Developing a Distributed System for Algebraic Geometry

Wolfgang Schreiner
Research Institute for Symbolic Computation (RISC-Linz)
Johannes Kepler University, A-4040 Linz, Austria

Wolfgang.Schreiner@risc.uni-linz.ac.at
http://www.risc.uni-linz.ac.at/people/schreine

Algebraic Geometry

Distributed Maple

Parallel Algebraic Geometry

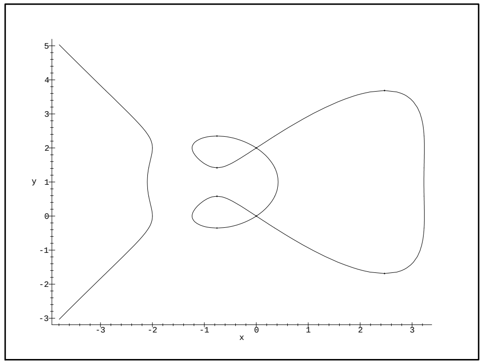
Conclusions

Algebraic Geometry

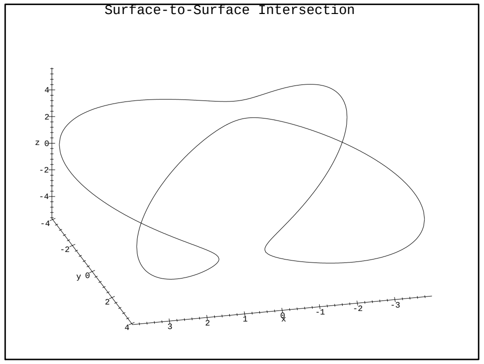
- Problem Area
  - Computing with and reasoning about geometric objects.
  - Representation by *algebraic sets*.
  - Manipulation by computer algebra methods.
- Software: CASA
  - *Computer Algebra Software for Constructive Algebraic Geometry*.
  - Implementation in the computer algebra system *Maple*.
  - Franz Winkler et al., 1990–
- Application: Scientific Visualization.
  - Plane curves.
  - Surface-to-surface intersections.

Scientific Visualization

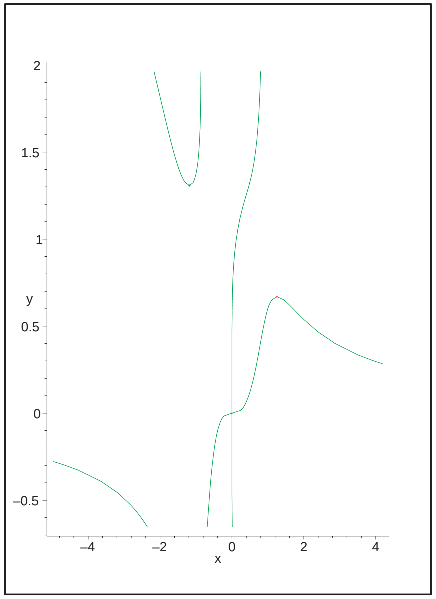
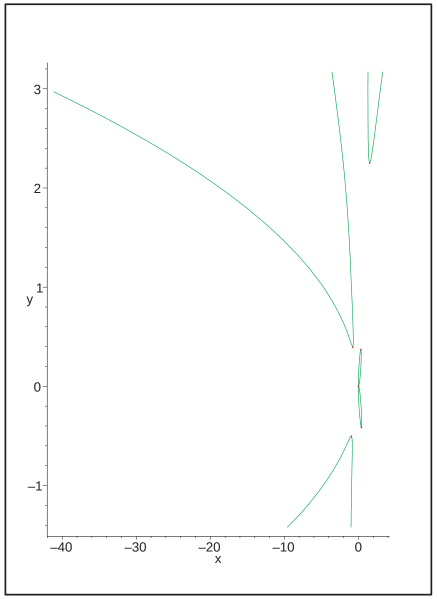
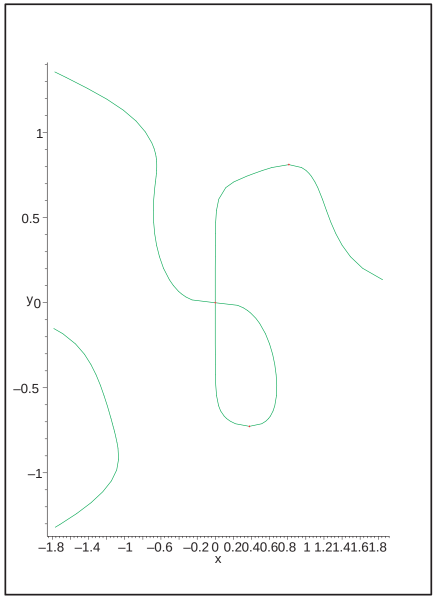
$$y^4 - 96a^2y^2 + 100a^2x^2 - x^4$$



