Polynomial Homotopy Continuation with PHCpack*

Jan Verschelde
Department of Mathematics, Statistics, and Computer Science
University of Illinois at Chicago
851 South Morgan (M/C 249)
Chicago, IL 60607, USA
jan@math.uic.edu

ABSTRACT

PHCpack is a software package to solve polynomial systems via homotopy continuation methods. In the last twenty years (since the start of the development of PHCpack), the notion of solving a polynomial system by PHCpack evolved from approximating all isolated complex solutions for systems with as many equations as variables into providing tools for a numerical irreducible decomposition. This document provides the outline for a software demonstration highlighting recent additions to the software, such as accepting polynomials with negative exponents and sweeping for real points that lie isolated on complex solution curves.

Categories and Subject Descriptors

G.1.5 [**Roots of Nonlinear Equations**]: Continuation (homotopy) methods

Keywords

continuation, homotopy, Laurent polynomial system, numerical algebraic geometry, path following, sweep

1. INTRODUCTION

PHCpack is a software package that contains many homotopy methods to solve polynomial systems. A homotopy method is a method to define a family of systems, connecting the given system to an easier to solve system. Numerical continuation or path following methods then track the solutions from the easier system to the given system. The Littlewood-Richardson homotopies [19] that will be presented at the conference are recently added homotopy methods to PHCpack.

The first public release of the source code (version 1.0) of PHCpack is archived in [22]. In [16] is a description of the tools available in PHCpack to compute a numerical irreducible decomposition. In [21], 22 algorithms are explicitly listed as part of PHCpack. Interfaces to Maple and MATLAB (or Octave) are documented respectively in [11, 12] and [7]. The blackbox solver and path trackers of PHCpack are also available to Python scripts. Thanks to the efforts of William Stein and Marshall Hampton, PHCpack is one of the optional packages in Sage. Since version 2.3.52 (released last October), the distribution of PHCpack contains a Macaulay2 package (based on [10]) to access the Littlewood-Richardson homotopies [19].

*Date: 3 June 2010. Extended abstract for a software demonstration at ISSAC 2010. This material is based upon work supported by the National Science Foundation under Grant DMS-0713018.

2. RELATED SOFTWARE

Homotopy continuation methods for polynomial systems are implemented in various computer programs, listed in alphabetic order: Bertini [1, 2], CONSOL [14], HOM4PS-2.0 [9], HomLab [18, Appendix C], NAG4M2 [10], PHoM [8], POLSYS_PLP [26], and POLSYS_GLP [20]. Both POLSYS_PLP and POLSYS_GLP are part of the development of HOMPACK [24, 25].

Polyhedral homotopies need mixed volumes, computed by DEMiCs [13], Mixvol [4], and MixedVol [5].

3. SOLVING SYSTEMS OF LAURENT POLYNOMIALS

The blackbox solver of PHCpack is invoked via the option `-b` of the executable `phc`. This solver expects a system that has as many equations as unknowns. If the coefficients are sufficiently generic, then the mixed volume of the Newton polytopes of the polynomials in the system counts the number of isolated solutions without zero components.

For example, as input to `phc -b`, we create a file `input` with its content the next three lines:

```
2
 x*y + x^-1 + y^-1 - 1;
 x^-1*y + x*y^-1 + 1;
```

The system has exactly 6 isolated complex solutions and despite the particular choice of the coefficients, the mixed volume is an exact root count.

The improved performance of the blackbox solver (thanks to the integration of MixedVol [5] into PHCpack) led to the calculation of stable mixed volumes for polynomial systems. The stable mixed volume is not computed if the user submits a polynomial system with negative exponents, a so-called Laurent system. Broadening the range of acceptable inputs to Laurent systems is very useful to deal with systems of rational expressions. Such systems occur for example as the maximum likelihood equations in algebraic statistics. Instead of clearing denominators – leading to higher degree polynomials and spurious solutions – one can replace the denominators by new variables raised to negative powers. Polyhedral homotopies by default avoid the calculation of solutions with zero components so spurious solutions that make denominators vanish are in many cases avoided.

