

# Uncovering Symmetries in Polynomial Systems

Viktor Larsson<sup>(✉)</sup> and Kalle Åström

Lund University, Lund, Sweden  
{viktor1,kalle}@maths.lth.se

**Abstract.** In this paper we study symmetries in polynomial equation systems and how they can be integrated into the action matrix method. The main contribution is a generalization of the partial  $p$ -fold symmetry and we provide new theoretical insights as to why these methods work. We show several examples of how to use this symmetry to construct more compact polynomial solvers. As a second contribution we present a simple and automatic method for finding these symmetries for a given problem. Finally we show two examples where these symmetries occur in real applications.

## 1 Introduction

Polynomial systems have become an integral part of Computer Vision due to their ability to encode many geometric constraints. Polynomial solvers have been successfully used for many minimal problem such as absolute pose estimation [1–3], relative pose estimation [4–6] and homography estimation [7, 8]. They have also been used for some non-minimal problem such as PnP [9, 10].

One technique for constructing polynomial solvers is the so called *action matrix* method. The method reduces the polynomial system to an eigenvalue problem for which there exist good numerical methods. For a brief introduction to polynomial solvers in Computer Vision we recommend Byröd et al. [11].

In [12] Ask et al. considers polynomial systems, where the degree of each monomial has the same remainder modulo  $p$ . This introduces a  $p$ -fold symmetry into the solution set. By taking this symmetry into account they construct smaller and more stable polynomial solvers. This work was later extended by Kuang et al. [13] to polynomial systems where this symmetry only exists in a subset of the variables. This type of symmetry has been used in [1, 9, 14–16].

In this paper we generalize the symmetry from [12, 13] and provide new theoretical insight as to why these methods work. We show that if the system has these symmetries the action matrix can be chosen block diagonal and by considering only a single block we can construct more compact solvers.

In [17] Corless et al. also use symmetries in the action matrix method. Their approach is based on studying the group structure of the symmetry using tools from linear representation theory. While the paper present theory for symmetries

**Electronic supplementary material** The online version of this chapter (doi:[10.1007/978-3-319-46487-9\\_16](https://doi.org/10.1007/978-3-319-46487-9_16)) contains supplementary material, which is available to authorized users.

of general group structure and show a few examples with integer or rational coefficients it is not clear how to directly apply it to build polynomial solvers for types of problems encountered in Computer Vision. In contrast the theory developed in this paper classifies the symmetry based on the monomial structure in the equations. This allows the method to integrate naturally into the standard methods for building polynomial solvers.

The main contributions in this paper are:

- We generalize the partial  $p$ -fold symmetry studied in [12, 13] and show how to exploit this symmetry to construct more compact polynomial solvers.
- We present a simple and automatic method for finding these symmetries in polynomial systems.
- We show two examples where these symmetries occur in the polynomial systems from real applications.

### 1.1 Background and Notation

The set of all polynomials over  $\mathbb{C}$  is denoted  $\mathbb{C}[X]$ . The solutions  $\mathcal{V} \subset \mathbb{C}^n$  to a set of polynomial equations  $f_i(\mathbf{x}) = 0, i = 1, 2, \dots, n$  is called an *affine variety*. The set of all polynomial combinations of  $f_i$ , i.e.  $I = \{\sum_i h_i(\mathbf{x})f_i(\mathbf{x}) \mid h_i \in \mathbb{C}[X]\}$  defines an ideal in the polynomial ring  $\mathbb{C}[X]$ . For an ideal  $I$  we can define the quotient ring  $\mathbb{C}[X]/I$ , which consists of equivalence classes with respect to  $I$ , i.e. two elements are equivalent if their difference lies in  $I$ . From algebraic geometry it is well known that if the set of solutions to a system is finite then the corresponding quotient ring  $\mathbb{C}[X]/I$  is a finite dimensional vectorspace. For an introduction to algebraic geometry we recommend [18].

Throughout the paper we use the multi-index notation for monomials, i.e.

$$\mathbf{x}^\alpha = \mathbf{x}^{(\alpha_1, \dots, \alpha_n)} = \prod_k x_k^{\alpha_k}, \quad \text{so e.g. } \mathbf{x}^{(2,0,1)} = x_1^2 x_3.$$

## 2 Symmetries in Polynomial Equation Systems

We start with a simple example.

*Example 1.* Consider the following system of polynomial equations

$$\begin{cases} x^2 + y - 2 = 0, \\ x^2 y^2 - 1 = 0. \end{cases} \tag{1}$$

The system has six solutions given by

$$(\pm 1, 1), (\pm \varphi, -\varphi^{-1}), (\pm \varphi^{-1}, \varphi) \quad \text{where} \quad \varphi = \frac{1 + \sqrt{5}}{2}. \tag{2}$$

Since each monomial has the  $x$ -variable raised to an even power, we can for any solution flip the sign of  $x$  and get another solution. This type of symmetry was studied in [12, 13] and is characterized in the following definition.

**Definition 1.** The polynomial  $f(\mathbf{x}, \mathbf{y})$  has a **partial  $p$ -fold symmetry** in  $\mathbf{x}$  if the sum of the exponents for  $\mathbf{x}$  of each monomial has the same remainder  $q$  modulo  $p$ , i.e.

$$f(\mathbf{x}, \mathbf{y}) = \sum_k a_k \mathbf{x}^{\alpha_k} \mathbf{y}^{\beta_k} \implies q \equiv \mathbf{1}^T \alpha_k \pmod p \quad \forall k. \tag{3}$$

In [13] it was shown that if we have a system of polynomials with this property the solution set will also have a  $p$ -fold symmetry. More specifically if  $\mathcal{V}$  is the set of solutions then

$$(\mathbf{x}, \mathbf{y}) \in \mathcal{V} \implies (e^{2\pi i \frac{k}{p}} \mathbf{x}, \mathbf{y}) \in \mathcal{V} \quad k = 0, 1, 2, \dots, p - 1. \tag{4}$$

*Example 2.* The polynomial system

$$\begin{cases} x^3 - 1 = 0, \\ xy - 1 = 0 \end{cases} \tag{5}$$

has three solutions given by

$$\mathcal{V} = \left\{ (1, 1), \left( \frac{1 + i\sqrt{3}}{2}, \frac{1 - i\sqrt{3}}{2} \right), \left( \frac{1 - i\sqrt{3}}{2}, \frac{1 + i\sqrt{3}}{2} \right) \right\}. \tag{6}$$

While this system does not have any partial  $p$ -fold symmetries, the solution set has the following property

$$(x, y) \in \mathcal{V} \implies (e^{2\pi i \frac{1}{3}} x, e^{2\pi i \frac{2}{3}} y) \in \mathcal{V}, \tag{7}$$

which is similar to that in (4). In this work we consider a generalization of the partial  $p$ -fold symmetry characterized by the following definition.

