NETWORKS OF EVOLUTIONARY PROCESSORS
A Historical Account

Gemma Bel-Enguix, M. Dolores Jiménez-López and Carlos Martín-Vide
GRLMC, Rovira i Virgili University, Pl. Imperial Tàrraco 1, Tarragona, Spain

Keywords: Networks of Evolutionary Processors, Hybrid NEPs, Accepting and Generating NEPs, Applications, Implementation.

Abstract: This paper provides a historical account of Networks of Evolutionary Processors (NEPs), a bioinspired model of computation based in the behaviour of colonies of cells. NEPs, introduced in (Castellanos et al., 2001), consist of several processors performing molecular operations, which are placed in an underlying graph. In the last years, NEPs have demonstrated their generating and accepting power, as well as a good capacity for solving hard computational problems. Whereas complexity results support the accuracy of the framework, different variants have been introduced in order to achieve several computational properties. In the future, NEPs can also be used as a model for research in other fields.

1 INTRODUCTION

This paper has a historical and genetic purpose. In the following pages, we attempt to draw a generation tree for Networks of Evolutionary Processors, highlighting the main developments and variants of the model from the first papers up to now. Two surveys about NEPs have appeared so far in (Martín-Vide and Mitrana, 2003) and (Martín-Vide and Mitrana, 2005) with a systematic and uniform exposition of results, perspectives and challenges of this computational device. Definitions and proofs of the main achievements in the area can be found in such articles. However, the main goal of this paper is to give a non-formal explanation of the generation of these cell-inspired computational mechanisms (Section 2) and to stress the growing nodes of the tree, the state-of-the-art and chief branching in every one of the subtypes (Section 3 and 4).

Being a relatively active area in theoretical computer science, networks of evolutionary processors should be a consistent formal framework for some other fields of research. In Section 5 we explore some attempts in this direction.

We close this general and non-formal overview of the area portraying some challenges and suggestions for future development of the topic (Section 6). Finally, the references of this paper intend to be an almost complete collection of papers and publications on NEPs.

2 NETWORKS OF EVOLUTIONARY PROCESSORS: PRECEDENTS AND INFLUENCES

The first paper on Networks of Evolutionary Processors appeared in 2001, as the conjunction of three main streams:

a) parallel symbolic processing given by Connection Machines (Hillis, 1985) and logic flow paradigm (Errico and Jessope, 1994).
b) networks of parallel language processors (Csuhaj-Varjú and Salomaa, 1997), which are mechanisms inspired in the precedent devices, introducing an important modification: data are strings of language.
c) and natural computing motivation, based on the optimum results obtained by DNA computing (Adleman, 1994), and membrane and cell computing (Păun, 2000).

Connection Machines and logic flow paradigm, as efficient architectures for parallel and distributed symbolic processing, have played a key role in the emergence of the model of networks of evolutionary processors. These basic constructs present a mechanism where several processors are placed in a node of a complete graph, being each of them able to handle data. In (Csuhaj-Varjú and Salomaa,

Bel-Enguix G., Jiménez-López M. and Martín-Vide C. (2009).
NETWORKS OF EVOLUTIONARY PROCESSORS - A Historical Account.
In *Proceedings of the International Conference on Agents and Artificial Intelligence*, pages 626-631
DOI: 10.5220/0001809406260631

1997), the formalism was conveniently adapted for the generation of formal languages, starting from the idea that data the nodes handle can be strings of words. By this modification the devices described in (Hillis, 1985) and (Errico and Jessope, 1994), became language generating contructs (Csuhaj-Varjú and Mitrana, 2000; Csuhaj-Varjú and Salomaa, 2003), named networks of parallel language processors (Csuhaj-Varjú and Salomaa, 1997), closely related to Grammar Systems (Csuhaj-Varjú et al., 1994). In the formal language model, the processors could be any type of generation device, not excluding molecular based ones.

