Evolutionary Topological Optimum Design

Marc Schoenauer
Equipe-Projet TAO
INRIA Saclay Île-deFrance

Agenda

Evolutionary Algorithms

- Background — Optimization
- The algorithm — Biological paradigm
- Two viewpoints
  - Evolution engine — Artificial Darwinism
  - Variation operators — Crossover and mutation
- Critical issues

Topological Optimum Design

Agenda

Evolutionary Algorithms

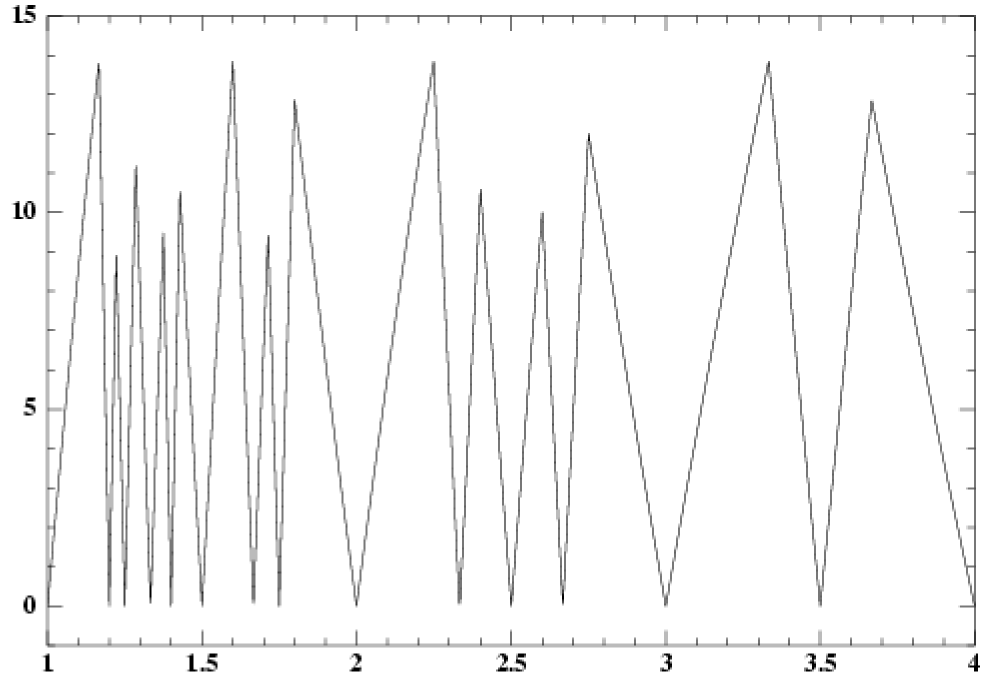
- Background — Optimization
- The algorithm — Biological paradigm
- Two viewpoints
  - Evolution engine — Artificial Darwinism
  - Variation operators — Crossover and mutation
- Critical issues

Topological Optimum Design

Rough Objective Function

L. Taieb, CMAP and Thomson

- Search Space: Continuous parameters   Interferometers
- Goal: Maximize tolerance, preserving accuracy