**Definition 2.** The polynomial  $f(\mathbf{x})$  has a **weighted  $p$ -fold symmetry** with weights  $\mathbf{c} \in \mathbb{Z}_p^n$  if the  $\mathbf{c}$ -weighted sum of the exponents for  $\mathbf{x}$  of each monomial has the same remainder  $q$  modulo  $p$ , i.e.

$$f(\mathbf{x}) = \sum_k a_k \mathbf{x}^{\alpha_k} \implies q \equiv \mathbf{c}^T \alpha_k \pmod p \quad \forall k. \tag{8}$$

*Example 3.* Below are three examples

$$\begin{aligned} f_1(x, y) &= x^3 - x^2y^2 + y^3, & p &= 3, & \mathbf{c} &= (2, 1), \\ f_2(x, y) &= x^5 + x^3y + x, & p &= 4, & \mathbf{c} &= (1, 2), \\ f_3(x, y, z) &= x + y^2 + yz - 1, & p &= 2, & \mathbf{c} &= (0, 1, 1). \end{aligned}$$

Note that the vector  $\mathbf{c}$  is not unique, e.g. for the first polynomial  $\mathbf{c} = (1, 2)$  would also work. If  $p$  is not prime then for any factor of  $p$  we also have a symmetry, e.g. the polynomial  $f_2$  also has a 2-fold symmetry. From the last example it becomes clear that the partial  $p$ -fold symmetry from [13] is a special case of the weighted symmetry where the weights are binary, corresponding to the symmetry variables.

Similarly to (4) we will see that for any polynomial equation system with  $\mathbf{c}$ -weighted  $p$ -fold symmetry, the solution set has a corresponding symmetry. For any vector  $\mathbf{c} \in \mathbb{Z}_p^n$  we define the matrix

$$D_p^{\mathbf{c}} = \text{diag} \left( \left\{ \exp(2\pi i \frac{c_k}{p}) \right\}_{k=1}^n \right). \tag{9}$$

**Definition 3.** A set  $\mathcal{V} \subset \mathbb{C}^n$  has a  $\mathbf{c}$ -weighted  $p$ -fold symmetry if and only if it is stable under  $D_p^{\mathbf{c}}$ , i.e.

$$D_p^{\mathbf{c}} \mathcal{V} \subset \mathcal{V}. \tag{10}$$

The following two theorems are directly adapted from Kuang et al. [13] where they are proved for regular partial  $p$ -fold symmetry.

**Theorem 1.** Let  $f_i(\mathbf{x}) = 0, i = 1, 2, \dots, m$  be a polynomial system and denote the set of solutions  $\mathcal{V} \subset \mathbb{C}^n$ . If each  $f_i$  has a  $\mathbf{c}$ -weighted  $p$ -fold symmetry then so does the set of solutions  $\mathcal{V}$ .

*Proof.* Take any  $\mathbf{x} \in \mathcal{V}$ . Consider some  $f_i$  and let  $q$  be the remainder from Definition 2. Let  $\mathbf{x}^\beta$  be any monomial from  $f_i$  and consider the effect of  $D_p^{\mathbf{c}}$ ,

$$(D_p^{\mathbf{c}} \mathbf{x})^\beta = \prod_k (e^{2\pi i \frac{c_k}{p}} x_k)^{\beta_k} = \prod_k e^{2\pi i \frac{c_k \beta_k}{p}} x_k^{\beta_k} = e^{2\pi i \frac{\mathbf{c}^T \boldsymbol{\beta}}{p}} \mathbf{x}^\beta = e^{2\pi i \frac{q}{p}} \mathbf{x}^\beta. \tag{11}$$

Then if  $f_i(\mathbf{x}) = \sum_k a_k \mathbf{x}^{\alpha_k}$  we have

$$f_i(D_p^{\mathbf{c}} \mathbf{x}) = \sum_k a_k (D_p^{\mathbf{c}} \mathbf{x})^{\alpha_k} = \sum_k a_k e^{2\pi i \frac{q}{p}} \mathbf{x}^{\alpha_k} = e^{2\pi i \frac{q}{p}} f_i(\mathbf{x}) = 0, \tag{12}$$

and since this holds for any  $i = 1, 2, \dots, m$  we must have that  $D_p^{\mathbf{c}} \mathbf{x} \in \mathcal{V}$ .  $\square$

**Theorem 2.** Let  $f_i(\mathbf{x}) = 0, i = 1, 2, \dots, m$  be a polynomial system where the solution set  $\mathcal{V}$  has a  $\mathbf{c}$ -weighted  $p$ -fold symmetry. Then there exist an equivalent system where each polynomial has  $\mathbf{c}$ -weighted  $p$ -fold symmetry.

*Proof.* Let  $\mathbf{x} \in \mathcal{V}$ . Then for any  $f_i$  we have

$$f_i((D_p^{\mathbf{c}})^k \mathbf{x}) = 0, \quad k = 0, 1, 2, \dots, p - 1. \tag{13}$$

Decompose  $f_i$  into  $f_i(\mathbf{x}) = g_0(\mathbf{x}) + g_1(\mathbf{x}) + \dots + g_{p-1}(\mathbf{x})$  such that each monomial in  $g_q$  has the  $\mathbf{c}$ -weighted remainder  $q$  modulo  $p$ , i.e.

$$g_q(\mathbf{x}) = \sum_k a_k \mathbf{x}^{\gamma_k} \implies \mathbf{c}^T \boldsymbol{\gamma}_k \equiv q \pmod{p}. \tag{14}$$