(Castellanos et al., 2001) finally introduced Networks of Evolutionary Processors as a natural derivation of the studies on natural and cell computing within symbolic networks paradigm. These formalisms were conceived as a type of networks of language generating systems with the particularity of having in each node a very simple evolutionary processor, able to perform only operations directly inspired in point mutations (DNA *in vivo* evolutionary processes): substitution, deletion and insertion of a base pair. The first definition of such devices was oriented to the resolution of NP problems, but later, in several papers, NEPs were developed as a powerful and efficient system of computation.

3 GENERATING NEPs AND HNEPs

The first contributions on NEPs theory (Castellanos et al., 2001; Castellanos et al., 2003) describe these generating constructions in a general way, allowing the processors to perform any of the molecular operations: *substitution*, *deletion*, *insertion*. But early in the literature, a variant is launched that introduces constraints in the processors: Hybrid NEPs (HNEPs) (Martín-Vide et al., 203). In HNEPs, only one type of operation is allowed in every node.

The differentiation between NEPs and HNEPs produces two large families in the general picture of NEPs genealogy. Although HNEPs are strictly a type of NEPs, these two variants are the starting point of the two main branches in the generation tree.

Several more NEPs variants have appeared from the first scission between NEPs/HNEPs. A number of subfamilies have been introduced in the last years: Pictorial NEPs (PNEPs) (Mitrana et al., 2003; Desanambika et al., 2004; Dersanambika et al., 2005), Massive Parallel NEPs (MPNEPs) (Gómez-Blas et al., 2008a), NEPs with filtered connections (NEPFCs) (de Mingo-López et al., 2005), Extended NEPs (ENEPs) (de Mingo et al., 2008), NEPs with nodes of two types NEPs (Dassow and Truthe, 2007a; Dassow and Truthe, 2007b; Alhazov et al., 2008) and NEPs with splicing rules (NSP) (Choudhary and Krithivasan, 2005a), which has other two variants depending on the types of contexts permitting/forbidding (Choudhary and Krithivasan, 2005c; Choudhary and Krithivasan, 2005b).

The subclasses of NEPs listed above modify the data inside the processors (PNEPs), the synchronization of the evolution and communication steps (MPNEPs), the filtering process (NEPFCs), the extension of the elements than can be communicated (ENEPs), the types of nodes (allowing only insertion-deletion, insertion-substitution, deletion-substitution) and the nature of rules (NSP).

Pictorial NEPs are variants of Networks of Evolutionary Processors that handle data which is not a string but a two dimensional construct.

Massive Parallel NEPs are devices in which evolution and communication are performed in parallel. These types of networks can solve NP-problems in linear time such as NEPs.

The main idea of NEPs with Filtered Connections is to move the filters from the processors to the connections, in order to decrease the complexity of the nodes of the graph. It has been demonstrated that NEPFC have the same power for solving problems than regular NEPs.

Extended NEPs aim to communicate not only the symbolic information placed inside the nodes, but also the evolution rules, in a way that the whole configuration of the system is evolving during the computation. This property provides a more realistic behaviour since operations are conceived as an exchangeable object as well.

NEPs with two types of nodes are formally restricted definitions of NEPs due to nodes with one of the three main operations are discarded. (Dassow and Truthe, 2007b) proved that nodes NEPs without insertion nodes produce only finite languages, NEPs without deletion nodes produce only context-sensitive languages and nodes without substitution nodes produce recursively enumerable languages.

NEPs with Splicing Rules are a variant of networks using splicing instead of point mutations. Splicing is also based in the behaviour of recombinant DNA, therefore the bio-inspired feature of NEPs remains. Two subtypes of NEPs consider the existence of permitting and forbidding contexts, simulating the switching on/off of the genes *in vivo* for the recombination to start.

In what refers to Generation HNEPs, a new subclass has been just introduced: OHNEPs (Alhazov et al., 2009), devices with a directed underlying graph, in which strings are discarded in every node where no operation can be performed.

4 ACCEPTING NEPs AND HNEPs

Both, NEPs and HNEPs were initially designed as generating devices. But after the introduction of HNEPs many papers started focusing in their Accepting power. However, after the first type of Accepting HNEPs (Margenstern et al., 2004) was introduced, many articles focused in the recognition power of HNEPs.

