Smoothed Analysis of Algorithms
Part II: Binary and Multiobjective Optimization

Heiko Röglin
Department of Computer Science

16 January 2013
Outline

1. **Binary Optimization Problems**
   When does a binary optimization problem have polynomial smoothed complexity?

2. **Multiobjective Optimization**
   How many Pareto-optimal solutions do usually exist?

3. **Conclusions**
1. **Binary Optimization Problems**

   When does a binary optimization problem have *polynomial smoothed complexity*?

2. **Multiobjective Optimization**

   How many *Pareto-optimal solutions* do usually exist?

3. **Conclusions**
Model

**Linear Binary Optimization Problem**

- set of feasible solutions $S \subseteq \{0, 1\}^n$
- solution $x = (x_1, \ldots, x_n) \in S$ consists of $n$ binary variables
- linear objective function
  \[
  \max c^T x = c_1 x_1 + \cdots + c_n x_n
  \]
Linear Binary Optimization Problem

- set of feasible solutions \( S \subseteq \{0, 1\}^n \)
- solution \( x = (x_1, \ldots, x_n) \in S \) consists of \( n \) binary variables
- linear objective function
  \[
  \max c^T x = c_1 x_1 + \cdots + c_n x_n
  \]

\( S \) can encode arbitrary combinatorial structure, e.g., for a given graph, all paths from \( s \) to \( t \), all Hamiltonian cycles, all spanning trees, \ldots

- Knapsack Problem: variable \( x_i \in \{0, 1\} \) for each item \( i \)
  \[
  S = \{ x \mid w_1 x_1 + \cdots + w_n x_n \leq t \}
  \]

- TSP: variable \( x_e \in \{0, 1\} \) for each \( e \in E \)
  \[
  S = \{ x \mid x \text{ encodes Hamiltonian cycle} \}.
  \]
Worst-case Analysis: Adversary chooses $S$ and $c_i \in [-1, 1]$. 
Worst-case Analysis: Adversary chooses $S$ and $c_i \in [-1, 1]$.

Smoothed Analysis:
Adversary chooses $S$ and a probability density $f_i: [-1, 1] \rightarrow [0, \phi]$ for every $c_i$ and some $\phi \geq 1$.
Every $c_i$ is drawn independently according to $f_i$. 
**Worst-case Analysis:** Adversary chooses $S$ and $c_i \in [-1, 1]$.

**Smoothed Analysis:**
Adversary chooses $S$ and a probability density $f_i: [-1, 1] \rightarrow [0, \phi]$ for every $c_i$ and some $\phi \geq 1$.
Every $c_i$ is **drawn independently** according to $f_i$.

**Remarks:**
- $\phi$ large $\approx$ worst case
- $\phi$ small $\approx$ average case
**Worst-case Analysis:** Adversary chooses $S$ and $c_i \in [-1, 1]$.

**Smoothed Analysis:**
Adversary chooses $S$ and a probability density $f_i : [-1, 1] \rightarrow [0, \phi]$ for every $c_i$ and some $\phi \geq 1$. Every $c_i$ is drawn independently according to $f_i$.

**Remarks:**
- $\phi$ large $\approx$ worst case
- $\phi$ small $\approx$ average case
**Worst-case Analysis:** Adversary chooses $S$ and $c_i \in [-1, 1]$.

**Smoothed Analysis:**
Adversary chooses $S$ and a probability density $f_i : [-1, 1] \rightarrow [0, \phi]$ for every $c_i$ and some $\phi \geq 1$. Every $c_i$ is drawn independently according to $f_i$.

**Remarks:**
- $\phi$ large $\approx$ worst case
- $\phi$ small $\approx$ average case
- $S$ is not perturbed!
Main Result

Theorem [Beier, Vöcking (STOC 2004)]

linear binary opt. problem has polynomial smoothed complexity

\[ \iff \]

pseudo-polynomial time \( \text{poly}(n, \max\{|c_i|\}) \) in the worst case

Knapsack Problem: Can be solved in time \( \mathcal{O}(n^2 P) \), where \( P \) is the largest profit.

