Stochastic Comparison Algorithm For Discrete Optimization With Estimation

Ouyang Wanlu

February 18, 2022

The optimization problem

The optimization problem is to find a configuration, i (not necessarily unique), from a discrete finite set of alternatives, S, that minimizes an objective function, g(i), i.e.,

$$\min_{i \in S}\{g(i)\}$$

with $g : S \to R$ and $S = \{1, 2, ..., s\}$. That is, we wish to find a global optimal configuration $i \in S$, where S is the global optimal set, given by

$$S^* = \{i \in S | g(i) \leq g(j) \, \forall j \in S\}$$

Assumption

- $|S|$ is very large.Let us denote the cardinality of the solution space S by $|S|$ (note that $|S| = s$)
- we do not have an analytic expression for the objective function $g(i)$ and that it can only be evaluated via Monte Carlo simulation.

Let $H(i)$ be a sample estimate of $g(i)$

- $g(i) = E[H(i)], \forall i \in S(i.e., H(i)$ is unbiased)
- variance of the estimate is finite, i.e.,$E[H(i) - E[H(i)]]^2 < \infty, \forall i \in S$.

standard definitions and assumption

Definition 2.1

For each $i \in S$, there exists a subset $N(i)$ of $S\{i\}$, which is called the set of neighbors of i.

Definition 2.2

A function $R : S \times S \to [0, 1]$ is said to be a generating probability for S and N if

- $R(i, j) > 0 \Leftrightarrow j \in N(i)$
- $\sum_{j \in S} R(i, j) = 1$ for $i, j \in S$

standard definitions and assumption

Assumption 1

For any pair $(i, j) \in S \times S$, j is reachable from i, i.e., there exists a finite sequence $\{n_m\}_{m=0}^{l}$ for some l, such that $i_{n_0} = i, i_{n_l} = j$, and $i_{n_{m+1}} \in N(i_{n_m})$ for $m = 0, 1, 2, ..., l - 1$.

convergence of SA and SR

- For SA(Simulated Annealing), it has been proved that a real sequence$\{T_k\}_{k=0}^{\infty}$ satisfying $T_k = \frac{\gamma}{log(k+k_0+1)}$, $k = 0, 1, 2, ...,$ for some positive numbers γ and k_0 will guarantee that the algorithm will converge to a global optimum. T_k is called the temperature at the kth iteration of the sequence and $\{T_k\}_{k=0}^{\infty}$ is called the cooling schedule.
- For SR(Stochastic Ruler), it has been proved that an integer sequence $\{M_k\}_{k=0}^{\infty}$ satisfying $M_k = \lfloor clog_{\sigma}(k + k_0 + 1)\rfloor, k = 0, 1, 2, ...(\lfloor\xi\rfloor$ denotes the greatest integer that is smaller than or equal to ξ), for some positive numbers $c, \sigma, andk_0$ will guarantee that the algorithm will converge to a global optimum. We call M_k the kth testing number and $\{M_k\}_{k=0}^{\infty}$ the testing sequence.

standard way to prove convergence

if we let X_k denote the configuration visited by the algorithm at the kth iteration, then $\{X_k\}_{k=1}^{\infty}$ is a Markov chain. Then, to prove the convergence of the algorithm, all one has to do is to show that the probability vector $e(k) = [e_1(k)...e_s(k)]$ with $e_i(k) \doteq Pr\{X_k = i\}$ for $i = 1, ..., s$ converges to an optimal probability vector $e = [e_1...e_s]$, i.e., that

$$e_i^* > 0 \; for \; i \in S^*$$

$$e_i^* = 0 \; for \; i \notin S^*$$

one-step transition probabilities of SA and SR

For SA, the one-step transition probabilities of the Markov chain X_k for a given temperature T are

$$P_{ij}(T) = \begin{cases} R(i,j)\min\left[1, e^{-\{g(j)-g(i)\}/T}\right] & \text{if} \quad j \in N(i) \\ 1 - \sum_{n\in N(i)} P_{in}(T) & \text{if } j = i \\ 0 & \text{otherwise} \end{cases}$$