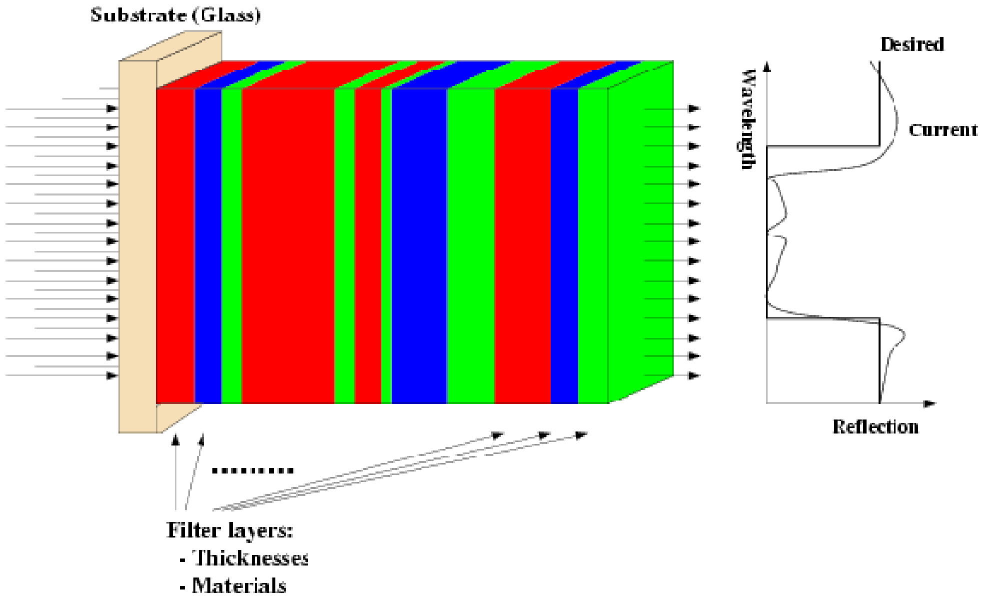


Objective function – 3 antennas

Mixed Search Space

Schutz & Bäck, Dortmund U. - Martin et al., Optique PVI & CMAP

- Search Space: lists of pairs (material, thickness)
- Goal: Fit the target response profile

