# Intervals Methods for Bounding the Range of Multivariate Polynomials
# — Confidential Acclaim Deliverable – *

Volker Stahl
Research Institute for Symbolic Computation
Johannes Kepler University
A-4040 Linz, Austria
vstahl@risc.uni-linz.ac.at

June 14, 1995

## Contents



*The research was done in the framework of the ACCLAIM project sponsored by European Community Basic Research Action (ESPRIT 7195) and Austrian Science Foundation (P9374-PHY).

**Abstract**

The overestimation of the range of functions is a fundamental problem which arises in many interval methods for constraint solving. We will not treat the problem in its full generality, rather we restrict ourselves to polynomials. Apart from linear functions, for which range computation is trival, polynomials are the "simplest" subclass of elementary transcendental functions. We can therefore expect more efficient methods and more refined results for polynomials than for arbitrary functions. The practical importance of polynomials is out of question, therefore it seems worthwhile to study them separately. The univariate case was considered in another paper, so we concentrate on multivariate polynomials.

# 1 Introduction

In this paper we are working on the following problem: Given a multivariate polynomial $f : \mathbb{R}^n \to \mathbb{R}$ and an interval vector $\boldsymbol{X}$, find an interval $Y$ such that $Y \supseteq \{f(\boldsymbol{x}) \mid \boldsymbol{x} \in \boldsymbol{X}\}$. This problem arises in many interval methods for constraint solving, such as Krawczyk operator [Krawczyk, 1969], Hansen–Sengypta operator [Hansen and Sengupta, 1981], Tightening [Hong and Stahl, 1994], etc. A nice survey of range computation methods is given in [Ratschek and Rokne, 1984].

Any range computation method is a compromise between efficiency and accuracy. The most efficient method is simply to return $[-\infty, \infty]$. The other extreme is to compute the range exactly by using infinite precision algebraic numbers. In this paper we review several methods, which seem to be good compromises. Most of the methods for univariate polynomials can be generalized straight forwardly. However, for efficiency it is very important that sparsity of the input polynomial is preserved during computation. Thus, we focus our attention on methods, which satisfy this important condition.

In Section 2 we fix some notational conventions. In Section 3 we discuss the Horner form and a variation of it. Section 4 contains quadratically convergent methods which are based on the mean value theorem. In Section 5 we compare the range computation algorithms at some random polynomials and by using them in a global optimization algorithm.

Throughout the paper we will refer to definitions and algorithms given in the paper "Intervals Methods for Bounding the Range of Multivariate Polynomials".

# 2 Notation

The set of real numbers is denoted by $\mathbb{R}$, the set of compact intervals over $\mathbb{R}$ is denoted by $\mathbb{IR}$. We use lower case letters to denote real numbers or functions between the reals and upper case letters to denote intervals or functions between intervals. For tuples we use bold face. The upper bound of an interval $X$ is denoted by $\overline{X}$, the lower bound by $\underline{X}$. The width of an interval is defined as

$$\mathrm{w}(X) = \overline{X} - \underline{X}.$$

The distance between two intervals is defined as

$$\mathrm{q}(X) = \max(|\overline{X} - \overline{Y}|, |\underline{X} - \underline{Y}|).$$

The magnitude of an interval is defined as

$$\mathrm{mag}(X) = \max\{|x| \mid x \in X\},$$

the mignitude is defined as

$$\mathrm{mig}(X) = \min\{|x| \mid x \in X\}.$$

The interval arithmetic operations are defined elementwise, i.e.

$$X \circ Y = \{x \circ y \mid x \in X, y \in Y\}$$

for $\circ \in \{+, -, *, /\}$.

In the following let $f(x_1, x_2, \ldots, x_n) \in \mathbb{R}[x_1, \ldots, x_n]$

$$f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{m} a_i \, x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}}$$

be a multivariate real polynomial in $n \geq 0$ variables, such that for $1 \leq i < j \leq m$

$$(d_{i,1}, d_{i,2}, \ldots, d_{i,n}) \succ (d_{j,1}, d_{j,2}, \ldots, d_{j,n}),$$

where $\succ$ means lexicographically greater.

# 3 Horner Form

The Horner Form of multivariate polynomials is defined recursively: If $n = 0$ then $f = a_1$ is a real number and the Horner Form of $f$ is $a_1$. Otherwise, we consider $f(x_1, \ldots, x_n)$ as a univariate polynomial in $x_1$ with coefficients in $\mathbb{R}[x_2, \ldots, x_n]$. By evaluating the Horner Form of the coefficients recursively, we obtain a univariate interval polynomial $F(x_1) \in \mathbb{IR}[x_1]$, and the Horner form of the multivariate polynomial $f$ is the Horner form of the univariate interval polynomial $F$.

Before we give a formal definition of the multivariate Horner form, we need some auxiliary definitions.

**Definition 3.1 (Coefficient)** *The coefficient of $x_1^d$ in $f$ is defined as*

$$\mathrm{coef}(f, d) = \sum_{\substack{i=1 \\ d_{i,1}=d}}^{m} a_i \, x_2^{d_{i,2}} x_3^{d_{i,3}} \cdots x_n^{d_{i,n}} \in \mathbb{R}[x_2, \ldots, x_n]. \ \square$$

**Definition 3.2 (Degree)** *The degree of $f$ is defined as*

$$\deg(f) = \begin{cases} d_{1,1} & \text{if } n > 0 \\ 0 & \text{else. } \square \end{cases}$$

**Definition 3.3 (Multivariate Horner Form)** *The multivariate Horner form $H_f : \mathbb{IR}^n \to \mathbb{IR}$ of $f$ is defined recursively as*

$$H_f(X_1, \ldots, X_n) = \begin{cases} a_1 & \text{if } n = 0 \\ H_F(X_1) & \text{else,} \end{cases}$$

*where*

$$F(x_1) = \sum_{i=0}^{\deg(f)} H_{\mathrm{coef}(f,i)}(X_2, \ldots, X_n) \, x_1^i \ \in \mathbb{IR}[x_1]. \ \square$$

We are using the same symbol $H$ for the univariate and the multivariate Horner form, because the forms coincide for $n = 1$.

**Theorem 3.4 (Inclusion Monotonicity)** *$H_f$ is an inclusion monotone interval extension of $f$.* $\square$

**Theorem 3.5 (Convergence)** *$H_f$ converges linearly to $f$.* $\square$

**Proof.** $H_f$ is an interval extensions of $f$ (Theorem 3.4) and Lipschitz. Hence $H_f$ converges linearly to $f$. $\square$

**Algorithm and Complexity.**

From Defintion 3.3 we obtain the following recursive algorithm for evaluating the multivariate Horner form.

**Algorithm 3.1** (HFM) [Multivariate Horner Form]

*In:* $f(x_1, \ldots, x_n) = \sum_{i=1}^{m} a_i \; x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}} \in \mathbb{F}[x_1, \ldots, x_n]$,
$\boldsymbol{X} \in \mathbb{IF}^n$.

*Out:* $\mathrm{HFM}(f, \boldsymbol{X}) \in \mathbb{IF}$, $\mathrm{HFM}(f, \boldsymbol{X}) \supseteq H_f(\boldsymbol{X})$.

(1) [Recursion Base.]
if $n = 0$ return $a_1$

(2) [Coefficients of Univariate Interval Polynomial.]
for $i = 0 \ldots \deg(f)$
$g_i(x_2, \ldots, x_n) \longleftarrow \mathrm{coef}(f, i)$
$A_i \longleftarrow \mathrm{HFM}(g_i(x_2, \ldots, x_n), X_2, \ldots, X_n)$

(3) [Evaluate Univariate Interval Polynomial.]
$F(x_1) \longleftarrow \sum_{i=0}^{\deg(f)} A_i x_1^i$
return $\mathrm{HFI}(F(x_1), X_1)$

**Theorem 3.6 (Complexity)** *Algorithm 3.1* (HFM) *costs*

$$\begin{array}{rl} nm & \textit{interval power computations,} \\ nm & \textit{interval multiplications and} \\ 2m-2 & \textit{number additions.} \; \square \end{array}$$

**Proof.** Let $\pi(m, n)$ be the number of interval power computations and interval multiplications, and let $\alpha(m, n)$ be the number of interval additions of Algorithm 3.1. Obviously Theorem 3.6 holds for $n = 0$. Let $k$ be the number of non–zero coefficients of powers of $x_1$ in $f$ and let $m_i$ be the number of monomials in the $i$-th coefficient, $i = 1, \ldots, k$. Step 2 costs $\sum_{i=1}^{k} \pi(m_i, n-1)$ power computations and multiplications and $\sum_{i=1}^{k} \alpha(m_i, n-1)$ additions. Step 3 costs $k$ power computations and multiplications and $k - 1$ additions. Hence, by induction on $n$ we obtain

$$\begin{aligned} \pi(m, n) &= \sum_{i=1}^{k} \pi(m_i, n-1) + k \\ &= \sum_{i=1}^{k} m_i(n-1) + k \\ &= m(n-1) + k \\ &\leq nm \\ \alpha(m, n) &= \sum_{i=1}^{k} \alpha(m_i, n-1) + k - 1 \\ &= \sum_{i=1}^{k} (m_i - 1) + k - 1 \\ &= m - 1. \; \square \end{aligned}$$

**Theorem 3.7** *If* $0 \notin \mathrm{int}(X_i)$ *for all* $i$, *then Algorithm 3.1* (HFM) *costs*

$$\begin{array}{rl} nm & \textit{interval power computations,} \\ 2nm & \textit{number multiplications and} \\ 2m-2 & \textit{number additions.} \; \square \end{array}$$

## 3.1 Nested Form

A simple way to obtain interval extensions $F$ of $f$ is as follows:

- Find an arithmetic expression $\mathsf{e}$ for $f$.
- Let $F(\boldsymbol{X})$ be the interval which is obtained by evaluating $\mathsf{e}$ on $\boldsymbol{X}$ using interval arithmetic.

The Horner form is obtained in this way, where the expression $\mathsf{e}$ has a particular shape. For example, let

$$f(x_1, x_2) = x_1^2 x_2^2 + x_1 x_2^2 + x_1 x_2.$$

Obviously

$$\mathsf{e} = x_1^2 x_2^2 + x_1 x_2(x_2 + 1)$$

is an expression for $f$, and the Horner form of $f$ is the interval evaluation of $\mathsf{e}$. Another expression for $f$ is

$$\mathsf{e}' = x_1 x_2(x_2(x_1 + 1) + 1).$$

Usually, the interval evaluation of $\mathsf{e}$ and $\mathsf{e}'$ yield different results. Further, it is well known that it depends on the value of $\boldsymbol{X}$ which expression gives a tighter interval. For example let

$$\boldsymbol{X} = ([0, 1], [-1, 1])$$

then

$$\mathsf{e}(\boldsymbol{X}) = [-2, 3] \subset [-3, 3] = \mathsf{e}'(\boldsymbol{X}),$$

whereas if

$$\boldsymbol{X} = ([-1, 0], [-1, 1])$$

then

$$\mathsf{e}(\boldsymbol{X}) = [-2, 3] \supset [-2, 2] = \mathsf{e}(\boldsymbol{X}).$$

It seems to be non–trivial to find an expression for given $f$ and $\boldsymbol{X}$ which yields an interval with smallest width efficiently. In the literature on optimizing compilers a simliar problem is treated. For a given expression an equivalent expression is searched, which allows evaluation with a minimal number of machine instructions, see e.g. [Sethi and Ullman, 1970], [Ullman, 1973], [Aho *et al.*, 1977]. However, the arithmetic operations are not assumed to be commutative or associative but mainly the register usage is optimized. Thus, the results cannot be applied here. Our experience is that searching an expression with minimizes the overestimation error, is very expensive and other interval extensions with comparable costs give usually tighter inclusions. Thus, instead of searching an optimal expression we are content with a suboptimum. In applications it is often the case that the same polynomial has to be evaluated over many different intervals. As the determination of a suboptimal expression can be costly, it should be done only once and not each time the polynomial is evaluated. Therefore, we are interested in an expression which gives tight inclusions for arbitrary input intervals.

Having in mind the subdistributivity law of interval arithmetic, one might be tempted to say that $\mathsf{e}'$ is always better than $\mathsf{e}$. However, this is wrong because the interval power function is not the same as repeated multiplication. On the other hand, there are several cases where an application of the distributivity rule

$$AX^a + BX^b \longrightarrow (AX^{a-c} + BX^{b-c})X^c, \quad 0 < c \leq \min(a, b) \tag{3.1}$$

leads to better expressions.

**Theorem 3.8** *Let $A, B, X \in \mathbb{IR}$, let $a, b, c \in \mathbb{N}$ such that $0 < c < \min(a, b)$. Let*

$$\begin{aligned} Y_1 &= AX^a + BX^b \\ Y_2 &= (AX^{a-c} + BX^{b-c})X^c \\ Y_3 &= (AX^{a-\min(a,b)} + BX^{b-\min(a,b)})X^{\min(a,b)}. \end{aligned}$$

1. *If $0 \notin \mathrm{int}(X)$ or $a, b$ are both even then $Y_3 \subseteq Y_2 \subseteq Y_1$*

2. *If $a, b$ are both odd and $Y_2 \subset Y_1$ then $Y_3 \subset Y_2$.*

**Remark.** The condition $Y_2 \subset Y_1$ in (2) cannot be replaced by $Y_2 \subseteq Y_1$ as the following example shows: Let

$$A = [-6, 6], B = [-2, 8] \text{ and } X = [-4, 3].$$

We obtain

$$AX^3 + BX^7 = [-131456, 33152] = (AX + BX^5)X^2,$$

but

$$(A + BX^4)X^3 = [-131456, 55458] \supset (AX + BX^5)X^2.$$

Further, the condition that both $a$ and $b$ are odd in (2) is necessary as the following example shows, where $a$ is even and $b$ is odd. Let

$$A = [-1, 0], B = [-3, 3] \text{ and } X = [-2, 1].$$

We obtain

$$AX^3 + BX^2 = [-13, 20] \supset [-12, 20] = (AX^2 + BX)X,$$

but

$$(AX + B)X^2 = [-16, 20] \supset (AX^2 + BX)X.$$

Thus, it seems reasonable to search for expressions which are highly nested. We give a simple algorithm which computes such an expressions for a given polynomial.

**Algorithm and Complexity.**

**Algorithm 3.2** (NEP) [Nested Expression for Polynomial]

*In:* $\mathsf{e} = \sum_{i=1}^{m} a_i\, x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}}$, an expression.

*Out:* NEP($\mathsf{e}$), an expression equivalent to $\mathsf{e}$ up to commutativity, associativity and distributivity.

(1) [Initialize.]
$\mathcal{L} \longleftarrow \text{List}(a_i\, x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}} \mid i = 1, \ldots, m)$

(2) [Iterate.]
The elements of $\mathcal{L}$ are expressions of the form $\mathsf{a}\, x_1^{d_1} x_2^{d_2}, \cdots x_n^{d_n}$ where $\mathsf{a}$ is a (compound) expression.
while Length($\mathcal{L}$) $> 1$

(2.1) [Search.]
Choose $\mathsf{e}' = \mathsf{a}' x_1^{d_1'} x_2^{d_2'}, \cdots x_n^{d_n'}$ and $\mathsf{e}'' = \mathsf{a}'' x_1^{d_1''} x_2^{d_2''}, \cdots x_n^{d_n''}$ from $\mathcal{L}$ such that $\sum_{i=1}^{n} \min(d_i', d_i'')$ is maximal. Remove $\mathsf{e}'$ and $\mathsf{e}''$ from $\mathcal{L}$.

(2.3) [Combine.]
for $i = 1, \ldots n$ do $d_i \longleftarrow \min(d_i', d_i'')$
$\mathsf{e} \longleftarrow (\mathsf{a}' \prod_{\substack{i=1 \\ d_i' \neq d_i}}^{n} x_i^{d_i' - d_i} + \mathsf{a}' \prod_{\substack{i=1 \\ d_i'' \neq d_i}}^{n} x_i^{d_i'' - d_i}) x_1^{d_1} x_2^{d_2} \cdots x_n^{d_n}.$
Add $\mathsf{e}$ to $\mathcal{L}$.

(3) [Return.]
return First($\mathcal{L}$).

Note that Step 2.1 is non–deterministic. For example, if $\mathsf{e} = x_1 x_2 + x_1 + x_2$ then Algorithm 3.2 (NEP) may either return $x_1(x_2 + 1) + x_2$ or $x_2(x_1 + 1) + x_1$. In the sequel we assume that an arbitrary but fixed strategy is used.

**Definition 3.9 (Nested Form)** *The nested form $N_f : \mathbb{IR}^n \to \mathbb{IR}$ of $f$ is the interval evaluation of the expression obtained by Algorithm 3.2 (*NEP*) with input expression*

$$\sum_{i=1}^{m} a_i\, x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}}. \;\square$$

An algorithm for evaluating the nested form is now straight forward.

**Algorithm 3.3** (NF) [Nested Form]

*In:* $f(x_1, \ldots, x_n) = \sum_{i=1}^{m} a_i\, x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}} \in \mathbb{F}[x_1, \ldots, x_n]$,
$\boldsymbol{X} \in \mathbb{IF}^n$.

*Out:* $\mathrm{NF}(f, \boldsymbol{X}) \in \mathbb{IF}$, $\mathrm{NF}(f, \boldsymbol{X}) \supseteq N_f(\boldsymbol{X})$.

(1) [Nested Expression.]
$\mathsf{e} \longleftarrow \mathrm{NF}(\sum_{i=1}^{m} a_i\, x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}})$

(2) [Evaluate.]
$Y \longleftarrow$ evaluation of $\mathsf{e}$ on $\boldsymbol{X}$ using floating point interval arithmetic.

(3) [Return.]
return $Y$

**Remark.** In the univariate case the nested form and the Horner form are identical.

**Remark.** If $N_f$ has to be computed repeatedly for different intervals $\boldsymbol{X}$ using Algorithm 3.3 (NF), then Step 1 has to be executed only once, i.e. $\mathsf{e}$ can be pre–computed.

**Theorem 3.10 (Complexity)** *Algorithm 3.3 (*NF*) costs*

$$\begin{array}{rl} nm & \textit{interval power computations,} \\ nm & \textit{interval multiplications and} \\ 2m-2 & \textit{number additions.}\ \square \end{array}$$

**Proof.** Step 1 involves merely symbolic and integer operations. For the cost of Step 2 we have to count the number of operation symbols in the expression $\mathsf{e}$. Therefore, we count the number of operation symbols of all expressions in $\mathcal{L}$ of Algorithm 3.2 (NEP). Upon termination of this algorithm $\mathcal{L}$ contains the single element $\mathsf{e}$. In Step 1, every element of $\mathcal{L}$ comprises $n$ power computations and $n$ multiplications, hence there are $nm$ power computations and $nm$ multiplications. In each iteration in Step 2 we remove in Step 2.1 the operations for $\mathsf{a}', \mathsf{a}''$, $2n$ power computations and $2n$ multiplications and add in Step 2.2 the operations for $\mathsf{a}', \mathsf{a}''$, at most $2n$ power computations, at most $2n$ multiplications and 1 addition. Hence, in each iteration in Step 2 we add 1 addition and, in the worst case, keep the number of power computations and multiplications constant. As $m-1$ iterations are performed, we add $m-1$ additions. Hence, upon termination there are at most $nm$ power computations, $nm$ multiplications and $m-1$ additions in $\mathcal{L}$.

**Remark.** Comparing Theorem 3.10 and Theorem 3.6 one might expect that Algorithm 3.3 (NF) and Algorithm 3.1 (HFM) are equally expensive. However, the exponents of the power computations in Algorithm (NF) are usually smaller than in Algorithm (HFM), which means that "at the average" Algorithm (NF) costs less arithmetic floating point operations than Algorithm (HFM). $\square$

## 3.2 Experimental Results

In this section we compare the Horner form $H_f$, the nested form $N_f$ and the distributed form $D_f$ which is defined as

$$D_f(\boldsymbol{X}) = \sum_{i=1}^{n} a_i\, X_1^{d_{i,1}} X_2^{d_{i,2}} \cdots X_n^{d_{i,n}}.$$

The distributed form is the evaluation of the "least nested" expression for $f$, hence $D_f$ and $N_f$ can be viewed as two extremes where $H_f$ is in between.

