An Adaptive Hyperbox Algorithm for High-Dimensional Discrete Optimization via Simulation Problems

Presented By LI Zaile

zaileli21@m.fudan.edu.cn
Department of Management Science
Fudan University

Xu, Jie, Barry L. Nelson, and L. Jeff Hong. INFORMS Journal on Computing 25.1 (2013): 133-146.

November 22, 2021

Random Search Algorithms for DOvS

- Nelder-Mead Simplex Modifications (1996, MS)
- Stochastic Ruler (1992, SIAM J. OPTIM.)
- Simulated Annealing (1999, MS)
- Stochastic Comparison (1999, SIAM J. OPTIM.)
- Nested Partition* (2000, OR; 2003, TOMACS)
- COMPASS (2006, OR)
- Framework for LCRS Algorithms (2007, TOMACS)
- Industrial Strength COMPASS (2010, TOMACS)
- AHA (2013, JOC)
- EEE, MRAS, GPS...

Industrial Strength COMPASS (ISC)

ISC solves linearly constrained DOvS problems with a **finite** solution space. The framework consists of three phases:

- The global search phase explores the feasible solution space and identifies promising regions for intensive local search.(a niching genetic algorithm)
- The local search phase investigates these regions and may return multiple locally optimal solutions. (COMPASS with constraint pruning)
- The cleanup phase then selects the best among these local optima and estimates the objective value with controlled error. (Two-stage R&S procedure)

ISC dramatically slows down when dimensionality increases beyond 10. The slowdown of ISC is due to COMPASS' behavior in higher-dimensional spaces.

Outline

Outline

The Problem

$$
\begin{aligned}
&\text{Minimize } g(\mathbf{x}) = \mathrm{E}[G(\mathbf{x})] \\
&\text{Subject to } \mathbf{x} \in \boldsymbol{\Theta} = \Phi \cap \mathscr{L}^D
\end{aligned} \tag{1}
$$

- assume the sample mean of observations of $G(\mathbf{x})$ is a strongly consistent estimator of $g(\mathbf{x}) = \mathrm{E}[G(\mathbf{x})]$.
- Φ is convex and compact, and $\mathscr{L}^D$ denotes the D-dimensional integer lattice. Therefore $\boldsymbol{\Theta}$ is **finite**.

Local Neighborhood/Minimum, LCRS

local neighborhood

Let $\mathcal{N}(\mathbf{x}) = \{\mathbf{y} : \mathbf{y} \in \boldsymbol{\Theta} \text{ and } \|\mathbf{x} - \mathbf{y}\| = 1\}$ be the local neighborhood of $\mathbf{x} \in \boldsymbol{\Theta}$, where $\|\mathbf{x} - \mathbf{y}\|$ denotes the Euclidean distance between $\mathbf{x}$ and $\mathbf{y}$.

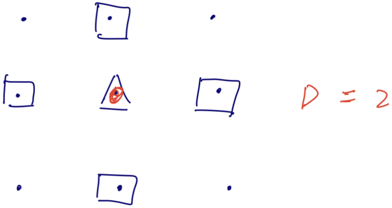
local minimum

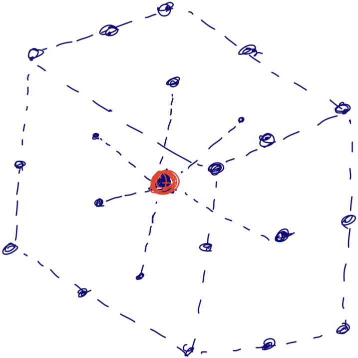
$\mathbf{x}$ is a local minimum if $\mathbf{x} \in \boldsymbol{\Theta}$ and either $\mathcal{N}(\mathbf{x}) = \varnothing$ or $g(\mathbf{x}) \leq g(\mathbf{y})$ for all $\mathbf{y} \in \mathcal{N}(\mathbf{x})$. Let $\mathscr{M}$ denote the set of local minimizers of the function g in $\boldsymbol{\Theta}$.

locally convergent random search (LCRS)

Let $\hat{\mathbf{x}}_k^*$ be the sample best solution at the end of iteration k. An algorithm is a LCRS algorithm if the infinite sequence $\{\hat{\mathbf{x}}_0^*, \hat{\mathbf{x}}_1^*, \ldots\}$ generated by the algorithm converges with probability 1 (w.p.1) to the set $\mathscr{M}$ in the sense that $\Pr\{\hat{\mathbf{x}}_k^* \notin \mathscr{M} \text{ infinitely often (i.o.) }\} = 0$.

