landscape's horizontal resolution is due to finite experimental resolution. Experimental resolution is determined by the size of the minimal volume units used to construct recipes. Resolution can be improved as much as desired by testing recipes at intermediate points in the fitness slice, thus increasing the testing in the lab. Note that the experimental resolution (10ul volume units) of the recipes in the fitness slices is twice that of experimental resolution (20ul) of the recipes tested by the GA.

Discussion

We used microscopy to determine the types of aggregate structures that were being produced in a number of wells in each generation. We could identify many vesicles, both unilamellar and multilamellar, as well as oily droplets. Aggregates such as micelles would be too small to detect by optical microscopy. Some structures appeared to be a combination of oil droplet and vesicle. These structures, which ranged in size from sub-micron to tens of microns in diameter, would contribute significantly to the turbidity readings. Vesicle size correlates with turbidity, but this correlation drops off when the diameter of the vesicles in the population exceeds about 500nm (Pozharski et al. 2001). Many of our vesicles had diameters that were at least an order of magnitude larger than this, which could explain why turbidity did not completely correspond to vesicle size in our experiment. However there is an evident difference in the vesicle size between the low and high turbidity recipes.

Figure 4 demonstrates that the improvement in fitness found by the GA is not merely a result of increased sampling. The most fit recipe, with fitness of 0.72, had a fitness almost twice as high as the most fit recipes in Generation 1 (with turbidity of 0.41). By itself, this does not prove that the GA finds more fit recipes than would be found from random sampling, for random sampling will inevitably eventually find all high-fitness recipes. The real proof that our GA is intelligently searching the space of possible recipes requires comes from comparing the fitness distribution of recipes created by mutation and crossover with the fitness distribution of recipes chosen at random (Figure 4).

The fitness slices shown in Figure 7 were designed to include some amphiphile combinations thought to have high fitness. Note that none of the recipes tested by the fitness slices had fitness higher than the most fit recipe found by the GA (Figures 3 and 4). This provides evidence that the GA did not overlook any high-fitness recipes in the sub-regions of recipe space that it explored.

Not only can our procedure select for and produce a winning recipe of amphiphiles from a much larger initial pool; it can also point out combinatorial effects and interactions among amphiphiles. Once specific recipes of amphiphiles are selected by the GA and identified, further detailed laboratory analysis can be performed to understand how distinct amphiphilic molecules interact within self-assembled structures such as vesicles.

A clear implication of the fitness distributions of all the recipes containing a single ingredient (Figure 5) and of the ingredients in the winning recipes (Figure 6) is that a recipe's fitness does not depend in any simple way on single ingredients. Instead, the chemical systems the GA created are complex, multidimensional systems with significant interactions among component amphiphiles. Another indication of this can be seen by comparing the turbidity of the recipe at the front corner of figure 7a with that of the recipe at the left corner of figure 7b. The first recipe is 100% SPM, and the second is 90% SPM and 10% TRIO. Note that this minimal change in recipe produces a very large change in turbidity. Although Figure 6 shows that the highest-fitness recipes found by the GA tend to have a large number of ingredients, the fitness slices show that recipes with only 2 (Figure 7b) or 3 ingredients (7a) can have relatively high fitness, so high fitness does not necessitate the interaction of 4 or 5 ingredients. This underscores the utility of optimizing chemical functionality of such systems with genetic algorithm.

One challenge faced by our method is experimental noise and finite experimental resolution. If fitness measurements among replicates are too inconsistent, a GA will have difficulty optimizing system properties because fitness measurements will be too inaccurate. And if experimental resolution is too low, relevant topography in the fitness landscape will be hidden from the GA. The topography revealed by the fitness slices (Figure 7) would presumably be much smoother if experimental resolution were improved and experimental noise were reduced. In any event, even though fitness slices reveal that our GA must cope with both experimental noise and finite resolution, the success of optimization process shows that these challenges can be managed.