⇒ polynomial smoothed complexity

TSP: strongly NP-hard (even if all edge lengths are 1 or 2)

⇒ no polynomial smoothed complexity
Main Result

Theorem [Beier, Vöcking (STOC 2004)]

Linear binary opt. problem has polynomial smoothed complexity

\[\iff\]

Pseudo-polynomial time \(\text{poly}(n, \max\{|c_i|\})\) in the worst case

\[\Rightarrow\]

**Knapsack Problem**: Can be solved in time \(O(n^2 P)\), where \(P\) is the largest profit.

\[\Rightarrow\text{ polynomial smoothed complexity}\]
Main Result

Theorem [Beier, Vöcking (STOC 2004)]

linear binary opt. problem has polynomial smoothed complexity

\[ \iff \]

pseudo-polynomial time \( \text{poly}(n, \max\{|c_i|\}) \) in the worst case

- **Knapsack Problem**: Can be solved in time \( O(n^2 P) \), where \( P \) is the largest profit.
  \[ \implies \text{polynomial smoothed complexity} \]

- **TSP**: strongly NP-hard
  (even if all edge lengths are 1 or 2)
  \[ \implies \text{no polynomial smoothed complexity} \]
Polynomial Smoothed Complexity

- \( A = \text{algorithm} \)
- \( \mathcal{I}_n = \text{set of inputs of length } n \)
- \( \text{per}_\phi(I) = \text{perturbation of instance } I \)
- \( T_A(I) = \text{running time of } A \text{ on instance } I \)
Polynomial Smoothed Complexity

- $A$ = algorithm
- $\mathcal{I}_n$ = set of inputs of length $n$
- $\text{per}_\phi(l)$ = perturbation of instance $l$
- $T_A(l)$ = running time of $A$ on instance $l$

**Definition (first attempt):**
Polyn. smoothed compl. $\iff \max_{l \in \mathcal{I}_n} \mathbb{E}[T_A(\text{per}_\phi(l))] = \text{poly}(n, \phi)$
Polynomial Smoothed Complexity

- $A =$ algorithm
- $\mathcal{I}_n =$ set of inputs of length $n$
- $\text{per}_\phi(I) =$ perturbation of instance $I$
- $T_A(I) =$ running time of $A$ on instance $I$

**Definition (first attempt):**
Polyn. smoothed compl. $\iff \max_{I \in \mathcal{I}_n} \mathbb{E}[T_A(\text{per}_\phi(I))] = \text{poly}(n, \phi)$

**Problem:** Not robust against change of machine model.
Polynomial Smoothed Complexity

- $A = \text{algorithm}$
- $\mathcal{I}_n = \text{set of inputs of length } n$
- $\text{per}_\phi (I) = \text{perturbation of instance } I$
- $T_A(I) = \text{running time of } A \text{ on instance } I$

**Definition (first attempt):**
Polyn. smoothed compl. $\iff \max_{I \in \mathcal{I}_n} E \left[ T_A (\text{per}_\phi (I)) \right] = \text{poly}(n, \phi)$

Problem: Not robust against change of machine model.

**Definition**
Algorithm $A$ has polynomial smoothed complexity if there exist $\alpha > 0$ and $\beta > 0$ with
\[ \max_{I \in \mathcal{I}_n} E \left[ T_A (\text{per}_\phi (I))^\alpha \right] \leq \beta n\phi. \]
Idea: Round coefficients $c_i$ and apply pseudo-polyn. algo
**Idea:** Round coefficients $c_i$ and apply pseudo-polyn. algo

\[
\Delta = c^T x^* - c^T x^{**}
\]

\[
\begin{align*}
c_1 &= 0.8961235 \\
c_2 &= 0.2674321 \\
c_3 &= 0.3738725 \\
c_4 &= 0.1902782
\end{align*}
\]
**Idea:** Round coefficients $c_i$ and apply pseudo-polyn. algo

\[ \Delta = c^T x^* - c^T x^{**} \]

$R = \text{max. change due to rounding}$

- $c_1 = 0.8961235$
- $c_2 = 0.2674321$
- $c_3 = 0.3738725$
- $c_4 = 0.1902782$
**Idea:** Round coefficients $c_i$ and apply pseudo-polyn. algo

\[ \Delta = c^T x^* - c^T x^{**} \]

\[ R = \text{max. change due to rounding} \]

