Evolution Artificielle

Optimisation sous Contraintes

M. Sebag

Equipe Inférence et Apprentissage

LRI (CNRS 8623)

Université Paris-Sud

Michele.Sebag@lri.fr

http://www.lri.fr/˜sebag

Février 2003

Overview

- **Projection methods**
  - Linear constraints: Genocop II — Specialized operators
  - Surface operators — Equality constraints or boundary of $\mathcal{F}$
  - Transforming the feasible space — General approach for inequality constraints
  - Evolutionary projection — Behavioral Memory
- **Penalty methods** — Difficult to tune
  - Static penalties
  - Dynamic penalties (exogenous)
  - Adaptive penalties (endogenous)

Michalewicz & Schoenauer, Evolutionary Computation **4**(1) 1996

. . . Already outdated!

Constrained Optimization

In continuous search spaces

Let $f,\ (g_i)_{i=1,\ldots,m} : \mathcal{S} \subset \mathbb{R}^n \mapsto \mathbb{R}$

$$\text{Let } \mathcal{F} = \{y \in \mathcal{S} \text{ s.t. } \begin{array}{l} g_i(x) \leq 0, \ \ i = 1, \ldots, q \text{ and} \\ g_i(x) = 0, \ \ i = q+1, \ldots, m\} \end{array}$$

$$\text{Find } x^* \in \mathcal{F} \text{ s.t. } f(x) = \min\{f(y);\ y \in \mathcal{F}\}$$

- f is the **objective** function
- $(g_i)_{i=1,\ldots,q}$: **inequality** constraints
- $(g_i)_{i=q+1,\ldots,m}$: **equality** constraints
- $\mathcal{F}$: *feasible* domain
- Inequality constraint g_i is **active** at point $x \in \mathcal{F}$ if $g_i(x) = 0$
- Points for which some constraints are active are on the **boundary** of the feasible domain if constraints are continuous

Projection methods

Only search the feasible space

but maintaining good search properties

- Easy in very specific cases — Balls or other convex spaces, ...
- Post-treatment — Death penalty, repair, ...

  Generally leads to a too biased search
- Specific algorithm
  - Initialization — Uniform on $\mathcal{F}$?
  - Closed operators on $\mathcal{F}$ — i.e. offspring of parents in $\mathcal{F}$ must be in $\mathcal{F}$
  - Respecting some basic principles — cf Radcliffe 93
- Transforming $\mathcal{F}$ — and back to an easy case

Linear constraints

. . . but non-linear fitness!

Genocop II (Michalewicz & Janikow 91) :

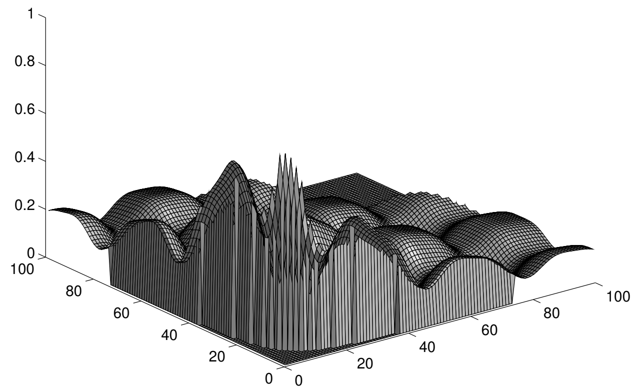
- Equality constraints:
  Elimination of variables
- Inequality constraints:
  - Arithmetical crossover (or BLX-α)
  - Truncated mutation
  - Initialization?

Generalization?

Searching the boundary

A first example (Michalewicz 96)

$$\begin{cases} \max f(\vec{x}) = \left| \frac{\sum_{i=1}^{n} cos^4(x_i) - 2\prod_{i=1}^{n} cos^2(x_i)}{\sqrt{\sum_{i=1}^{n} i x_i^2}} \right|, \\ 0 \leq x_i \leq 10 \\ \prod_{i=1}^{n} x_i \geq 0.75 \end{cases}$$



Second constraint is active at optimum:

Search on hyperboloid $\prod_{i=1}^{n} x_i = 0.75$

Search on hyperboloid

- **Initialization**

$$x_i = U[0, 10], i \in [1, n]$$
$$x_i \leftarrow \frac{0.75}{\prod_{j=1}^{n} x_j} x_i$$

- **Geometrical crossover**

$$(x_i)(y_i) \rightarrow (x_i^{\alpha} y_i^{1-\alpha})$$
$$\text{with } \alpha = U[0, 1]$$

- **Geometrical mutation**

$$x_i \rightarrow q x_i, x_j \rightarrow \frac{x_j}{q}.$$
$$\text{with } q = U[0, 1]$$

Best known results (until very recently :-(

Other example (Michalewicz 96)

$$\begin{cases} Prod(\vec{x}) = (\sqrt{n})^n \cdot \prod_{i=1}^n x_i, \\ 0 \leq x_i \leq 1, \\ \sum_{i=1}^n x_i = 1. \end{cases}$$

- **Initialization**
  $U([0,1]^n)$ + projection (**not** uniform).
- **Sphere crossover**

$$(x_i)(y_i) \rightarrow (\sqrt{\alpha x_i^2 + (1-\alpha) y_i^2})$$
$$\text{with } \alpha = U[0,1]$$

- **Sphere mutation**

$$x_i \rightarrow p \cdot x_i$$
$$x_j \rightarrow q \cdot x_j,$$
$$\text{with } p = U[0,1] \text{ et } q = \sqrt{(\frac{x_i}{x_j})^2 (1-p^2) + 1}.$$

