Solving Polynomial Systems with PHCpack and phcpy

Jan Verschelde

University of Illinois at Chicago
Department of Mathematics, Statistics, and Computer Science
http://www.math.uic.edu/~jan
jan@math.uic.edu

Graduate Computational Algebraic Geometry Seminar

Outline

1. Introduction
   - what is PHCpack? why phcpy?
   - need for the software and the competition
   - sustainable software development

2. Mathematical versus Software Problems
   - applying mathematical theorems

3. Ongoing and Future Developments
   - building the package phcpy
   - accelerating with Graphics Processing Units (GPUs)
   - web interface

Solving Polynomial Systems with PHCpack and phcpy

1. Introduction
   - what is PHCpack? why phcpy?
   - need for the software and the competition
   - sustainable software development
2. Mathematical versus Software Problems
   - applying mathematical theorems
3. Ongoing and Future Developments
   - building the package phcpy
   - accelerating with Graphics Processing Units (GPUs)
   - web interface

what is PHCpack?

- Free and open source software package, GNU GPL license. ACM TOMS archived version 1.0 as Algorithm 795.
- PHC = Polynomial Homotopy Continuation

  A homotopy is a family of polynomial systems, e.g.:

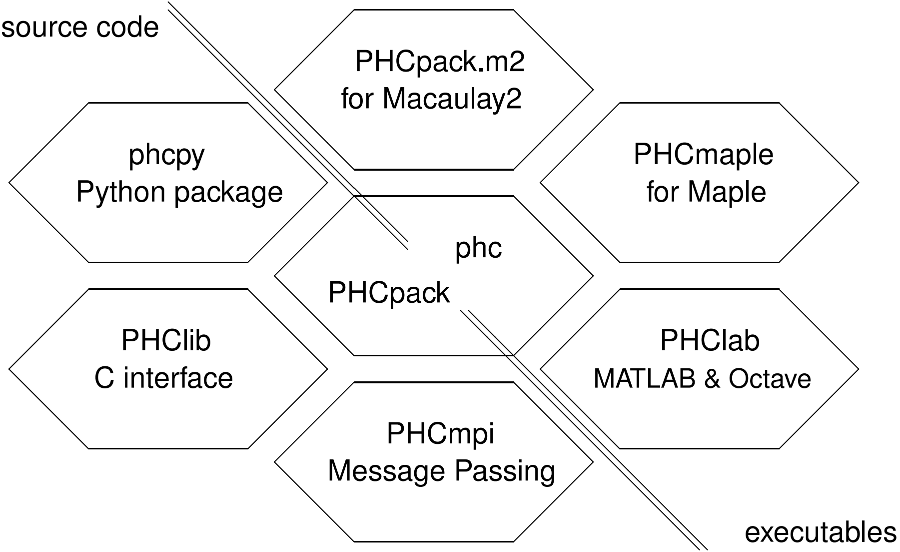
  $$h(\mathbf{x}, t) = \gamma(1 - t) \cdot g(\mathbf{x}) + t \cdot f(\mathbf{x}) = \mathbf{0}, \quad \gamma \in \mathbb{C},$$

  where t goes from 0 to 1

  - starting at the solutions of the good system $g(\mathbf{x}) = \mathbf{0}$,
  - ending at the solutions of the system $f(\mathbf{x}) = \mathbf{0}$ we want to solve.

  Continuation methods are also called path tracking methods, applying predictor-corrector methods to track the solution paths defined by the homotopy $h(\mathbf{x}, t) = \mathbf{0}$.

interfaces to PHCpack



the role of phcpy

`phcpy` is an API, Application Programmers Interface, to PHCpack.

Users of `phc`

- navigates through menus, and
- work with input and output files.

Users of `phcpy`

- have persistent objects replacing intermediate files,
- solve polynomial systems via scripts or in interactive shell.

Instead of using another computer algebra or scientific software system to wrap the `phc` executable,
with `phcpy` we develop software around PHCpack.

why would phcpy be any better than other interfaces?

Setting up a generator for the points along a path:

```
>>> from phcpy.solver import total_degree_start_system
>>> p = ['x**2 + 4*x**2 - 4;', '2*y**2 - x;']
>>> (q,s) = total_degree_start_system(p)
>>> from phcpy.trackers import initialize_standard_tracker
>>> from phcpy.trackers import initialize_standard_solution
>>> from phcpy.trackers import next_standard_solution
>>> initialize_standard_tracker(p,q)
>>> initialize_standard_solution(len(p),s[0])
>>> s1 = next_standard_solution()
```

With `next_standard_solution()` we get the next point on a solution path, computing in standard double precision.

The `next()` is similar to the `yield` construction of Python.

getting points on a solution path

```
>>> print s1
t :  1.00000000000000E-01   0.00000000000000E+00
m : 1
the solution for t :
 x :  9.96338438384030E-01   4.70831004481527E-03
 y :  9.96408320626402E-01   4.95310952563875E-03
== err :  2.375E-05 = rco :  1.000E+00 = res :  3.619E-10 =
>>> print next_standard_solution()
t :  2.00000000000000E-01   0.00000000000000E+00
m : 1
the solution for t :
 x :  9.80919860804043E-01   1.78496473654540E-02
 y :  9.81218221286503E-01   2.32056259678926E-02
== err :  1.671E-08 = rco :  1.000E+00 = res :  1.424E-16 =
```

filling a need

Polynomial systems occur in many fields of science and engineering.

Algebraic geometry studies solutions of polynomial systems.

Polyhedral homotopies are too complicated and too delicate to be left as an implementation dedicated to solve one particular application.

