Overview of Stochastic Methods (1)

- Two phases: global and local
  - global phase: function is evaluated at a number of randomly sampled points
  - local phase: sample points are manipulated, for example, by local searches, to yield candidate global minima
- Challenge for global optimization algorithms
  - increase their efficiency while maintaining reliability
- Many stochastic methods for global optimization
  - random search, multistart, clustering, controlled random search, simulated annealing, acceptance-rejection (A-R), stochastic integration, genetic, tabu search, and some other nature-inspired methods

Overview of Stochastic Methods (2)

- Based on some variation of pure random search
  - develop stopping criteria
  - develop techniques to approximate the region of attraction for a local minimum point
    - When the search for the local minimum started from a point within a certain region around the minimum converges to the same minimum point
- Characteristics
  - do not offer an absolute guarantee of success
  - probability that a point within a distance ε will be found approaches 1 as the sample size increases
  - algorithm run at different times from the same starting point can generate different design histories and local minima → run several times before the solution is accepted as the global optimum

Pure Random Search Method

- simplest stochastic method for global optimization
- consists only of a global phase
  - Evaluate f(x) at N sample points drawn from a random uniform distribution over the set S_b
  - smallest function value found
- asymptotically guaranteed to converge, in a probabilistic sense, to the global minimum point
- quite inefficient because of the large number of function evaluations
- single start: simple extension
  - single local search is performed (if the problem is continuous) starting from the best point in the sample set at the end of pure random search

Multistart Method

- extension of pure random search: (global + local) phase
  - Step 1: Take a random point $x^{(0)}$ from a uniform distribution over the set S_b
  - Step 2: Start a local minimization procedure from $x^{(0)}$
  - Step 3: Return to step 1 unless a stopping criterion is satisfied
  - local minimum with the smallest function value
- reliable, but not efficient
  - many sample points will lead to the same local minimum
- Stopping Criterion (M = K)
  - local minimum has a fixed probability of being found in each trial
  - Given that M distinct local minima have been found in L searches, the optimal Bayesian estimate of the unknown number of local minima K is given by

$$K = \mathrm{int}\left[\frac{M\left(L-1\right)}{L-M-2}\right] \quad \text{if } L \geq \left(M+3\right)$$

Clustering Method (1)

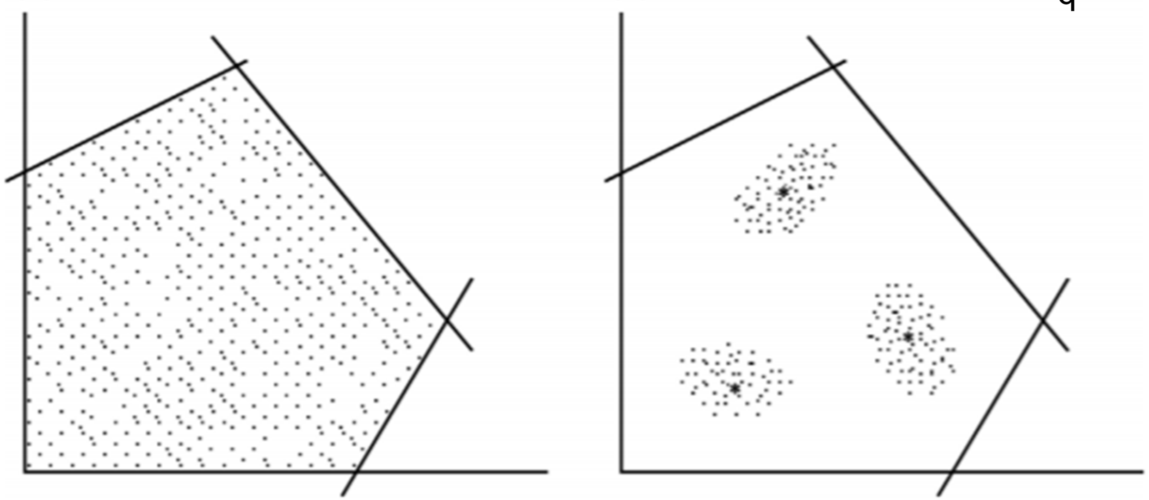
- remove the inefficiency of the multistart method
  - use the local search procedure only once for each local minimum point
  - random sample points are linked into groups to form clusters
  - Each cluster → region of attraction for a local minimum point
- major disadvantage
  - performance depends heavily on the dimension of the problem
- Reduced sample points
  - set of random points → regions that are likely to contain local minima
  - Reduction: set of sample points → number of components (disjointed)
    - Each component (cluster) → at least one local minimum point

