Cannon's algorithm

From Wikipedia, the free encyclopedia

In computer science, **Cannon's algorithm** is a distributed algorithm for matrix multiplication for two-dimensional meshes first described in 1969 by Lynn Elliot Cannon.[1][2]

It is especially suitable for computers laid out in an $N \times N$ mesh.[3] While Cannon's algorithm works well in homogeneous 2D grids, extending it to heterogeneous 2D grids has been shown to be difficult.[4]

The main advantage of the algorithm is that its storage requirements remain constant and are independent of the number of processors.[2]

The Scalable Universal Matrix Multiplication Algorithm (SUMMA)[5] is a more practical algorithm that requires less workspace and overcomes the need for a square 2D grid. It is used by the ScaLAPACK, PLAPACK, and Elemental 🔗 libraries.

See also   [edit]

- Systolic array

References   [edit]

1. ^ Lynn Elliot Cannon, *A cellular computer to implement the Kalman Filter Algorithm* 🔗, Technical report, Ph.D. Thesis, Montana State University, 14 July 1969.
2. ^ *a b* Gupta, H.; Sadayappan, P.: Communication Efficient Matrix-Multiplication on Hypercubes 🔗, dbpubs.stanford.edu
3. ^ 4.2 Matrix Multiplication on a Distributed Memory Machine 🔗, www.phy.ornl.gov
4. ^ Ph.D. Research 🔗, graal.ens-lyon.fr. The thesis itself is not available from the archived link.
5. ^ Robert A. van de Geijn and Jerrell Watts, SUMMA: scalable universal matrix multiplication algorithm 🔗, Concurrency: Practice and Experience. Volume 9, Issue 4, pages 255–274, April 1997.

External links   [edit]

- Lecture at Berkeley 🔗
- mu.oz.au 🔗

v · t · e	Numerical linear algebra			[hide]
Key concepts	Floating point · Numerical stability			
Problems	Matrix multiplication (algorithms) · Matrix decompositions · Linear equations · Sparse problems			
Hardware	CPU cache · TLB · Cache-oblivious algorithm · SIMD · Multiprocessing			
Software	BLAS · Specialized libraries · General purpose software			

➕➗ *This applied mathematics-related article is a stub. You can help Wikipedia by expanding it.*

Categories: Distributed algorithms | Matrix multiplication algorithms | Applied mathematics stubs