Then if we denote  $\omega = e^{2\pi i \frac{1}{p}}$  we have

$$f_i(\mathbf{x}) = g_0(\mathbf{x}) + g_1(\mathbf{x}) + \dots + g_{p-1}(\mathbf{x}), \tag{15}$$

$$f_i(D_p^{\mathbf{c}} \mathbf{x}) = g_0(\mathbf{x}) + \omega g_1(\mathbf{x}) + \dots + \omega^{p-1} g_{p-1}(\mathbf{x}), \tag{16}$$

...

$$f_i((D_p^{\mathbf{c}})^{p-1} \mathbf{x}) = g_0(\mathbf{x}) + \omega^{p-1} g_1(\mathbf{x}) + \dots + \omega g_{p-1}(\mathbf{x}). \tag{17}$$

Since each  $f_i((D_p^c)^k \mathbf{x}) = 0$  we can rewrite this as

$$\begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^2 & \dots & \omega^{p-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{p-1} & \omega^{p-2} & \dots & \omega \end{bmatrix} \begin{bmatrix} g_0(\mathbf{x}) \\ g_1(\mathbf{x}) \\ g_2(\mathbf{x}) \\ \vdots \\ g_{p-1}(\mathbf{x}) \end{bmatrix} = \mathbf{0}. \tag{18}$$

Since the matrix is non-singular we must have  $g_k(\mathbf{x}) = 0$  for  $k = 0, 1, 2, \dots, p-1$ . By definition each  $g_k$  will have a  $\mathbf{c}$ -weighted symmetry and by replacing each equation  $f_i(\mathbf{x}) = 0$  by the equations  $\{g_k(\mathbf{x}) = 0\}_{k=0}^{p-1}$  it is clear that we get an equivalent system where each polynomial has the correct symmetry.  $\square$

**Corollary 1.** *Take any polynomial  $f \in I = \langle f_1, f_2, \dots, f_m \rangle$  where each  $f_i$  has  $\mathbf{c}$ -weighted  $p$ -fold symmetry and decompose*

$$f(\mathbf{x}) = g_0(\mathbf{x}) + g_1(\mathbf{x}) + \dots + g_{p-1}(\mathbf{x}), \tag{19}$$

such that each monomial in  $g_q$  has the  $\mathbf{c}$ -weighted remainder  $q$  modulo  $p$ , then each component  $g_q$  is a polynomial in  $I$ .

*Proof.* This is a consequence of the proof of the previous theorem.  $\square$

**Theorem 3.** *Let  $f_i(\mathbf{x}) = 0, i = 1, 2, \dots, m$  be a polynomial system with  $\mathbf{c}$ -weighted  $p$ -fold symmetry and  $\mathcal{B}$  be a monomial (linear) basis for  $\mathbb{C}[X]/I$ . Let  $\mathcal{B}_q \subset \mathcal{B}$  be the set of basis monomials with  $\mathbf{c}$ -weighted exponent remainder  $q$  modulo  $p$ . Then any element  $[h(\mathbf{x})] \in \mathbb{C}[X]/I$  where  $h(\mathbf{x})$  has a  $\mathbf{c}$ -weighted symmetry can be expressed in the quotient ring as a linear combination of the basis elements with the same remainder, i.e.*

$$[h(\mathbf{x})] \in [\text{span } \mathcal{B}_q] \tag{20}$$

for some  $q$ .

*Proof.* Since  $\mathcal{B}$  is a linear basis for  $\mathbb{C}[X]/I$  there exist coefficients  $a_k$  such that

$$[h(\mathbf{x})] = \left[ \sum_k a_k b_k(\mathbf{x}) \right] \quad \text{where } b_k \in \mathcal{B}. \tag{21}$$

Since the ideal vanishes on  $\mathcal{V}$  we have

$$h(\mathbf{x}) = \sum_k a_k b_k(\mathbf{x}) \quad \mathbf{x} \in \mathcal{V}. \tag{22}$$

But this means that  $(h(\mathbf{x}) - \sum_k a_k b_k(\mathbf{x})) \in I$  and from Corollary 1 we know that we can split this into  $p$  terms with different  $\mathbf{c}$ -weighted remainders, which all belong to  $I$ . In particular if  $h(\mathbf{x})$  has  $\mathbf{c}$ -weighted remainder  $q$  we have,

$$h(\mathbf{x}) = \sum_{b_k \in \mathcal{B}_q} a_k b_k(\mathbf{x}) \quad \mathbf{x} \in \mathcal{V}, \tag{23}$$

which is sufficient to show  $[h(\mathbf{x})] = \left[ \sum_{b_k \in \mathcal{B}_q} a_k b_k(\mathbf{x}) \right]$  since the elements in  $\mathbb{C}[X]/I$  are uniquely determined by their values on  $\mathcal{V}$ .  $\square$

**Corollary 2.** *For any action polynomial  $a(\mathbf{x}) \in \mathbb{C}[X]$  with  $\mathbf{c}$ -weighted remainder zero the corresponding action matrix becomes block diagonal.*

*Proof.* If  $a(\mathbf{x})$  has  $\mathbf{c}$ -weighted remainder zero then all polynomials in  $a(\mathbf{x})\mathcal{B}_q$  will have remainder  $q$ . From Theorem 3 we get  $[a(\mathbf{x})\mathcal{B}_q] \subset [\text{span } \mathcal{B}_q]$  and the result follows.

### 2.1 Solving Equation Systems with Symmetries

Once the symmetries have been identified they can be used to construct more compact polynomial solvers. From Corollary 2 we know that if we choose our action polynomial  $a(\mathbf{x}) \in \mathbb{C}[X]$  to be invariant with respect to the symmetry the corresponding action matrix will be block diagonal. The idea is then to only consider a single block of the matrix, i.e. we only consider the action of  $a(\mathbf{x})$  on a subset of the basis monomials  $\mathcal{B}$ . Next we show two concrete examples where we construct the partial action matrix and use it to recover the solutions.