- Rounding after the $b$-th bit \( \Rightarrow |c_i - \lfloor c_i \rfloor| \leq 2^{-b} \)
- \( \Rightarrow \forall x \in S : |c^T x - \lfloor c \rfloor^T x| \leq n2^{-b} = R \)

\[ c_1 = 0.8961235 \]
\[ c_2 = 0.2674321 \]
\[ c_3 = 0.3738725 \]
\[ c_4 = 0.1902782 \]
**Idea:** Round coefficients $c_i$ and apply pseudo-polyn. algo

$$\Delta = c^T x^* - c^T x^{**} \quad R = \text{max. change due to rounding}$$

- Rounding after the $b$-th bit $\Rightarrow |c_i - [c_i]| \leq 2^{-b}$
- $\Rightarrow \forall x \in S : |c^T x - [c]^T x| \leq n2^{-b} = R$
- $\Delta > 2R \Rightarrow$ rounding does not change optimal solution

$c_1 = 0.8961235$
$c_2 = 0.2674321$
$c_3 = 0.3738725$
$c_4 = 0.1902782$
Isolation Lemma

For all $S$ and all densities $f_i : [-1, 1] \rightarrow [0, \phi]$

$$\Pr[\Delta < \varepsilon] \leq 2n\phi\varepsilon.$$
Isolation Lemma

For all $S$ and all densities $f_i : [-1, 1] \rightarrow [0, \phi]$

$$ \Pr[\Delta < \varepsilon] \leq 2n\phi\varepsilon. $$

Corollary

For every $p \in (0, 1]$ and $b \geq \log \left( \frac{n^2 \phi}{p} \right) + 2$, 

$$ \Pr[\text{rounding changes optimal solution}] \leq p. $$
Isolation Lemma

For all $S$ and all densities $f_i : [-1, 1] \rightarrow [0, \phi]$

$$\Pr[\Delta < \epsilon] \leq 2n\phi\epsilon.$$  

Corollary

For every $p \in (0, 1]$ and $b \geq \log \left( \frac{n^2 \phi}{p} \right) + 2$,

$$\Pr[\text{rounding changes optimal solution}] \leq p.$$  

pseudo-polynomial algorithm $\Rightarrow$ polynomial smoothed complexity
Isolation Lemma

For all $S$ and all densities $f_i : [-1, 1] \rightarrow [0, \phi]$

$$\Pr[\Delta < \varepsilon] \leq 2n\phi \varepsilon.$$

Corollary

For every $p \in (0, 1]$ and $b \geq \log \left( \frac{n^2 \phi}{p} \right) + 2$,

$$\Pr[\text{rounding changes optimal solution}] \leq p.$$

pseudo-polynomial algorithm $\Rightarrow$ polynomial smoothed complexity

- Round coefficients after a logarithmic number of bits and call pseudo-polynomial algorithm.
Isolation Lemma

For all \( S \) and all densities \( f_i : [-1, 1] \rightarrow [0, \phi] \)

\[
\Pr[\Delta < \varepsilon] \leq 2n\phi\varepsilon.
\]

Corollary

For every \( p \in (0, 1] \) and \( b \geq \log \left( \frac{n^2\phi}{p} \right) + 2 \),

\[
\Pr[\text{rounding changes optimal solution}] \leq p.
\]

decrease.  

 pseudo-polynomial algorithm \Rightarrow\) polynomial smoothed complexity

- Round coefficients after a logarithmic number of bits and call pseudo-polynomial algorithm.
- If necessary, increase precision and repeat.
Proof of Isolation Lemma

**Lemma**

For $\varepsilon \geq 0$, $\Pr[\Delta < \varepsilon] \leq 2n\phi\varepsilon$. 

**Diagram:**

- $c^T x$
- $x_i = 0$  $x_i = 1$
- $\Delta$
- $x^{**}$
- $x^*$
- $S$
Proof of Isolation Lemma

Lemma

For $\varepsilon \geq 0$, $\Pr[\Delta < \varepsilon] \leq 2n\phi\varepsilon$.

- Say $x_i^* = 1$ and $x_i^{**} = 0$ for some $i$. 

---

Heiko Rögnin
Smoothed Analysis of Algorithms
Proof of Isolation Lemma

**Lemma**

For $\varepsilon \geq 0$, $\Pr[\Delta < \varepsilon] \leq 2n\phi\varepsilon$.

- Say $x_i^* = 1$ and $x_i^{**} = 0$ for some $i$.
- Then $x^* = \arg\max_{x \in S} c^T x$ and $x^{**} = \arg\max_{x \in S} c^T x$.

Heiko Röglin  Smoothed Analysis of Algorithms
Proof of Isolation Lemma

**Lemma**

For $\varepsilon \geq 0$, $\Pr[\Delta < \varepsilon] \leq 2n\phi\varepsilon$.

