Tools for Simplicial Branch and Bound in Global Optimization

Sam Karhbet
and
Ralph Baker Kearfott

Department of Mathematics
University of Louisiana at Lafayette

SCAN 2016, September 27, 2016
11:45–12:10
Introduction

Elements of Branch and Bound (B&B) algorithms

- Our goal is to minimize an objective function $\varphi$ subject to various equality and inequality constraints, and to do this in a mathematically rigorous way.
Introduction

Elements of Branch and Bound (B&B) algorithms

- Our goal is to minimize an objective function $\varphi$ subject to various equality and inequality constraints, and to do this in a mathematically rigorous way.

- The algorithm essentials relevant here are:

```
1  while Termination criteria are not met do
2    Select a region $D$ from a list of unprocessed regions;
      Bound: Apply filters involving bounds on ranges to eliminate $D$ or portions of it from the search;
3    if $D$ cannot be eliminated or stored then
4      Branch: Split $D$ into two or more sub-regions whose union is $D$;
5        Put each of the sub-regions into the list of unprocessed regions;
6    end
7  end
```
Introduction

Bounding ranges of a function $f$ over a region $\mathcal{D}$:
Is $\mathcal{D}$ a box or a simplex?
In most B&B algorithms, $\mathcal{D}$ is a  **box** or  **set of bounds on the coordinates**.
Introduction

Bounding ranges of a function $f$ over a region $\mathcal{D}$: Is $\mathcal{D}$ a box or a simplex?

- In most B&B algorithms, $\mathcal{D}$ is a box or set of bounds on the coordinates.
  - For boxes, bounds on the ranges of functions $f$ can be computed rigorously with simple interval evaluations or with well-studied linear relaxations.
Introduction
Bounding ranges of a function \( f \) over a region \( \mathcal{D} \):
Is \( \mathcal{D} \) a box or a simplex?

- In most B&B algorithms, \( \mathcal{D} \) is a box or set of bounds on the coordinates.
  - For boxes, bounds on the ranges of functions \( f \) can be computed rigorously with simple interval evaluations or with well-studied linear relaxations.
- In some problems, the natural region is an \( n \)-simplex (e.g. a triangle for \( n = 2 \), a tetrahedron for \( n = 3 \), defined by \( n + 1 \) vertices), rather than a box.
Introduction
Bounding ranges of a function $f$ over a region $\mathcal{D}$:
Is $\mathcal{D}$ a box or a simplex?

- In most B&B algorithms, $\mathcal{D}$ is a box or set of bounds on the coordinates.
  - For boxes, bounds on the ranges of functions $f$ can be computed rigorously with simple interval evaluations or with well-studied linear relaxations.
- In some problems, the natural region is an $n$-simplex (e.g. a triangle for $n = 2$, a tetrahedron for $n = 3$, defined by $n + 1$ vertices), rather than a box.
  - Rigorously bounding ranges over a simplex has been less studied.
Introduction

Bounding ranges of a function $f$ over a region $\mathcal{D}$:
Is $\mathcal{D}$ a box or a simplex?

- In most B&B algorithms, $\mathcal{D}$ is a box or set of bounds on the coordinates.
  - For boxes, bounds on the ranges of functions $f$ can be computed rigorously with simple interval evaluations or with well-studied linear relaxations.
- In some problems, the natural region is an $n$-simplex (e.g. a triangle for $n = 2$, a tetrahedron for $n = 3$, defined by $n + 1$ vertices), rather than a box.
  - Rigorously bounding ranges over a simplex has been less studied.
  - Two different representations of a simplex are useful in B&B algorithms, and how do we convert between these representations?
Related Work
Simplicial B&B and range computation over simplices

- Stenger, Kearfott, Stynes (1970’s)
Related Work
Simplicial B&B and range computation over simplices

- **Stenger, Kearfott, Stynes (1970’s)**
  - These works used a B&B algorithm based on simplicial subdivision to compute the topological degree of maps.
Related Work

Simplicial B&B and range computation over simplices

- Stenger, Kearfott, Stynes (1970’s)
  - These works used a B&B algorithm based on simplicial subdivision to compute the topological degree of maps.
  - These used heuristics to bound ranges.

