

# SimSoup: Artificial Chemistry Meets Pauling

Chris Gordon-Smith

SimSoup

c.gordonsmith@gmail.com

<http://www.simsoup.info>

**Abstract.** Theories of the Origin of Life can be categorised as ‘template replication first’ and ‘metabolism first’. A key question for metabolism first theories is the mechanism for transfer of inherited information. Earlier work presented a mechanism based on catalytic cycles, along with supporting results from the SimSoup artificial chemistry simulator<sup>1</sup>. The current paper presents an enhanced SimSoup model that is closer to real chemistry and more open ended. Molecules and the types of Interactions between them are constructed by the model itself using simple rules based on valence theory. Results of a preliminary run of the model are presented. Most of the Molecules produced are of a few simple types with low molecular weight. There is a ‘long tail’ of many low frequency Molecules, many of which are more complex with high molecular weight.

## 1 Introduction

### 1.1 Motivation

There is currently no complete and generally accepted explanation of the Origin of Life. Evolution requires inherited information to be passed from parent to offspring; contemporary organisms do this using template replicating molecules (eg DNA). However, the mechanism requires the assistance of highly evolved proteins, which would not have been present in the prebiotic world. Metabolic theories of the Origin of Life propose that early organisms were metabolic systems that transmitted inherited information without the use of template replicators.

The motivation for the SimSoup project is to show that a metabolic network has sufficient information carrying properties to enable evolution to begin without the assistance of highly evolved molecules such as proteins or DNA.

### 1.2 Conceptual Background

The conceptual background for SimSoup includes the following:-

- The metabolic theories of Aleksandr Oparin (Oparin [8]), Stuart Kauffman (Kauffman [7]), Freeman Dyson (Dyson [2]), Fernando and Rowe (Fernando and Rowe [3]), and more specifically the Lipid World and the GARD model of Doron Lancet’s group (Segré et al. [10])

---

<sup>1</sup> SimSoup has been implemented as a computer simulation. The C++ source code is available at Ref [11]

- Network theory, particularly the work of Sanjay Jain and Sandeep Krishna (Jain and Krishna [6])
- Günter Wächtershäuser’s chemo-autotrophic Iron-Sulphur World (Wächtershäuser [13] [14])
- Chemical bond theory, as presented in Pauling [9].

## 2 The SimSoup Artificial Chemistry Model

### 2.1 SimSoup as a Network Simulator

The following summarises the model presented in Gordon-Smith [4] and [5]:

- **Chemical Network:** A network is defined in which the nodes are Molecule Types and connections are Interaction Types. Molecule Types have Mass and Potential Energy, but no other properties. Interaction Types take three forms: Construction ( $A + B \rightarrow C$ ), Fission ( $A \rightarrow B + C$ ), and Transformation ( $A \rightarrow B$ ). Each Interaction Type must conserve Mass, and has an Activated Complex Energy that is used along with the Potential Energies of the Molecule Types to determine the forward and reverse Rate Constants. This makes the model thermodynamically realistic
- **Network Dynamics:** Interactions take place between Molecules in a well stirred Reactor. Each bimolecular Interaction Type (Construction) occurs at a rate equal to the product of the Rate Constant and the concentrations of the two Reactants. Each unimolecular Interaction Type (Fission or Transformation) takes place at a rate equal to the product of the Rate Constant and the concentration of the (single) Reactant
- **Compound Interactions:** Interaction Types can be combined in ways that have catalytic properties. For example,  $A + X \rightarrow I$  followed by  $I + B \rightarrow J$  and finally  $J \rightarrow X + C$  together make up a Compound Interaction with overall scheme  $A + B \xrightarrow{X} C$ , with X operating as a catalyst, and I and J as intermediates
- **Trackers and Cycle Detection:** A Tracker is an object that can be attached to a Molecule. As Molecules take part in Interactions, the Trackers are passed from Reactant Molecule to Product Molecule. This enables cycles to be detected and monitored
- **Data Series Plots:** These show the real-time behaviour of a range of variables that are monitored as the simulation runs
- **Manhattan Plot:** This shows the variability in the composition of the material in the Reactor over time

### 2.2 SimSoup Extended as a Network Explorer

**A More Open-Ended Approach - Adding Molecular Structure:** In the earlier model of Section 2.1, the chemical network is defined as part of each model scenario. This limits the scope for the system to generate novelty. To deal with this, SimSoup has been enhanced as follows:

- Each Molecule Type has a structure
- Molecules can Join or Split to form Molecules of different types
- Joining and Splitting occur in ways that depend on the structural properties of the Reactant(s)

The rules that determine the structure of Molecules and the way in which they Join and Split are designed to be analogous to real chemistry, while keeping the model conceptually simple and computationally tractable. The key objective is to produce an *open ended model in which the opportunities for novelty are similar to those that exist in real chemistry.*

**Overview of Molecular Structure in SimSoup:** Molecules are two dimensional rigid structures built from Atoms that are bonded together such that they occupy fixed positions on a square ‘Board’ (similar to a chess board). Each square may contain at most one Atom. Bond angles are always  $90^\circ$  or  $180^\circ$ , and bond lengths are all equal. Atoms bond together in a way that is broadly consistent with valence bond theory in real chemistry, and with comparable bond energies.

**Atom Types:** SimSoup Atoms are of different types, with an Atom Type being characterised by Mass, plus the following properties:-

- $N_{\text{ElecFullShell}}$ : Number of electrons to fill the outer electron shell
- $N_{\text{ValElec}}$ : Number of (valence) electrons in the outer shell.

This is by analogy with real chemistry, in which the ability of atoms to bond depends on shell structure. Atoms with full shells include stable noble gases, which rarely bond to other atoms.

A SimSoup Atom can bond to other Atoms, with the maximum number of bonds that can be supported being  $N_{\text{PossBond}} = N_{\text{ElecFullShell}} - N_{\text{ValElec}}$ .<sup>2</sup>

$N_{\text{ElecFullShell}}$  and  $N_{\text{ValElec}}$  (and so  $N_{\text{PossBond}}$ ) for SimSoup Atom Types are based on real chemistry. The Atom Types modelled are Hydrogen, Carbon, Nitrogen and Oxygen, with values of 1,4,3 and 2 respectively for  $N_{\text{PossBond}}$ .

**Bonds, Bond Order and Bond Energy:** SimSoup models single, double and triple order (covalent) bonds, so that each of the following is possible: O—H, C=O, O=C=O, C≡N. Bond Energies are the averages for bonds in real chemistry, taken from Pauling [9] and Atkins [1].

**Example Molecular Structures:** Figure 1 shows some simple Molecules that have been produced by SimSoup, and the real molecules to which they are analogous. The analogy is not perfect. For example, real methane is three dimensional and tetrahedral with bond angle  $109.5^\circ$ , and the bond angle in real water is  $104.5^\circ$ , rather than  $90^\circ$ . In addition, there are two forms of ‘water’ in SimSoup, the second being linear: H—O—H. The correspondence is closer for acetylene and hydrogen cyanide, which are linear both in real chemistry and in SimSoup.

<sup>2</sup> Real atoms sometimes do not have such a simple limit on the number of possible bonds. The five bonds in phosphorous pentachloride provide an example.

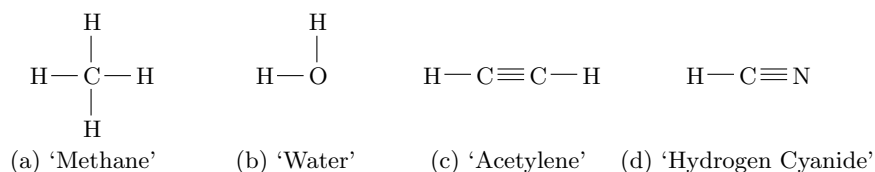


Fig. 1: Example SimSoup Molecules

**Joining Molecular Structures:** Whenever a Molecule Type becomes actualised in the Reactor (the number of Molecules of the type present changes from zero to one), it is checked against all the other actualised Molecule Types to determine, in each case, whether a join is possible.