- Say $x_i^* = 1$ and $x_i^{**} = 0$ for some $i$.
- Then $x^* = \arg \max_{x \in S} c^T x$ and $x^{**} = \arg \max_{x \in S} c^T x$.
- Principle of deferred decisions: Fix all $c_j$ for $j \neq i$. 

---

Heiko Röglin  
Smoothed Analysis of Algorithms
Proof of Isolation Lemma

**Lemma**

For $\varepsilon \geq 0$, $\Pr[\Delta < \varepsilon] \leq 2n\phi \varepsilon$.

- Say $x_i^* = 1$ and $x_i^{**} = 0$ for some $i$.
- Then $x^* = \arg\max_{x \in S} c^T x$ and $x^{**} = \arg\max_{x \in S} c^T x$.

**Principle of deferred decisions:** Fix all $c_j$ for $j \neq i$.

$\Rightarrow$ Identity of $x^*$ and $x^{**}$ fixed.
Proof of Isolation Lemma

Lemma
For $\varepsilon \geq 0$, $\Pr[\Delta < \varepsilon] \leq 2n\phi\varepsilon$.

- Say $x_i^* = 1$ and $x_i^{**} = 0$ for some $i$.
- Then $x^* = \arg \max_{x \in S} c^T x \quad$ and $\quad x^{**} = \arg \max_{x \in S} c^T x$.

- Principle of deferred decisions: Fix all $c_j$ for $j \neq i$.
- $\Rightarrow$ Identity of $x^*$ and $x^{**}$ fixed.
- $\Rightarrow$ $\Delta = c^T x^* - c^T x^{**} = \kappa + c_i$ for constant $\kappa$.
Proof of Isolation Lemma

**Lemma**

For \( \varepsilon \geq 0 \), \( \Pr[\Delta < \varepsilon] \leq 2n\phi \varepsilon \).

- Say \( x_i^* = 1 \) and \( x_i^{**} = 0 \) for some \( i \).
- Then \( x^* = \arg \max_{x \in S} c^T x \) and \( x^{**} = \arg \max_{x \in S} c^T x \).

- **Principle of deferred decisions:** Fix all \( c_j \) for \( j \neq i \).
- \( \Rightarrow \) Identity of \( x^* \) and \( x^{**} \) fixed.
- \( \Rightarrow \Delta = c^T x^* - c^T x^{**} = \kappa + c_i \) for constant \( \kappa \)
- \( \Pr[\Delta \in [0, \varepsilon)] = \Pr[c_i \in [\kappa, -\kappa + \varepsilon]] \leq \varepsilon \phi \)
Proof of Isolation Lemma

**Lemma**

For $\epsilon \geq 0$, $\Pr[\Delta < \epsilon] \leq 2n\phi\epsilon$.

- Say $x_i^* = 1$ and $x_i^{**} = 0$ for some $i$.
- Then $x^* = \arg \max_{x \in S} c^T x$ and $x^{**} = \arg \max_{x \in S} c^T x$.
- Principle of deferred decisions: Fix all $c_j$ for $j \neq i$.
- $\Rightarrow$ Identity of $x^*$ and $x^{**}$ fixed.
- $\Rightarrow \Delta = c^T x^* - c^T x^{**} = \kappa + c_i$ for constant $\kappa$.
- $\Pr[\Delta \in [0, \epsilon)] = \Pr[c_i \in [-\kappa, -\kappa + \epsilon)] \leq \epsilon\phi$.
- Union Bound over all $n$ choices for $i$. 

Heiko Röglin | Smoothed Analysis of Algorithms
Extensions

Theorem [Beier, Vöcking (STOC 2004)]

linear binary opt. problem has polynomial smoothed complexity

\[ \iff \]

pseudo-polynomial time \(\text{poly}(n, \max\{|c_i|\})\) in the worst case

[Beier, Vöcking (STOC 2004)]
Theorem remains true if linear constraints are perturbed.

[R., Vöcking (IPCO 2005)]
Theorem remains true for integer optimization problems.
Outline

1. Binary Optimization Problems
   When does a binary optimization problem have polynomial smoothed complexity?

2. Multiobjective Optimization
   How many Pareto-optimal solutions do usually exist?

3. Conclusions
Single-criterion Optimization Problem: \( \min f(x) \) subject to \( x \in S \).

Example: Shortest Path Problem

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>1</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>9</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Real-life logistical problems often involve multiple objectives. (travel time, fare, departure time, etc.)