As the computation of the range of a multivariate polynomial is expensive, we do not report overestimation errors. Instead, we compare the width of the nested and the dense form with the width of the Horner form. The costs are given in terms of arithmetic floating point instructions. The coefficients of the test polynomials are uniformly distributed in $[-1, 1]$. The endpoints of the components of the input interval vector are uniformly distributed in $[-1, 1]$. We report the average over 1000 random polynomials and input interval vectors. Table 3.1 shows a comparison for random polynomials with different numbers of variables $n$ and monomials $m$. Each degree vector contains $n$ non–zero components and the degree in each variable is at most 5.

*Degree 5, Dense Monomials*

| Variables $n$, Monomials $m$ | $2,3$ | $2,6$ | $4,4$ | $4,8$ | $8,5$ | $8,10$ | $16,6$ | $16,12$ |
|---|---|---|---|---|---|---|---|---|
| Flops for $H_f(\boldsymbol{X})$ | 31.8 | 53.6 | 87.5 | 157.7 | 232.1 | 440.8 | 589.5 | 1148.6 |
| Flops for $N_f(\boldsymbol{X})$ | 22.8 | 36.3 | 53.2 | 86.9 | 127.0 | 215.9 | 300.9 | 531.1 |
| Flops for $D_f(\boldsymbol{X})$ | 40.9 | 82.6 | 104.5 | 208.0 | 258.7 | 515.9 | 625.8 | 1251.5 |
| $\mathrm{w}(N_f(\boldsymbol{X}))/\mathrm{w}(H_f(\boldsymbol{X}))$ | 0.960 | 0.937 | 1.016 | 0.996 | 1.066 | 1.051 | 1.061 | 1.056 |
| $\mathrm{w}(D_f(\boldsymbol{X}))/\mathrm{w}(H_f(\boldsymbol{X}))$ | 1.086 | 1.205 | 0.989 | 0.997 | 0.985 | 0.982 | 0.988 | 0.990 |

Table 3.1: Comparison of $H_f$, $N_f$ and $D_f$ for random polynomials with different numbers of variables and monomials. Each degree vector contains $n$ non–zero components and the degree in each variable is at most 5.

The accuracy of all three forms is roughly the same. However, the cost of $N_f$ is significantly smaller than for $H_f$, in particular if the number of monomials is large.

In fact, we expected that the accuracy of $N_f$ would be much better than the accuracy of $H_f$. According to Theorem 3.8, $N_f$ is at least as accurate as $D_f$ if $0 \notin X_i$ for all $i$. Thus, in Table 3.2 we repeat the experiment, but consider only cases where this condition holds. The result is disappointing. In general,

*Degree 5, Dense Monomials, $0 \notin X_i$*

| Variables $n$, Monomials $m$ | $2,3$ | $2,6$ | $4,4$ | $4,8$ | $8,5$ | $8,10$ | $16,6$ | $16,12$ |
|---|---|---|---|---|---|---|---|---|
| Flops for $H_f(\boldsymbol{X})$ | 33.2 | 54.6 | 90.7 | 162.5 | 237.6 | 449.7 | 588.1 | 1146.2 |
| Flops for $N_f(\boldsymbol{X})$ | 23.5 | 35.9 | 52.7 | 84.3 | 121.9 | 204.4 | 283.1 | 495.3 |
| Flops for $D_f(\boldsymbol{X})$ | 44.6 | 88.7 | 110.6 | 221.0 | 265.9 | 531.5 | 626.8 | 1253.3 |
| $\mathrm{w}(N_f(\boldsymbol{X}))/\mathrm{w}(H_f(\boldsymbol{X}))$ | 0.903 | 0.882 | 0.907 | 0.864 | 0.971 | 0.962 | 0.998 | 0.997 |
| $\mathrm{w}(D_f(\boldsymbol{X}))/\mathrm{w}(H_f(\boldsymbol{X}))$ | 1.166 | 1.334 | 1.029 | 1.054 | 1.000 | 1.001 | 1.000 | 1.000 |

Table 3.2: Same experiment as above, but this time $0 \notin X_i$ for all $i$.

$N_f$ is only slightly more accurate than $D_f$ and $H_f$.

Table 3.3 shows the same experiment as in Table 3.1, but this time each degree vector has at most 2 non–zero components. Qualitatively, we obtain the same result concerning accuracy and costs.

Finally, in Table 3.4 we repeat the experiment of Table 3.3, but this time the degree in each variable is up to 15. Again, the result is qualitatively the same as in the previous experiments.

Thus, we conclude that $N_f$ is in general not more accurate than $H_f$ but costs significantly less in terms of arithmetic floating point operations. The non–numeric overhead for computing the nested expression for $N_f$ pays off if $f$ has to be evaluated several times for different input intervals.

*Degree 5, Sparse Monomials*

| Variables $n$, Monomials $m$ | $2,3$ | $2,6$ | $4,4$ | $4,8$ | $8,5$ | $8,10$ | $16,6$ | $16,12$ |
|---|---|---|---|---|---|---|---|---|
| Flops for $H_f(\boldsymbol{X})$ | 31.8 | 53.6 | 62.3 | 113.7 | 114.9 | 211.8 | 218.7 | 404.2 |
| Flops for $N_f(\boldsymbol{X})$ | 22.8 | 36.3 | 47.0 | 77.5 | 70.7 | 116.1 | 102.8 | 163.3 |
| Flops for $D_f(\boldsymbol{X})$ | 40.9 | 82.6 | 69.5 | 140.1 | 124.1 | 248.1 | 238.8 | 476.1 |
| $\mathrm{w}(N_f(\boldsymbol{X}))/\mathrm{w}(H_f(\boldsymbol{X}))$ | 0.960 | 0.937 | 0.993 | 0.982 | 0.993 | 0.982 | 0.995 | 0.991 |
| $\mathrm{w}(D_f(\boldsymbol{X}))/\mathrm{w}(H_f(\boldsymbol{X}))$ | 1.086 | 1.205 | 1.020 | 1.054 | 1.012 | 1.019 | 1.009 | 1.010 |

Table 3.3: Same experiment as above, but this time each degree vector has at most 2 non–zero components.

*Degree 15, Sparse Monomials*

| Variables $n$, Monomials $m$ | $2,3$ | $2,6$ | $4,4$ | $4,8$ | $8,5$ | $8,10$ | $16,6$ | $16,12$ |
|---|---|---|---|---|---|---|---|---|
| Flops for $H_f(\boldsymbol{X})$ | 53.0 | 93.5 | 90.9 | 166.7 | 149.6 | 278.7 | 259.2 | 483.5 |
| Flops for $N_f(\boldsymbol{X})$ | 43.2 | 72.1 | 74.7 | 130.1 | 105.6 | 184.0 | 144.3 | 246.1 |
| Flops for $D_f(\boldsymbol{X})$ | 62.3 | 125.4 | 98.4 | 195.6 | 159.5 | 317.8 | 278.2 | 556.6 |
| $\mathrm{w}(N_f(\boldsymbol{X}))/\mathrm{w}(H_f(\boldsymbol{X}))$ | 0.978 | 0.978 | 0.991 | 0.991 | 0.998 | 0.995 | 1.004 | 1.003 |
| $\mathrm{w}(D_f(\boldsymbol{X}))/\mathrm{w}(H_f(\boldsymbol{X}))$ | 1.014 | 1.028 | 0.995 | 1.007 | 1.000 | 1.000 | 0.999 | 0.998 |

Table 3.4: Same experiment as above, but this time the degree in each variable is at most 15 and each degree vector has at most 2 non–zero components.

# 4 Mean Value Form

The mean value form of multivariate polynomials is a straight forward generalization of the univariate case. It is quadratically convergent ([Alefeld and Herzberger, 1974], [Skelboe, 1974], [Caprani and Madsen, 1980], [Alefeld and Herzberger, 1983]) and inclusion monotone ([Caprani and Madsen, 1980]).

An important improvement of the mean value form due to [Hansen, 1968] is subject of Section 4.1. In Section 4.3 we generalize the bicentered mean value form to the multivariate case.

The multivariate mean value form can be derived easily from the mean value Theorem: For all $\boldsymbol{X} \in \mathbb{IR}^n$ and for all $\boldsymbol{x}, \boldsymbol{c} \in \boldsymbol{X}$ there exists $\boldsymbol{\xi} \in \boldsymbol{X}$ such that

$$f(\boldsymbol{x}) = f(\boldsymbol{c}) + \sum_{i=1}^{n} \partial_i f(\boldsymbol{\xi})(x_i - c_i),$$

where

$$\partial_i f = \frac{\partial f}{\partial x_i}.$$

Hence,

$$f(\boldsymbol{x}) \in f(\boldsymbol{c}) + \sum_{i=1}^{n} \partial_i f(\boldsymbol{X})(X_i - c_i).$$

Thus, every interval extension $F_i$ of $\partial_i f$, $i = 1, \ldots, n$ and any choice of $\boldsymbol{c} \in \boldsymbol{X}$ give rise to an interval extension $F$ of $f$:

$$F(\boldsymbol{X}) = f(\boldsymbol{c}) + \sum_{i=1}^{n} F_i(\boldsymbol{X})(X_i - c_i) \supseteq f(\boldsymbol{X}). \tag{4.1}$$

The mean value form is a special case of (4.1), where $F_i$ is the Horner form of $\partial_i f$ and $\boldsymbol{c} = \mathrm{mid}(\boldsymbol{X})$.

**Definition 4.1 (Multivariate Mean Value Form)** *The multivariate mean value form $M_f : \mathbb{IR}^n \to \mathbb{IR}$ is defined as*

$$M_f(\boldsymbol{X}) = f(\text{mid}(\boldsymbol{X})) + \sum_{i=1}^{n} H_{\partial_i f}(\boldsymbol{X})(X_i - \text{mid}(X_i)). \ \Box$$

**Theorem 4.2 (Interval Extension)** *$M_f$ is an interval extension of $f$.* $\Box$

**Proof.** Follows from (4.1). $\Box$

The following theorem is taken from [Caprani and Madsen, 1980].

**Theorem 4.3 (Inclusion Monotonicity)** *$M_f$ is inclusion montone.* $\Box$

**Proof.** Let $\boldsymbol{X}_\diamond \subseteq \boldsymbol{X}$, $\boldsymbol{c}_\diamond = \text{mid}(\boldsymbol{X}_\diamond)$, $\boldsymbol{c} = \text{mid}(\boldsymbol{X})$, $\boldsymbol{r}_\diamond = \text{rad}(\boldsymbol{X}_\diamond)$, $\boldsymbol{r} = \text{rad}(\boldsymbol{X})$. We have to show that $M_f(\boldsymbol{X}_\diamond) \subseteq M_f(\boldsymbol{X})$. As $X_i - c_i$ and $X_{i\diamond} - c_{i\diamond}$ are centered intervals for all $i$, it holds that

$$\begin{aligned} M_f(\boldsymbol{X}) &= f(\boldsymbol{c}) + \sum_{i=1}^{n} \text{mag}(H_{\partial_i f}(\boldsymbol{X})) r_i [-1,1] \\ M_f(\boldsymbol{X}_\diamond) &= f(\boldsymbol{c}_\diamond) + \sum_{i=1}^{n} \text{mag}(H_{\partial_i f}(\boldsymbol{X}_\diamond)) r_{i\diamond} [-1,1]. \end{aligned}$$

Hence, we have to show that

$$\begin{aligned} f(\boldsymbol{c}_\diamond) + \sum_{i=1}^{n} \text{mag}(H_{\partial_i f}(\boldsymbol{X}_\diamond)) r_{i\diamond} &\leq f(\boldsymbol{c}) + \sum_{i=1}^{n} \text{mag}(H_{\partial_i f}(\boldsymbol{X})) r_i \\ f(\boldsymbol{c}_\diamond) - \sum_{i=1}^{n} \text{mag}(H_{\partial_i f}(\boldsymbol{X}_\diamond)) r_{i\diamond} &\geq f(\boldsymbol{c}) - \sum_{i=1}^{n} \text{mag}(H_{\partial_i f}(\boldsymbol{X})) r_i. \end{aligned}$$

Note that for all $i$

$$\begin{aligned} |c_{i\diamond} - c_i| &= 1/2 |(\overline{X_{i\diamond}} + \underline{X_{i\diamond}} - \overline{X}_i - \underline{X}_i)| \\ &= 1/2 \max\{\overline{X_{i\diamond}} + \underline{X_{i\diamond}} - \overline{X}_i - \underline{X}_i, \overline{X}_i + \underline{X}_i - \overline{X_{i\diamond}} - \underline{X_{i\diamond}}\} \\ &\leq 1/2(\overline{X}_i + \underline{X_{i\diamond}} - \overline{X_{i\diamond}} - \underline{X}_i) \\ &= r_i - r_{i\diamond}. \end{aligned}$$

Further, $H_{\partial_i f}(\boldsymbol{X}_\diamond) \subseteq H_{\partial_i f}(\boldsymbol{X})$. Let $\boldsymbol{\xi} \in \boldsymbol{X}$ such that

$$f(\boldsymbol{c}_\diamond) = f(\boldsymbol{c}) + \sum_{i=1}^{n} \partial_i f(\boldsymbol{\xi})(c_{i\diamond} - c_i).$$

Then

$$\begin{aligned} f(\boldsymbol{c}_\diamond) + \sum_{i=1}^{n} \text{mag}(H_{\partial_i f}(\boldsymbol{X}_\diamond)) r_{i\diamond} &= f(\boldsymbol{c}) + \sum_{i=1}^{n} \partial_i f(\boldsymbol{\xi})(c_{i\diamond} - c_i) + \text{mag}(H_{\partial_i f}(\boldsymbol{X}_\diamond)) r_{i\diamond} \\ &\leq f(\boldsymbol{c}) + \sum_{i=1}^{n} |\partial_i f(\boldsymbol{\xi})||(c_{i\diamond} - c_i)| + \text{mag}(H_{\partial_i f}(\boldsymbol{X}_\diamond)) r_{i\diamond} \\ &\leq f(\boldsymbol{c}) + \sum_{i=1}^{n} \text{mag}(H_{\partial_i f}(\boldsymbol{X}))(r_i - r_{i\diamond}) + \text{mag}(H_{\partial_i f}(\boldsymbol{X})) r_{i\diamond} \\ &= f(\boldsymbol{c}) + \sum_{i=1}^{n} \text{mag}(H_{\partial_i f}(\boldsymbol{X})) r_i \\ f(\boldsymbol{c}_\diamond) - \sum_{i=1}^{n} \text{mag}(H_{\partial_i f}(\boldsymbol{X}_\diamond)) r_{i\diamond} &= f(\boldsymbol{c}) + \sum_{i=1}^{n} \partial_i f(\boldsymbol{\xi})(c_{i\diamond} - c_i) - \text{mag}(H_{\partial_i f}(\boldsymbol{X}_\diamond)) r_{i\diamond} \end{aligned}$$

$$\begin{aligned}
&\geq f(\boldsymbol{c}) + \sum_{i=1}^{n} -|\partial_i f(\boldsymbol{\xi})||(c_{i\diamondsuit} - c_i)| - \mathrm{mag}(H_{\partial_i f}(\boldsymbol{X}_{\diamondsuit}))r_{i\diamondsuit} \\
&\geq f(\boldsymbol{c}) + \sum_{i=1}^{n} -\mathrm{mag}(H_{\partial_i f}(\boldsymbol{X}))(r_i - r_{i\diamondsuit}) - \mathrm{mag}(H_{\partial_i f}(\boldsymbol{X}))r_{i\diamondsuit} \\
&= f(\boldsymbol{c}) - \sum_{i=1}^{n} \mathrm{mag}(H_{\partial_i f}(\boldsymbol{X}))r_i. \ \square
\end{aligned}$$

**Algorithm and Complexity.**

Algorithm 4.1 (MFM) for evaluating the multivariate mean value form follows immediately from Definition 4.1. The algorithm requires evaluation of the Horner form of the partial derivatives $\partial_i f$, $i = 1, \ldots, n$. As the coefficients of $\partial_i f$ need not be floating point numbers, we have to enclose the coefficients of the $\partial_i f$ by intervals. The Horner form of multivariate interval polynomials is computed by algorithm HFMI, which is a trivial modification of algorithm HFM and which is therefore not given explicitly. Further, we extend algorithms MID and CONVERT to interval vectors componentwise.

**Algorithm 4.1** (MFM) [Multivariate Mean Value Form]

*In:* $f(\boldsymbol{x}) = \sum_{i=1}^{m} a_i \, x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}} \in \mathbb{F}[x_1, \ldots, x_n]$,
$\boldsymbol{X} \in \mathbb{IF}^n$.

*Out:* $\mathrm{MFM}(f, \boldsymbol{X}) \in \mathbb{IF}$, $\mathrm{MFM}(f, \boldsymbol{X}) \supseteq M_f(\boldsymbol{X})$.

(1) [Midpoint.]
$\boldsymbol{c} \longleftarrow \mathrm{MID}(\boldsymbol{X})$

(2) [Evaluate $f(\boldsymbol{c})$.]
$Y \longleftarrow \mathrm{HFM}(f(\boldsymbol{x}), [\boldsymbol{c}])$

(3) [Partial Derivatives.]
for $i = 1, \ldots, n$ do $F_i(\boldsymbol{x}) \longleftarrow \frac{\partial f}{\partial x_i}$ (interval arithmetic for coefficients)

(4) [Evaluate Partial Derivatives.]
for $i = 1, \ldots, n$ do $F_i \longleftarrow \mathrm{HFMI}(F_i(\boldsymbol{x}), \boldsymbol{X})$

(5) [Accumulate.]
for $i = 1, \ldots, n$ do $Y \longleftarrow Y + F_i * (X_i - c_i)$

(6) [Return.]
return $Y$

**Theorem 4.4 (Complexity)** *Algorithm 4.1* (MFM) *costs*

$$\begin{array}{rl}
n^2m + nm & \textit{interval power computations,} \\
n^2m + n & \textit{interval multiplications,} \\
4nm + n & \textit{number multiplications and} \\
2nm + 4n + 2m - 2 & \textit{number additions.} \ \square
\end{array}$$

**Proof.**

- Step 1 costs $n$ number multiplications and $2n$ number additions.
- Step 2 costs $nm$ interval power computations, $2nm$ number multiplications and $2m - 2$ number additions.
- Step 3 costs $2nm$ number multiplications.

- Step 4 costs $n^2m$ interval power computations, $n^2m$ interval multiplications and $2nm-2n$ number additions.
- Step 5 costs $n$ interval multiplications and $4n$ number additions. $\square$

In Algorithm 4.1 (MFM) we explicitly computed polynomials for the partial derivatives of $f$ and evaluated all of them on $\boldsymbol{X}$. Another possibility for obtaining the value of the $i$-th partial derivative is to evaluate $f$ first in the variables $x_{i+1},\ldots,x_n$, next compute the formal $i$-th partial derivative of this $i$-variate interval polynomial and evaluate it. This can be done successively as follows. Let

$$F^{(n)}(x_1,\ldots,x_n) = f(x_1,\ldots,x_n).$$

For $i = n-1\ldots 0$ evaluate $F^{(i+1)}(x_1,\ldots,x_{i+1})$ in its last variable $x_i$ by $X_i$ using the Horner scheme and obtain an $i$-variate interval polynomial $F^{(i)}(x_1,\ldots,x_i)$.

$$F^{(i)}(x_1,\ldots,x_i) = F^{(i+1)}(x_1,\ldots,x_i,X_{i+1}).$$

Next, let $F_i^{(i)}$ be the formal $i$-th partial derivative of $F^{(i)}$ and evaluate $F_i^{(i)}$ on $X_1,\ldots,X_n$ using the Horner scheme. One checks that

$$H_{F_i^{(i)}}(X_1,\ldots,X_i) = H_{\partial_i f}(X_1,\ldots,X_n).$$

An advantage of this method is that $F^0() = H_f(\boldsymbol{X})$, i.e. we obtain the Horner form as a side product with minimal additional costs, such that we can compute easily $M_f(\boldsymbol{X})\cap H_f(\boldsymbol{X})$.

**Definition 4.5 (Mean Value – Horner Form)** *The mean value – Horner form* $M_f^{(\mathrm{H})} : \mathbb{IR}^n \longrightarrow \mathbb{IR}$ *is defined as*

$$M_f^{(\mathrm{H})}(\boldsymbol{X}) = M_f(\boldsymbol{X})\cap H_f(\boldsymbol{X}). \ \square$$

We formalize the procedure described above in Algorithm 4.3 (MHF) and compare its complexity against Algorithm 4.1 (MFM). First, we give an auxiliary algorithm for evaluating an interval polynomial in its last variable.