It is worth noting that the fitness landscape over our recipe space is not necessarily smooth everywhere. Some regions might exhibit sharp discontinuities, as a result of phase transitions in the aggregations created by slightly different amphiphile formulations. Such a sharp transition might explain the difference between the recipe at the front corner of figure 7a with that of the recipe at the left corner of figure 7b.

Our work reported here employs a genetic algorithm, but analogous methods could employ other machine learning methods involving neural networks, Bayesian networks, or other predictive statistical models. Also, these methods could be used to optimize a wide variety of chemical functionalities, besides those involving amphiphiles, such as those relevant for *in vitro* evolution.

The power of a GA increases with increasing population size and number of generations, so our method places a premium on being able to quickly create and assess a large number of chemical systems. Available high-throughput screening assays therefore constrain the chemical systems for which this method is useful. Our choice of a 96-well plate format for a population of recipes and automated turbidity measurement made our procedure feasible. Other methods for making vesicles, such as electro-formation (Angelova and Dimitrov 1986, Dimitrov and Angelova 1988), would be prohibitively expensive in terms of materials and laboratory time. This was one motivation for developing our simple and high-throughput vesicle-formation

process involving only simple lipid film rehydration with applied heat that could be executed in a 96-well format. This enables one generation of recipes (one or more plates) to be processed in one day, so that reasonably sized optimization tasks could be performed in a matter of weeks.

Current work includes optimizing in recipe spaces that are orders of magnitude larger, using generations that are much larger. We are in the process of also optimizing the meta-parameters of the genetic algorithm, such as selection function, mutation rate, and crossing methods, as well as developing other optimization methods such as estimating a model of fitness from the recipes tested so far, and using the model to predict which untested recipes might have highest fitness. In addition to the spectrophotometry used here, we are employing other high-throughput assays, involving fluorimetry, fluorescent microscopy, and microscopy image analysis. Current optimization targets include vesicle entrapped volume and membrane permeability.

Conclusion

We have demonstrated a proof of principle of using genetic algorithm to optimize the properties of real chemical systems. Specifically, we have optimized the turbidity of vesicle populations by applying a genetic algorithm to combinatorial recipes of amphiphiles. Over the course of only five generations in which only 1% of all possible recipes were explicitly tested, we see a quantitative increase in the turbidity of amphiphile formulations. Variants of this method can be used to optimize a variety of other kinds of chemical functionalities. The key bottleneck in our method is the time required for laboratory procedures, and a key constraint is available high-throughput screening methods.

Our study provides a gateway to a host of related applications. By combining the power of machine learning methods with high-throughput chemical screening, desired functionalities of complex chemical systems can be automatically designed. This work illustrates a transition in the contemporary study of complex systems to practical methods for creating real chemical systems with desired functionality.

Acknowledgements

Thanks to the ECLT, which helped facilitate this work and provided opportunities to have it presented and discussed. This work was supported by the EU in the PACE integrated project, EC-FP6-IST-FET-IP-002035. And thanks to our colleagues at ProtoLife, for extensive valuable discussion of this project.

References

- Angelova, M.I., Dimitrov, D.S. 1989. Liposome electroformation. *Faraday Discuss. Chem. Soc.* 81, 303-311.
- Bedau, M. A., A. Buchanan, G. Gazzola, M. Hanczyc, T. Maeke, J. S. McCaskill, I. Poli, N. H. Packard. 2005. Evolutionary design of a DDPD model of ligation. *Lecture Notes in Computer Science* 3871, 201-212.
- Bull, L. 1999. On model-based evolutionary computing. *Soft Computing* 3, 183-190.

