Knowledge-Based Transformation Ordering

Mani B. Srivastava
AT&T Bell Laboratories
Murray Hill, NJ 07974

Miodrag Potkonjak
Computer Science Department
University of California
Los Angeles, CA 90095

ABSTRACT: We propose a two-step approach for transformation ordering which combines the use of optimization-intensive CAD techniques with knowledge-based user-driven search strategy. The first step is development of basic building blocks which target small sets of transformations which are well suited for optimization intensive CAD treatment. Next, transformation orderings are developed using knowledge about mathematical laws, an application domain, and the relationship among transformations. Transformation ordering scripts combine several building blocks to form effective approaches for optimization of several design metrics in many common computational structures. As the highlight of the approach, we developed a method which efficiently simultaneously optimizes throughput, latency, power, and area of linear computations.

1. MOTIVATION

Until now use of transformations in behavioral and algorithmic design has been subjected to complete dichotomy. On one side, transformations in compiler and high level synthesis [Cha92, Pot91] domains have been utilized through fully automatic approaches. The experience shows that the effectiveness of these approaches is limited due to lack of a proper way to encapsulate knowledge about mathematical structures and laws, enabling/disabling effects between transformations, and the key properties of the targeted application domain.

On the other side, the developers of VLSI designs [Par89], computer arithmetic units [Erc92], numerical and DSP algorithms have relied on comprehensive mathematical knowledge and sophisticated insights to manually apply transformations. The key limitation of this strategy for transformation ordering is related to human inability to solve numerous, complex, and computationally demanding combinatorial optimization tasks and bookkeeping details related to the application of transformations.

The main goal in this paper is to demonstrate the effectiveness of a new methodology for exploration of behavioral transformations. The key premise of the methodology is that both CAD optimization intensive tools for exploration of transformations and user knowledge and creativity are required for realizing the full potential of transformations.

The methodology is advocating the development or assembling of integrated systems of transformations which treats a set of transformations using CAD optimization intensive techniques. The systems of transformations are consequently used as basic blocks for developments of approaches where the user uses his knowledge and global view to combine blocks so that a variety of design goals on a variety of domains is achieved. In such a way to both CAD tools and to the application and design developer tasks for which they are best equipped are assigned. The tools handle cumbersome details and manually intractable optimization problems. The developer is left with task where he can demonstrate his greatest strengths: global view, mathematical and application specific knowledge, design intuition, and his, at least presumably, greatest strength - creativity.

Trying the same approach in VLSI DSP domain is certainly an attractive possibility. Unfortunately, it is apparent that mathematical and application domain (e.g. DSP, image processing) knowledge associated with the application of transformations is far larger and more complex than one required for playing chess. Therefore, the current AI and databases techniques are, and will continue to be, inadequate for encapsulating design, algorithmic, and mathematical knowledge. This, however, does not prevent coordinated use of approaches where computationally intensive task are covered by CAD tools, and global guidance, analytical derivations and general and domain specific knowledge is provided by the developer.

Transformations have been widely used in compilers, computer arithmetic, VLSI design, theoretical computer science, mathematics (linear algebra and numerical analysis), high level synthesis, DSP and control theory, symbolic manipulation algebra systems, algorithm design, several CAD domains (logic synthesis and physical lay-

0-7803-3192-3/96 $5.00©1996 IEEE

out) and many other scientific and engineering domains. The consensus from all those domains is that effectiveness of the transformations applications is mainly correlated to the ability of the user to develop a proper sequence in which transformations are applied.

In addition to completely manual approach which is highly effective but exceptionally complex and cumbersome, four automatic approaches are dominant. The first approach, peephole optimization [McK65], is based on use of simple heuristics which consider only very limited sections of code iteratively and fully locally. The second approach, "generate and test" [Jar84] is based on the exactly opposite paradigm: all possible sequences of transformations are applied globally on the whole program The third approach aims at manual derivation of scripts which enforces static ordering of transformations according to the experience achieved by extensive experimentation on small benchmark sets. The fourth approach has been recently gaining popularity. It is based on use of probabilistic algorithms as a search mechanism [Cha92, Pot91a]

The new approach aims at closing the gap between the run time and the quality of results, by combining the power of computationally intensive optimization for derivation of proper subsequences of transformations, and the user ability to conduct global search process.