$$y^4 - 384y^2 + 400x^2 - x^4 + z^3,$$
$$x^2 + y^2 + z^2 - 30$$



High Degree Curves

Scientific Visualization

- Numerical methods: fast but unreliable.
  - Qualitatively wrong results possible.
  - E.g: missing or non-existing *critical points*.
- Symbolic methods: reliable but slow.
  - Exponential in degree of polynomial.
  - Up to degree 5 or so.
- Hybrid symbolic-numerical methods: reliable and faster.
  - Still exponential complexity.
  - Up to degree 10 or so.

Parallel processing to investigate high-degree curves/surfaces.

Pragmatics

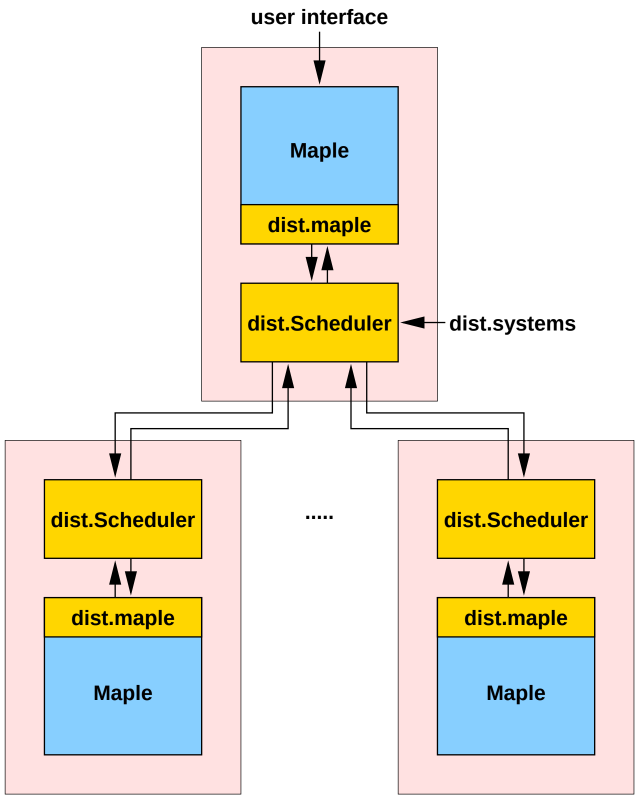
- Large existing code base in Maple.
  - Porting to other system unrealistic.
- Maple is *sequential*.
  - No multi-threading.
  - No message passing.
- Maple is *closed*.
  - No access to internal kernel structures.
- Only external interface: input/output.
  - Linearization of internal data structures.
  - Reading/writing from/to files or pipes.

Constraints for integration into a parallel environment.

Distributed Maple

- Coordination Layer.
  - Distributed interaction and task scheduling.
  - Java program *dist.Scheduler*.
- Interface Layer.
  - Connecting Maple kernel to scheduler.
  - Maple package *dist.maple*.
- Computing Layer.
  - Maple kernels as computational engines.

No modification of Maple!



Example

```
    |\^/|     Maple V Release 4 (Johannes Kepler University Linz)
._|\|   |/|_. Copyright (c) 1981-1996 by Waterloo Maple Inc. All rights
 \  MAPLE  /  reserved. Maple and Maple V are registered trademarks of
 <____ ____>  Waterloo Maple Inc.
      |       Type ? for help.
> read `dist.maple`;
Distributed Maple V1.0 (c) 1998 Wolfgang Schreiner (RISC-Linz)
See http://www.risc.uni-linz.ac.at/software/distmaple
> dist[initialize]([[deneb,solaris], [iris,irix]]);
connecting deneb...
connecting iris...
                                     okay
> t1 := dist[start](int, x^n, x):
> t2 := dist[start](int, x^n, n):
> dist[wait](t1) + dist[wait](t2);
                                (n + 1)     n
                               x           x
                               -------- + -----
                                n + 1      ln(x)
```