Multiobjective Opt. Problem: \( \min f_1(x), \ldots, \min f_d(x) \) s.t. \( x \in S \).

Usually, there is no solution that is simultaneously optimal for all \( f_i \).

Question: What can we do algorithmically to support the decision maker?
Optimization Problems

**Single-criterion Optimization Problem:** \[ \min f(x) \text{ subject to } x \in S. \]

Example: 
**Shortest Path Problem**

Real-life logistical problems often involve multiple objectives. (travel time, fare, departure time, etc.)
Single-criterion Optimization Problem: \( \min f(x) \) subject to \( x \in S \).

Example:
Shortest Path Problem

Real-life logistical problems often involve multiple objectives. (travel time, fare, departure time, etc.)

Multiobjective Opt. Problem: \( \min f_1(x), \ldots, \min f_d(x) \) s.t. \( x \in S \).

Usually, there is no solution that is simultaneously optimal for all \( f_i \).

Question
What can we do algorithmically to support the decision maker?
Pareto-optimal Solutions

Multiobjective Opt. Problem: \( \min w^1(x), \ldots, \min w^d(x) \) s.t. \( x \in S \)

\( x \in S \) dominates \( y \in S \) \iff 
\( \forall i: w^i(x) \leq w^i(y) \) and
\( \exists i: w^i(x) < w^i(y) \)

Often the Pareto curve is generated:

\[ \lambda w^1(x) + \ldots + \lambda w^d(x) \]

Tool for solving single-criterion problems

Central Question

How large is the Pareto curve?
Pareto-optimal Solutions

Multiobjective Opt. Problem: \( \min w^1(x), \ldots, \min w^d(x) \) s.t. \( x \in S \)

\( x \in S \) dominates \( y \in S \) \iff \(
\forall i: w^i(x) \leq w^i(y) \) and \\
\exists i: w^i(x) < w^i(y) \\

\( x \in S \) Pareto-optimal \iff \\
\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\:\\
Pareto-optimal Solutions

Multiobjective Opt. Problem: \( \min w^1(x), \ldots, \min w^d(x) \) s.t. \( x \in S \)

\( x \in S \text{ dominates } y \in S \iff \forall i: w^i(x) \leq w^i(y) \) and
\( \exists i: w^i(x) < w^i(y) \)

\( x \in S \text{ Pareto-optimal } \iff \forall y \in S: y \text{ dominates } x \)

Often the Pareto curve is generated:

- Pareto curve limits options for decision maker.
- Monotone functions are optimized by Pareto-optimal solutions, e.g., \( \lambda_1 w^1(x) + \ldots + \lambda_d w^d(x) \) or \( w^1(x) \cdot \ldots \cdot w^d(x) \).
- Tool for solving single-criterion problems
Pareto-optimal Solutions

Multiobjective Opt. Problem: \( \min w^1(x), \ldots, \min w^d(x) \) s.t. \( x \in S \)

\( x \in S \) dominates \( y \in S \) \( \iff \)
\( \forall i : w^i(x) \leq w^i(y) \) and
\( \exists i : w^i(x) < w^i(y) \)

\( x \in S \) Pareto-optimal \( \iff \)
\( \forall y \in S : y \) dominates \( x \)

Often the Pareto curve is generated:

- Pareto curve limits options for decision maker.
- Monotone functions are optimized by Pareto-optimal solutions, e.g., \( \lambda_1 w^1(x) + \ldots + \lambda_d w^d(x) \) or \( w^1(x) \ldots \ldots w^d(x) \).
- Tool for solving single-criterion problems

Central Question

How large is the Pareto curve?
Model

**Linear Binary Optimization Problem**

- set of feasible solutions $S \subseteq \{0, 1\}^n$
  - solution $x = (x_1, \ldots, x_n) \in S$ consists of $n$ binary variables
- $d$ linear objective functions:
  - $\forall i \in \{1, \ldots, d\}$: $\min w^i(x) = w^i_1 x_1 + \cdots + w^i_n x_n$

How large is the Pareto curve?
Exponential in the worst case for almost all problems.
In practice, often few Pareto optimal solutions.
Example: Train Connections
w.r.t. travel time, fare, number of train changes

[Müller-Hannemann, Weihe 2001]
Model

**Linear Binary Optimization Problem**

- set of feasible solutions $S \subseteq \{0, 1\}^n$
  solution $x = (x_1, \ldots, x_n) \in S$ consists of $n$ binary variables
- $d$ linear objective functions:
  $\forall i \in \{1, \ldots, d\}$: $\min w^i(x) = w^i_1 x_1 + \cdots + w^i_n x_n$

