OpenDreamKit's Workpackage 5: High Performance Mathematical Computing

Clément Pernet
and all participants of Workpackage 5

Second OpenDreamKit Project review

JNCF'19, Luminy, Feb 4, 2018

Goal: delivering high performance to math-software users

Systems : GAP | PARI/GP | SageMath | Singular

Components : FLINT | MPIR | LinBox | PPL | NumPy

Goal: delivering high performance to math-software users

Harnessing modern hardware $\rightsquigarrow$ parallelisation

- in-core parallelism (SIMD vectorisation)
- multi-core parallelism
- distributed computing: clusters, cloud

Systems : GAP, PARI/GP, SageMath, Singular

Components : FLINT, MPIR, LinBox, PPL, NumPy

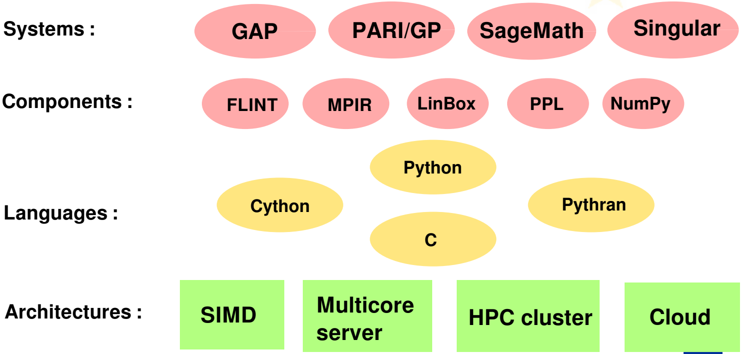
Architectures : SIMD, Multicore server, HPC cluster, Cloud

Goal: delivering high performance to math-software users

Languages

- Computational Maths software uses high level languages (e.g. Python)
- High performance delivered by languages close to the metal (C, assembly)

$\rightsquigarrow$ compilation, automated optimisation



High performance mathematical computing

Goal:

- Improve/Develop parallelization of software components
- Expose them through the software stack
- Offer High Performance Computing to VRE's users

Milestone M8: Seamless use of parallel computing architecture in the VRE (proof of concept)

Astrid wants to run compute intensive routines involving both dense linear algebra and combinatorics. She has access through a JupyterHub-based VRE to a high end multi-core machine which includes a vanilla SAGE installation. She automatically benefits from the HPC features of the underlying specialized libraries (LinBox, ...). This is a proof of concept of the overall framework to integrate the HPC advances of specialized libraries into a general purpose VRE. It will prepare the final integration of a broader set of such parallel features for the end of the project

Organization of WorkPackage 5

Component	Review 1	Review 2	Final review
T5.1 Pari/GP			D5.16
T5.2 GAP			D5.15
T5.3 LinBox		D5.12	D5.14
T5.4 Singular	D5.6, D5.7		D5.13
T5.5 MPIR	D5.5, D5.7		
T5.6 Combinatorics	D5.1	D5.11	
T5.7 Pythran	D5.2	D5.11	
T5.8 SunGrid Engine	D5.3		

Overall

- 20+ software releases
- 7 research papers

Outline

Improving Software Systems

- T5.1: Pari
- T5.2: GAP
- T5.4: Singular

Exact linear algebra (T5.3)

- Exact linear algebra algorithms and implementations (D5.12).
- Distributed computing

Outline

Improving Software Systems

- T5.1: Pari
- T5.2: GAP
- T5.4: Singular

Exact linear algebra (T5.3)

- Exact linear algebra algorithms and implementations (D5.12).
- Distributed computing

T5.1: Pari

D5.16: Pari Suite release, fully supporting parallelization

- Generic parallelization engine is now mature (released since Nov 2016). Support POSIX-threads and MPI.
- Current work: applying it throughout the library
  - Chinese remaindering
  - Rational linear algebra
  - Discrete logarithm
  - Resultants
  - APRCL primality testing

T5.2: GAP

D5.15: Final report of GAP development

- 8 releases were produced
- Towards an integration of HPC-GAP: main release GAP-4.9
  - Build system refactoring
  - Ability to compile in HPC-GAP compatibility mode
- Work in progress:
  - Multithreaded linear algebra: at the level of the `Meataxe` library
  - Introspection functionalities: on-the-fly optimisation decision