**Algorithm 4.2** (HFIL) [Horner Form in Last Variable]

*In:* $F(\boldsymbol{x}) = \sum_{i=1}^m A_i\, x_1^{d_{i,1}} x_2^{d_{i,2}}\cdots x_n^{d_{i,n}} \in \mathbb{IF}[x_1,\ldots,x_n]$
such that $(d_{i,1},d_{i,2},\ldots,d_{i,n}) \succ (d_{j,1},d_{j,2},\ldots,d_{j,n})$ for $i<j$,
$X\in\mathbb{IF}$.

*Out:* $\mathrm{HFIL}(F,X)\in\mathbb{IF}[x_1,\ldots,x_{n-1}]$, $\mathrm{HFIL}(F,X)\supseteq H_F(x_1,\ldots,x_{n-1},X)$.

(1) [Partition $f$.]
Let

$$k = |\{(d_{i,1},d_{i,2},\ldots,d_{i,n-1}) \mid i=1,\ldots,m\}|.$$

and let $1 = m_1 < \ldots < m_{k+1} = m+1$ such that

$$F_j(x_n) = \sum_{i=m_j}^{m_{j+1}-1} A_i x_n^{d_{i,n}} \in \mathbb{IF}[x_n]$$

and

$$F(\boldsymbol{x}) = \sum_{j=1}^{k} F_j(x_n)\, x_1^{d_{m_j,1}} x_2^{d_{m_j,2}}\cdots x_{n-1}^{d_{m_j,n-1}}.$$

(2) [Evaluate.]
for $j=1,\ldots,k$ do $B_j \longleftarrow \mathrm{HFI}(F_j(x_n),X)$

(3) [Return.]
return $\sum_{j=1}^k B_j\, x_1^{d_{m_j,1}} x_2^{d_{m_j,2}}\cdots x_{n-1}^{d_{m_j,n-1}}$

**Theorem 4.6 (Complexity)** *Algorithm 4.2* (HFIL) *costs*

$$\begin{array}{rl} m & \textit{interval power computations,} \\ m & \textit{interval multiplications and} \\ 2m-2k & \textit{number additions,} \end{array}$$

*where*

$$k = |\{(d_{i,1}, d_{i,2}, \ldots, d_{i,n-1}) \mid i = 1, \ldots, m\}|. \ \square$$

**Proof.** The computation of $\mathrm{HFI}(F_j, X)$ in Step 2 costs $m_{j+1}-m_j$ interval power computations, $m_{j+1}-m_j$ multiplications and $m_{j+1}-m_j-1$ interval additions. Hence, Step 2 costs

$$\sum_{j=1}^{k} m_{j+1} - m_j = m_{k+1} - m_1 = m$$

interval power computations and multiplications and

$$\sum_{j=1}^{k} m_{j+1} - m_j - 1 = m_{k+1} - m_1 - k = m - k$$

interval additions. $\square$

**Theorem 4.7 (Complexity)** *If* $0 \notin \mathrm{int}(X)$ *then Algorithm 4.2* (HFIL) *costs*

$$\begin{array}{rl} m & \textit{interval power computations,} \\ 2m & \textit{number multiplications and} \\ 2m-2k & \textit{number additions.} \ \square \end{array}$$

**Algorithm 4.3 (MHF)** [Mean Value – Horner Form]

*In:* $f(\boldsymbol{x}) = \sum_{i=1}^{m} a_i\, x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}} \in \mathbb{F}[x_1, \ldots, x_n]$,
$\boldsymbol{X} \in \mathbb{IF}^n$.

*Out:* $\mathrm{MHF}(f, \boldsymbol{X}) \in \mathbb{IF}$, $\mathrm{MHF}(f, \boldsymbol{X}) \supseteq M_f^{(\mathrm{H})}(\boldsymbol{X})$.

(1) [Midpoint.]
$\boldsymbol{c} \longleftarrow \mathrm{mid}(\boldsymbol{X})$

(2) [Evaluate $f(\boldsymbol{c})$.]
$Y \longleftarrow \mathrm{HFM}(f(\boldsymbol{x}), [\boldsymbol{c}])$

(3) [Convert $f$ to an Interval Polynomial.]
$F^{(n)}(x_1, \ldots, x_n) \longleftarrow f(x_1, \ldots, x_n)$

(4) [Evaluate $f$ successively in $\boldsymbol{X}$.]
for $i = n, \ldots, 1$ do $F^{(i-1)}(x_1, \ldots, x_{i-1}) \longleftarrow \mathrm{HFIL}(F^{(i)}(x_1, \ldots, x_i), X_i)$

(5) [Partial Derivatives.]
for $i = n, \ldots, 1$ do $F_i^{(i)}(x_1, \ldots, x_i) \longleftarrow \frac{\partial}{\partial x_i} F^{(i)}(x_1, \ldots, x_i)$

(6) [Evaluate Partial Derivatives.]
for $i = n, \ldots, 1$ do $F_i \longleftarrow \mathrm{HFMI}(F_i^{(i)}(x_1, \ldots, x_i), X_1, \ldots, X_i)$

(6) [Accumulate.]
for $i = n, \ldots, 1$ do $Y \longleftarrow Y + F_i * (X_i - c_i)$

(7) [Intersect with $H_f(\boldsymbol{X})$.]
$Y \longleftarrow Y \cap F^{(0)}()$

(8) [Return.]
return $Y$

**Theorem 4.8 (Complexity)** *Algorithm 4.3 (MHF) costs*

$$\begin{array}{rl} 1/2n^2m + 5/2nm & \textit{interval power computations,} \\ 1/2n^2m + 3/2nm & \textit{interval multiplications,} \\ 4nm + n & \textit{number multiplications and} \\ 2nm + 4n + 4m - 4 & \textit{number additions.} \ \Box \end{array}$$

**Proof.** We assume the worst case where all $F^{(i)}$ have $m$ monomials, i.e. no two degree vectors of $f$ have a common prefix.

- Step 1 costs $n$ number multiplications and $2n$ number additions.
- Step 2 costs $nm$ interval power computations, $2nm$ number multiplications and $2m - 2$ number additions.
- Step 4 costs $nm$ interval power computations, $nm$ interval multiplications and $2m - 2$ number additions.
- Step 5 costs $2nm$ number multiplications.
- The $i$-th iteration in Step 6 costs $im$ interval power computations, $im$ interval multiplications and $2m - 2$ number additions. Hence, Step 6 costs $1/2n(n + 1)m$ interval power computations, $1/2n(n + 1)m$ interval multiplications and $2nm - 2n$ number additions.
- Step 7 costs $n$ interval multiplications and $4n$ number additions. □

Thus, asymtotically Algorithm 4.3 (MHF) is twice as expensive as Algorithm 4.1 (MFM). However, the average case is rather different as the experiments in Section 3.2 will show.

## 4.1 Successive Mean Value Form

An important improvement of the mean value form is presented in [Hansen, 1968]. Recall that for the mean value form we have to evaluate the partial derivatives of $f$ on $(X_1, \dots, X_n)$. If, instead, we evaluate the $i$-th partial derivative only on

$$(X_1, \dots, X_i, c_{i+1}, \dots, c_n) \subseteq (X_1, \dots, X_n),$$

we still obtain an interval extension, which is called successive mean value form. Obviously, the successive mean value form is usually tighter than the mean value form.

In order to show that the successive mean value form is an interval extension, we apply the mean value Theorem successively to the variables of $f$. Let $\boldsymbol{X} \in \mathbb{IR}^n$ and let $\boldsymbol{x}, \boldsymbol{c} \in \boldsymbol{X}$. First, consider $f(x_1, \dots, x_n)$ as a univariate function in $x_n$. According to the mean value Theorem there exists $\xi_n \in X_n$ such that

$$f(x_1, \dots, x_n) = f(x_1, \dots, x_{n-1}, c_n) + \partial_n f(x_1, \dots, x_{n-1}, \xi_n)(x_n - c_n).$$

Next, consider $f(x_1, \dots, x_{n-1}, c_n)$ as a univariate function in $x_{n-1}$. Applying the mean value Theorem again, we obtain

$$f(x_1, \dots, x_{n-1}, c_n) = f(x_1, \dots, x_{n-2}, c_{n-1}, c_n) + \partial_{n-1} f(x_1, \dots, x_{n-2}, \xi_{n-1}, c_n)(x_{n-1} - c_{n-1})$$

for some $\xi_{n-1} \in X_{n-1}$. Hence,

$$\begin{array}{rcl} f(x_1, \dots, x_n) & = & f(x_1, \dots, x_{n-2}, c_{n-1}, c_n) \\ & + & \partial_{n-1} f(x_1, \dots, x_{n-2}, \xi_{n-1}, c_n)(x_{n-1} - c_{n-1}) \\ & + & \partial_n f(x_1, \dots, x_{n-1}, \xi_n)(x_n - c_n). \end{array}$$

Next, we apply the mean value Theorem to $f(x_1, \dots, x_{n-2}, c_{n-1}, c_n)$ and $x_{n-2}$, and so on. Finally we obtain

$$f(x_1, \dots, x_n) = f(c_1, \dots, c_n) + \sum_{i=1}^{n} \partial_i f(x_1, \dots, x_{i-1}, \xi_i, c_{i+1}, \dots, c_n)(x_i - c_i)$$

for some $\boldsymbol{\xi} \in \boldsymbol{X}$ and

$$f(x_1, \ldots, x_n) \in f(c_1, \ldots, c_n) + \sum_{i=1}^{n} \partial_i f(X_1, \ldots, X_i, c_{i+1}, \ldots, c_n)(X_i - c_i). \tag{4.2}$$

Thus, every interval extension $F_i$ of $\partial_i f$, $i = 1, \ldots, n$ and any choice of $\boldsymbol{c} \in \boldsymbol{X}$ give rise to an interval extension $F$ of $f$:

$$F(\boldsymbol{X}) = f(\boldsymbol{c}) + \sum_{i=1}^{n} F_i(X_1, \ldots, X_i, c_{i+1}, \ldots, c_n)(X_i - c_i) \supseteq f(\boldsymbol{X}). \tag{4.3}$$

The successive mean value form is a special case of (4.3), where $F_i$ is the Horner form of $\partial_i f$ and $\boldsymbol{c} = \mathrm{mid}(\boldsymbol{X})$.

**Definition 4.9 (Successive Mean Value Form)** *The successive mean value form* $M_f^{\leadsto} : \mathbb{IR}^n \to \mathbb{IR}$ *is defined as*

$$M_f^{\leadsto}(\boldsymbol{X}) = f(\boldsymbol{c}) + \sum_{i=1}^{n} H_{\partial_i f}(X_1, \ldots, X_i, c_{i+1}, \ldots, c_n)(X_i - c_i),$$

*where* $\boldsymbol{c} = \mathrm{mid}(\boldsymbol{X})$. □

**Theorem 4.10 (Interval Extension)** $M_f^{\leadsto}$ *is an interval extension of* $f$. □

**Proof.** Follows from (4.3). □

**Theorem 4.11 (Accuracy)** *For all* $\boldsymbol{X} \in \mathbb{IR}^n$ *it holds that*

$$M_f^{\leadsto}(X) \subseteq M_f(X). \ \square$$

**Proof.** Trivial. □

**Corollary 4.12 (Convergence)** $M_f^{\leadsto}$ *converges quadratically to* $f$. □

**Theorem 4.13 (Inclusion Monotonicity)** $M_f^{\leadsto}$ *is* not *inclusion monotone.* □

**Proof.** Consider the polynomial $f(x_1, x_2) = x_1^2 x_2$. We obtain

$$M_f^{\leadsto}(X_1, X_2) = \mathrm{mid}(X_1)^2 \mathrm{mid}(X_2) + 2X_1 \mathrm{mid}(X_2)(X_1 - \mathrm{mid}(X_1)) + X_1^2(X_2 - \mathrm{mid}(X_2)).$$

Obviously

$$\begin{pmatrix} [-1,1] \\ [0,2] \end{pmatrix} \supset \begin{pmatrix} [-1,1] \\ [2,2] \end{pmatrix},$$

but

$$M_f^{\leadsto}([-1,1],[0,2]) = [-3,3] \subset [-4,4] = M_f^{\leadsto}([-1,1],[2,2]). \ \square$$

**Algorithm and Complexity.**

Algorithm 4.4 (SMF) for evaluating the successive mean value form follows immediately from Defintion 4.9. As in Algorithm (MHF), we do *not* compute the $i$-th partial derivative of $f$ and evaluate it on $(X_1, \ldots, X_i, c_{i+1}, \ldots, c_n)$. Instead, we first compute successively $i-$variate polynomials

$$f^{(i)}(x_1, \ldots, x_i) = f(x_1, \ldots, x_i, c_{i+1}, \ldots, c_n)$$

and then evaluate the $i$-th partial derivative of $f^{(i)}$ on $(X_1, \ldots, X_i)$. As a side product, we obtain $f(c_1, \ldots, c_n)$ which is needed anyways. As usual, the intermediate polynomials have to be enclosed by interval polynomials because of rounding errors.

**Algorithm 4.4 (SMF)** [Successive Mean Value Form]

*In:* $f(\boldsymbol{x}) = \sum_{i=1}^{m} a_i\, x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}} \in \mathbb{F}[x_1, \dots, x_n]$,
$\boldsymbol{X} \in \mathbb{IF}^n$.

*Out:* $\mathrm{SMF}(f, \boldsymbol{X}) \in \mathbb{IF}$, $\mathrm{SMF}(f, \boldsymbol{X}) \supseteq M_f^{\sim}(\boldsymbol{X})$.

(1) [Midpoint.]
$\boldsymbol{c} \longleftarrow \mathrm{MID}(\boldsymbol{X})$

(2) [Convert $f$ to an Interval Polynomial.]
$F^{(n)}(x_1, \dots, x_n) \longleftarrow f(x_1, \dots, x_n)$

(3) [Evaluate $f$ successively in $\boldsymbol{c}$.]
for $i = n, \dots, 1$ do $F^{(i-1)}(x_1, \dots, x_{i-1}) \longleftarrow \mathrm{HFIL}(F^{(i)}(x_1, \dots, x_i), [c_i])$

(4) [Partial Derivatives.]
for $i = n, \dots, 1$ do $F_i^{(i)}(x_1, \dots, x_i) \longleftarrow \frac{\partial}{\partial x_i} F^{(i)}(x_1, \dots, x_i)$

(5) [Evaluate Partial Derivatives.]
for $i = n, \dots, 1$ do $F_i \longleftarrow \mathrm{HFMI}(F_i^{(i)}(x_1, \dots, x_i), X_1, \dots, X_i)$

(6) [Accumulate.]
$Y \longleftarrow F^{(0)}()$
for $i = n, \dots, 1$ do $Y \longleftarrow Y + F_i * (X_i - c_i)$

(7) [Return.]
return $Y$

**Theorem 4.14 (Complexity)** *Algorithm 4.4 (SMF) costs*

| | |
|---|---|
| $1/2n(n+3)m$ | *interval power computations,* |
| $1/2n(n+1)m + n$ | *interval multiplications,* |
| $4nm + n$ | *number multiplications and* |
| $2nm + 4n + 2m - 2$ | *number additions.* □ |

**Proof.** We assume the worst case where all $F^{(i)}$ have $m$ monomials, i.e. no two degree vectors of $f$ have a common prefix.

- Step 1 costs $n$ number multiplications and $2n$ number additions.
- Step 3 costs $nm$ interval power computations, $2nm$ number multiplications and $2m - 2$ number additions.
- Step 4 costs $2nm$ number multiplications.
- The $i$-th iteration in Step 5 costs $im$ interval power computations, $im$ interval multiplications and $2m - 2$ number additions. Hence, Step 5 costs $1/2n(n+1)m$ interval power computations, $1/2n(n+1)m$ interval multiplications and $2nm - 2n$ number additions.
- Step 6 costs $n$ interval multiplications and $4n$ number additions. □

**Remark.** Algorithm 4.4 (SMF) is faster than Algorithm 4.3 (MHF) by one Horner evaluation.

## 4.2 Successive Slope Form

The univariate mean value form was improved significantly by replacing the derivative by a slope. Now, we apply this idea to the multivariate case. The interval extension which we derive can also be obtained by applying the formulas of [Krawczyk and Neumaier, 1985] for rational expressions in a specific way. However, for polynomials there is a more direct method, which reveals possibilities to optimize a corresponding algorithm.

Generalizing the univariate case means finding polynomials $g_i(\boldsymbol{x}, \boldsymbol{c})$, $i = 1, \ldots, n$ such that

$$f(\boldsymbol{x}) = f(\boldsymbol{c}) + \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{c})(\boldsymbol{x} - \boldsymbol{c}). \tag{4.4}$$

As the $g_i$ are not uniquely determined by 4.4 we have to impose a further condition: We require that $g_i$ depends only on $x_1, \ldots, x_i, c_i, c_{i+1}, \ldots, c_n$. Therefore, we write

$$g_i(x_1, \ldots, x_i, c_i, c_{i+1}, \ldots, c_n)$$

instead of $g_i(\boldsymbol{x}, \boldsymbol{c})$.

The polynomials $g_i$ can be constructed as follows. First, consider $f(x_1, \ldots, x_n)$ as a univarate function in $x_n$. Let $g_n(x_1, \ldots, x_n, c_n)$ be the uniquely defined polynomial such that

$$f(x_1, \ldots, x_n) = f(x_1, \ldots, x_{n-1}, c_n) + g_n(x_1, \ldots, x_n, c_n)(x_n - c_n).$$

Next, consider $f(x_1, \ldots, x_{n-1}, c_n)$ as a univariate function in $x_{n-1}$ and let $g_{n-1}(x_1, \ldots, x_{n-1}, c_{n-1}, c_n)$ be the uniquely defined polynomial such that

$$f(x_1, \ldots, x_{n-1}, c_n) = f(x_1, \ldots, x_{n-2}, c_{n-1}, c_n) + g_{n-1}(x_1, \ldots, x_{n-1}, c_{n-1}, c_n)(x_{n-1} - c_{n-1}).$$

Hence,

$$\begin{aligned} f(x_1, \ldots, x_n) &= f(x_1, \ldots, x_{n-2}, c_{n-1}, c_n) \\ &+ g_{n-1}(x_1, \ldots, x_{n-1}, c_{n-1}, c_n)(x_{n-1} - c_{n-1}) \\ &+ g_n(x_1, \ldots, x_n, c_n)(x_n - c_n). \end{aligned}$$

Applying this procedure successively to $x_{n-2}, \ldots, x_1$ we obtain polynomials $g_i(x_1, \ldots, x_i, c_i, c_{i+1}, \ldots, c_n)$ such that

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} g_i(x_1, \ldots, x_i, c_i, c_{i+1}, \ldots, c_n)(x_i - c_i). \tag{4.5}$$

Thus, every interval extension $G_i$ of $g_i$, $i = 1, \ldots, n$ and any choice of $\boldsymbol{c} \in \mathbb{R}^n$ gives rise to an interval extension $F$ of $f$:

$$F(\boldsymbol{X}) = f(\boldsymbol{c}) + \sum_{i=1}^{n} G_i(X_1, \ldots, X_i, c_i, c_{i+1}, \ldots, c_n)(X_i - c_i) \supseteq f(\boldsymbol{X}). \tag{4.6}$$

The successive slope form is a special case of (4.6) where $G_i$ is the Horner form of $g_i$.

**Definition 4.15 (Successive Slope Form)** *The successive slope form* $M_f^{\rightrightarrows} : \mathbb{IR}^n \to \mathbb{IR}$ *is defined as*

$$M_f^{\rightrightarrows}(\boldsymbol{X}) = f(\boldsymbol{c}) + \sum_{i=1}^{n} H_{g_i}(X_1, \ldots, X_i, c_i, c_{i+1}, \ldots, c_n)(X_i - c_i),$$

*where* $\boldsymbol{c} = \operatorname{mid}(\boldsymbol{X})$ *and* $g_i$ *is as in (4.5).* □

**Theorem 4.16 (Convergence)** $M_f^{\rightrightarrows}$ *converges quadratically to* $f$. □

**Algorithm and Complexity.**

Before we devise an algorithm for $M_f^{\rightrightarrows}$ we give formulas for the polynomials $g_i$.