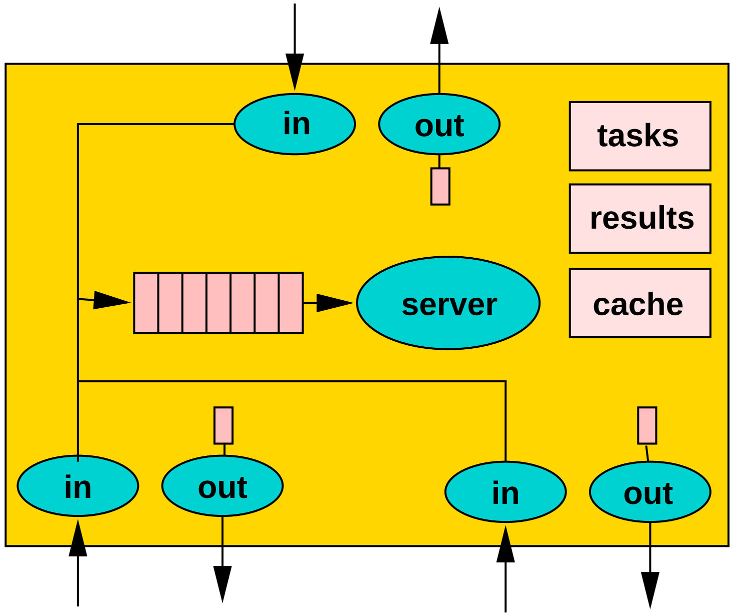
Programming Model

- Functional Parallelism
  - Task Creation: t `= dist[start](`f`, ...)`.
  - Synchronization: r `= dist[wait](`t`)`.
- Non-Determinism
  - Selection: `(`i`, `r`) = dist[select](`$tlist$`)`.
  - Abortion: `dist[delete](`t`)`.
- Shared Data Objects
  - Create empty object: d `= dist[data]()`.
  - Write data: `dist[put](`d`, `v`)`.
  - Wait for data: v `= dist[get](`d`)`.

Derived from functional/logic/dataflow models.

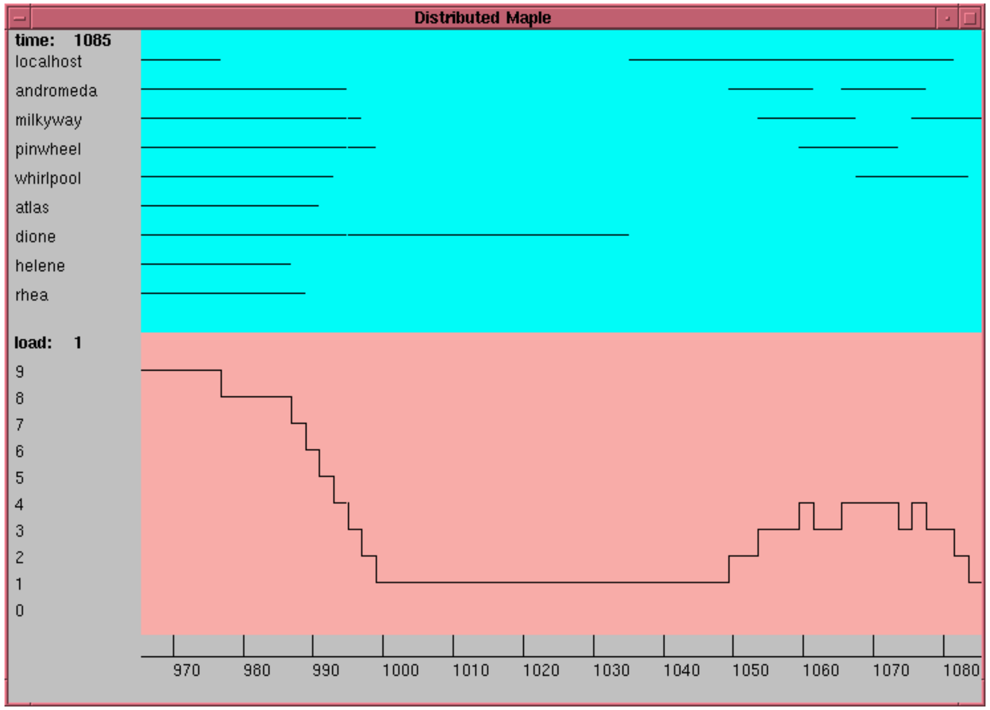
Scheduler

- Task Management.
  - Tasks to be scheduled for execution.
  - Task results and result caches.
- Distributed Operation.
  - Remote system startup.
  - Multi-threaded service handling.
  - Watchdog threads monitor connectivity.
- Visualization
  - Online visualization.
  - Generation of trace files.



