# Global Optimization for Polynomial Programming Problems Using *m*-homogeneous Polynomial Homotopies

**Layne T. Watson and Alexander P. Morgan**

**TR 89-31**

This paper has appeared in
Journal of Computational and Applied
Mathematics 43 (1992), 373-82

Journal of Computational and Applied Mathematics 43 (1992) 373–382
North-Holland



CAM 1148

# Polynomial programming using multi-homogeneous polynomial continuation

Layne T. Watson *

*Department of Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061, United States*

Alexander P. Morgan

*Mathematics Department, General Motors Research Laboratories, Warren, MI 48090, United States*

Received 12 August 1991

*Abstract*

Watson, L.T. and A.P. Morgan, Polynomial programming using multi-homogeneous polynomial continuation, Journal of Computational and Applied Mathematics 43 (1992) 373–382.

A polynomial programming problem is a nonlinear programming problem where the objective function, inequality constraints and equality constraints all consist of polynomial functions. The necessary optimality conditions for such a problem can be formulated as a polynomial system of equations, among whose zeros the global optimum must lie. This note applies the theory of *multi-homogeneous* (also called *m-homogeneous*) polynomials and recent homotopy algorithms to the polynomial system formulation of the necessary optimality conditions, significantly reducing the work of a naive homotopy approach. The $m$-homogeneous homotopy approach, providing the global optimum, is practical for small problems. For example, the geometric modeling problem of finding the distance between two polynomial surfaces is a polynomial programming problem. The theory is illustrated in the last section by this geometric modeling problem and a prototype structural design problem.

*Keywords:* Global optimization; globally convergent; homotopy algorithm; $m$-homogeneous; multi-homogeneous; parallel optimization; polynomial continuation; polynomial program.

## 0. Notation

Let $E^n$ denote $n$-dimensional real Euclidean space, $C^n$ denote $n$-dimensional complex Euclidean space, $E^{m\times n}$ the set of real $m\times n$ matrices, and $C^{m\times n}$ the set of complex $m\times n$

*Correspondence to:* Prof. L.T. Watson, Department of Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061-0106, United States.

* The work of this author was supported in part by Department of Energy grant DE-FG05-88ER25068, National Aeronautics and Space Administration grant NAG-1-1079 and Air Force Office of Scientific Research grant ...0497.

matrices. Lower case Greek letters will be real or complex scalars or scalar-valued functions, and Roman letters will generally denote vectors and vector-valued functions. $x_i$ denotes the $i$th component of the vector $x$, and for a matrix $A$, $A_{ij}$ denotes the $i$, $j$ entry, $A_{.j}$ denotes the $j$th column, and $A_{i.}$ denotes the $i$th row. The Jacobian matrix of a function $f(x)$ is written $\nabla f(x)$.

## 1. Introduction

Consider the general nonlinear programming problem

$$\begin{aligned} &\min && \theta(x), \\ &\text{subject to} && g(x) \leqslant 0,\ h(x) = 0, \end{aligned} \tag{1}$$

where $\theta : E^n \to E$, $g : E^n \to E^p$ and $h : E^n \to E^q$ are polynomial functions. Precisely, each component of $\theta$, $g$ and $h$ has the form

$$\sum_{k=1}^{n_i} \alpha_{ik} \prod_{j=1}^{n} x_j^{d_{ijk}},$$

where the $\alpha_{ik}$ are real and the $d_{ijk}$ are nonnegative integers. Such a problem will be called a *polynomial programming problem* [1].

By adding slack variables, the inequality constraints

$$g_k(x) \leqslant 0$$

can be converted to equality constraints

$$g_k(x) + y_k^2 = 0.$$

Henceforth, assume that all the constraints are equality constraints, so that (1) becomes

$$\begin{aligned} &\min && \theta(x), \\ &\text{subject to} && h(x) = 0. \end{aligned} \tag{2}$$

The Lagrange Multiplier Theorem says that if $\bar{x}$ is a locally optimal solution for (2) and rank $\nabla h(\bar{x}) = q$, then $\exists \bar{r} \in E^q$ such that

$$\nabla\theta(\bar{x}) + \bar{r}^{\mathrm{t}}\, \nabla h(\bar{x}) = 0, \quad h(\bar{x}) = 0. \tag{3}$$

Since $\theta$ and $h$ are polynomial functions, so are $\nabla\theta$ and $\nabla h$. Thus (3) is a polynomial system of $n + q$ equations in the $n + q$ unknowns $x$ and $r$. So the *polynomial programming problem* (1) reduces to the *polynomial system of equations* (3).