**Theorem 4.17** *Let*

$$g_i(x_1, \ldots, x_i, c_i, c_{i+1}, \ldots, c_n) = \sum_{j=1}^{m} a_j x_1^{d_{j,1}} \cdots x_{i-1}^{d_{j,i-1}} \, h^{(d_{j,i})}(x_i, c_i) \, c_{i+1}^{d_{j,i+1}} \cdots c_n^{d_{j,n}},$$

*where*
$$h^{(d)}(x_i, c_i) = \sum_{\substack{u+v=d-1 \\ u\geq 0,\, v\geq 0}} x_i^u c_i^v.$$
*Then*
$$f(\boldsymbol{x}) = f(\boldsymbol{c}) + \sum_{i=1}^{n} g_i(x_1, \ldots, x_i, c_i, c_{i+1}, \ldots, c_n)(x_i - c_i). \ \square$$

**Proof.** We have to show that for all $i$
$$f(x_1, \ldots, x_i, c_{i+1}, \ldots, c_n) - f(x_1, \ldots, x_{i-1}, c_i, \ldots, c_n) = g_i(x_1, \ldots, x_i, c_i, c_{i+1}, \ldots, c_n)(x_i - c_i).$$
Note that
$$h^{(d)}(x_i, c_i)(x_i - c_i) = x_i^d - c_i^d.$$
$$\begin{aligned}
& f(x_1, \ldots, x_i, c_{i+1}, \ldots, c_n) - f(x_1, \ldots, x_{i-1}, c_i, \ldots, c_n) \\
= {} & \sum_{j=1}^{m} a_j\, x_1^{d_{j,1}} \cdots x_i^{d_{j,i}} c_{i+1}^{d_{j,i+1}} \cdots c_n^{d_{j,n}} - \sum_{j=1}^{m} a_j\, x_1^{d_{j,1}} \cdots x_{i-1}^{d_{j,i-1}} c_i^{d_{j,i}} \cdots c_n^{d_{j,n}} \\
= {} & \sum_{j=1}^{m} a_j\, x_1^{d_{j,1}} \cdots x_{i-1}^{d_{j,i-1}} c_{i+1}^{d_{j,i+1}} \cdots c_n^{d_{j,n}} (x_i^{d_{j,i}} - c_i^{d_{j,i}}) \\
= {} & \sum_{j=1}^{m} a_j\, x_1^{d_{j,1}} \cdots x_{i-1}^{d_{j,i-1}} c_{i+1}^{d_{j,i+1}} \cdots c_n^{d_{j,n}} h^{(d_{j,i})}(x_i, c_i)(x_i - c_i) \\
= {} & g_i(x_1, \ldots, x_i, c_i, c_{i+1}, \ldots, c_n)(x_i - c_i). \ \square
\end{aligned}$$

The following observations are important for the efficiency of an implementation of $M_f^{\Rightarrow}$.

- First, the polynomials $h^{(d)}$ have the property
$$h^{(d+1)}(x_i, c_i) = h^{(d)}(x_i, c_i)x_i + c_i^d.$$
As the intervals $H_{h^{(d_{j,i})}}(X_i, c_i)$ have to be computed for $j = 1, \ldots, m$, it is advantageous to pre compute intervals
$$\begin{aligned}
H_i^{(0)} &= 0 \\
H_i^{(1)} &= 1 \\
H_i^{(d)} &= H_i^{(d-1)} X_i + c_i^{d-1} \text{ for } \ d = 2, \ldots, \max_j d_{j,i}
\end{aligned}$$
as well as the powers of $c_i$, $i = 0, \ldots, \max_j d_{j,i}$ where $\boldsymbol{X}$ is the input interval and $\boldsymbol{c} = \text{mid}(\boldsymbol{X})$.
- Further, by evaluating $f$ successively in its last variable on $c_i$, $i = n, \ldots, 1$ we obtain polynomials
$$\begin{aligned}
f^{(i)}(x_1, \ldots, x_i) &= f^{(i+1)}(x_1, \ldots, x_i, c_i) \\
&= \sum_{j=1}^{m^{(i)}} a_j^{(i)} x_1^{d_{j,1}^{(i)}} \cdots x_i^{d_{j,i}^{(i)}}
\end{aligned}$$
The coefficients of these polynomials can be used for the evaluation of $g_i$, as
$$g_i(x_1, \ldots, x_i, c_i, c_{i+1}, \ldots, c_n) = \sum_{j=1}^{m^{(i)}} a_j^{(i)} x_1^{d_{j,1}^{(i)}} \cdots x_{i-1}^{d_{j,i-1}^{(i)}} h^{(d_{j,i}^{(i)})}(x_i, c_i).$$

- Finally, let $a_i^*$, $i = 1, \ldots, \deg(f^{(1)})$ such that

  $$f^{(1)}(x_1) = \sum_{i=1}^{\deg(f^{(1)})} a_i^* x_1^i.$$

  Then

  $$g_1(x_1, c_1, c_2, \ldots, c_n) = \sum_{i=1}^{\deg(f^{(1)})} b_i^* x_1^{i-1}$$

  where

  $$b_i^* = \begin{cases} a_i^* & \text{if } i = \deg(f^{(1)}) \\ b_{i+1}^* c_1 + a_i^* & \text{else.} \end{cases}$$

  Thus,

  $$f(c_1, \ldots, c_n) + H_{g_1}(X_1, c_1, c_2, \ldots, c_n)(X_1 - c_1) = M_{\widetilde{f^{(1)}}}(X_1),$$

  i.e. we end up with the univariate slope form.

**Remark.** The polynomials $h^{(d)}(x_i, c_i)$ have an interesting structure and one may expect that the computation of the exact range $h(X_i, \mathrm{mid}(X_i))$ is cheap. In fact, it holds that if $0 \notin \mathrm{int}(X_i)$ then

$$H_i^{(d)} = h^{(d)}(X_i, \mathrm{mid}(X_i)).$$

Further, if $d$ is even, then $h^{(d)}(x_i, c_i)$ is monotone in $x_i$, hence

$$[h^{(d)}(\underline{X}_i, \mathrm{mid}(X_i)), h^{(d)}(\overline{X}_i, \mathrm{mid}(X_i))] = h^{(d)}(X_i, \mathrm{mid}(X_i)).$$

However, we will not make use of this observation.

**Algorithm 4.5 (SSF)** [Successive Slope Form]

*In:* $f(\boldsymbol{x}) = \sum_{i=1}^{m} a_i\, x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}} \in \mathbb{F}[x_1, \ldots, x_n]$,
$\boldsymbol{X} \in \mathbb{IF}^n$.

*Out:* $\mathrm{SSF}(f, \boldsymbol{X}) \in \mathbb{IF}$, $\mathrm{SSF}(f, \boldsymbol{X}) \supseteq M_f^{\Rightarrow}(\boldsymbol{X})$.

(1) [Powers of $c_i$ and $H_i^{(j)}$.]
for $i = 2, \ldots, n$

(1.1) [Degree.]
$d \longleftarrow$ degree of $f$ in $x_i$

(1.2) [Powers of $c_i$.]
$c_i \longleftarrow \mathrm{MID}(X_i)$
$C_i^{(0)} \longleftarrow 1$
for $j = 1, \ldots, d$ do $C_i^{(j)} \longleftarrow C_i^{(j-1)} * [c_i]$

(1.3) [Compute $H_i^{(d)}$.]
$H_i^{(0)} \longleftarrow 0$
$H_i^{(1)} \longleftarrow 1$
for $j = 2, \ldots, d$ do $H_i^{(j)} \longleftarrow H_i^{(j-1)} X_i + C_i^{(j-1)}$

(2) [Sucessively Evaluate Slopes.]
$F^{(n)}(x_1, \ldots, x_n) \longleftarrow f(x_1, \ldots, x_n)$
for $i = n, \ldots, 2$

(2.1) [Slope.]
Let $F^{(i)}(x_1, \ldots, x_i) = \sum_{j=1}^{m^{(i)}} a_j^{(i)} x_1^{d_{j,1}^{(i)}} \cdots x_i^{d_{j,i}^{(i)}}$
$\tilde{F}^{(i)}(x_1, \ldots, x_{i-1}) \longleftarrow \sum_{j=1}^{m^{(i)}} a_j^{(i)} x_1^{d_{j,1}^{(i)}} \cdots x_{i-1}^{d_{j,i-1}^{(i)}} H_i^{d_{j,i}^{(i)}}$
$F_i \longleftarrow \mathrm{HFMI}(\tilde{F}^{(i)}(x_1, \ldots, x_{i-1}), X_1, \ldots, X_{i-1})$

(2.2) [Successively Evaluate $f$.]
$F^{(i-1)} \longleftarrow \mathrm{HFIL}(F^{(i)}(x_1, \ldots, x_i), [c_i])$

(3) [Univarate Slope Form.]
$Y \longleftarrow \mathrm{DSF}(F^{(1)}(x_1), X_1)$

(4) [Add Slopes.]
for $i = 2, \ldots, n$ do $Y \longleftarrow Y + F_i * (X_i - c_i)$

(5) [Return.]
return $Y$

**Remark.** In Step 3 we call Algorithm (DSF) for evaluating the dense univariate slope form. As the first argument is an interval polynomial, we have to modify Algorithm (DSF). However, the modification is trivial and the complexity remains the same. In order to conform precisely to Definition 4.15, we would have to evaluate the slope form instead of the dense slope form.

**Theorem 4.18 (Complexity)** *Let $d$ be the maximum degree of $f$ in each variable. Then Algorithm 4.4* (SSF) *costs*

$$\begin{array}{rl} 1/2n^2m + 1/2nm - m & \textit{interval power computations} \\ 1/2n^2 + nm + 2nd - 1/2n - m + d - 1 & \textit{interval multiplications} \\ 2d + 1 & \textit{number multiplications} \\ 2n^2 + 2nmd - 2nd + 4n + 2d + 2m - 4 & \textit{number additions.} \end{array}$$

**Proof.**

- Let $d$ be the maximum degree of $f$ in each variable. First, we compute the cost of the $i$-th iteration of Step 1. Step 1.2 costs $d$ interval multiplications, 1 number multiplication and 2 number additions. Step 1.3 costs $d-1$ interval multiplications and $2d-2$ number additions. Thus, Step 1 costs $2nd-n$ interval multiplications, $n$ number multiplications and $2nd$ number additions.
- The $i$-th iteration of Step 2.1 costs $m^{(i)}$ interval multiplications $2(m^{(i)} - m^{(i-1)})$ number additions for the computation of the coefficients of $\tilde{F}^{(i)}$ and $(i-1)m^{(i-1)}$ interval power computations, $(i-1)m^{(i-1)}$ interval multiplications and $2m^{(i-1)} - 2$ number additions for its evaluation. In the worst case $m^{(i)} = m$ for $i > 0$ and we obtain $1/2n(n-1)m$ interval power computations, $(n-1)m$ interval multiplications and $2(n-1)(m-1)$ number additions for all iterations of Step 2.1. The total cost of Step 2.2 is $(n-1)m$ interval power computations, $2(n-1)m$ number multiplications and $2m-2$ number additions. So, Step 2 costs $1/2(n-1)(n+2)m$ interval power computations $(n-1)m$ interval multiplications and $2n(m-1)$ number additions.
- Step 3 costs $d$ interval multiplications, $2d+1$ number multiplications and $2d+2m$ number additions.
- Step 4 costs $1/2n(n+1) - 1$ interval multiplications and $2n(n+1) - 4$ number additions. □

## 4.3 Bicentered Mean Value Form

The multivariate bicentered mean value form is a straight forward generalization of the univariate case. Both have been introduced by [Baumann, 1988].

The basic idea is to evaluate the mean value form twice with different centers $\boldsymbol{c}^{\uparrow}$ and $\boldsymbol{c}^{\downarrow}$ and intersect the results. The centers are optimal in the sense that $\boldsymbol{c}^{\uparrow}$ minimizes the upper bound and $\boldsymbol{c}^{\downarrow}$ maximizes the lower bound. The bicentered mean value form is therefore particularly suitable for the separate computation of an upper or lower bound of $f$ in $\boldsymbol{X}$.

In the sequel let

$$M_f(\boldsymbol{X}, \boldsymbol{c}) = f(\boldsymbol{c}) + \sum_{i=1}^{n} H_{\partial_i f}(\boldsymbol{X})(X_i - c_i).$$

**Definition 4.19 (Multivariate Bicentered Mean Value Form)** *The multivariate bicentered mean value form* $\ddot{M}_f : \mathbb{IR}^n \to \mathbb{IR}$ *is defined as*

$$\ddot{M}_f(\boldsymbol{X}) = [\underline{M_f(\boldsymbol{X}, \boldsymbol{c}^{\downarrow})}, \overline{M_f(\boldsymbol{X}, \boldsymbol{c}^{\uparrow})}],$$

*where the optimal centers* $\boldsymbol{c}^{\uparrow}$ *and* $\boldsymbol{c}^{\downarrow}$ *are*

$$\begin{aligned} c_i^{\uparrow} &= \begin{cases} \overline{X}_i & \text{if } \underline{F_i} \geq 0 \\ \underline{X_i} & \text{if } \overline{F}_i \leq 0 \\ (\overline{F}_i \overline{X}_i - \underline{F_i}\, \underline{X_i})/\mathrm{w}(F_i) & \text{else} \end{cases} \\ c_i^{\downarrow} &= \begin{cases} \underline{X_i} & \text{if } \underline{F_i} \geq 0 \\ \overline{X}_i & \text{if } \overline{F}_i \leq 0 \\ (\overline{F}_i \underline{X_i} - \underline{F_i}\, \overline{X}_i)/\mathrm{w}(F_i) & \text{else} \end{cases} \end{aligned}$$

*and* $F_i = H_{\partial_i f}(\boldsymbol{X})$. □

In the sequel let $\boldsymbol{c}^{\uparrow}, \boldsymbol{c}^{\downarrow}$ and $F_i$ as in Definition 4.19.

The following theorems are generalizations of the corresponding theorems for the univariate case. As the proofs are completely analogous, we omit them.

**Theorem 4.20 (Interval Extension)** $\ddot{M}_f$ *is an interval extension of* $f$. □

**Theorem 4.21 (Non–Overestimation)** *If* $0 \notin \mathrm{int}(F_i)$ *for all* $i$ *then*

$$\ddot{M}_f(\boldsymbol{X}) = f(\boldsymbol{X}). \ \square$$

**Theorem 4.22 (Optimality of Centers)** *For all* $\boldsymbol{c} \in \boldsymbol{X}$ *it holds that*

$$\begin{aligned} \overline{M_f(\boldsymbol{X}, \boldsymbol{c}^{\uparrow})} &\leq \overline{M_f(\boldsymbol{X}, \boldsymbol{c})} \\ \underline{M_f(\boldsymbol{X}, \boldsymbol{c}^{\downarrow})} &\geq \underline{M_f(\boldsymbol{X}, \boldsymbol{c})}. \ \square \end{aligned}$$

**Corollary 4.23**

$$\ddot{M}_f(\boldsymbol{X}) = \bigcap_{\boldsymbol{c} \in \boldsymbol{X}} M_f(\boldsymbol{X}, \boldsymbol{c}). \ \square$$

**Corollary 4.24 (Convergence)** $\ddot{M}_f$ *is quadratically convergent.* □

**Theorem 4.25 (Inclusion Monotonicity)** $\ddot{M}_f$ *is inclusion monotone.* □

**Theorem 4.26** *Let* $C_i = [c_i^{\downarrow}, c_i^{\uparrow}]$ *and let* $\boldsymbol{c} \in \boldsymbol{X}$.

$$\mathrm{w}(M_f(\boldsymbol{X}, \boldsymbol{c})) = \sum_{i=1}^{n} \mathrm{mag}(F_i)\mathrm{w}(X_i) + \mathrm{w}(F_i)\mathrm{mig}(C_i - c_i). \ \square$$

**Corollary 4.27 (Optimality of the Midpoint for the Mean Value Form)** *For all* $\boldsymbol{c} \in \boldsymbol{X}$ *it holds that*

$$\mathrm{w}(\ddot{M}_f(\boldsymbol{X})) \leq \mathrm{w}(M_f(\boldsymbol{X}, \boldsymbol{c})). \ \square$$

**Algorithm and Complexity.**

Algorithm 4.6 (BMFM) for evaluating the bicentered mean value form follows immediately from Defintion 4.19. As in the univariate case, we use Algorithm (OC) for approximating the optimal centers $\boldsymbol{c}^{\uparrow}$ and $\boldsymbol{c}^{\downarrow}$.

**Algorithm 4.6** (BMFM) [Multivariate Bicentered Mean Value Form]

*In:* $f(\boldsymbol{x}) = \sum_{i=1}^{m} a_i \, x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}} \in \mathbb{F}[x_1, \ldots, x_n]$,
$\boldsymbol{X} \in \mathbb{IF}^n$.

*Out:* $\mathrm{BMFM}(f, \boldsymbol{X}) \in \mathbb{IF}$, $\mathrm{BMFM}(f, \boldsymbol{X}) \supseteq \ddot{M}_f(\boldsymbol{X})$.

(1) [Initialize.]
$y^{\uparrow} \longleftarrow 0$
$y^{\downarrow} \longleftarrow 0$

(2) [Iterate over all Variables.]
for $i = 1, \ldots, n$ do

(2.1) [Partial Derivative.]
$F_i(\boldsymbol{x}) \longleftarrow \frac{\partial f(\boldsymbol{x})}{\partial x_i}$ (interval arithmetic for coefficients)

(2.2) [Evaluate Partial Derivative.]
$F_i \longleftarrow \mathrm{HFMI}(F_i(\boldsymbol{x}), \boldsymbol{X})$

(2.3) [Optimal Centers.]
$c_i^{\uparrow}, c_i^{\downarrow} \longleftarrow \mathrm{OC}(F_i, X_i)$

(2.4) [Accumulate.]
$y^{\uparrow} \longleftarrow y^{\uparrow} \mathbin{\hat{+}} \overline{F_i * (X_i - c_i^{\uparrow})}$
$y^{\downarrow} \longleftarrow y^{\downarrow} \mathbin{\check{+}} \underline{F_i * (X_i - c_i^{\downarrow})}$

(3) [Evaluate at Centers.]
$y^{\uparrow} \longleftarrow y^{\uparrow} \mathbin{\hat{+}} \overline{\mathrm{HFM}(f(\boldsymbol{x}), [\boldsymbol{c}^{\uparrow}])}$
$y^{\downarrow} \longleftarrow y^{\downarrow} \mathbin{\check{+}} \underline{\mathrm{HFM}(f(\boldsymbol{x}), [\boldsymbol{c}^{\downarrow}])}$
return $[y^{\downarrow}, y^{\uparrow}]$

**Theorem 4.28 (Complexity)** *Algorithm 4.6 (*BMFM*) costs*

$$\begin{array}{rl} n^2m + 2nm & \textit{interval power computations,} \\ n^2m + 2n & \textit{interval multiplications,} \\ 2n & \textit{number divisions,} \\ 6nm + 4n & \textit{number multiplications and} \\ 2nm + 7n + 4m - 2 & \textit{number additions.} \ \square \end{array}$$

**Proof.** The costs for one iteration of Step 2 are as follows:

- Step 2.1 costs $2m$ number multiplications.
- Step 2.2 costs $nm$ interval power computations, $nm$ interval multiplications and $2m - 2$ number additions.
- Step 2.3 costs 2 number divisions, 4 number multiplications and 3 number additions (Theorem 4.38).
- Step 2.4 costs 2 interval multiplications and 6 number additions.

Summarizing, Step 2 costs $n^2m$ interval power computations, $n^2m + 2n$ interval multiplications, $2n$ number divisions, $2nm + 4n$ number multiplications and $2nm + 7n$ number additions. Further, Step 3 costs $2nm$ interval power computations, $4nm$ number multiplications and $4m - 2$ number additions. $\square$

As in the case of the mean value form, we can evaluate the partial derivatives successively and intersect with $H_f(\boldsymbol{X})$ which is obtained as a side product.