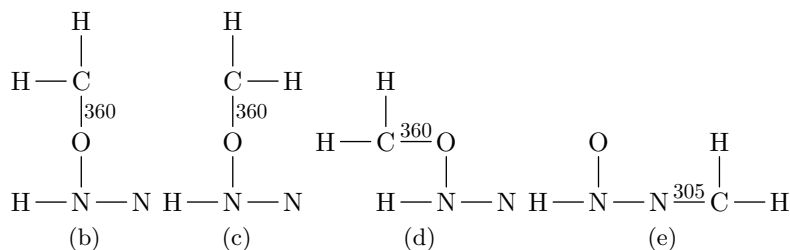
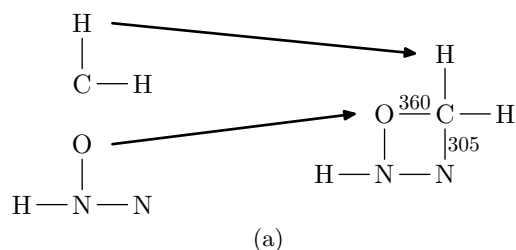


Fig. 2: (a) Example Construction showing Reactants and most stable (highest total bond energy) Product. The new bonds created are labelled with the bond energy ( $\text{kJ mol}^{-1}$ ) in each case. (b) to (e) show some of the possible alternative Products.

Two Molecules can Join (in a Construction Interaction) subject to the following rules:

- Two Atoms cannot occupy the same Board position in the joined Molecule
- No Atom can have more than  $N_{\text{PossBond}}$  bonds
- Bond Energies are the averages for the corresponding real chemistry bonds
- When a pair of Atoms is considered for bonding, the highest order bond that is feasible (within the constraint on  $N_{\text{PossBond}}$  for each Atom) is formed

- Molecules Join in a way that maximises total Bond Energy. If multiple Joins with the same (highest) energy are possible, the first one found is selected.

Figure 2 shows an example Interaction Type (a Construction) in which two Molecules Join to form a single Product. The figure also shows several (but not all) of the ‘rejected’ Products with lower total bond energies. The Product in Figure 2(a) is the only one that entails the formation of two new Bonds.

**Splitting Molecular Structures:** When a Molecule Type is actualised for the first time, a Fission Interaction Type is determined. This defines the two Products into which a Molecule of the type can Split. The Products are identified by finding a set of Bonds that, if broken, would Split the Molecule into two unconnected parts. Where there are multiple ways in which the Molecule can be split, as is usually the case, the split identified is one that entails breaking bonds with the lowest possible total energy.

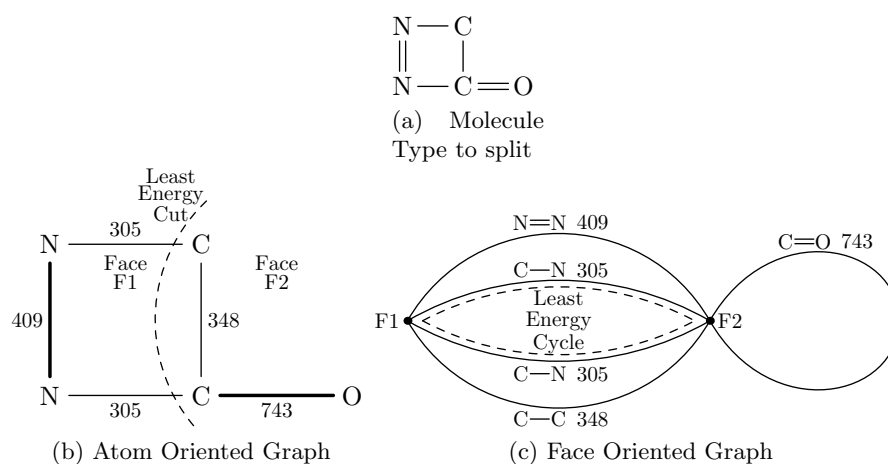


Fig. 3: Illustration of the SimSoup Splitting algorithm

The splitting algorithm works on Molecules of arbitrary complexity. It uses techniques of graph theory<sup>3</sup>, and is illustrated in Figure 3. The major steps are:

- Represent the Molecule Type to be split as an ‘Atom Oriented Graph’
- Transform this into a ‘Face Oriented Graph’
- Use the Dijkstra shortest paths algorithm to find a least energy cyclic path in the Face Oriented Graph. The path edges identify the Bonds to break.

The remainder of this section explains the algorithm.

<sup>3</sup> The SimSoup implementation makes use of the Boost Graph Library code.

*Representing The Molecule as an Atom Oriented Graph:* Each node in this graph represents an Atom in the Molecule to be split, and each edge represents the (single or multiple) Bond between the Atoms that the edge connects. Each edge has a weight indicating Bond Energy. Figure 3(b) shows the Atom Oriented Graph for the Molecule Type in Figure 3(a). The thicker edges represent double bonds. The numbers by the edges are the Bond Energies ( $\text{kJ mol}^{-1}$ ).