For SR, the one-step transition probabilities of the Markov chain $\{X_k\}$ for a given testing number M are

$$P_{ij}(M) = \begin{cases} R(i,j)\{P(j,a,b)\}^M & \text{if } j \in N(i) \\ 1 - \sum_{n\in N(i)} P_{in}(M) & \text{if } j = i \\ 0 & \text{otherwise} \end{cases}$$

where $P(i,a,b) = P[H(i) \leq \theta(a,b)]$, $P_{ij}(M)$ is the probability that the search goes from configuration i to con- figuration j when the testing number is M.

Limitation of SA and SR

- SA does not converge when the objective function estimates are noisy.SR is more robust with respect to estimation error.
- The SA algorithm must often visit poor configurations so as not to overlook the possibility that there might be a very good configuration surrounded by poor configurations.
- For the SR algorithm, one has to choose the size of the stochastic ruler, which can also be difficult in practice.

motivation

- SC algorithm differs from the SR algorithm in that, instead of comparing candidate configurations to a stochastic ruler, it directly compares the current configuration to a candidate configuration.
- The SC algorithm, therefore, does not require any knowledge whatsoever about the structure of the search space.
- This does mean, however, that convergence is only guaranteed when any configuration (in the whole configuration space) can be reached from any other in one step. In other words, we have eliminated the neighborhood structure for the sake of convergence.

An alternative problem

Define

$$sp(i) = \sum_{j \in S/\{i\}} Pr[H(i) < H(j)]$$

The SC algorithm, therefore, seeks to identify a member of the optimum set $\overline{S}^*$, where $\overline{S}^* = \{i \in S | sp(i) \geq sp(j) \, \forall j \in S\}$. Let $W_i = H(i) - g(i)$ denote the estimation error, and assume that it satisfies the following conditions.

Assumption 3.1

1. $W_i, i \in S$ are i.i.d.
2. Each $W_i, i \in S$, has a symmetric continuous probability density function with a zero mean.

Theorem 3.1

Under Assumption 3.1,

$$E[H(i)] < E[H(j)] \Leftrightarrow sp(i) > sp(j) \forall i \neq j, i, j \in S$$

.

Corollary 3.1

Given $H(i), H(j) \forall i \neq j, i, j \in S$, we have

$$E[H(i)] < E[H(j)] \Leftrightarrow Pr[H(i) < H(j)] > Pr[H(j) < H(i)]$$

.

Lemma 3.1

Given $H(i), H(j), H(k)\,\forall i \neq j, j \neq k, k \neq i; i, j, k \in S$, and Assumption 3.1, the following two conditions are equivalent:

1. $E[H(i)] < E[H(j)] < E[H(k)]$;
2. $Pr[H(i) < H(k)] > Pr[H(j) < H(k)]$ and $Pr[H(i) < H(k)] > Pr[H(i) < H(j)]$.

Assumption 3.2

$R(i,j) > 0\,\forall i,j \in S$ and $i \neq j$

SC algorithm

The stochastic comparison algorithm.

Data: R, $\{M_k\}$, $i_0 \in S$.

Step 0: Set $X_0 = i_0$ and $k = 0$.

Step 1: Given $X_k = i$, choose a candidate Z_k from $S \setminus \{i\}$ with probability

$$P[Z_k = j | X_k = i] = R(i,j), \qquad j \in S \setminus \{i\}.$$

Step 2: Given $Z_k = j$, set

$$X_{k+1} = \begin{cases} Z_k & \text{if} \quad H_\ell(j) < H_\ell(i) \; \forall \ell = 1, \ldots, M_k, \\ X_k & \text{otherwise.} \end{cases}$$

Step 3: Set $k = k + 1$ and go to Step 1.