Very good results rapidly obtained

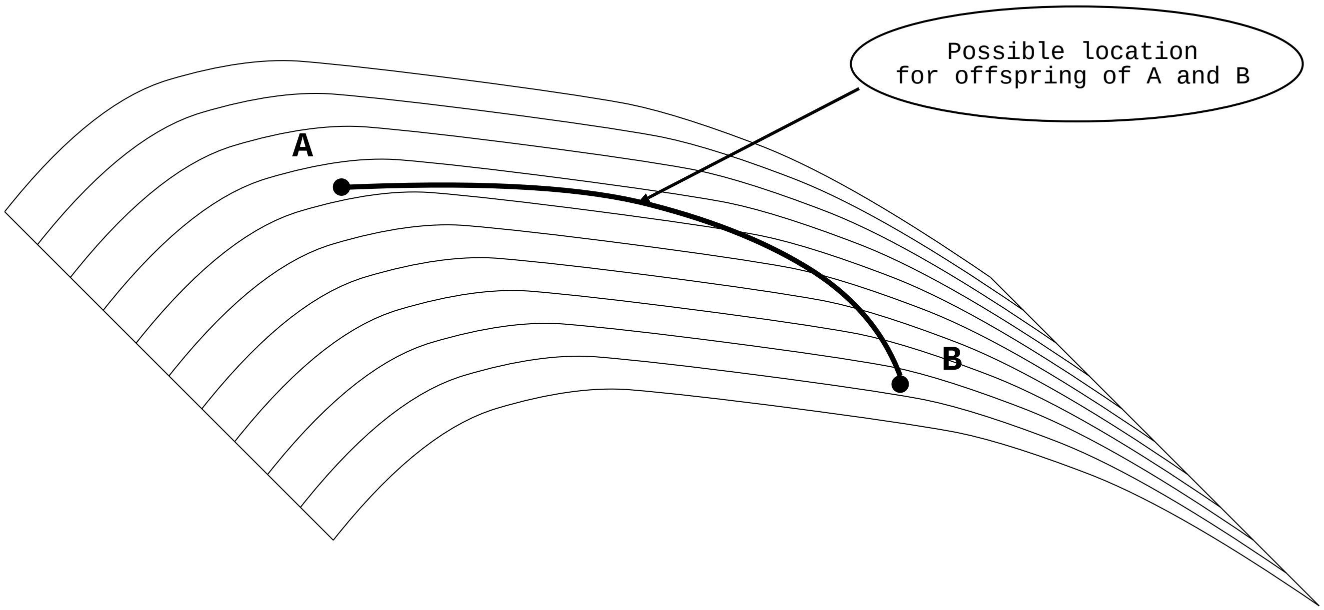
Search a surface

- For equality constraints,
- or if active constraints are known in advance

- Restrict the search to a surface (Riemanian variety) $\mathcal{B}$.
- Need to define surface operators
- **highly** problem dependent
- but always possible on the sphere

Curve-based operators

Crossover:



Mutation:

Crossover with random point . . . if you can generate random points on $\mathcal{B}$

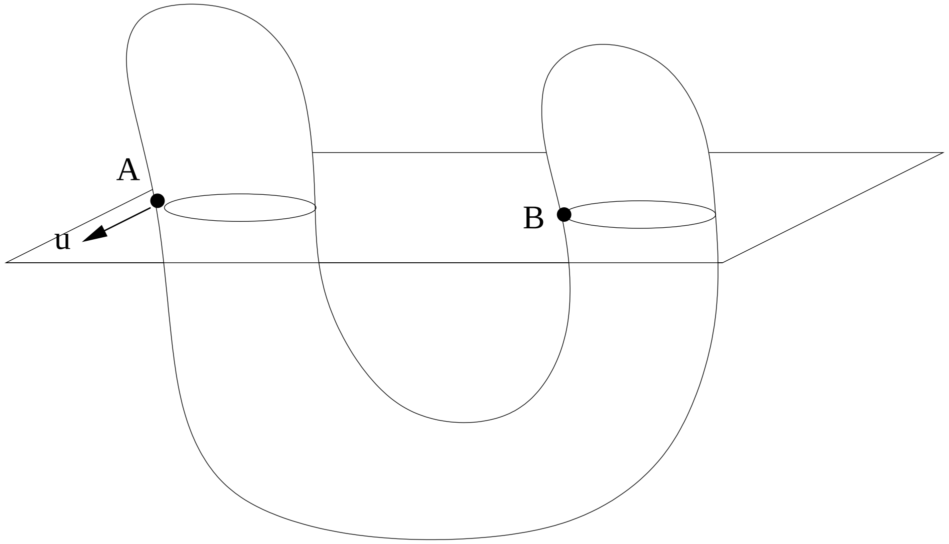
Riemanian geometry

- Theoretical results
  - Geodesical curves: Existence and (generally) uniqueness
  - Covering of local neighborhood for mutation
- No general numerical algorithm
- Requires heavy symbolic calculations ...

... easy on the sphere.

Plane-based curves

- Intersection of $\mathcal{B}$ and plane $(A, \vec{AB}, \vec{u})$ is a curve on $\mathcal{B}$ for almost all (random) vectors $\vec{u}$.
- might not connect A and B!



... no problem on the sphere.

Parametric representations

$$\mathcal{B} = \{\vec{\Phi}(t_j), (t_j) \in [a_j, b_j]\}$$

Initialization:

Uniform random choice of parameters $(t_j) \in [a_j, b_j]$.

Intermediate crossover: $\alpha \in [0, 1]$

$$\vec{\Phi}(t_j), \vec{\Phi}(u_j) \longrightarrow \vec{\Phi}(\alpha t_j + (1-\alpha) u_j)$$

Gaussian mutation:

$$\vec{\Phi}(t_j) \longrightarrow \vec{\Phi}(t_j + N(0, \sigma_j))$$

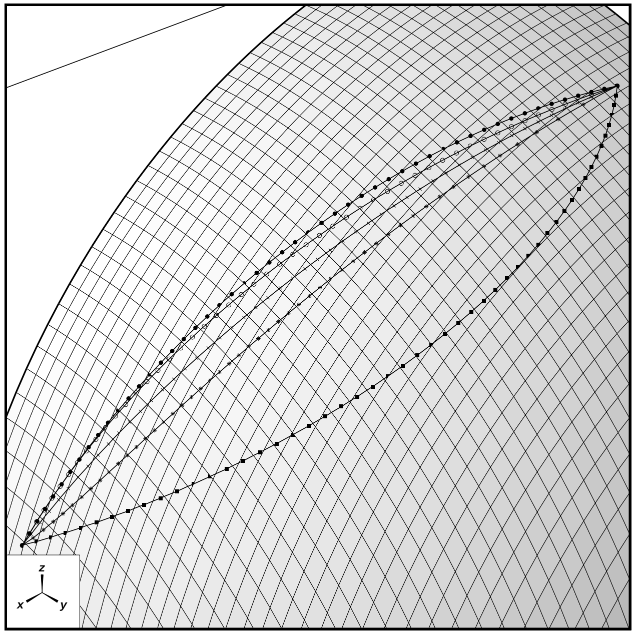
Provision for adaptive mutation rate.

Parametric representation of the sphere

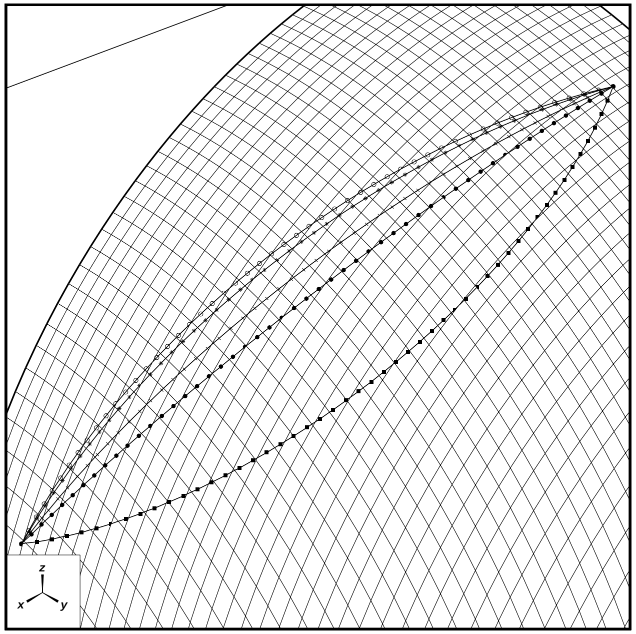
$$\left\{\begin{array}{lcl} x_1 & = & \cos(t_1) \\ x_2 & = & \sin(t_1)\cos(t_2) \\ \ldots & & \\ x_{n-1} & = & \sin(t_1)\ldots\sin(t_{n-2})\cos(t_{n-1}) \\ x_n & = & \sin(t_1)\ldots\sin(t_{n-2})\sin(t_{n-1}) \end{array}\right. \quad (1)$$

$n!$ different representations.