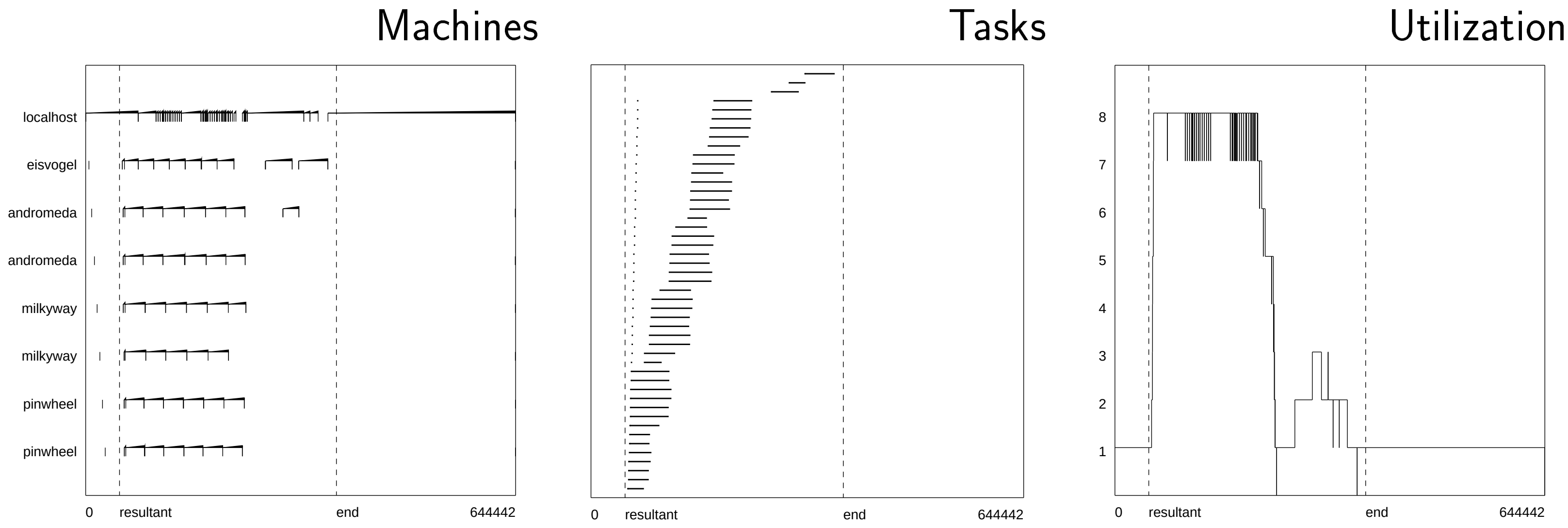
Can be combined with any computational engine.

Online Visualization



Current system status.

Post-Execution Visualization



Analysis of trace files.

