# Supplementary Materials

## Source code

All the source code along with results of numerical calculations, which were used to produce the results of the current article are available on GitHub at https://github.com/kkkmail/CLM/.

The system uses various methods for finding solutions. Two major methods are called "Evolution Run" and "Bifurcation Run". The first method uses Mathematica differential equation solver (NDSolve) to find an evolutionary solution. The second method performs bifurcation analysis first by finding the fixed points of the system and then performing stability analysis by finding eigenvalues of the system linearized near fixed point(s). This method then attempts to maximize the real part of the eigenvalues by varying the parameters of the model, thus trying to find the values where bifurcation can occur. However, as the complexity of the system increases, numerical bifurcation analysis becomes less and less stable due to the following reasons. First, as the complexity increases, the overall polynomial power of the system grows rapidly. Finding the roots of polynomials numerically becomes a challenging task and it requires a substantial increase in precision, sometimes up to 50-100 digits. However, the worst part comes from the fact that a linearized matrix often has at least one zero or nearly zero eigenvalue. That makes the system insusceptible to the change of variables. Subsequently the minimization problem becomes too complex too fast and all together the solver often cannot find the solution. It is our current view that it is close to impossible to utilize bifurcation analysis for the systems with peptide lengths larger than three, especially if many parameters are used. Therefore, we used the evolutionary method in the current work.

Homogeneous and isotropic chemical systems can be fully described by specifying all the reactions in such systems. Each reaction is described by input and output reagents as well as the coefficient of the reaction. Therefore, given the description of the reactions, a system of ordinary differential equations, which describe such a system, can be generated. The system core does exactly that and it allows adding arbitrary reaction types by specifying such descriptors of reactions. A descriptor is effectively a stoichiometric matrix along with the coefficients of forward and backward reactions.

However, crystallization is not a chemical reaction but rather a complicated physical process, which depends on many conditions. Therefore, some simplifications were utilized to keep the solution time within reasonable limits while making sure that the obtained results have meaning. Two different models of crystallization are implemented in the system. One uses a simplified Noyes-Whitney equation, which treats the "rates" of forward and backward crystallization as a unified function of concentration of the relevant substance and a total amount of its sediment. This process is reversible, and it is slow to compute. An alternative method, which we called the "direct crystallization" utilizes the fact that crystallization is relatively fast in comparison to chemical reactions in consideration and that the system is designed to remove any sediment. The required assumption is that a substance, for which we apply "direct crystallization" method, must have very low solubility. A comparison between Noyes-Whitney type sedimentation with subsequent removal of the sediment and "direct crystallization" for peptides up to level three was performed to ensure that the latter method produces meaningful results. No significant discrepancies were found.

In addition, there are various statistical models used in the system. The purpose of these models is to generate some distributions of coefficients in large systems. This statistical part was not used in the current article.

System core is in subfolder **Kernel**. The modules to run various models are in subfolder **Run**. A detailed description has not yet been prepared but comments can be provided upon request.

## Main results of the article

The results used for the preparation of the current article are in the folder:

https://github.com/kkkmail/CLM/tree/master/Results/ISSOL\_201612

and below.

File https://github.com/kkkmail/CLM/blob/master/Results/ISSOL\_201612/CLM\_ResultInfo.m plots figures, which graphically represents the result of the current article. Parameter **PathList** must be set to point to the correct location of the data (described below) and then parameter **resultID** should be varied from 1 to 4 to plot the relevant figures. The naming conventions are described below.

The code for APED-related part of the article is in the folder: https://github.com/kkkmail/CLM/tree/master/Results/ISSOL\_201612/Calc/APED.

There are only two files there and, therefore, no substantial naming convention was introduced.

## Naming and Folder conventions

All the raw calculations results are in folder **Calc** and precomputed data files are in **Data**.