The Atom Oriented Graph is always planar in SimSoup<sup>4</sup>; that is, it can be embedded in a flat plane in such a way that the edges intersect only at their endpoints. A planar embedding divides the plane into regions or *faces*. One face is unbounded or *external*<sup>5</sup>. In Figure 3(b), face F2 is the external face.

It can be shown that for any single component planar graph, *separating the graph into two components requires a ‘cut’ that starts on one face, crosses one or more edges, and returns to the starting face*. The task is to find a ‘least energy cut’ that breaks the bonds with the least total energy.

It can be seen from Figure 3(b) that the structure will split with the breaking of two different bonds. Although the least energy cut crosses two edges, the energy is less than for the alternative cut that starts and finishes on face F2 and removes only the one edge for the C=O double bond.

*The Role and Usage of the Face Oriented Graph:* For a Molecule of arbitrary complexity there will typically be many faces and many possible cuts, and it is not immediately obvious how to find the least energy cut.

The problem can be dealt with by transforming the Atom Oriented Graph to a form that enables the well known Dijkstra shortest paths algorithm to be used. This is the purpose of the transformation to the Face Oriented Graph.

The vertices in the Face Oriented Graph correspond to the faces in the Atom Oriented Graph. The edges correspond to the edges in the Atom Oriented Graph, but the vertices (or vertex) to which an edge is connected correspond(s) to the face(s) on either side of the edge in the Atom Oriented Graph. For example, the C—C bond in Figure 3(b) separates faces F1 and F2; it corresponds to the lowest edge in Figure 3(c) that joins vertices F1 and F2.

The problem of finding a least energy cut of the Atom Oriented Graph is equivalent to finding a *cyclic path through the Face Oriented Graph whose edges have a total Bond Energy less than or equal to that for any other cyclic path*.

SimSoup uses Dijkstra’s shortest path algorithm to find such a path, with the energy of each edge being interpreted as its ‘length’. Each edge in the Face Oriented Graph is considered in turn, and Dijkstra’s algorithm is used to find a (different) least energy path (if any) between the edge’s endpoints, thereby completing a cycle. The cycle ‘length’ is the energy of the edge plus the total energy of the (different) shortest path. By considering each edge in this way, a ‘shortest’ cycle with the lowest total energy is found. The edges in the cycle identify the bonds to break to split the Molecule Type.

<sup>4</sup> This is a result of the ‘Board’ layout of Molecules and the fact that Bonds are always with Atoms in neighbouring board squares.

<sup>5</sup> The external face has no particular significance. If the graph were embedded in the surface of a sphere there would be no external face.

### 3 Preliminary Results

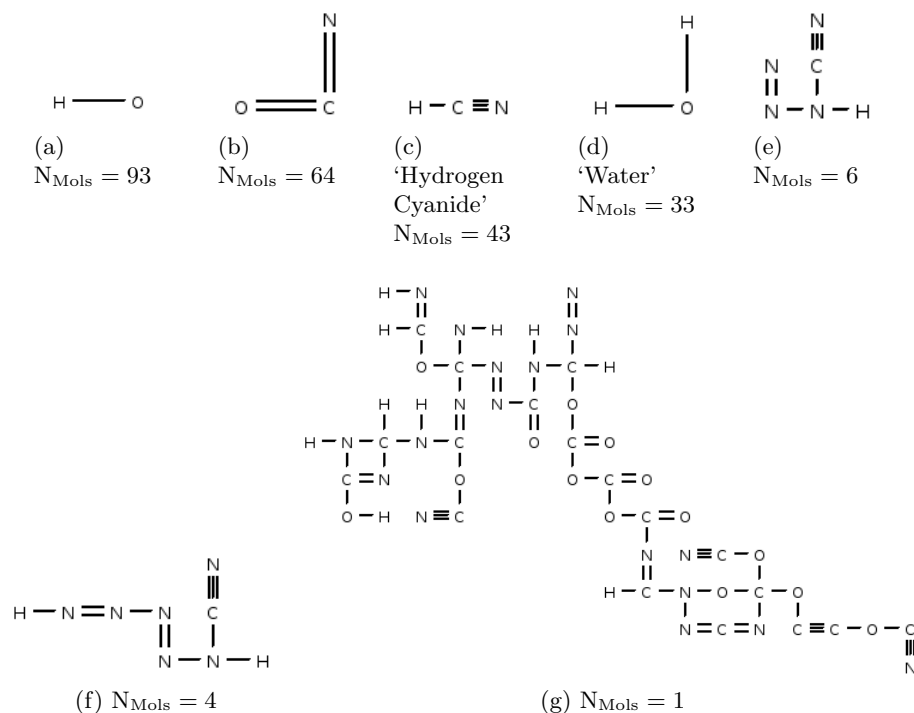


Fig. 4: Molecule Types in a preliminary model run, and number ( $N_{\text{Mols}}$ ) in each case