$$f_q\text{-level set of } f(\mathbf{x}) \text{ or reduced set} : A_q = \left\{ \mathbf{x} \in A_N \mid f(\mathbf{x}) \le f_q \right\}$$

  - Concentration: a few steepest–descent steps are applied to every sample point
    - transformed points are not uniformly distributed

Clustering Method (2)

- Four clustering methods
  - density, single linkage, mode analysis, and vector quantization multistart
- Assumptions
  - All local minima of f(x) lie in the interior of S_b
  - Stationary points are isolated
  - Local search is always that of descent
  - The way A_q changes with different samples does not affect analysis
  - each local minimum with a function value smaller than f_q is actually found



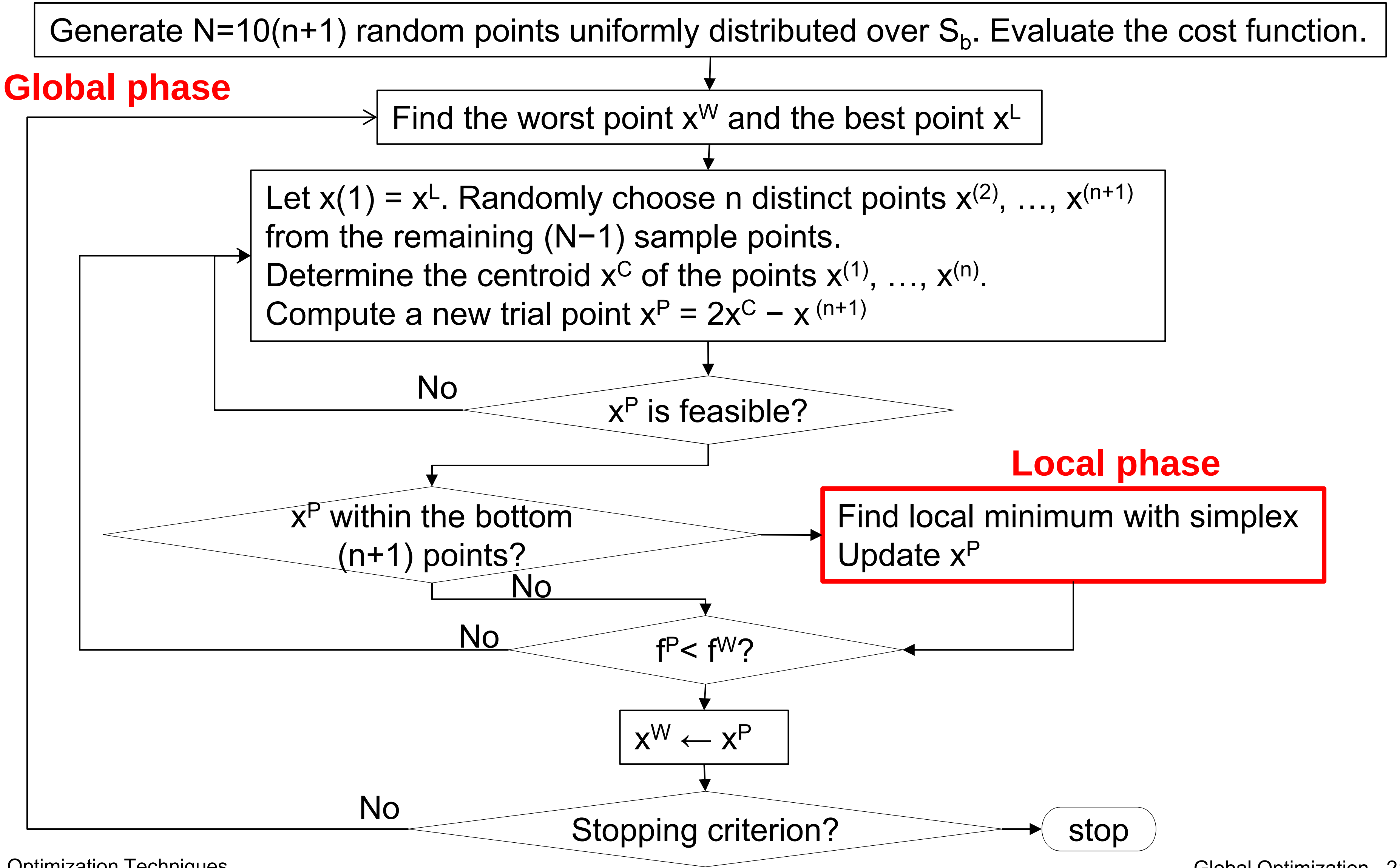
Clustering Method (3)