T5.4 Singular: Parallel sparse polynomial multiplication

`FLINT` now supports fast sparse multivariate polynomials:

- addition, subtraction, multiplication,
- division, division with remainder, GCD
- evaluation, partial evaluation, composition

T5.4 Singular: Parallel sparse polynomial multiplication

`FLINT` now supports fast sparse multivariate polynomials:

- addition, subtraction, multiplication,
- division, division with remainder, GCD
- evaluation, partial evaluation, composition

Parallelization of the (sparse) multiplication

$$(1+x+y+2z^2+3t^3+5u^5)^{21} \times (1+u+t+2z^2+3y^3+5x^5)^{21}$$

# cores	$\mathbb{Z}$	$\mathbb{Z}_{2^{64}-1}$
1	34145(1.0x)	30349(1.0x)
2	14856(2.3x)	12641(2.4x)
3	10844(3.1x)	8294(3.6x)
4	7205(4.7x)	6274(4.8x)
5	7621(4.4x)	5315(5.7x)
6	6156(5.5x)	4152(7.3x)
7	5679(6.0x)	3739(8.1x)
8	4458(7.6x)	3047(9.9x)

T5.4 Singular: Parallel sparse polynomial multiplication

`FLINT` now supports fast sparse multivariate polynomials:

- addition, subtraction, multiplication,
- division, division with remainder, GCD
- evaluation, partial evaluation, composition

Parallelization of the (sparse) multiplication

$$(1+x+y+2z^2+3t^3+5u^5)^{21} \times (1+u+t+2z^2+3y^3+5x^5)^{21}$$

# cores	$\mathbb{Z}$	$\mathbb{Z}_{2^{64}-1}$
1	34145(1.0x)	30349(1.0x)
2	14856(2.3x)	12641(2.4x)
3	10844(3.1x)	8294(3.6x)
4	7205(4.7x)	6274(4.8x)
5	7621(4.4x)	5315(5.7x)
6	6156(5.5x)	4152(7.3x)
7	5679(6.0x)	3739(8.1x)
8	4458(7.6x)	3047(9.9x)

- Planned improvements to the memory manager $\Rightarrow$ closer to linear scaling
- Parallel division and GCD implementations are in progress.
- Integration into Factory/Singular remains to be done

Outline

Improving Software Systems

- T5.1: Pari
- T5.2: GAP
- T5.4: Singular

Exact linear algebra (T5.3)

- Exact linear algebra algorithms and implementations (D5.12).
- Distributed computing

Task 5.3: LinBox, High performance exact linear algebra

Mathematics is the art of reducing any problem to linear algebra

– W. Stein

Task 5.3: LinBox, High performance exact linear algebra

Mathematics is the art of reducing any problem to linear algebra

– *W. Stein*

Linear algebra: a key building block for HPC

Similarities with numerical HPC

- central elementary problem to which others reduce to
- (rather) simple algorithmic
- high compute/memory intensity

Task 5.3: LinBox, High performance exact linear algebra

Mathematics is the art of reducing any problem to linear algebra
– W. Stein

Linear algebra: a key building block for HPC

Similarities with numerical HPC

- central elementary problem to which others reduce to
- (rather) simple algorithmic
- high compute/memory intensity

Specificities

- Multiprecision arithmetic $\Rightarrow$ lifting from finite precision ($\mathbb{F}_p$)
- Rank deficiency $\Rightarrow$ unbalanced dynamic blocking
- Early adopter of subcubic matrix arithemtic $\Rightarrow$ recursion

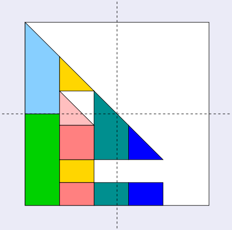
Task 5.3: LinBox, High performance exact linear algebra

1. Algorithmic innovations:
   1.1 Rank deficient dense Gaussian elimination
   1.2 Quasiseparable matrices
   1.3 Outsourced computing security
2. Software releases and integration:
   2.1 LinBox ecosystem: `LinBox`, `fflas-ffpack`, `givaro`
   2.2 `SageMath` integration
3. Distributed Parallel linear algebra: rational solver
   3.1 Chinese remaindering based
   3.2 Dixon Lifting based