Some preliminary output<sup>6</sup> is included in Figure 4. The figure shows some Molecule Types present in the SimSoup Reactor at the end of a run. The run was setup such that 10 Atoms of (each of) Hydrogen, Carbon, Oxygen and Nitrogen were added at intervals of 100 timesteps. After 5000 timesteps, there were 74 Molecule Types actualised in the Reactor. This was about 1% of the total number of Molecule Types that had been actualised during the run. The total number of Molecule Types discovered was 33383, not all of which had been actualised.

Of the 74 Molecule Types present, a few of the 11 most frequent are shown (Figure 4(a) to 4(f)), along with the number of Molecules ( $N_{\text{Mols}}$ ) in each case. For the other 63 Molecule Types, only one of each was present. Several of these were larger Molecules with more than 15 Atoms. Figure 4(g) shows an example with 60 Atoms.

<sup>6</sup> Results were produced using SimSoup revision 97, source code available at Ref [11].

## 4 Conclusions and Prospects

### 4.1 Conclusions

The SimSoup model represents chemistry at the level of elementary reactions (unimolecular and bimolecular), with catalysis as a property of the network. The model has been extended to represent molecular structure in a way that is analogous to real chemistry.<sup>7</sup> This makes it capable of exploring the behaviour of a vast space of (artificial) metabolic networks in a very open ended way.

Output from a preliminary model run has been presented. Most of the Molecules produced are small and of only a few types. The remaining Molecules exist in low numbers, but are more diverse and typically have a transient existence. At the end of the run, there were 74 Molecule Types present in the Reactor, but in excess of 7000 different Molecule Types had been actualised during the run.

### 4.2 Prospects

Possible avenues for future investigation using SimSoup include:

- High activity networks and the role played by catalytic cycles
- The ability of such networks to ‘remember’ perturbations and thereby carry inherited information
- The role of molecular structure in (non template based) inheritance

## References

1. Atkins, P.: Physical Chemistry. Oxford University Press (2006)
2. Dyson, F.: Origin Of Life. Cambridge University Press (1999)
3. Fernando, C., Rowe, J.: Natural selection in chemical evolution. *Journal of Theoretical Biology* **247** (2007) 152-167
4. Gordon-Smith, C.: SimSoup: An Artificial Chemistry Model for Investigation of the Evolution of Metabolic Networks. ECAL 2005, Artificial Chemistry workshop paper. <http://www.simsoup.info/Publications.html>
5. Gordon-Smith, C.: Evolution Without Smart Molecules. ECAL 2007, Extending the Darwinian Framework workshop paper. <http://www.simsoup.info/Publications.html>
6. Jain, S., Krishna, S.: Autocatalytic Sets and the Growth of Complexity in an Evolutionary Model. <http://arxiv.org/abs/adap-org/9809003>
7. Kauffman, S.A.: The Origins Of Order. Oxford University Press (1993)
8. Oparin, A.I.: The Origin Of Life On The Earth. Oliver And Boyd (1957)
9. Pauling, L.: The Nature Of The Chemical Bond. Cornell University Press (1960)
10. Segré, D., Dafna, B., Deamer, D., Lancet, D.: The Lipid World. *Origins Life Evol. Biosphere* **31** (2001) 119-145
11. SimSoup Source Code: <http://code.google.com/p/simsoup/>
12. Szathmáry, E.: The origin of replicators and reproducers. *Phil. Trans. R. Soc.* (2006) **361** 1761-1776.
13. Wächtershäuser, G.: Evolution of the first metabolic cycles *Proc. Natl. Acad. Sci. USA* **87** (1990) 200-204
14. Wächtershäuser, G.: The Origin of Life and its Methodological Challenge. *J. Theor. Biol.* **187** (1997) 483-494

<sup>7</sup> Resonance and electronegativity are not modelled. Doing so could be considered, but is not essential for the current objective of making the model more open ended.