- Density clustering
  - Regions of attraction are approximated by hyperspheres or ellipsoids with centers at local minimum points
  - Reduced sample points are added to clusters based on their distance from the centers (seed point)
  - best-unclustered point is used in a local search procedure to find a local minimum point: new?→seed, otherwise
- Single linkage clustering
  - Points are linked to others in their proximity as opposed to being linked to the clusters' centers or seeds
  - A point is assigned to a cluster if it falls within a critical distance r_k from any point that already belongs to that cluster

Clustering Method (4)

- Mode analysis clustering
  - information at only two points at a time (density, single linkage) → clusters are formed using more information
  - set S_b is partitioned into nonoverlapping, small hypercubic cells that cover S_b entirely
- Vector quantization
  - theories of lattices and vector quantization to form clusters
  - entire space S_b is divided into a finite number of cells (mode analysis) and a code point is associated with each one
    - need not be sample points (centroids of the cells)
  - most suitable code point: smallest function value of a cell
  - Identification of a cluster is done using vector quantization of the reduced sample points

Controlled Random Search (CRS)



Acceptance–Rejection Methods

- modifications of the multistart algorithm
  - improve its efficiency by using ideas from statistical mechanics
  - tunnel below irrelevant local minima → random tunneling
- acceptance phase
  - start local minimization from a randomly generated point even if it has a higher cost function value than that at a previously obtained local minimum
- rejection phase
  - local minimization procedure produces a local minimum that has a higher cost function value than a previously obtained minimum, then the new minimum point is rejected

$$p\left(\Delta E\right) = e^{-\frac{\Delta E}{kT}} \leftrightarrow p\left(\mathbf{x}\right) = e^{-\frac{\max(0,\left[f(\mathbf{x})-\bar{f}\right]}{F}}$$

$\bar{f}$: estimate of the upper bound of the global minimum

F : target value for the global minimum

Stochastic Integration

- Introduce suitable stochastic perturbation of the system of equations for the trajectory methods
  - By changing some coefficients in the differential equations, we get different solution processes starting from the same initial point

$$\left.\begin{aligned}&\dot{\mathbf{x}}(t) = -\nabla f(\mathbf{x}) \\ &\rightarrow d\mathbf{x}(t) = -\nabla f(\mathbf{x})\,dt + \varepsilon(t)\,d\mathbf{w}(t)\end{aligned}\right\}\ \text{with }\ \mathbf{x}(t) = \mathbf{x}^{(0)}$$

$\mathbf{w}(t)$: n-dimensional standard Wiener process $\xrightarrow{\text{actual implementation}}$ standard Gaussian distribution