- Garloff and others (1980’s to the present)
  - Mathematically rigorous range computation of polynomials and rational functions is done with Bernstein polynomial expansions over simplices.
  - Nataraj has used this approach to range computation of polynomials, but, to our knowledge, not over simplices.
  - We are looking forward to investigation of the relative efficiency of these techniques.

- Paulavičius, Žilinskas, et al (current)
  - They have extensively studied use of simplices in B&B algorithms for optimization.
  - However, their published results involve heuristic or probabilistic bounds for ranges.
Related Work
Simplicial B&B and range computation over simplices

- **Stenger, Kearfott, Stynes (1970’s)**
  - These works used a B&B algorithm based on simplicial subdivision to compute the topological degree of maps.
  - These used heuristics to bound ranges.

- **Garloff and others (1980’s to the present)**
Related Work
Simplicial B&B and range computation over simplices

- **Stenger, Kearfott, Stynes** (1970’s)
  - These works used a B&B algorithm based on simplicial subdivision to compute the topological degree of maps.
  - These used heuristics to bound ranges.

- **Garloff and others** (1980’s to the present)
  - Mathematically rigorous range computation of polynomials and rational functions is done with Bernstein polynomial expansions over simplices.
Related Work
Simplicial B&B and range computation over simplices

► **Stenger, Kearfott, Stynes** (1970’s)
  - These works used a B&B algorithm based on simplicial subdivision to compute the topological degree of maps.
  - These used heuristics to bound ranges.

► **Garloff and others** (1980’s to the present)
  - Mathematically rigorous range computation of polynomials and rational functions is done with Bernstein polynomial expansions over simplices.
  - **Nataraj** has used this approach to range computation of polynomials, but, to our knowledge, not over simplices.
Related Work
Simplicial B&B and range computation over simplices

- Stenger, Kearfott, Stynes (1970’s)
  - These works used a B&B algorithm based on simplicial subdivision to compute the topological degree of maps.
  - These used heuristics to bound ranges.

- Garloff and others (1980’s to the present)
  - Mathematically rigorous range computation of polynomials and rational functions is done with Bernstein polynomial expansions over simplices.
  - Nataraj has used this approach to range computation of polynomials, but, to our knowledge, not over simplices.
  - We are looking forward to investigation of the relative efficiency efficiency of these techniques.

- Paulavičius, Žilinskas, et al (current)
Related Work
Simplicial B&B and range computation over simplices

- **Stenger, Kearfott, Stynes** (1970’s)
  - These works used a B&B algorithm based on simplicial subdivision to compute the topological degree of maps.
  - These used heuristics to bound ranges.

- **Garloff and others** (1980’s to the present)
  - Mathematically rigorous range computation of polynomials and rational functions is done with Bernstein polynomial expansions over simplices.
  - Nataraj has used this approach to range computation of polynomials, but, to our knowledge, not over simplices.
  - We are looking forward to investigation of the relative efficiency of these techniques.

- **Paulavičius, Žilinskas, et al** (current)
  - They have extensively studied use of simplices in B&B algorithms for optimization.
Related Work
Simplicial B&B and range computation over simplices

- **Stenger, Kearfott, Stynes** (1970’s)
  - These works used a B&B algorithm based on simplicial subdivision to compute the topological degree of maps.
  - These used heuristics to bound ranges.

- **Garloff and others** (1980’s to the present)
  - Mathematically rigorous range computation of polynomials and rational functions is done with Bernstein polynomial expansions over simplices.
  - **Nataraj** has used this approach to range computation of polynomials, but, to our knowledge, not over simplices.
  - We are looking forward to investigation of the relative efficiency of these techniques.