*Example 4.* Consider again the system in Example 1,

$$\begin{cases} x^2 + y - 2 = 0, \\ x^2 y^2 - 1 = 0. \end{cases} \tag{24}$$

We saw earlier that this system has six solutions and a 2-fold partial symmetry in the  $x$  variable, or equivalently a  $(1, 0)$ -weighted 2-fold symmetry. For this system the quotient ring  $\mathbb{C}[X]/I$  is spanned by the monomials

$$\mathcal{B} = \{1, x, y, xy, y^2, xy^2\}. \tag{25}$$

We can group these into two sets, based on their  $(1, 0)$ -weighted remainder modulo 2,

$$\mathcal{B}_0 = \{1, y, y^2\} \quad \text{and} \quad \mathcal{B}_1 = \{x, xy, xy^2\}. \tag{26}$$

Now instead of working with the entire basis  $\mathcal{B}$  we will only consider the subset  $\mathcal{B}_0$ . If we choose  $x^2$  to be our action polynomial (note that this has  $(1, 0)$ -remainder zero) we have the following multiplication maps <sup>1</sup>

$$T_{x^2}[1] = x^2 = 2 - y, \quad T_{x^2}[y] = x^2 y, \quad T_{x^2}[y^2] = x^2 y^2 = 1. \tag{27}$$

Since one of the monomials was not in the span of  $\mathcal{B}_0$  we need to generate more equations. Multiplying the first equation by  $y$  we get

$$x^2 y + y^2 - 2y = 0 \implies T_{x^2}[y] = x^2 y = 2y - y^2 \in \text{span } \mathcal{B}_0. \tag{28}$$

---

<sup>1</sup>  $T_\alpha : \mathbb{C}[X]/I \rightarrow \mathbb{C}[X]/I$  is the linear map corresponding to multiplication by  $\alpha(\mathbf{x})$ .

Finally we can construct our action matrix,

$$\begin{bmatrix} 0 & 0 & 1 \\ -1 & 2 & 0 \\ 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} y^2 \\ y \\ 1 \end{bmatrix} = x^2 \begin{bmatrix} y^2 \\ y \\ 1 \end{bmatrix}. \tag{29}$$

Even though the system has six solutions we only have to solve a  $3 \times 3$  eigenvalue problem. From each eigenvector we can construct two solutions with different signs for  $x$ .

Next we show a similar example but where the equation system has multiple symmetries.

*Example 5.* Consider the equation system

$$\begin{cases} x^2 + y^2 - 2 = 0, \\ xy^2 - x = 0. \end{cases} \tag{30}$$

This system has a 2-fold partial symmetry in the  $x$  variable and 2-fold partial symmetry in  $y$ , or equivalently two 2-fold symmetries with weights  $\mathbf{c}_1 = (1, 0)$  and  $\mathbf{c}_2 = (0, 1)$ . The equation system has six solutions and a basis for the quotient ring  $\mathbb{C}[X]/I$  is given by

$$\mathcal{B} = \{1, x, y, xy, y^2, y^3\}. \tag{31}$$

Grouping the basis monomials based on their  $\mathbf{c}_k$ -weighted remainders modulo 2:

$$\mathcal{B}_{0,0} = \{1, y^2\}, \mathcal{B}_{0,1} = \{y, y^3\}, \mathcal{B}_{1,0} = \{x\}, \mathcal{B}_{1,1} = \{xy\}. \tag{32}$$

Let us choose to work with  $\mathcal{B}_{0,0}$ . By multiplying the second equation with  $x$  we get that all the monomials in the system have the same remainder as our monomial basis,

$$\begin{cases} x^2 + y^2 - 2 = 0, \\ x^2y^2 - x^2 = 0. \end{cases} \tag{33}$$

Choosing again the action polynomial as  $x^2$  we get the following multiplications

$$T_{x^2}[1] = x^2 = 2 - y^2 \in \text{span } \mathcal{B}_{0,0}, \quad T_{x^2}[y^2] = x^2y^2 = x^2 = 2 - y^2 \in \text{span } \mathcal{B}_{0,0}, \tag{34}$$

which allows us to construct a  $2 \times 2$  action matrix

$$\begin{bmatrix} -1 & 2 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} y^2 \\ 1 \end{bmatrix} = x^2 \begin{bmatrix} y^2 \\ 1 \end{bmatrix}. \tag{35}$$

This matrix has eigenvalues 0 and 1 with corresponding eigenvectors  $\begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ .

The two eigenpairs give us the following possibilities

$$\begin{cases} y^2 = 2 \\ x^2 = 0 \end{cases} \quad \text{and} \quad \begin{cases} y^2 = 1 \\ x^2 = 1 \end{cases}. \tag{36}$$

The first eigenvector gives two solutions  $(0, \sqrt{2})$  and  $(0, -\sqrt{2})$  and the second gives four solutions  $(1, \pm 1), (-1, \pm 1)$ . In this example we chose the basis  $\mathcal{B}_{0,0}$  but we could also have used  $\mathcal{B}_{0,1}$  to recover the solutions. However the other two choices would not have allowed us to recover the complete solution set.

### 3 Revealing Hidden Symmetries

In the previous section we have studied symmetries which depend on the exponents of the monomials. These properties are however not preserved under a linear change of variables and for some problems there can exist weighted symmetries which only appear after a change of variables.

*Example 6.* For  $\alpha \in (-\sqrt{2}, \sqrt{2})$  consider the following family of polynomial systems

$$\begin{cases} x^2 + y^2 = 1 \\ x + y = \alpha \end{cases} \tag{37}$$

Clearly the solution set is stable under the transform which switches  $x$  and  $y$  since the equation system is unchanged. But in this formulation the system does not have any weighted  $p$ -fold symmetries. Performing a change of variables

$$\begin{cases} \hat{x} = x + y \\ \hat{y} = x - y \end{cases} \implies \begin{cases} \frac{1}{2}\hat{x}^2 + \frac{1}{2}\hat{y}^2 = 1 \\ \hat{x} = \alpha \end{cases} \tag{38}$$

reveals a 2-fold symmetry in the  $\hat{y}$  variable.

The previous example showed a polynomial system which was invariant to a specific linear transform. After a change of variables the symmetry was transformed into a weighted  $p$ -fold symmetry as in Sect. 2. The following theorem shows that under some weak assumptions this can be done in general.