*Polynomial continuation* [2,5,8] provides a globally convergent homotopy algorithm guaranteed to find *all* the solutions of (3), and thus the global optimum of (1). In [9,10], homotopy methods are used to solve optimization problems, but only for *local* (not *global*) optimization. In [3,4] traditional polynomial continuation is used to solve the global polynomial programming problem. In this paper, the more efficient multi-homogeneous theory is applied. The next section updates traditional polynomial continuation theory to the multi-homogeneous context. Section 3 develops a standard multi-homogeneous formulation of (3), and Section 4 illustrates the theory with several examples.

## 2. Multi-homogeneous polynomial continuation

Polynomial continuation is a numerical method for computing all the geometrically isolated solutions to polynomial systems. Let $f(z)=0$ denote a system of $N$ polynomial equations in $N$ unknowns. The degree of the $i$th equation is $d_i = \max_k \Sigma_{j=1}^N d_{ijk}$ and

$$\text{td} = \prod_{i=1}^{N} d_i$$

is the *total degree* of the system. Traditional polynomial continuation computes the full list of geometrically isolated solutions to $f(z)=0$ by numerically tracking td paths in the space $C^N \times [0, 1]$. See, for example, [2,5,8].

Although this method works quite well when td is relatively small, the computational cost for larger systems can be prohibitive. A recent advance in polynomial continuation, the $m$-homogeneous approach of Morgan and Sommese [6], reduces the number of paths that must be tracked in many cases. (We use $m$-homogeneous and multi-homogeneous interchangeably.) By partitioning the variables to create an *m-homogeneous structure*, we can solve the system tracking only the *Bezout number* of paths. (The terminology "$m$-homogeneous" and "Bezout number" were first coined in [6].) Frequently, we can arrange for the Bezout number to be smaller than the total degree. The mechanics of numerically tracking the paths are essentially the same as for the traditional method. Here, we describe how to create an $m$-homogeneous structure and find the Bezout number. In [6], the method and its theory are more fully developed. See [7] for a tutorial overview with illustrative applications from the kinematics of mechanisms.

We create an $m$-homogeneous structure for $f(z)$ by partitioning the variables $z_1, z_2, \ldots, z_N$ into $m$ nonempty sets. It will be simpler for the exposition if we re-index with double subscripts. Thus

$$\{z_1, \ldots, z_N\} = \bigcup_{j=1}^{m} \{z_{1,j}, \ldots, z_{k_j,j}\},$$

where $\Sigma_{j=1}^m k_j = N$. Now create homogeneous variables $z_{0,j}$ for $j = 1, \ldots, m$ and define

$$Z_j = \{z_{0,j}, z_{1,j}, \ldots, z_{k_j,j}\},$$

for $j=1$ to $m$. Then evoke the substitution $z_{i,j} \leftarrow z_{i,j}/z_{0,j}$ for $i = 1, \ldots, k_j$ and $j = 1, \ldots, m$, generating a system $f'=0$ of $N$ equations in $N+m$ unknowns (after clearing the denominators of powers of the $z_{0,j}$). This $f'$ is called *m-homogeneous* because the variables are partitioned into $m$ collections $Z_1, \ldots, Z_m$, so that $f'$ is homogeneous as a system in the variables of any one of the collections. We take $d_{j,l}$ to denote the $j$th degree of the $l$th polynomial; that is, with all variables held fixed except those in $Z_j$, $f_l'$ has homogeneous degree $d_{j,l}$. Polynomial $f_l'$ is said to have *type* $= (d_{1,l}, \ldots, d_{m,l})$.