**Definition 4.29 (Bicentered Mean Value – Horner Form)** *The bicentered mean valiue – Horner form* $\ddot{M}_f^{(\mathrm{H})} : \mathbb{IR}^n \to \mathbb{IR}$ *is defined as*

$$\ddot{M}_f^{(\mathrm{H})}(\boldsymbol{X}) = \ddot{M}_f(\boldsymbol{X}) \cap H_f(\boldsymbol{X}). \ \square$$

Algorithm 4.7 (BMHF) evaluates $\ddot{M}_f^{(\mathrm{H})}$.

**Algorithm 4.7** (BMHF) [Bicentered Mean Value – Horner Form]

*In:* $f(\boldsymbol{x}) = \sum_{i=1}^{m} a_i\, x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}} \in \mathbb{F}[x_1, \ldots, x_n]$,
$\boldsymbol{X} \in \mathbb{IF}^n$.

*Out:* $\mathrm{BMHF}(f, \boldsymbol{X}) \in \mathbb{IF}$, $\mathrm{BMHF}(f, \boldsymbol{X}) \supseteq \ddot{M}_f(\boldsymbol{X})$.

(1) [Initialize.]
$y^{\uparrow} \longleftarrow 0$
$y^{\downarrow} \longleftarrow 0$
$F^{(n)}(x_1, \ldots, x_n) \longleftarrow f(x_1, \ldots, x_n)$

(2) [Iterate over all Variables.]
for $i = n, \ldots, 1$ do

(2.1) [Partial Derivative.]
$F_i^{(i)}(x_1, \ldots, x_i) \longleftarrow \frac{\partial}{\partial x_i} F^{(i)}(x_1, \ldots, x_i)$

(2.2) [Evaluate Partial Derivative.]
$F_i \longleftarrow \mathrm{HFMI}(F_i^{(i)}(x_1, \ldots, x_i), X_1, \ldots, X_i)$

(2.3) [Optimal Centers.]
$c_i^{\uparrow}, c_i^{\downarrow} \longleftarrow \mathrm{OC}(F_i, X_i)$

(2.4) [Accumulate.]
$y^{\uparrow} \longleftarrow y^{\uparrow} \mathbin{\hat{+}} \overline{F_i * (X_i - c_i^{\uparrow})}$
$y^{\downarrow} \longleftarrow y^{\downarrow} \mathbin{\check{+}} \underline{F_i * (X_i - c_i^{\downarrow})}$

(2.5) [Evaluate $f$ Successively.]
$F^{(i-1)}(x_1, \ldots, x_{i-1}) \longleftarrow \mathrm{HFIL}(F^{(i)}(x_1, \ldots, x_i), X_i)$

(3) [Evaluate at Centers.]
$y^{\uparrow} \longleftarrow y^{\uparrow} \mathbin{\hat{+}} \overline{\mathrm{HFM}(f(\boldsymbol{x}), [\boldsymbol{c}^{\uparrow}])}$
$y^{\downarrow} \longleftarrow y^{\downarrow} \mathbin{\check{+}} \underline{\mathrm{HFM}(f(\boldsymbol{x}), [\boldsymbol{c}^{\downarrow}])}$
$Y \longleftarrow [y^{\downarrow}, y^{\uparrow}]$

(4) [Intersect with $H_f(\boldsymbol{X})$.]
$Y \longleftarrow Y \cap F^{(0)}()$

(5) [Return.]
return $Y$

**Theorem 4.30 (Complexity)** *Algorithm 4.7 (*BMHF*) costs*

| | |
|---|---|
| $1/2n^2m + 9/2nm$ | *interval power computations,* |
| $1/2n^2m + 5/2nm + 2n$ | *interval multiplications,* |
| $2n$ | *number divisions,* |
| $6nm + 4n$ | *number multiplications and* |
| $2nm + 7n + 8m - 6$ | *number additions.* $\square$ |

**Proof.** As in the proof of Theorem 4.8 we assume the worst case where all $F^{(i)}$ have $m$ monomials. Hence, the successive evaluation of $f$ in Step 2.5 costs $2nm$ interval power computations, $2nm$ interval multiplications and $4m-4$ number additions.

The remaining costs for the $i$-th iteration of Step 2 are:

- Step 2.1 costs $2m$ number multiplications.
- Step 2.2 costs $im$ interval power computations, $im$ interval multiplications and $2m-2$ number additions.
- Step 2.3 costs 2 number divisions, 4 number multiplications and 3 number additions (Theorem 4.38).
- Step 2.4 costs 2 interval multiplications and 6 number additions.

Summarizing, Step 2 costs $1/2n(n+1)m + 2nm$ interval power computations, $1/2n(n+1)m + 2n + 2nm$ interval multiplications, $2n$ number divisions, $2nm + 4n$ number multiplications and $2nm + 7n + 4m - 4$ number additions.

Further, Step 3 costs $2nm$ interval power computations, $4nm$ number multiplications and $4m-2$ number additions. $\square$

## 4.4 Successive Bicentered Mean Value Form

In this section we present a new interval extension called successive bicentered mean value form, which combines the ideas of the successive and the bicentered mean value form: The successive mean value form is evaluated twice with different centers and the results are intersected. The centers are chosen in a similiar way as for the bicentered mean value form. However, the centers are not optimal and it can even happen that the successive mean value form gives more accurate results than the successive bicentered mean value form. On the other hand, the successive bicentered mean value form is always at least as accurate as the bicentered mean value form.

In the sequel let

$$M_f^{\leadsto}(\boldsymbol{X}, \boldsymbol{c}) = f(\boldsymbol{c}) + \sum_{i=1}^{n} H_{\partial_i f}(X_1, \ldots, X_i, c_{i+1}, \ldots, c_n).$$

**Definition 4.31 (Successive Bicentered Mean Value Form)** *The successive bicentered mean value form $\ddot{M}_f^{\leadsto} : \mathbb{IR}^n \to \mathbb{IR}$ is defined as*

$$\ddot{M}_f^{\leadsto}(\boldsymbol{X}) = [\underline{M_f^{\leadsto}(\boldsymbol{X}, \boldsymbol{c}^{\downarrow})}, \overline{M_f^{\leadsto}(\boldsymbol{X}, \boldsymbol{c}^{\uparrow})}], \tag{4.7}$$

*where the centers $\boldsymbol{c}^{\uparrow}$ and $\boldsymbol{c}^{\downarrow}$ are defined recursively as*

$$\begin{aligned} c_i^{\uparrow} &= \begin{cases} \overline{X}_i & \text{if } \underline{F}_i^{\uparrow} \geq 0 \\ \underline{X}_i & \text{if } \overline{F}_i^{\uparrow} \leq 0 \\ (\overline{F}_i^{\uparrow}\, \overline{X}_i - \underline{F}_i^{\uparrow}\, \underline{X}_i)/\mathrm{w}(F_i^{\uparrow}) & \text{else} \end{cases} \\ c_i^{\downarrow} &= \begin{cases} \underline{X}_i & \text{if } \underline{F}_i^{\downarrow} \geq 0 \\ \overline{X}_i & \text{if } \overline{F}_i^{\downarrow} \leq 0 \\ (\overline{F}_i^{\downarrow}\, \underline{X}_i - \underline{F}_i^{\downarrow}\, \overline{X}_i)/\mathrm{w}(F_i^{\downarrow}) & \text{else} \end{cases} \end{aligned}$$

*and*

$$\begin{aligned} F_i^{\uparrow} &= H_{\partial_i f}(X_1, \ldots, X_i, c_{i+1}^{\uparrow}, \ldots, c_n^{\uparrow}) \\ F_i^{\downarrow} &= H_{\partial_i f}(X_1, \ldots, X_i, c_{i+1}^{\downarrow}, \ldots, c_n^{\downarrow}). \ \square \end{aligned}$$

In the sequel let $\boldsymbol{c}^{\uparrow}, \boldsymbol{c}^{\downarrow}$ and $\boldsymbol{F}^{\uparrow}, \boldsymbol{F}^{\downarrow}$ as in Definition 4.31.

**Theorem 4.32 (Interval Extension)** *$\ddot{M}_f^{\leadsto}$ is an interval extension of $f$.* $\square$

**Proof.** As $c_i^{\uparrow}, c_i^{\downarrow}$ are convex linear combinations of $\overline{X}_i$ and $\underline{X}_i$, it holds that $\boldsymbol{c}^{\uparrow}, \boldsymbol{c}^{\downarrow} \in \boldsymbol{X}$. From (4.7) it follows that

$$M_f^{\leadsto}(\boldsymbol{X}, \boldsymbol{c}^{\downarrow}) \cap M_f^{\leadsto}(\boldsymbol{X}, \boldsymbol{c}^{\uparrow}) \subseteq \ddot{M}_f^{\leadsto}(X).$$

According to (4.3)

$$f(\boldsymbol{X}) \subseteq M_f^{\leadsto}(\boldsymbol{X}, \boldsymbol{c})$$

for all $\boldsymbol{c} \in \boldsymbol{X}$. Hence,

$$f(X) \subseteq M_f^{\leadsto}(\boldsymbol{X}, \boldsymbol{c}^{\downarrow}) \cap M_f^{\leadsto}(\boldsymbol{X}, \boldsymbol{c}^{\uparrow}) \subseteq \ddot{M}_f^{\leadsto}(\boldsymbol{X}). \ \square$$

**Theorem 4.33 (Non–Overestimation)**

- *If $0 \notin \mathrm{int}(F_i^{\uparrow})$ for all $i$ then*
$$\overline{\ddot{M}_f^{\leadsto}(\boldsymbol{X})} = \overline{f(\boldsymbol{X})}. \tag{4.8}$$
- *If $0 \notin \mathrm{int}(F_i^{\downarrow})$ for all $i$ then*
$$\underline{\ddot{M}_f^{\leadsto}(\boldsymbol{X})} = \underline{f(\boldsymbol{X})}. \ \square \tag{4.9}$$

**Proof.** We give a proof of (4.8), the proof of (4.9) is analogous. According to Definition 4.31

$$\begin{aligned}
\overline{\ddot{M}_f^{\leadsto}(\boldsymbol{X})} &= \overline{M_f^{\leadsto}(\boldsymbol{X}, \boldsymbol{c}^{\uparrow})} \\
&= f(\boldsymbol{c}^{\uparrow}) + \sum_{i=1}^{n} \overline{H_{\partial_i f}(X_1, \ldots, X_i, c_{i+1}^{\uparrow}, \ldots, c_n^{\uparrow})(X_i - c_i^{\uparrow})} \\
&= f(\boldsymbol{c}^{\uparrow}) + \sum_{i=1}^{n} \overline{F_i^{\uparrow} * (X_i - c_i^{\uparrow})}.
\end{aligned}$$

Assume $0 \notin \mathrm{int}(F_i^{\uparrow})$ for all $i$. If $\underline{F}_i^{\uparrow} \geq 0$ then $c_i^{\uparrow} = \overline{X}_i$. If $\overline{F}_i^{\uparrow} \leq 0$ then $c_i^{\uparrow} = \underline{X}_i$. In both cases

$$\overline{F_i^{\uparrow} * (X_i - c_i^{\uparrow})} = 0,$$

hence, as $\boldsymbol{c}^{\uparrow} \in \boldsymbol{X}$,

$$\overline{\ddot{M}_f^{\leadsto}(\boldsymbol{X})} = f(\boldsymbol{c}^{\uparrow}) \in f(\boldsymbol{X}).$$

From $f(\boldsymbol{X}) \subseteq \ddot{M}_f^{\leadsto}(\boldsymbol{X})$, it follows that

$$\overline{\ddot{M}_f^{\leadsto}(\boldsymbol{X})} = f(\boldsymbol{X}). \ \square$$

The centers $\boldsymbol{c}^{\uparrow}, \boldsymbol{c}^{\downarrow}$ are not optimal in the sense that $\boldsymbol{c}^{\uparrow}$ minimizes the upper bound of the successive mean value form and $\boldsymbol{c}^{\downarrow}$ maximizes the lower bound. In particluar there are cases where $M_f^{\leadsto}$ is more accurate than $\ddot{M}_f^{\leadsto}$, i.e. $\boldsymbol{c} = \mathrm{mid}(\boldsymbol{X})$ leads to a smaller upper bound than $\boldsymbol{c}^{\uparrow}$ and to a larger lower bound than $\boldsymbol{c}^{\downarrow}$.

**Theorem 4.34 (Comparison with Successive Mean Value Form)** *There exist $f \in \mathbb{R}[x_1, \ldots, x_n]$ and $\boldsymbol{X} \in \mathbb{IR}^n$ such that*

$$M_f^{\leadsto}(\boldsymbol{X}) \subset \ddot{M}_f^{\leadsto}(\boldsymbol{X}). \ \square$$

**Proof.** Let $f(x_1, x_2) = x_1^2 x_2$ and let $\boldsymbol{X} = ([-1, 1], [-1, 1])$. Then

$$M_f^{\leadsto}(\boldsymbol{X}) = [-1, 1] \subset [-2, 2] = \ddot{M}_f^{\leadsto}(\boldsymbol{X}). \ \square$$

On the other hand, $\ddot{M}_f^{\leadsto}$ is always at least as accurate as $\ddot{M}_f$.

**Theorem 4.35 (Comparison with Bicentered Mean Value Form)** *For all $f \in \mathbb{R}[x_1, \ldots, x_n]$ and for all $\boldsymbol{X} \in \mathbb{IR}^n$ it holds that*

$$\ddot{M}_f^{\sim}(\boldsymbol{X}) \subseteq \ddot{M}_f(\boldsymbol{X}). \ \square$$

**Proof.** Let $f \in \mathbb{R}[x_1, \ldots, x_n]$ and $\boldsymbol{X} \in \mathbb{IR}^n$ arbitrary but fixed. We have to show that

$$\begin{aligned} \overline{\ddot{M}_f^{\sim}(\boldsymbol{X})} &\leq \overline{\ddot{M}_f(\boldsymbol{X})} && (4.10)\\ \underline{\ddot{M}_f^{\sim}(\boldsymbol{X})} &\geq \underline{\ddot{M}_f(\boldsymbol{X})}. && (4.11)\end{aligned}$$

We give a proof of (4.10), the proof of (4.11) is analogous. In order to avoid confusion of notation, we replace $c_i^{\uparrow}$ by $d_i^{\uparrow}$ and $F_i^{\uparrow}$ by $G_i^{\uparrow}$ for the successive bicentered mean value form and keep the symbols $c_i^{\uparrow}$, $F_i$ for the bicentered mean value form. Thus,

$$d_i^{\uparrow} = \begin{cases} \overline{X}_i & \text{if } \underline{G}_i^{\uparrow} \geq 0 \\ \underline{X}_i & \text{if } \overline{G}_i^{\uparrow} \leq 0 \\ (\overline{G}_i^{\uparrow}\, \overline{X}_i - \underline{G}_i^{\uparrow}\, \underline{X}_i)/\mathrm{w}(G_i^{\uparrow}) & \text{else} \end{cases}$$

where

$$G_i^{\uparrow} = H_{\partial_i f}(X_1, \ldots, X_i, d_{i+1}^{\uparrow}, \ldots, d_n^{\uparrow})$$

and

$$c_i^{\uparrow} = \begin{cases} \overline{X}_i & \text{if } \underline{F}_i \geq 0 \\ \underline{X}_i & \text{if } \overline{F}_i \leq 0 \\ (\overline{F}_i\, \overline{X}_i - \underline{F}_i\, \underline{X}_i)/\mathrm{w}(F_i) & \text{else} \end{cases}$$

where

$$F_i = H_{\partial_i f}(X_1, \ldots, X_n).$$

We have to show that

$$f(\boldsymbol{d}^{\uparrow}) + \sum_{i=1}^{n} \overline{G_i^{\uparrow} * (X_i - d_i^{\uparrow})} \leq f(\boldsymbol{c}^{\uparrow}) + \sum_{i=1}^{n} \overline{F_i * (X_i - c_i^{\uparrow})}.$$

From the mean value Theorem it follows that there exists $\boldsymbol{\xi} \in \boldsymbol{X}$ such that

$$f(\boldsymbol{d}^{\uparrow}) = f(\boldsymbol{c}^{\uparrow}) + \sum_{i=1}^{n} \partial_i f(\boldsymbol{\xi})(d_i^{\uparrow} - c_i^{\uparrow}).$$

Thus,

$$\begin{aligned} f(\boldsymbol{d}^{\uparrow}) + \sum_{i=1}^{n} \overline{G_i^{\uparrow} * (X_i - d_i^{\uparrow})} &= f(\boldsymbol{c}^{\uparrow}) + \sum_{i=1}^{n} \partial_i f(\boldsymbol{\xi})(d_i^{\uparrow} - c_i^{\uparrow}) + \overline{G_i^{\uparrow} * (X_i - d_i^{\uparrow})} \\ &\leq f(\boldsymbol{c}^{\uparrow}) + \sum_{i=1}^{n} \overline{F_i * (d_i^{\uparrow} - c_i^{\uparrow})} + \overline{G_i^{\uparrow} * (X_i - d_i^{\uparrow})}. \end{aligned}$$

In order to complete the proof, we show for each $i$ that

$$\overline{F_i * (d_i^{\uparrow} - c_i^{\uparrow})} + \overline{G_i^{\uparrow} * (X_i - d_i^{\uparrow})} \leq \overline{F_i * (X_i - c_i^{\uparrow})}.$$

Hence, let $i$ arbitrary but fixed. We distinguish the cases whether or not $0 \in \mathrm{int}(G_i^{\uparrow})$.

- Assume $\underline{G}_i^{\uparrow} \geq 0$ or $\overline{G}_i^{\uparrow} \leq 0$. From the definition of $d_i^{\uparrow}$ it follows that

  $$\overline{G_i^{\uparrow} * (X_i - d_i^{\uparrow})} = 0,$$

  hence

  $$\begin{aligned} \overline{F_i * (d_i^{\uparrow} - c_i^{\uparrow})} + \overline{G_i^{\uparrow} * (X_i - d_i^{\uparrow})} &= \overline{F_i * (d_i^{\uparrow} - c_i^{\uparrow})} \\ &\leq \overline{F_i * (X_i - c_i^{\uparrow})}. \end{aligned}$$

- Assume $0 \in \mathrm{int}(G_i^\uparrow)$. As $G_i^\uparrow \subseteq F_i$, it holds that $0 \in \mathrm{int}(F_i)$ and from the definition of $d_i^\uparrow$ and $c_i^\uparrow$ it follows that

$$\begin{array}{rclcl} \overline{G_i^\uparrow * (X_i - d_i^\uparrow)} & = & \overline{G_i^\uparrow} * (\overline{X}_i - d_i^\uparrow) & = & \underline{G_i^\uparrow} * (\underline{X}_i - d_i^\uparrow) \\ \overline{F_i * (X_i - c_i^\uparrow)} & = & \overline{F}_i * (\overline{X}_i - c_i^\uparrow) & = & \underline{F}_i * (\underline{X}_i - c_i^\uparrow). \end{array}$$

Next, we distinguish the cases $d_i > c_i$ and $d_i \leq c_i$.

  - Assume $d_i > c_i$.

$$\begin{array}{rcl} \overline{F_i * (d_i^\uparrow - c_i^\uparrow)} + \overline{G_i^\uparrow * (X_i - d_i^\uparrow)} & = & \overline{F}_i * (d_i^\uparrow - c_i^\uparrow) + \overline{G}_i^\uparrow * (\overline{X}_i - d_i^\uparrow) \\ & \leq & \overline{F}_i * (d_i^\uparrow - c_i^\uparrow) + \overline{F}_i * (\overline{X}_i - d_i^\uparrow) \\ & = & \overline{F}_i * (\overline{X}_i - c_i^\uparrow) \\ & = & \overline{F_i * (X_i - c_i^\uparrow)}. \end{array}$$

  - Assume $d_i \leq c_i$.

$$\begin{array}{rcl} \overline{F_i * (d_i^\uparrow - c_i^\uparrow)} + \overline{G_i^\uparrow * (X_i - d_i^\uparrow)} & = & \underline{F}_i * (d_i^\uparrow - c_i^\uparrow) + \underline{G}_i^\uparrow * (\underline{X}_i - d_i^\uparrow) \\ & \leq & \underline{F}_i * (d_i^\uparrow - c_i^\uparrow) + \underline{F}_i * (\underline{X}_i - d_i^\uparrow) \\ & = & \underline{F}_i * (\underline{X}_i - c_i^\uparrow) \\ & = & \overline{F_i * (X_i - c_i^\uparrow)}. \ \square \end{array}$$

**Corollary 4.36 (Convergence)** *$\ddot{M}_f^{\rightsquigarrow}$ converges quadratically to $f$.* $\square$

**Theorem 4.37 (Inclusion Monotonicity)** *$\ddot{M}_f^{\rightsquigarrow}$ is* not *inclusion monotone.* $\square$

**Proof.** Consider the polynomial $f(x_1, x_2) = x_1^2 x_2$. Obviously

$$\begin{pmatrix} [-1,1] \\ [0,2] \end{pmatrix} \supset \begin{pmatrix} [-1,1] \\ [2,2] \end{pmatrix},$$

but

$$\ddot{M}_f^{\rightsquigarrow}([-1,1],[0,2]) = [0,2] \subset [-4,4] = \ddot{M}_f^{\rightsquigarrow}([-1,1],[2,2]). \ \square$$

**Algorithm and Complexity.**

Algorithm 4.9 (SBM) computes the successive bicentered mean value form. First, we have to modify Algorithm (OC), because now the centers $c_i^\uparrow$ and $c_i^\downarrow$ depend on different derivatives $F_i^\uparrow$ and $F_i^\downarrow$.