**Theorem 4.** *Let  $f_i(x) = 0, i = 1, 2, \dots, m$  be a polynomial system with a finite number of solutions. If there exist an invertible matrix  $A \neq I$  such that the solution set  $\mathcal{V} = \{x \mid f(x) = 0\} \subset \mathbb{C}^n$  is stable under  $A$ , then the polynomial system exhibits a  $c$ -weighted  $p$ -fold symmetry after a linear change of variables.*

*Proof.* We start by noting that we can without loss of generality assume that  $\text{span } \mathcal{V} = \mathbb{C}^n$ . If this does not hold it will be sufficient to consider the restriction of  $A$  to the span of  $\mathcal{V}$ , i.e.  $A|_{\mathcal{V}} : \text{span } \mathcal{V} \rightarrow \text{span } \mathcal{V}$ .

Since  $\mathcal{V}$  is finite and  $A$  injective we have that  $A\mathcal{V} = \mathcal{V}$ . This means that  $A$  acts as a permutation on the elements of  $\mathcal{V}$ . It follows that there must exist  $p \in \mathbb{N}$  such that

$$A^p s = s \quad \forall s \in \mathcal{V}, \tag{39}$$

since there are only a finite number of possible permutations. This implies

$$A^p = I, \tag{40}$$



since the elements of  $\mathcal{V}$  span  $\mathbb{C}^n$ . This is a sufficient condition for  $A$  to be diagonalizable and that the eigenvalues of  $A$  are  $p$ :th roots of unity, i.e.  $\lambda_k = e^{2\pi i \frac{c_k}{p}}$ . Let  $S$  be the matrix which diagonalizes  $A$ , then

$$A = SDS^{-1} = S \text{diag} \left( \{e^{2\pi i \frac{c_k}{p}}\}_{k=1}^n \right) S^{-1}. \tag{41}$$

Note that by definition  $D = D_p^c$  for  $\mathbf{c} = (c_1, c_2, \dots, c_n) \in \mathbb{Z}_p^n$  and if we perform the change of variables  $\hat{\mathbf{x}} = S^{-1}\mathbf{x}$  the solution set instead becomes stable under  $D_p^c$  and thus the system has a  $\mathbf{c}$ -weighted  $p$ -fold symmetry.  $\square$

If the system has multiple symmetry matrices we can use all of them if we are able to diagonalize them simultaneously. It is a well-known fact from linear algebra that a set of diagonalizable matrices can be simultaneously diagonalized if and only if they commute.

### 4 Finding Symmetries in Practice

Unless there is some problem specific knowledge it can be difficult to find the change of variables which reveals the symmetry. In this section we present a simple and automatic method for determining if a given problem has any symmetries of the type presented in this paper.

Assume that we are given a family of polynomial systems  $\{f_i(\mathbf{x}, \mathbf{a}) = 0\}$ , which depends on some data  $\mathbf{a} \in \mathbb{C}^m$ , i.e. for fix  $\mathbf{a}$  each  $f_i(\mathbf{x}, \mathbf{a})$  is polynomial in  $\mathbf{x}$ . To find the symmetries we do the following:

1. Take some instance  $\mathbf{a}^0 \in \mathbb{C}^m$  and solve the problem  $\{f_i(\mathbf{x}, \mathbf{a}^0) = 0\}$  using any method. This can for example be accomplished by selecting some  $\mathbf{a}^0$  where the solutions are known, or by using some numerical solver (e.g. PHCPack [19]). Denote the solutions  $\mathbf{s}_k^0$ ,  $k = 1, 2, \dots, N$ .
2. Next we generate a sequence of polynomial systems by updating data in small increments,  $\mathbf{a}^{t+1} = \mathbf{a}^t + \epsilon$ . For each solution  $\mathbf{s}_k^0$  we generate a *solution trajectory* by tracking the solution using non-linear refinement methods (e.g. Newton-Raphson). So to find  $\mathbf{s}_k^{t+1}$  we solve  $\{f_i(\mathbf{x}, \mathbf{a}^{t+1}) = 0\}$  by starting non-linear refinement at  $\mathbf{s}_k^t$ . This is repeated until the matrices

$$S_k = [\mathbf{s}_k^0 \ \mathbf{s}_k^1 \ \dots \ \mathbf{s}_k^t] \in \mathbb{C}^{n \times t}, \quad k = 1, 2, \dots, N \tag{42}$$

are all of full rank and  $t > n$ .

3. For each pair,  $i \neq j$ , we try to find a matrix  $A_{ij} \in \mathbb{C}^{n \times n}$  such that

$$A_{ij}S_i = S_j. \tag{43}$$

Since the system is overdetermined and the solution matrices  $S_i$  might have some small errors we solve (43) in a least square sense. If the residuals are sufficiently close to zero, we add the matrix  $A_{ij}$  to a list of possible symmetry matrices.

4. For each matrix  $A_{ij}$  we check if the solution set is stable, i.e. if for each  $k$  there exist  $\ell \neq k$  such that  $A_{ij}S_k = S_\ell$ . The matrices which satisfy this are the symmetry matrices.
5. Finally we generate new instances and check if the symmetry matrices work.

## 5 Weak Perspective- $n$ -Points

In this section we show a practical example where we can construct a more compact polynomial solver by exploiting a symmetry in the problem. We consider the problem of estimating a weak perspective camera (also known as scaled orthographic camera) from  $n$  2D-3D point correspondences. We find the pose which minimizes the squared reprojection error. By eliminating the translation and performing a change of variables the problem can be reduced to (see supplementary material)

$$\min_{s,R} \left\| R \operatorname{diag}(a_1, a_2, a_3) - \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{bmatrix} \right\|_F^2 \quad \text{s.t.} \quad RR^T = s^2 I_2, \quad (44)$$

where  $a_1 \geq a_2 \geq a_3 \geq 0$ .

### 5.1 Parameterizing the Constraints

We use the unconstrained quaternion parametrization of the scaled  $2 \times 3$  rotation,

$$R(q) = \begin{bmatrix} q_1^2 + q_2^2 - q_3^2 - q_4^2 & 2(q_2q_3 - q_1q_4) & 2(q_1q_3 + q_2q_4) \\ 2(q_1q_4 + q_2q_3) & q_1^2 - q_2^2 + q_3^2 - q_4^2 & 2(q_3q_4 - q_1q_2) \end{bmatrix}, \quad (45)$$

where  $q = (q_1, q_2, q_3, q_4)$  and  $\|q\|_2 = s$ . In this parametrization the problem in (44) becomes unconstrained and the cost function

$$f(q) = \|R(q)A - B\|_F^2. \quad (46)$$

is a quartic polynomial in the elements of  $q$ . Since each element in  $R(q)$  is of degree two we have that  $f(q)$  only contains monomials of degrees 0, 2 and 4.

