Lipidia: An Artificial Chemistry of Self-Replicating Assemblies of Lipid-like Molecules

Barak Naveh[1], Moshe Sipper[1], Doron Lancet[2] and Barak Shenhav[2]
[1]Dept. of Computer Science, Ben-Gurion University, Beer-Sheva 84105, Israel
[2]Dept. of Molecular Genetics and the Crown Human Genome Center,
The Weizmann Institute of Science, Rehovot 76100, Israel
{barnav,sipper}@cs.bgu.ac.il, {doron.lancet,barak.shenhav}@weizmann.ac.il

Abstract

Lipidia is a new simulation system that is related to the "Lipid World" scenario for the origin of life. Lipidia allows for conducting experiments with a population of assemblies containing lipid-like molecules on a two dimensional grid. The dynamics of the assemblies is modelled using the Graded Autocatalysis Replication Domain (GARD) model. New experiments using a finite environment model with GARD were conducted with Lipidia. The experiments show that more self-replicating assembly species appear when using a model of finite environment than when using a model of infinite environment. In many species the number of individuals increases as well.

Introduction

The "RNA World" is possibly today's most popular theory for the origins of life (Gilbert, 1986; Joyce, 2002). Because RNA molecules can act as catalysts in addition to acting as templates, it is hypothesized they might have been able to do both: to store alphabet-based genetic information *and* to catalyze their own creation. Life, according to this theory, began when certain RNA molecules achieved the capability to replicate themselves. This scenario, despite its elegance, suffers from difficulties.

In an attempt to come up with a probable scenario, having observed that no known bio-molecule is capable of self replication in its naked form, it has been suggested that self replication might not have been achieved by a single molecule, but rather by a molecular ensemble (Kauffman, 1995). This work is based on "The Lipid World" scenario (Segre et al., 2001) which follows that line of thought. The scenario assumes that self-replication was initially achieved by non-covalent assemblies of lipid-like molecules that contained mutually catalytic sets (Segre et al., 2000). RNA according to this scenario, while possibly playing an important role, came later.

Why Lipids?

Lipid-like amphiphiles (molecules that have one end that "loves" water, and another end that "hates" water) are assumed to have been present in the primordial soup (Deamer, 1997; Luisi et al., 1999). They are known to be capable of self-organizing into higher-level structures (e.g. micelles and vesicles) (Luisi et al., 1999; Gompper and Schick, 1994; Tanford, 1978). Lipid vesicles have been "shown to be capable of enhancing the rates at which precursors are converted into vesicle-forming amphiphiles (Bachmann et al., 1992). In some settings, this leads to an auto-catalytic expansion of the molecular assemblies, a process resembling cell growth" (Segre and Lancet, 2000). Random fission process can cause occasional divisions. Altogether, we have assemblies of molecules that demonstrate a primitive form of growth and division, in a process that is, although noisy, capable of self-replication with a reasonable fidelity (Segre et al., 2000; Segre and Lancet, 2000). The inside of similes of such assemblies, namely lipid vesicles, is shielded from the surrounding environment and thus hypothesized to be capable of offering "hospitable conditions" under which RNA replication can be more likely. Once some coupling is formed between these two replication systems, an early cell could come into existence (Szostak et al., 2001).

Scope of Current Work

We use the *Graded Autocatalysis Replication Domain* (GARD) model (Shenhav et al., 2004; Segre et al., 1998) to quantitatively model and simulate the developmental process of non-covalent assemblies of lipid-like molecules. Previous studies using the GARD model have mostly examined such assemblies in a one-at-a-time fashion. The behavior of assembly populations has been largely unexplored (Segre et al., 2000). In this work we expand the model to a *population* of assemblies and obtain quantitative and qualitative results regarding its behavior. Also, previous studies assumed idealization of an infinite environment where the assembly's effect on the environment is negligible and "food" molecules are in infinite supply. In this work we introduce a finite environment to the model, which allows cross-interactions between assemblies via the environment. We compare the effect of finite environment vs. infinite environment.