A priori comparison



P_1, P_2, G, Sp, P_3.



Angle with gradient

$0, 15, 20, -15, -57$ degrees.

Localizations of possible offspring two parents for different choices of crossover on the sphere

Experimental comparisons

Algorithms:

- Generational real-coded GA, 2-tournament selection, population size 60, $p_{cross} \in [0.6, 1]$ and $p_{mut} \in [0, 0.4]$.
- $(7 + 60) - ES$ $p_{cross} \in [0.6, 1]$ and $p_{mut} \equiv 1$

Operators:

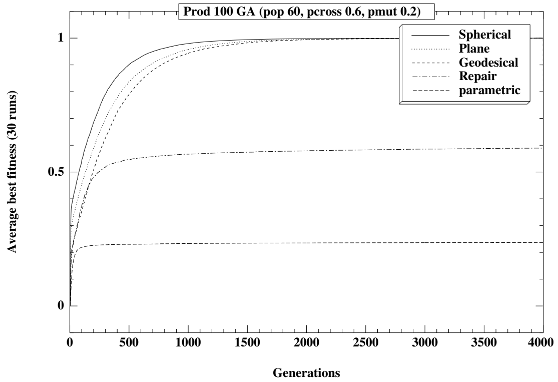
- Standard (arithmetical crossover and Gaussian mutation) + repair
- Spherical
- Geodesical
- Plane-based
- Parametric (standard operators on parameters).

All results are averages over 31 runs.

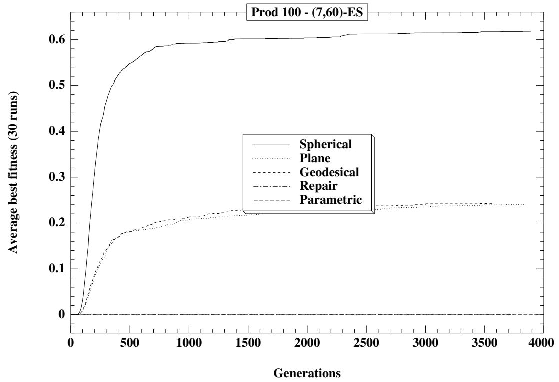
The product function

$P(\vec{x}) = (\sqrt{n})^n \cdot \prod_{i=1}^{n} x_i$

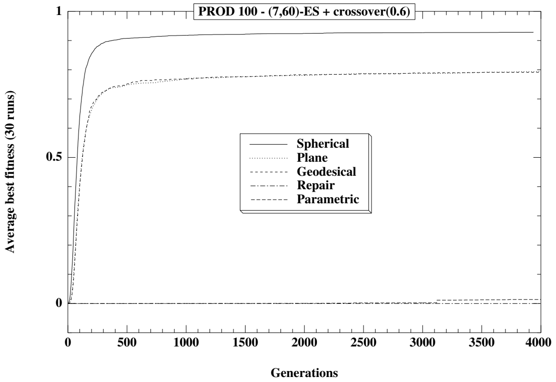
Max. value is 1 at point $(\frac{1}{\sqrt{n}}, \ldots, \frac{1}{\sqrt{n}})$



GA, best parameter settings



(7,60)-ES, $p_c = 0$



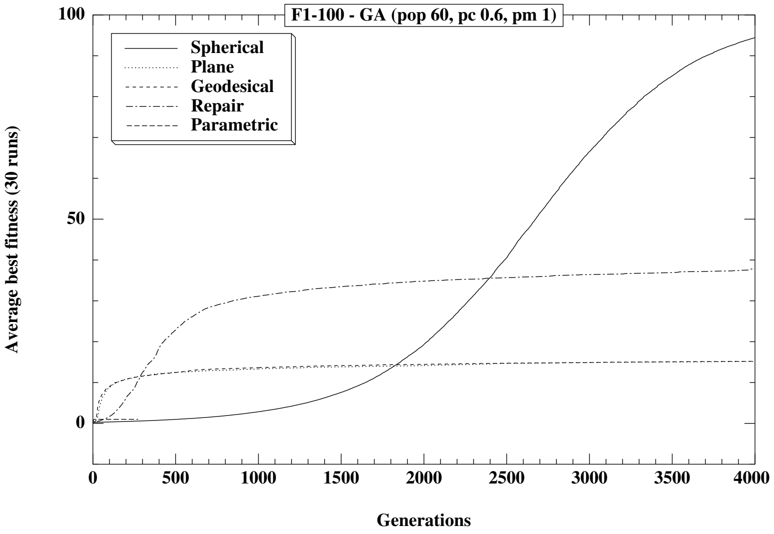
(7,60)-ES, $p_c = 0.6$.

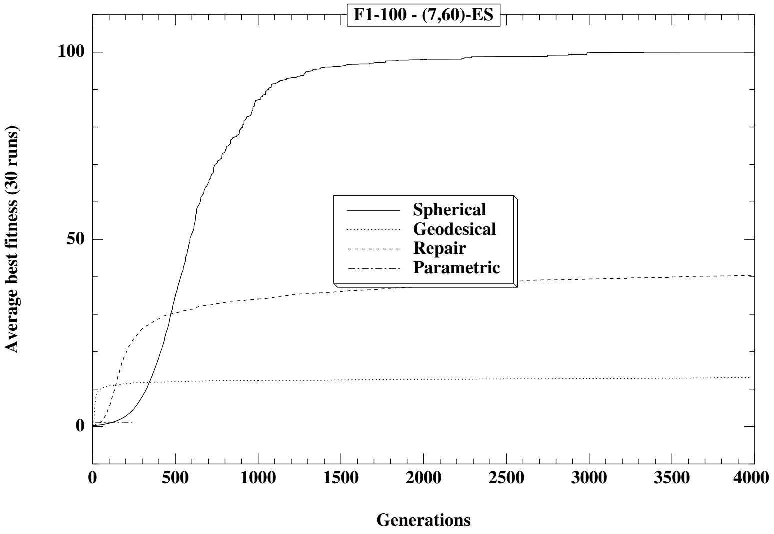
Baluja F_1

Maximize F_1:

$$\begin{cases} F_1(\vec{x}) = \dfrac{100}{0.00001 + \sum_1^n |y_i|} \\ y_i = x_i + y_{i-1} \\ x_i \geq 0 \end{cases}$$