We find the optimal pose by studying the first-order necessary conditions for (46). Since the problem is now unconstrained in the unscaled quaternion representation we simply solve for the critical points, i.e.

$$g(q) = \nabla_q f(q) = 0. \quad (47)$$

Since  $f(q)$  only contains even terms, this equation system  $g(q) = 0$  can only contain odd terms (degree 1 or 3). Thus we can directly see that the equation system will have at least a two-fold symmetry. This symmetry correspond to the sign ambiguity of the quaternion representation, i.e.  $R(q) = R(-q)$ .

### 5.2 Additional Symmetries in the Solutions

The quaternion parametrization is inherently ambiguous since the sign of the quaternion does not matter. But it turns out that there is a further ambiguity which comes from the fact that the third row of the rotation matrix is ignored. By studying (45) we can see that  $R(q_1, q_2, q_3, q_4) = R(iq_4, iq_3, -iq_2, -iq_1)$  holds for all  $q \in \mathbb{C}^4$ . For the full  $3 \times 3$  rotation matrix this corresponds to changing

sign of the third row. Together these ambiguities introduce a four-fold symmetry into the solution set which can be described by the matrices

$$A_1 = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \quad \text{and} \quad A_2 = \begin{pmatrix} 0 & 0 & 0 & i \\ 0 & 0 & i & 0 \\ 0 & -i & 0 & 0 \\ -i & 0 & 0 & 0 \end{pmatrix}. \tag{48}$$

To diagonalize the matrices we perform the change of variables,

$$\hat{\mathbf{q}} = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & -i & -i & 0 \\ -1 & 0 & 0 & 1 \\ i & 0 & 0 & i \\ 0 & -1 & 1 & 0 \end{bmatrix} \mathbf{q}. \tag{49}$$

In these new variables the rotation matrix becomes

$$R(\hat{\mathbf{q}}) = \begin{bmatrix} \hat{q}_1^2 - \hat{q}_2^2 - \hat{q}_3^2 + \hat{q}_4^2 & -i\hat{q}_1^2 - i\hat{q}_2^2 + i\hat{q}_3^2 + i\hat{q}_4^2 & 2(\hat{q}_1\hat{q}_2 + 2\hat{q}_3\hat{q}_4) \\ -i\hat{q}_1^2 + i\hat{q}_2^2 - i\hat{q}_3^2 + i\hat{q}_4^2 & -\hat{q}_1^2 - \hat{q}_2^2 - \hat{q}_3^2 - \hat{q}_4^2 & 2i(\hat{q}_3\hat{q}_4 - \hat{q}_1\hat{q}_2) \end{bmatrix}. \tag{50}$$

Note that the only mixed terms are  $\hat{q}_1\hat{q}_2$  and  $\hat{q}_3\hat{q}_4$ . This leads to one 2-fold symmetry in  $(\hat{q}_1, \hat{q}_2)$  and one 2-fold symmetry in  $(\hat{q}_3, \hat{q}_4)$ .

### 5.3 Constructing a Polynomial Solver

By studying the problem in Macaulay2 [20] and Maple [21] we find that the system has 33 solutions. Ignoring the trivial solution, the rest of the solutions can be grouped into eight groups of four solutions. Using the method from Sect. 2.1 we constructed a polynomial solver. We used the following eight basis monomials,

$$\mathcal{B}_{0,0} = \{\hat{q}_1\hat{q}_2, \hat{q}_2^2, \hat{q}_3^2\hat{q}_4^2, \hat{q}_3^2, \hat{q}_3\hat{q}_4^3, \hat{q}_3\hat{q}_4, \hat{q}_4^4, \hat{q}_4^2\}, \tag{51}$$

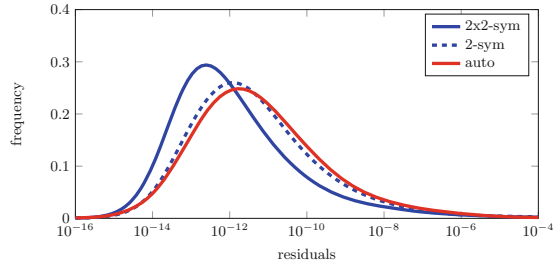
which have the weighted remainder zero for both symmetries. For our action polynomial we used  $a(\hat{\mathbf{q}}) = \hat{q}_1\hat{q}_2$ , which also has zero remainder.

### 5.4 Experimental Evaluation

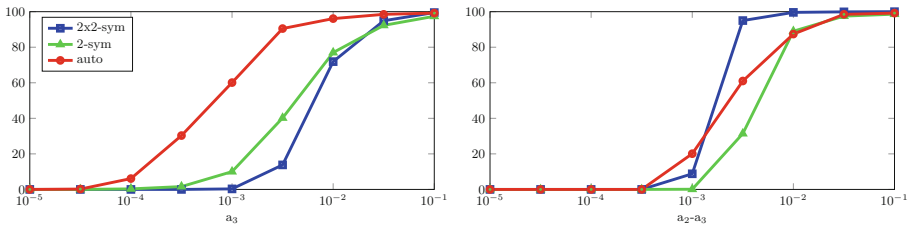
In this section we experimentally evaluate the polynomial solver from the previous section. Using the same method we also constructed a solver which only uses the partial 2-fold symmetry present in the original formulation. For comparison we also used the automatic method from Kukulova et al. [22] to generate a polynomial solver. This solver does not take any symmetry into account. The

**Table 1.** Size of the elimination template and action matrix for the three solvers.

	2x2-sym	2-sym	Kukulova et al. [22]
Elimination template	104 × 90	234 × 276	243 × 276
Action matrix	8 × 8	16 × 16	33 × 33



**Fig. 1.** Histogram over the residuals for 1000 random instances.



**Fig. 2.** Percentage of successful instances (all residuals smaller than  $10^{-6}$ ) when the structure is close to planar, i.e.  $a_3 \approx 0$  (Left) and when there are two almost equal singular values, i.e.  $a_2 \approx a_3$ . (Right)

sizes of the elimination templates and action matrices for the three methods can be seen in Table 1.