- Chapman, K.B., and J.W. Szostak. 1994. In vitro selection of catalytic RNAs. *Current Opinions in Structural Biology* 4, 618-22.
- Ellington, A.D., and J.W. Szostak. 1990. In vitro selection of RNA molecules that bind specific ligands. *Nature* 346, 818-822.
- Dimitrov, D.S. and Angelova, M.I. 1988. Lipid swelling and liposome formation mediated by electric fields. *Bioelectrochemistry and bioenergetics* 19, 323-336.
- Forlin, M, I. Poli, D. De March, N. Packard, G. Gazzola, R. Serra, M. Bedau, M. Theis, M. Hanczyc. 2006. Modelling evolutionary experimental designs for complex chemical systems. Submitted.
- Forrest, S. 1993. Genetic algorithms: principles of natural selection applied to computation. *Science* 261, 872-878.
- Goldberg, D. E. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA: Addison-Wesley.
- Holland, J. H. 1975. *Adaptation in Natural and Artificial Systems*. Ann Arbor: University of Michigan Press. Reprinted by MIT Press, Cambridge, MA, 1992.
- Irvine, D., Tuerk, C., and L. Gold. 1991. SELEXION. Systematic evolution of ligands by exponential enrichment with integrated optimization by non-linear analysis. *Journal of Molecular Biology* 222, 739-61.
- Jiang, Y., and B. Travis, C. Knutson, J. Zhang, and P. Weronki. Forthcoming. Numerical methods for protocell simulations. In Rasmussen, S., M. A. Bedau, L. Chen, D. Deamer, D. C. Krakauer, N. H. Packard, and P. F. Stadler, eds., *Protocells: Bridging Nonliving and Living Matter*. Cambridge: MIT Press.
- Joyce, G. 2004. Directed evolution of nucleic acid enzymes. *Annual Review of Biochemistry* 73, 791-836.
- Lipson, H. and J. Pollack. 2000. Automatic design and manufacture of robotic life forms. *Nature* 406, 974-978.
- Luisi, P.L. and Walde, P. (editors) 2000. *Giant Vesicles: Perspectives in Supramolecular Chemistry* 6. John Wiley and Sons, LTD, West Sussex, England.
- Menger, F. M., and M. I. Angelova. 1998. Giant vesicles: imitating the cytological processes of cell membranes. *Accounts of Chemical Research* 31, 789-797.
- Nichols-Smith, S., The, S., Kuhl, T.L. 2004. Thermodynamic and mechanical properties of model mitochondrial membranes. *Biochimica et Biophysica Acta* 1663, 82-88.
- Patel, S. W. 2004. Method for finding solutions. US Patent 6,768,973.
- Pozharski, E. V., L. McWilliams, and R. C. MacDonald. 2001. Relationship between turbidity of lipid vesicle suspensions and particle size. *Analytical Biochemistry* 291, 158-162.
- Prades, J., Funari, S.S., Escriba, P.V., Barcelo, F. 2003. Effects of Unsaturated fatty acids and triacylglycerols on phosphatidylethanolamine membrane structure. *Journal of Lipid Res.* 44(9), 1720-7.
- Rohatgi, R., D.P. Bartel, and J.W. Szostak. 1996. Nonenzymatic, template-directed ligation of oligoribonucleotides is highly regioselective for the formation of 3'-5' phosphodiester bonds. *Journal of the American Chemical Society* 118, 3340-3344.
- Singh, J., M. A. Ator, E. P. Jaeger, M. P. Allen, D. A. Whipple, J. E. Solowej, S. Chowdhary, and A. M. Treasurywala. 1996. Application of genetic algorithms to combinatorial synthesis: a computational approach to lead compound identification and lead optimization. *Journal of the American Chemical Society* 118, 1669-1676.
- Tuerk, C., and L. Gold. 1990. Systematic evolution of ligands by exponential enrichment: RNA ligands to bacteriophage T4 DNA polymerase. *Science* 249, 505-510.
- Weber, L., S. Wallbaum, C. Broger, and K. Gubernator. 1995. *Angewandte Chemie International Edition in English* 34 (20), 2280-2282.
- Wright, M., and G. Joyce. 1997. Continuous in vitro evolution of catalytic function. *Science* 276, 614-617.