$$|\mathcal{N}(\mathbf{x})| \leq 2D$$





The Generic LCRS Algorithm

1. Let $\mathbf{x}_0$ be the starting solution. Set the iteration counter $k = 0$. Let $\mathscr{S}_0 = \mathscr{S}(0) = \{\mathbf{x}_0\}$ and $\hat{\mathbf{x}}_0^* = \mathbf{x}_0$. Set $\mathscr{E}_0 = \{\mathbf{x}_0\}$. Determine $a_0(\mathbf{x}_0)$. Take $a_0(\mathbf{x}_0)$ observations from $\mathbf{x}_0$, set $N_0(\mathbf{x}_0) = a_0(\mathbf{x}_0)$, and calculate $\bar{G}_0(\mathbf{x}_0)$.
2. **Sampling:** Let $k = k + 1$. Determine the MPA $\mathscr{C}_k$ and the sampling distribution F_k on $\mathscr{C}_k$. Sample m_k solutions $\mathbf{x}_{k1}, \mathbf{x}_{k2}, \ldots, \mathbf{x}_{km_k}$ from $\mathscr{C}_k$ using F_k. Remove any duplicates from $\mathbf{x}_{k1}, \mathbf{x}_{k2}, \ldots, \mathbf{x}_{km_k}$, and let $\mathscr{S}_k$ be the remaining set. Let $\mathscr{S}(k) = \mathscr{S}(k-1) \cup \mathscr{S}_k$.
3. **Estimation:** Determine $\mathscr{E}_k \subset \mathscr{S}(k)$ according to the estimation scheme. For all $\mathbf{x} \in \mathscr{E}_k$, take $a_k(\mathbf{x})$ simulation observations. Update $N_k(\mathbf{x})$ and $\bar{G}_k(\mathbf{x})$. For all $\mathbf{x} \notin \mathscr{C}_k$, let $N_k(\mathbf{x}) = N_{k-1}(\mathbf{x})$ and $\bar{G}_k(\mathbf{x}) = \bar{G}_{k-1}(\mathbf{x})$.
4. Let $\hat{\mathbf{x}}_k^* = \arg\min_{\mathbf{x} \in \ell_k} \bar{G}_k(\mathbf{x})$. Go to Step 2.

Review of Notations

- $\mathscr{S}(k)$: the set of all **sampled** solutions through iteration k.
- $\mathscr{S}_k$: the set of unique sampled solutions on iteration k.
- $\mathscr{C}_k \subseteq \mathbf{\Theta}$: the most promising area on iteration k.
- The estimation scheme chooses a subset of solutions $\mathscr{E}_k \subseteq \mathscr{C}(k)$ and allocates $a_k(\mathbf{x})$ additional simulation observations to all $\mathbf{x} \in \mathscr{E}_k$:
  - $\mathscr{E}_k = \mathscr{S}(k)$ in COMPASS
  - $\mathscr{E}_k = \mathscr{S}_k \cup \left\{\hat{\mathbf{x}}_{k-1}^*\right\}$ in AHA
  - In these two cases, all sampled solutions are all estimated.

Analysis of COMPASS as a LCRS

- $\mathscr{C}_1 = \mathbf{\Theta}$.
- $\mathscr{E}_k = \mathscr{S}(k)$ so all sampled solutions are all estimated.
- $\mathscr{C}_k = \{\mathbf{x} : \mathbf{x} \in \mathbf{\Theta}$ and $\left\|\mathbf{x} - \hat{\mathbf{x}}^*_{k-1}\right\| \leq \|\mathbf{x} - \mathbf{y}\|, \forall \mathbf{y} \in \mathscr{S}(k-1)\}$ for $k > 1$.
- uses a uniform distribution defined on $\mathscr{C}_k$ as the sampling distribution F_k.
- $m_k = m$.

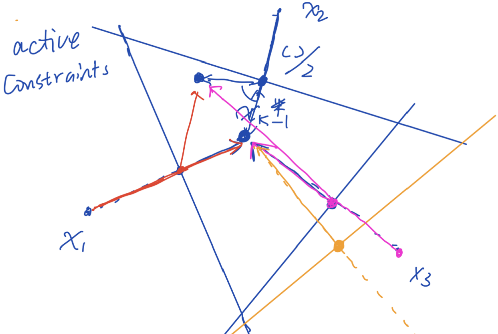
When $\mathbf{\Theta}$ is finite, COMPASS is locally convergent as long as each sample mean $\bar{G}_k(\mathbf{x})$ satisfies a strong law of large numbers and the estimation scheme guarantees that $N_k(\mathbf{x})$ goes to infinity as $k \to \infty$ for all $\mathbf{x} \in \mathscr{S}(k)$.

