Uniformity: uncovering the frontier of parallelism

Niall Murphy[⋆1] and Damien Woods[⋆⋆2]

[1] Dept. of Computer Science, National University of Ireland Maynooth, Ireland
nmurphy@cs.nuim.ie
[2] Dept. of Computer Science & Artificial Intelligence, University of Seville, Spain
dwoods@us.es

Abstract. We summarise some current results for active membrane systems using uniformity below P. Many of the systems we consider are easily to simulate on parallel hardware and provide interesting new directions for the complexity theory of membrane systems as well as those seeking to simulate membrane systems.

1 Familiar frontiers

The majority of complexity results to date in membrane systems (also known as P-systems) have been focused on the frontier of tractability. This frontier is also known as the $\mathsf{P} \stackrel{?}{=} \mathsf{NP}$ conjecture. This boundary has been fruitfully explored using polynomial time (semi-)uniform families of membrane systems. However, when the uniformity condition is restricted to being computed in classes below P many new and interesting things about families of active membrane systems without charges ($\mathcal{AM}^0$) become clear.

One result [6] is that logspace semi-uniform families (when dissolution rules are excluded, denoted $\mathcal{AM}^0_{-d}$) solve all problems in NL. (When using P semi-uniformity, families of $\mathcal{AM}^0_{-d}$ were shown to solve all of P. [4]) The problems in NL are solvable using very little memory ($\mathcal{O}(\log^2 n)$) on a deterministic polynomial time Turing machine [11]. Furthermore since $\mathsf{NL} \subseteq \mathsf{NC}$, this places us on the far side of another frontier: the parallelisable frontier. The parallelisable frontier is also known as the $\mathsf{NC} \stackrel{?}{=} \mathsf{P}$ conjecture and is almost as significant in its implications as the $\mathsf{P} \stackrel{?}{=} \mathsf{NP}$ conjecture [2]. Problems in NC ($\cup_{i \geq 0} \mathsf{NC}^i$) are those which are decided in poly-logarithmic ($\mathcal{O}(\log^i n)$) time when using a polynomial number of processors, that is they are efficiently parallelisable. However P-complete problems are thought to be intrinsically sequential and no significant speed-up is achieved when the number of processors working on the problem is increased[2].

We have also shown[5,6] the first P characterisation for $\mathcal{AM}^0$ systems with dissolution rules where the lowerbound is not dependant on P uniformity.

⋆ Funded by the Irish Research Council for Science, Engineering & Technology
⋆⋆ Supported by a Project of Excellence from the Junta de Andalucía, grant number TIC-581.

Another result is that for $\mathcal{AM}^0_{-d}$ the notions of uniformity and semi-uniformity are formally different. This result may be applicable to other types of membrane systems and models of computation.

2 First Results

We now summarise the first results from beyond the P frontier. The key to these results has been to use uniformity conditions weaker than P (note Obtułowicz [9] was the first to explicitly use logspace).

Theorem 1 ([6]). *The set of problems solved by semi-uniform families of recogniser active membrane systems without charges and without dissolution rules equals* NL*, formally* (AC^0)–$\mathsf{PMC}^*_{\mathcal{AM}^0_{-d}} = \mathsf{NL}$*. This result holds if the semi-uniformity function is* AC^0*,* NC^1*,* L*, or* NL *computable.*

With a slight restriction on the way rules are allowed to interact in the system, the set of problems solvable shrinks to L [8]. The proofs of these results show that it is possible to simulate a $\mathcal{AM}^0_{-d}$ system with a membrane structure by using a system with a single membrane and only evolution rules. Also clarified is the power of dissolution, with dissolution a semi-uniform $\mathcal{AM}^0$ system with strong non-elementary division rules solves PSPACE, however without dissolution rules, the system solves only NL [1]. Similarly dissolution rules provide the first P characterisation that is robust to uniformity conditions below P.