Figure 1: SC algorithm

For convergence, the testing sequence used by the SC algorithm must satisfy the same conditions as those required by the SR algorithm; i.e., M_k must be such that $M_k = clog_\sigma(k + k_0 + 1), k = 0, 1, 2, ...,$for some positive numbers c, σ, and k_0.

Due to the i.i.d. assumption of $\{H_l(i), l = 1, ..., M_k; i \in S\}$, the state transition probability from i to j is

$$R(i,j)Pr[H_1(j) < H_1(i), ..., H_{M_k}(j) < H_{M_k}(i)]$$

$$= R(i,j)Pr[H(j) < H(i)]^{M_k}$$

Thus, the sequence of configurations visited by the SC algorithm forms a time-inhomogeneous Markov chain $\{X_k\}$.

An outline of our analysis is as follows.
1. Set $M_k = M$ and study the corresponding Markov chain at its steady state (the steady-state probability distribution is denoted by $\pi(M)$).
2. Let M go to infinity and show that
(a) $\pi(M)$ converges to an optimal probability vector; and
(b) for large M, $\pi(M)$ is monotonic in M.
3. Show that the Markov chain with $M_k = M$ is weakly ergodic by calculating the coefficient of ergodicity.
4. Show that the Markov chain with $M_k = M$ is strongly ergodic.
5. Show the convergence of the Markov chain $\{X_k\}$ based on its strong ergodicity.

Markov chain equations

The one-step state transition probabilities of the Markov chain $\{X_k\}$ generated by the SC algorithm for a given testing number M are

$$P_{ij}(M) = \begin{cases} R(i,j)\{\Pr[H(j) < H(i)]\}^M & \text{if} \quad j \neq i \\ 1 - \sum_{n=1,n\neq i}^{s} P_{in}(M) & \text{if} \quad j = i \end{cases}$$

To simplify notation, we will let $r_{ij} = R(i,j)$, $p_{ij} = Pr[H(j) < H(i)]$, and $t_{ij} = r_{ij} p_{ij}^M (i \neq j)$, where $s = |S|$ represents the size of the configuration space. Using our shorthand notation, we can write the one-step transition probabilities as

$$P(M) = \begin{bmatrix} 1 - \sum_{n=2}^{s} t_{1n} & t_{12} & t_{13} & \cdots \\ t_{21} & 1 - \sum_{n=1,n\neq 2}^{s} t_{2n} & t_{23} & \cdots \\ t_{31} & t_{32} & 1 - \sum_{n=1,n\neq 3}^{t} t_{3n} & \cdots \\ \vdots & \vdots & \vdots & \ddots \\ t_{s1} & t_{s2} & t_{s3} & \cdots & 1 \end{bmatrix}$$

$\pi(M) = \pi(M)P(M)$ and $\sum_{i \in S} \pi_i(M) = 1 \Leftrightarrow A\pi^T(M) = b$ where

$$A = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ t_{12} & -\sum_{n=1,n\neq 2}^{s} t_{2n} & t_{32} & \cdots & t_{s2} \\ t_{13} & t_{23} & -\sum_{n=1,n\neq 3}^{s} t_{3n} & \cdots & t_{s3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{1s} & t_{2s} & t_{3s} & \cdots & -\sum_{n=1}^{s-1} t_{sn} \end{bmatrix}$$
$$= \begin{bmatrix} a_1 & a_2 & a_3 & \cdots & a_s \end{bmatrix}$$

$$b = [1\,0\,0\ldots0]^T$$

Define

$$B_m = \begin{bmatrix} a_1 & \cdots & a_{m-1} & b & a_{m+1} & \cdots & a_s \end{bmatrix}$$

i.e., $\{B_m\}_{m=1}^{s}$ is obtained by replacing the mth column of the matrix A by the vector b.
then $|A| = |B_1| + |B_2| + + |B_s|$ and $\pi_i(M) = \frac{|B_i|}{|A|}$