Constraint Pruning to Speed Up Sampling

The constraints defining the COMPASS MPA have the form

$$\left(\hat{\mathbf{x}}_{k-1}^{*} - \mathbf{x}_i\right)' \left(\mathbf{x} - \frac{\hat{\mathbf{x}}_{k-1}^{*} + \mathbf{x}_i}{2}\right) \geq 0, \quad \mathbf{x}_i \in \mathscr{S}(k-1) \qquad (2)$$

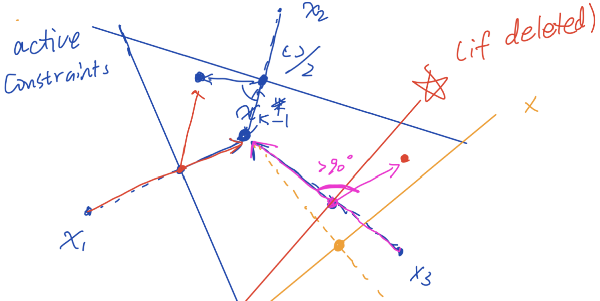
Not all constraints are required to define $\mathscr{C}_k$. Hong and Nelson (2007) noted that to guarantee local convergence, it is sufficient to drive $N_k(\mathbf{x})$ to infinity only for those solutions $\mathbf{x}_i \in \mathscr{S}(k-1)$ that yield active constraints defining the MPA $\mathscr{C}_k$.



To determine whether solution $\mathbf{x}_i \in \mathscr{S}(k-1)$ defines an active constraint, Xu et al. (2010) showed that one can solve the following LP:

$$\begin{aligned} \min_{\mathbf{x}} & \left(\hat{\mathbf{x}}_{k-1}^* - \mathbf{x}_i\right)' \left(\mathbf{x} - \frac{\hat{\mathbf{x}}_{k-1}^* + \mathbf{x}_i}{2}\right) \\ \text{s.t.} & \left(\hat{\mathbf{x}}_{k-1}^* - \mathbf{x}_j\right)' \left(\mathbf{x} - \frac{\hat{\mathbf{x}}_{k-1}^* + \mathbf{x}_j}{2}\right) \geq 0 \\ & \forall \mathbf{x}_j \in \mathscr{S}(k-1) \backslash \left\{\hat{\mathbf{x}}_{k-1}^*\right\}, j \neq i \end{aligned} \tag{3}$$

The solution $\mathbf{x}_i$ defines an active constraint if and only if the objective function value is negative.



COMPASS' behavior in high-dimensional spaces

- COMPASS closes in on a locally optimal solution by **progressively adding linear constraints (to shrink the MPA)** that define the most promising area for exploration.
- As dimension increases, the number of constraints that COMPASS needs to define the most promising area with visited solutions quickly increases, and **sampling new solutions from the most promising area is time-consuming.**
- COMPASS can employ constraint pruning. Constraint pruning involves solving linear programs (LPs), and it is needed much more frequently in high-dimensional problems. it is essential to keep COMPASS from slowing down in problems where it visits many solutions.
- *The geometry of the COMPASS MPA is also an impediment to solve large-dimension problems.* (explained later)

Outline

AHA as a LCRS

1. Let $\mathbf{x}_0$ be the starting solution. Set the iteration counter $k=0$. Let $\mathscr{S}_0 = \mathscr{S}(0) = \{\mathbf{x}_0\}$ and $\hat{\mathbf{x}}_0^* = \mathbf{x}_0$. Set $\mathscr{E}_0 = \{\mathbf{x}_0\}$. Determine $a_0(\mathbf{x}_0)$. Take $a_0(\mathbf{x}_0)$ observations from $\mathbf{x}_0$, set $N_0(\mathbf{x}_0) = a_0(\mathbf{x}_0)$, and calculate $\bar{G}_0(\mathbf{x}_0)$.
2. **Sampling:** Let $k=k+1$. **Identify** $\mathcal{U}_k$ **and** $\mathscr{L}_k$ **and thus** $\mathscr{H}_k$ (**for** $k=1, \mathcal{U}_k = \varnothing, \mathscr{L}_k = \varnothing$**, and** $\mathscr{C}_k = \Theta$). **Let** $\mathscr{C}_k = \mathscr{H}_k \cap \Theta$**.** Sample m_k solutions $\mathbf{x}_{k1}, \mathbf{x}_{k2}, \ldots, \mathbf{x}_{km}$ from $\mathscr{C}_k$ using F_k **(Uniform Sampling)**. Remove any duplicates from $\mathbf{x}_{k1}, \mathbf{x}_{k2}, \ldots, \mathbf{x}_{km}$, and let $\mathscr{S}_k$ be the remaining set. Let $\mathscr{S}(k) = \mathscr{S}(k-1) \cup \mathscr{S}_k$. **Notice that** $m_k = m$**.**
3. **Estimation: Let** $\mathscr{E}_k = \mathscr{S}_k \cup \left\{\hat{\mathbf{x}}_{k-1}^*\right\}$**.** For all $\mathbf{x} \in \mathscr{E}_k$, take $a_k(\mathbf{x})$ simulation observations. Update $N_k(\mathbf{x})$ and $\bar{G}_k(\mathbf{x})$. For all $\mathbf{x} \notin \mathscr{C}_k$, let $N_k(\mathbf{x}) = N_{k-1}(\mathbf{x})$ and $\bar{G}_k(\mathbf{x}) = \bar{G}_{k-1}(\mathbf{x})$.
4. Let $\hat{\mathbf{x}}_k^* = \arg\min_{\mathbf{x} \in \ell_k} \bar{G}_k(\mathbf{x})$. Go to Step 2.