The *Bezout number* $d$ of an $m$-homogeneous polynomial system is given by

$$d = \text{Coef}\left[D, \prod_{j=1}^{m} \phi_j^{k_j}\right], \tag{4}$$

where

$$D = \prod_{i=1}^{n} \sum_{j=1}^{m} d_{j,l}\phi_j, \tag{5}$$

i.e., $d$ is the coefficient of the $\prod_{j=1}^{m}\phi_j^{k_j}$ term of $D$. Frequently, an $m$-homogenization of $f$ for $m>1$ has a (much) smaller Bezout number than the $m=1$ case, where $d$ equals the total degree of $f$. There are several fully worked out examples in [6,7].

**Example 1.** Consider the following system:

$$\begin{aligned} &z_1z_2z_3z_4+1=0, \qquad z_1z_3+z_2z_4+z_1z_4=0,\\ &4z_1z_3z_4-2z_2z_3z_4+1=0, \qquad z_1+z_2=0. \end{aligned} \tag{6}$$

By grouping the variables of (6) into different sets, we create different $m$-homogeneous structures and Bezout numbers. Normally, we would want to solve such a system with the $m$-homogeneous structure that gives the smallest Bezout number. For each grouping of variables, we will form the combinatorial product $D$ defined in (5), and then pick out the distinguished coefficient that gives the Bezout number $d$, as in the following examples.

**Example 1.1.** Group variables as: $\{z_1, z_2\} \cup \{z_3, z_4\}$. $m=2$, so there are two homogeneous variables, $z_{01}$ and $z_{02}$. For example, the 2-homogenized third equation becomes $4z_1z_3z_4 - 2z_2z_3z_4 + z_{01}z_{02}^2 = 0$. The types of the equations are (2, 2), (1, 1), (1, 2), (1, 0); these become the integer coefficients of the two symbols $\phi_1$ and $\phi_2$ ($m=2$) in the polynomial (5). Then. $D \equiv (2\phi_1+2\phi_2)(\phi_1+\phi_2)(\phi_1+2\phi_2)(\phi_1+0\phi_2)$ and $d = \text{Coef}[D, \phi_1^2\phi_2^2] = 10$.

**Example 1.2.** Group variables as: $\{z_1, z_2\} \cup \{z_3\} \cup \{z_4\}$. Then, $D = (2\phi_1+\phi_2+\phi_3)(\phi_1+\phi_2+\phi_3)^2(\phi_1+0\phi_2+0\phi_3)$ and $d = \text{Coef}[D, \phi_1^2\phi_2\phi_3] = 8$.

**Example 1.3.** Group variables as: $\{z_1\} \cup \{z_2\} \cup \{z_3, z_4\}$. Then, $D = (\phi_1+\phi_2+2\phi_3)^2(\phi_1+\phi_2+\phi_3)(\phi_1+\phi_2+0\phi_3)$ and $d = \text{Coef}[D, \phi_1\phi_2\phi_3^2] = 16$.

We see that Example 1.2 gives the smallest Bezout number. Thus, while the 1-homogeneous (traditional) polynomial continuation yields a 24-path homotopy (i.e., the total degree), we can (easily) find a 3-homogeneous 8-path homotopy. Such a savings in computer work (i.e., by a factor of $\frac{1}{3}$) can be significant in some applications.

To summarize: given a system of $N$ polynomial equations in $N$ unknowns, the traditional constructions from polynomial continuation require tracking the total-degree number of paths in order to compute the complete list of geometrically isolated solutions. The purpose of the $m$-homogeneous approach is to reduce the number of paths needed to solve the problem. thereby realizing a savings in computational work. The $m$-homogeneous method is begun by partitioning the variables into sets. Then (4) yields the Bezout number, the associated number of paths to be tracked. A different partitioning of the variables yields a different Bezout number, and there is no known systematic way to find the partitioning that yields the smallest Bezout number, aside from exhaustive search.

## 3. A multi-homogeneous homotopy formulation of the polynomial programming problem

This section gives a partitioning of the variables of the polynomial programming problem and presents a simple formula for the associated Bezout number. It should be noted that for any particular problem, additional savings can often be obtained by customizing the $m$-homogeneous partitioning to the problem. Specific examples are given in the next section.

Partition the variables into two sets $S_1$ and $S_2$, where

$$S_1 = \{x_1, x_2, \ldots, x_n\} \quad \text{and} \quad S_2 = \{r_1, r_2, \ldots, r_q\},$$

corresponding to a 2-homogeneous structure.