Parallel Algebraic Geometry

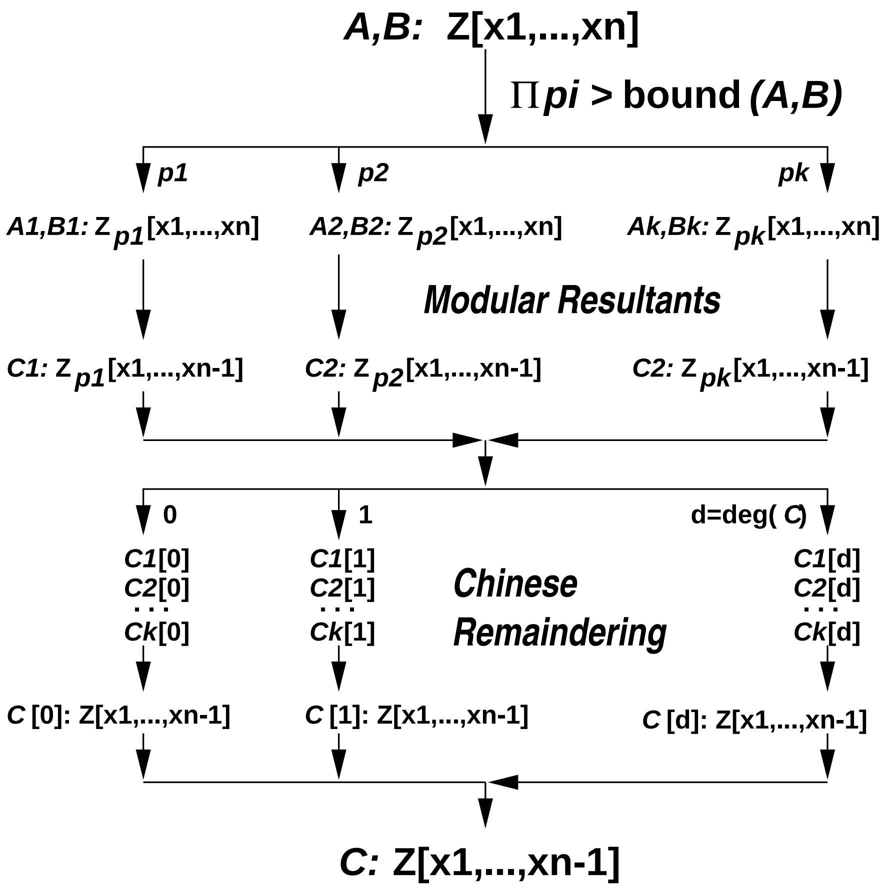
- Visualization of Plane Curves.
  - CASA `pacPlot`.
  - Symbolic phase ≈ 95–99%
  - Numerical phase ≈ 1–5%
- Parallelization of Symbolic Phase.
  1. Resultant computations ≈ 20–80%
     - Parallel algorithm.
  2. Real root isolations ≈ 20–80%
     - Parallel algorithm.
     - Multiple real roots in parallel.

Multiple parallelization layers.

Parallel Resultant Computation

- Polynomial Resultants
  - Given: $A, B \in \mathbb{Z}[x_1, \ldots, x_n]$
  - Wanted: $C \in \mathbb{Z}[x_1, \ldots, x_{n-1}]$, the resultant of A and B.
- Modular Algorithm
  - Solve problem in multiple finite precision integer domains.
  - Lift result back to the domain of (arbitrary precision) integers.

Result coefficients become extremely large!



Distributed Maple Implementation

- Modification of Maple `resultant`.
  - Replacement of `bivariate/modular`.
  - Called for bivariate polynomials of high degree.
  - Other cases solved faster by sequential methods.
- Performance Tuning
  - M images per modular task.
  - C/M images per CRA task.
  - $\Rightarrow$ proportional task granularities.
  - $\approx$ 10–30s per task.

Parallelization based on functional model.

Execution Environment

- Sequential Execution.
  - 1 Linux PC (1 Pentium@333MHz).
- Parallel Execution
  - 3 Linux PCs (1 Pentium@333MHz, local Ethernet).
  - 4 SGI Octanes (2 R10000@250MHz, local Ethernet).
  - 5 Linux PCs (1 Pentium@266MHz, local Ethernet).
  - 1 Sun UltraSparc (1 Sparc@296MHz, local Ethernet).
  - 1 SGI Origin (32 R10000@195MHz, 7 used, wide area ATM).
- Raw Performance.
  - Processor Number: 24.
  - Performance Factor: 17.99

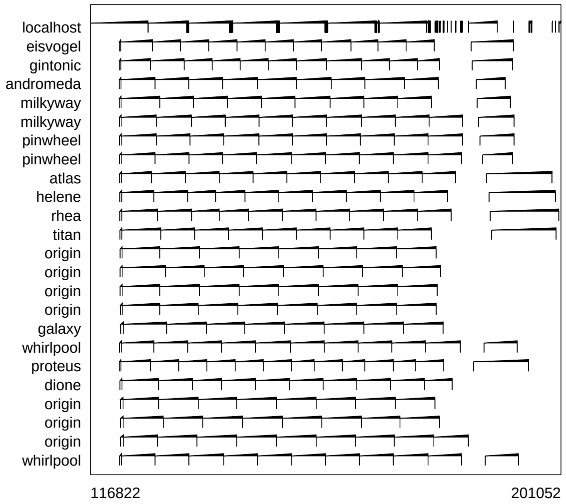
Heterogeneous network of PCs, workstations, multiprocessors.