Convention of naming files and folders is easier to explain using some examples. Consider a file (Name 1):

**CLM\_ER\_A\_Cmp\_Lzz\_En\_L5\_\_P0\_\_324\_\_Res\_03.m**

in the folder:

https://github.com/kkkmail/CLM/tree/master/Results/ISSOL\_201612/Calc/Evolution/L5/A/Cmp/g%3D010

and a file (Name 2):

**CLM\_ER\_NA\_Cpm\_Lzz\_En\_L5\_lm030\_\_P0\_\_323\_\_R1.m**

in the folder:

https://github.com/kkkmail/CLM/tree/master/Results/ISSOL\_201612/Calc/Evolution/L5/NA/Cpm/g%3D010

Combined with the relevant part of the path they look like:

**Evolution/L5/A/Cmp/g=010/CLM\_ER\_A\_Cmp\_Lzz\_En\_L5\_\_P0\_\_324\_\_Res\_03.m**

**Evolution/L5/NA/Cpm/g=010/CLM\_ER\_NA\_Cpm\_Lzz\_En\_L5\_lm030\_\_P0\_\_323\_\_R1.m**

Enantioselective reactions are coded by a capital letter of the reaction type (**C** – catalysis, **L** – ligation (polymerization), **E** – epimerization), followed by one or two lower case letter coding for enantioselectivity (**p** – positive, **m** – negative, **z** – no enantioselectivity, **n** – no reaction). Two letter coding is used when there are distinct forward and backward reactions. Epimerization effectively does not have a reverse reaction because it is symmetric to the relabeling of molecules.

The underscores serve as separators and so these folders/files have the following parts:

|  |  |  |
| --- | --- | --- |
| Name 1 | Name 2 | Description |
| Evolution | Evolution | Folder where all evolution models are stored |
| L5 | L5 | Models with peptides up to length 5 |
| A | NA | **A** means models with activation. Models without activation have value **NA**. |
| Cmp | Cpm | Forward enantioselectivity of a given catalyst is slightly negative (**m**inus) and backward enantioselectivity is slightly positive (**p**lus). The other allowed combinations are **pm** , **zz**, or **nn**. |
| g=010 | g=010 | , whereas the signs are determined by **Cmp** / **Cpm** coding. |
| CLM | CLM | Overall name of the model: **C**hiral **L**ife **M**odel. |
| ER | ER | Prefix to distinguish that the "**E**volution **R**un" engine was used to produce results. The other available engine is "**B**ifurcation **R**un". |
| A | NA | **A** means models with activation. Repeats folder coding for convenience. Models without activation have value **NA**. |
| Cmp | Cpm | Repeats folder coding of signs for forward / backward enantioselectivity for convenience. |
| Lzz | Lzz | **L**igation has **z**ero forward and **z**ero backward enantioselectivity. |
| En | En | **E**pimerization is **n**ot used. |
| L5 | L5 | Repeats folder coding for maximum peptide length (5) for convenience. |
|  | lm030 | Optional. If used, then encodes the value of backward ligation coefficient (0.30). Helps to distinguish the files where only that parameter was varied. |
| P0 | P0 | Page 0. Evolution models with length 5 are calculated in 9 pages (pages 0 through 8) because Mathematica's parallel engine does not go well with internally generated code. Therefore, instead of making parallel evaluations, 9+ independent Mathematica sessions were run simultaneously to speed up computations. |
| 324 | 323 | Major version number. The changes between versions 3.22 through 3.24 were mostly due to some restructuring and, therefore some files may have a run version inside slightly different from the code in the file name. |
| Res\_03 | R1 | **Res\_03** is the result number 3. Some of the results were discarded due to errors or other reasons. Each subsequent run of the same model was assigned a consecutive result number. Alternative coding is just the letter "R" followed by the result number, like **R1**. |

Raw data files are combined in the relevant Excel files, which follow similar naming conventions. Each file has several sheets (the sheets that are not described were not used in the current work and may contain some obsolete data):

1. Sheet **Params** contains the parameters of the model.
2. Results of the calculations were copied into sheets **Cnn+**, **Cnn-** (combined into **CnnAll**), **Cpm+**, **Cpm-**, **Cmp+**, **Cmp-** (combined in **CpmAll**).
3. The result from sheets **CnnAll** and **CpmAll** were then copied into Mathematica modules, which are in the folder **Data**. These Mathematica modules are then used by **CLM\_ResultInfo.m** as a source for plotting various figures.