Lipidia

This section introduces Lipidia's terminology and describes its objects and interactions.

Structure

Lipidia is based on a two dimensional interaction *grid*, as with cellular automata. Each square on the grid is called a *grid-location* (or *location* for short). For each location there is a defined *environment* containing a variety of *molecules*. Each location may contain zero or more *assemblies* of molecules. The location's environment is common for all assemblies contained within it. Molecules from the environment may *join* an assembly, and molecules from the assemblies may *leave* their assembly back to the environment. "Matter" on the grid is therefore preserved — no matter is ever lost or created[1].

Each grid location has eight neighboring locations, except the locations on the grid's edge, which border a surrounding *gutter*. The gutter is a special location that takes care of objects falling-off the grid. A few gutter policies can be applied. A common policy is to insert objects falling from one edge to the opposite edge, hence turning the grid into a toroid.

Initial Configuration

The environment is seeded with an arbitrary number of molecules, of N_G different *types*. We usually start with all grid locations empty of assemblies, and their environments uniformly seeded with a constant number e_0 of molecules for each of the N_G types.

Assembly Birth

Assemblies spontaneously come to existence at some constant low rate. We call this appearance *assembly birth*, and it can happen at any grid location. Ideally, birth rate should depend on the numbers and types of molecules available at each location. It is reasonable to assume that the rate of assembly creation will degrade as the environment material runs out. However, for simplicity we simulate a constant birth rate.

Assembly Growth

As the simulation progresses, molecules from the environment may join assemblies, and molecules from the assemblies may leave them and return back to the environment. Join and leave reactions establish the assembly growth; however, growth is also affected indirectly by other assemblies via the shared environment. The dynamics of assembly growth are governed by the GARD model (Segre et al., 2000) as follows.

The system contains a set of N_G types of molecules, and mutual catalysis can occur between any molecule pair. The catalytic rate enhancement exerted by molecule type j on molecule type i is denoted by a matrix element β_{ij}. Values for β matrix are assigned in accordance with previously developed Receptor Affinity Distribution (RAD) model (Lancet et al., 1993). The basal reaction rates k_f and k_b (forward and backward) respectively specify spontaneous join and leave rates.

An assembly s is represented by an N_G-dimensional vector, where each component s_i denotes the number of molecules of the i-th type in the assembly. An environment e is represented similarly.

Join Reaction The join rate J_i of molecule type i in the environment e of an assembly s is given by:

$$J_i = k_f e_i (1 + \sum_{j=1}^{N_G} \beta_{ij} s_j) \quad (1)$$

Therefore, J_i increases the higher the count, e_i, of molecules of type i in the environment. The spontaneous rate, k_f, is enhanced by a catalysis generated by molecules within the assembly: each molecule of type j contributes β_{ij} for that rate enhancement. Hence equation 1 above.

Leave Reaction The leave rate L_i of molecule type i from assembly s to its environment e is given by:

$$L_i = k_b s_i (1 + \sum_{j=1}^{N_G} \beta_{ij} s_j - \beta_{ii}) \quad (2)$$

Therefore, L_i increases the higher the count, s_i, of molecules of type i within the assembly. The spontaneous rate ,k_b, is enhanced by a catalysis generated by molecules within the assembly: each molecule of type j contributes β_{ij} for that rate enhancement, except of one molecule of type i that can not catalyze its own leave. Hence equation 2 above.

Assembly Division (Split)

When the number of molecules in an assembly reaches a certain value, denoted by N_0, the assembly is divided into two daughter assemblies. When this division takes place, every molecule in the original assembly is randomly joined to one of the daughter assemblies. The exact structure of the assembly is not modelled.

Assembly Diffusion

Each assembly diffuses at some low rate from its current location to a neighboring location on the grid. It may as well diffuse into the gutter and handled according to the gutter policy.

Environment Diffusion

As a result of diffusion, environments of two neighboring locations mix some percentage of their molecules at some low rate. Such diffusion may also occur into the gutter, then to be handled according to the gutter policy.

