CECM Research Accomplishments 2016–2021

Assembled by Michael Monagan, Director

These research accomplishments by CECM members listed here all used the CECM computers. The CECM computers provide a range of very accessible machines with a number of cores and an amount of memory that goes well beyond that available in a desktop machine. Our two largest servers are `gaby` (16 cores, 64 gigabytes) and `jude` (20 cores, 128 gigabytes).

The software used includes Magma, Maple, Macaulay2, and C programs. **CECM members are highlighted in blue.** HQP are underlined.

1: Computational Algebra

Probably the single most important operation for the overall efficiency of a computer algebra system like Magma, Maple and Mathematica, is the polynomial greatest common divisor operation (GCD). It is applied to simplify a formula which is a fraction A/B of two polynomials A and B. In

> J. Hu and M.B. Monagan. A Fast Parallel Sparse Polynomial GCD Algorithm.
> *J. Symbolic Computation* **105**:(1) 28–63, Springer, July 2021.
> DOI: https://doi.org/10.1016/j.jsc.2020.06.001

Jiaxiong Hu and **Michael Monagan** developed a new algorithm for computing $G = \gcd(A, B)$ which uses a new sparse interpolation method that in some sense is optimial; it uses the minimum number of points necessary, up to a constant factor, to interpolate G. The Table below shows how much faster the new GCD algorithm is than Maple's and Magma's GCD algorithms for a typical large problem where G has 10^4 terms and A and B have 10^6 terms. Timings are in seconds.

New(1 core)	New(16 cores)	Maple	Magma
7.61s	0.685s	22111.6s	1786.0s

Hu and Monagan used the `gaby` server to develop a parallel implementation of the algorithm in Cilk C and to obtain Maple and Magma timings. In current work, Monagan is collaborating with Maplesoft personnel to recode the algorithm in Maple so that it can be integrated into the Maple library. Monagan will present the work at the 2021 Maple Conference in November.

2: Cryptography

Almost perfect non-linear (APN) functions are important building blocks of block ciphers in cryptography. They map binary vectors to binary vectors. Of particular interest are those APN functions that are also permutations of their domain. Not many APN permutations are known, and this is a very active area of research. **Petr Lisonek** with his student Benjamin Chase used CECM computers to gain deeper insight into several classes of APN permutations. They characterized exactly when certain types of quadrinomials (so-called Kim functions) are APN permutations, and they gave theoretical (computer-free) constructions for some other classes of APN permutations.

B. Chase, P. Lisonek, Kim-type APN functions are affine equivalent to Gold functions. *Cryptogr. Commun.* (2021). DOI: https://doi.org/10.1007/s12095-021-00490-2.

B. Chase, P. Lisonek, Constructions and applications of Walsh zero spaces. The 6th International Workshop on Boolean Functions and their Applications (BFA), Rosendal, Norway, September 2021. (accepted)

3: Combinatorial Designs

Jonathan Jedwab and his Masters student Jingzhou Na have made extensive use of the CECM machines. Jingzhou gave the opening talk "Perfect sequence covering arrays" at the minisymposium on Algebraic and Combinatorial Approaches to Designs and Codes, Canadian Discrete and Algorithmic Mathematics Conference, in May 2021.

The objective is to determine the smallest repetition constant, in terms of the pair (n, k), for which a perfect sequence covering array involving k-subsequences of an n-sequence exists. Until now, only one such constant greater than 1 has been precisely determined, and the author who achieved this in 2020 for the pair $(n, k) = (5, 3)$ wrote that finding further such values "seems challenging". By combining combinatorial arguments with an assumed group structure and running computer searchers, we were able to determine an exact value for three further pairs: $(n, k) = (6, 3)$ and $(7, 3)$ and $(7, 4)$. Further calculations are in progress on 12 cores of the `jude` server.

4: Algebraic Geometry

Nathan Ilten used the Macaulay2 computer algebra system for the following works. Macaulay2 is a special purpose system aimed at computations in algebraic geometry. It has revolutionized the discipline of algebraic geometry.

For the first paper below (submitted) Nathan used the Schubert2 package in Macaulay2 (running on `jude`) to compute some examples in enumerate geometry of the number of lines contained in some special varieties. For the second paper (published) Nathan used Macaulay2 (again running on `jude`) to compute the projective dual to a surface. For the third paper below (submitted) Nathan computed the tropical dual of a space curve in Tropical tangents for complete intersection curves.

Nathan Ilten, Tyler L. Kelly
Fano Schemes of Complete Intersections in Toric Varieties
Submitted, 23 pages, 2019. https://arxiv.org/abs/1910.05593

Nathan Ilten and Yoav Len
Projective duals to algebraic and tropical hypersurfaces.
Proceedings of the London Mathematical Society, **119**(5):1234–1278, Wiley, November 2019.