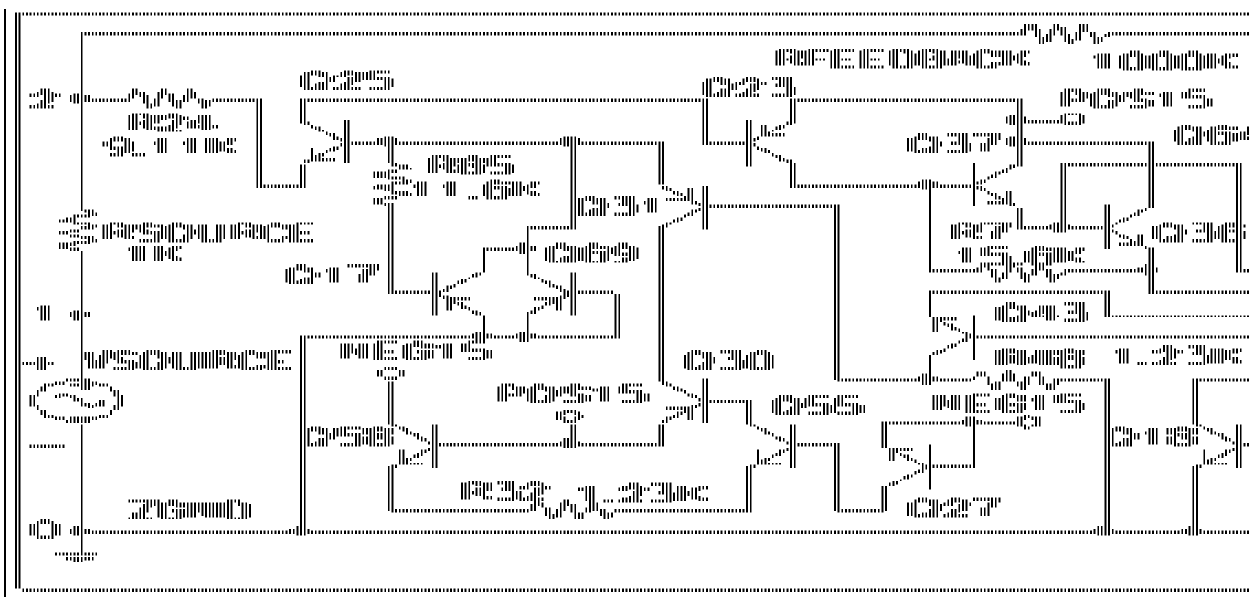


High and Low frequency filter

Digital circuits

Koza et al., Genetic Programming Inc. & Stanford

- Search Space: Valued graphs
- Goal: Target functionalities



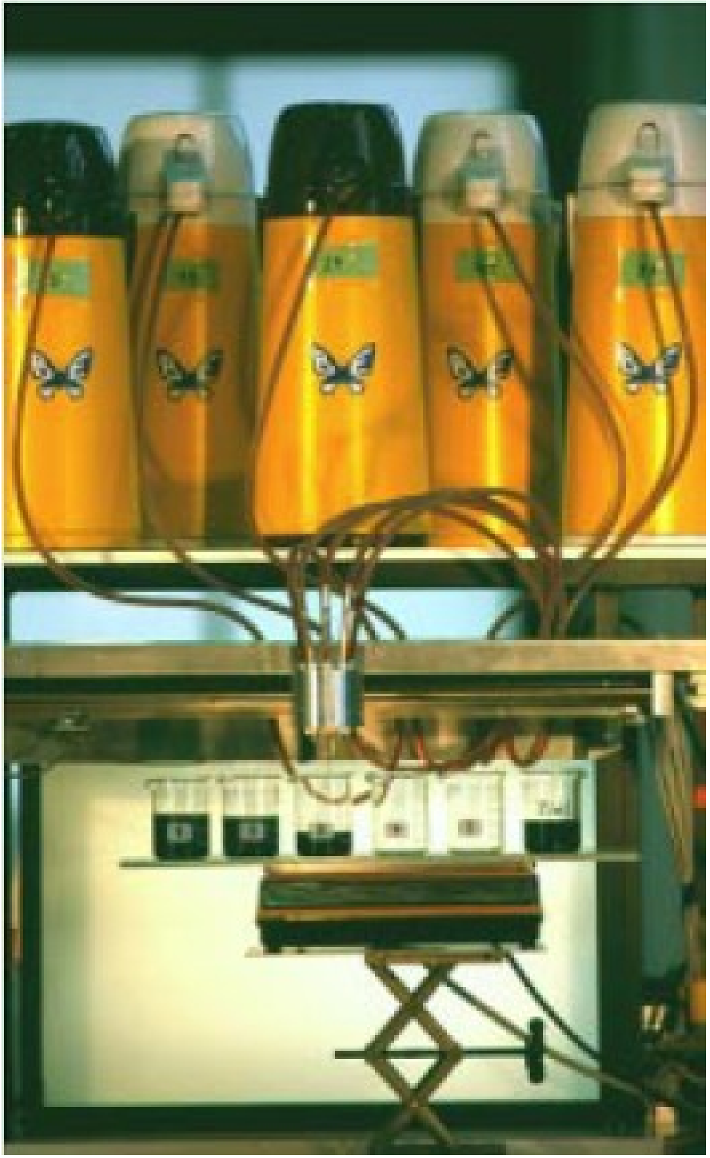
Evolved cubic root extractor

Non-computable Objective Function

Herdy et al., Technische Univ. Berlin

- Search Space: Blend proportions
- Goal: Find a target flavor

Expert knowledge



Optimization Algorithms

- Enumerative methods
- Gradient-based algorithms
- Hill-Climbing
- Stochastic methods

Meta-heuristiques

Comparison issues

- Nature of search space
- Smoothness of objective (constraints)
- Local vs global search

Agenda

Evolutionary Algorithms

- Background — Optimization
- The algorithm — Biological paradigm
- Two viewpoints
  - Evolution engine — Artificial Darwinism
  - Variation operators — Crossover and mutation
- Critical issues

Topological Optimum Design

From hill-climbing to meta-heuristics (1)

Simple Hill-Climbing

- Choose X_0 uniformly in Ω, and compute $F(X_0)$
- Loop — e.g., until no improvement
  - $y = \mathit{ArgMax}\{F(x) ; x \in \mathcal{N}(X_t)\}$ — neighborhood $\mathcal{N}$
  - Compute $F(y)$
  - If $F(y) > F(X_t)$ then $X_{t+1} = y$ — assume maximization
    else $X_{t+1} = X_t$
  - $t=t+1$

Neighborhoods and EVE dilemma

Size matters

- $\mathcal{N}(X_t) = \Omega \rightarrow$ Monte-Carlo — Memoryless exploration
- $\mathcal{N}(X_t) =$ Closest neighbors(X_t) — Purely local exploitation

Enhancements

- Generalize neighborhoods — probabiliy distributions
- Relax selection — accept worse points
- Population-based algorithms

From hill-climbing to meta-heuristics (2)

Stochastic Hill-Climbing

- Choose X_0 uniformly in Ω, and compute $F(X_0)$
- Loop — e.g., until no improvement
  - $y = U[\mathcal{N}(X_t)]$ — uniform choice
  - Compute $F(y)$
  - If $F(y) > F(X_t)$ then $X_{t+1} = y$ — acceptation
    else $X_{t+1} = X_t$
  - $t=t+1$

From hill-climbing to meta-heuristics (3)