[1] The finite environment model is replaceable with an infinite environment model of fixed concentrations.

Population Control and Assembly Death

An assembly may divide into two daughter assemblies, which may divide further. In time we obtain an exponential explosion of assemblies. Following are means that limit population growth, some of which involve assembly *death*, whereby an assembly disassembles and return its molecules back to the environment.

Natural Death An assembly undergoes spontaneous decomposition following a certain amount of time after its creation. This sort of death is natural to the model.

Finite Environment A finite environment provides a natural means to limit population growth. It puts a bound on the total number of molecules, and therefore on the number of assemblies that can be created.

The Reaper The above means are part of the model and therefore "natural". However, to avoid excessive computation we also need an "artificial" means. The reaper keeps the global assembly population on the grid below some bound. When the number of assemblies exceeds the bound, the reaper selects an assembly at random and "kills" it.

Scheduling of Random Events

All simulation reactions and behaviors mentioned above can be collectively called *events*. Many of these events are stochastic and occur at defined rates. The simulation schedules the events in a stochastic but fair method, which reflects their rates. The method is best visualized as a giant roulette wheel, where each event "owns" one roulette slot. Unlike true roulette, slots may vary in size making them more or less likely to occur. Thus, the size of the slot corresponds to the rate of the owning event.

Upon every cycle, the simulation engine gives the roulette wheel a spin to choose the next event. The event is activated and the state of the system is modified, possibly changing the rates of other events, whereby their corresponding slots become wider or narrower. The algorithm implementing the roulette wheel requires $O(log(n))$ time and $O(n)$ space for n events.

Attractors

This section discusses the important concept of *attractors* in the context of Lipidia; how they are defined and how to find them.

Composition Stability

The *normalized composition* (or *composition* for short) of an assembly s is given by:

$$\tilde{s} = s / \|s\|, \tag{3}$$

where $\|s\|$ is the norm of the vector s.

Due to mutual catalysis, we expect to find compositions that are *stable* over time. We say that a composition is stable if it remains *similar* along splits. This stability involves a quasi-stationary state, and should be distinguished from equilibrium-type stability. Thus, even though during assembly growth molecules can join or leave, when it splits (actually, just before it does) its composition is similar to its parent's composition (at the time of split).

Composition Similarity and Self Replication

We estimate the similarity of two assemblies s_p and s_q by using the scalar product of their compositions:

$$H(s_p, s_q) = \tilde{s_p} \cdot \tilde{s_q} \tag{4}$$

therefore, $H = 1$ denotes perfect similarity and $H = 0$ denotes perfect dissimilarity. Note that H only measures "how far" composition s_p is from s_q, but does not measure "how hard" it is, in terms of reactions, to get from composition s_p to composition s_q. When a parent assembly splits into two daughter assemblies that grow to have a similar composition as the parent, we say that the parent replicated.

Trajectories in Composition Space

A composition we measure at any time point, and specifically at the time of assembly split can be thought of as a point in the composition space, which includes all possible compositions. Each of the assembly's ancestors might have visited another point in the composition space. An assembly's lineage can be thought of as the *trajectory* along these points.

Attractors and Basins of Attraction

Each point in the composition space is theoretically reachable. However, due to mutual catalysis some points are more likely then others. The composition space may be perceived as a landscape, where low points represent compositions that are easier to get (in terms of reactions), while high points represent the opposite. The trajectories are therefore likely to "fall" into lower areas. If an area has a basin shape, each trajectory, having fallen in, will have a hard time escaping the basin: it will tend to keep falling back into the *attractor*, that is, the foot of the basin. Therefore, an assembly having a composition within an attractor will tend to be stable.

Attractors as Assembly Species

When we observe species in nature we see that individuals of the same species may vary from each other. However, they seem to be "trapped" within some "cloud of variations" that represents their species. We can therefore see our attractors as representing assembly types (or species). Assemblies of the same attractor will vary from each other, but will stay "trapped" within the attractor, in the same way as above[2].

