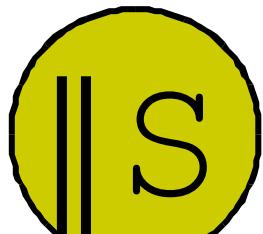


Matrix Multiplication using Parallel Solution Technology

8 January 2025

By Parallel Solutions

yehuda.singer@llp.co.il
+972-52-2306-311

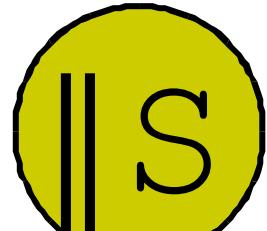


Problem Definition

- Multiplication of tow large matrices A and B to produce matrix C .
- Every matrix has a minimum size of 1024X1024 elements.
- We shall run it serial and parallel and compare performance.
- The advantage is at least by a factor of number of processors np .

Definitions

- \mathbf{A} is an $m \times n$ matrix
- \mathbf{B} is an $n \times p$ matrix
- The result matrix \mathbf{C} is $m \times p$ matrix.



Cache Memory Optimization

- Every element in matrix C ,

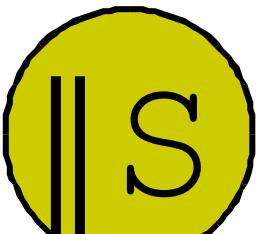
$$c_{ij} = \sum_{k=0}^n a_{ik} * b_{kj}$$

$$i = 1, \dots, m \text{ and } j = 1, \dots, p.$$

For each iteration we keep the row i of the matrix A in the cache memory.

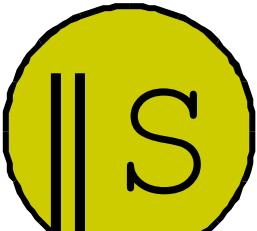
Therefore, only one access to a row implies reading the main memory.

For example: In each iteration i , we compute $C_{i1}, C_{i2}, C_{i3} \dots$



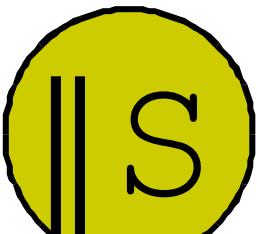
Experiment

- To ease debugging, we use identity matrices of different sizes.
- Every size is of size $s=2^l$ where $l = 1..12$.
- Every element is a *long* type of 8 bytes size.



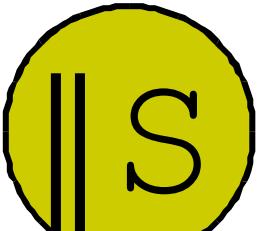
Processor's Performance Data

- For long data types:
 1. Multiplication of two long variables takes 1 clock.
 2. Addition of two long variables takes 1 clock.
- Memory Access
 1. Cache memory access is at the order of 2 clocks cycles.
 2. Main memory access is at the order of 20 clocks cycles.



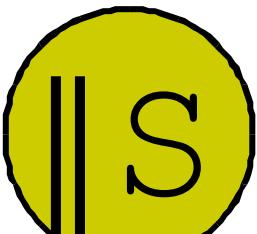
Software & Hardware Configuration

- Computer: Intel *i9*, 16 cores 80x86
- Memory: 32GB
- SUSE Linux operating system.
- Eclipse as an IDE.
- GNU tools.
- C++ Version 17.



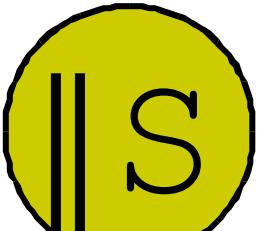
Holding matrices in files

- Every matrix is stored in a set of binary files.
- The size of a binary file is limited to 64K bytes.
- For matrices greater than 256x256 of long type elements and above, we split the files into quantities of 64k bytes.



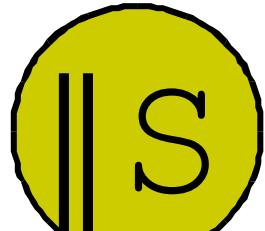
Memory

- All the matrices are stored in ONE continuous memory.
- The threads access the matrices by pointers which are *read only*.