Consider the $i$th Lagrangian equation

$$(\nabla_x L)_i = \nabla\theta_i + \sum_{j=1}^{q} r_j \, \nabla h_{j,i},$$

where $L(x, r) = \theta(x) + r^t h(x)$ is the Lagrangian function for (2) and

$$\nabla h_{j,i} = \frac{\partial h_j}{\partial x_i}.$$

Define, for $i = 1, \ldots, n$,

$$\delta_i^{\nabla\theta} = \begin{cases} -1, & \text{if } \nabla\theta_i \equiv 0, \\ \deg(\nabla\theta_i), & \text{otherwise,} \end{cases}$$

and

$$\delta_i^{\nabla h} = \begin{cases} -1, & \text{if } \nabla h_{j,i} \equiv 0 \text{ for } j = 1, \ldots, q, \\ \max\{\deg(\nabla h_{j,i}) \mid j = 1, \ldots, q\}, & \text{otherwise.} \end{cases}$$

By re-ordering the indices of the $x_i$ if necessary, there is a nonnegative integer $q_0 \leqslant q$ so that $\delta_i^{\nabla h} = -1$ for $i = 1, \ldots, q_0$, and $\delta_i^{\nabla h} \neq -1$ for $i = q_0 + 1, \ldots, q$. Take $q_0 = 0$ if no $\delta_i^{\nabla h} = -1$. By the definition of $q_0$, note that $x_1, x_2, \ldots, x_{q_0}$ are the variables that do not appear in $h$ and $x_{q_0+1}, \ldots, x_n$ are the ones that do.

Define

$$\delta_i = \max\{\delta_i^{\nabla\theta}, \delta_i^{\nabla h}\}, \quad \text{for } i = 1, \ldots, n,$$

and

$$\delta_i^h = \deg(h_i), \quad \text{for } i = 1, \ldots, q.$$

Assume (without loss of generality) that $\delta_i \neq -1$, $\delta_i^h \neq 0$, and, if $\delta_i^{\nabla\theta} = 0$, then $\delta_i^{\nabla h} \neq -1$, for any $i$.

We can now generate the Bezout number. The type of the $i$th Lagrangian equation is $(\delta_i, 1)$ if $i > q_0$ and $(\delta_i, 0)$ if $i \leqslant q_0$. The type of the $i$th constraint equation is $(\delta_i^h, 0)$. Then $D$ from (5) is defined from these types as

$$D = \prod_{i=1}^{q} \delta_i^h \phi_1 \prod_{i=1}^{q_0} \delta_i \phi_1 \prod_{i=q_0+1}^{n} (\delta_i \phi_1 + \phi_2),$$

where the second factor is omitted if $q_0 = 0$, and $\phi_1$ and $\phi_2$ correspond to $S_1$ and $S_2$, respectively. Then the Bezout number is given by

$$d = \mathrm{Coef}[D, \phi_1^n \phi_2^q].$$

Note that we will consider only the case $q < n$. If $q > n$, then either the constraint set cannot be satisfied or it can be reduced by omitting redundant equations. If $q = n$, then either the constraint set can be solved as an independent system or it can be reduced by omitting redundant equations. The usual condition that rank $\nabla h(\bar{x}) = q$ rules out these redundant cases.

We see that

$$d = \prod_{i=1}^{q} \delta_i^h \prod_{i=1}^{q_0} \delta_i \ \mathrm{Coef}[D', \phi_1^{n-q_0-q} \phi_2^q],$$

where

$$D' = \prod_{i=q_0+1}^{n} (\delta_i \phi_1 + \phi_2),$$

and, by some simple combinatorial observations, we conclude that

$$d = \prod_{i=1}^{q} \delta_i^h \prod_{i=1}^{q_0} \delta_i \sum_{q_0+1 \leqslant i_1 < i_2 < \cdots < i_{n-q_0-q} \leqslant n} \delta_{i_1} \delta_{i_2} \cdots \delta_{i_{n-q_0-q}}. \tag{7}$$

By comparison, the total degree is

$$\mathrm{td} = \prod_{i=1}^{q} \delta_i^h \prod_{i=1}^{n} \max\{\delta_i^{\nabla\theta}, \delta_i^{\nabla h} + 1\}.$$