- **Paulavičius, Žilinskas, et al** (current)
  - They have extensively studied use of simplices in B&B algorithms for optimization.
  - However, their published results involve heuristic or probabilistic bounds for ranges.
Two Simplex Representations
Vertex and halfspace representations

Introduction

Previous Work

Two Simplex Representations

Various Bounding Strategies

Converting Between Representations
Two Simplex Representations
Vertex and halfspace representations

The vertex representation of a simplex \( \mathcal{D} = S \) is in terms of the cartesian coordinates of its \( n + 1 \) vertices, i.e.
\[
S = \langle P_0, P_1, \ldots, P_n \rangle.
\]
Two Simplex Representations
Vertex and halfspace representations

- **The vertex representation** of a simplex $\mathcal{D} = S$ is in terms of the cartesian coordinates of its $n + 1$ vertices, i.e.
  $S = \langle P_0, P_1, \ldots, P_n \rangle$.

- **The half-plane representation** of a simplex is in terms of the feasible set of $n + 1$ inequalities $Ax \geq b$, $A \in \mathbb{R}^{n+1 \times n}$, $b \in \mathbb{R}^{n+1}$.
Two Simplex Representations
Vertex and halfspace representations

- The **vertex representation** of a simplex $D = S$ is in terms of the cartesian coordinates of its $n + 1$ vertices, i.e. $S = \langle P_0, P_1, \ldots P_n \rangle$.

- The **half-plane representation** of a simplex is in terms of the feasible set of $n + 1$ inequalities $Ax \geq b$, $A \in \mathbb{R}^{n+1 \times n}$, $b \in \mathbb{R}^{n+1}$.

  - Each face $S_{-i}$ of $S$ opposite a vertex $P_i$ of $S$ is contained in a hyperplane $\tilde{A}_i : x = b_i$, where $\tilde{A}_i : = \pm A_i :$. 

  - Each face $S_{-i}$ of $S$ opposite a vertex $P_i$ of $S$ is contained in a hyperplane $\tilde{A}_i : x = b_i$, where $\tilde{A}_i : = \pm A_i :$. 

- The vertex representation is most useful in the branching, etc., while the halfspace representation is most useful in constraint-propagation-based filters.

- We have studied mathematically rigorous conversions between these two representations.
Two Simplex Representations
Vertex and halfspace representations

- The vertex representation of a simplex $\mathcal{D} = S$ is in terms of the cartesian coordinates of its $n + 1$ vertices, i.e.
  $S = \langle P_0, P_1, \ldots P_n \rangle$.

- The half-plane representation of a simplex is in terms of the feasible set of $n + 1$ inequalities $Ax \geq b$, $A \in \mathbb{R}^{n+1 \times n}$, $b \in \mathbb{R}^{n+1}$.
  - Each face $S_{\neg i}$ of $S$ opposite a vertex $P_i$ of $S$ is contained in a hyperplane $\tilde{A}_i \cdot x = b_i$, where $\tilde{A}_i = \pm A_i$.
  - The side of the hyperplane upon which $P_i$ lies determines the sense of the inequality $A_i \cdot x \geq b_i$. 

▶ The vertex representation is most useful in the branching, etc., while the halfspace representation is most useful in constraint-propagation-based filters.

▶ We have studied mathematically rigorous conversions between these two representations.
Two Simplex Representations
Vertex and halfspace representations

- The **vertex representation** of a simplex $D = S$ is in terms of the cartesian coordinates of its $n + 1$ vertices, i.e. $S = \langle P_0, P_1, \ldots, P_n \rangle$.

- The **half-plane representation** of a simplex is in terms of the feasible set of $n + 1$ inequalities $Ax \geq b$, $A \in \mathbb{R}^{n+1 \times n}$, $b \in \mathbb{R}^{n+1}$.
  - Each face $S_{\neg i}$ of $S$ opposite a vertex $P_i$ of $S$ is contained in a hyperplane $\tilde{A}_i;:x = b_i$, where $\tilde{A}_i; = \pm A_i;:$.
  - The side of the hyperplane upon which $P_i$ lies determines the sense of the inequality $A_i;:x \geq b_i$.