- Amounts to minimize $\displaystyle\sum_1^n (n+1-i)x_i$
- N local maxima at $e_i = (0, \ldots, 0, 1, 0, \ldots, 0)$
  These are not local optima on $\mathbb{R}_+^N$.
- Global maximum at e_n (99.999)
- Tailored for spherical mutation



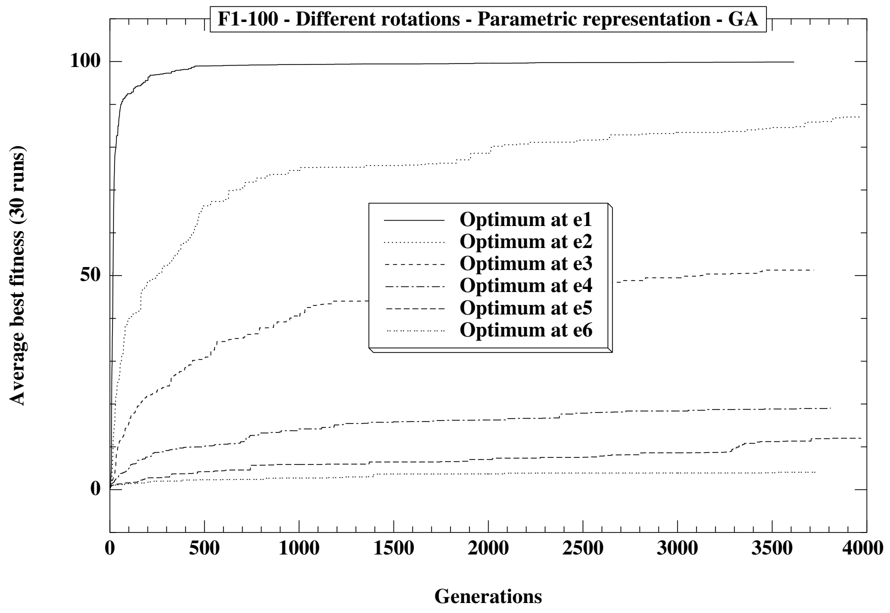


GA, $p_{mut} = 1$ (7,60)-ES

Slightly better without crossover Worse with crossover

Rotated Baluja F_1

F_1 is rotated $\rightarrow$ maximum at e_i



Results for different parametric representations

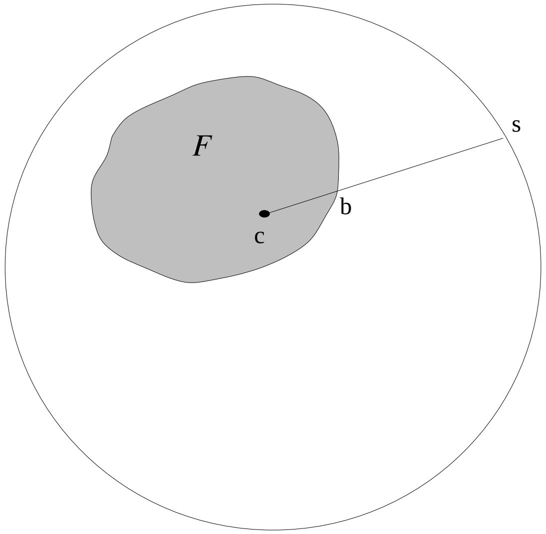
Other trials on rotated F_1:

- Best combination: problem dependent
  (choice of operators, evolution scheme and operator-rates), **but**
- Crossover useful
- Spherical GA almost always best
- Plane $\geq$ geodesical
- Repair almost always lousy

What if you are not on the sphere?

Generalization

Transform $\mathcal{B}$ into (part of) the sphere



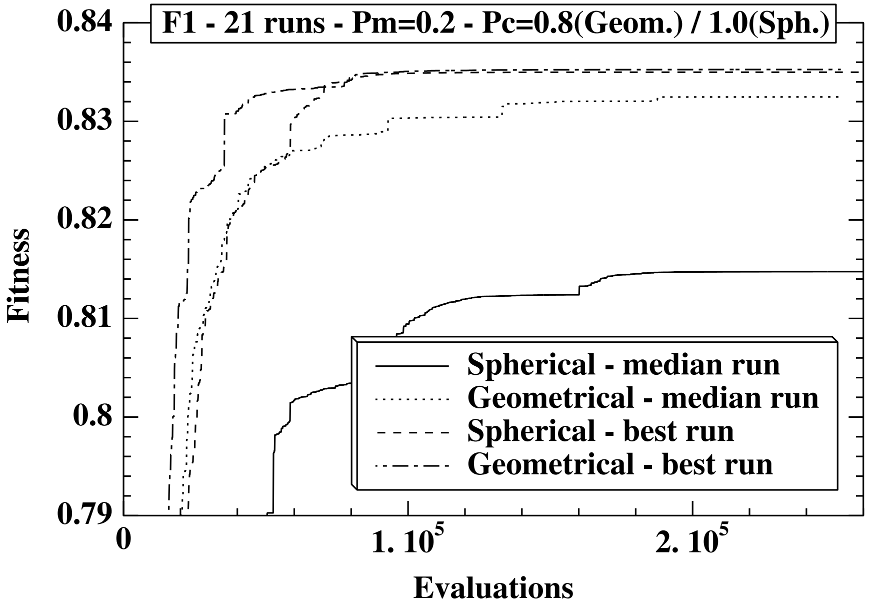
- Genotypes are defined on the sphere – using specific operators
- Phenotypes are the corresponding points of $\mathcal{B}$.
- Easy (though time-consuming) numerical computation (e.g. by bisection) if $\mathcal{F}$ is convex (or star-shaped)

$\rightarrow$ comparison with best specific operators (alg.= AG).