Stochastic Local(?) Search

- Choose X_0 uniformly in Ω, and compute $F(X_0)$
- Loop — e.g., until no improvement
  - y = Move(X_t) — operator==distribution
  - Compute $F(y)$
  - If $F(y) > F(X_t)$ then $X_{t+1} = y$ — acceptation
    else $X_{t+1} = X_t$
  - $t=t+1$

From hill-climbing to meta-heuristics (4)

Stochastic Search (e.g. Simulated Annealing)

- Choose X_0 uniformly in Ω, and compute $F(X_0)$
- Loop — e.g., until no improvement
  - $y = Move(X_t)$ — operator==distribution
  - Compute $F(y)$
  - $X_{t+1} = Select(y, X_t)$ — selection
  - $t=t+1$

From hill-climbing to meta-heuristics (5)

(1+λ)-Evolution Strategy

- Choose X_0 uniformly in Ω, and compute $F(X_0)$
- Loop — e.g., until no improvement
  - For i=1, ..., λ
    - y_i = Move(X_t) — operator==distribution
    - Compute $F(y_i)$
  - X_{t+1} = Select (y_1, ..., y_λ , X_t) — selection
  - t=t+1

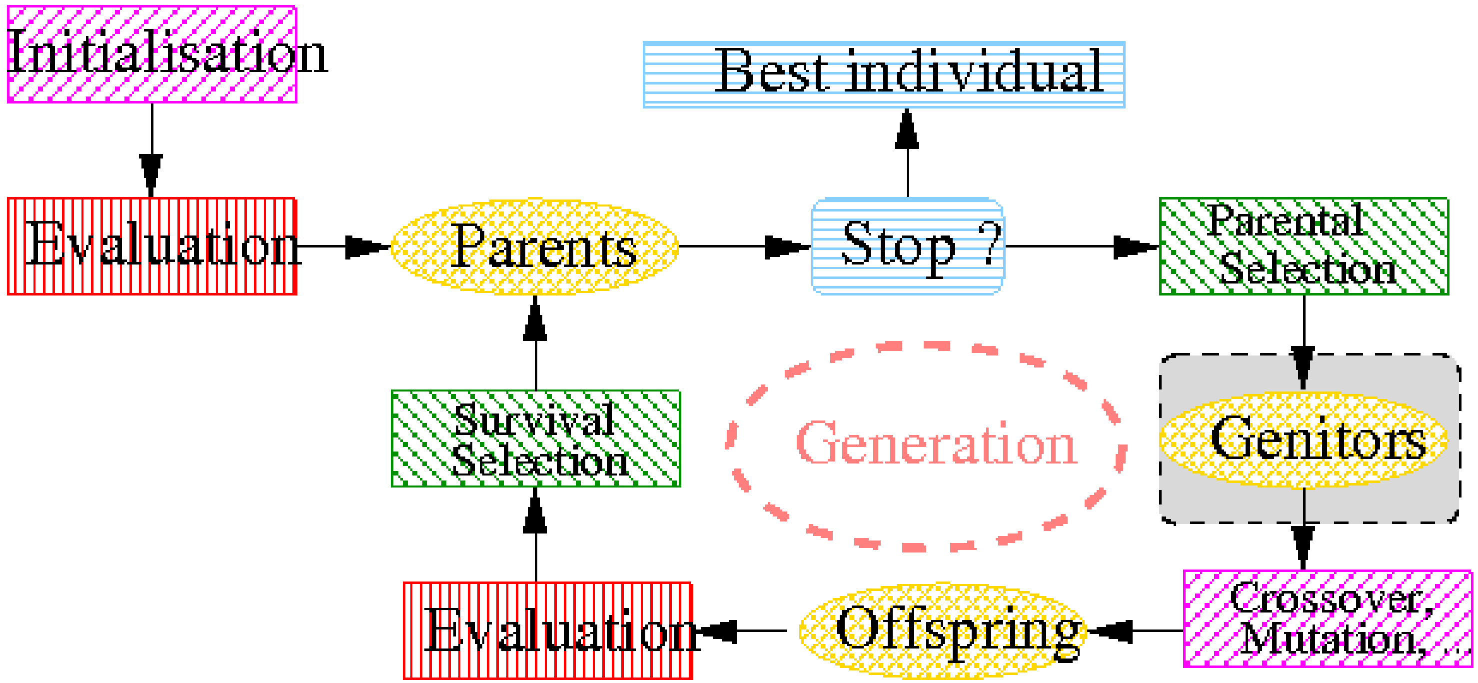
Evolutionary Paradigm

- Natural selection — bias toward fittest individuals
- Blind variations — Parents → offspring by undirected variations (i.e. independent of fitness)
- Individual "Objective": survival and reproduction
- Result: adapted species — e.g. resistant bacteria