Theorem 2 ([5]). *The set of problems solved by semi-uniform families of recogniser active membrane systems without charges and using dissolution and symmetric division rules equals* P*, formally* (AC^0)–$\mathsf{PMC}^*_{\mathcal{AM}^0_{+d,-a}} = \mathsf{P}$*. This result holds if the semi-uniformity function is* AC^0*,* NC^1*,* L*,* NL*, or* P *computable.*

Theorem 1 highlights the importance of choosing an appropriate uniformity condition. If the uniformity function is computed in polynomial time then the $\mathcal{AM}^0_{-d}$ system is found to solve all of the problems in P [4]. Families of membrane systems are sensitive (more than circuits for example) to the strength of their uniformity conditions as the (semi-)uniformity function accesses the input word. Thus an active membrane system with just 2 rules, $[\, a \rightarrow \texttt{yes}\,]$ and $[\, b \rightarrow \texttt{no}\,]$ solves a P-complete problem if its uniformity function is polynomial time computable, the input encoder simply solves the problem using the input word! In Figure 2 we see how the power of a system increases in step with its uniformity until a certain threshold is crossed, intuitively this threshold represents the actual computing power of the system.

Now we consider the power of uniform families of active membranes without dissolution. It has been shown in a number of models that whether one chooses to use uniformity or semi-uniformity does not affect the power of the model. However, we have shown [7] that these notions are not equivalent, resolving Open Problem C in [10]. Our result proves that choosing one notion over another gives characterisations of completely different complexity classes, including

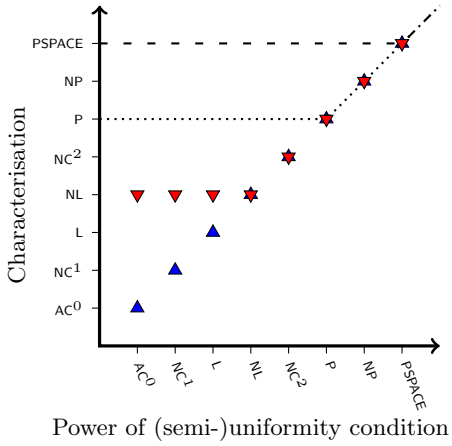

Fig. 1. Complexity classes characterised by membrane systems. Characterisations by uniform families of $\mathcal{AM}^0_{-d}$ systems are denoted by ▲, and semi-uniform by ▼. Theorem 2 is indicated by ····· and a PSPACE-characterisation [1,13] with various (semi-)uniformity conditions indicated by - - , for these, semi-uniform and uniform classes have the same power.

known distinct classes. This is surprising because in all membrane system models studied to date, the classes of problems solved by semi-uniform and uniform families turned out to be equal [1,5,12]. Uniform families of $\mathcal{AM}^0_{-d}$ are so weak that the complexity of their encoding functions (down as far as AC^0) gives the upperbound of solvable problems.

Theorem 3 ([7]). AC^0 *uniform families of active membrane systems without charges and without dissolution rules characterise a strict subset of their semi-uniform equivalent.* $\mathsf{AC}^0 = (\mathsf{AC}^0)\text{–}\mathsf{PMC}_{\mathcal{AM}^0_{-d}} \subsetneq (\mathsf{AC}^0, \mathsf{AC}^0)\text{–}\mathsf{PMC}^*_{\mathcal{AM}^0_{-d}} = \mathsf{NL}$.

3 Applications

We have mentioned that problems in NL (such as $\mathcal{AM}^0_{-d}$ prediction [8]) are those solvable using very little memory on a sequential computer, but we can also exploit the parallelisable aspects of these systems. Recall that $\mathsf{NL} \subseteq \mathsf{NC}^2$, this implies that with a polynomial number of processors the system can be simulated in $\mathcal{O}(\log^2 n)$ time. To simulate recogniser $\mathcal{AM}^0_{-d}$ systems on a parallel processing system (such as CUDA) we use a technique known as *transitive closure* [14]. Given the dependency graph [4] of a recogniser $\mathcal{AM}^0_{-d}$ system, we construct a square binary matrix M (whose size is the square of the number of objects and labels) where the rows and columns both represent all object-membrane combinations. Coordinates $M_{\langle o,h\rangle,\langle u,g\rangle} = 1$ where there is an edge linking (o,h) and (u,g) in the graph, all other coordinates in the matrix are 0. By squaring

this matrix log times we calculate the transitive closure of the dependency graph. If this yields a 1 in the matrix at coordinate $M_{\langle x,in\rangle,\langle yes,out\rangle}$, where x is an input object and *yes* is an output object, then the system is an accepting one. This efficient simulation technique indicates that $\mathcal{AM}^0_{-d}$ systems naively make good choice to model cellular systems.