On computers with multiple cores, the blackbox solver will run faster when called with the option `-t` followed by the number of threads, e.g.: `phc -b -t4 input output` to solve a system in the file `input` with 4 threads. The results in the file `output` will be written faster, at most by a factor of four in this example. Implementation details about this recent feature will appear in [23].

4. NUMERICAL ALGEBRAIC GEOMETRY

The basic data structure manipulated by PHCpack is a pair of lists: a list of polynomials and a list of solutions. At first this format seems limited to isolated solutions, although by the use of slack variables, this pair also stores a numerical representation for a positive dimensional solution set, a so-called witness set in numerical algebraic geometry [17],[18].

We can divide the algorithms in two classes:

1. Given a polynomial system, compute witness sets for solution sets at various dimensions.
2. Given a witness set for a pure dimensional solution set, factor the set into irreducible components.

In symbolic computation, the computation of the dimension of an algebraic set appears in [6]. In [3], algorithms for absolute factorization are explained.

As running example, we save the following lines

```
3
 (x^2 - y)*(x - 0.1);
 (x^3 - z)*(y - 0.3);
 (x*y - z)*(z - 0.5);
```

in the file `ex1`. Because the polynomials occur in factored form, we recognize the twisted cubic (x, x^2, x^3) for any x, along with an isolated point `(0.1,0.3,0.5)`. But there are more isolated solutions to this system. Running the blackbox solver directly on this system will still give all isolated solutions as well-conditioned points but `phc -b` will not give a numerical representation for the twisted cubic.

Solving the system in the file `ex1` now becomes computing a numerical irreducible decomposition. In the top-down approach, this happens in three stages:

1. `phc -c, menu option #0`: run a cascade

   After entering names of input and output file, the user is prompted for the top dimension to start the cascade of homotopies. As we know the top dimension to be one for our example, we enter 1 so the program will add one hyperplane with random complex coefficients to compute generic points on the twisted cubic. If we choose `ex1out` as name of the output file, the three generic points on the twisted cubic will be written to the file `ex1out_sw1`. Candidate isolated points are in the output file `ex1out_sw0`. The suffix `_sw` stands for super witness set as the solution lists may contain junk.

2. `phc -f, menu option #2`: breakup a witness set

   We give the file `ex1out_sw1` as input to `phc -f` after selecting the second option on the menu. Because 1 is the top dimension in our example, we know that all points on file are generic. Factoring this pure dimensional solution set into irreducible components is done by partitioning the witness set into subsets of generic points for each irreducible component. The homotopies in PHCpack view the algebraic curves as Riemann surfaces and exploit monodromy to group generic points along irreducible factors. Monodromy loops are certified by linear traces. As the program finds loops to connect all generic points for our example, we conclude that the twisted cubic is irreducible.

3. `phc -f, menu option #1`: filter junk

   At the end of the cascade, there are nine solutions on file in `ex1out_sw0`. Some of these nine solutions will be classified as junk because they lie on the twisted cubic. To filter junk points, the user must provide a witness set for the twisted cubic (in the file `ex1out_sw1`) and the solutions in the file `ex1out_sw0`. The homotopy membership test will show that five out of the nine given solutions lie on the twisted cubic. So the system has four isolated roots in addition to the twisted cubic.

In the bottom-up approach, the equation-by-equation solver in PHCpack is available as `phc -a`. As alternative to stage 1 described above this solver has the advantage that is relieves the user of entering the top dimension.

5. SWEEPING FOR REAL SOLUTIONS

All homotopies used above are constructed so that, by the random choice of coefficients in the start system, with probability one singular solutions occur only at the end of the solution paths. However, many polynomial systems arising in practical applications have parameters and naturally one then wants to track solution paths extending the solution set from a particular instance of the parameters to some range. With these natural parameter homotopies, one can no longer guarantee the absence of singularities along a path.