For example, if $q = n - 1$ (e.g., the number of constraints is one less than the number of variables) and $q_0 = 0$, then we get

$$d = \prod_{i=1}^{q} \delta_i^h \sum_{i=1}^{n} \delta_i.$$

Another case of interest occurs when the objective function and constraints are all quadratics. For simplicity, assume each variable occurs in at least one constraint raised to the second power. (The resulting Bezout number will be an upper bound for the other quadratic cases.) Then $q_0 = 0$ and $\delta_i^{\nabla h} = 1$ for all $i$. It follows that

$$d = 2^q \sum_{i \leqslant i_1 < \cdots < i_{n-q} \leqslant n} 1 = 2^q \binom{n}{n-q} = 2^q \binom{n}{q}. \tag{8}$$

Compare this to the generally much larger total degree in this case:

$$\mathrm{td} = 2^{q+n}.$$

## 4. Examples

This section presents two specific examples. The first is a realistic "small" problem that arises in geometric modeling. The second is a prototype structural design problem.

*4.1. Geometric modeling problem*

Let $P_1$ and $P_2$ be two polynomial surfaces in $E^3$. The problem is to compute the distance between $P_1$ and $P_2$, defined to be the length of the smallest line segment connecting them. Assuming that

$$P_i = \{z \in E^3 \mid h_i(z) = 0\},$$

where $h_i$ is a $d_i$th-degree polynomial, the problem becomes

$$\begin{array}{ll} \min & \theta(x, y) = \|x - y\|^2, \\ \text{subject to} & h_1(x) = 0,\ h_2(y) = 0. \end{array}$$

The necessary conditions (3) in this case are

$$\begin{aligned} \nabla_{(x,y)} L(x, y, r) = 2[&x_1 - y_1,\ x_2 - y_2,\ x_3 - y_3,\ -(x_1 - y_1),\ -(x_2 - y_2),\ -(x_3 - y_3)] \\ &+ r_1[\nabla h_{1,1}, \nabla h_{1,2}, \nabla h_{1,3}, 0, 0, 0] + r_2[0, 0, 0, \nabla h_{2,4}, \nabla h_{2,5}, \nabla h_{2,6}], \end{aligned}$$

$$h_1(x) = 0, \qquad h_2(y) = 0.$$

Here $n = 6$, $q = 2$, and $\delta_i^{\nabla\theta} = 1$ for $i = 1, \ldots, 6$. Then

$$\delta_i^{\nabla h} = \begin{cases} -1, & \text{if } \nabla h_{1,i} \equiv 0, \\ \deg(\nabla h_{1,i}), & \text{otherwise}, \end{cases} \quad \text{for } i = 1, \ldots, 3,$$

$$\delta_i^{\nabla h} = \begin{cases} -1, & \text{if } \nabla h_{2,i} \equiv 0, \\ \deg(\nabla h_{2,i}), & \text{otherwise}, \end{cases} \quad \text{for } i = 4, \ldots, 6,$$

$$\delta_i = \max\{1, \delta_i^{\nabla h}\}, \quad \text{for } i = 1, \ldots, 6,$$

and

$$\delta_i^h = d_i, \quad \text{for } i = 1, 2.$$

For the special case $q_0 = 0$, (7) gives

$$d = \delta_1^h \delta_2^h \sum_{1 \leq i_1 < i_2 < i_3 < i_4 \leq 6} \delta_{i_1} \delta_{i_2} \delta_{i_3} \delta_{i_4}.$$

If $q_0 = 0$ and $h_1$ and $h_2$ are quadratics, then $\delta_i = 1$ for all $i$, and by (8),

$$d = 4\binom{6}{2} = 60,$$

while the total degree is $2^8 = 256$.