Spherical vs geometrical

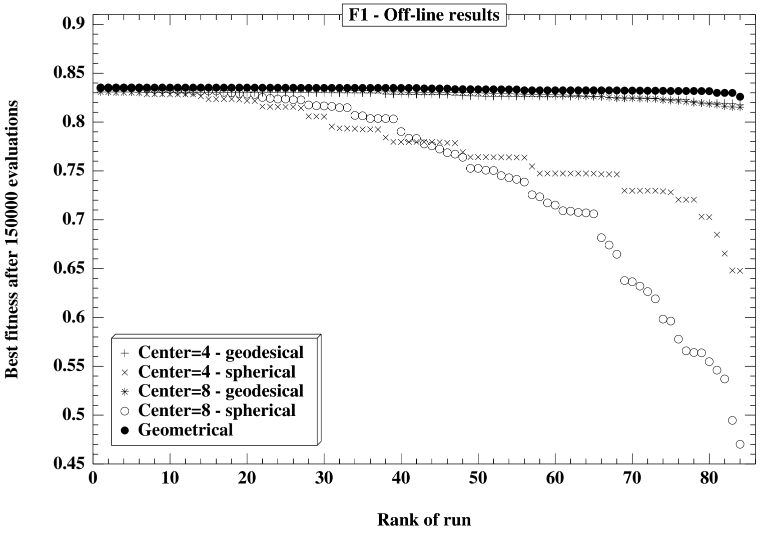
Keane's function on the hyperboloid

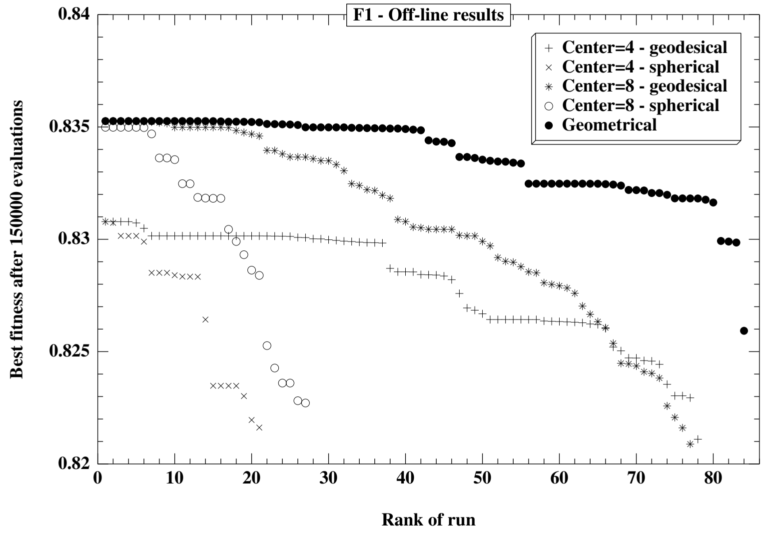
Positive part of the sphere mapped onto the hyperboloid.
Center of mapping: $(8, \ldots, 8)$.



Same order of results for optimal setting of parameters

Spherical vs geometrical: the robustness issue





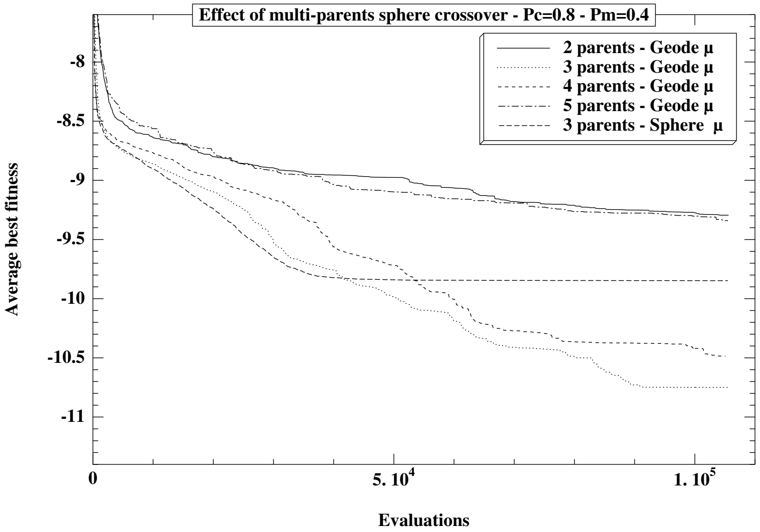
Results of 21 runs for $(P_c, P_m) \in \{0.8, 1.0\} \times \{0.1, 0.2\}$.
Influence of the position of the center of the mapping $((8, \ldots, 8)$ *or* $(4, \ldots, 4))$ *and the type of operators used on the sphere ("spherical" vs geodesical).*

Linear constraints

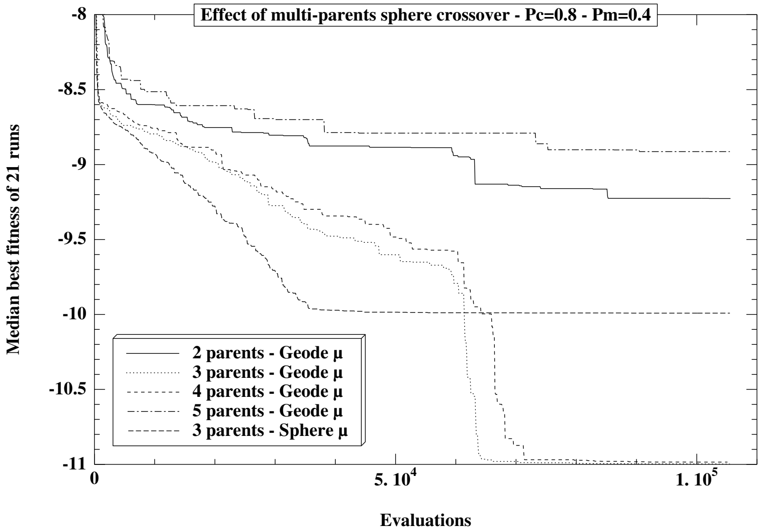
Minimize $f3(x, \overline{Y}) = 6.5x - 0.5x^2 - y_1 - 2y_2 - 3y_3 - 2y_4 - y_5$,
subject to:
$x + 2y_1 + 8y_2 + y_3 + 3y_4 + 5y_5 \leq 16$,
$-8x - 4y_1 - 2y_2 + 2y_3 + 4y_4 - y_5 \leq -1$,
$2x + 0.5y_1 + 0.2y_2 - 3y_3 - y_4 - 4y_5 \leq 24$,
$0.2x + 2y_1 + 0.1y_2 - 4y_3 + 2y_4 + 2y_5 \leq 12$,
$-0.1x - 0.5y_1 + 2y_2 + 5y_3 - 5y_4 + 3y_5 \leq 3$,
$y_3 \leq 1$, $y_4 \leq 1$, and $y_5 \leq 2$,
$x \geq 0$, $y_i \geq 0$, for $1 \leq i \leq 5$.
Optimum $(x, \overline{Y}^*) = (0, 6, 0, 1, 1, 0)$, $f3(x, \overline{Y}^*) = -11$.

Good results of proposed method, but
GENOCOP II one order of magnitude better (precision and speed).