**Algorithm 4.8** (OC2) [Optimal Centers for Successive Bicentered Mean Value Form]

*In:* $F^\uparrow, F^\downarrow \in \mathbb{IF}$,
$X \in \mathbb{IF}$.

*Out:* $c^\uparrow, c^\downarrow \in X$, approximations of the centers according to Definition 4.31.

(1) [Computation of $c^\uparrow$.]

(1.1) [Monotonicity Test.]
if $\underline{F}^\uparrow \geq 0$ then $c^\uparrow \longleftarrow \overline{X}$, goto Step (2)
if $\overline{F}^\uparrow \leq 0$ then $c^\uparrow \longleftarrow \underline{X}$, goto Step (2)

(1.2) [Approximate.]
clear invalid operation flag
$c^\uparrow \longleftarrow (\overline{F}^\uparrow \mathbin{\tilde{\tilde{*}}} \overline{X} \mathbin{\tilde{\tilde{-}}} \underline{F}^\uparrow \mathbin{\tilde{\tilde{*}}} \underline{X}) \mathbin{\tilde{\tilde{/}}} (\overline{F}^\uparrow \mathbin{\tilde{\tilde{-}}} \underline{F}^\uparrow)$

(1.3) [Check Floating Point Errors.]
if invalid operation flag is raised then $c^{\uparrow} \longleftarrow \overline{X}$
if $c^{\uparrow} < \underline{X}$ then $c^{\uparrow} \longleftarrow \underline{X}$
if $c^{\uparrow} > \overline{X}$ then $c^{\uparrow} \longleftarrow \overline{X}$

(2) [Computation of $c^{\downarrow}$.]

(2.1) [Monotonicity Test.]
if $\underline{F}^{\downarrow} \geq 0$ then $c^{\downarrow} \longleftarrow \underline{X}$, goto Step (3)
if $\overline{F}^{\downarrow} \leq 0$ then $c^{\downarrow} \longleftarrow \overline{X}$, goto Step (3)

(2.2) [Approximate.]
clear invalid operation flag
$c^{\downarrow} \longleftarrow (\overline{F}^{\downarrow} \overset{\approx}{*} \underline{X} \overset{\approx}{-} \underline{F}^{\downarrow} \overset{\approx}{*} \overline{X}) \overset{\approx}{/} (\overline{F}^{\downarrow} \overset{\approx}{-} \underline{F}^{\downarrow})$

(2.3) [Check Floating Point Errors.]
if invalid operation flag is raised then $c^{\downarrow} \longleftarrow \underline{X}$
if $c^{\downarrow} < \underline{X}$ then $c^{\downarrow} \longleftarrow \underline{X}$
if $c^{\downarrow} > \overline{X}$ then $c^{\downarrow} \longleftarrow \overline{X}$

(3) [Return.]
return $c^{\uparrow}, c^{\downarrow}$

**Theorem 4.38 (Complexity)** *Algorithm 4.8* (OC2) *costs*

2 *number divisions,*
4 *number multiplications and*
4 *number additions.* □

In Algorithm 4.9 (SBM) the $n$-th variable is handled separately because $F_n^{\uparrow} = F_n^{\downarrow}$ and therefore the $n$-th partial derivative has to be computed only once.

**Algorithm 4.9** (SBM) [Successive Bicentered Mean Value Form]

*In:* $f(\boldsymbol{x}) = \sum_{i=1}^{m} a_i \, x_1^{d_{i,1}} x_2^{d_{i,2}} \cdots x_n^{d_{i,n}} \in \mathbb{F}[x_1, \ldots, x_n]$,
$\boldsymbol{X} \in \mathbb{IF}^n$.

*Out:* $\mathrm{SBM}(f, \boldsymbol{X}) \in \mathbb{IF}$, $\mathrm{SBM}(f, \boldsymbol{X}) \supseteq \ddot{M}_{\tilde{f}}(\boldsymbol{X})$.

(1) [Last Variable.]

(1.1) [$n$-th Partial Derivative.]
$F_n(x_1, \ldots, x_n) \longleftarrow \frac{\partial}{\partial x_n} f(x_1, \ldots, x_n)$

(1.2) [Evaluate $n$-th Partial Derivative.]
$F_n \longleftarrow \mathrm{HFMI}(F_n(x_1, \ldots, x_n), X_1, \ldots, X_n)$

(1.3) [Centers.]
$c_n^{\uparrow}, c_n^{\downarrow} \longleftarrow \mathrm{OC}(F_n, X_n)$

(1.4) [Accumulate.]
$y^{\uparrow} \longleftarrow \overline{F_n * (X_n - c_n^{\uparrow})}$
$y^{\downarrow} \longleftarrow \underline{F_n * (X_n - c_n^{\downarrow})}$

(1.5) [Evaluate Successively.]
$F^{\uparrow}(x_1, \ldots, x_{n-1}) \longleftarrow \mathrm{HFIL}(f(x_1, \ldots, x_n), [c_n^{\uparrow}])$
$F^{\downarrow}(x_1, \ldots, x_{n-1}) \longleftarrow \mathrm{HFIL}(f(x_1, \ldots, x_n), [c_n^{\downarrow}])$

(2) [Other Variables.]
for $i = n-1, \ldots, 1$ do

(2.1) [$i$-th Partial Derivative.]
$F_i^{\uparrow}(x_1,\ldots,x_i) \longleftarrow \frac{\partial}{\partial x_i} F^{\uparrow}(x_1,\ldots,x_i)$
$F_i^{\downarrow}(x_1,\ldots,x_i) \longleftarrow \frac{\partial}{\partial x_i} F^{\downarrow}(x_1,\ldots,x_i)$

(2.2) [Evaluate $i$-th Partial Derivative.]
$F_i^{\uparrow} \longleftarrow \mathrm{HFMI}(F_i^{\uparrow}(x_1,\ldots,x_i), X_1,\ldots,X_i)$
$F_i^{\downarrow} \longleftarrow \mathrm{HFMI}(F_i^{\downarrow}(x_1,\ldots,x_i), X_1,\ldots,X_i)$

(2.3) [Centers.]
$c_i^{\uparrow}, c_i^{\downarrow} \longleftarrow \mathrm{OC2}(F_i^{\uparrow}, F_i^{\downarrow}, X_i)$

(2.4) [Accumulate.]
$y^{\uparrow} \longleftarrow y^{\uparrow} \mathbin{\hat{+}} \overline{F_i^{\uparrow} * (X_i - c_i^{\uparrow})}$
$y^{\downarrow} \longleftarrow y^{\downarrow} \mathbin{\check{+}} \underline{F_i^{\downarrow} * (X_i - c_i^{\downarrow})}$

(2.5) [Evaluate Successively.]
$F^{\uparrow}(x_1,\ldots,x_{i-1}) \longleftarrow \mathrm{HFIL}(F^{\uparrow}(x_1,\ldots,x_i), [c_i^{\uparrow}])$
$F^{\downarrow}(x_1,\ldots,x_{i-1}) \longleftarrow \mathrm{HFIL}(F^{\uparrow}(x_1,\ldots,x_i), [c_i^{\downarrow}])$

(3) [Add $f(\boldsymbol{c}^{\uparrow}), f(\boldsymbol{c}^{\downarrow})$.]
$y^{\uparrow} \longleftarrow y^{\uparrow} \mathbin{\hat{+}} F^{\uparrow}()$
$y^{\downarrow} \longleftarrow y^{\downarrow} \mathbin{\check{+}} F^{\downarrow}()$
return $[y^{\downarrow}, y^{\uparrow}]$

**Theorem 4.39 (Complexity)** *Algorithm 4.9 (*SBM*) costs*

$$\begin{array}{rl} n^2m + 2nm & \textit{interval power computations,} \\ n^2m + 2n & \textit{interval multiplications,} \\ 2n & \textit{number divisions,} \\ 8nm + 4n - 2m & \textit{number multiplications and} \\ 4nm + 6n + 2m - 3 & \textit{number additions.} \ \square \end{array}$$

**Proof.** As in the proof of Theorem 4.8 we assume the worst case that during the successive evaluation of $f$ in its last variable the number of monomials does not decrease. Hence, the successive evaluation of $f$ in Step 1.5 and Step 2.5 costs $2nm$ interval power computations, $4nm$ number multiplications and $4m - 4$ number additions.

The remaining costs for Step 1 are:

- Step 1.1 costs $2m$ number multiplications.
- Step 1.2 costs $nm$ interval power computations, $nm$ interval multiplications and $2m - 2$ number additions (Theorem 3.6).
- Step 1.3 costs 2 number divisions, 4 number multiplications and 3 number additions (Theorem 4.38).
- Step 1.4 costs 2 interval multiplications and 4 number additions.

Summarizing, Step 1.1 – Step 1.4 cost $nm$ interval power computations, $nm + 2$ interval multiplications, 2 number divisions, $2m + 4$ number multiplications and $2m + 5$ number additions.

In the $i$-th iteration, the costs for Step 2.1 – Step 2.4 are:

- Step 2.1 costs $4m$ number multiplications.
- Step 2.2 costs $2im$ interval power computations, $2im$ interval multiplications and $4m - 4$ number additions (Theorem 3.6).

- Step 2.3 costs 2 number divisions, 4 number multiplications and 4 number additions (Theorem 4.38).
- Step 2.4 costs 2 interval multiplications and 6 number additions.

Summarizing, in the $i$-th iteration Step 2.1 – Step 2.4 cost $2im$ interval power computations, $2im + 2$ interval multiplications, 2 number divisions, $4m+4$ number multiplications and $4m+6$ number additions. Summing up for $i = n-1, \ldots, 1$ we obtain $n^2m - nm$ interval power computations, $n^2m - nm + 2n - 2$ interval multiplications, $2n - 2$ number divisions, $4nm + 4n - 4m - 4$ number multiplications and $4nm + 6n - 4m - 6$ number additions.

Finally, Step 3 costs 2 number additions. □

## 4.5 Experimental Results

The mean value forms which were described in the previous sections are compared experimentally at some classes of random polynomials and input intervals. As the computation of the range of a multivariate polynomial is expensive, we do not report overestimation errors. Instead, we compare the width of the refined mean value forms and the width of the ordinary mean value form. The cost for evaluating each mean value form is given in terms of arithmetic floating point instructions. The coefficients of the test polynomials are uniformly distributed in $[-1, 1]$. The midpoints of the components of the input interval vectors are uniformly distributed in $[-1, 1]$, the widths are uniformly distributed in $[0, 0.1]$. The reason why we chose such small widths is that otherwise the Horner form or the nested form seem preferable, see Section 5. We report the average over 1000 random polynomials and input interval vectors. Table 4.1 shows a comparison for random polynomials with different numbers of variables $n$ and monomials $m$. Each degree vector contains $n$ non–zero components and the degree in each variable is at most 5.

**Accuracy.** The intersection with $H_f$ leads to a significant improvement of $M_f^{(\mathrm{H})}$ and $\ddot{M}_f^{(\mathrm{H})}$ compared to $M_f$ respectively $\ddot{M}_f$, especially if the number of variables is large. In all cases $\ddot{M}_f^{\rightsquigarrow}$ is more accurate than $M_f^{\rightsquigarrow}$, $M_f^{\Rightarrow}$ and $\ddot{M}_f$. The accuracy of $\ddot{M}_f$ is usually better than the accuracy of $M_f^{\rightsquigarrow}$, but with increasing $n$, the difference shrinks. In the last column, $M_f^{\rightsquigarrow}$ is more accurate than $\ddot{M}_f$.

**Cost.** The successive evaluation reduces the cost of $M_f^{(\mathrm{H})}$ and $\ddot{M}_f^{(\mathrm{H})}$ almost by half compared to $M_f$ and $\ddot{M}_f$ respectively if the number of variables is large. Otherwise, the costs are comparable. The evaluation of $M_f^{\rightsquigarrow}$ is only about half as expensive as $M_f$, $\ddot{M}_f$ and $\ddot{M}_f^{\rightsquigarrow}$. These observations conform to the complexity considerations in the previous sections. The evaluation of $M_f^{\Rightarrow}$ is even a bit cheaper than $M_f^{\rightsquigarrow}$.

Table 4.2 shows the same experiment, but this time the polynomials are such that each degree vector has only 2 non–zero components.

**Accuracy.** The accuracy is almost independend of the number of variables and monomials in all cases. Again, $\ddot{M}_f$ is significantly more accurate than $M_f^{\rightsquigarrow}$, even if the number of variables is high. The accuracy of $M_f^{\Rightarrow}$ is between $M_f^{\rightsquigarrow}$ and $\ddot{M}_f$. The accuracies of $\ddot{M}_f^{\rightsquigarrow}$, $\ddot{M}_f$ and $\ddot{M}_f^{(\mathrm{H})}$ are comparable. Intersection with $H_f(\boldsymbol{X})$ leads to an improvement of $M_f^{(\mathrm{H})}$ over $M_f$ but not of $\ddot{M}_f^{(\mathrm{H})}$ over $\ddot{M}_f$.

**Cost.** In contrast to the previous experiment, the costs of $M_f$ and $M_f^{(\mathrm{H})}$ and the costs of $\ddot{M}_f$ and $\ddot{M}_f^{(\mathrm{H})}$ are comparable. As the degree vectors are sparse, the worst case assumptions in the complexity considerations in Theorem 4.8 and Theorem 4.30 are not adequate. Again, $M_f^{\rightsquigarrow}$ and $M_f^{\Rightarrow}$ are about half as expensive as $\ddot{M}_f$ and $\ddot{M}_f^{\rightsquigarrow}$.

*Dense Monomials*

| Variables $n$, Monomials $m$ | $2,3$ | $2,6$ | $4,4$ | $4,8$ | $8,5$ | $8,10$ | $16,6$ | $16,12$ |
|---|---|---|---|---|---|---|---|---|
| Flops for $M_f(\boldsymbol{X})$ | 124.0 | 197.9 | 500.5 | 878.4 | 2215.9 | 4147.4 | 10087.4 | 19530.9 |
| Flops for $M_f^{(\mathrm{H})}(\boldsymbol{X})$ | 134.2 | 205.7 | 431.8 | 724.0 | 1555.4 | 2808.8 | 6103.0 | 11527.8 |
| Flops for $M_f^{\rightsquigarrow}(\boldsymbol{X})$ | 102.6 | 152.9 | 342.8 | 563.4 | 1321.5 | 2366.3 | 5541.2 | 10442.2 |
| Flops for $M_f^{\rightrightarrows}(\boldsymbol{X})$ | 92.9 | 123.0 | 300.4 | 437.3 | 1170.5 | 1941.9 | 5093.0 | 9241.7 |
| Flops for $\ddot{M}_f(\boldsymbol{X})$ | 169.1 | 265.3 | 622.0 | 1075.5 | 2534.9 | 4686.5 | 10895.3 | 20908.6 |
| Flops for $\ddot{M}_f^{(\mathrm{H})}(\boldsymbol{X})$ | 179.3 | 273.1 | 553.4 | 921.1 | 1874.3 | 3347.9 | 6910.9 | 12905.5 |
| Flops for $\ddot{M}_f^{\rightsquigarrow}(\boldsymbol{X})$ | 155.2 | 231.6 | 574.2 | 944.0 | 2382.7 | 4262.7 | 10493.3 | 19748.4 |
| $\mathrm{w}(M_f^{(\mathrm{H})}(\boldsymbol{X}))/\mathrm{w}(M_f(\boldsymbol{X}))$ | 0.810 | 0.876 | 0.617 | 0.697 | 0.348 | 0.393 | 0.160 | 0.176 |
| $\mathrm{w}(M_f^{\rightsquigarrow}(\boldsymbol{X}))/\mathrm{w}(M_f(\boldsymbol{X}))$ | 0.919 | 0.909 | 0.767 | 0.764 | 0.558 | 0.572 | 0.319 | 0.330 |
| $\mathrm{w}(M_f^{\rightrightarrows}(\boldsymbol{X}))/\mathrm{w}(M_f(\boldsymbol{X}))$ | 0.818 | 0.822 | 0.671 | 0.679 | 0.479 | 0.502 | 0.264 | 0.279 |
| $\mathrm{w}(\ddot{M}_f(\boldsymbol{X}))/\mathrm{w}(M_f(\boldsymbol{X}))$ | 0.721 | 0.739 | 0.561 | 0.591 | 0.415 | 0.457 | 0.326 | 0.370 |
| $\mathrm{w}(\ddot{M}_f^{(\mathrm{H})}(\boldsymbol{X}))/\mathrm{w}(M_f(\boldsymbol{X}))$ | 0.714 | 0.730 | 0.536 | 0.569 | 0.333 | 0.371 | 0.160 | 0.176 |
| $\mathrm{w}(\ddot{M}_f^{\rightsquigarrow}(\boldsymbol{X}))/\mathrm{w}(M_f(\boldsymbol{X}))$ | 0.710 | 0.719 | 0.516 | 0.532 | 0.323 | 0.347 | 0.168 | 0.183 |

Table 4.1: Comparison of mean value forms for random polynomials with different numbers of variables and monomials. Each degree vector has $n$ non–zero components and the degree in each variable is at most 5.

*Sparse Monomials*

| Variables $n$, Monomials $m$ | $2,3$ | $2,6$ | $4,4$ | $4,8$ | $8,5$ | $8,10$ | $16,6$ | $16,12$ |
|---|---|---|---|---|---|---|---|---|
| Flops for $M_f(\boldsymbol{X})$ | 124.0 | 197.9 | 239.1 | 416.3 | 437.8 | 788.2 | 841.1 | 1532.5 |
| Flops for $M_f^{(\mathrm{H})}(\boldsymbol{X})$ | 134.2 | 205.7 | 256.6 | 434.3 | 454.3 | 800.7 | 841.7 | 1503.5 |
| Flops for $M_f^{\rightsquigarrow}(\boldsymbol{X})$ | 102.6 | 152.9 | 192.7 | 318.6 | 335.8 | 581.4 | 614.9 | 1082.5 |
| Flops for $M_f^{\rightrightarrows}(\boldsymbol{X})$ | 92.9 | 123.0 | 190.1 | 286.4 | 348.8 | 569.8 | 645.3 | 1101.2 |
| Flops for $\ddot{M}_f(\boldsymbol{X})$ | 169.1 | 265.3 | 321.7 | 553.8 | 584.3 | 1041.6 | 1114.0 | 2006.0 |
| Flops for $\ddot{M}_f^{(\mathrm{H})}(\boldsymbol{X})$ | 179.3 | 273.1 | 339.2 | 571.8 | 600.9 | 1054.1 | 1114.6 | 1976.9 |
| Flops for $\ddot{M}_f^{\rightsquigarrow}(\boldsymbol{X})$ | 155.2 | 231.6 | 325.7 | 547.8 | 595.8 | 1057.1 | 1117.7 | 2020.3 |
| $\mathrm{w}(M_f^{(\mathrm{H})}(\boldsymbol{X}))/\mathrm{w}(M_f(\boldsymbol{X}))$ | 0.810 | 0.876 | 0.829 | 0.884 | 0.825 | 0.879 | 0.818 | 0.860 |
| $\mathrm{w}(M_f^{\rightsquigarrow}(\boldsymbol{X}))/\mathrm{w}(M_f(\boldsymbol{X}))$ | 0.919 | 0.909 | 0.935 | 0.927 | 0.938 | 0.935 | 0.944 | 0.941 |
| $\mathrm{w}(M_f^{\rightrightarrows}(\boldsymbol{X}))/\mathrm{w}(M_f(\boldsymbol{X}))$ | 0.818 | 0.822 | 0.859 | 0.850 | 0.867 | 0.866 | 0.878 | 0.875 |
| $\mathrm{w}(\ddot{M}_f(\boldsymbol{X}))/\mathrm{w}(M_f(\boldsymbol{X}))$ | 0.721 | 0.739 | 0.758 | 0.746 | 0.767 | 0.764 | 0.781 | 0.776 |
| $\mathrm{w}(\ddot{M}_f^{(\mathrm{H})}(\boldsymbol{X}))/\mathrm{w}(M_f(\boldsymbol{X}))$ | 0.714 | 0.730 | 0.755 | 0.745 | 0.766 | 0.763 | 0.780 | 0.776 |
| $\mathrm{w}(\ddot{M}_f^{\rightsquigarrow}(\boldsymbol{X}))/\mathrm{w}(M_f(\boldsymbol{X}))$ | 0.710 | 0.719 | 0.755 | 0.740 | 0.765 | 0.760 | 0.780 | 0.774 |

Table 4.2: Same experiment as before, but this time each degree vector has only 2 non–zero components.