[2] Differentiating cells in multicellular organisms have been similarly viewed as shifting away from each other while falling into

Finding Attractors

Finding attractors in the multidimensional composition space is not easy, and various clustering techniques can be employed. We used a simple algorithm that is by no means optimal. The algorithm has two components: *filter* and *clusterizer*. The filter checks that a new assembly s_n, upon split, is similar enough to its parent ($H(s_n, parent(s_n)) \geq T_{similar}$) and that such similarity has been maintained for the last T_{stable} splits. If it has, the clusterizer is invoked to decide to which cluster s_n belongs. The clusterizer holds a list of assemblies, $L = (s_1, s_2, ..., s_k)$, where assembly s_i represents the cluster i. When invoked, the clusterizer finds $s_m \in L$ such that $H(s_n, s_m)$ is maximal, that is, s_m that is most similar to s_n. If $H(s_n, s_m) < T_{similar}$ then the match is not considered good enough and the clusterizer adds s_n to L, thus creating a new cluster. The thresholds $T_{similar}$ and T_{stable} can be adjusted. The list L resulting from a simulation approximates the attractors.

Results

This section summarizes our preliminary results and observations, obtained from a series of experiments conducted using three types of simulation settings.

Type 1: Basic GARD

To establish reference results we first configured Lipidia to simulate basic GARD (Segre et al., 2000). It was achieved by: creating a 1×1 grid with only one location, setting the environment model to infinite, and setting the reaper to maintain a population of a single assembly.

Under these conditions some attractors were discovered, which means that self-replicating assemblies of various species were found. This result is consistent with previous works with GARD (Segre et al., 2000; Segre et al., 1998). It should be noted that in these settings, a very small number of attractors, sometimes a single one, tended to dominate and to attract most of the assemblies. Assemblies did occur in other attractors, but rarely (Fig. 1).

Type 2: Multi GARD with infinite environment

In the second type of experiments we kept the same conditions as in the first, except for setting the reaper to maintain a constant population of 16 assemblies, instead of one. Because the environment was infinite, the 16 assemblies did not have any effect on each other, and developed independently. The results obtained were therefore similar to the first type (Fig. 2). The overall simulated time extended to about $\frac{1}{16}$th of experiments of type 1. This was expected since the unchanged total of 15,000,000 reactions was "consumed" by 16 assemblies developing in parallel, instead of by one.

At time 50, about 30 attractors were discovered in the type-2 experiments, while at the same time about 18 were discovered in type-1 experiments. Not surprisingly, the "parallel search" done by the 16 independent assemblies found more attractors than a single assembly did (for same amount of time). However, for the same total number of reactions, the type-1 experiments yielded a discovery of almost twice as many attractors as the type-2 experiments did. This "inefficiency" in type-2 experiments, in terms of number of reactions, may be attributed to the independence of the 16 assemblies: because there is no coupling between the assemblies, nothing prevents many of them of doing "the same". Such coupling was established in experiments of type 3 .

Type 3: Multi GARD with finite environment

In the third type of experiments we replaced the infinite environment model with a finite environment model. All other settings were kept as in the second type. As Fig. 3 shows, this change was significant.

At time 100, about 24 attractors were discovered in type-1 experiments, while at the same time about 130 were discovered in type-3 experiments: more than a five-fold increase in attractor discovery rate! The discovery was also more efficient in terms of reactions: for the same total number of reactions, experiments of type 3 yielded the discovery of almost three times as many attractors as experiments of type 1 did. It should also be noted that in experiments of type 3 assemblies have occurred more frequently in more attractors. Diversity has increased.

The increase of diversity is attributed to essential molecules coming in short supply in the finite environment. In the case of infinite environment, the composition that produces the fastest stably growing assemblies is quickly becoming dominant. Optimal assemblies can always be constructed to produce the strongest auto-catalysis possible by picking the most suitable molecules from the given molecular repertoire. This, however, is not possible with finite environment. Some optimal assemblies can surly be constructed but their very construction consumes the molecules essential for their own composition. Their count in the population is therefore limited and new niches are becoming available for compositions that take advantage of the remaining molecular repertoire.