Nathan Ilten and Yoav Len
Tropical tangents for complete intersection curves. 40 pages, April 2021.
Preprint. https://arxiv.org/abs/2104.15059

5: Number Theory

Imin Chen has been using the Magma computer algebra system on the CECM servers for computations needed to resolve generalized Fermat equations, which in particular rely on computations of Hilbert modular forms and point counting on abelian varieties. For some equations, the initial computations took a long time (two months), so having a server to run the programs in the background

was crucial. The resulting output and profiling information from the computations motivated new theoretical improvements which can now prove the same results in about 3 hours.

The following publications relate to this work. In the second one, Frey abelian varieties are used for the first time to completely resolve the family of generalized Fermat equations

$$x^7 + y^7 = 3z^n \quad \text{for } n \geq 2.$$

As another application of higher dimensional Frey varieties, we prove both an asymptotic and optimized result for the generalized Fermat equation,

$$x^{11} + y^{11} = z^p,$$

where p is a prime exponent. This is the first time such a family of signatures has been treated and the use of higher dimensional Frey varieties is essential for this application, allowing a proof of an asymptotic result in a few minutes.

N. Billerey, I. Chen, L. Dembélé, L. Dieulefait, N. Freitas.
Some extensions of the modular method and Fermat equations of signature $(13, 13, n)$.
Submitted to *Algebra and Number Theory*, 21 pages.

N. Billerey, I. Chen, L. Dieulefait, and N. Freitas.
A multi-Frey approach to Fermat equations of signature (r, r, p).
Trans. Amer. Math. Soc. **371**:8651–8677, AMS, 2019.

I. Chen and G. Glebov. On Chudnovsky-Ramanujan type formulae.
Ramanujan J. **46**(3):677–712, 2018.

N. Billerey, I. Chen, L. Dieulefait, and N. Freitas.
A result on the equation $x^p + y^p = z^r$ using Frey abelian varieties.
Proc. Amer. Math. Soc. **145** (2017), no. 10, 4111-4117.

6: Computational Algebra

In June 2020 **Michael Monagan** factored a polynomial of degree one billion over a large prime field (over $\mathbb{F}_p$ where $p = 5 \times 2^{55} + 1$) which sets a **world record** for the largest such polynomial ever factored. The factorization ran on the `jude` server. After much work it now takes under 4000 seconds on 10 cores and needs only 121 gigabytes of RAM.

This is joint work with Joris van der Hoeven of the Laboratoire d'informatique de l'École polytechnique, Paris. Joris visited the CECM from September 2019 through July 2020, staying in Vancouver during the pandemic. In January 2020 he gave two joint seminars to the Computer Algebra and Discrete Mathematics groups on his $O(n \log n)$ algorithm for integer multiplication.

The software includes a C library of asymptotically fast algorithms which utilize the Fast Fourier Transform. Parallelization of the algorithms was done using Cilk C.

Joris van der Hoeven and Michael Monagan.
Computing one billion roots using the tangent Graeffe method.
Communications in Computer Algebra, **54**(3): 65–85, September 2020.
Preprint: https://hal.archives-ouvertes.fr/hal-02525408/

7: Maple Contribution

In 2019, **Michael Monagan** and Baris Tuncer, in collaboration with Jürgen Gerhard of Maplesoft, installed a new polynomial factorization algorithm and associated library into Maple. Polynomial factorization is one of the main capabilities offered by Maple. The significance of this work is that the new algorithm factors polynomials in n variables with integer coefficients in random polynomial time. The old algorithm is exponential in the number of variables.

This work encompasses 6 papers, 5 conference presentations, one invited talk at the Chinese Academy of Sciences, and 4 departmental seminars. All implementation work was done on the CECM computers. The paper below presents the new algorithm, shows some benchmarks, and gives an average case complexity analysis.

> Michael Monagan and Baris Tuncer.
> The complexity of sparse Hensel lifting and sparse polynomial factorization.
> *J. Symbolic Computation* **99**:189–220, Elsevier, 2020.
> DOI: https://doi.org/10.1016/j.jsc.2019.05.001

Subsequent work by **Monagan** and PhD students Baris Tuncer and Tian Chen has focused on the development of a first parallel algorithm for polynomial factorization. We made heavy use of the CECM servers `gaby` and `jude` for parallel experiments and benchmarking. Our software is typically over 1000 times faster than other software for polynomial factorization.

> Michael Monagan and Baris Tuncer.
> Sparse multivariate polynomial factorization: a high-performance design and implementation.
> *Proceedings of ICMS 2018*, LNCS **10931**:359–368, Springer, 2018.

> Tian Chen and Michael Monagan.
> The Complexity and Parallel Implementation of two Sparse Multivariate Hensel Lifting Algorithms for Polynomial Factorization.
> *Proceedings of CASC 2020* LNCS **12291**:150–169, Springer, 2020.