Illustration of Hyperbox

For a visited solution $\mathbf{x}$, let $x^{(d)}$ be its d th coordinate, $1 \le d \le D$. Let $l_k^{(d)} = \max_{\mathbf{x} \in \mathscr{S}(k), \mathbf{x} \neq \hat{\mathbf{x}}_k^*} \left\{ x^{(d)} : x^{(d)} < \hat{x}^{*(d)} \right\}$ if it exists; otherwise, let $l_k^{(d)} = -\infty$. Similarly, let $u_k^{(d)} = \min_{\mathbf{x} \in \mathscr{S}(k), \mathbf{x} \neq \hat{\mathbf{x}}_k^*} \left\{ x^{(d)} : x^{(d)} > \hat{x}_k^{*(d)} \right\}$ if it exists; otherwise, let $u_k^{(d)} = \infty$. Let $\mathscr{L}_k = \left(l_k^{(1)}, \ldots, l_k^{(D)} \right)$ and $\mathscr{U}_k = \left(u_k^{(1)}, \ldots, u_k^{(D)} \right)$.

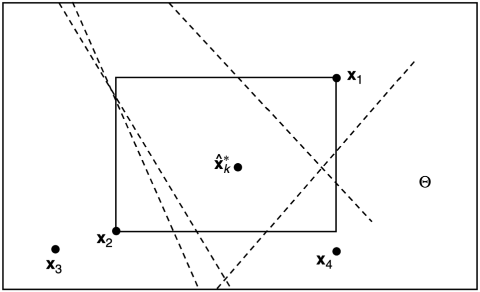


Figure 1 COMPASS MPA vs. AHA MPA: A Two-Dimensional Example

- The volume of the COMPASS MPA may be much larger than that of the AHA MPA for the same set of visited solutions, especially when D is large.
- It is much easier to identify $\mathscr{H}_k$ than to identify the set of active solutions for the COMPASS.
- **AHA is very aggressive in closing toward locally optimal solutions.** (closing = shrinking)
- Hyper-box preserving: in each iteration, the MPA constructed is a hyper-box.

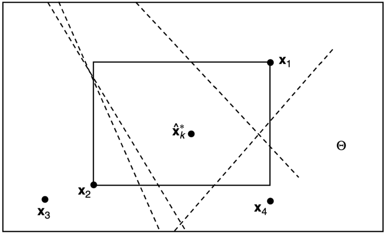


Figure 1 **COMPASS MPA vs. AHA MPA: A Two-Dimensional Example**

Convergence of the Generic LCRS

Assumption 1.

For all $\mathbf{x} \in \Theta$,

$$\lim_{r\to\infty} \frac{1}{r} \sum_{i=1}^{r} G_i(\mathbf{x}) = g(\mathbf{x}) \quad \text{w.p. } 1$$

Condition 1.

The sampling distribution F_k guarantees that $\Pr\{\mathbf{x} \in \mathscr{S}_k\} \geq \epsilon$ for all $\mathbf{x} \in \mathcal{N}\left(\hat{\mathbf{x}}^*_{k-1}\right)$ for some $\epsilon > 0$ that is independent of k.

Condition 2.

The estimation scheme satisfies the following requirements:

- $\mathscr{E}_k$ is a subset of $\mathscr{S}(k)$, $\mathscr{E}_k$ contains $\hat{\mathbf{x}}^*_{k-1}$ and $\mathscr{S}_k$; and
- $a_k(\mathbf{x})$ is allocated such that $\min_{\mathbf{x}\in\mathscr{E}_k} N_k(\mathbf{x}) \geq 1$ for all $k = 1, 2, \ldots$, and $\min_{\mathbf{x}\in\mathscr{E}_k} N_k(\mathbf{x}) \to \infty$ w.p. 1 as $k \to \infty$. (e.g., $a_k(\mathbf{x}) = \min\left\{5, \left\lceil 5(\log k)^{1.01} \right\rceil\right\} - N_{k-1}(\mathbf{x})$)

Proposition 1.