- The vertex representation is most useful in the branching, etc., while the halfspace representation is most useful in constraint-propagation-based filters.
Two Simplex Representations
Vertex and halfspace representations

- The vertex representation of a simplex $\mathcal{D} = S$ is in terms of the cartesian coordinates of its $n + 1$ vertices, i.e. $S = \langle P_0, P_1, \ldots P_n \rangle$.

- The half-plane representation of a simplex is in terms of the feasible set of $n + 1$ inequalities $Ax \geq b$, $A \in \mathbb{R}^{n+1 \times n}$, $b \in \mathbb{R}^{n+1}$.
  - Each face $S_{-i}$ of $S$ opposite a vertex $P_i$ of $S$ is contained in a hyperplane $\tilde{A}_{i,:}x = b_i$, where $\tilde{A}_{i,:} = \pm A_{i,:}$.
  - The side of the hyperplane upon which $P_i$ lies determines the sense of the inequality $A_{i,:}x \geq b_i$.

- The vertex representation is most useful in the branching, etc., while the halfspace representation is most useful in constraint-propagation-based filters.

- We have studied mathematically rigorous conversions between these two representations.
Bounding the Range of $f$ Over a Simplex

$f$ is normally represented in terms of cartesian (box-based) coordinates.

Various possibilities

Simplicial Branch and Bound Tools

Introduction

Previous Work

Two Simplex Representations

Various Bounding Strategies

Converting Between Representations
Bounding the Range of $f$ Over a Simplex

$f$ is normally represented in terms of cartesian (box-based) coordinates.

Various possibilities

- We can enclose $S$ in a box, then use traditional interval extensions over the box.
Bounding the Range of $f$ Over a Simplex

$f$ is normally represented in terms of cartesian (box-based) coordinates.

Various possibilities

- We can enclose $S$ in a box, then use traditional interval extensions over the box.
  - This is simple, but with significant overestimation.
Bounding the Range of $f$ Over a Simplex

$f$ is normally represented in terms of cartesian (box-based) coordinates.

Various possibilities

- We can enclose $S$ in a box, then use traditional interval extensions over the box.
  - This is simple, but with significant overestimation.
- We can use the halfspace representation and constraint propagation.
Bounding the Range of $f$ Over a Simplex

$f$ is normally represented in terms of cartesian (box-based) coordinates.

Various possibilities

- We can enclose $S$ in a box, then use traditional interval extensions over the box.
  - This is simple, but with significant overestimation.
- We can use the halfspace representation and constraint propagation.
  - This can result in less overestimation, but not necessarily.
Bounding the Range of $f$ Over a Simplex

$f$ is normally represented in terms of cartesian (box-based) coordinates.

Various possibilities

- We can enclose $S$ in a box, then use traditional interval extensions over the box.
  - This is simple, but with significant overestimation.

- We can use the halfspace representation and constraint propagation.
  - This can result in less overestimation, but not necessarily.
  - This adds complication and, depending on the problem and how implemented, could involve an amount of computation comparable to that required to totally solve the original problem.
Bounding the Range of $f$ Over a Simplex

$f$ is normally represented in terms of cartesian (box-based) coordinates.

Various possibilities

- We can enclose $S$ in a box, then use traditional interval extensions over the box.
  - This is simple, but with significant overestimation.

- We can use the halfspace representation and constraint propagation.
  - This can result in less overestimation, but not necessarily.
  - This adds complication and, depending on the problem and how implemented, could involve an amount of computation comparable to that required to totally solve the original problem.

