A `sympy/sage` Module for Computing Polynomial Remainder Sequences

Alkiviadis G. Akritas[*] Gennadi I. Malaschonok[†] Panagiotis S. Vigklas[‡]

March 2, 2017

Extended Abstract

Given the polynomials $f, g \in \mathbb{Z}[x]$, we are interested in the following four polynomial remainder sequences (prs's):

(a) Euclidean prs,

(b) Modified Euclidean prs,

(c) Subresultant prs, and

(d) Modified Subresultant prs.

The Modified Euclidean prs is obtained by modifying the sign of the remainder of each polynomial division performed for the computation of the Euclidean prs. Analogously, the Modified Subresultant prs is obtained by modifying the matrix from which the Subresultant prs is obtained.

Even though prs's (c) and (d) are computed by evaluating sub-determinants of given matrices, our objective is to compute *all four prs's* using the *same* type of polynomial divisions over the ring $\mathbb{Z}[\mathtt{x}]$.

Our objective is not at all trivial and has eluded the efforts of great mathematicians, as our brief review below indicates.

Initially, Collins, Brown and Traub [8], [9], [11], [12] used the so called `prem` pseudo-remainder function defined by

$$\mathrm{LC}(g)^{\delta} \cdot f = q \cdot g + h, \tag{1}$$

where LC(g) is the leading coefficient of the divisor g, and

$$\delta = \mathrm{degree}(f) - \mathrm{degree}(g) + 1. \tag{2}$$

[*] Department of Electrical and Computer Engineering, University of Thessaly, GR-38221, Volos, Greece, Tel.: +30 242110 74886, Fax: +30 24210 74997, akritas@uth.gr

[†] Laboratory for Algebraic Computations, Tambov State University Internatsionalnaya, 33, RU-392000 Tambov, Russia, malaschonok@gmail.com

[‡] Department of Electrical and Computer Engineering, University of Thessaly, GR-38221, Volos, Greece, pviglas@uth.gr

However, using `prem` *only* the signs of prs (c) can be *exactly* computed ([10], pp. 277–283). The signs of the other three prs's, (a), (b) and (d), *may* not be exactly computed when the prs is *incomplete*, i.e. when there are gaps in the degree sequence of the polynomial remainders.

Basu, Pollack, and Roy [7] employ the so called *signed* `prem` function defined by

$$\mathrm{LC}(g)^{\delta} \cdot f = q \cdot g + h, \tag{3}$$

whereby, if $\mathrm{mod}(\delta, 2) = 1$ they set it to $\delta = \delta + 1$. This way they are able to exactly compute the signs of prs's (b) and (d), which are, therefore, called *signed* prs's. The signs of the other two prs's, (a) and (c), *may* not be exactly computed, and are, hence, called *non-signed* prs's.

Instead, we employ the so called `rem_z` pseudo-remainder function defined by

$$|\,\mathrm{LC}(g)|^{\delta} \cdot f = q \cdot g + h \tag{4}$$

and are able to exactly compute the signs of *all four* prs's. Moreover, we have shown that these four prs's are related as shown in Figure 1.

$$\begin{tikzcd}
\textit{Euclidean PRS} \arrow[r, Leftrightarrow, "\mathcal{SAM}"'] \arrow[d, Leftrightarrow, "\mathcal{AMV}\text{-2015}"'] & \textit{modified Euclidean PRS} \arrow[d, Leftrightarrow, "\mathcal{PG}\text{-1917}"] \\
\textit{Subresultant PRS} \arrow[r, Leftrightarrow, "\mathcal{AMV}\text{-2015}"'] \arrow[ur, Leftrightarrow, "\mathcal{AMV}\text{-2015}"] & \textit{modified Subresultant PRS} \\
\textit{sylvester}1\ (1840) \arrow[u, Rightarrow] & \textit{sylvester}2\ (1853) \arrow[u, Rightarrow]
\end{tikzcd}$$

Figure 1: The double ended arrows indicate one-to-one correspondences that exist between the coefficients of the polynomials in the respective nodes. The labels indicate those who first established the correspondences and when. Two different matrices by Sylvester are used [16], [17].