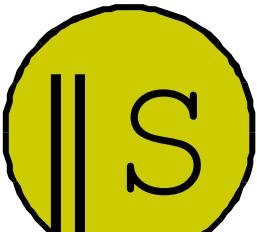
Parallel Computation

- We have 16 threads working in parallel.
- Each thread computes one element c_{ij}
- The execution time of each thread is dependent on the matrices' size.



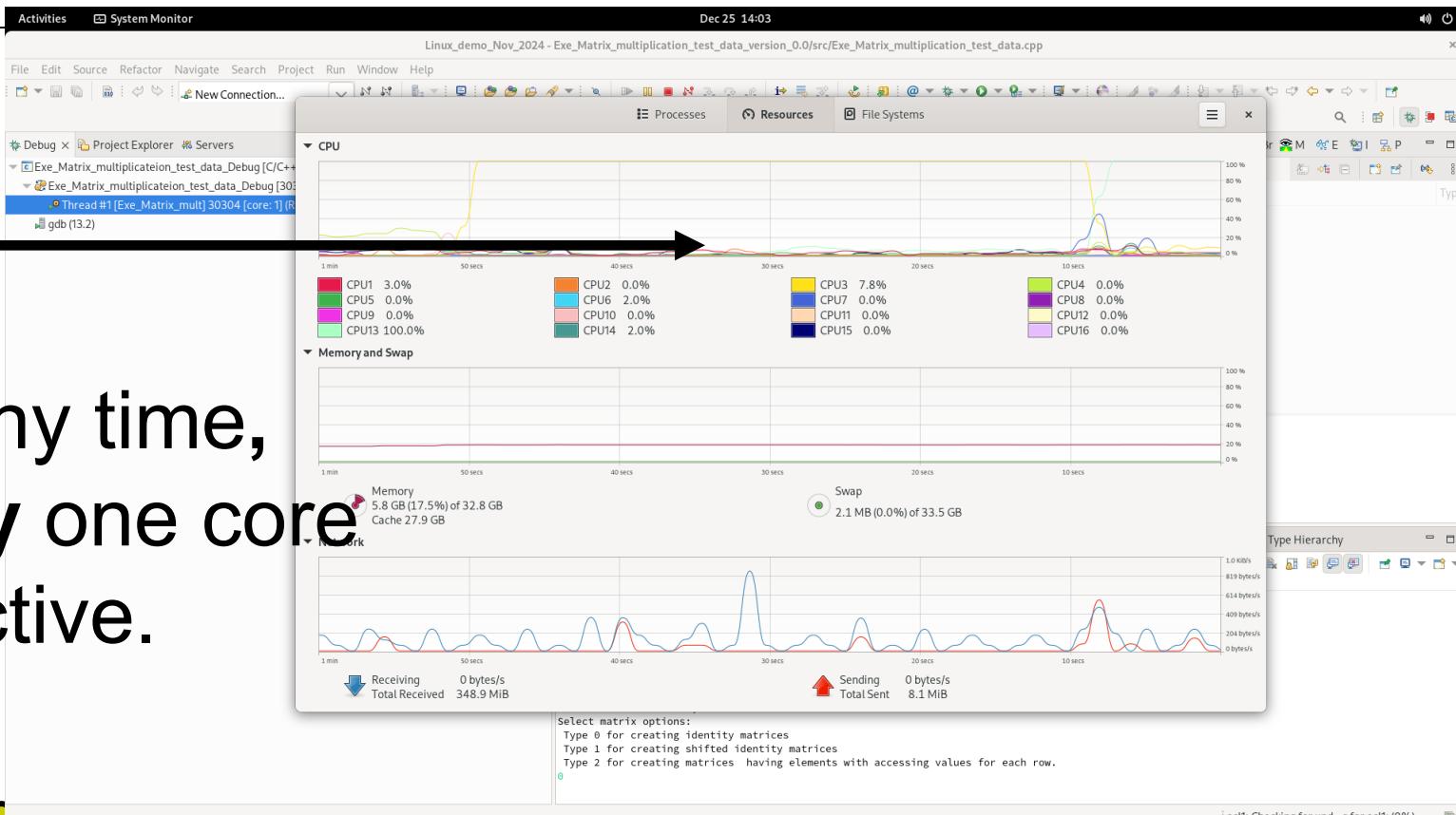
OS Limitations

- We work with physical memory - no SWAPPING.
- Number of threads is limited by the memory size of the computer system.



Serial Running on a 16 cores Processor

At any time,
Only one core
is active.



Linux System Monitoring Utility

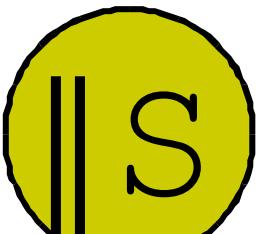