We evaluated the three solvers on synthetic instances. Figure 1 shows the distribution of the residuals over 1000 random instances. The automatically generated solver and the solver using only the partial 2-fold symmetry has very similar performance, while the solver utilizing the full symmetry performs slightly better. The average runtimes for the instances were 11.2ms for the `2-sym` solver and 2.7ms for the `2x2-sym` solver. For the automatically generated solver the average runtime was 2.1 s.<sup>2</sup>

**Degeneracies.** Under some conditions the problem changes nature and the polynomial solvers break down. For this problem we empirically found that this happens when either the structure is planar ( $a_3 = 0$ ) or if two of the singular values are equal ( $a_1 = a_2$  or  $a_2 = a_3$ ). Close to these configurations the solvers become numerically unstable. We compare the performance of the polynomial solver on random instances close to these degeneracies. We generated random instances where the third singular value approached zero and instances where  $|a_2 - a_3|$  approached zero. Figure 2 shows the percentage of successful instances

<sup>2</sup> The large runtime is due to use of the MATLAB function `rref` for performing the elimination. In principal this could be replaced by a faster implementation, which most likely would yield runtimes similar to the `2-sym` solver.

(defined as all residuals less than  $10^{-6}$ ) as we approach the degenerate configurations. For the planar degeneracy the automatically generated solver has slightly better performance while for the other degeneracy the results are comparable.

## 6 Absolute Pose with Unknown Focal Length

In this section we show another practical example, where symmetry occurs naturally in the polynomial systems. We consider the absolute pose problem with unknown focal length. The problem has 7 degrees of freedom and requires at least 3.5 points to be determined. We consider the (almost) minimal problem of estimation from four 2D-3D point correspondences.

The normalized image points  $\mathbf{x}_k$  should satisfy the projection equations

$$\lambda_k \mathbf{x}_k = \lambda_k (x_k \ y_k \ f)^T = R \mathbf{X}_k + \mathbf{t}, \quad k = 1, 2, 3, 4. \quad (52)$$

This means that the four vectors  $\lambda_1 \mathbf{x}_1, \lambda_2 \mathbf{x}_2, \lambda_3 \mathbf{x}_3$  and  $\lambda_4 \mathbf{x}_4$  differ only by a rigid transformation from the 3D points  $\mathbf{X}_k$ . We form the vectors from the first point to the others, i.e.

$$\mathbf{V} = [(\mathbf{X}_2 - \mathbf{X}_1), (\mathbf{X}_3 - \mathbf{X}_1), (\mathbf{X}_4 - \mathbf{X}_1)], \quad (53)$$

$$\mathbf{v} = [(\lambda_2 \mathbf{x}_2 - \lambda_1 \mathbf{x}_1), (\lambda_3 \mathbf{x}_3 - \lambda_1 \mathbf{x}_1), (\lambda_4 \mathbf{x}_4 - \lambda_1 \mathbf{x}_1)]. \quad (54)$$

Since rigid transformations preserve lengths and angles we must have

$$\mathbf{V}^T \mathbf{V} = \mathbf{v}^T \mathbf{v}, \quad (55)$$

which gives six independent equations in the five unknowns:  $f, \lambda_1, \lambda_2, \lambda_3$  and  $\lambda_4$ .

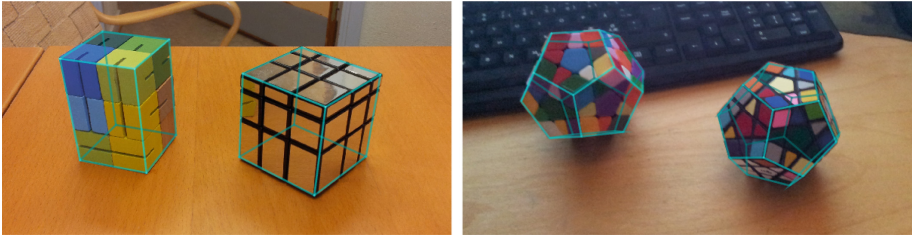
In this formulation the problem has two 2-fold symmetries, corresponding to the sign ambiguities in the focal length and the projective depths  $\lambda_k$ . Together these introduce a four-fold symmetry in the solution set.

Since the problem is overconstrained we cannot use all equations from (55). Using Macaulay2 [20] we found that if we discard one of the off-diagonal equations ( $\mathbf{V}_i^T \mathbf{V}_j = \mathbf{v}_i^T \mathbf{v}_j$ ) the remaining system has 40 solutions. However if we discard one of the diagonal equations ( $\|\mathbf{V}_i\|^2 = \|\mathbf{v}_i\|^2$ ) we get 24 solutions. By exploiting the symmetry we only have to find 10 and 6 of these solutions respectively.

Using the method from Sect. 2.1 we construct a polynomial solver for the case with 24 solutions. The solver uses both symmetries and only has to find 6 solutions. We used  $a(\mathbf{x}) = \lambda_1 \lambda_2$  as our action polynomial and as our monomial basis we used

$$\mathcal{B}_{0,0} = \{1, \lambda_1 \lambda_4, \lambda_2 \lambda_4, \lambda_3^2, \lambda_3 \lambda_4, \lambda_4^2\}. \quad (56)$$

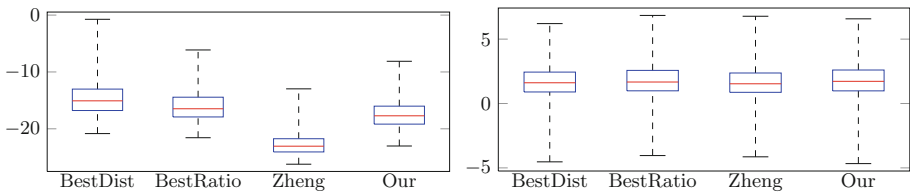
The elimination template is of size  $139 \times 185$  and the resulting action matrix is  $6 \times 6$ . Note that neither the action polynomial nor the monomial basis contain the focal length  $f$ . However this does not pose any problem, as once the lengths  $\lambda_i$  are recovered the equations in (55) reduce to a second degree polynomial in  $f$  which can be easily solved. Figure 3 show two examples where we have applied the solver to pose estimation problems from real images.