Another recent addition to PHCpack is a sweeping homotopy [15] to locate real points (properly isolated as real points) on complex algebraic curves. As in most applications, users are mainly interested in real solutions, but real solutions which appear as isolated real points on positive dimensional solution sets require special types of homotopies.

For example, the input file to `phc -p` starts with

```
4 5
 x1*x2^2 + x1*x3^2 - A*x1 + 1;
 x2*x1^2 + x2*x3^2 - A*x2 + 1;
 x3*x1^2 + x3*x2^2 - A*x3 + 1;
 (1-t)*(A-0.1) + t*(A+0.1);
```

where `A` is a natural parameter and `t` is an artificial parameter. As `t` goes from 0 to 1, we will sweep the parameter `A` from $+0.1$ to -0.1. The input file also contains the 21 solutions for the value at `t` equal to zero.

As result of the sweep, the tracking stops at `t` equal to 0.5 for four of the 21 solution paths because at $\mathtt{t} = 0.5$, there is a quadruple root. Three paths start to diverge to infinity as `t` approaches 0.5.

6. REFERENCES

[1] D.J. Bates, J.D. Hauenstein, A.J. Sommese, and C.W. Wampler. Bertini: Software for numerical algebraic geometry. Available at `http://www.nd.edu/~sommese/bertini/`.

[2] D.J. Bates, J.D. Hauenstein, A.J. Sommese, and C.W. Wampler. Software for numerical algebraic geometry: a paradigm and progress towards its implementation. In M.E. Stillman, N. Takayama, and J. Verschelde, editors, *Software for Algebraic Geometry*, volume 148 of *The IMA Volumes in Mathematics and its Applications*, pages 1–14. Springer-Verlag, 2008.

[3] G. Chèze and A. Galligo. Four lectures on polynomial absolute factorization. In A. Dickenstein and I.Z. Emiris, editors, *Solving Polynomial Equations. Foundations, Algorithms and Applications*, volume 14 of *Algorithms and Computation in Mathematics*, pages 339–394. Springer-Verlag, 2005.

[4] I.Z. Emiris and J.F. Canny. Efficient incremental algorithms for the sparse resultant and the mixed volume. *J. Symbolic Computation*, 20(2):117–149, 1995. Software at `http://cgi.di.uoa.gr/~emiris/soft_geo.html`.

[5] T. Gao, T.Y. Li, and M. Wu. Algorithm 846: MixedVol: a software package for mixed-volume computation. *ACM Trans. Math. Softw.*, 31(4):555–560, 2005.

[6] M. Giusti, K. Hägele, G. Lecerf, J. Marchand, and B. Salvy. The Projective Noether Maple Package: computing the dimension of a projective variety. *J. Symbolic Computation*, 30(3):291–307, 2000.

[7] Y. Guan and J. Verschelde. PHClab: A MATLAB/Octave interface to PHCpack. In M.E. Stillman, N. Takayama, and J. Verschelde, editors, *Software for Algebraic Geometry*, volume 148 of *The IMA Volumes in Mathematics and its Applications*, pages 15–32. Springer-Verlag, 2008.

[8] T. Gunji, S. Kim, M. Kojima, A. Takeda, K. Fujisawa, and T. Mizutani. PHoM – a polyhedral homotopy continuation method for polynomial systems. *Computing*, 73(4):55–77, 2004.

[9] T.L. Lee, T.Y. Li, and C.H. Tsai. HOM4PS-2.0: a software package for solving polynomial systems by the polyhedral homotopy continuation method. *Computing*, 83(2-3):109–133, 2008. Software at `http://www.math.msu.edu/~li/Software.htm`.

[10] A. Leykin. Numerical algebraic geometry for Macaulay2. `arXiv:0911.1783v1 [math.AG]`.

[11] A. Leykin and J. Verschelde. PHCmaple: A Maple interface to the numerical homotopy algorithms in PHCpack. In Quoc-Nam Tran, editor, *Proceedings of the Tenth International Conference on Applications of Computer Algebra (ACA'2004)*, pages 139–147, 2004.