**How large is the Pareto curve?**

- Exponential in the worst case for almost all problems.
- In practice, often few Pareto optimal solutions.
  Example: Train Connections
  w.r.t. travel time, fare, number of train changes
  [Müller-Hannemann, Weihe 2001]
Adversary chooses $S$ and a probability density $f_j^i : [-1, 1] \to [0, \phi]$ for every $w_j^i$ and some $\phi \geq 1$. Every $w_j^i$ is drawn independently according to $f_j^i$.

$$P_d(n, \phi) = \max_{S, f_j^i} \mathbb{E} \left[ \text{number of Pareto-optimal sol. for } S \text{ and } f_j^i \right]$$
Adversary chooses $S$ and a probability density $f^i_j : [-1, 1] \rightarrow [0, \phi]$ for every $w^i_j$ and some $\phi \geq 1$. Every $w^i_j$ is drawn independently according to $f^i_j$.

$$P_d(n, \phi) = \max_{S, f^i_j} E \left[ \text{number of Pareto-optimal sol. for } S \text{ and } f^i_j \right]$$

**Bicriteria Optimization ($d = 2$):**

Theorem [Beier, Vöcking (STOC 2003)]

$$P_2(n, \phi) = O(n^4 \phi) \quad P_2(n, \phi) = \Omega(n^2)$$
Adversary chooses $S$ and a probability density $f_j^i : [-1, 1] \rightarrow [0, \phi]$ for every $w_j^i$ and some $\phi \geq 1$. Every $w_j^i$ is drawn independently according to $f_j^i$.

$$P_d(n, \phi) = \max_{S, f_j^i} \mathbb{E}[\text{number of Pareto-optimal sol. for } S \text{ and } f_j^i]$$

### Bicriteria Optimization ($d = 2$):

<table>
<thead>
<tr>
<th>Theorem [Beier, Vöcking (STOC 2003)]</th>
<th>$P_2(n, \phi) = O(n^4 \phi)$</th>
<th>$P_2(n, \phi) = \Omega(n^2)$</th>
</tr>
</thead>
</table>

| Theorem [Beier, R., Vöcking (IPCO 2007)] | $P_2(n, \phi) = O(n^2 \phi)$ | extension to integer optimization problems |
Multiobjective Optimization (d arbitrary constant):

Theorem [R., Teng (FOCS 2009)]

\[ P_d(n, \phi) = O((n\phi)^{h(d)}) \text{ for some function } h \]
Multiobjective Optimization ($d$ arbitrary constant):

Theorem [R., Teng (FOCS 2009)]

$$P_d(n, \phi) = O((n\phi)^{h(d)})$$

for some function $h$

Theorem [Moitra, O’Donnell (STOC 2011)]

$$P_d(n, \phi) = O(n^{2d}\phi^{\Theta(d^2)})$$
## Results (Multiobjective Optimization)

### Multiobjective Optimization ($d$ arbitrary constant):

- **Theorem [R., Teng (FOCS 2009)]**
  \[ P_d(n, \phi) = O((n\phi)^{h(d)}) \] for some function $h$

- **Theorem [Moitra, O’Donnell (STOC 2011)]**
  \[ P_d(n, \phi) = O(n^{2d}\phi^\Theta(d^2)) \]

- **Theorem [Brunschi, R. (TAMC 2011, STOC 2012)]**
  \[ P_d(n, \phi) = O(n^{2d}\phi^d) \]
  \[ P_d(n, \phi) = \Omega(n^{d-1.5}\phi^d) \]
  extension to non-linear objective functions
Beier, R., Vöcking (IPCO 2007)

- \( \min w^1(x) = w_1 x_1 + \cdots + w_n x_n \) and \( \min w^2(x) \)
- subject to \( x \in S \subseteq \{0, 1\}^n \), \( S \) arbitrary
- \( w_j \) drawn according to \( f_j : [0, 1] \rightarrow [0, \phi] \) for \( \phi \geq 1 \)

\[
P_2(n, \phi) = O(n^2 \phi)
\]
Bicriteria Optimization

Beier, R., Vöcking (IPCO 2007)