However, one can do better with a customized $m$-homogeneous breakdown; namely, partition the variables as

$$\{x_1, x_2, x_3, y_1, y_2, y_3\} \cup \{r_1\} \cup \{r_2\}.$$

Then, for $q_0 = 0$, (5) and (4) become

$$D = \delta_1^h \phi_1 \delta_2^h \phi_1 \prod_{i=1}^{3} (\delta_i \phi_1 + \phi_2) \prod_{i=4}^{6} (\delta_i \phi_1 + \phi_3)$$

and

$$d = \text{Coef}\left[D,\ \phi_1^6\phi_2\phi_3\right].$$

If in addition the $h_i$ are quadratics, then

$$D = 2\phi_1 2\phi_1(\phi_1+\phi_2)^3(\phi_1+\phi_3)^3$$

and

$$d = \text{Coef}\left[D,\ \phi_1^6\phi_2\phi_3\right] = 36.$$

Thus, the customized $m$-homogeneous structure reduces the Bezout number for the case $q_0 = 0$ and $h_i$ quadratic from 60 to 36.

If, in fact, $q_0 \neq 0$, then further reductions are possible. Consider the case that $P_1$ is a cylinder and $P_2$ is a sphere, as follows:

$$h_1(x) = x_2^2 + x_3^2 - 1,$$

$$h_2(y) = y_1^2 + (y_2-3)^2 + y_3^2 - 1.$$

The variable $x_1$ does not appear in the constraint set. Therefore $q_0 = 1$, and

$$D = 2\phi_1 2\phi_1\ \phi_1(\phi_1+\phi_2)^2(\phi_1+\phi_3)^3$$

and

$$d = \text{Coef}\left[D,\ \phi_1^6\phi_2\phi_3\right] = 24.$$

Going back to the standard 2-homogeneous Bezout number, (7) yields

$$d = 2^2 1\binom{6-1}{6-1-2} = 40.$$

Thus, the customized $m$-homogeneous approach (yielding 24) is better than the general approach (yielding 40), but in either case it is better to exploit $q_0 = 1$ than to use $q_0 = 0$.

### 4.2. Structural design problem

Let us consider the following prototype structural design problem [10]:

$$\begin{aligned}
&\min && c_1x_1 + \cdots + c_{2k}x_{2k},\\
&\text{subject to} && x_{2i-1}^2 + x_{2i}^2 - b_i \leqslant 0,\ i = 1,\ldots,k,\\
& && \sum_{j=1}^{2k} a_{i,j}x_j = 0,\ i = 1,\ldots,s,
\end{aligned}$$

where $c_i$, $b_i > 0$, $a_{i,j}$ are constants and $k$ and $s$ are positive integers with $s \leqslant 2k$.

Introducing slack variables and our standard notation gives

$$n = 3k, \qquad q = k+s, \qquad q_0 = 0,$$

$$\theta(x) = c_1x_1 + \cdots + c_{2k}x_{2k},$$

$$h_i(x) = x_{2i-1}^2 + x_{2i}^2 + x_{2k+i}^2 - b_i, \quad i = 1,\ldots,k,$$

$$h_{k+i}(x) = \sum_{j=1}^{2k} a_{i,j}x_j, \quad i = 1,\ldots,s,$$

Table 1
Total degree, standard Bezout number and customized Bezout number for the prototype structural design problem

| $k$ | $s$ | Total degree | Standard Bezout | Customized Bezout |
|---|---|---|---|---|
| 4 | 1 | 65536 | 12672 | 6912 |
| 4 | 2 | 65536 | 14784 | 29376 |
| 4 | 3 | 65536 | 12672 | 94462 |
| 4 | 4 | 65536 | 7920 | 214080 |
| 4 | 5 | 65536 | 3520 | 314880 |
| 4 | 6 | 65536 | 1056 | 293760 |
| 4 | 7 | 65536 | 192 | 161280 |

and thus

$$(\nabla_x L)_{2i-1} = c_{2i-1} + 2r_i x_{2i-1} + \sum_{l=1}^{s} r_{k+l} a_{l,2i-1},$$

$$(\nabla_x L)_{2i} = c_{2i} + 2r_i x_{2i} + \sum_{l=1}^{s} r_{k+l} a_{l,2i},$$

$$(\nabla_x L)_{2k+i} = 2r_i x_{2k+i},$$

for $i = 1, \ldots, k$. Then

$$\delta_i^{\nabla\theta} = 0 \text{ or } -1, \quad i = 1, \ldots, n, \qquad \delta_i^{\nabla h} = 1, \quad i = 1, \ldots, n,$$

$$\delta_i = 1, \quad i = 1, \ldots, n, \qquad \delta_i^h = 2, \quad i = 1, \ldots, k, \qquad \delta_i^h = 1, \quad i = k+1, \ldots, k+s.$$