Conclusions

We described Lipidia; a new simulation system that allows to conduct experiments with a population of lipid-like assemblies on a two dimensional grid, using finite and infinite environment models. We further described a series of experiments performed using Lipidia. Our results show that a finite environment produces more attractors (species), and faster, than an infinite environment. A finite environment allows more assemblies to occur in more attractors and in greater numbers. Thus, diversity increases.

The results might be considered surprising. One might think that having an infinite supply of resources, in the form

stable cell types, that is, attractors (Kauffman, 1995).

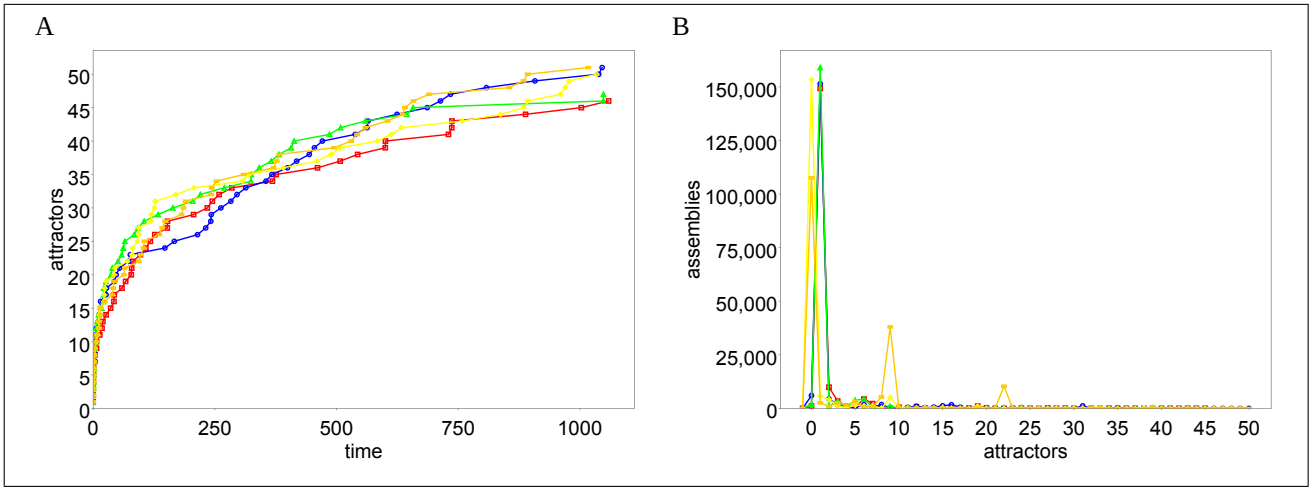

Figure 1: Results of Lipidia simulating basic GARD. (A) The number of attractors as they are discovered in time. (B) Attractor frequencies: the number of assemblies that occurred in each attractor throughout the entire simulation. Attractors are numbered sequentially as they are discovered, thus each attractor number identifies an attractor and also denotes the order of discovery. The graphs show that a very small number of attractors tend to dominate while others are rarer. Results were obtained using a grid with a single location and an infinite environment model. Population was limited to a single assembly and the environment included 100 molecules for each of the N_G types. The first assembly was seeded. Here $N_G = 100, k_f = 0.01, k_b = 0.00001$ and $N_0 = 80$. The rate enhancement factors β_{ij} were sampled from a log-normal distribution with mean $\mu = -6$ and standard deviation $\sigma = 4$, in accordance with RAD model (Lancet et al., 1993). The experiment was repeated 5 times, each simulated for 15,000,000 reactions. Each run is shown in a different shade.