expand each $|B_i|$

$$|B_i| =$$

$$\left| \begin{matrix} -\sum_{n=2}^{s} t_{1n} & \cdots & t_{(i-1)1} & t_{(i+1)1} & \cdots \\ \vdots & \ddots & \vdots & \vdots & \ddots \\ t_{1(i-1)} & \cdots & -\sum_{n=1,n\neq i-1}^{s} t_{(i-1)n} & t_{(i+1)(i-1)} & \cdots \\ t_{1(i+1)} & \cdots & t_{(i-1)(i+1)} & -\sum_{n=1,n\neq i+1}^{s} t_{(i+1)n} & \cdots \\ \vdots & \ddots & \vdots & \vdots & \ddots \\ t_{1s} & \cdots & t_{(i-1)s} & t_{(i+1)s} & \cdots \end{matrix} \right.$$

1. *Permutation set:*
   $PS_i \triangleq \{\{k_j\}_{j=1,j\neq i}^{s} \mid \{k_j\}_{j=1,j\neq i}^{s}$ *is a permutation of* $s-1$ *integers in* $\{1,\ldots,s\}\setminus\{i\}\}$;
   $PS_i' \triangleq PS_i \setminus \{\{k_j\}_{j=1,j\neq i}^{s} | k_j = j\}$;
   $IPS_i \triangleq \{\ell \mid \ell$ *is an index for each* $\{k_j\}_{j=1,j\neq i}^{s} \in PS_i\}$;
   $IPS_i' \triangleq \{\ell \mid \ell$ *is an index for each* $\{k_j\}_{j=1,j\neq i}^{s} \in PS_i'\}$.
2. *Combination set:*
   $QS_i \triangleq \{\{k_j\}_{j=1,j\neq i}^{s} \mid k_j \in \{1,\ldots,s\}\setminus\{j\}, j=1,\ldots,s, j\neq i\}$;
   $QS_i' \triangleq QS_i \setminus \{\{k_j\}_{j=1,j\neq i}^{s} | k_j = i\}$;
   $IQS_i \triangleq \{\ell \mid \ell$ *is an index for each* $\{k_j\}_{j=1,j\neq i}^{s} \in QS_i\}$;
   $IQS_i' \triangleq \{\ell \mid \ell$ *is an index for each* $\{k_j\}_{j=1,j\neq i}^{s} \in QS_i'\}$.
3. *Optimum combination set:*
   $OS_i \triangleq \{\{k_j\}_{j=1,j\neq i}^{s} \in QS_i \mid k_j \in S^*,$ *for each* $j=1,\ldots,s, j\neq i\}$;
   $IOS_i \triangleq \{\ell \mid \ell$ *is an index for each* $\{k_j\}_{j=1,j\neq i}^{s} \in OS_i\}$.
4. *Nonoptimum combination set:*
   $NS_i \triangleq QS_i \setminus OS_i$;
   $INS_i \triangleq IQS_i \setminus IOS_i$.

$INS_1 = IQS_1 \setminus IOS_1.$

For example, if the configuration set $S = \{1,2,3\}$ and the global optimum set, as defined in equation (1), $S^* = \{1\}$, then

$PS_1 = \{\{2,3\},\{3,2\}\}$,

$PS_1' = \{\{2,3\},\{3,2\}\} \setminus \{2,3\} = \{\{3,2\}\}$,

$IPS_1 = \{A_1, A_2 | A_1$ represents $\{2,3\}, A_2$ represents $\{3,2\}\}$,

$IPS_1' = \{A_2 | A_2$ represents $\{3,2\}\}$,

$QS_1 = \{\{1,1\},\{1,2\},\{3,1\},\{3,2\}\}$,

$QS_1' = QS_1 \setminus \{1,1\} = \{\{1,2\},\{3,1\},\{3,2\}\}$,

$IQS_1 = \{B_1, B_2, B_3, B_4 | B_1$ represents $\{1,1\}, B_2$ represents $\{1,2\}$,
B_3 represents $\{3,1\}, B_4$ represents $\{3,2\}\}$,

$IQS_1' = \{B_2, B_3, B_4 | B_2$ represents $\{1,2\}, B_3$ represents $\{3,1\}$,
B_4 represents $\{3,2\}\}$,

$OS_1 = \{\{1,1\}\}$,

$IOS_1 = \{C_1 | C_1$ represents $\{1,1\}\}$.