- We can analyze relationships between coordinates in the simplex to derive simple formulas that give sharper bounds than interval extensions over the containing boxes.
Bounding the Range of $f$ Over a Simplex $S$

A specially derived formula for $S = \langle P_0, P_1, \ldots P_n \rangle$, with $P_i = (p_{i,1}, \ldots, p_{i,n})$

Begin with non-sharp bounds $f = [\underline{f}, \bar{f}]$, say, obtained by evaluating over a box $\mathbf{x}$ containing $S$. 

$\underline{f}$ is often narrower than $\bar{f}$.

(The theorem is proven by considering $S$ in terms of barycentric coordinates and an associated LP.)
Bounding the Range of $f$ Over a Simplex $S$

A specially derived formula for $S = \langle P_0, P_1, \ldots, P_n \rangle$, with $P_i = (p_{i,1}, \ldots, p_{i,n})$

Begin with non-sharp bounds $f = [\underline{f}, \overline{f}]$, say, obtained by evaluating over a box $x$ containing $S$.

**Theorem:**

\[
\begin{align*}
L_i &= \text{Inf} \left( \sum_{j=1}^{n} p_{i,j} \tilde{f}_j(\text{sgn}(p_{i,j})) \right), \quad \text{and} \\
U_i &= \text{Sup} \left( \sum_{j=1}^{n} p_{i,j} \tilde{f}_j(-\text{sgn}(p_{i,j})) \right), \quad \text{where} \\
\tilde{f}_j(p) &= \begin{cases} 
\underline{f}_j & \text{if } p \geq 0, \\
\overline{f}_j & \text{if } p < 0, \\
\underline{f}_j & \text{if } 0 \in p
\end{cases}.
\end{align*}
\]
Bounding the Range of $f$ Over a Simplex $S$

A specially derived formula for $S = \langle P_0, P_1, \ldots, P_n \rangle$, with $P_i = (p_{i,1}, \ldots, p_{i,n})$

Begin with non-sharp bounds $f = [\underline{f}, \bar{f}]$, say, obtained by evaluating over a box $x$ containing $S$.

**Theorem:**

Let

$$L_i = \inf \left( \sum_{j=1}^{n} p_{i,j} \bar{f}_j(\text{sgn}(p_{i,j})) \right),$$

$$U_i = \sup \left( \sum_{j=1}^{n} p_{i,j} \bar{f}_j(-\text{sgn}(p_{i,j})) \right),$$

where

$$\bar{f}_j(p) = \begin{cases} f_j & \text{if } p \geq 0, \\ \bar{f}_j & \text{if } p < 0, \\ f_j & \text{if } 0 \in p \end{cases}.$$

**Assume** the domain of $f$ has been translated so the barycenter $\frac{1}{n+1} \sum_{i=0}^{n} P_i$ is the origin $(0, \ldots, 0)$, and the range of $f$ has been translated so $f(0, \ldots, 0) = 0$. 

$I_0$ is often narrower than $f$. (The theorem is proven by considering $S$ in terms of barycentric coordinates and an associated LP.)
Bounding the Range of $f$ Over a Simplex $S$

A specially derived formula for $S = \langle P_0, P_1, \ldots, P_n \rangle$, with

$P_i = (p_{i,1}, \ldots, p_{i,n})$

Begin with non-sharp bounds $f = [\ell, \bar{f}]$, say, obtained by evaluating over a box $x$ containing $S$.

**Theorem:**

Let $L_i = \text{Inf} \left( \sum_{j=1}^{n} p_{i,j} \mathring{f}_j(\text{sgn}(p_{i,j})) \right)$, and

$U_i = \text{Sup} \left( \sum_{j=1}^{n} p_{i,j} \mathring{f}_j(-\text{sgn}(p_{i,j})) \right)$, where

$\mathring{f}_j(p) = \begin{cases} f_j & \text{if } p \geq 0, \\ \bar{f}_j & \text{if } p < 0, \\ f_j & \text{if } 0 \in p \end{cases}$.

