Computer Science > Discrete Mathematics

Optimal Folding of Data Flow Graphs based on Finite Projective Geometry using Lattice Embedding

Swadesh Choudhary, Hrishikesh Sharma, Sachin Patkar (Department of Electrical Engg., Indian Institute of Technology, Bombay)

(Submitted on 1 Jul 2011)

A number of computations exist, especially in area of error-control coding and matrix computations, whose underlying data flow graphs are based on finite projective-geometry(PG) based balanced bipartite graphs. Many of these applications are actively being researched upon. Almost all these applications need bipartite graphs of the order of tens of thousands in practice, whose nodes represent parallel computations. To reduce its implementation cost, reducing amount of system/hardware resources during design is an important engineering objective. In this context, we present a scheme to reduce resource utilization when performing computations derived from PG-based graphs. In a fully parallel design based on PG concepts, the number of processing units is equal to the number of vertices, each performing an atomic computation. To reduce the number of processing units used for implementation, we present an easy way of partitioning the vertex set. Each block of partition is then assigned to a processing unit. A processing unit performs the computations corresponding to the vertices in the block assigned to it in a sequential fashion, thus creating the effect of folding the overall computation. These blocks have certain symmetric properties that enable us to develop a conflict-free schedule. The scheme achieves the best possible throughput, in lack of any overhead of shuffling data across memories while scheduling another computation on the same processing unit. This paper reports two folding schemes, which are based on same lattice embedding approach, based on partitioning. We first provide a scheme for a projective space of dimension five, and the corresponding schedules. Both the folding schemes that we present have been verified by both simulation and hardware prototyping for different applications. We later generalize this scheme to arbitrary projective spaces.

Comments:	31 pages, to be submitted to some discrete mathematics journal
Subjects:	**Discrete Mathematics (cs.DM)**; Combinatorics (math.CO)
MSC classes:	51E20
Cite as:	**arXiv:1107.0118 [cs.DM]**
	(or **arXiv:1107.0118v1 [cs.DM]** for this version)

Submission history

From: Hrishikesh Sharma [view email]
[v1] Fri, 1 Jul 2011 06:50:46 GMT (69kb)