A Chronology of Soft Systolic Development (Abstract)

Emily Degley and Brian J. d'Auriol
Department of Mathematics and Computer Science
The University of Akron
Akron, Ohio, 44325, USA

1 Introduction

We present a chronology of systolic array developments that emphasize pivotal research contributions. A classification scheme is proposed for the chronological order of this research. We also present a graphical notation for this classification scheme. We believe that the chronology described in this paper provides useful insight into the advancement of the systolic research area.

2 Classification Scheme

We provide a layered classification of a representative selection of important research contributions in the broad area of systolic arrays. The upper layer of this classification describes the advancement of research in three principal categories, namely Theoritcal, Soft-Systolic and Hardware. The middle layer describes significant and established research in each of these categories. The lowest layer describes specific contributions. In addition, our classification chronologically describes the approximate time-line of contributions classified in the middle and lower layer. We have observed that established research often exhibits periods of changing activity levels ranging from intensive to almost none.

In Figure 1, the three separate sections refer to the upper layer, box notation refers to the middle layer, and unboxed notation refers to the lowest layer. An approximate time-line shown by the thick black arrows which also separate the three upper layer categories. Periods of intensive research activity for the middle layer contributions are shown by the width of the box and is in relation to the time-line. Periods of low activity are indicated by horizontal arrows extending from the boxes. Lastly specific influential relationships between specific contributions are shown by vertical arrows.

3 Principal Systolic Research and Development Areas

In this section we briefly discuss on the various inter- and intra-contributory relationships existing between

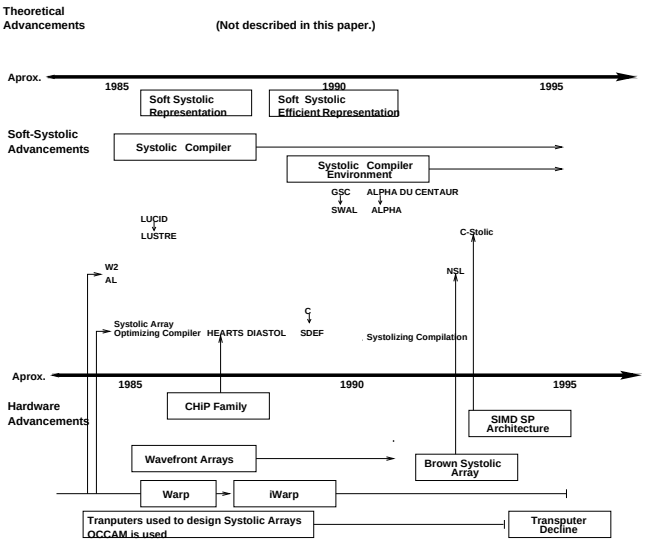

Figure 1: The classification of principal systolic research contributions.

specific research contributions. The classification scheme presented previously is used in this section for both organizational purposes and chronological placement. Specifically, only soft-systolic advancements and associated relationships are described.

Languages: The soft-systolic approach to systolic array implementation requires, at the most fundamental level, the ability to represent the systolic array in some language.

We have noted the following four language groups:

1. General purpose languages which are available for either one or more architectures. The earliest language used for systolic representation is occam. Initially occam was used to simulate array architectures. These occam simulations or programs could then be executed on single or multiple INMOS transputers [1]. Occam proved to be very successful for representing the requirements of systolic arrays.

2. Specific general languages influenced by particular hardware architectures. Among other early languages used were W2 and AL [2]. These languages were developed specifically for the warp systolic computer. These languages may be considered as general purpose since non-systolic algorithms can be represented in these languages. Another language in this category is New Systolic Language (NSL) introduced by Hughey.

NSL was developed for the Brown systolic array and reflects some of the characteristics of the Brown systolic array.

3. Specific but non-general languages influenced by particular hardware architectures. One example is FP [3]. Also C-Stolic and Lustre are further examples. All three allow systolic specification in a non-general way.

4. Languages which are associated with models that abstract architecture. Languages based on models of systolic computation started appearing by 1990. One example is ALPHA [4].

Representation: Soft systolic representation initially started as the straight-forward specification of systolic processing requirements in a language. Here we note a turning point in soft-systolic representation from merely the representation of systolic processing requirements to include efficient execution on the associated computer platform.

We note that Samwell in [5] refers to the fact that algorithms and architecture models have been implemented in occam on the transputer. Researchers seemed to become concerned regarding efficient implementations in this time frame. Megson found that the systolic approach is only useful if the basic operation can be upgraded to balance communication and computation costs. Megson acknowledges that *emulation* and *simulation* are important first steps in prototyping a design and checking timing/dataflows [6].

Compilers and Compiling Environments: Research into different compiling environments designed specifically for the systolic/wavefront processing is noted. Some examples include: Systolic Array Optimizing Compiler (based on the Warp systolic machine [2]), Systolizing compiler (compiles programs into a systolic form [7]), and SDEF [8]. We also note that DIASTOL, Hearts, and Generalized Systolic Computing (GSC) are further examples.

4 Conclusion

The chronology presented in this paper establishes approximate time periods of important research contributions. Also a classification scheme for the ordering of this chronology is presented. This classification scheme describes relationships between important contributions. Lastly we have developed a graphical notation for this classification scheme.

We note that two emerging areas of study, namely polytope model [9] and systolic optical interconnection networks [10] are based on systolic computing. We believe that an understanding of systolic processing would be useful in further study of these emerging areas.

References

[1] R. Chapman, et al.,"Design strategies for implementing systolic and wavefront arrays using occam," *ICASSP 85, IEEE ICASSP*, Tampa, FL., March, 26-29, 1985, V. 1, (ed.), pp. 292–295, March 1985.

[2] P. Tseng, *A Systolic Array Parallelizing Compiler*. 101 Philip Drive, Assinippi Park, Norwell, Mass. 02061, USA: Kluwer Academic Publishers, 1990.

[3] Y.-C. Lin, "Translating from fp to occam for systolic algorithms," *Microprocessors and Microsystems*, Vol. 15, pp. 435–444, Oct. 1991.

[4] H. le Verge and P. Quinton, "The palindrome systolic array revisited," *Research Directions in High-Level Parallel Programming Languages. Proc.*, Mont Saint-Michel, France, June, 17-19, 1991, Berlin, Germany: Springer-Verlag 1992, pp. 298–308, June 1991.

[5] P. Samwell, "Experience with occam for simulating systolic and wavefront arrays," *Software Engineering Journal*, Vol. 1, pp. 196–204, Sept. 1986.

[6] G. Megson, *An Introduction to Systolic Algorithm Design*. New York, USA: Oxford University Press, 1992.

[7] C. Lengauer, et al., "Towards systolizing compilation," *Distributed Computing*, Vol. 5, pp. 7–24, . 1991.

[8] B. R. Engstrom and P. R. Cappello, "The sdef programming system," *Journal of Parallel and Distributed Computing*, Vol. 7, pp. 201–231, 1989.

[9] C. Lengauer, "Loop parallelization in the polytope model," *CONCUR'93*, 1993, Lecture Notes in Computer Science 715, Springer-Verlag, pp. 398–416, 1993.

[10] S. M. Chai, et al., "Systolic processing architecture using optoelectronic interconnects," *Proc. of MPPOI*, Montreal, Canada, June, 22-24, 1997, pp. 160–166, June 1997.