$\varepsilon(t)$: real function called the noise coefficient

when $\varepsilon(t) = \varepsilon_0$, probability density function of $\mathbf{x}(t) \rightarrow$ limit density $Ze^{\left[-\frac{2f(\mathbf{x})}{\varepsilon_0^2}\right]}$ as $t \rightarrow \infty$

peaks become narrower with a smaller ε_0

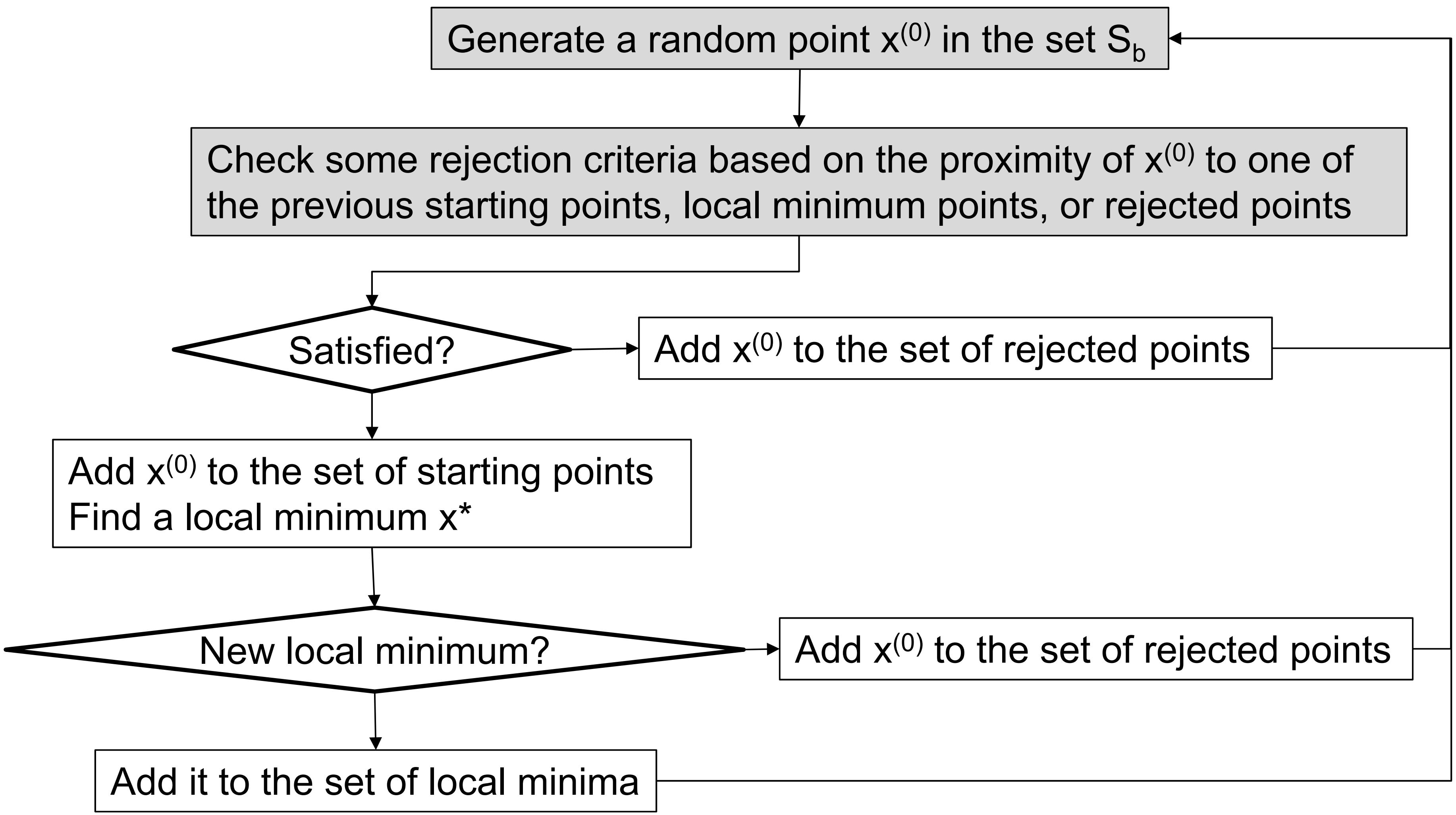
$\varepsilon_0 \leftrightarrow$ target level F that decreases in the simulated annealing method

$\varepsilon(t)$ is continuous and suitably tends to zero as $t \rightarrow \infty$