Some papers tackle the capacity of AHNEPs for solving NP-problems (Manea and Mitrana, 2007). In (Manea, 2004; Manea, 2007), AHNEPs are used to recognize CF languages. Moreover, a number of papers have been devoted to study the complexity and demonstrate the universality of such devices. Examples of this line of research can be found in (Manea et al., 2006b; Manea et al., 2007b).

One variant of AHNEPs has been introduced so far, Timed AHNEPs (Manea, 2005), that provides the AHNEP of a clock for regulating the number of steps and the stopping points of the system.

In what refers to NEPs, only several subclasses of the model have been applied to recognition, or have been created specially for it. The first attempts were focused in the use of Networks with splicing processors (Manea et al., 2005). This variant has been proved to have the same potential for solving NP-problems than regular NEPs (Manea et al., 2006a). On the other hand, complexity results have been studied for the model in (Manea et al., 2007a), and also in (Loos et al., 2008), where it is shown that accepting networks of splicing processors (ANSPs) of size 2 are computationally complete.

Besides, a version of Accepting NEPs with filtered networks has been also defined (Drăgoi et al., 2007), and approached with from the complexity issues (Drăgoi and Manea, 2008).

Finally, a version combining ANSP and ANEPFC has also been introduced, Accepting Networks of Splicing Processors with Filtered Connections (ANSPFC) (Castellanos et al., 2007), that attempts simplifying Accepting Networks of Splicing Processors by moving the filters from the nodes to the edges.

A new device, accepting networks of non-inserting evolutionary processors (ANNIEPs), has been formalized and studied in (Dassow and Mitrana, 2008) with several variants. These networks are somehow similar to the NEPs with nodes of two types, since insertion does not exist in the nodes.

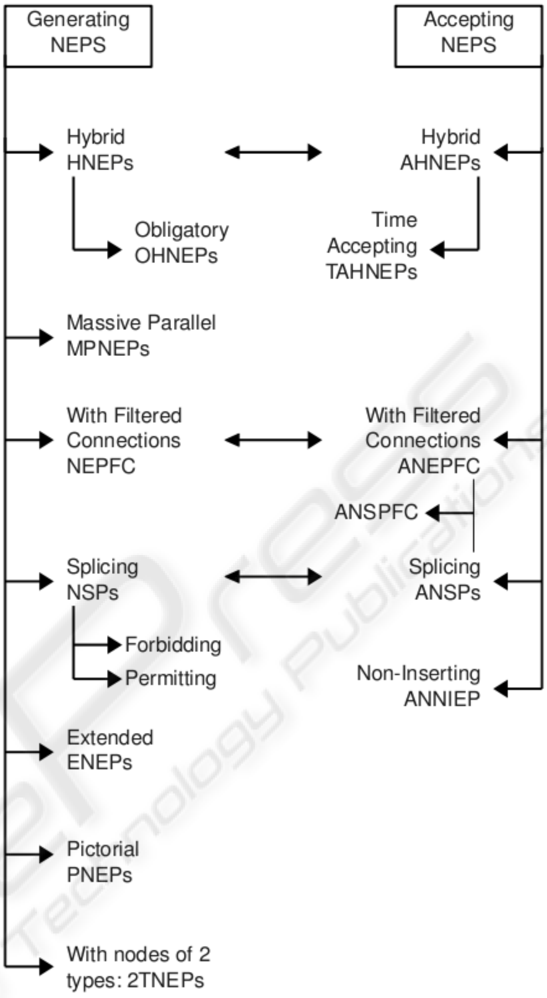

Figure 1: Generating and Accepting classes of NEPs.

5 APPLICATIONS AND IMPLEMENTATIONS

NEPs have been developed up to the point of being able to become a model for several other fields. Two different applications have been developed so far, one of them considering the suitability of NEPs to be used as Decision Systems (Gómez-Blas et al., 2008b; Gómez-Blas, 2008) and Symbolic information and rule-based behaviour make Networks of Evolutionary Processors an efficient tool to obtain decisions based on objects present in the network.