8: Abstract algebra and cryptography

Petr Lisonek studied constructions of maximally non-associative quasigroups from nearfields and fields. Maximally non-associative quasigroups were proposed as possible resources for constructions of hash functions in cryptography. Until 2018 it was conjectured that these objects do not exist. Based on extensive computations and tuning of parameters of the proposed constructions, Lisonek was able to obtain two infinite families of maximally non-associative quasigroups. Further extensive computations were required to investigate the densities of favorable parameters of the new constructions, which determine the probability of success of the randomized constructions. The probability of success was lower bounded theoretically, but the computations showed that the actual rate of success is about twice the lower bound.

All computations were done in Magma on the CECM network using `gaby` and `michelle`.

References:

> A. Drapal, P. Lisonek. Maximal nonassociativity via nearfields.
> *Finite Fields and Their Applications* **62** (2020), 101610.

> P. Lisonek. Maximal nonassociativity via fields.
> Designs, Codes and Cryptography **88**(12):2521–2530, 2020. Preprint arXiv:1910.09825.

9: Quantum codes

Quantum computers are very delicate physical systems, and the information stored in them is highly susceptible to corruption. It is important to implement measures to make quantum computers more robust and fault-tolerant. Quantum codes are one of tools to increase fault tolerance. There is a canonical construction of quantum codes from classical codes, which requires the classical linear code to be self-orthogonal.

Petr Lisonek with students has been developing constructions of quantum codes in which the self-orthogonality condition is slightly relaxed, and they have shown that many new and better codes can be constructed. Computing the minimum distance (a parameter that determines the error correction capacity) of the resulting codes is very time intensive, and CECM computers are being used extensively for this purpose. Altogether several dozens of new codes better than the previously known ones have been found, and many more codes can be obtained from them by secondary constructions.

References:

R. Dastbasteh, P. Lisonek.
Constructions of quantum codes from nearly self-orthogonal codes. Preprint, 2020.

P. Lisonek. Constructions of quantum codes.
The 3rd International Workshop on Boolean Functions and their Applications (BFA), Loen, Norway, 2018. https://people.uib.no/chunlei.li/workshops/BFA2018/Slides/Lisonek.pdf

10: Golay sequences

In the work below **Jonathan Jedwab** searched for new Golay sequences.

Three-phase Golay sequence and array triads by Aki Avis and Jonathan Jedwab.
J. Comb. Theory A **180** 105422:1–22, 2021. Preprint arXiv:1910.05661

I used six CECM machines (four cores each) to run parallel programs in C to obtain the data in Table 1 for the two largest sequence lengths (23 and 24). This took approx two weeks, instead of the circa 1 year that I would have needed if I'd just used my own machine. The data for the existence pattern of these objects allowed us to see the strengths and limitations of our constructions.

11: Computational Arithmetic Geometry

Nils Bruin has used Magma for computations in arithmetic geometry for which the CECM has several licensed servers.

Nils Bruin, E. Victor Flynn, Ari Shnidman, Genus two curves with full sqrt(3)-level structure and Tate-Shafarevich groups, ArXiv preprint arXiv:2102.04319 (2021).

This work studies the arithmetic of an interesting family of genus two curves and their Jacobians. It has substantial arithmetic-geometric computations in it.

Nils Bruin, Daniel Lewis.
Two-cover descent on plane quartics with rational bitangents.
Presented at 14th Algorithmic Number Theory Symposium, June 2020. *Proceedings of ANTS XIV*, The Open Book Series 4–1 (2020) 73–89. Preprint: arXiv:2003.00666

The main contribution in the work above is a practical method for deciding if certain quartic equations have rational solutions. Approximately a CPU-week of computations using MAGMA. Computations provide valuable evidence for long-standing conjectures, as well as illustrated surprising effectiveness of the proposed methods.

> Nils Bruin, Jordan Thomas, and Anthony Várilly-Alvarado.
> Explicit computation of symmetric differentials and its application to quasi-hyperbolicity.
> Submitted to *Algebra and Number Theory*, 24 pages. Preprint: arXiv:1912.08908.

This work required several weeks worth of computations using MAGMA. Computations represent examples of applications of new theoretical methods to actual examples, illustrating effectiveness and practicality of new methods.

12: Gröbner basis computation.

Roman Pearce and **Michael Monagan** collaborated with Jürgen Gerhard of Maplesoft in an NSERC CRD research grant from 2013 to 2017 to develop a new Gröbner basis engine for Maple. Gröbner bases are the main tool used to solve systems of polynomial equations and other problems in algebraic geometry. Pearce and Monagan designed and implemented a *parallel* Gröbner basis engine on the CECM multi-core servers. This work was integrated into Maple.

> Michael Monagan and Roman Pearce. An algorithm for splitting polynomial systems based on F4. Presented at PASCO 2017, Kaiserslautern, Germany, July 23–24, 2017.
> *Proceedings of PASCO 2017*, ACM, July 2017.
> DOI: https://dl.acm.org/doi/10.1145/3115936.3115948

> Michael Monagan and Roman Pearce. A Compact Parallel Implementation of F4.
> Presented at PASCO 2015, Bath, England, July 10–11, 2015.
> *Proceedings of PASCO 2015*, ACM Press, pp. 95–100, 2015.