Benchmark Curves

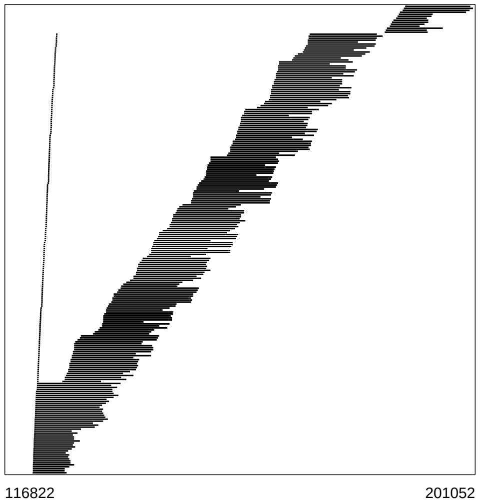
Name	Tot.Deg.	Value
9-11-3	14	$23x^4y^8 - 17x^6y^6 + 22x^8 + 84x^7y^7 + 20x^4y^6 - 49x^3y^8 - 80x^4y^2 - 5x^3y - 92x^3y^2 - 47y^8 + 5xy^5$
10-7-2	13	$55x^8y^2 + 66x^9y^4 + 27y^3 + 31x^2y^5 - 75x^2y^6 - 70y^8 - 97x^2y^9$
10-8-2	18	$-22x^9y^9 - 26x^2y^5 + 85x^8y - 50x^2y^6 - 74x^6y^4 - 31x^8y^7 - 172x^7y^7$
12-7-2	20	$99xy^9 - 12x^6y^2 - 26x^2y^{11} - 47x^8y^{11} - 61x^5y^8 - 90x^9y^{11} + 94x^2y$