Rank deficient dense Gaussian elimination

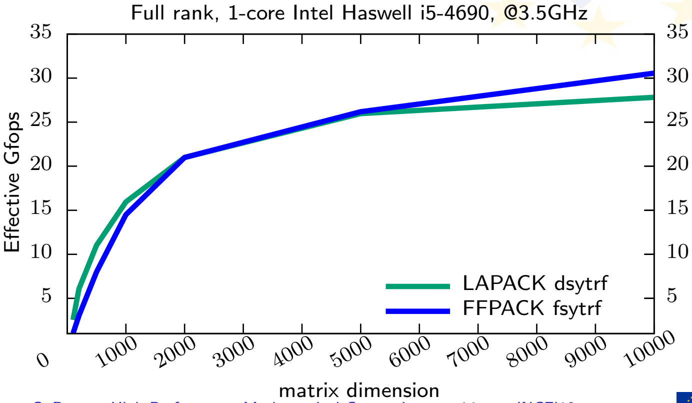
[ISSAC'18] Symmetric triangular factorization

- First unconditional recursive algorithm
- Pivoting revealing the Rank Profile Matrix
- $O(n^2 r^{\omega-2})$ ($= 1/3n^3$ with $\omega = 3, r = n$)
- Also hot topic in numerical linear algebra (LAPACK Working notes 294, Dec'17)



LAPACK vs FFPACK modulo 8 388 593

n	LAPACK (numerical) dsytrf (LDLT)	FFPACK (exact) fsytrf (LDLT)
5000	1.60s	1.59s
10000	11.98s	10.90s

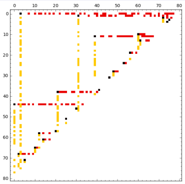


Quasiseparable matrices

Matrices with low off-diagonal rank

[ISSAC'16, JSC'18] New compact representation and algorithms

- Matches the best space complexities
- Reduction to matrix multiplication
- **Breakthrough:** flat representation (non hierarchical)

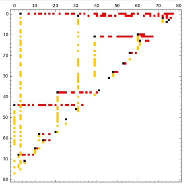


Quasiseparable matrices

Matrices with low off-diagonal rank

[ISSAC'16, JSC'18] New compact representation and algorithms

- Matches the best space complexities
- Reduction to matrix multiplication
- **Breakthrough:** flat representation (non hierarchical)



Follow-up: on-going work with numerical HPC colleagues:

- S. Chandrasekaran (UCSB)
- T. Mary (U. Manchester, `Mumps`)

Software releases and integration

LinBox ecosystem

`givaro`: field/ring arithmetic

`fflas-ffpack`: dense linear algebra over finite field

`LinBox`: exact linear algebra

Tightly integrated in `SageMath`

Software releases and integration

LinBox ecosystem

`givaro`: field/ring arithmetic	4 releases
`fflas-ffpack`: dense linear algebra over finite field	6 releases
`LinBox`: exact linear algebra	6 releases
Tightly integrated in `SageMath`	13 tickets

Software releases and integration

LinBox ecosystem

`givaro`: field/ring arithmetic	4 releases
`fflas-ffpack`: dense linear algebra over finite field	6 releases
`LinBox`: exact linear algebra	6 releases
Tightly integrated in `SageMath`	13 tickets

Featuring

- Full functional implementations of new algorithmic contributions
- Improved vectorization and parallel routines
- Drastic improvement of reliability (continuous integration, test-suite coverage, randomized certificates, etc)

Distributed computing (on-going work)

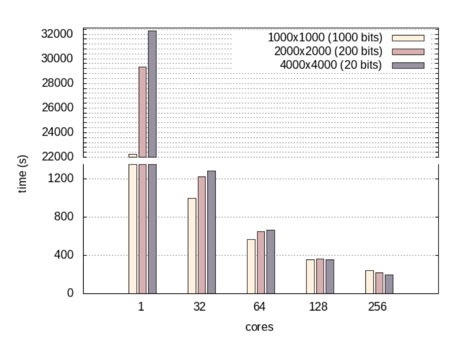
D5.14: Distributed exact linear system solving

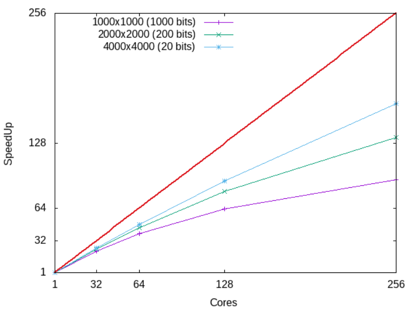
- 2 full time engineers
- Communication and serialization layer done
- Prototype MPI parallelization of Chinese remainder based solver.