8 January 2025

Proprietary Information Parallel Solutions

Results Initial Computation method

Measured by the internal clock of the Linux system.
Accuracy 1ns.

	Matrix size	Serial Time	Parallel version time	Acceleration Ratio
	1024X1024	2.03451	1.038041	1.9599
	2048X2048	65.056	6.1396	8.9
	4096X4096	589.051	37.076	15.88

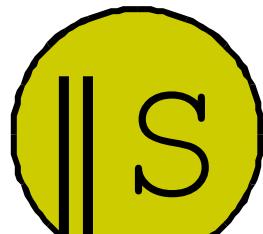


Results of Optimization the threads of previous calculation

Measured by the internal clock of the Linux system.

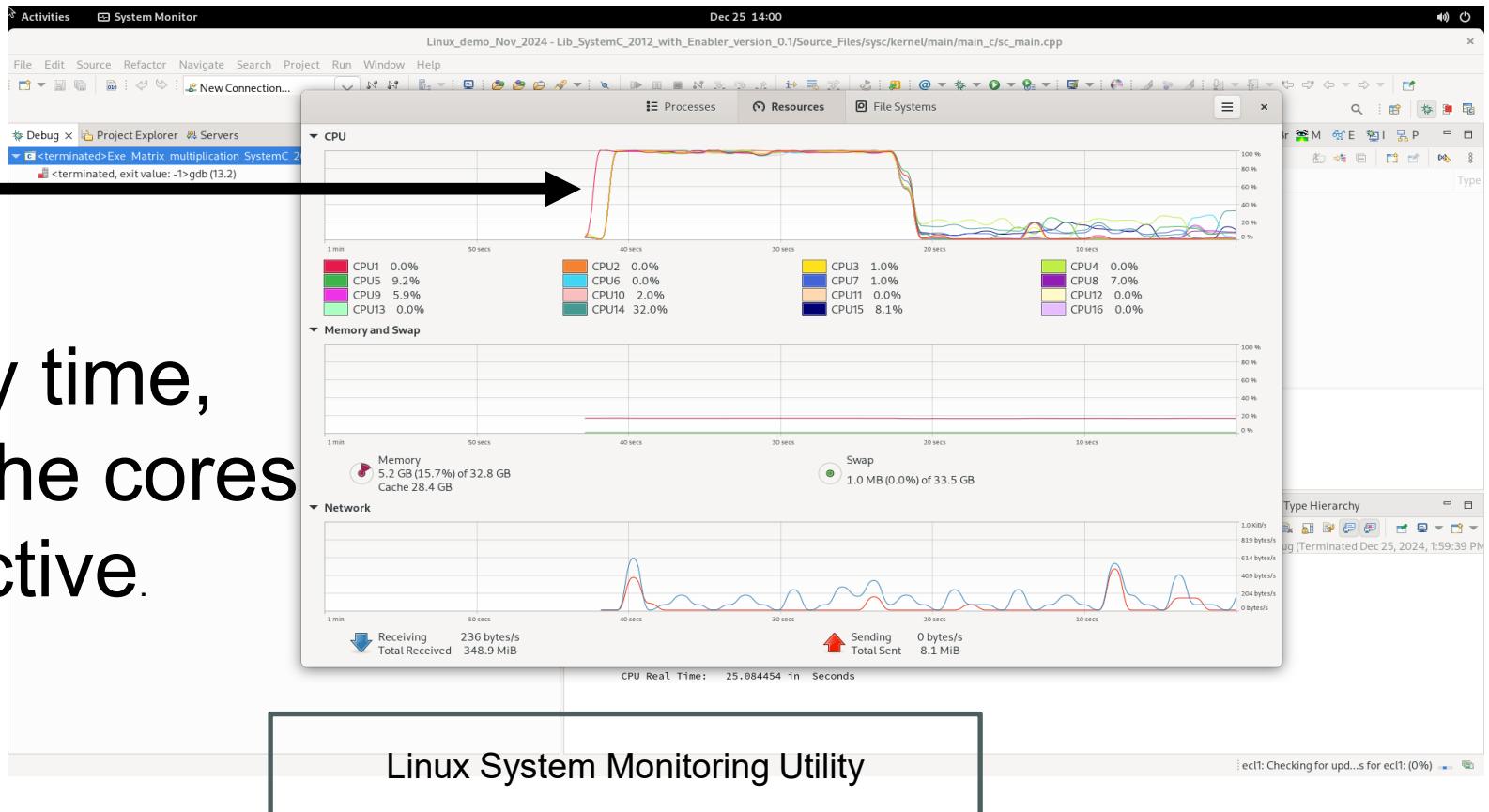
Accuracy 1ns.