Let $\hat{\mathbf{x}}_{k'}^{*} k = 0, 1, 2, \ldots$ be a sequence of solutions generated by Algorithm 1 when applied to problem (1). Suppose that Assumption 1 is satisfied. If Conditions 1 and 2 hold, then $\Pr\{\hat{\mathbf{x}}_k^* \notin \mathscr{M} \text{ i.o. }\} = 0$.

Proof of this Proposition will be introduced later if time is allowed.

Convergence of AHA

Proposition 2.

AHA is an instance of the general LCRS algorithm when solutions are uniformly randomly sampled within the MPA $\mathscr{C}_k$ at each iteration k.

PROOF. To verify Condition 1 , we need to compute $\Pr\{\mathbf{x} \in \mathscr{S}_k\}$ for all $\mathbf{x} \in \mathcal{N}\left(\hat{\mathbf{x}}^*_{k-1}\right)$. Notice that $\mathcal{N}\left(\hat{\mathbf{x}}^*_{k-1}\right) \subseteq \mathscr{H}_k \cap \Theta = \mathscr{C}_{k-1}$ by construction. Denote the m solutions independently and uniformly sampled within $\mathscr{C}_{k-1}$ as $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m$. For all $\mathbf{x} \in \mathcal{N}\left(\hat{\mathbf{x}}^*_{k-1}\right)$ we have $\Pr\{\mathbf{x} \in \mathscr{S}_k\} = 1 - \Pr\{\mathbf{x} \notin \mathscr{S}_k\} = 1 - \Pr\{\mathbf{x}_1 \neq \mathbf{x}, \ldots, \mathbf{x}_m \neq \mathbf{x}\} = 1 - \Pr\{\mathbf{x}_1 \neq \mathbf{x}\}^m$. Then we have $\Pr\{\mathbf{x}_1 \neq \mathbf{x}\} = \left(|\mathscr{C}_{k-1}| - 1\right) / |\mathscr{C}_{k-1}|$. So

$$\Pr\{\mathbf{x} \in \mathscr{S}_k\} = 1 - \left(1 - \frac{1}{|\mathscr{C}_{k-1}|}\right)^m \geq 1 - \left(1 - \frac{1}{|\Theta|}\right)^m > 0$$

Thus Condition 1 is satisfied.

$$\mathscr{E}_k = \mathscr{S}_k \cup \left\{ \hat{\mathbf{x}}^*_{k-1} \right\}$$

Next, we check Condition 2. It is clear that $\mathscr{E}_k$ is a subset of $\mathscr{S}(k)$. So the first requirement is satisfied. By construction, AHA also satisfies the second part of Condition 2. The third requirement is on the sample allocation schedule, and thus we can use (e.g., $a_k(\mathbf{x}) = \min\left\{5, \left\lceil 5(\log k)^{1.01} \right\rceil\right\} - N_{k-1}(\mathbf{x})$), which satisfies this requirement. Therefore, Condition 2 is also satisfied.

Implementation of MPA

A straightforward implementation is to use a data structure to **record the positions of all visited solutions for each coordinate**, and then for each $\hat{\mathbf{x}}_k^*$, search the entire map for $u_k^{(d)}$ and $l_k^{(d)}$. The algorithmic complexity is $O(|\mathscr{S}(k)|\log(|\mathscr{S}(k)|))$ (with a special data structure in C++).

Compared with the constraint pruning in the original COMPASS algorithm, this overhead is also quite small.

The Local Optimality Stopping Test for AHA

AHA is asymptotically convergent; a finite-time stopping test is required.

An approach is to compare the sample best solution and all of its neighbors by taking i.i.d. observations of those solutions.

Xu et al. (2010) developed one such test:

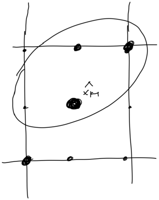
$$H_0 : g\left(\hat{\mathbf{x}}_k^*\right) \le \min_{\mathbf{y} \in \mathcal{N}\left(\hat{\mathbf{x}}_k^*\right)} g(\mathbf{y}) \quad \text{versus} \quad H_1 : g\left(\hat{\mathbf{x}}_k^*\right) > \min_{\mathbf{y} \in \mathcal{N}\left(\hat{\mathbf{x}}_k^*\right)} g(\mathbf{y}).$$

There are two outcomes of the test:

- $\hat{\mathbf{x}}_k^*$ passes the test and is accepted.
- some other solution in $\mathcal{N}\left(\hat{\mathbf{x}}_k^*\right)$ is returned as the current sample best solution and the search continues.

Timing of The Local Optimality Stopping Test

The construction of $\mathscr{C}_k$ means that AHA will always have more than one solution inside the MPA.