2. KNOWLEDGE-BASED TRANSFORMATION ORDERING: NEW PARADIGM

The first component of the approach are the sets of transformations which optimize particular goals and constraints. While some of those sets are limited to a particular application domain, others are general. The sets of transformations are selected in a such way that they are very effective for the selected goals and constraints on some important and often used application domain. The application of transformation from the set is often done in computationally intensive way.

Interestingly several such basic building blocks components are already readily available. For example, methods such as block processing [Par89] and scatter look ahead [Par89] maximally fast implementation of linear programs [Pot91b], and heuristic and optimum techniques for simultaneous of latency and throughput [Sri94] are examples of such transformational scripts.

The knowledge-based user-driven transformation ordering is supported by the following high level synthesis software and knowledge infrastructure. (Interestingly all infrastructure required by the new approach is already available from a number of high level and system level synthesis systems):

(1) *Estimation tools.* Several high level synthesis systems contain tools which display a sharp lower bounds on best achievable implementation under design and timing constraints. For example, Hyper has a suit of relaxation-based estimation tools [Rab91] and provides information about intangible properties (e.g. regularity, temporal and spatial locality) which often have dominant role in influencing the quality of the final implementation [Gue94].

(2) *Guidance Rules for transformation ordering.* Information about enabling/disabling relationship among transformations in some cases helps in reducing the search space. A transformation is enabling, if its application does not improve objective function, but makes feasible application of another transformation which significantly improve the quality of design. Enabling and disabling effects among transformations is summarized in [Whi90, Pot91b]

(3) *Asymptotic analysis information.* Often it is of interest to analyze the effectiveness of transformations as the size of target problem instance increases. This analysis is not of just theoretical interest because there is a simple way to increase the size of any ASIC computation on infinite stream of data by unfolding the computation along the time loop.

The key advantages of the knowledge-based approach for transformation ordering include ability to utilize mathematical knowledge and to explore domain specific knowledge, to use knowledge about the relationship between transformations and conduct asymptotic analysis, to use of intangible computation properties. [Gue94] and design and architecture intuition.

3. LOW POWER OPTIMIZATION USING KNOWLEDGE- BASED TRANSFORMATIONS ORDERING

We now briefly describe how the knowledge-based approach for transformation ordering combines the method proposed in [Sri94] and the MCM approach to significantly reduce power consumptions in linear designs while satisfying strict throughput and latency constraints. The new transformation ordering for simultaneous optimization of latency, throughput, power, and area can be introduced using the following pseudocode:

Transformation ordering for simultaneous optimization of latency, throughput, power, and area