But

- Inspiration
- Explanation
- **Not justification**

The Skeleton



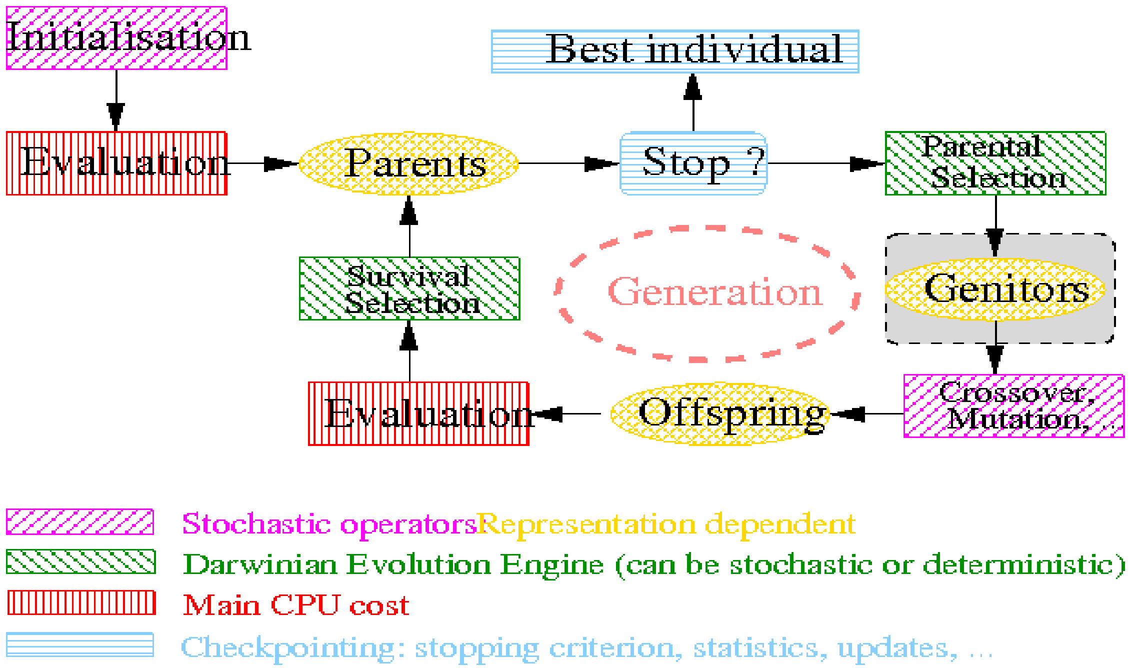
Stochastic operators Representation dependent

Darwinian Evolution Engine (can be stochastic or deterministic)

Main CPU cost

Checkpointing: stopping criterion, statistics, updates, ...

Two orthogonal points of view



- Artificial Darwinism (selection steps) only depend on **fitness**
- Initialization and variation operators only depend on the **representation** (i.e. the search space)

Agenda

Evolutionary Algorithms

- Background — Optimization
- The algorithm — Biological paradigm
- Two viewpoints
  - Evolution engine — Artificial Darwinism
  - Variation operators — Crossover and mutation
- Critical issues

Topological Optimum Design

Artificial Darwinism

Two selection steps

- Parental selection can select an individual multiple times
- Survival selection selects or not each individual

Issues

- Bias toward fitter individual
  - Too large bias → pure local search — Premature convergence
  - Too small bias → random walk — No convergence
- Can be deterministic or stochastic

Tournament Selection

Stochastic selections

- Deterministic tournament - size T
  - Choose T individuals uniformly
  - Return best
- Stochastic tournament – probability t ∈ [0.5,1]
  - Choose 2 individuals uniformly
  - Return best with probability t (worse otherwise)

Advantages

- Comparison-based → invariance properties
- Easy parameterization from t=0.5 to T=P

Deterministic Survival Selection

Evolution Strategies: μ parents, λ offspring (historical)