Another idea has been to apply the accepting power of HNEPs to the parsing of natural languages (Bel-Enguix and Jiménez-López, 2005). This is a challenging and promising topic, but its main issues, like the optimization and the complexity for every type of sentences, remain to be studied yet.

There is still a lack of computational implementations of the theoretical constructions. An implementation of Massive Parallel NEPs has been already presented (Díaz et al., 2007). There exist also at least two Java implementations (Díaz et al., 2008; Ortega, 2009), as well as several ongoing works dealing with the same objective: to achieve computational applications able to operate with data as the theoretical constructs of NEPs do.

6 FINAL REMARKS

From the introduction of the first models of NEPs, such devices have shown their efficiency and optimality in both, generation and recognition of languages, as well as their capacity to solve hard computational problems. Some important computational properties have been proved, and the issues of optimization and complexity have been tackled in a number of papers. Several variants have been introduced for improving the generating and accepting power of NEPs, performing modifications in different elements of the systems (nodes, underlying graph, rules, input data, filters), demonstrating again their flexibility.

The main open field of research for the future is to build models of computation based on NEPs that can work as a framework in other areas, such as game theory, linguistics or engineering. From the modular theories of mind to automatic learning or hardware implementation, a wide area of applications should be explored for optimizing and spreading this bio-inspired cooperative model of computing.

REFERENCES

Adleman, L. (1994). Molecular computation of solutions to combinatorial problems. *Science*, (226):1021–1024.

Alhazov, A., Bel-Enguix, G., and Rogozhin, Y. (2009). Obligatory hibrid networks of evolutionary processors. In *Proceedings of ICAART 2009*.

Alhazov, A., Dassow, J., Martín-Vide, C., Rogozhin, Y., and Truthe, B. (2008). On networks of evolutionary processors with nodes of two types. *Fundamenta Informaticae*, (to appear).

Bel-Enguix, G. and Jiménez-López, M. D. (2005). Analysing sentences with networks of evolutionary processors. In José Mira, J. R. Á., editor, *Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach*, volume 3562 of *Lecture Notes in Computer Science*, pages 102–111. First International Work-Conference on the Interplay Between Natural and Artificial Computation, IWINAC 2005, Las Palmas, Canary Islands, Spain, June 15-18, 2005, Proceedings, Part II.

Castellanos, J., Manea, F., de Mingo-López, L. F., and Mitrana, V. (2007). Accepting networks of splicing processors with filtered connections. In *MCU 2007*, pages 218–229.

Castellanos, J., Martín-Vide, C., Mitrana, V., and Sempere, J. M. (2001). Solving np-complete problems with networks of evolutionary processors. *Lecture Notes in Computer Science*, IWANN 2001(2048):621–628.

Castellanos, J., Martín-Vide, C., Mitrana, V., and Sempere, J. M. (2003). Networks of evolutionary processors. *Acta Informatica*, (39):517–529.

Choudhary, A. and Krithivasan, K. (2005a). Networks of evolutionary processors with splicing rules. In José Mira, J. R. Á., editor, *Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach*, volume 3561 of *Lecture Notes in Computer Science*, pages 290–299. First International Work-Conference on the Interplay Between Natural and Artificial Computation, IWINAC 2005, Las Palmas, Canary Islands, Spain, June 15-18, 2005, Proceedings, Part I.

Choudhary, A. and Krithivasan, K. (2005b). Networks of evolutionary processors with splicing rules and forbidding context. In José Mira, J. R. Á., editor, *Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach*, volume 3561 of *Lecture Notes in Computer Science*, pages 300–309. First International Work-Conference on the Interplay Between Natural and Artificial Computation, IWINAC 2005, Las Palmas, Canary Islands, Spain, June 15-18, 2005, Proceedings, Part I.