Thus by (7),

$$d = 2^k 1^s \sum_{1 \leqslant i_1 < \cdots < i_{2k-s} \leqslant n} \delta_1 \cdots \delta_{2k-s} = 2^k \binom{3k}{2k-s} = 2^k \binom{3k}{k+s}.$$

(Compare (8).) The total degree, by contrast, is

$$\text{td} = 2^{4k}.$$

$d$ and td are compared in Table 1, where this $d$ is referred to as the "standard Bezout" number.

Sometimes, one can do better with a customized Bezout breakdown. Consider the partitioning of variables:

$$\bigcup_{i=1}^{k} \{x_{2i-1}, x_{2i}, x_{2k+i}\} \bigcup_{i=1}^{k} \{r_i\} \cup \{r_{k+1}, \ldots, r_{r+s}\}.$$

To compute the combinatorial product $D$, assign dummy variables $\phi_1, \phi_2, \ldots, \phi_k$ to the first $k$ groups, $\phi_{k+1}, \phi_{k+2}, \ldots, \phi_{2k}$ to the second $k$ groups, and $\phi_{2k+1}$ to the last group. Then

$$D = 2^k \phi_1 \cdots \phi_k (\phi_1 + \cdots + \phi_k)^s \prod_{i=1}^{k} (\phi_i + \phi_{k+i} + \phi_{2k+1})^2 (\phi_i + \phi_{k+i})$$

and

$$d = \text{Coef}\left[D, \left(\prod_{i=1}^{k} \phi_i^3 \phi_{k+i}\right) \phi_{2k+1}^s\right].$$

This simplifies to

$$d = 2^k \text{Coef}\left[D', \left(\prod_{i=1}^{k} \phi_i^2 \phi_{k+i}\right) \phi_{2k+1}^s\right],$$

where

$$D' = \left[\prod_{i=1}^{k} 3\phi_i^2 \phi_{k+i} + 2\phi_i(\phi_i + 2\phi_{k+i})\phi_{2k+1} + (\phi_i + \phi_{k+i})\phi_{2k+1}^2\right](\phi_1 + \cdots + \phi_k)^s.$$

Table 1 gives the total degree, standard Bezout number and customized Bezout number for the case $k = 4$. Note that the customized Bezout number is better than the standard Bezout number for $s = 1$ but worse for $s > 1$.

## References


[1] D.P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods* (Academic Press, New York, 1982).

[2] C.B. Garcia and W.I. Zangwill, *Pathways to Solutions, Fixed Points, and Equilibria* (Prentice-Hall, Englewood Cliffs, NJ, 1981).

[3] M.M. Kostreva, Nonconvexity in noncooperative game theory, *Internat. J. Game Theory* **18** (1989) 247–259.

[4] M.M. Kostreva and L.A. Kinard, A differentiable homotopy approach for solving polynomial optimization problems and noncooperative games, Preprint, Dept. Math. Sci., Clemson Univ., SC, 1987.

[5] A.P. Morgan, *Solving Polynomial Systems Using Continuation for Scientific and Engineering Problems* (Prentice-Hall, Englewood Cliffs, NJ, 1987).

[6] A.P. Morgan and A.J. Sommese, A homotopy for solving general polynomial systems that respects $m$-homogeneous structures, *Appl. Math. Comput.* **24** (1987) 101–113.

[7] C.W. Wampler, A.P. Morgan and A.J. Sommese, Numerical continuation methods for solving polynomial systems arising in kinematics, *ASME J. Design* **112** (1990) 59–68.

[8] 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. Software* **13** (1987) 281–310.

[9] L.T. Watson and R.T. Haftka, Modern homotopy methods in optimization, Tech. Rep. 88-51, Dept. Comput. Sci., Virginia Polytechnic Institute and State Univ., Blacksburg, VA, 1988.

[10] L.T. Watson and W.H. Yang, Some numerical methods for limit analysis in continuum mechanics, *Comput. Methods Appl. Mech. Engrg.* **15** (1978) 353–364.