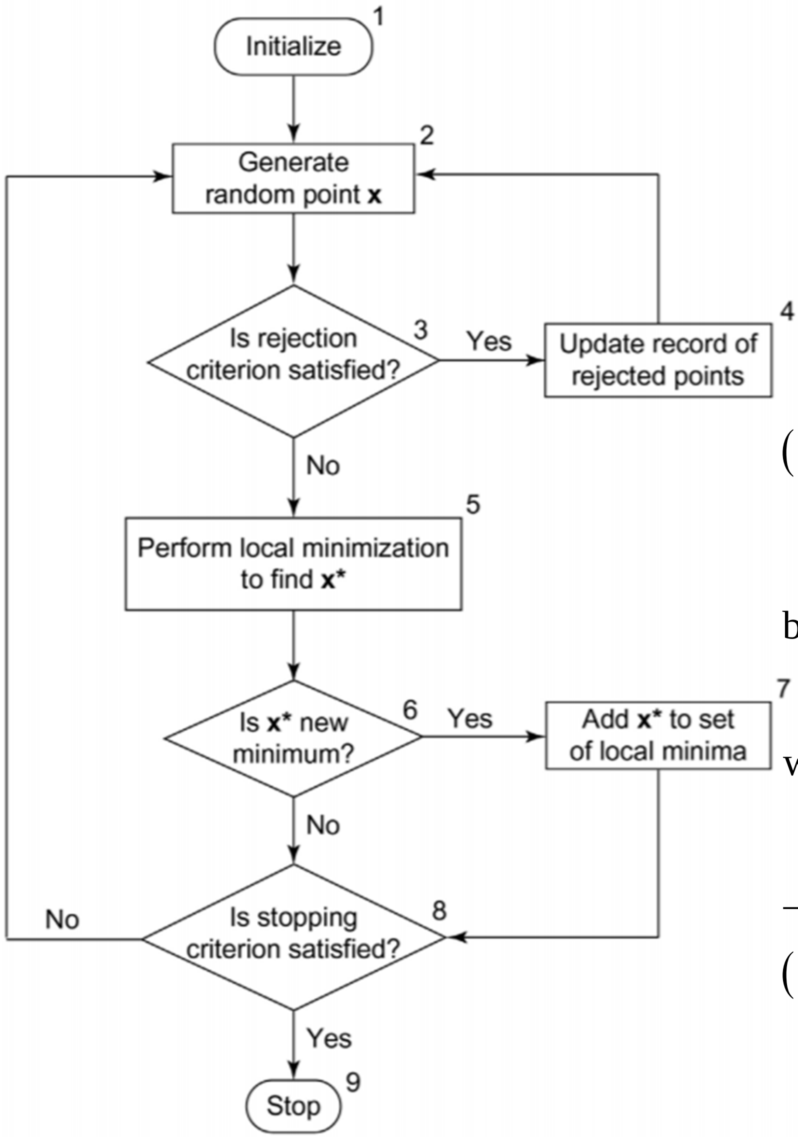
Local-Global Stochastic Method (1)

- modification of the multistart procedure but with the ability to learn as the search progresses
- explore the entire feasible domain in a systematic way for the global minimum
  - generation and evaluation of a random point are much cheaper than one local minimization
  - avoids searching near any local minimum point and all of the points leading to it → increase the chance of finding a new local minimum in the unexplored region
  - several sets that contain certain types of points are constructed
    - Set of local minima (S_*)
    - Set of starting points $x^{(0)}$ (S_0)
    - Set of rejected points (S_r)

Local-Global Stochastic Method (2)