NS_1, INS_1 can be expressed similarly and are omitted here.

LEMMA 4.2. *Let* $\mathcal{R}_i = \prod_{j=1,j\neq i}^{s} r_{ji}$ *and* $\mathcal{P}_i = \prod_{j=1,j\neq i}^{s} p_{ji}$ $\forall i \in S$. *Let also* $\mathcal{R}_i^{(\ell)} = \prod_{j=1,j\neq i}^{s} r_{jk_j}$ *and* $\mathcal{P}_i^{(\ell)} = \prod_{j=1,j\neq i}^{s} p_{jk_j}$ $\forall i \in S$, *where* $\{k_j\}_{j=1,j\neq i}^{s} \in QS_i'$ *and* $\ell \in IQS_i'$.

Then the expansion of $|B_i|$, $\forall i \in S$, *has the following properties:*

1. $|B_i| = (-1)^{(s-1)}[\mathcal{R}_i\mathcal{P}_i + \sum_{\ell \in IQS_i'} C_i^{(\ell)}\mathcal{R}_i^{(\ell)}\mathcal{P}_i^{(\ell)}]$, *where* $C_i^{(\ell)}$ *is the number of times that* $\mathcal{R}_i^{(\ell)}\mathcal{P}_i^{(\ell)}$ *appears in the summation;*
2. $(-1)^{(s-1)}|B_i| > 0$;
3. $\forall i \in S^*, \mathcal{P}_i = \mathcal{P}_i^{(\ell)}$, *if* $\ell \in IOS_i$;
4. $\forall i \in S^*, \mathcal{P}_i > \mathcal{P}_i^{(\ell)}$, *if* $\ell \in INS_i$;
5. $\forall i \in S^*, \mathcal{P}_i > \mathcal{P}_n$, *if* $n \in S \setminus S^*$;
6. $\forall i \in S^*, \mathcal{P}_i > \mathcal{P}_n^{(\ell)}$, *if* $n \in S \setminus S^*$.

$$\mathcal{P}_i = \prod_{j=1,j\neq i}^{s} p_{ji} = \prod_{j=1,j\neq i}^{s} \Pr[H(i) < H(j)]$$

$$\mathcal{P}_i^{\ell} = \prod_{j=1,j\neq i}^{s} p_{jk_j} = \prod_{j=1,j\neq i}^{s} \Pr\left[H\left(k_j\right) < H(j)\right]$$

4.2. Convergence of $\pi(M)$ to an optimal probability vector. Let $M \to \infty$ and note that $\mathcal{P}_i^M$ and $[\mathcal{P}_i^{(\ell)}]^M$ $\forall i \in S^*$ and $\forall \ell \in IOS_i$ will dominate all other $[\mathcal{P}_i^{(\ell)}]^M$ $\forall \ell \in INS_i$ and $\mathcal{P}_j^M$, $[\mathcal{P}_j^{(\ell)}]^M$ $\forall j \in S \setminus S^*$ and $\forall \ell \in IQS_j$. Therefore, we have

$$\lim_{M \to \infty} \pi_i(M) = \begin{cases} e_i^* > 0 & \text{if} \quad i \in S^*, \\ 0 & \text{if} \quad i \in S \setminus S^*. \end{cases}$$

Hence as $M \to \infty$, the quasi-stationary probability vector converges to an optimal probability vector.