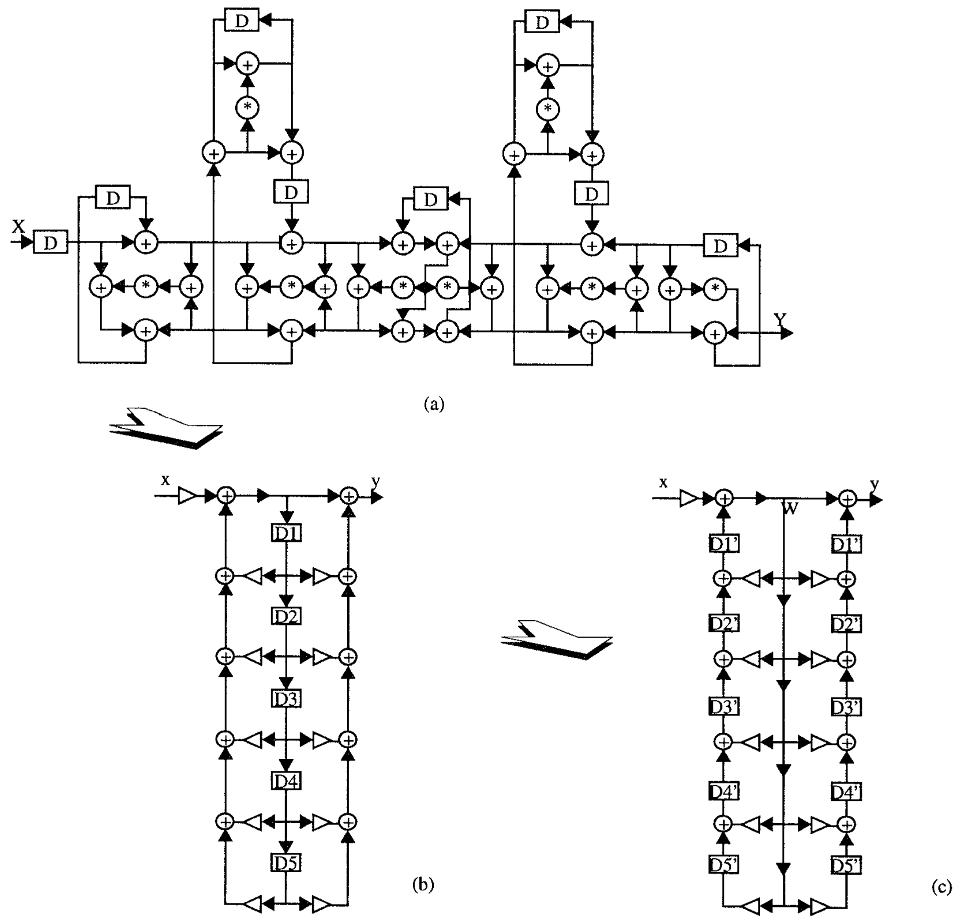

Figure 1. Illustration of the application of transformation ordering for simultaneous optimization of latency, throughput, power, and area: 5th order elliptical wave digital filter: (a) the initial structure, (b) key intermediate structure; (c) structure amenable for application of the MCM transformation.

Design	Initial area	Area After Optimization	Improvement Factor (Area)	Initial Energy	Energy After Optimization	Improvement Factor (Power)
mat	14.07	2.29	6.14	59.2	8.35	7.09
ellip	52.58	6.71	7.84	222	21.9	10.1
lin4	67.59	8.92	7.58	288	14.1	20.4
lin5	223.31	12.92	17.3	756	21.1	35.8
iir5 (wdf5)	28.92	8.23	3.51	118	18.2	6.48
iir6	10.90	4.41	2.47	40.2	12.2	3.30
iir8	29.25	5.09	5.75	119	9.87	12.06
iir10	22.69	8.99	2.52	96.7	19.0	5.09
iir11	20.58	7.18	2.87	81.3	14.6	5.57
iir12	25.73	9.95	2.59	89.2	14.8	6.03
steam	82.34	7.88	10.45	377	21.5	17.5

Table 1: Improvements in area over the initial design using the new transformation ordering. Area is measured in mm^2, power in nJ/sample, and latency and throughput in the number of cycles

(1) *Apply set of transformations for the optimum (heuristic) optimization of latency and throughput;*

(2) *Apply the set of transformations for maximally fast implementation*

(3) *Apply the MCM-for-power-minimization;*

The idea behind the new script is simple. The first step produces the high throughput and low latency design. The second step rearranges the computation into the form which is suitable for the application of the MCM transformation, while preserving the achieved latency and throughput. The final step minimizes the power by exploring potential for common subexpression elimination exposed by the first two steps.

A good intuition behind the new transformational script can be obtained by observing the first two steps of the transformation order in graphical form (Fig. 1a-c). Triangles denote multiplication of a variable by a constant. The Fig. 1a shows the popular high level synthesis benchmark 5th order elliptical wave digital filter. The Fig 1b shows the key intermediate step during this process. The computational structure shown in Fig 1c is exceptionally well suited for the application of the MCM transformation. For example, many intermediate variables and the output are formed by multiplying the variable w with various constants.