There are two obvious options for the local optimality test:

- OPTION 1. once $\hat{\mathbf{x}}_k^*$ is the only interior solution of the MPA $\mathscr{C}_k$, the algorithm hands $\hat{\mathbf{x}}_k^*$ and all of its neighbors (some of which may not have been visited yet) to the stopping test procedure.

However, this may lead to too many premature tests and consume a lot of simulation replications unnecessarily, since **AHA is very aggressive in closing toward locally optimal solutions.**

Timing of The Local Optimality Stopping Test

- OPTION 2.Once $\hat{\mathbf{x}}_k^*$ is the only interior solution of the MPA $\mathscr{C}_k$, and all of $\hat{\mathbf{x}}_k^{*\prime}$ s neighbors have been visited, the algorithm hands $\hat{\mathbf{x}}_k^*$ and all of its neighbors to the stopping test procedure.

However, waiting for all neighbors to be visited by uniform random sampling can take a long time.

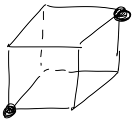
Strengthening Neighborhood Sampling for AHA

- Step 3.1. Check if $\hat{\mathbf{x}}_k^*$ is the only interior solution in $\mathscr{C}_k$. If not, continue with Step 2 in AHA.
- Step 3.2. Let $k = k + 1$. For all $\mathbf{x} \in \mathcal{N}\left(\hat{\mathbf{x}}_{k-1}^*\right)$, check if $\mathbf{x} \in \mathscr{S}(k-1)$. Let $\mathscr{D}_k \subseteq \{1, 2, \ldots, D\}$ be the set of coordinate directions along which there are unvisited neighbors. If $\mathscr{D}_k = \varnothing$, invoke stopping test.
- Step 3.3. Randomly pick a coordinate direction from $\mathscr{D}_k$ and sample one neighbor along that direction. Repeat the process $m-2$ times to generate $\mathbf{x}_{k1}, \mathbf{x}_{k2}, \ldots, \mathbf{x}_{k(m-1)}$ (**Neighborhood Sampling**). Sample randomly within $\mathscr{C}_{k-1}$ to generate $\mathbf{x}_{km}$ **(for convergence)**. Remove any duplicates from $\mathbf{x}_{k1}, \mathbf{x}_{k2}, \ldots, \mathbf{x}_{km}$, and let $\mathscr{S}_k$ be the remaining set. Let $\mathscr{S}(k) = \mathscr{S}(k-1) \cup \mathscr{S}_k$ and $\mathscr{C}_k = \mathscr{C}_{k-1}$. Continue with Step 3 in AHA.

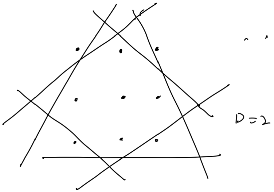
Outline

Analysis of MPA

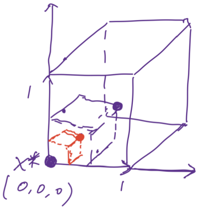
- Intuitively, AHA should scale well in high-dimensional problems because at most $2D$ solutions and **as few as two solutions** are needed to construct a hyperbox containing $\mathbf{x}_k^*$.



- COMPASS needs to have more and more solutions to define an enclosing MPA containing $\mathbf{x}_k^*$ as dimension increases.



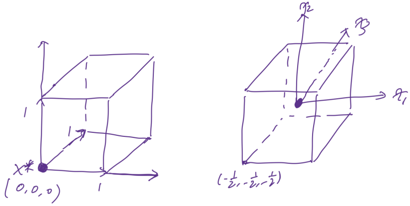
- Beyond the asymptotic analysis, it is possible to analyze one aspect of the algorithm's finite-time behavior: **how it cuts down/shrink the MPA for a deterministic problem.**



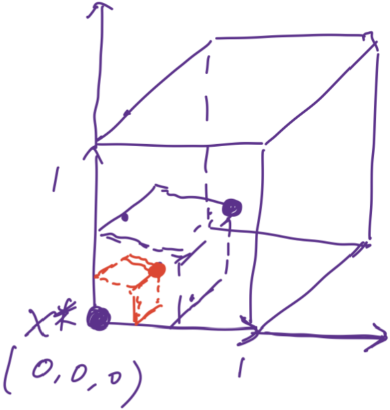
- The effectiveness of a locally convergent DOvS algorithm depends on its ability to focus in on a locally optimal solution (to some extent). (Shrinking too quickly could also be inefficient.)
- **Dimensionality:** An algorithm whose ability to do so degrades as dimension increases in the best case (deterministic output) will certainly struggle in a stochastic problem.