**Fig. 3.** Examples of pose estimation from real images.

## 6.1 Experimental Evaluation

We compare the polynomial solver with the most accurate solvers from Bujnak [23] based on the ratio and distance constraints (denoted **Best-Ratio** and **Best-Dist**). We also compare with the more recent method from Zheng et al. [10] (denoted **Zheng** in the experiments). In 2015 Wu [2] presented another minimal solver for this problem. We do not include a comparison with this method since we were unable to find any implementation, but in their paper the performance is similar to that of [10]. For the experiment we generated 3D points uniformly in the box  $[-2, 2] \times [-2, 2] \times [4, 8]$  in the camera's coordinate system. The cameras were then randomly generated with focal length  $f = 1000$ . Figure 4 shows the reprojection error for 1000 random instances, both with and without random noise added to the image coordinates. We can see that the solver has comparable performance to current state-of-the-art methods. The average runtimes for the four methods were 149 ms (**Best-Dist**), 4.44 ms (**Best-Ratio**), 4.65 ms (**Zheng**) and 3.90 ms for our solver.



**Fig. 4.** The squared reprojection error for 1000 random instances. If there are multiple solutions we take the smallest error. *Left:* No noise added. *Right:* Gaussian noise with 5 px standard deviation added to the image points.

## 7 Conclusions

In this paper we have presented new techniques for using symmetry in the action matrix method. The theory relies on a generalization of the partial  $p$ -fold symmetry from [13]. However in contrast to [13] our method allows us to handle

multiple independent symmetries, resulting in even more compact polynomial solvers. Furthermore, we have shown that these symmetries are not restricted to theoretical examples, but occur in real problems from Computer Vision.

**Acknowledgments.** This work has been funded by the Swedish Research Council (grant no. 2012-4213), the Strategic Research Area ELLIIT and by the Wallenberg Autonomous Systems Program.

## References

1. Haner, S., Åström, K.: Absolute pose for cameras under flat refractive interfaces. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1428–1436 (2015)
2. Wu, C.: P3.5p: pose estimation with unknown focal length. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2440–2448 (2015)
3. Albl, C., Kukulova, Z., Pajdla, T.: R6p-rolling shutter absolute camera pose. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2292–2300(2015)
4. Stewénius, H., Nistér, D., Oskarsson, M., Åström, K.: Solutions to minimal generalized relative pose problems. In: Workshop on Omnidirectional Vision (OMNIVIS), vol. 1, p. 3(2005)
5. Nistér, D.: An efficient solution to the five-point relative pose problem. *IEEE Trans. Pattern Anal. Mach. Intell. (PAMI)* **26**(6), 756–770 (2004)
6. Kuang, Y., Solem, J.E., Kahl, F., Åström, K.: Minimal solvers for relative pose with a single unknown radial distortion. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 33–40. IEEE (2014)
7. Kukulova, Z., Heller, J., Bujnak, M., Pajdla, T.: Radial distortion homography. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 639–647 (2015)
8. Brown, M., Hartley, R.I., Nistér, D.: Minimal solutions for panoramic stitching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8. IEEE (2007)
9. Zheng, Y., Kuang, Y., Sugimoto, S., Åström, K., Okutomi, M.: Revisiting the pnp problem: a fast, general and optimal solution. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 2344–2351 (2013)
10. Zheng, Y., Sugimoto, S., Sato, I., Okutomi, M.: A general and simple method for camera pose and focal length determination. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 430–437 (2014)
11. Byröd, M., Josephson, K., Åström, K.: Fast and stable polynomial equation solving and its application to computer vision. *Int. J. Comput. Vision (IJCV)* **84**(3), 237–256 (2009)
12. Ask, E., Kuang, Y., Åström, K.: Exploiting p-fold symmetries for faster polynomial equation solving. In: Proceedings of the International Conference on Pattern Recognition (ICPR), pp. 3232–3235. IEEE (2012)
13. Kuang, Y., Zheng, Y., Åström, K.: Partial symmetry in polynomial systems and its applications in computer vision. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2014

14. Kneip, L., Li, H., Seo, Y.: UPnP: an optimal  $O(n)$  solution to the absolute pose problem with universal applicability. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 127–142. Springer, Heidelberg (2014). doi:[10.1007/978-3-319-10590-1\\_9](https://doi.org/10.1007/978-3-319-10590-1_9)
15. Ask, E., Enqvist, O., Kahl, F.: Optimal geometric fitting under the truncated l2-norm. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1722–1729 (2013)
16. Enqvist, O., Ask, E., Kahl, F., Åström, K.: Tractable algorithms for robust model estimation. *Int. J. Comput. Vision (IJCV)* **112**(1), 115–129 (2015)
17. Corless, R.M., Gattermann, K., Kotsireas, I.S.: Using symmetries in the eigenvalue method for polynomial systems. *J. Symbolic Comput.* **44**(11), 1536–1550 (2009)
18. Cox, D.A., Little, J., O’Shea, D.: *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*, 3/e (Undergraduate Texts in Mathematics). Springer-Verlag New York Inc, Secaucus (2007)
19. Verschelde, J.: Algorithm 795: phpack: a general-purpose solver for polynomial systems by homotopy continuation. *ACM Trans. Math. Softw. (TOMS)* **25**(2), 251–276 (1999)
20. Grayson, D.R., Stillman, M.E.: Macaulay2, a software system for research in algebraic geometry. <http://www.math.uiuc.edu/Macaulay2/>
21. Monagan, M.B., Geddes, K.O., Heal, K.M., Labahn, G., Vorkoetter, S.M., McCarron, J., DeMarco, P.: *Maple 10 Programming Guide*. Maplesoft, Waterloo ON, Canada (2005)
22. Kukulova, Z., Bujnak, M., Pajdla, T.: Automatic generator of minimal problem solvers. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008. LNCS, vol. 5304, pp. 302–315. Springer, Heidelberg (2008). doi:[10.1007/978-3-540-88690-7\\_23](https://doi.org/10.1007/978-3-540-88690-7_23)
23. Bujnák, M.: Algebraic solutions to absolute pose problems. Ph.D. thesis (2012)