Choudhary, A. and Krithivasan, K. (2005c). Networks of evolutionary processors with splicing rules and permitting context. In *Proceedings of Fifth International Workshop on Information Processing in Cells and Tissues: IPCAT 2005*.

Csuhaj-Varjú, E., Dassow, J., Kelemen, J., and Păun, G. (1994). *Grammar Systems. A Grammatical Approach to Distribution and Cooperation*. Gordon and Breach, London.

Csuhaj-Varjú, E. and Mitrana, V. (2000). Evolutionary systems: a language generating device inspired by evolving communities of cells. *Acta Informatica 36*, pages 913–926.

Csuhaj-Varjú, E. and Salomaa, A. (1997). Networks of parallel language processors. *Lecture Notes on Computer Science 1218*, pages 299–318.

Csuhaj-Varjú, E. and Salomaa, A. (2003). Networks of watson-crick d0l systems. In *Proc. International Conference Words, Languages & Combinatorics*.

Dassow, J. and Mitrana, V. (2008). Accepting networks of non-inserting evolutionary processors. In *Proceedings of NCGT 2008: Workshop on Natural Computing and Graph Transformations*, pages 29–42.

Dassow, J. and Truthe, B. (2007a). On the power of networks of evolutionary processors. In *Machines, Computations and Universality, MCU 2007,Orléans,* *France, September 10?13, 2007, Proceedings*, pages 158–169.

Dassow, J. and Truthe, B. (2007b). On the power of networks of evolutionary processors. Technical Report 4/2007, Otto-von-Guericke-Universität Magdeburg, Fakultät für Informatik.

de Mingo, L. F., Gómez-Blas, N., Gisbert, F., and Peña, M. A. (2008). Extended networks of evolutionary processors. In Markov, K., Ivanova, K., and Mitov, I., editors, *Advanced Research in Artificial Intelligence*, number Number 2 in International Book Series INFORMATION SCIENCE & COMPUTING. Institute of Information Theories and Applications FOI ITHEA.

de Mingo-López, L. F., Menéndez, E. S., and Gisbert, F. (2005). Filtered networks of evolutionary processors. *International Journal Information Theories & Applications*, 12:5–11.

Dersanambika, K., Subramanian, K., and Mary, A. R. S. (2005). 2d and 3d pictural networks of evolutionary processors. In José Mira, J. R. Á., editor, *Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach*, volume 3562 of *Lecture Notes in Computer Science*, pages 92–101. First International Work-Conference on the Interplay Between Natural and Artificial Computation, IWINAC 2005, Las Palmas, Canary Islands, Spain, June 15-18, 2005, Proceedings, Part II.

Desanambika, K., Krithivasan, K., and Subramanian, K. (2004). Hexagonal pictorial network of evolutionary processors. In *Proceedings of the workshop on symbolic networks, ECAI 2004*.

Díaz, M. A., de Mingo, L. F., and Gómez-Blas, N. (2008). Networks of evulutionary processors: Java implementation of a threaded processor. *International Journal Information Theories & Applications*, 15:37–43.

Díaz, M. A., de Mingo, L. F., Gómez-Blas, N., and Castellanos, J. (2007). Implementation of massive parallel networks of evolutionary processors (mpnep): 3-colorability problem. In *NICSO 2007*.

Drăgoi, C. and Manea, F. (2008). On the descriptional complexity of accepting networks of evolutionary processors with filtered connections. *International Journal of Foundations of Computer Science*, 19(5):1113–1132.

Drăgoi, C., Manea, F., and Mitrana, V. (2007). Accepting networks of evolutionary processors with filtered connections. *J. UCS*, 13(11):1598–1614.

Errico, L. and Jessope, C. (1994). Towards a new architecture of symbolic processing. In Plander, I., editor, *ARtificial Intelligence and Information-Control Systems of Robots'94*, pages 31–40, Singapore. World Scientific.

Gómez-Blas, N. (2008). *Redes de Procesadores Evolutivos: Autoaprendizaje de Filtros en las Conexiones, PhD Thesis*. PhD thesis, Politechnical University of Madrid.