# 5 Experimental Comparison

## 5.1 Efficiency and Accuracy for Random Polynomials

In this section we compare the accuracy of $H_f, N_f, M_f, \ddot{M}_f, M_f^{\rightsquigarrow}$ and $\ddot{M}_f^{\rightsquigarrow}$ As the computation of the exact range of a multivariate polynomial is expensive, we compare the widths of the interval extensions and the width of the Horner form. More precisely, we report the average over 100 random polynomials and inputs of

$$\mathrm{w}(F(\boldsymbol{X}))/\mathrm{w}(H_f(\boldsymbol{X})),$$

where $F$ is one of $N_f, M_f, \ddot{M}_f, M_f^{\rightsquigarrow}$ or $\ddot{M}_f^{\rightsquigarrow}$.

For the experiments, we use 2 classes of random polynomials and 2 classes of input interval vectors.

- In both classes of random polynomials the number of variables is $n = 8$, each polynomial consists of $m = 10$ monomials and the degree in each variable is at most 5.
  - In the first class of random polynomials, all components of the degree vectors are randomly chosen (Figure 5.1,5.2).
  - In the second class, each degree vector has only two non–zero components (Figure 5.3,5.4).
- The input interval vectors $\boldsymbol{X}$ are such that $\mathrm{w}(X_i)$ is randomly chosen from $[0, w]$ for all $i$ and $w$ varies in our experiments.
  - In the first class of random input interval vectors we fix $\mathrm{mid}(X_i) \equiv 0.5$ for all $i$ (Figure 5.1,5.3).
  - In the second class we fix $\mathrm{mid}(X_i) \equiv 0.0$ for all $i$ (Figure 5.2,5.4).

It seems that the results of these and several other experiments can be generalized as follows:

- The accuracy of $N_f$ is usually slightly better than the accuracy of $H_f$.
- If $\mathrm{mid}(X_i) \approx 0$ (Figure 5.2,5.4) then $H_f$, $N_f$, $M_f^{(\mathrm{H})}$ and $\ddot{M}_f^{(\mathrm{H})}$ are best. $M_f^{\Rightarrow}$ is only slightly less accurate. If $\mathrm{mid}(X_i) = 0$ then $H_f$ is equivalent to the quadratically convergent multivariate extension of the Taylor form, which explains the good performance of $H_f$.
- If $\mathrm{mid}(X_i) \approx 0$ (Figure 5.2,5.4) then $M_f^{\Rightarrow}$ is best among all mean value forms. In particular, $M_f^{\Rightarrow}$ and even $M_f^{\rightsquigarrow}$ are better than $\ddot{M}_f^{\rightsquigarrow}$. If all degree vectors are dense (Figure 5.2). and if $\mathrm{mid}(X_i) = 0$ then $c_i = 0$ and $\partial_i f(X_1, \ldots, X_i, c_{i+1}, \ldots, c_n) = 0$ for $i \neq n$. Hence,

  $$M_f^{\rightsquigarrow}(\boldsymbol{X}) = f(0, \ldots, 0) + H_{\partial_n f}(X_1, \ldots, X_n)(X_n - c_n).$$

  This means that the choice $\boldsymbol{c} = 0$ minimizes $\mathrm{w}(M_f^{\rightsquigarrow}(\boldsymbol{X}, \boldsymbol{c}))$.
- If $\mathrm{mid}(X_i) \neq 0$ (Figure 5.1,5.3) then $\ddot{M}_f^{\rightsquigarrow}$ is more accurate than $M_f$, $M_f^{\rightsquigarrow}$, $M_f^{\Rightarrow}$ and $\ddot{M}_f$. If $\mathrm{w}(\boldsymbol{X})$ is small, then $\ddot{M}_f^{\rightsquigarrow}$ is more accurate than $H_f$ and $N_f$ because of its quadratic convergence. If the degree vectors are sparse (Figure 5.3) then $\ddot{M}_f$ is better than $M_f^{\rightsquigarrow}$.

## 5.2 Global Optimization

In this section we compare interval extensions at the problem of finding the global minimum of a multivariate polynomial $f$ in a given interval vector $\boldsymbol{A}$. The problem is solved by the algorithm presented in [Hammer *et al.*, 1993]. The algoritm is modified slightly in dependence of the interval extension which is used to bound $f(\boldsymbol{X})$:

- For $M_f$ and $\ddot{M}_f$ the coefficients of the partial derivatives are computed only once and not in every iteration.

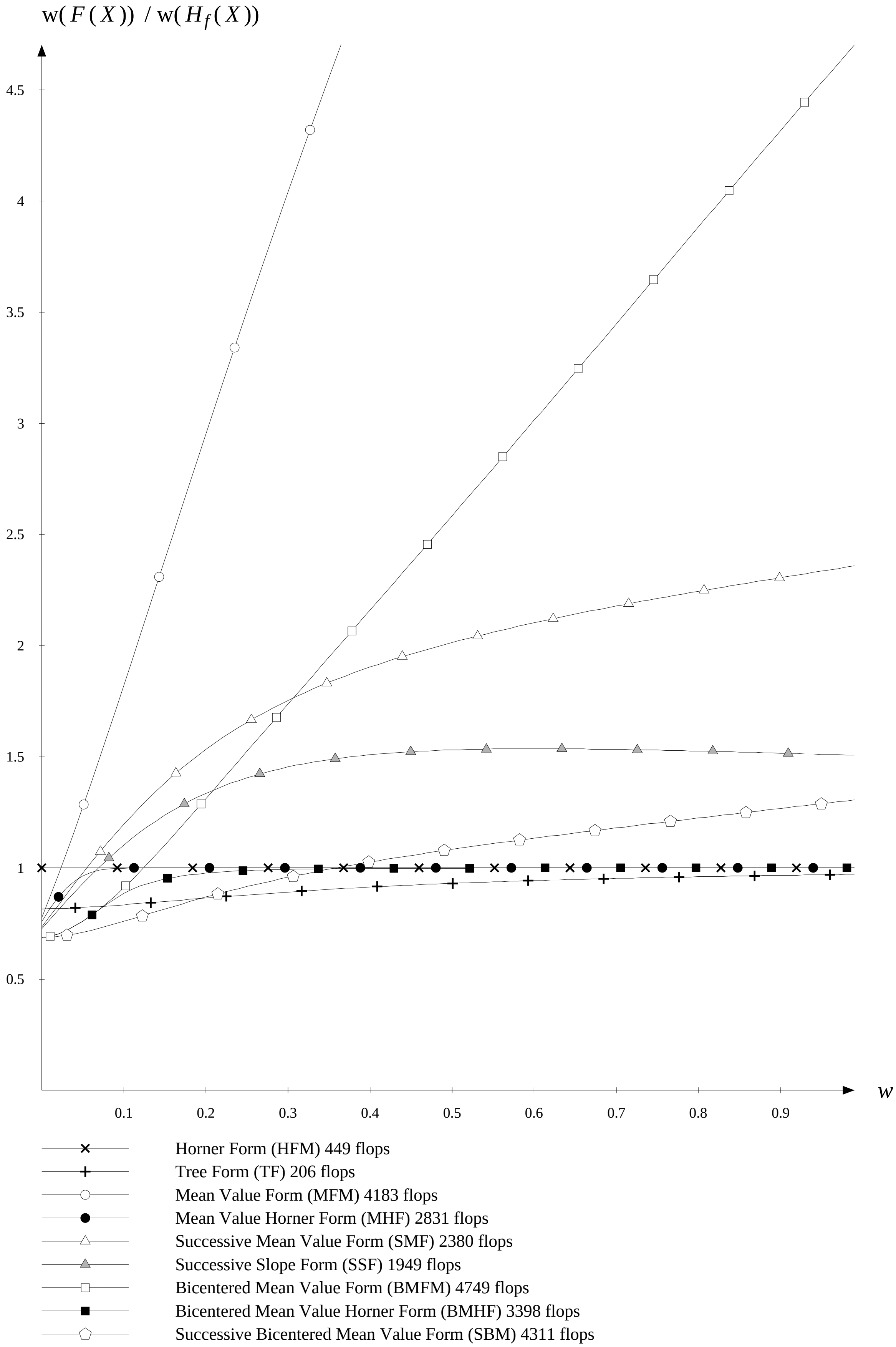


Figure 5.1: Accuracy for random polynomials with $n = 8$ variables, $m = 10$ monomials and degree $\leq 5$ in each variable. The degree vectors have 8 non–zero components. The widths of the $X_i$ are randomly chosen from $[0, w]$ and $\mathrm{mid}(X_i) \equiv 0.5$ for all $i$.

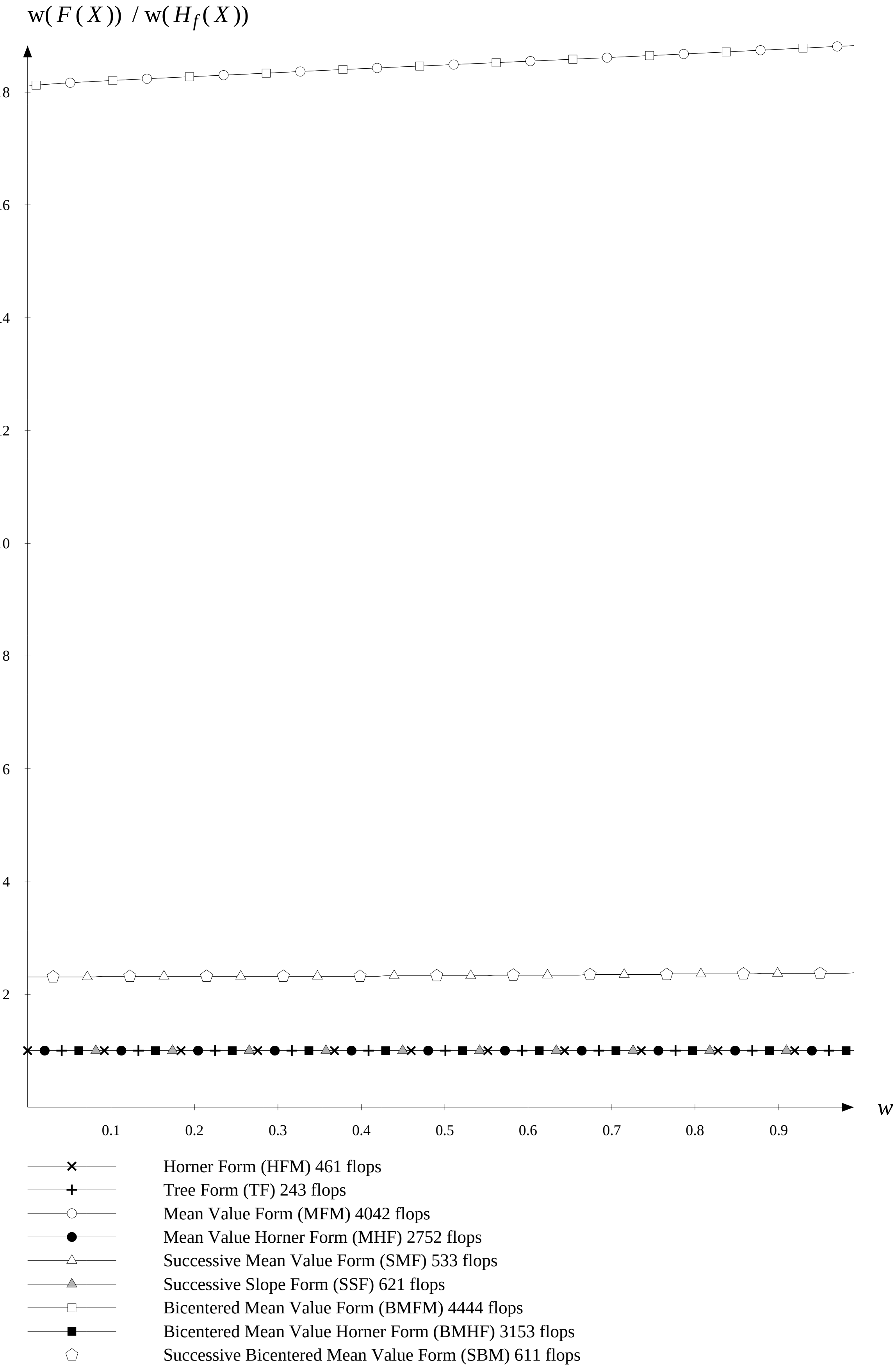


Figure 5.2: Accuracy for random polynomials with $n = 8$ variables, $m = 10$ monomials and degree $\leq 5$ in each variable. The degree vectors have 8 non–zero components. The widths of the $X_i$ are randomly chosen from $[0, w]$ and $\mathrm{mid}(X_i) \equiv 0$ for all $i$.

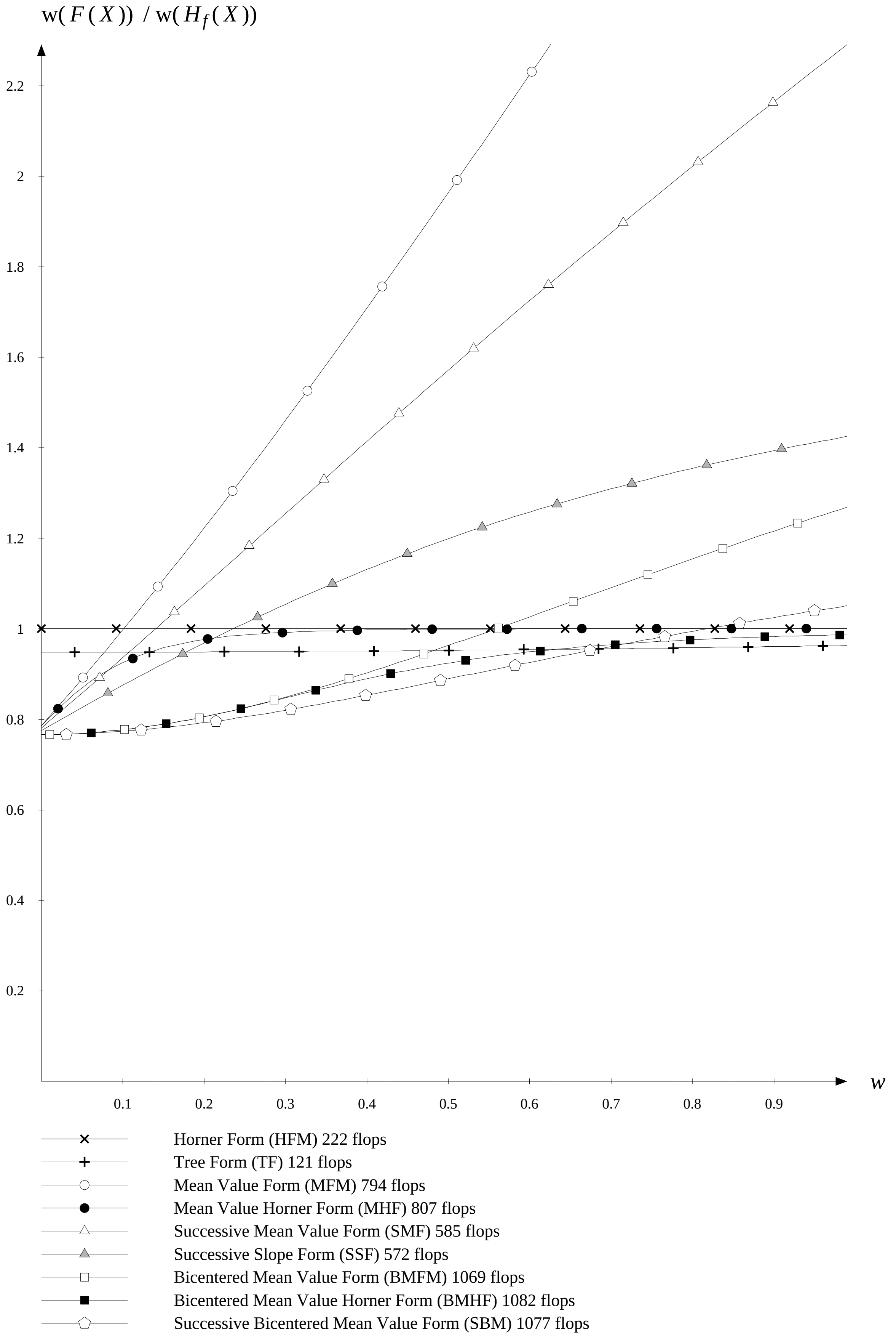


Figure 5.3: Accuracy for random polynomials with $n = 8$ variables, $m = 10$ monomials and degree $\leq 5$ in each variable. The degree vectors have 2 non–zero components. The widths of the $X_i$ are randomly chosen from $[0, w]$ and $\mathrm{mid}(X_i) \equiv 0.5$ for all $i$.

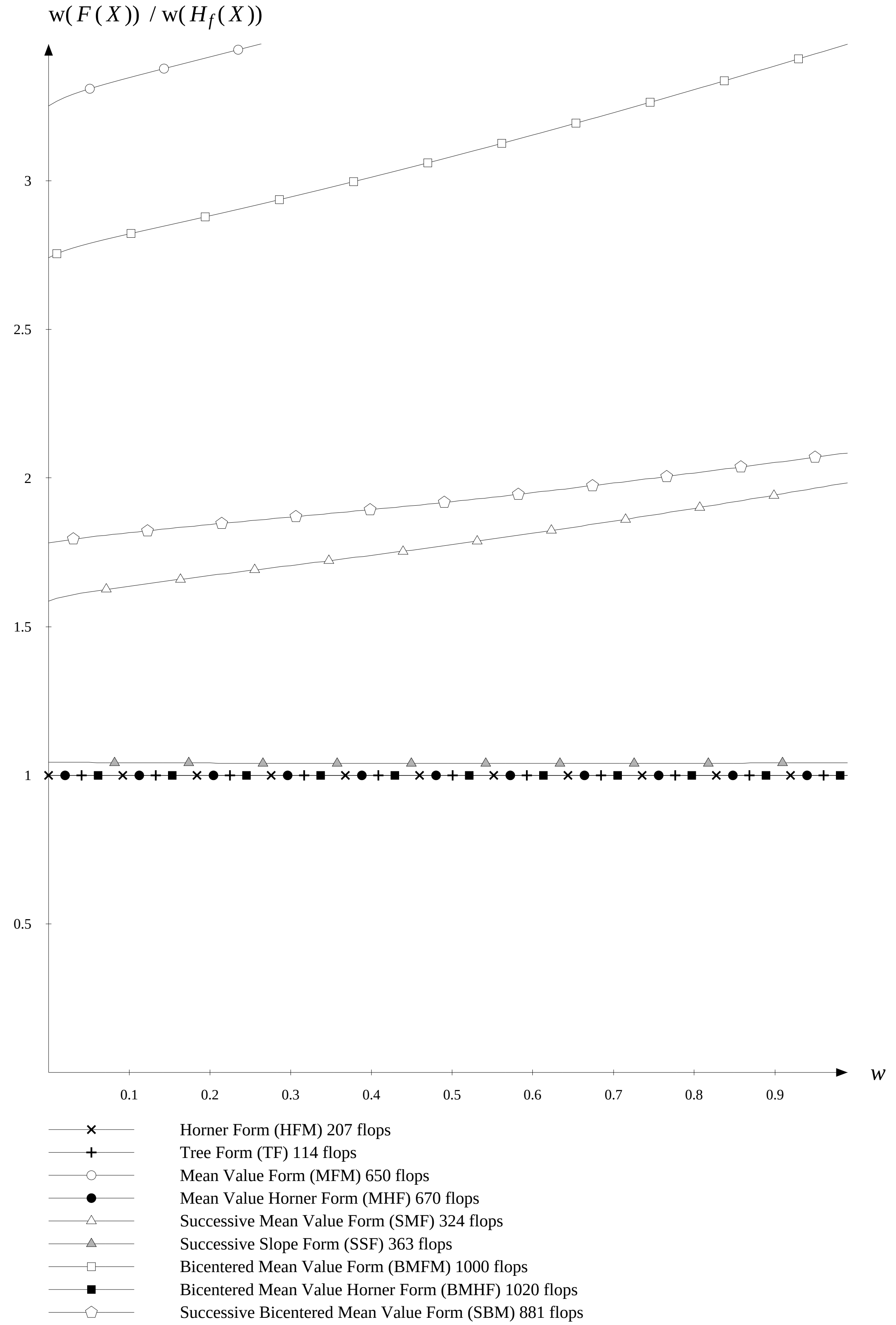


Figure 5.4: Accuracy for random polynomials with $n = 8$ variables, $m = 10$ monomials and degree $\leq 5$ in each variable. The degree vectors have 2 non–zero components. The widths of the $X_i$ are randomly chosen from $[0, w]$ and $\mathrm{mid}(X_i) \equiv 0$ for all $i$.