Matrix size	Serial Time	Optimized Parallel time	Acceleration Ratio
1024X1024	2.03451	0.044720	45.4944
2048X2048	65.056	3.011256	21.159
4096X4096	589.051	25.0819	23.485



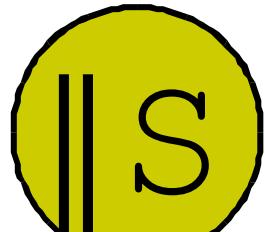
Parallel Running on a 16 cores processor

At any time,
ALL the cores
are active.



Summary

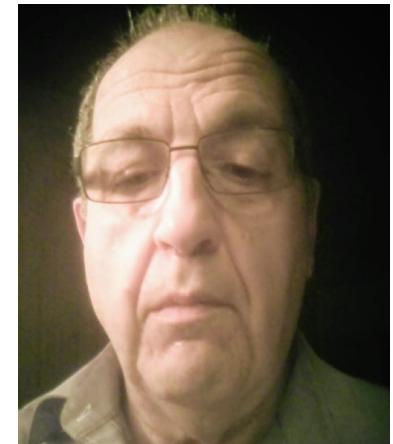
- Our technology proved the acceleration ratio of matrix multiplication this application give results far better than the number of cores.
- We have optimized automatically the cache memory usage.
- We got far better acceleration ratio than the number of cores.



Parallel Solutions: Leading Team

- **Dr. Yehuda Singer**

Ph.D. in computer science, specialized in computer architecture, ***Real-Time*** and high performance. Over 42 years of experience in the development of embedded systems and FPGAs in various multi-disciplinary applications. Dr. Singer was the leader of the Computer Studies of the extension of Derby university in Israel. Dr. Singer holds an M.Sc. and Ph.D. degrees in computer science from Weizmann Institute and Bar-Ilan university respectively.



- **Dr. Joshua Gur**

Joshua - a physicist with vast experience in multidisciplinary system design, Dr. JG has been the chief Display & Video Engineer for the past 30 years at one security plant of the Israeli Aircraft Industries. He has over 45 years of extensive experience in video systems, electro-optics and multidisciplinary system design and holds 9 patents. Dr. Gur holds a B.Sc. in Physics and Mathematics, M. Sc in Electro-optics from the Hebrew University of Jerusalem, and a Ph.D. in optics from Rochester University of NY USA.