Modeling the MPA Volume Reduction

- The feasible region is a hyperbox of volume 1 in a D-dimension solution space.
- A unique locally optimal solution $\mathbf{x}^* = (0, 0, \ldots, 0)^T$.
- The Corner Case (the most favorable): $\mathbf{\Theta}_1 = [0, 1]^D$
- The Center Case (the least favorable): $\mathbf{\Theta}_2 = [-1/2, 1/2]^D$



- We assume that $\mathbf{x}^*$ is the initial solution. In one iteration, we sample m solutions $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m$ uniformly from $\mathbf{\Theta}_1$ or $\mathbf{\Theta}_2$. We then construct the MPA with these solutions using AHA or COMPASS.
- The measure is EV, where V is the volume of MPA in each case. The smaller EV is, the more efficient the algorithm is.


1
1
(0,0,0)

The AHA MPA

Single Iteration for the Corner Case

The expected volume of the MPA constructed according to AHA using m solutions $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m$ uniformly randomly sampled within $\mathbf{\Theta}_1$ is

$$\mathrm{E}(V) = \left(\frac{1}{m+1}\right)^D \tag{4}$$

Single Iteration for the Center Case

$$\mathrm{E}(V) = \left\{\frac{2}{m+1}\left[1 - \left(\frac{1}{2}\right)^{m+1}\right]\right\}^D \approx \left(\frac{2}{m+1}\right)^D \tag{5}$$

For arbitrary $\mathbf{x}^*$, AHA reduces the volume of the MPA at a rate of $(C/(m+1))^D$, where C is a constant between 1 and 2 and varies from iteration to iteration.

Suppose we have a fixed budget of M simulation replications, and it takes one replication to evaluate each sampled solution. If we sample m solutions at each iteration, then we will use up the simulation budget at iteration $k \approx M/m$.

Fixed Budget Multiple Iterations

The expected volume is

$$\mathrm{E}\left(V_k\right) \sim (C/(m+1))^{DM/m}.$$

$\mathrm{E}\left(V_k\right)$ is minimized by letting $m = 2.8$ (when $C = 1$) or $m = 5.5$ (when $C = 2$). Therefore it is reasonable to sample three to six solutions at each iteration.

COMPASS MPA

Unlike AHA, the geometry of COMPASS is much more complicated, and there are no simple closed-form expressions for $\mathrm{E}(V)$. However, we are able to derive asymptotic lower bounds when D is large.

Single Iteration for the Corner Case

$$\mathrm{E}(V) \geq \Phi(-0.49\sqrt{D})^m \tag{6}$$

Single Iteration for the Center Case

$$\mathrm{E}(V) \geq \Phi(0.46\sqrt{D})^m \tag{7}$$

The MPA formed by COMPASS is no longer a hyperbox, and we can not extend the analysis to multiple iterations.

Comparison Result

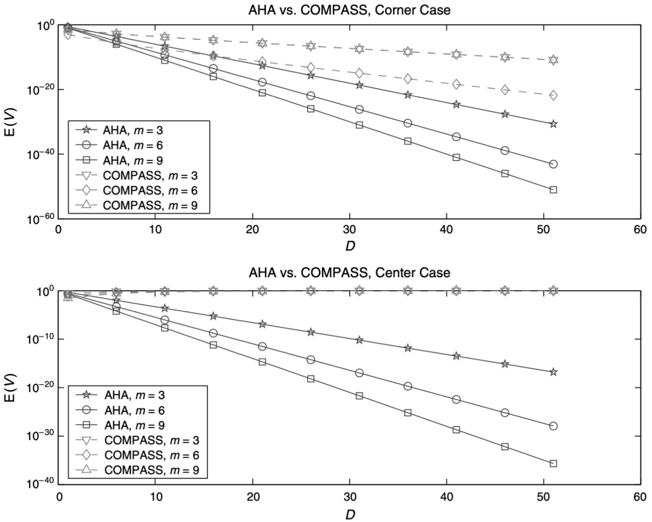


Figure 2 E(V) as a Function of D for Different m for AHA and COMPASS

Comparison Result

- AHA's ability to shrink the MPA is more robust than COMPASS with respect to the location of $\mathbf{x}^*$.
- COMPASS may be more efficient at shrinking the MPA than AHA when D is small and m is large for the Corner Case, however, as D increases, the efficiency of AHA quickly catches up and surpass COMPASS.
- AHA's ability to shrink the volume of the MPA keeps up with the exponential increase in the number of feasible solutions as D increases.
- For the Center Case, as D increases, $\mathrm{E}(V)$ actually increases, which helps to explain why COMPASS slows down dramatically as the dimension of $\boldsymbol{\Theta}$ increases.