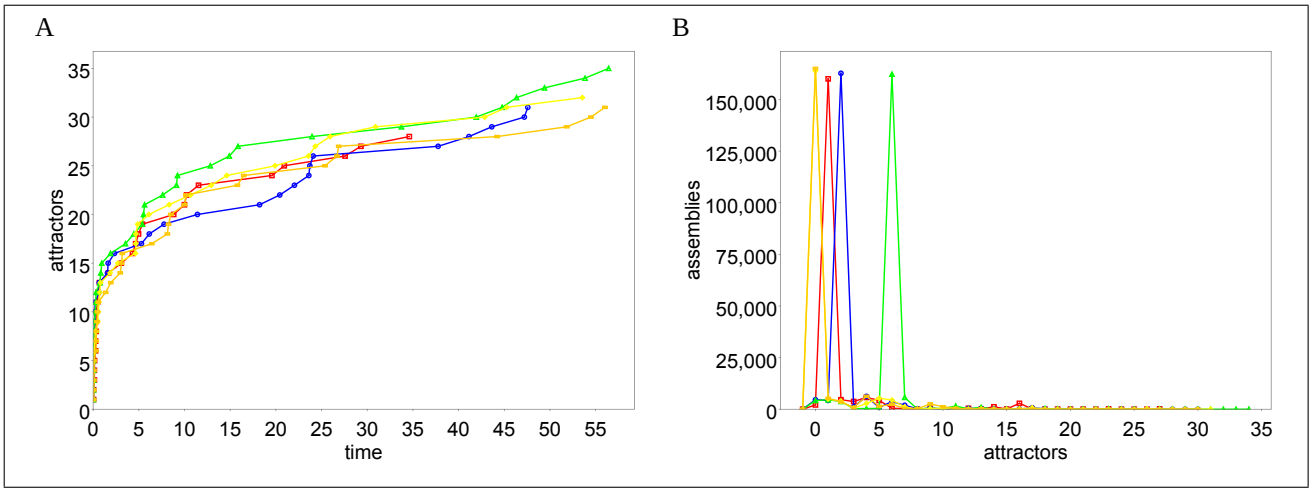

Figure 2: Results of Lipidia simulating 16 assemblies in an infinite environment. (A) The number of attractors as they are discovered in time. (B) Attractor frequencies: the number of assemblies that occurred in each attractor throughout the entire simulation. The graphs show that a single attractor tends to dominate while others are rarer. Attractor discovery rate is improved due to parallelism, but that parallelism is wasteful in terms of reactions: many of the 16 assemblies are "doing the same". Results were obtained using the same settings as in Fig. 1, except the population limit that was set to 16 assemblies, instead of one.