- Instead of $\ddot{M}_f(\boldsymbol{X})$, $\ddot{M}_f^{(\mathrm{H})}(\boldsymbol{X})$ and $\ddot{M}_f^{\leadsto}(\boldsymbol{X})$ we use the intervals

$$\begin{aligned}&[\underline{M_f(\boldsymbol{X},\boldsymbol{c}^{\downarrow})}, H_f(\boldsymbol{c}^{\downarrow})]\\&[\underline{M_f(\boldsymbol{X},\boldsymbol{c}^{\downarrow})}, H_f(\boldsymbol{c}^{\downarrow})] \cap H_f(\boldsymbol{X})\\&[\underline{M_f^{\leadsto}(\boldsymbol{X},\boldsymbol{c}^{\downarrow})}, H_f(\boldsymbol{c}^{\downarrow})],\end{aligned}$$

  where $\boldsymbol{c}^{\downarrow}$ is as in Definition 4.19 respectively Definition 4.31. In the case of $M_f^{\leadsto}$, this reduces the costs approximately by half.

- Instead of $M_f(\boldsymbol{X})$, $M_f^{(\mathrm{H})}(\boldsymbol{X})$,$M_f^{\leadsto}(\boldsymbol{X})$ and $M_f^{\Rightarrow}(\boldsymbol{X})$ we use the intervals

$$\begin{aligned}&[\underline{M_f(\boldsymbol{X})}, H_f(\boldsymbol{c})]\\&[\underline{M_f(\boldsymbol{X})}, H_f(\boldsymbol{c})] \cap H_f(\boldsymbol{X})\\&[\underline{M_f^{\leadsto}(\boldsymbol{X})}, H_f(\boldsymbol{c})]\\&[\underline{M_f^{\Rightarrow}(\boldsymbol{X})}, H_f(\boldsymbol{c})],\end{aligned}$$

  which give tighter inclusions of the minumum with no additional costs.

The algorithms can be improved if intermediate results such as partially evaluated polynomials are memorized between the iterations. However, in order to keep the presentation simple we will not make use of this. In Figure 5.5 – Figure 5.7 we trace the width of the inclusion of the global minimum in dependence of the number of executed arithmetic floating point operations. The figures show the average over 1000 random polynomials where each polynomial has $m = 5$ monomials and the degree in each variable is at most 5. The search interval $\boldsymbol{A}$ is in all cases $A_i = [-1, 1]$, $i = 1, \ldots, n$.

- In Figure 5.5 the polynomials have 3 variables and each degree vector has 2 non–zero components. The computation is stopped after 20 Kflop. The bicentered forms are best in the order $\ddot{M}_f^{\leadsto}$, $\ddot{M}_f^{(\mathrm{H})}$, $\ddot{M}_f$. Next comes $N_f$, which is slightly better than $H_f$ and $M_f^{\Rightarrow}$.
- In Figure 5.6 the polynomials have $n = 6$ variables and the degree vectors are dense. The computation is stopped after 150 Kflop. Here, $N_f$ is clearly best, followed by $H_f$ and $\ddot{M}_f^{(\mathrm{H})}$.
- In Figure 5.7 the polynomials have $n = 6$ variables and each degree vector has 2 non–zero components. The computation is stopped after 150 Kflop. As in Figure 5.5, the bicentered forms are best, but this time $\ddot{M}_f^{(\mathrm{H})}$ is slightly better than $\ddot{M}_f^{\leadsto}$.

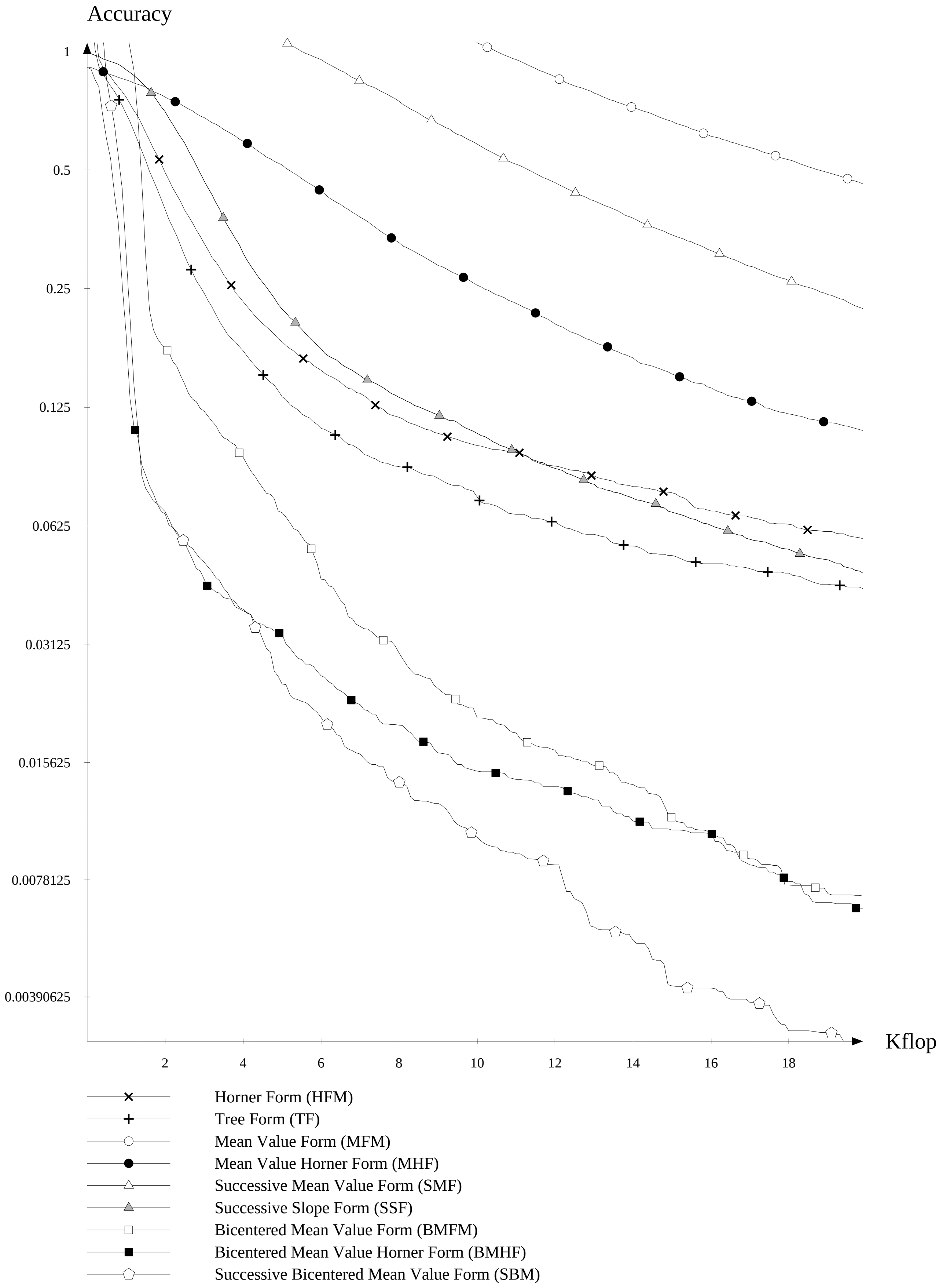


Figure 5.5: Accuracy for random polynomials with $n = 3$ variables, $m = 5$ monomials and degree $\leq 5$ in each variable. The degree vectors have 2 non–zero components.

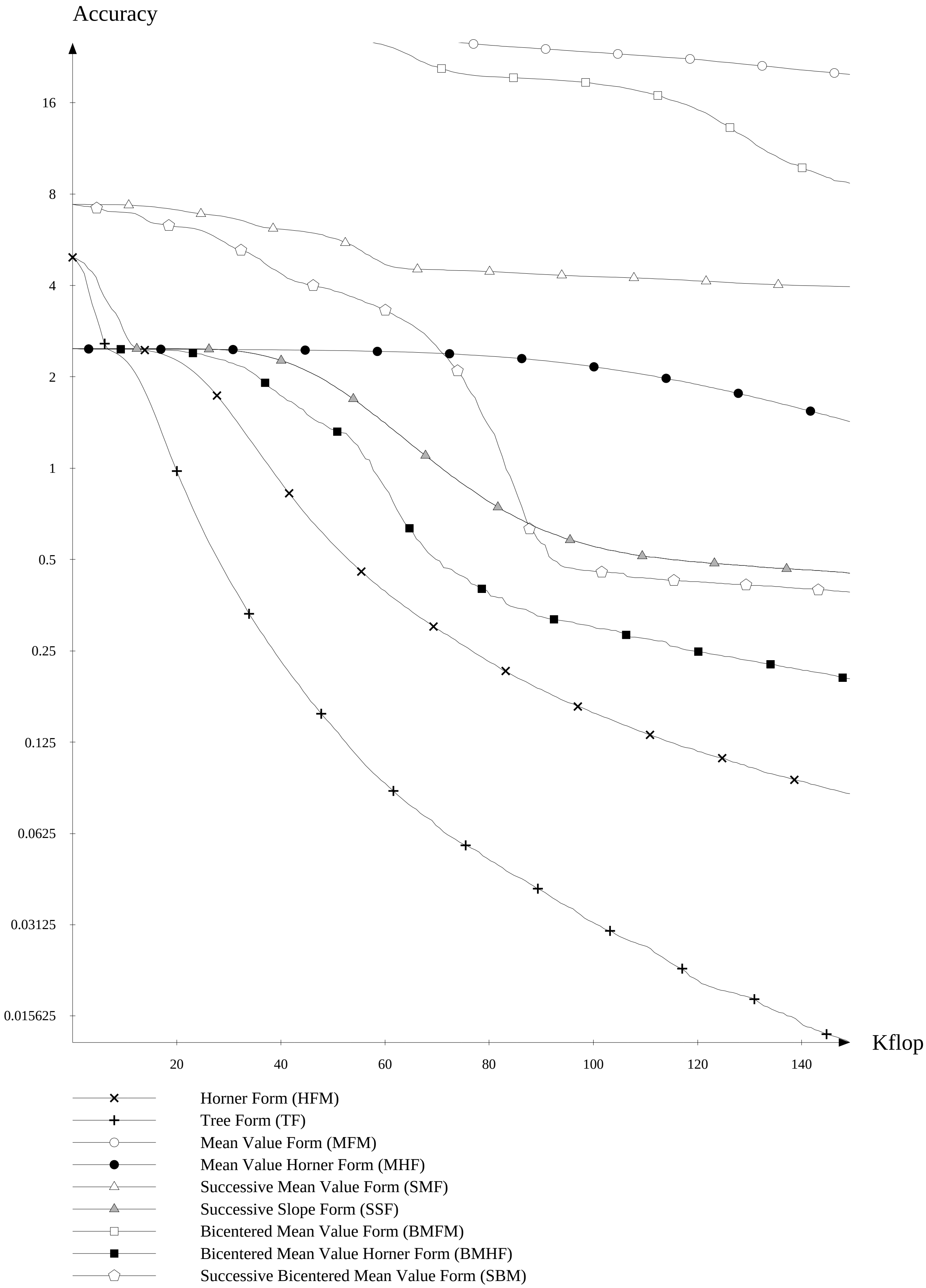


Figure 5.6: Accuracy for random polynomials with $n = 6$ variables, $m = 5$ monomials and degree $\leq 5$ in each variable. The degree vectors have 6 non–zero components.

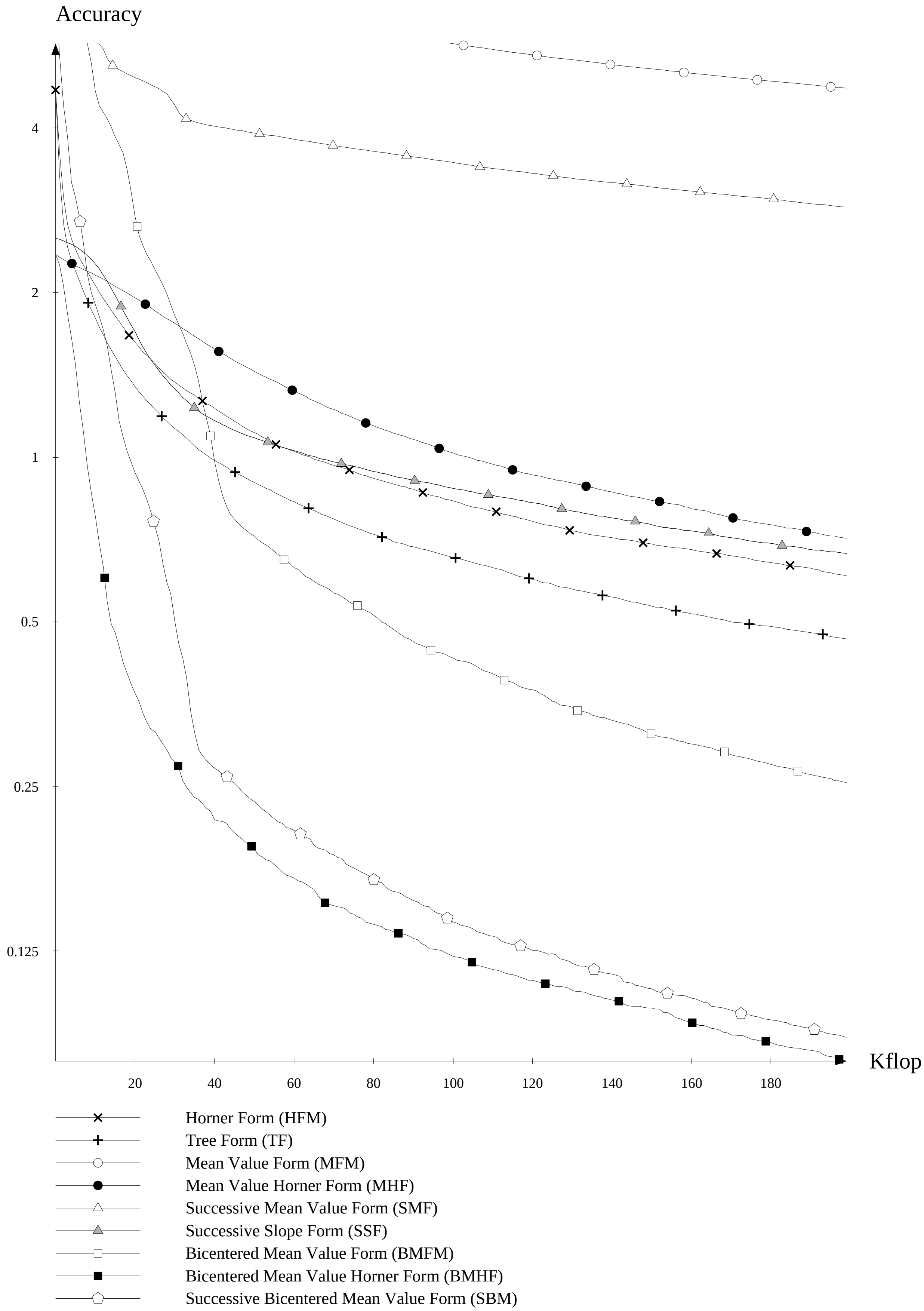


Figure 5.7: Accuracy for random polynomials with $n = 6$ variables, $m = 5$ monomials and degree $\leq 5$ in each variable. The degree vectors have 2 non–zero components.

# Bibliography

**[Aho *et al.*, 1977]**
A. V. Aho, S. C. Johnson, and J. D. Ullman.
Code Generation for Expressions with Common Subexpression.
*J. of the Association for Computing Machinery*, 24(1):146–160, 1977.

**[Alefeld and Herzberger, 1974]**
G. Alefeld and J. Herzberger.
*Einführung in die Intervallrechnung.*
Bibliographisches Institut, Mannheim–Wien–Zürich, 1974.

**[Alefeld and Herzberger, 1983]**
G. Alefeld and J. Herzberger.
*Introduction to Interval Computations.*
Academic Press, N.Y., 1983.

**[Baumann, 1988]**
E. Baumann.
Optimal Centered Forms.
*BIT*, 28:80–87, 1988
**Keywords:** Range Computation, Centered Forms.

**[Caprani and Madsen, 1980]**
O. Caprani and K. Madsen.
Mean Value Forms in Interval Analysis.
*Computing*, 25:147–154, 1980
**Keywords:** Range Computation, Centered Forms.

**[Hammer *et al.*, 1993]**
R. Hammer, M. Hocks, U. Kulisch, and D. Ratz.
*Numerical Toolbox for Verified Computing I.*
Springer, 1993.

**[Hansen and Sengupta, 1981]**
E. Hansen and S. Sengupta.
Bounding Solutions of Systems of Equations using Interval Analysis.
*BIT*, 21:203–211, 1981
**Keywords:** Nonlinear Equation Systems, Bounding Solutions.

**[Hansen, 1968]**
E. Hansen.
On Solving Systems of Equations using Interval Arithmetic.
*Math. Comp.*, 22:374–384, 1968
**Keywords:** Range Computation, Nonlinear Equation Systems.

**[Hong and Stahl, 1994]**
H. Hong and V. Stahl.
Safe Starting Regions by Fixed Points and Tightening.
Technical Report 94-29, Research Institute for Symbolic Computation, Johannes Kepler University, Linz, Austria, 1994.

**[Krawczyk and Neumaier, 1985]**
R. Krawczyk and A. Neumaier.
Interval Slopes for Rational Functions and Associated Centered Forms.
*SIAM J. Numerical Analysis*, 22(3):604–615, 1985
**Keywords:** Range Computation, Centered Forms.

**[Krawczyk, 1969]**
R. Krawczyk.
Newton-Algorithmen zur Bestimmung von Nullstellen mit Fehlerschranken.
*Computing*, 4:187–201, 1969
**Keywords:** Nonlinear Equation Systems, Bounding Solutions.

**[Ratschek and Rokne, 1984]**
H. Ratschek and J. Rokne.
*Computer Methods for the Range of Functions.*
Ellis Horwood Limited, 1984.

**[Sethi and Ullman, 1970]**
R. Sethi and J. D. Ullman.
The Generation of Optimal Code for Arithmetic Expressions.
*J. of the Association for Computing Machinery*, 17:715–728, 1970.

**[Skelboe, 1974]**
S. Skelboe.
Computation of Rational Interval Functions.
*BIT*, 14:87–95, 1974
**Keywords:** Range Computation, Iterative Methods.

**[Ullman, 1973]**
J. D. Ullman.
Fast Algorithms for the Elimination of Common Subexpressions.
*Acta Informatica*, 2:191–213, 1973.