- (μ+λ)-ES: the μ best of μ old parents + λ offspring become next parents
  - Pratical robustness
  - Premature convergence
- (μ,λ)-ES: the μ best of λ offspring become next parents
  - Can lose best individuals
  - Better exploration

Agenda

Evolutionary Algorithms

- Background — Optimization
- The algorithm — Biological paradigm
- Two viewpoints
  - Evolution engine — Artificial Darwinism
  - Variation operators — Crossover and mutation
- Critical issues

Topological Optimum Design

Variation Operators

Crossover: Two (or more) parents -> one offspring

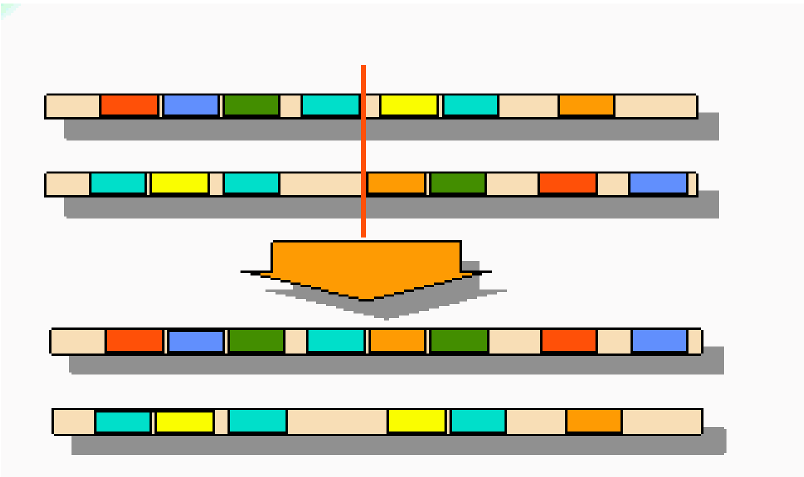
- Exchange of information — 'linearity' of fitness function
- Start of evolution: exploration
- Close to convergence: exploitation

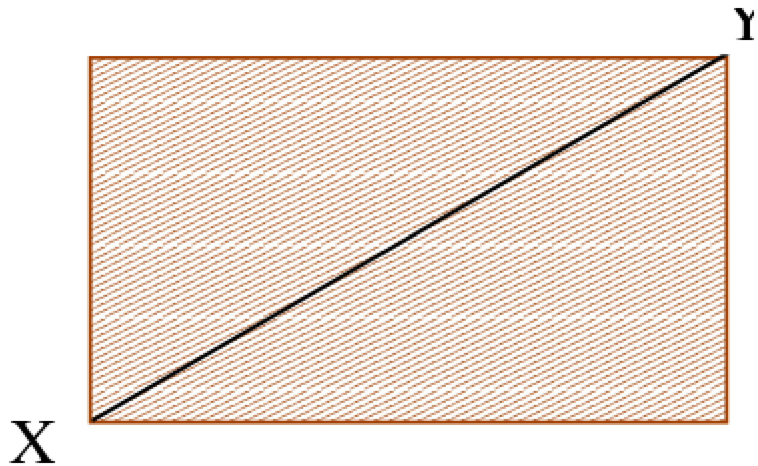
Mutation: One parent → one offspring

- Reintroduces diversity
- Ergodicity
- "Strong Causality" — 'continuity' of fitness function

Crossover

Standard examples





Exchange of 'genes'

Crossover of real parameters

Five parents for a surrealist offpsring

La foule subjuguée boira ses paroles enflammées

Ce plat exquis enchanta leurs papilles expertes

L'aube aux doigts de roses se leva sur un jour nouveau

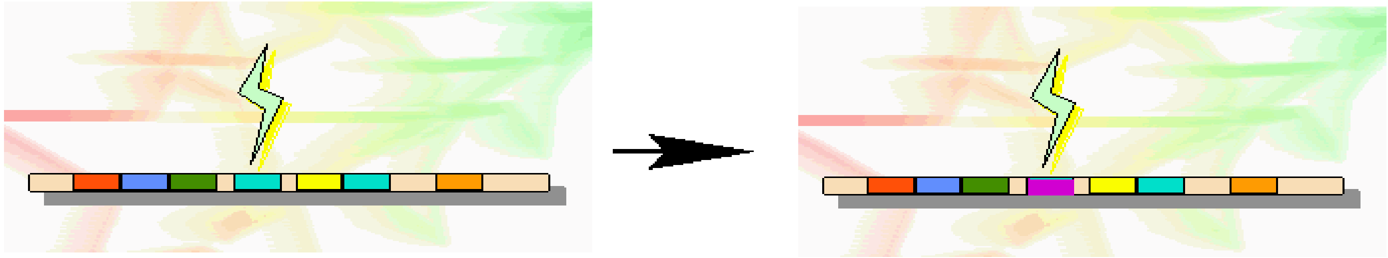
Le cadavre sanguinolent encombrait la police nationale