Over 70 research publications, not written by co-developers of PHCpack, document applications solved with the aid of the software.

The blackbox solver `phc -b` is a convenient to use.

PHCpack prototyped many new homotopy continuation algorithms

- solve sparse systems with polyhedral methods
- Pieri homotopies and Littlewood-Richardson homotopies
- compute a numerical irreducible decomposition
- deflate isolated singularities, etc...

the competition

There are three major alternatives:

1. HOM4PS (Michigan State, directed by Tien-Yien Li)
   - \+ fast mixed volume calculators and polyhedral homotopies
   - – current versions are not free nor open source
2. Bertini (Notre Dame, directed by Andrew Sommese)
   - – no support for polyhedral methods
   - \+ adaptive multiprecision path trackers
3. NAG4M2 (Georgia Tech, directed by Anton Leykin)
   - \+ well integrated with Macaulay2
   - \+ does not reinvent the wheel:
     $\rightarrow$ interfaces to Bertini, HOM4PS, PHCpack.

sustainable software development

The first release of `phc` as an executable occurred in 1995, almost twenty years ago.

In computational science, *reproducibility* is important:

- Can you reproduce computational results from the literature?
- Can anybody else reproduce the results?

Application of software engineering principles:

- release early and release often,
- version control via github,
- documentation generated with Sphinx,
- documented test and use cases.

Let us keep PHCpack and phcpy relevant for the next twenty years...

Solving Polynomial Systems with PHCpack and phcpy

1. Introduction
   - what is PHCpack? why phcpy?
   - need for the software and the competition
   - sustainable software development
2. Mathematical versus Software Problems
   - applying mathematical theorems
3. Ongoing and Future Developments
   - building the package phcpy
   - accelerating with Graphics Processing Units (GPUs)
   - web interface

applying mathematical theorems

What does solving mean?

- For isolated solutions: apply polyhedral homotopies, based on:

Theorem (Bernshteǐn, 1976)

The number of isolated solutions with nonzero coordinates is bounded by the mixed volume of the Newton polytopes of the polynomials.

- For positive dimensional solution sets we compute a numerical irreducible decomposition.
  - Cascade of homotopies separate sets of different dimensions.
  - Equidimensional solution sets are represented by witness sets.
  - Factorization methods yield the irreducible components.
- Exploiting structure: symmetry and sparsity.

tropical algebraic geometry

Apply the fundamental theorem of tropical algebraic geometry to solving polynomial systems.

A polyhedral method for positive dimensional solution sets:

- Intersecting a positive dimensional solution set with coordinate planes corresponds to taking an asymptotic view on the set.
- At infinity, the dimension of solution sets drops and solutions at infinity correspond to solutions of initial forms.
- The initial forms define the leading terms in a Puiseux series expansion of the positive dimensional solution set.

Success on the cyclic n-roots problem provides a proof-of-concept for this new polyhedral method. What remains to do:

- Develop robust software tools for this method.
- Apply on a wide range of benchmark problems.

Solving Polynomial Systems with PHCpack and phcpy

1 Introduction
- what is PHCpack? why phcpy?
- need for the software and the competition
- sustainable software development

2 Mathematical versus Software Problems
- applying mathematical theorems

3 Ongoing and Future Developments
- building the package phcpy
- accelerating with Graphics Processing Units (GPUs)
- web interface

building the package phcpy

The main goals of the Python API phcpy:

- export all functionaly offered by phc,
- innovative uses, e.g.: as in the next().

Prototypes of algorithms in Python connect with phcpy.
For computational bottlenecks go to a compiled language.

The current version of phcpy is still preliminary: 0.2.1.
Better installation support is needed, especially on Windows.

Additional modules to phcpy should help the user define customized homotopies and manage path tracking jobs.

GPU acceleration

Motivation for using Graphics Processing Units:

- theoretical peak performance of a GPU is one teraflop,
- less expensive than CPU and more energy effiicient.

Challenges for software development:

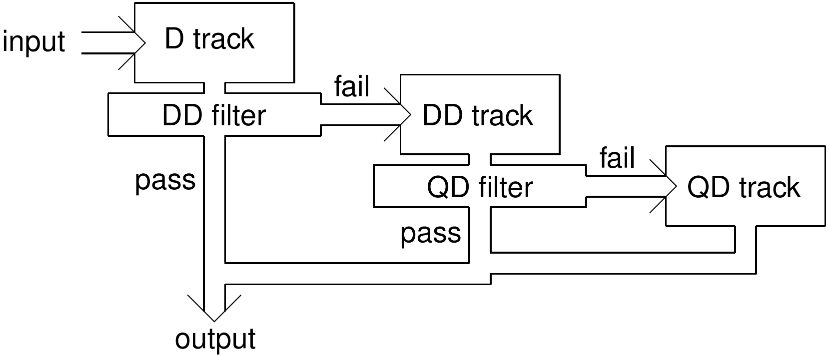
- need for massively parallel algorithms,
- different programming models require rewriting of codes.

The main benefit is to achieve a quality up:

- apply double double and quad double arithmetic,
- speedups compensate for cost overhead.

pipeline model to staggering precision

D = double, DD = double double, QD = quad double



We may replace the “track” by “newton”.

web interface

Current beta version gives access to blackbox solver

- personalized account, signup via email exchange,
- management of solved polynomial systems.

Instead of calling `phc -b`, the user of the web interface ought to be able to run interactive sessions with `phcpy`.

For a scalable cloud service, a pool of workstations forms a compute server to help to solve the load of the web server.