In our work [1] – [6] — which relies heavily on the work by Pell and Gordon [15] — we have shown that *all four prs's* are *signed*, i.e. their signs are *uniquely* defined. To wit, the signs of the prs's computed in $\mathbb{Z}\,[x]$ are identical to those computed in $\mathbb{Q}\,[x]$.

Moreover, we have developed the `sympy/sage` module `subresultants_qq_zz.py`[1] for exactly computing the signs of all four prs's of Figure 1, employing the so called `rem_z` pseudo-remainder function defined in (4).

Our talk will focus on the functions included in this module — filling thus a vacuum in the educational process. Namely, when people teach about prs's in general — and subresultant prs's in particular — they would have a module to work with in order to compute the sequences with their correct signs. Otherwise they would have to say that they compute the sequences "up to sign" ([13], p. 182) & ([14], Example 4.7).

[1] `https://github.com/sympy/sympy/blob/master/sympy/polys/subresultants_qq_zz.py.`

References

[1] Akritas, A. G. A Simple Proof of the Validity of the Reduced PRS Algorithm. *Computing*, **38**, (1987), 369–372.

[2] Akritas, A. G., G. I. Malaschonok, P. S. Vigklas On a Theorem by Van Vleck Regarding Sturm Sequences. *Serdica Journal of Computing*, **7**(4), 101–134, 2013.

[3] Akritas, A. G., G. I. Malaschonok, P. S. Vigklas Sturm Sequences and Modified Subresultant Polynomial Remainder Sequences. *Serdica Journal of Computing*, **8**(1), 29–46, 2014.

[4] Akritas, A. G. Three New Methods for Computing Subresultant Polynomial Remainder Sequences (PRS's). *Serdica Journal of Computing*, *Serdica Journal of Computing*, **9**(1) (2015), 1–26.

[5] Akritas, A. G., G. I. Malaschonok, P. S. Vigklas On the Remainders Obtained in Finding the Greatest Common Divisor of Two Polynomials. *Serdica Journal of Computing*, **9**(2) (2015), 123–138.

[6] Akritas, A. G., G. I. Malaschonok, P. S. Vigklas A Basic Result on the Theory of Subresultants. *Serdica Journal of Computing*, to appear.

[7] Basu, S., R. Pollack, M. F. Roy *Algorithms in Real Algebraic Geometry*, 2nd Edition, Springer, 2006.

[8] Brown, W. S. The subresultant PRS Algorithm. *ACM Transactions on Mathematical Software*, **4**(3), (1978), 237–249.

[9] Brown, W. S., J. F. Traub On Euclid's Algorithm and the Theory of Subresultants. *Journal of the Association for Computing Machinery*, **18**, (1971), 505–514.

[10] Cohen, J. E. *Computer Algebra and Symbolic Computation – Mathematical Methods.* A.K. Peters, Massachusetts, (2003).

[11] Collins, G. E. Polynomial Remainder Sequences and Determinants. *American Mathematical Monthly*, **73**(7), (1966), 708–712.

[12] Collins, G. E. Subresultants and Reduced Polynomial Remainder Sequences. *Journal of the Association for Computing Machinery*, **14**, (1967), 128–142.

[13] von zur Gathen, J., J. Gerhard *Modern Computer Algebra.* Cambridge University Press, (1999).

[14] von zur Gathen, J., T. Lücking Subresultants Revisited. *Theoretical Computer Science*, **297**(1-3), (2003), 199–239.

[15] Pell, A. J., R. L. Gordon The Modified Remainders Obtained in Finding the Highest Common Factor of Two Polynomials. *Annals of Mathematics*, Second Series, **18**(4), (Jun., 1917), 188–193.

[16] Sylvester, J. J. A method of determining by mere inspection the derivatives from two equations of any degree. Philosophical Magazine, **16**, (1840), 132–135.

[17] Sylvester, J. J. On the Theory of Syzygetic Relations of Two Rational Integral Functions, Comprising an Application to the Theory of Sturm's Functions, and that of the Greatest Algebraical Common Measure. *Philosophical Transactions*, **143**, (1853), 407–548.