Distributed computing (on-going work)

D5.14: Distributed exact linear system solving

- 2 full time engineers
- Communication and serialization layer done
- Prototype MPI parallelization of Chinese remainder based solver.



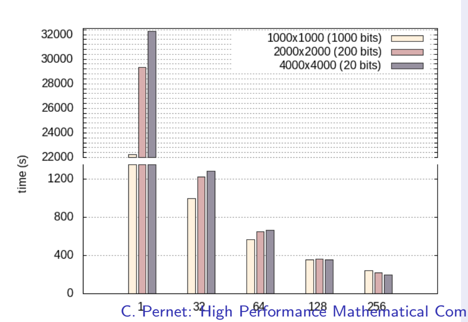


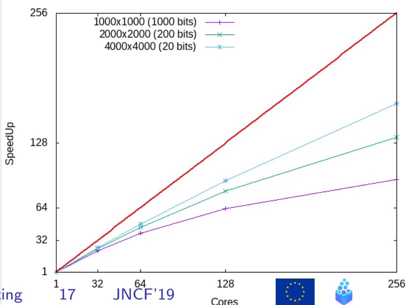
Distributed computing (on-going work)

D5.14: Distributed exact linear system solving

Work in progress:

- Major refactorization of LinBox solver code
- Hyrbid OpenMP-MPI implementation
  - Better memory usage,
  - yet still slower than full-MPI
- Hybrid combination of CRT+Dixon and parallelization.





Exploratory aspects: security of outsourced computing

Trust and reliability over the Cloud

Outsourcing computations:

- trusted lightweight client computer
- untrusted powerful cloud server

Contributions:

- Linear time certif. for LU, Det, Rank Profile Matrix [ISSAC'16,17]
- Error correction in LU decomp.

Exploratory aspects: security of outsourced computing

Trust and reliability over the Cloud

Outsourcing computations:

- trusted lightweight client computer
- untrusted powerful cloud server

Contributions:

- Linear time certif. for LU, Det, Rank Profile Matrix [ISSAC'16,17]
- Error correction in LU decomp.

Secure Multiparty Computation

- each player contribute with a share of the input
- shares must remain private

- SMC protocol for Strassen's MatMul ([D. Lucas' talk Wed. 6])
- SMC protocol for LUP, and LinSys

Exploratory aspects: security of outsourced computing

Trust and reliability over the Cloud

Outsourcing computations:

- trusted lightweight client computer
- untrusted powerful cloud server

Contributions:

- Linear time certif. for LU, Det, Rank Profile Matrix [ISSAC'16,17]
- Error correction in LU decomp.

Secure Multiparty Computation

- each player contribute with a share of the input
- shares must remain private

- SMC protocol for Strassen's MatMul ([D. Lucas' talk Wed. 6])
- SMC protocol for LUP, and LinSys

Angle: algorithm/application-based solutions $\rightsquigarrow$ towards practicality

Conclusion

Outcome of WorkPackage 5

	Review 1	Review 2	Final review
T5.1 Pari/GP			D5.16
T5.2 GAP			D5.15
T5.3 LinBox		D5.12	D5.14
T5.4 Singular	D5.6, D5.7		D5.13
T5.5 MPIR	D5.5, D5.7		
T5.6 Combinatorics	D5.1	D5.11	
T5.7 Pythran	D5.2	D5.11	
T5.8 SunGrid Engine	D5.3		

Overall

- 20+ software relases
- 7 research papers

Lessons learnt

From dedicated to general purpose HPC components:

- Early instances of HPC computer algebra: dedicated to some target application (cryptanalysis, etc)
- Building a general purpose HPC component:
  - challenging
  - longer term sustainability

Risk of technology dependency

- Cilk: from success to shut-down
- Interchangeability and modularity

Security for outsourced computing

- exploit algebraic properties of the problem
  $\rightsquigarrow$ well suited for computer algebra