- \( \min w^1(x) = w_1 x_1 + \cdots + w_n x_n \) and \( \min w^2(x) \)
- subject to \( x \in S \subseteq \{0, 1\}^n \), \( S \) arbitrary
- \( w_j \) drawn according to \( f_j : [0, 1] \rightarrow [0, \phi] \) for \( \phi \geq 1 \)

\[ P_2(n, \phi) = O(n^2 \phi) \]

\[
\mathbb{E}[|P|] = \sum_{i=0}^{k-1} \mathbb{E}\left[ \{ x \in P : w^1(x) \in [t_i, t_{i+1}) \} \right]
\]
Bicriteria Optimization

Beier, R., Vocking (IPCO 2007)

- \( \min w^1(x) = w_1 x_1 + \cdots + w_n x_n \) and \( \min w^2(x) \)
- subject to \( x \in S \subseteq \{0, 1\}^n \), \( S \) arbitrary
- \( w_j \) drawn according to \( f_j : [0, 1] \rightarrow [0, \phi] \) for \( \phi \geq 1 \)

\[
P_2(n, \phi) = O(n^2 \phi)
\]

\[
\begin{align*}
\mathbb{E}[|\mathcal{P}|] & = \sum_{i=0}^{k-1} \mathbb{E}[\{x \in \mathcal{P} : w^1(x) \in [t_i, t_{i+1})\}] \\
& \approx \sum_{i=0}^{k-1} \mathbb{P}[\exists x \in \mathcal{P} : w^1(x) \in [t_i, t_{i+1})]
\end{align*}
\]
Bicriteria Optimization

Beier, R., Vöcking (IPCO 2007)

- \( \min w^1(x) = w_1 x_1 + \cdots + w_n x_n \) and \( \min w^2(x) \)
- subject to \( x \in S \subseteq \{0, 1\}^n \), \( S \) arbitrary
- \( w_j \) drawn according to \( f_j : [0, 1] \rightarrow [0, \phi] \) for \( \phi \geq 1 \)

\[
P_2(n, \phi) = O(n^2 \phi)
\]

\[
E[|\mathcal{P}|]
= \sum_{i=0}^{k-1} E \left[ \{x \in \mathcal{P} : w^1(x) \in [t_i, t_{i+1})\} \right]
= \lim_{k \to \infty} \sum_{i=0}^{k-1} \Pr \left[ \exists x \in \mathcal{P} : w^1(x) \in [t_i, t_{i+1}) \right]
\]
\[ \Pr \left[ \exists x \in \mathcal{P} : w^1(x) \in [t, t + \varepsilon) \right] \]
Loser Gap

\[ \Pr \left[ \exists x \in \mathcal{P} : w^1(x) \in [t, t + \varepsilon) \right] \]

- **single-criterion problem:** \( \min w^2(x) \) s.t. \( w^1(x) \leq t \) and \( x \in S \)
- **winner:** \( x^* = \) optimal solution
Loser Gap

\[ \Pr \left[ \exists x \in P : w^1(x) \in [t, t + \varepsilon) \right] \]

- **single-criterion problem**: \( \min w^2(x) \) s.t. \( w^1(x) \leq t \) and \( x \in S \)
- **winner**: \( x^* = \) optimal solution
- **loser set**: \( \mathcal{L} = \) all solutions \( x \in S \) with \( w^2(x) < w^2(x^*) \)
Loser Gap

\[
\Pr \left[ \exists x \in \mathcal{P} : w^1(x) \in [t, t + \varepsilon) \right]
\]

- **single-criterion problem:** \( \min w^2(x) \) s.t. \( w^1(x) \leq t \) and \( x \in S \)
- **winner:** \( x^* = \) optimal solution
- **loser set:** \( \mathcal{L} = \) all solutions \( x \in S \) with \( w^2(x) < w^2(x^*) \)
- **loser gap:** \( \Lambda(t) = \) distance of loser set \( \mathcal{L} \) from \( t \)
## Loser Gap

\[
\Pr \left[ \exists x \in \mathcal{P} : w^1(x) \in [t, t + \varepsilon) \right]
\]

- **single-criterion problem:** \( \min w^2(x) \) s.t. \( w^1(x) \leq t \) and \( x \in S \)
- **winner:** \( x^* = \) optimal solution
- **loser set:** \( \mathcal{L} = \) all solutions \( x \in S \) with \( w^2(x) < w^2(x^*) \)
- **loser gap:** \( \Lambda(t) = \) distance of loser set \( \mathcal{L} \) from \( t \)

\[
\exists x \in \mathcal{P} : w^1(x) \in [t, t + \varepsilon) \iff \Lambda(t) \leq \varepsilon
\]
Loser Gap

\[ \text{Pr} \left[ \exists x \in \mathcal{P} : w^1(x) \in [t, t + \varepsilon) \right] \]