Effect of multi-parents crossover (see [Eiben & al. 97])



Mean values



Median values

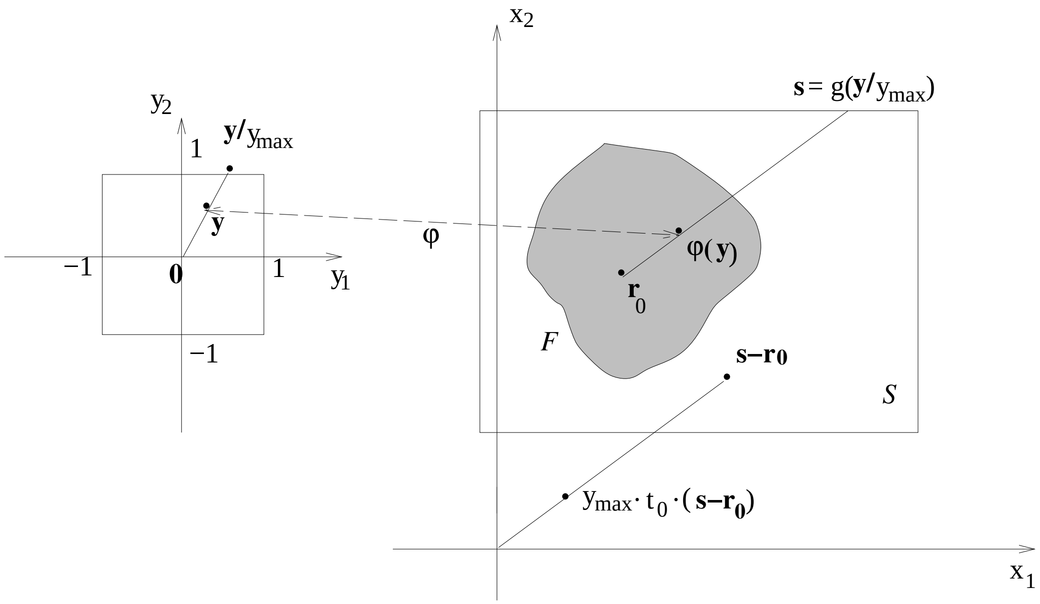
(half the runs are better than that!)

On $F3$, geodesical better than spherical, and multi-parents better than two-parents.

Transforming $\mathcal{F}$

Koziel 97, Koziel & Michalewicz 98

In case of inequality constraints, and convex (or star-shaped) feasible space, you can transform $\mathcal{F}$ into the hypercube $[-1, 1]^n$:

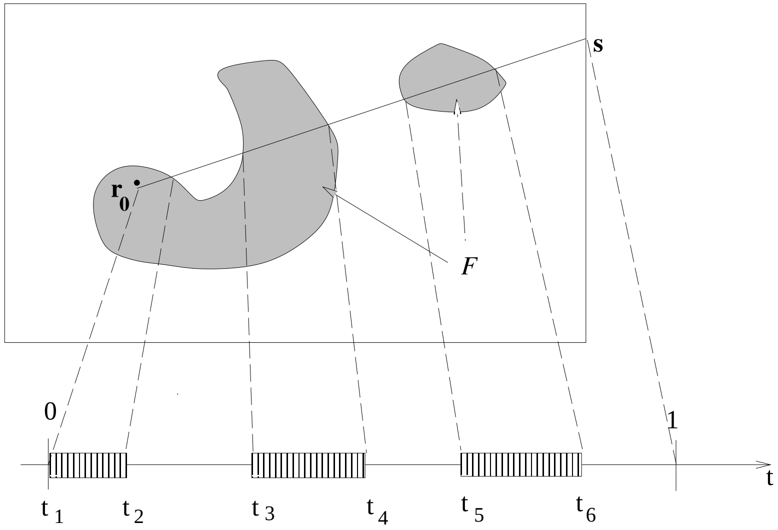


- Search on $[-1, 1]^n$ with standard evolution schemes and operators
- Fitness of $\vec{y} = f(\phi(y))$
- Easy (though time-consuming) numerical computation by bisection.

Results

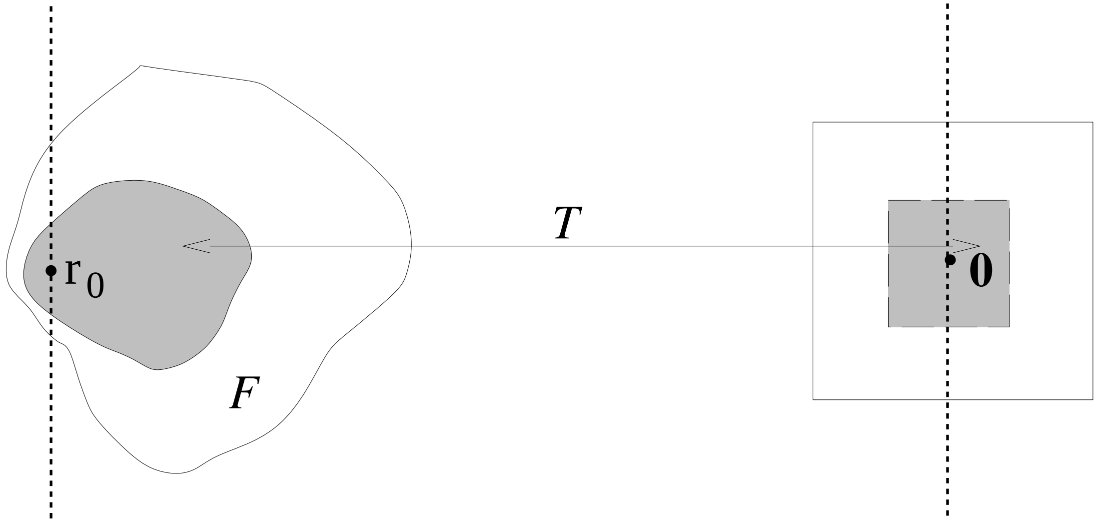
- Excellent results on **all** all test functions (see M&S 96)

- Convexity is a problem:



Theoretically always possible

Which center?



Zoom close to the center

- Inertia center of $\mathcal{F}$ (from random sampling)
- From a priori information
- From a first run of the method

  somewhere between previous center and best-so-far solution

- Adaptive?