4 Conclusions and open problems

In membrane systems it is vital to choose the correct uniformity condition. If the uniformity is too strong you may miss the true power of the system you are trying to analyse. For example, a P upper bound result for Tissue systems using a dependency graph such as [3] can be trivially tightened to NL if a more suitable uniformity is used. Since AC^0 has a strong separation from other classes it makes an excellent choice for a uniformity condition.

The problems in NP are intractable, any attempt to use brute force parallelism to solve them will run short of processors. However problems in NC (e.g. $\mathcal{AM}^0_{-d}$ simulation) are easy to solve in parallel. The transitive closure technique is applicable for any system that can be modeled using a dependency graph.

Some open problems that this work has raised.

1. Can all recogniser active membrane systems without charges be simulated by a system with at most one copy of each object?
2. Can we characterise the levels of the NC hierarchy (or polynomial hierarchy) using active membrane systems?
3. What happens if we adjust the membrane uniformity definition to remove the encoding of the input, making it similar to circuit uniformity?
4. For which systems do the semi-uniform and uniform versions have different computing power?

References

1. A. Alhazov and M. J. Pérez-Jiménez. Uniform solution to QSAT using polarizationless active membranes. In *MCU*, volume 4664 of *LNCS*, pages 122–133. Springer, 2007.
2. R. Greenlaw, H. J. Hoover, and W. L. Ruzzo. *Limits to parallel computation:P-completeness Theory*. Oxford University Press, Oxford, 1995.
3. R. Gutiérrez-Escudero, M. J. Pérez-Jiménez, and M. Rius-Font. Characterizing tractability by tissue-like P systems. In *BWMC7*, volume 2, pages 169–180, 2009.
4. M. A. Gutiérrez-Naranjo, M. J. Pérez-Jiménez, A. Riscos-Núñez, and F. J. Romero-Campero. Computational efficiency of dissolution rules in membrane systems. *International Journal of Computer Mathematics*, 83(7):593–611, 2006.
5. N. Murphy and D. Woods. Active membrane systems without charges and using only symmetric elementary division characterise P. In *Invited Papers from WMC8*, volume 4860 of *LNCS*, pages 367–384. Springer, 2007.
6. N. Murphy and D. Woods. A characterisation of NL using membrane systems without charges and dissolution. *UC7 2008, LNCS*, 5204:164–176, 2008.

7. N. Murphy and D. Woods. The computational complexity of uniformity and semi-uniformity in membrane systems. In *BWMC7*, volume 2, pages 73–84, 2009.
8. N. Murphy and D. Woods. On acceptance conditions for membrane systems: characterisations of L & NL. In *EPTCS vol 1: CSP2008*, pages 172–184, 2009.
9. A. Obtułowicz. Note on some recursive family of P systems with active membranes. `http://ppage.psystems.eu/index.php/Papers`, 2001.
10. G. Păun. Further twenty six open problems in membrane computing. In *BWMC3*, pages 249–262. Fénix Editoria, 2005.
11. W. J. Savitch. Relationships between nondeterministic and deterministic tape complexities. *Journal of Computer and Systems Sciences*, 4(2):177–192, 1970.
12. P. Sosík. The computational power of cell division in P systems: Beating down parallel computers? *Natural Computing*, 2(3):287–298, Aug. 2003.
13. P. Sosík and A. Rodríguez-Patón. Membrane computing and complexity theory: A characterization of PSPACE. *J. Comput. System Sci.*, 73(1):137–152, 2007.
14. P. van Emde Boas. Machine models and simulations. In J. van Leeuwen, editor, *Handbook of TCS*, volume A, pages 1–66. Elsevier Science/MIT Press, 1990.