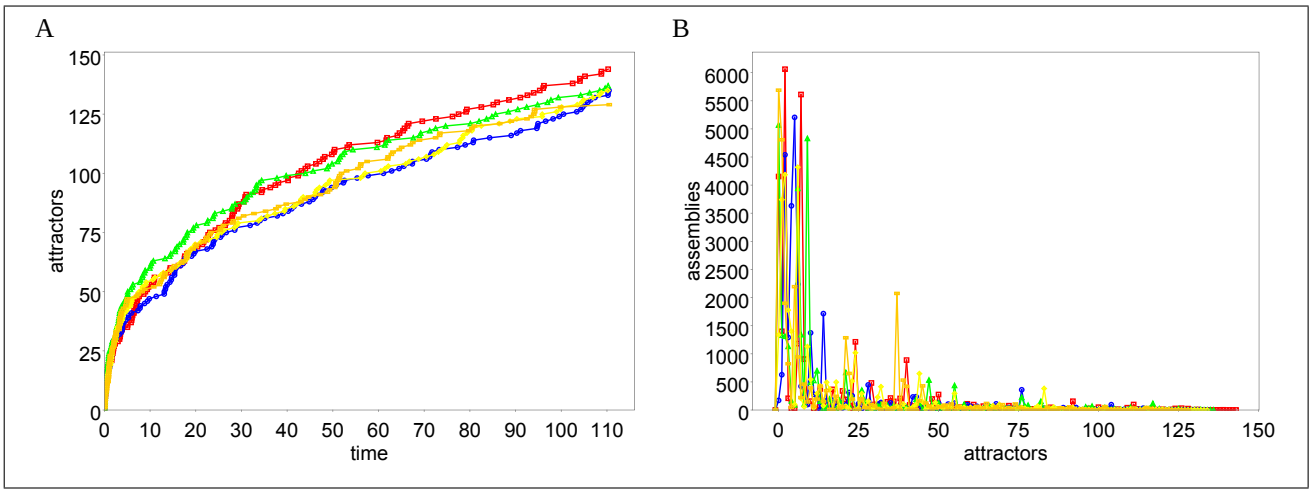

Figure 3: Results of Lipidia simulating 16 assemblies in finite environment. (A) The number of attractors as they are discovered in time. (B) Attractor frequencies: the number of assemblies that occurred in each attractor throughout the entire simulation. The graphs show that the introduction of a finite environment accelerates the discovery of attractors and relaxes the dominance of the single attractor, or the very few ones. Results were obtained using the same settings as in Fig. 2, except for the infinite environment model that was replaced with a finite environment model.

of "food" molecules, might help to "do more". According to our results, it only helps to "do more of the same". Diversity seems to spring when resources are limited. It is when resources for the "best solutions" run out that the race towards alternative solutions begins.

A complete description of Lipidia has been given, although some of its features have not yet been used and are due for future work. Lipidia is capable of simulating a few hundred assemblies within a few hundred grid locations. Future experiments could use these large scale capabilities to explore large grids containing large assembly populations. Diffusion, birth, and death could also be employed. Lipidia is implemented in Java and can run on many platforms. It is available online at: `http://ool.weizmann.ac.il/lipidia`.

References

Bachmann, P. A., Luisi, P. L., and Lang, J. (1992). Autocatalytic self-replicating micelles as models for prebiotic structures. *Nature*, 357:57–59.

Deamer, D. W. (1997). The first living systems: a bioenergetic perspective. *Microbiology and Molecular Biology Reviews*, 61(2):239–61.

Gilbert, W. (1986). The RNA world. *Nature*, 319:618.

Gompper, G. and Schick, M. (1994). *Self-assembling amphiphilic systems*. Academic Press, London.

Joyce, G. F. (2002). The antiquity of RNA-based evolution. *Nature*, 418(6894):214–21.

Kauffman, S. A. (1995). *At Home in the Universe: The Search for the Laws of Self-Organization and Complexity*. Oxford University Press.

Lancet, D., Sadovsky, E., and Seidemann, E. (1993). Probability model for molecular recognition in biological receptor repertoires: Significance to the olfactory system. *Proceedings of the National Academy of Sciences of the USA*, 90(8):3715–9.

Luisi, P. L., Walde, P., and Oberholzer, T. (1999). Lipid vesicles as possible intermediates in the origin of life. *Curr. Opin. in Colloid and Interface Science*, 4(1):33–39.

Segre, D., Ben-Eli, D., Deamer, D. W., and Lancet, D. (2001). The lipid world. *Origins of Life and Evolution of the Biosphere*, 31(1-2):119–45.

Segre, D., Ben-Eli, D., and Lancet, D. (2000). Compositional genomes: Prebiotic information transfer in mutually catalytic noncovalent assemblies. *Proceedings of the National Academy of Sciences of the USA*, 97(8):4112–7.

Segre, D. and Lancet, D. (2000). Composing life. *EMBO Reports*, 1(3):217–22.

Segre, D., Lancet, D., Kedem, O., and Pilpel, Y. (1998). Graded autocatalysis replication domain (GARD): Kinetic analysis of self-replication in mutually catalytic sets. *Origins of Life and Evolution of the Biosphere*, 28(4-6):501–14.

Shenhav, B., Kafri, R., and Lancet, D. (2004). Graded artificial chemistry in restricted boundaries. In *Proceedings of Artificial Life IX (this issue)*, Boston, MA. MIT Press.

Szostak, J. W., Bartel, D. P., and Luisi, P. L. (2001). Synthesizing life. *Nature*, 409:387–390.

Tanford, C. (1978). The hydrophobic effect and the organization of living matter. *Science*, 200(4345):1012–8.