[12] A. Leykin and J. Verschelde. Interfacing with the numerical homotopy algorithms in PHCpack. In N. Takayama and A. Iglesias, editors, *Proceedings of ICMS 2006*, volume 4151 of *LNCS*, pages 354–360. Springer-Verlag, 2006.

[13] T. Mizutani and A. Takeda. DEMiCs: A software package for computing the mixed volume via dynamic enumeration of all mixed cells. In M.E. Stillman, N. Takayama, and J. Verschelde, editors, *Software for Algebraic Geometry*, volume 148 of *The IMA Volumes in Mathematics and its Applications*, pages 59–79. Springer-Verlag, 2008. Software at `http://www.is.titech.ac.jp/~mizutan8/DEMiCs`.

[14] A. Morgan. *Solving polynomial systems using continuation for engineering and scientific problems.* Prentice-Hall, 1987. Volume 57 of Classics in Applied Mathematics Series, SIAM 2009. Pages with code at `http://www.siam.org/books/cl57`.

[15] K. Piret and J. Verschelde. Sweeping algebraic curves for singular solutions. *Journal of Computational and Applied Mathematics*, 234(4):1228–1237, 2010.

[16] A.J. Sommese, J. Verschelde, and C.W. Wampler. Numerical irreducible decomposition using PHCpack. In M. Joswig and N. Takayama, editors, *Algebra, Geometry, and Software Systems*, pages 109–130. Springer-Verlag, 2003.

[17] A.J. Sommese, J. Verschelde, and C.W. Wampler. Introduction to numerical algebraic geometry. In A. Dickenstein and I.Z. Emiris, editors, *Solving Polynomial Equations. Foundations, Algorithms and Applications*, volume 14 of *Algorithms and Computation in Mathematics*, pages 301–337. Springer-Verlag, 2005.

[18] A.J. Sommese and C.W. Wampler. *The Numerical solution of systems of polynomials arising in engineering and science*. World Scientific Press, Singapore, 2005.

[19] F. Sottile, R. Vakil, and J. Verschelde. Solving Schubert problems with Littlewood-Richardson Homotopies. To appear in the ISSAC'10 proceedings.

[20] H.-J. Su, J.M. McCarthy, M. Sosonkina, and L.T. Watson. Algorithm 857: POLSYS_GLP: A parallel general linear product homotopy code for solving polynomial systems of equations. *ACM Trans. Math. Softw.*, 32(4):561–579, 2006.

[21] J. Verschelde. Homotopy Methods for Solving Polynomial Systems. Notes for a tutorial at ISSAC'05, Beijing, China, 24 July 2005. Available at `http://www.math.uic.edu/~jan/preprints`.

[22] J. Verschelde. Algorithm 795: PHCpack: A general-purpose solver for polynomial systems by homotopy continuation. *ACM Trans. Math. Softw.*, 25(2):251–276, 1999. Software at `http://www.math.uic.edu/~jan/download`.

[23] J. Verschelde and G. Yoffe. Polynomial homotopies on multicore workstations. Accepted for publication in the proceedings of PASCO 2010.

[24] L.T. Watson, S.C. Billups, and A.P. Morgan. Algorithm 652: HOMPACK: a suite of codes for globally convergent homotopy algorithms. *ACM Trans. Math. Softw.*, 13(3):281–310, 1987.

[25] L.T. Watson, M. Sosonkina, R.C. Melville, A.P. Morgan, and H.F. Walker. HOMPACK90: A suite of Fortran 90 codes for globally convergent homotopy algorithms. *ACM Trans. Math. Softw.*, 23(4):514–549, 1997.

[26] S.M. Wise, A.J. Sommese, and L.T. Watson. Algorithm 801: POLSYS_PLP: a partitioned linear product homotopy code for solving polynomial systems of equations. *ACM Trans. Math. Softw.*, 26(1):176–200, 2000.