Gómez-Blas, N., de Mingo, L. F., and Santos, E. (2008a). Massive parallel networks of evolutionary processors as np-problem solvers. In *ICEIS 2008 Proceedings*, pages 588–592.

Gómez-Blas, N., Díaz, M. A., Castellanos, J., and Serradilla, F. (2008b). Networks of evolutionary processors (nep) as descision support systems. *International Journal Information Theories& Applications*, 15:31–36.

Hillis, W. (1985). *The Connection Machine*. MIT Press, Cambridge, MA.

Loos, R., Manea, F., and Mitrana, V. (2008). On small, reduced, and fast universal accepting networks of splicing processors. *Theoretical Computer Science*, doi:10.1016/j.tcs.2008.09.048.

Manea, F. (2004). Using ahneps in the recognition of context-free languages. In *Proceedings of the Workshop on Symbolic Networks, ECAI 2004*.

Manea, F. (2005). Timed accepting hybrid networks of evolutionary processors. In José Mira, J. R. Á., editor, *Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach*, volume 3562 of *Lecture Notes in Computer Science*, pages 122–132. First International Work-Conference on the Interplay Between Natural and Artificial Computation, IWINAC 2005, Las Palmas, Canary Islands, Spain, June 15-18, 2005, Proceedings, Part II.

Manea, F. (2007). On the recognition of context-free languages using accepting hybrid networks of evolutionary processors. *Int. J. Comput. Math.*, 84(3):273–285.

Manea, F., Martín-Vide, C., and Mitrana, V. (2005). Accepting networks of splicing processors. In *CiE 2005*, pages 300–309.

Manea, F., Martín-Vide, C., and Mitrana, V. (2006a). All np-problems can be solved in polynomial time by accepting networks of splicing processors of constant size. In *DNA 2006*, pages 47–57.

Manea, F., Martín-Vide, C., and Mitrana, V. (2006b). A universal accepting hybrid network of evolutionary processors. *Electronic Notes in Theoretical Computer Science*, 135(3):95–105.

Manea, F., Martín-Vide, C., and Mitrana, V. (2007a). Accepting networks of splicing processors: Complexity results. *Theoretical Computer Science*, 371(1-2):72–82.

Manea, F., Martín-Vide, C., and Mitrana, V. (2007b). On the size complexity of universal accepting hybrid networks of evolutionary processors. *Mathematical Structures in Computer Science*, 17(4):753–771.

Manea, F. and Mitrana, V. (2007). All np-problems can be solved in polynomial time by accepting hybrid networks of evolutionary processors of constant size. *Inf. Process. Lett.*, 103(3):112–118.

Margenstern, M., Mitrana, V., and Jiménez, M. P. (2004). Accepting hybrid networks of evolutionary processors. In *Pre-proceedings of DNA 10*, pages 107–117.

Martín-Vide, C. and Mitrana, V. (2003). Networks of evoutionary processors: a survey. *Theoria: Revista de teora, hitoria y fundamentos de la ciencia*, 18(46):59–70.

Martín-Vide, C. and Mitrana, V. (2005). Networks of evolutionary processors: results and perspectives. In Gheorghe, M., editor, *Molecular computational Models: Unconventional Approaches*, pages 78–114. Idea Group Publishing, Hershey.

Martín-Vide, C., Mitrana, V., Pérez-Jiménez, M., and Sancho-Caparrini, F. (203). Hybrid networks of evolutionary processors. In *Proceedings of GECCO 2003*, Lecture Notes in Computer Science 2723, pages 401–412, Berlin. Springer.

Mitrana, V., Subramanian, K., and Tataram, M. (2003). Pictural networks of evolutionary processors. *Romanian Journal of Information Science and Technology*, 6(1-2):189–199.

Ortega, A. (2009). On the solutions of np-complete problems by means of jnep run on computers. In *Proceedings ICAART 2009*.

Păun, G. (2000). Computing with membranes. *Journal of Computer and Systems Sciences*, 61(1):108–143.