Assume the domain of $f$ has been translated so the barycenter $\frac{1}{n+1} \sum_{i=0}^{n} P_i$ is the origin $(0, \ldots, 0)$, and the range of $f$ has been translated so $f(0, \ldots, 0) = 0$.

Then the range of $f$ over $S$ is contained in the interval $I_0 = [\min_{0 \leq i \leq n} L_i, \max_{0 \leq i \leq n} U_i]$. 
Bounding the Range of $f$ Over a Simplex $S$

A specially derived formula for $S = \langle P_0, P_1, \ldots, P_n \rangle$, with $P_i = (p_{i,1}, \ldots, p_{i,n})$

Begin with non-sharp bounds $f = [\ell, \bar{f}]$, say, obtained by evaluating over a box $x$ containing $S$.

**Theorem:**

\[
L_i = \text{Inf} \left( \sum_{j=1}^{n} p_{i,j} \tilde{f}_j(\text{sgn}(p_{i,j})) \right), \quad \text{and} \\
U_i = \text{Sup} \left( \sum_{j=1}^{n} p_{i,j} \tilde{f}_j(-\text{sgn}(p_{i,j})) \right),
\]

where \( \tilde{f}_j(p) = \begin{cases} f_j & \text{if } p \geq 0, \\ \bar{f}_j & \text{if } p < 0, \\ f_j & \text{if } 0 \in p \end{cases} \).

Assume the domain of $f$ has been translated so the barycenter \( \frac{1}{n+1} \sum_{i=0}^{n} P_i \) is the origin \((0, \ldots, 0)\), and the range of $f$ has been translated so $f(0, \ldots, 0) = 0$.

Then the range of $f$ over $S$ is contained in the interval $I_0 = [\min_{0 \leq i \leq n} L_i, \max_{0 \leq i \leq n} U_i]$.

\( I_0 \) is often narrower than $f$. 

\[8 / 12\]
Bounding the Range of $f$ Over a Simplex $S$

A specially derived formula for $S = \langle P_0, P_1, \ldots, P_n \rangle$, with $P_i = (p_{i,1}, \ldots, p_{i,n})$

Begin with non-sharp bounds $f = [\underline{f}, \overline{f}]$, say, obtained by evaluating over a box $x$ containing $S$.

**Theorem:**

Let $L_i = \text{Inf} \left( \sum_{j=1}^{n} p_{i,j} \tilde{f}_j(\text{sgn}(p_{i,j})) \right)$, and $U_i = \text{Sup} \left( \sum_{j=1}^{n} p_{i,j} \tilde{f}_j(-\text{sgn}(p_{i,j})) \right)$, where $
\tilde{f}_j(p) = \begin{cases} f_j \text{ if } p \geq 0, \\ \overline{f}_j \text{ if } p < 0, \text{ and } f_j \text{ if } 0 \in p \end{cases}$.

Assume the domain of $f$ has been translated so the barycenter $\frac{1}{n+1} \sum_{i=0}^{n} P_i$ is the origin $(0, \ldots, 0)$, and the range of $f$ has been translated so $f(0, \ldots, 0) = 0$.

Then the range of $f$ over $S$ is contained in the interval $I_0 = [\min_{0 \leq i \leq n} L_i, \max_{0 \leq i \leq n} U_i]$.

$I_0$ is often narrower than $f$.

(The theorem is proven by considering $S$ in terms of barycentric coordinates and an associated LP.)
The Vertex and Halfspace Representations
Computing a rigorous enclosure of $S$ in a halfspace representation from a rigorous enclosure for $S$ in a vertex representation.
The Vertex and Halfspace Representations
Computing a rigorous enclosure of $S$ in a halfspace representation from a rigorous enclosure for $S$ in a vertex representation
The Vertex and Halfspace Representations
Computing a rigorous enclosure of $S$ in a halfspace representation from a rigorous enclosure for $S$ in a vertex representation

- We bound the set of all possible halfplane equations subject to uncertainties in the vertices.
The Vertex and Halfspace Representations