Behavioral Memory

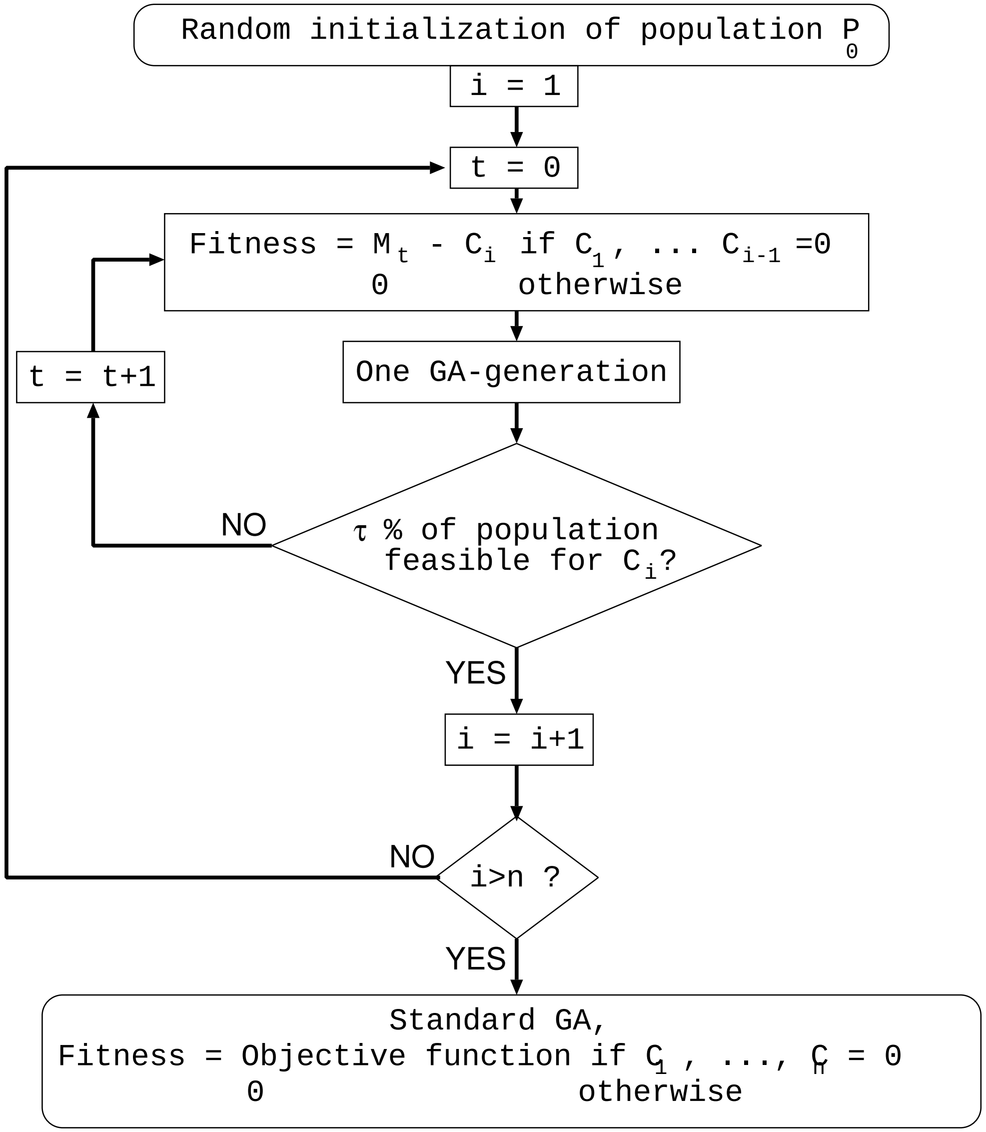
Schoenauer & Xanthakis 93

- $\mathcal{F}$ out of reach
- $\mathcal{F}$ very sparse
- fitness not computable outside $\mathcal{F}$

Evolutionary sampling of $\mathcal{F}$

Evolutionary sampling

- Random initialization
- For each constrain i, from preceding final population, minimize the violation of i
  Give 0 fitness to individuals that violate constraints 1,...,$i-1$, until a proportion τ % of the population becomes feasible for those constraints.
- Optimize the fitness using last population feasible for all constraints



Behavioral Memory constraint handling

Behavioral Memory: function G8

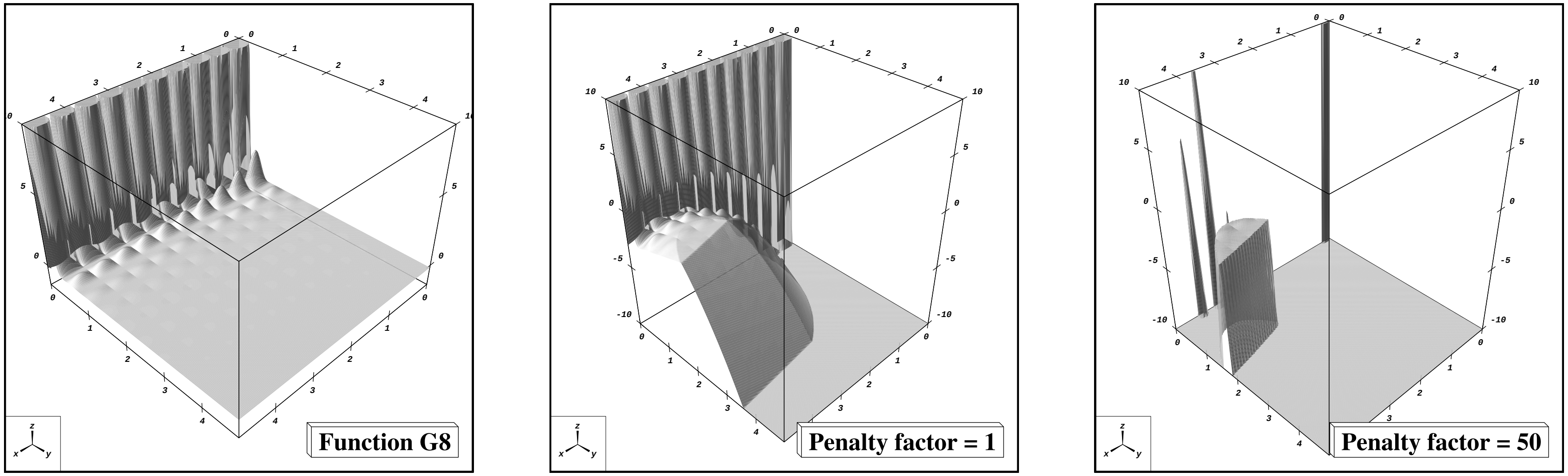
Maximize function

$$G8(\vec{x}) = \frac{\sin^3(2\pi x_1)\cdot\sin(2\pi x_2)}{x^3\cdot(x_1+x_2)},$$
$$0 \leq x_1 \leq 10 \text{ and } 0 \leq x_2 \leq 10.$$

with constraints

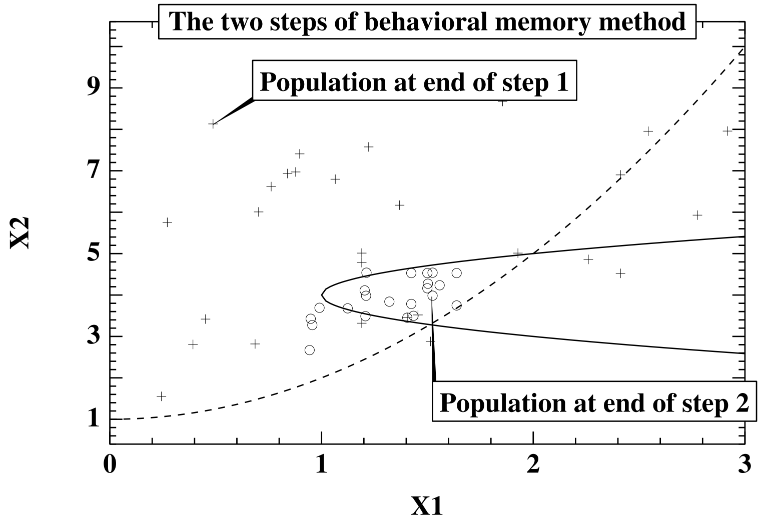
$$c1(\vec{x}) = x_1^2 - x_2 + 1 \leq 0,$$
$$c2(\vec{x}) = 1 - x_1 + (x_2 - 4)^2 \leq 0$$