Domain Elimination Method (1)



c_1 : number of elements in S_r, the set of rejected points

c_2 : number of rejected points that were near S_r

c_3 : number of elements in S_0, the set of starting points

c_4 : number of elements in S_*, the set of local minimum points

$$(3)+(4)\text{: if } \mathbf{x}^R \text{ is } \begin{cases} \text{in or near } S_* \textit{ or } S_0 \rightarrow \text{add } \mathbf{x}^R \text{ to } S_r \text{ and } c_1 = c_1 + 1 \\ \text{in or near } S_r \rightarrow c_2 = c_2 + 1 \\ \text{near a trajectory } \left(\text{a mapping } S_0 \rightarrow S_*\right) \\ \rightarrow \text{add } \mathbf{x}^R \text{ to } S_r \text{ and } c_1 = c_1 + 1 \end{cases}$$

before (5): add $\mathbf{x}^R$ to S_0 and set $\mathbf{x}^{(0)} = \mathbf{x}^R$ and $c_3 = c_3 + 1$

$$\text{within } (5)\text{: if } \mathbf{x}^M \text{ is } \begin{cases} \text{in or near } S_0 \rightarrow \text{add } \mathbf{x}^R \text{ to } S_r \text{ and } c_1 = c_1 + 1 \\ \text{in or near } S_r \rightarrow c_2 = c_2 + 1 \\ \text{near a trajectory} \rightarrow \text{add } \mathbf{x}^R \text{ to } S_r \text{ and } c_1 = c_1 + 1 \end{cases}$$

$\rightarrow$ store the trajectory from $\mathbf{x}^{(0)}$ to $\mathbf{x}^M$ $\rightarrow$ go to (2)

(7): add it to S_* and $c_4 = c_4 + 1$

Domain Elimination Method (2)

- Stopping criteria
  - Maximum size of sets (based on the number of design variables and a confidence level parameter)
    - Set of local minima (S_*): 10n
    - Set of starting points $x^{(0)}$ (S_0): 40n
    - Set of rejected points (S_r): 40n
  - Number of local minima ≥ (Bayesian estimate for the number of local minima)

Stochastic Zooming Method

- extension of the zooming method
  - as the target level for the cost function grows closer to the global optimum, it becomes difficult to find a feasible point in the set S
  - Eventually, the modified problem needs to be declared infeasible to stop the algorithm
- major difference from domain elimination
  - addition of the zooming constraint
- domain elimination algorithm can be used with some minor modification
  - keep track of the number of local searches (steps 4 and 5) that did not terminate at a feasible minimum point
  - additional stopping criterion: if the cost function value reaches a target value F specified by the user

Operations Analysis of Methods (1)

- Distance between a point and a set of point (checking the proximity of a point to a set)

$\mathbf{x}^S$: point belonging to one of the sets

$\mathbf{x}^M$: either a random point or an intermediate point

(1) hypersphere of either a constant or a variable radius around $\mathbf{x}^S$

point is rejected if $D\left(=\left\|\mathbf{x}^S-\mathbf{x}^M\right\|\right)\leq D_{cr}\left(=\alpha\left\|\mathbf{x}\right\| \text{ where } \mathbf{x}=\mathbf{x}^R \text{ or } \mathbf{x}^M \text{ and } 0.01\leq\alpha\leq 0.2\right)$

(2) hyperprism around $\mathbf{x}^S$

If the proposed point lies inside the hyperprism, it is rejected.

the distance between the two points is not required

Operations Analysis of Methods (2)

- Approximation of the trajectory between a starting point and a local minimum (trajectory approximation)
  - Trajectory: design history from a starting point to the corresponding local minimum point
    - straight line connecting $x^{(0)}$ and the corresponding x^*
    - least squares straight line through several points along the trajectory
    - straight-line segments through selected points along the trajectory
    - a quadratic curve through three points
    - quadratic segments through groups of three points
    - higher-order polynomial or spline approximations
  - Several issues affect the choice of the technique to use
    - number of points needed (which have to be stored), number of operations, accuracy of the approximation

Operations Analysis of Methods (3)

- Distance between a point and a trajectory
  - decision of whether a point x lies near the trajectory
  - Triangle method
    - calculate the internal angle $x^{(0)}-x-x^*$ of the triangle formed by the three points
    - point x is rejected if the angle is larger than a threshold value (150~175°)
    - ellipsoidal body around the line segment $x^{(0)}-x^*$
    - critical offset distance ~ the trajectory's length ~ size of the region of attraction
  - Offset method
    - calculate an offset distance by generating x_bar as a projection of x on the line $x^{(0)}-x^*$
    - If x_bar lies outside the line segment $x^{(0)}-x^*$, then it is accepted
    - cylinder constructed with $x^{(0)}-x^*$ as its axis
    - uniform critical distance
  - Truncated cone
    - the larger base at x^* and the smaller base at $x^{(0)}$, thus allowing a better identification of close regions of attraction