The transformation order presented in this section illustrates well both power and simplicity of the new approach for transformation ordering. All what was necessary to derive powerful ordering of transformations, was to connect three already available compound transformations from literature. Note that all of those transformations require involved optimization procedures. For example, transformations for the optimum (heuristic) optimization of latency and throughput [Sri94, Pot94a] uses both symbolic algebraic manipulation techniques and classical control-data flow transformations. The MCM transformation is, as we already stated, effective only when comprehensive combinatorial optimization is conducted [Pot94b].

4. EXPERIMENTAL RESULTS

Table 1 contains results of the power reduction when throughput and latency are kept constants The average and median improvements are by factors 11.8 and 7.08 respectively. Table 1 also contains results of the area reduction when throughput and latency are kept constants. The average and median improvements are by factors 6.27 and 5.75 respectively.

5. CONCLUSION

We introduced a new two-component approach for transformation ordering. The first component is development of small, well characterized scripts of transformations which utilize computation-intensive optimization CAD techniques. The second component is knowledge-based user-driven search. The knowledge-based methodology for transformation ordering is used to develop several new orderings for behavioral transformations. The utility of the proposed techniques is verified on numerous examples.

6. REFERENCES

[Bla85] R. E. Blahut: *Fast Algorithms for Digital Signal Processing*, Addison-Wesley Publishing Company, 1985.

[Cha92] A.P. Chandrakasan, M. Potkonjak, J. Rabaey, R. Brodersen: "An Approach for Power Minimization Using Transformations", 1992 IEEE Workshop on VLSI Signal Processing, CA, pp. 500-509, October 1992.

[Erc92] M.D. Ercegovac, T. Lang: "Fast Arithmetic for Recursive Computations", VLSI Signal Processing V, pp. 14-28, IEEE Press, New York, NY, 1992.

[Gue94] L. Guerra, M. Potkonjak, J. Rabaey: "System-Level Design Guidance Using Algorithm Properties", VLSI Signal Processing Workshop, pp.

[Jar84] M. Jarke, J. Koch: "Query optimization in databases systems", ACM Computing Surveys, Vol. 167, No. 2, pp. 111-152, 1984.

[McK65] W.M. McKeeman: "Peephole optimization", Communications of the ACM, Vol. 8, No. 7, pp. 443-444, 1965.

[Par89] K. K. Parhi, D. Messerschmitt: "Pipeline interleaving and parallelism in recursive Filters, Parts 1 and 2", IEEE Trans. on ASSP, Vol. 37, pp. 1099-1117 and 1117-1134, 1989.

[Pot91] M. Potkonjak, J. Rabaey: "Optimizing Resource Utilization Using Transformations", IEEE International Conference on Computer-Aided Design, pp. 88-91, 1991.

[Pot91b] M. Potkonjak, J. Rabaey: "Maximally Fast and Arbitrarily Fast Implementation of Linear Computations", Technical Report, NEC USA, Princeton, NJ, 1991.

[Pot94] M. Potkonjak, M.B. Srivastava, A. Chandrakasan: "Efficient Substitution of Multiple Constant Multiplications by Shifts and Additions using Iterative Pairwise Matching", 31st DAC, pp. 189-194, 1994.

[Rab91] J. Rabaey, et al.: "Fast Prototyping of Data Path Intensive Architecture", IEEE Design and Test, pp. 40-51, 1991.

[Sri94] M. B. Srivastava, M. Potkonjak: "Transforming Linear Systems for Joint Latency and Throughput Optimization", EDAC-94, pp. 267-271, 1994.

[Whi90] D. Whitfield, M. L. Soffa: "An Approach to Ordering Optimizing Transformations", Second ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, pp. 137-147, 1990.