Effect of penalization factors



Fitness landscapes for different penalization factors

Results



Population after first and second steps

- **Crucial**: preserve genetic diversity

Sharing, or another niching mechanism

- Sensitive to constraint ordering

Multi-criteria approach

Surry & Radcliffe 96

- $m+1$ criteria : f + violation of constraint i
- Adapt techniques from multi-objective optimization

One generation:

- Choose τ_0 and p_{cout} (e.g. 0.5).
- Evaluate (f and all g_i)
- Compute Pareto ranks for the constraints violations
- Select p_{cout} of the parents from fitness, the rest from Pareto rank
- Apply variation operators
- Adjust p_{cout}: if $\tau(t) < \tau_0$, decrease p_{cout} otherwise increase it

Good results on a real-world problem (British gas network optimization).

A sexual approach

Your brain and my beauty

Hinterding and Michalewicz 98

- Uses the selection-seduction paradigm (Ronald 97)

  Select mate based on sexual preferences ot the first
- Select first parent from the fitness
- Choose mate to maximize the number of constraints met by the happy couple

Some good results on some test functions

Projection methods: conclusions

- Useful when penalty methods fail

Or for equality constraints

- Always return feasible solutions
- *Ad hoc* operators $\geq$ generic ones (as foreseen!) for surface operators
- Problem if non-convex and non-star-shaped feasible domain

Perspectives

- Coupling of both projection methods?

  To handle both inequality and equality constraints

- Adaptive centers?
- What operators on the sphere?
- Behavioral Memory to initialize the population?

  When feasible space out of reach

More generally

What algorithm? And what links between evolution scheme and operators

Penalty methods

Minimize $\mathcal{G}(x) = f(x) + Penal(x)$

where $Penal(X) \geq 0$, and

$Penal(X) = 0$ iff $x \in \mathcal{F}$

- Back to unconstrained optimization

No need to change our favorite algorithm

Works with **any** optimization algorithm

- Choice of penalty term?

Types of penalty

- Death penalty (Bäck et al 91):
  $Penal(X) = +\infty$ if $x \notin F$

  Infeasible individuals are eliminated

  Unstable

- Number of violated constraints

  Only if many constraints

  e.g. SAT problems

- Measure of violation
  Typically

$$Penal(x) = \sum_{i=1}^{i=m} \alpha_i g_i(x)^+ + \sum_{i=m+1}^{i=q} \alpha_i |g_i(x)|$$

  Problem: choice of α_i ?

In evolutionary constrained optimization, infeasible points should be penalized by the smallest amount so that the solution of the constrained problem is the minimum of the penalized function.
Richardson et al. 1989

Static penalties

α_i are fixed by the user

- α_i too small : the minimum of $\mathcal{G}$ can be infeasible
- α_i too large : search is restricted to $\mathcal{F}$

No shortcut across infeasible region (see death penalty)

Specific static penalties

Homaifar et al. - 94

$$Penal(x) = R(g^+(x))\ [g^+(x)]^2$$

with $\quad R(g^+(x)) = R_i \quad$ si $\quad G_{i-1} < g^+(x) \leq G_i \quad i = 1, l$

$\Longrightarrow m(2l+1)$ sensitive parameters to tune!

Penalties based on the superiority of feasible points

Any feasible point has a better fitness than any infeasible point

- Powell and Skolnick 93

$$\text{If } x \in \mathcal{F}, Penal(x) = r\ g^+(x)\ +\ \theta(t)$$

$$\theta(t) = \quad \max_{x \in \mathcal{F}}\{f(x)\} \quad - \quad \min_{x \in \mathcal{S}-\mathcal{F}}\{f(x) + rg^+(x)\}\}$$

- Deb 99

$$\mathcal{G}(x) = \begin{cases} f(x) & \text{if } x \in \mathcal{F} \\ r\ g^+(x)\ +\ \max_{x \in \mathcal{F}}\{f(x)\} & \text{otherwise} \end{cases}$$

Infeasible points are only pushed toward feasibility

Dynamic penalties (exogenous)

α is modified according to a prescribed schedule.

Generally increasing

- Annealing penalty – Michalewicz and Attia 94

$$Penal(x) = \frac{1}{2\tau(t)}[g^+(x)]^2$$

with $\tau(t) \; = \; \alpha\tau(t-1) \qquad \alpha \in [0,1]$

- Joines and Houck 94

$$Penal(x,t) = (Ct)^\beta [g + (x)]^\gamma$$

- Kane and Schoenauer 96

$$alpha(t) = alpha_0 + (alpha_F - alpha_0)\left(\frac{t}{T}\right)^\beta$$

where T is the maximal number of generations

But

- Convergence is very sensitive to parameters.
- Need for an automatic tuning

from initial population???

Adaptive penalties (endogenous)

Tune penalty according to past evolution

- Bean and Hadj-Alouane 92

$$\alpha(t+1) = \begin{cases} \frac{\alpha(t)}{\beta_1} & \text{if } x_b(t), \ldots x_b(t-k) \in \mathcal{F} \\ \beta_2 \alpha(t) & \text{if } x_b(t), \ldots x_b(t-k) \notin \mathcal{F} \\ \alpha(t) & \text{otherwise} \end{cases}$$

β_1 , $\beta_2 > 1$, $x_b(t)$ best of generation t.

- Smith and Tate 93
  Adaptive/dynamic penalty based on best feasible and best overall fitnesses seen so far.

But

Only considers the past best individual
No guarantee of feasibility!

"Segregated" GA   LeRiche et al. 97

Uses two values for α, one "too large" and one "too small" to try to explore both sides of the boundary of $\mathcal{F}$.

1. Choose $\alpha^0 < \alpha^1$
2. Initialize the population of size $2 * P$
   Evaluate f and all g_i for all individuals
3. Make 2 lists, one sorted using $f + \alpha^0 g$, the other using $f + \alpha^1 g$
4. Do P times
   Take the best one not yet chosen alternatively in each list
5. Do one "standard" generation with those P individuals
   Ranking selection, variation operators
6. Back to 3 with P parents and P offspring   (P+P)-ES replacement

Only tested on the optimization of composite materials

Methodology

- No free lunch (again!)
- Always start using penalization

  Static, then exogenous or endogenous

- Cheap constraints?

  Projection on the hypercube

- Analytical constraints?

  Specific operators

- Feasible space hard to sample?

  Behavioral Memory