4.3. Monotone property of the quasi-stationary probabilities. From the form of $\pi_1(M)$, $\pi_2(M), \dots, \pi_s(M)$, we see that $\exists M^* < \infty$, such that for $M_k > M^*$ the quasi-stationary probabilities have a monotone property, namely,

$$\begin{aligned} \pi_i(M_{k+1}) > \pi_i(M_k) \quad & \text{for} \quad i \in S^*, \\ \pi_i(M_{k+1}) < \pi_i(M_k) \quad & \text{for} \quad i \in S \setminus S^*. \end{aligned}$$

Convergence of the SC algorithm

Define

$$\mathrm{f}^{(k)} = \mathrm{f}^{(0)} P_1 P_2 \cdots P_k \quad \text{and} \quad \mathrm{f}^{(m,k)} = \mathrm{f}^{(0)} P_{m+1} P_{m+2} \cdots P_{m+k}$$

A time-inhomogeneous Markov chain $\{Y_k\}$ is called weakly ergodic if, $\forall m$

$$\lim_{k\to\infty} \sup_{\mathrm{f}^{(0)},\mathrm{g}^{(0)}} \left\| \mathrm{f}^{(m,k)} - \mathrm{g}^{(m,k)} \right\| = 0$$

where $f(0)$ and $g(0)$ are starting probability vectors.
A time-inhomogeneous Markov chain $\{Y_k\}$ is called strongly ergodic if there exists a probability vector q such that, $\forall m$

$$\lim_{k\to\infty} \sup_{\mathrm{f}^{(0)}} \left\| \mathrm{f}^{(m,k)} - \mathrm{q} \right\| = 0$$

where $f(0)$ is a starting probability vector.

weak ergodicity

Theorem A.1 (Theorem V.3.2 of [19])

Let $\{X_n\}$ be a nonstationary Markov chain with transition matrices,$\{P_n\}_{n=1}^{\infty}$. The chain, $\{X_n\}$, is weakly ergodic if and only if there exists a subdivision of $P_1 \cdot P_2 \cdot P_3 \cdot\cdot$ into blocks of matrices $[P_1 \cdot P_2 \cdots P_{n_1}] \cdot [P_{n_1+1} \cdot P_{n_1+2} \cdots P_{n_2}] \cdots [P_{n_j+1} \cdot P_{n_j+2} \cdots P_{n_j+1}]\ldots$ such that

$$\sum_{j=0}^{\infty} \alpha \left(P^{\left(n_j, n_{j+1}\right)} \right) = \infty$$

where $n_0 = 0$, $P(m,k) = P_{m+1} \cdot P_{m+2} \cdots P_k$,
$\alpha(P) = \min_{i,k} \sum_{j=1}^{\infty} min(p_{ij}, p_{kj})$

strong ergodicity

Theorem A.2 (Theorem V.4.3 of [19])

Let Pn be a sequence of transition matrices corresponding to a nonstationary weakly ergodic Markov chain with $P_n \in A$ for all n. If there exists a corresponding sequence of left eigenvectors ϕ_n, satisfying

$$\sum_{j=0}^{\infty} \|\phi_j - \phi_{j+1}\| < \infty$$

then the chain is strongly ergodic.

A testbed system

1.We design a testbed system with one million configurations.
2.we divided the interval [10.0,110.0] into five subintervals with equal lengths of 20.0. Then we generated $pl\%, l = 1, ..., 5$, from the total configurations with objective function value uniformly distributed in subinterval l. we allocate fewer points in the first interval ("good interval") than in others.
3.Each sample of objective function for configuration i is generated according to $H(i) = g(i) + W_i$, where W_i models the behavior of a Monte Carlo simulator. For these experiments, we take $W_i \sim unif[a/2, a/2] \, \forall i \in S$.

Comparison of the SC and SR algorithms

SC:
1.$M_k = 1 + k/500$
2.$R(i,j) = 1/999999 \forall i, j \in S, i \neq j$
SR:
1.$M_k = 1 + k/500$
2.$(a, b) unif [0, 120]$
3.two sets of experiments, one with a "closed neighborhood structure" and one with an "open neighborhood structure."