- **single-criterion problem:** \( \min w^2(x) \) s.t. \( w^1(x) \leq t \) and \( x \in S \)
- **winner:** \( x^* \) = optimal solution
- **loser set:** \( \mathcal{L} = \) all solutions \( x \in S \) with \( w^2(x) < w^2(x^*) \)
- **loser gap:** \( \Lambda(t) = \) distance of loser set \( \mathcal{L} \) from \( t \)

\[ \exists x \in \mathcal{P} : w^1(x) \in [t, t + \varepsilon) \iff \Lambda(t) \leq \varepsilon \]
Loser Gap

\[ \Pr[\exists x \in \mathcal{P} : w^1(x) \in [t, t + \varepsilon)] \]

- single-criterion problem: \( \min w^2(x) \) s.t. \( w^1(x) \leq t \) and \( x \in S \)
- winner: \( x^* \) = optimal solution
- loser set: \( \mathcal{L} = \) all solutions \( x \in S \) with \( w^2(x) < w^2(x^*) \)
- loser gap: \( \Lambda(t) = \) distance of loser set \( \mathcal{L} \) from \( t \)

\[ \exists x \in \mathcal{P} : w^1(x) \in [t, t + \varepsilon) \iff \Lambda(t) \leq \varepsilon \]

Lemma [Beier, Vöcking (STOC 2004)]

For every \( \varepsilon \geq 0 \) and \( t \in \mathbb{R} \), \( \Pr[\Lambda(t) \leq \varepsilon] \leq n\phi\varepsilon \).
Lemma [Beier, Vöcking (STOC 2004)]

For every $\varepsilon \geq 0$ and $t \in \mathbb{R}$, $\Pr[\Lambda(t) \leq \varepsilon] \leq n\phi\varepsilon$. 

$$P_2(n, \phi) \leq \lim_{k \to \infty} \sum_{i=0}^{k-1} \Pr[\exists x \in \mathcal{P} : w^1(x) \in [t_i, t_{i+1})]$$
Lemma [Beier, Vöcking (STOC 2004)]

For every $\varepsilon \geq 0$ and $t \in \mathbb{R}$, $\Pr[\Lambda(t) \leq \varepsilon] \leq n\phi\varepsilon$.

\[ P_2(n, \phi) \leq \lim_{k \to \infty} \sum_{i=0}^{k-1} \Pr[\exists x \in \mathcal{P} : w^1(x) \in [t_i, t_{i+1})] \leq \lim_{k \to \infty} \sum_{i=0}^{k-1} \Pr[\Lambda(t_i) \leq \frac{n}{k}] \]
Lemma [Beier, Vöcking (STOC 2004)]

For every $\varepsilon \geq 0$ and $t \in \mathbb{R}$, $\Pr[\Lambda(t) \leq \varepsilon] \leq n\phi \varepsilon$.

\[
P_2(n, \phi) \leq \lim_{k \to \infty} \sum_{i=0}^{k-1} \Pr[\exists x \in P : w^1(x) \in [t_i, t_{i+1})]
\leq \lim_{k \to \infty} \sum_{i=0}^{k-1} \Pr[\Lambda(t_i) \leq \frac{n}{k}]
\leq \lim_{k \to \infty} \sum_{i=0}^{k-1} \frac{n^2 \phi}{k} = n^2 \phi.
\]

Beier, R., Vöcking (IPCO 2007)

$P_2(n, \phi) = \mathcal{O}(n^2 \phi)$
Outline

1. Binary Optimization Problems
   When does a binary optimization problem have polynomial smoothed complexity?

2. Multiobjective Optimization
   How many Pareto-optimal solutions do usually exist?

3. Conclusions
Conclusions

Summary
Smoothed analysis is a promising framework for a more realistic theory of algorithms. It explains success of simplex algorithm, 2-Opt, and many other algorithms.
Summary
Smoothed analysis is a promising framework for a more realistic theory of algorithms. It explains success of simplex algorithm, 2-Opt, and many other algorithms.

Open Questions
- analyze other pivot rules for simplex method
- improve exponents of smoothed running time for 2-Opt etc.
- analyze your favorite problem/algo that is hard in the worst case
- use insights to develop better algorithms
- explore other frameworks for realistic theory