Outline

Test Problems

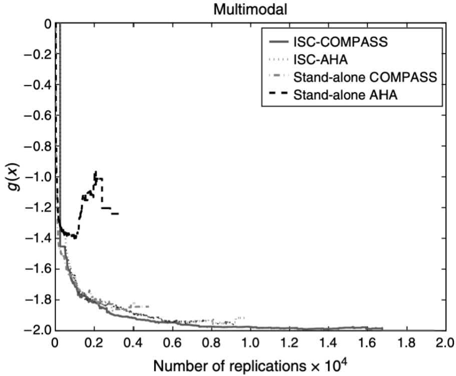


Figure 3 Performance Plot for the Multimodal Function

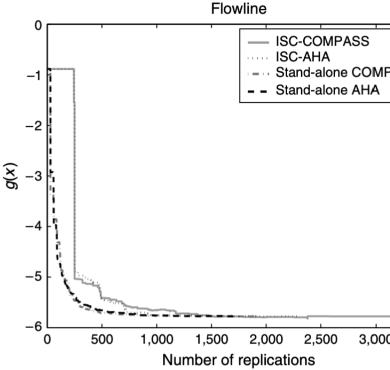


Figure 5 Performance Plot for the Flowline Problem

High-Dimensional Problem with a Single Local Minimum

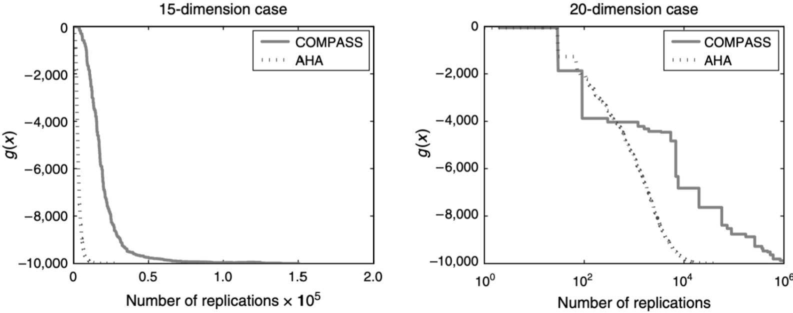


Figure 6 **Performance Plot for the High-Dimensional Test Problem:** $D = 5, 10, 15$, **and** 20

High-Dimensional Problem with Many Local Minimums

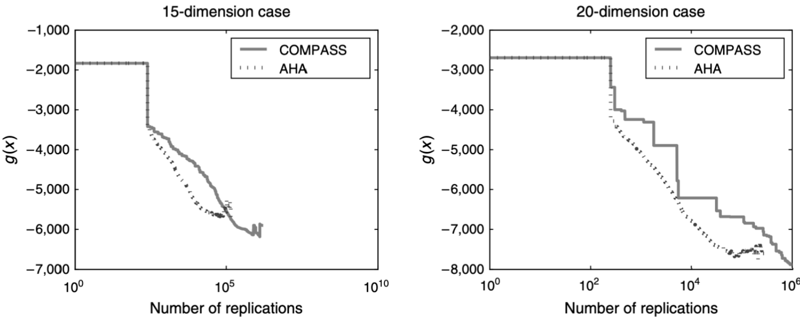


Figure 7 Performance Plot for the High-Dimensional Multimodal Test Problem: $D = 5, 10, 15,$ **and** 20

Observations

Table 2 **The Average CPU Time (Seconds) for High-Dimensional Test Problems**

Variable	$D=5$	$D=10$	$D=15$	$D=20$	$D=50$	$D=100$
COMPASS	12.5600	432	2,716.6	289,122	NA	NA
AHA	0.5856	3.42	8.541	17.3556	463.12	6,456.94

- AHA achieves performance comparable to that of COMPASS for low-dimensional problems ($D \leq 10$) and is much more efficient in solving high-dimensional problems ($D > 10$).
- AHA converges to locally optimal solutions very quickly and therefore risks being trapped in inferior ones in the presence of multiple locally optimal solutions.
- However, the global phase of the ISC software largely alleviates this problem.

Outline

Conclusion

- AHA has performance similar to or slightly inferior to that of COMPASS when dimension is low, say, less than 10. For high-dimensional problems, AHA outperforms COMPASS by a significant margin.
- It is of vital importance to maintain a proper balance between premature local optimality testing and the time it takes AHA to sample all neighbors of the current sample best solution.
- It adopts a hyperbox-shaped MPA geometry but still uses the same uniform random sampling as COMPASS does to sample solutions inside the MPA.

Comments

- AHA is simpler to implement than COMPASS.
- ISC is required for a good performance.
- Stopping test may be time consuming.
- The sampling distribution of solutions may have significant impact on algorithms' performance.
- Next step: EEE, GPS, BO...