Les coureurs assoiffés se jetèrent sur le vin pourtant mauvais

Mutation

Standard examples

- 'Gene' mutation



- Adding Gaussian noise to real-valued parameters

A surrealistic example

La terre est comme un orange bleue

La terre est bleue comme une orange

Gaussian mutations

Gaussian mutation

$$X \rightarrow X + \sigma \, \mathcal{N}(0, C)$$

- $\sigma > 0$ mutation step-size
- C covariance matrix (symmetric definite positive)

Adaptation of σ and C

- According to history of evolution: favor directions and step-size that produced fitness improvements
- → **CMA-ES**, state-of-the-art algorithm

Agenda

Evolutionary Algorithms

- Background — Optimization
- The algorithm — Biological paradigm
- Two viewpoints
  - Evolution engine — Artificial Darwinism
  - Variation operators — Crossover and mutation
- Critical issues

Topological Optimum Design

Genotype vs phenotype

- Potential solutions are represented (encoded) in the genotype space, where evolution happens
- They are decoded back into the phenotype space for evaluation
- The same phenotype space can be encoded in several genotype space
- Find the best representation, and you're half way to the solution

Critical Issues

- No Free Lunch Theorem
- Success criterion : Design vs Production
  - At least once an excellent solution
  - On average a good-enough solution
- Do not draw any conclusion from a single run!
- A population, not an individual — Diversity is critical
- **E**xploration **v**s **E**xploitation dilemma
- No strong theoretical results (yet)
  but lessons from many successful applications

Agenda

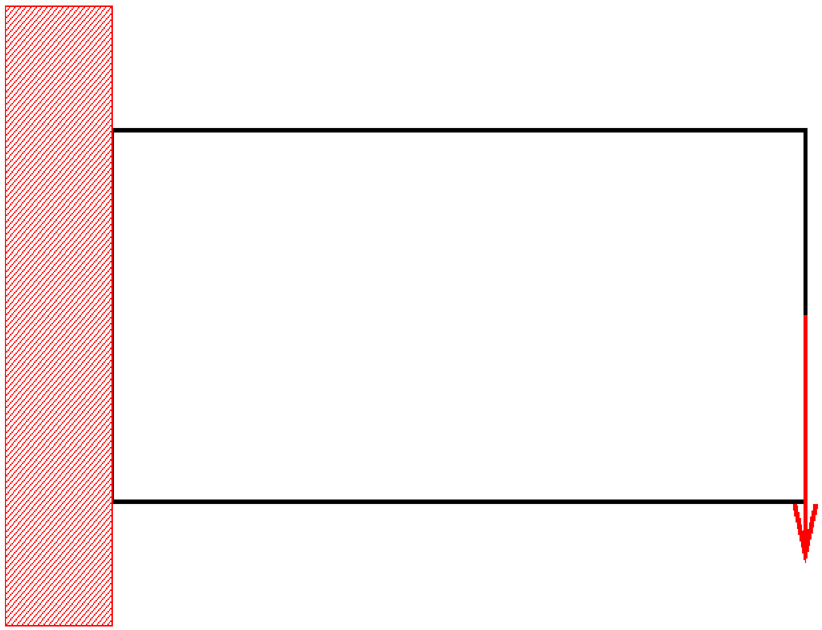
Evolutionary Algorithms

Topological Optimum Design

- The fitness function
- The bitarray representation
- The Voronoi representation
- Multi-objective optimization
- Modularity and Scalability

