Giac/Xcas and Pari/GP

B. Parisse

University of Grenoble

January, 2016

Outline

Giac vs Xcas

- Giac/Xcas is a **general purpose** symbolic algebra software: $\sin(x)$ is left as is, not converted to a series.
- Giac is an open-source C++ library (GPL3). It can optionally be linked to libraries like NTL, Pari, GSL, Lapack.
- Xcas is the "native" GUI interface, icas is the commandline interface, both provide access to PARI functions (in other words, Xcas is to Giac what PARI is to GP)
- Xcas is also available in your browser (w/o PARI)
  `www-fourier.ujf-grenoble.fr/~parisse/xcas.html`
  Works everywhere (smartphone, tablet) without install, once downloaded, no server/Internet required, computations are done locally. Performance penalty 2 to 7.

Interfaces

- Giac has interfaces for Java (JNI), Python (giacpy) and Javascript (compiled by emscripten)
- Giac is used by projects like Geogebra, the HP Prime calculator, ported to the TI nspire calculator, and used by some apps (Xcas Pad, PocketCAS, CAS Calc P11),
- public SVN access via Geogebra
  `dev.geogebra.org/trac/browser/trunk/geogebra/giac/`
- The javascript version can be used to build interactive HTML output from LaTeX source files with the `hevea` compiler. Available from the `share` directory of the `giac` archive or
  `www-fourier.ujf-grenoble.fr/~parisse/emgiac.tgz`

Implicit calls of PARI inside Giac

- Ducos multivariate resultant algorithm
- Approx. univariate factorization for large degree and coefficients, for example `katsura8` rational univariate representation
- Univariate factorization if there are many modular factors that recombine over $\mathbb{Z}$ and NTL is not available
- Number field factorization and computation of galois conjugates
- Integer primality certification,
- Integer factorization if the number has ECM-range factors and is too large for giac own quadratic sieve implementations.

Explicit calls of PARI inside Giac

- `pari()` exports all PARI commandnames like Giac/Xcas user commands.
- if a commandname is the same as a native Giac command, add a `pari_` prefix
- C++ converters `gen2GEN` and `GEN2gen` (`pari.cc` source file)

Limitations of PARI/GP inside Giac

- Some of these limitations might be removed with the help of PARI experts!
- Mutex protection: only one thread can access to PARI at the same time
- Memory : PARI stack initialization is done once. The stack can not be increased from Giac with `allocatemem` (segfault). Default 64M, might be overriden at initialization by `export PARI_SIZE=`
- Some objects (e.g. multivariate polynomials) are converted using strings (inefficient, problems with variable ordering)
- I have compiled a javascript version of `libpari.a`, `www-fourier.ujf-grenoble.fr/~parisse/parixcas.html` but there are many bugs... Working matdet, pari_factor, expm1... Not working lngamma, fibonacci...

Multivariate polynomials in Giac

- Multivariate polynomials are represented as sparse, distributed (unlike recursive in PARI/GP).
- Fast multiplication : `n:=20; f:=(1+x+y+z+t)^n+1; normal(g:=f*(f+1));` giac 0.66s / gp 7.4s
- Fast GCD modular algorithm for coefficients in $\mathbb{Z}$, also available for coefficients in algebraic extensions of $\mathbb{Q}$
- Fast multivariate factorization (speed comparable to recent versions of Maple) `factor(g)` giac 6.7s
- Fast Gröbner basis f4 algorithm (revlex order and double revlex for elimination), fast rational univariate representation. Most of the time (much) faster than Singular (e.g. cyclic9 5mn, cyclic10 14h), over $\mathbb{Q}$ the modular algorithm speed is comparable to magma or mgb.

Linear algebra in Giac

- Fast system solving with coefficients in $\mathbb{Z}$
- Fast determinant, characteristic polynomial and inverse of matrices with integer coefficients (probabilistic or deterministic). `n:=500; a:=ranm(n,n,10); time(det(a))` giac 0.3s, PARI 16.3s
- Is there interest in calling Giac from PARI for these kind of algorithms ?