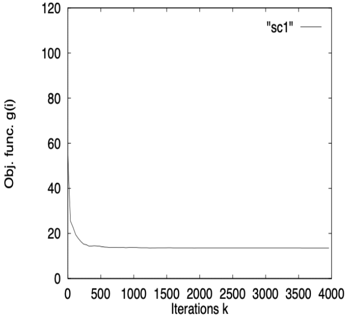

FIG. 1. *Optimization trajectory of SC* $(a = 10)$.

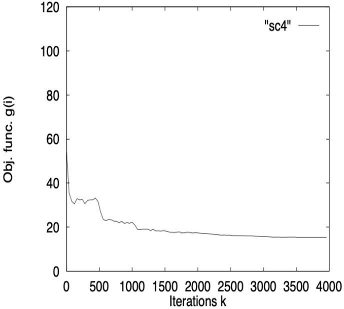

FIG. 2. *Optimization trajectory of SC* $(a = 40)$.

Figure 2

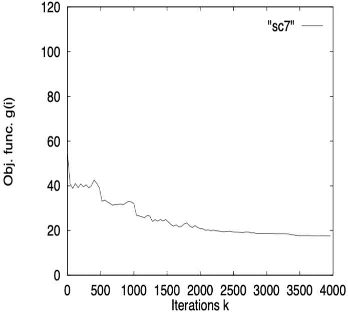

FIG. 3. *Optimization trajectory of SC* ($a = 70$).

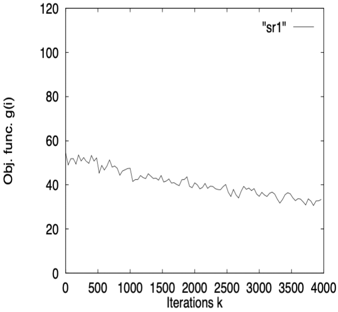

FIG. 4. *Optimization trajectory of SR with a closed neighborhood* ($a = 10$).

Figure 3

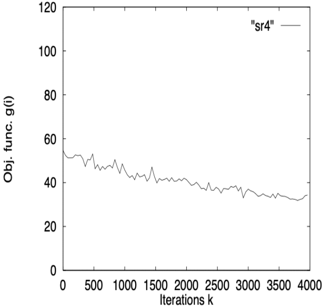

FIG. 5. *Optimization trajectory of SR with a closed neighborhood* ($a = 40$).

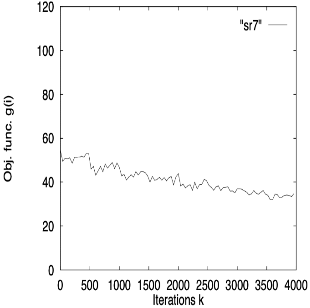

FIG. 6. *Optimization trajectory of SR with a closed neighborhood* ($a = 70$).

Figure 4

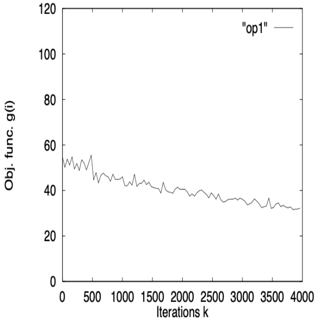

FIG. 7. *Optimization trajectory of SR with an open neighborhood* $(a = 10)$.

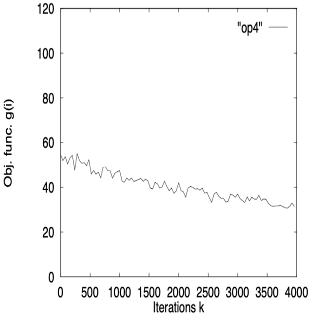

FIG. 8. *Optimization trajectory of SR with an open neighborhood* $(a = 40)$.

Figure 5

- SC converges to a good solution very quickly, even when the estimates of the objective function are very noisy
- As can be seen from the figures, the SC algorithm performs much better than the SR algorithm on the particular optimization problem examined.

Thank you!