Computing a rigorous enclosure of $S$ in a halfspace representation from a rigorous enclosure for $S$ in a vertex representation

- We bound the set of all possible halfplane equations subject to uncertainties in the vertices.
- We select certain halfplanes arbitrarily to construct the system $Ax \geq b$. 
Vertex Enclosure to Halfspace Enclosure

The computations for the $i$-th halfspace, $0 \leq i \leq n$ corresponding to $S_{-i} = \langle \tilde{P}_0, \tilde{P}_1, \ldots, \tilde{P}_{n-1} \rangle$
Vertex Enclosure to Halfspace Enclosure

The computations for the $i$-th halfspace, $0 \leq i \leq n$
corresponding to $S_{-i} = \langle \tilde{P}_0, \tilde{P}_1, \ldots, \tilde{P}_{n-1} \rangle$

- Begin with enclosures $\tilde{P}_i$ to the actual vertices $\tilde{P}_i$. 
Vertex Enclosure to Halfspace Enclosure

The computations for the $i$-th halfspace, $0 \leq i \leq n$
corresponding to $S_{-i} = \langle \tilde{P}_0, \tilde{P}_1, \ldots, \tilde{P}_{n-1} \rangle$

- Begin with enclosures $\tilde{P}_i$ to the actual vertices $\tilde{P}_i$.
- For the $i$-th row of $A$, consider an interval enclosure to the system

$$Ma_i = \begin{pmatrix}
(\tilde{P}_1 - \tilde{P}_0)^T \\
\vdots \\
(\tilde{P}_{n-1} - \tilde{P}_0)^T
\end{pmatrix} a_i = 0.$$
Vertex Enclosure to Halfspace Enclosure

The computations for the $i$-th halfspace, $0 \leq i \leq n$
corresponding to $S_{-i} = \langle \tilde{P}_0, \tilde{P}_1, \ldots, \tilde{P}_{n-1} \rangle$

- Begin with enclosures $\tilde{P}_i$ to the actual vertices $\tilde{P}_i$.
- For the $i$-th row of $A$, consider an interval enclosure to
the system

$$M a_i = \begin{pmatrix} (\tilde{P}_1 - \tilde{P}_0)^T \\ \vdots \\ (\tilde{P}_{n-1} - \tilde{P}_0)^T \end{pmatrix} a_i = 0.$$ 

- We obtain a floating point approximation $z$ to $M \tilde{a}_i = 0$, $\|z\|_2 = 1$ using a common null-space-finding procedure.
Vertex Enclosure to Halfspace Enclosure

The computations for the $i$-th halfspace, $0 \leq i \leq n$ corresponding to $S_{-i} = \langle \bar{P}_0, \bar{P}_1, \ldots, \bar{P}_{n-1} \rangle$

- Begin with enclosures $\bar{P}_i$ to the actual vertices $\bar{P}_i$.
- For the $i$-th row of $A$, consider an interval enclosure to the system

$$Ma_i = \begin{pmatrix} (\bar{P}_1 - \bar{P}_0)^T \\ \vdots \\ (\bar{P}_{n-1} - \bar{P}_0)^T \end{pmatrix} a_i = 0.$$

- We obtain a floating point approximation $z$ to $Ma_i = 0$, $\|z\|_2 = 1$ using a common null-space-finding procedure.
- We construct a sufficiently large box $a^{(0)}$ around $z$, and apply an interval Newton method to the system $Mz = 0$, $z^Tz = 1$ to prove a unique solution for every $M \in M$ and generating an enclosure $a_i$ for the normal vector perpendicular to $S_{-i}$. 
We possibly reverse the sign of \( a_i \) depending on the sign of \( a_i^T(\tilde{P}_i - P_0) \).
We possibly reverse the sign of $a_i$ depending on the sign of $a_i^T(\tilde{P}_i - P_0)$.