Summary of Features of Methods

Method	Can solve discrete problems?	General constraints?	Tries to find all x*?	Phases	Needs gradients?
Covering (D)	No	No	Yes	G	1
Zooming (D)	Yes[1]	Yes	No	L	1
Generalized descent (D)	No	No	No	G	Yes
Tunneling (D)	No	Yes	No	L + G	1
Multistart (S)	Yes[1]	Yes	Yes	L + G	1
Clustering (S)	Yes[1]	Yes	Yes	L + G	1
Controlled random search (S)	Yes	No	No	L + G	No
Acceptance–rejection (S)	Yes[1]	Yes	No	G	No
Stochastic integration (S)	No	No	No	G	No
Genetic (S)	Yes	No	No	G	No
Stochastic zooming (S)	Yes[1]	Yes	No	L + G	1
Domain elimination (S)	Yes[1]	Yes	Yes	L + G	1

Note: *D*, deterministic methods; *S*, stochastic methods; *G*, global phase; *L*, local phase.
[1]*Depends on the local minimization procedure used.*

Performance of Some Methods with Unconstrained Problems

- Methods: covering, A–R, controlled random search (CRS), and simulated annealing (SA)
- 29 unconstrained problems with known global solutions
- one to six design variables and only explicit bounds on the variables
- Results: CRS > A-R > SA > covering
  - Covering: not practical (n>2), difficult to estimate Lipschitz constant
  - A-R: no stopping criterion
  - CRS: not successful to treat general constraints explicitly

Performance of Stochastic Zooming and Domain Elimination Methods (1)

- Methods: stochastic zooming method (ZOOM) and domain elimination (DE), CRS, SA
  - local search: sequential quadratic programming (SQP) method
  - ZOOM: percent reduction of 15% (γ=0.85)
- Ten mathematical programming test problems
  - Four problems had no constraints
  - The number of design variables varied from 2 to 15
  - The total number of general constraints varied from 2 to 29
  - Two problems had equality constraints
  - All problems had 2 or more local minima
  - Two problems had 2 global minima and one had 4
  - One problem had a global minimum of 0
  - Four problems had negative global minimum values

Performance of Stochastic Zooming and Domain Elimination Methods (2)

- Evaluation criteria (averages after five times)
  - Number of random starting points
  - Number of local searches performed
  - Number of iterations used during the local search
  - Number of local minima found by the method
  - Cost function value of the best local minimum (the global minimum)
  - Total number of calls for function evaluations
  - CPU time used
- Results
  - Global solutions: DE(9/10) > ZOOM(7/10)
    - DE found more local minima than ZOOM did (tunnel under some minima)
  - number of function evaluations and CPU time: (CRS >) DE > ZOOM
  - SA: failed six problems, even successful, (3~4)times longer CPU time than DE, more suitable for problems with discrete variables only

Global Optimization of Structural Design Problems

- Six structures → 28 test problems
  - 10-bar cantilever truss, 200-bar truss, 1-bay & 2-story frame, 2-bay & 6-story frame, 10-member cantilever frame, 200-member frame
- Objectives: minimize the weight of the structure
- number of design variables: 4~116
  - cross-sectional shape of members to hollow circular tubes or I-sections
  - material from steel to aluminum
- number of stress constraints: 10~600
- number of deflection constraints: 8~675
- number of local buckling constraints: 0~72
- total number of general inequality constraints: 19~1276

Global Optimization of Structural Design Problems

- Results: five runs
  - ZOOM found only one local minimum for all but two problems
    - For most of the problems, the global minimum was found with the first random starting point
  - DE found many local minima for all problems except for one, which turned out to be an infeasible problem
  - CPU times
    - difficult to draw a general conclusion about the relative efficiency of the two methods
  - only the global minimum? ZOOM
  - all or most of the local minima are wanted? DE