Visualization

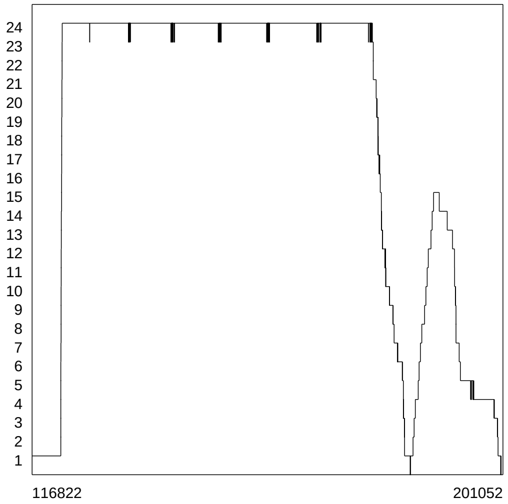
Machines



Tasks

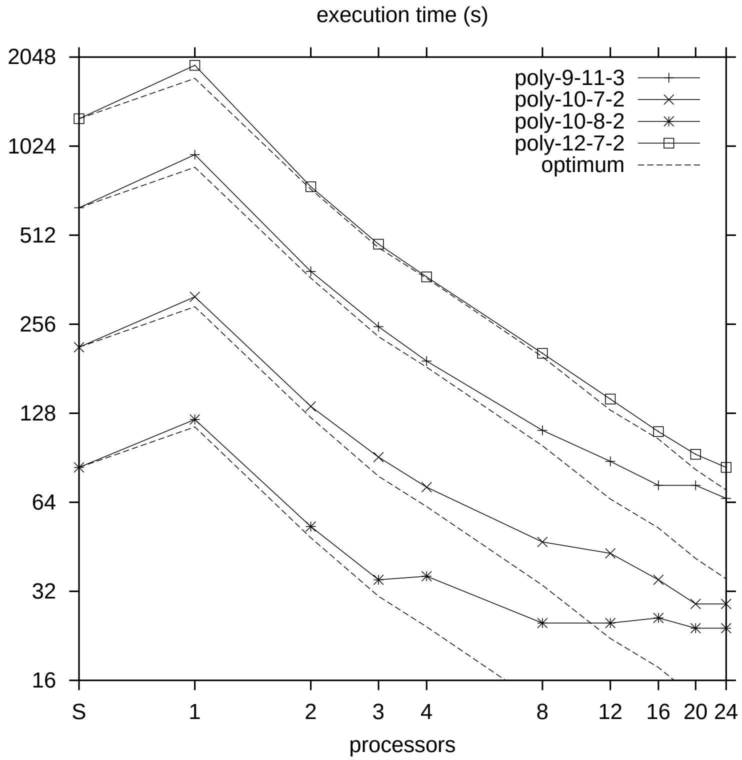


Utilization



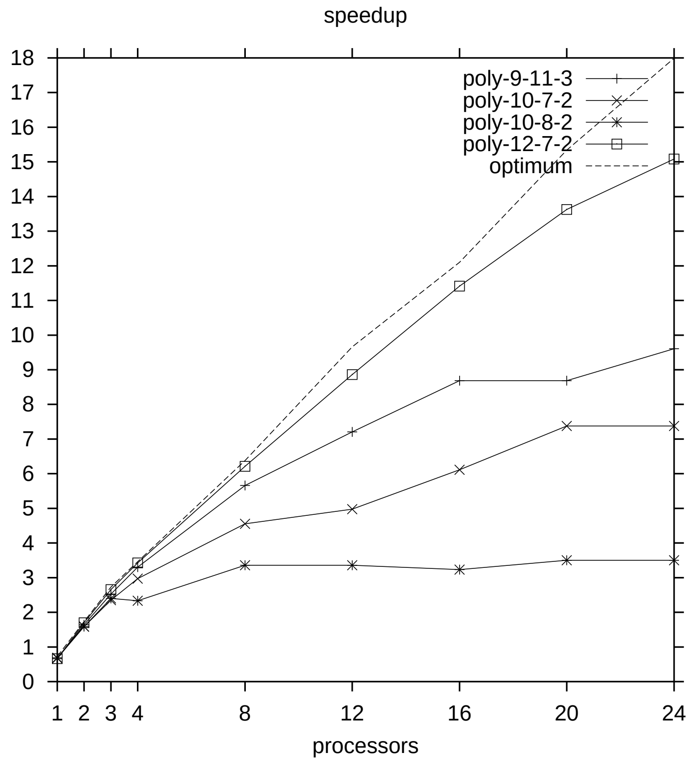
Chinese remaindering becomes bottleneck.

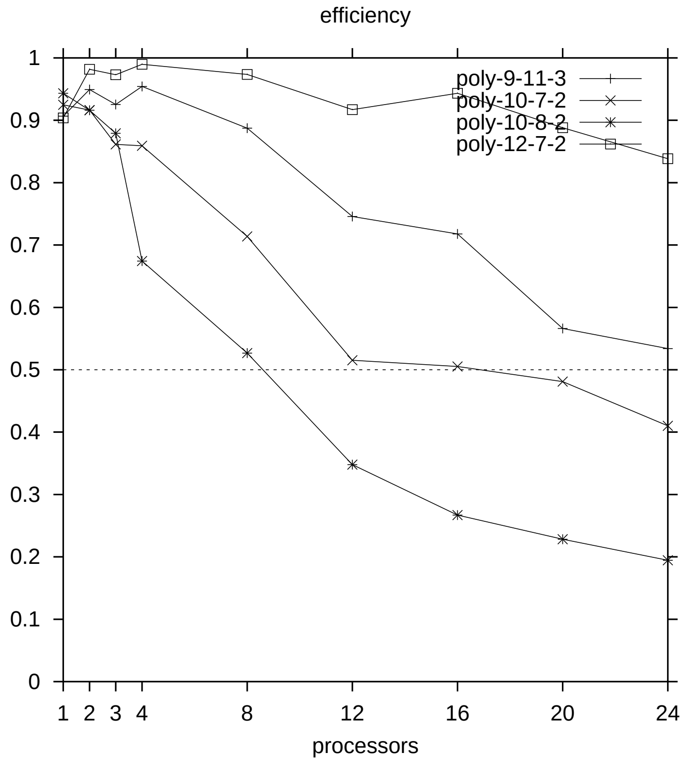
Execution Time



From 1267s (fastest single processor) to 84s (network).

Speedup and Efficiency





80% efficiency with 24 processors in a heterogeneous network.

Conclusions

- Achievements
  - Portable Java-based parallel execution framework.
  - Maple integration via standard I/O interface.
  - High-level parallel programming model.
  - Speedup of 15 for a major CASA algorithm.
- Current Work
  - Parallelization of real root isolation.
  - Fine-tuning of performance parameters.
  - More CASA algorithms.
- Future Work
  - Mathematica integration.
  - Fault tolerance and error recovery.

Further Information

`http://www.risc.uni-linz.ac.at/software/distmaple`