Compute $b_i \approx a_i^T\tilde{P}_0$ using floating point computations.
Vertex Enclosure to Halfspace Enclosure
Computations for the $i$-th halfspace (continued)

- We possibly reverse the sign of $a_i$ depending on the sign of $a_i^T(\tilde{P}_i - P_0)$.
- Compute $b_i \approx a_i^T\tilde{P}_0$ using floating point computations.
- Gradually decrease $b_i$ until a $b_j$ with $a_i^TP_j \geq b_i$ for $0 \leq j \leq n$. 

\[\text{Proposition:} \quad H_i = \{x: a_i^T x \geq b_i\} \quad \text{Verification of} \quad a_i^T P_j \geq b_i \quad (j = 0, 1, \ldots, n) \implies S \subset H_i. \]
Vertex Enclosure to Halfspace Enclosure
Computations for the $i$-th halfspace (continued)

- We possibly reverse the sign of $a_i$ depending on the sign of $a_i^T(\tilde{P}_i - P_0)$.
- Compute $b_i \approx a_i^T \tilde{P}_0$ using floating point computations.
- Gradually decrease $b_i$ until a $b_i$ with $a_i^T P_j \geq b_i$ for $0 \leq j \leq n$.
- Proposition: Let $H_i = \{ x : a_i^T x \geq b_i \}$. Verification of $a_i^T P_j \geq b_i$ $(j = 0, 1, \ldots, n)$ implies $S \subset H_i$. 
We possibly reverse the sign of $a_i$ depending on the sign of $a_i^T(\tilde{P}_i - P_0)$.

Compute $b_i \approx a_i^T \tilde{P}_0$ using floating point computations.

Gradually decrease $b_i$ until a $b_i$ with $a_i^T P_j \geq b_i$ for $0 \leq j \leq n$.

**Proposition:** Let $H_i = \{x : a_i^T x \geq b_i\}$. Verification of $a_i^T P_j \geq b_i$ ($j = 0, 1, \ldots, n$) implies $S \subset H_i$.

Since $a_i^T P_j \geq b_i$, $a_i^T P_j \geq b_i$ for any $a_i \in a_i$, so, with the same reasoning behind the proposition, $S \subset H_i = \{x : a_i^T x \geq b_i\}$.
Vertex Enclosure to Halfspace Enclosure
Computations for the \(i\)-th halfspace (continued)

▶ We possibly reverse the sign of \(a_i\) depending on the sign of \(a_i^T(\tilde{P}_i - P_0)\).

▶ Compute \(b_i \approx a_i^T \tilde{P}_0\) using floating point computations.

▶ Gradually decrease \(b_i\) until a \(b_i\) with \(a_i^T P_j \geq b_i\) for \(0 \leq j \leq n\).

▶ Proposition: Let \(H_i = \{x : a_i^T x \geq b_i\}\). Verification of \(a_i^T P_j \geq b_i\) \((j = 0, 1, \ldots, n)\) implies \(S \subset H_i\).

▶ Since \(a_i^T P_j \geq b_i\), \(a_i^T P_j \geq b_i\) for any \(a_i \in a_i\), so, with the same reasoning behind the proposition, \(S \subset H_i = \{x : a_i^T x \geq b_i\}\).

▶ In other words, \(a_i\) can be any floating-point quantity in \(a_i\).
What next?
Comparisons of simplicial-based and box-based B&B

- Sam has initial implementations of the same basic B&B algorithm using both simplices and boxes, incorporating the techniques we have explained here.
What next?
Comparisons of simplicial-based and box-based B&B

► Sam has initial implementations of the same basic B&B algorithm using both simplices and boxes, incorporating the techniques we have explained here.

► We have selected both general test problems and test problems on which there is an underlying simplicial geometry.
What next?
Comparisons of simplicial-based and box-based B&B

Sam has initial implementations of the same basic B&B algorithm using both simplices and boxes, incorporating the techniques we have explained here.

We have selected both general test problems and test problems on which there is an underlying simplicial geometry.

This work is in progress.