Sample problem

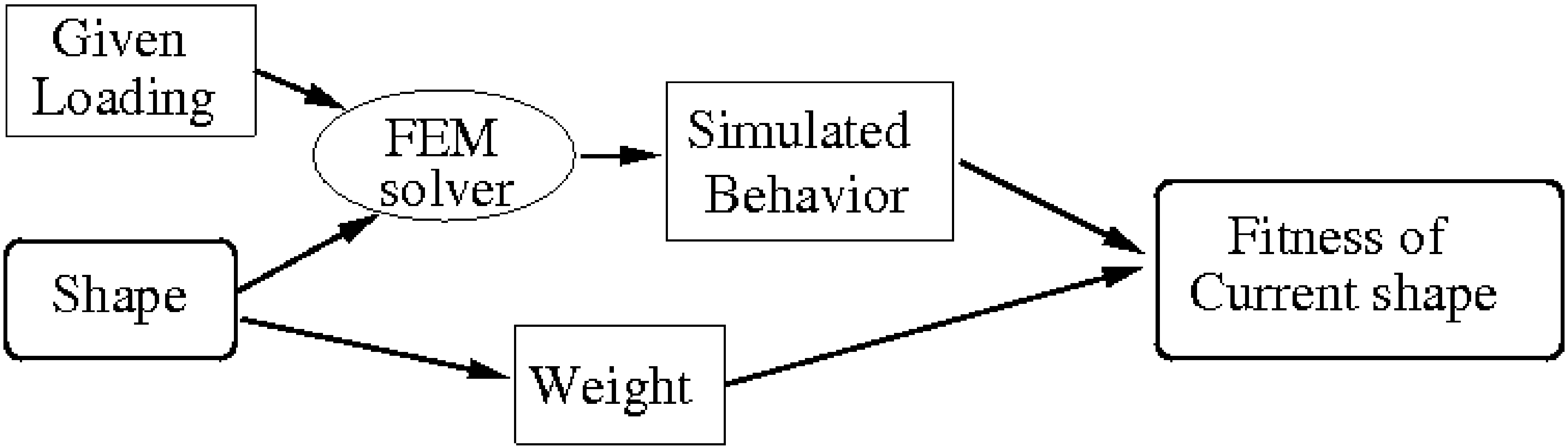
- Find a shape in a given design domain
- Of minimal weight
- With constraints on the mechanical behavior



Example: The cantilever problem, bounds on the maximal displacement

Evolutionary Approach

Only requires a direct solver



What fitness?

What representation?

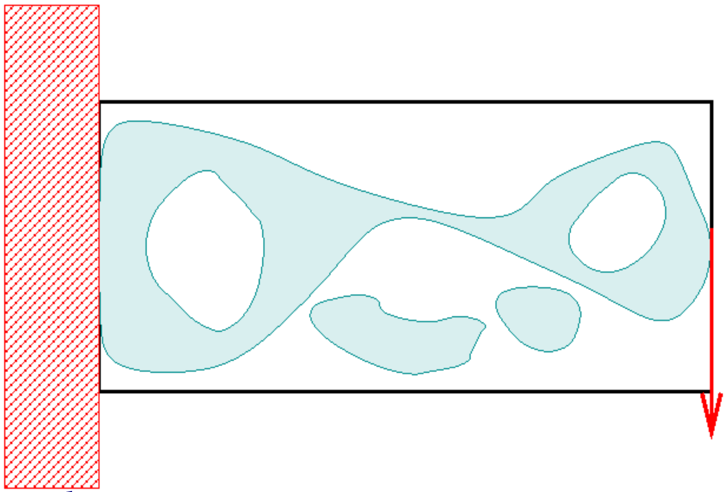
Agenda

Evolutionary Algorithms

Topological Optimum Design

- The fitness function
- The bitarray representation
- The Voronoi representation
- Multi-objective optimization
- Modularity

Fitness function



- A shape can be non-viable
  - Fitness = +∞
- Only connected parts are useful
  - Slightly penalize unconnected parts

Problem

$$\text{Min } (W_{connected} + \varepsilon\, W_{unconnected})$$

$$\text{with } D^i_{max} \leq D^i_{lim} \text{ for each loading } i$$

Constraint handling

Penalization

Minimize $W_{connected} + \varepsilon\, W_{unconnected} + \sum_i \alpha_i\, (D^i_{max} - D^i_{lim})^+$

Choice of α_i?

Fixed penalty

- Too small: optimum unfeasible
- Too large: no exploration of unfeasible regions

Dynamic penalty

- Small at beginning of evolution, large in the end
- Difficult to correctly tune

Adaptive penalty

Penalty changes every generation:

τ(t): proportion of feasible individuals at generation t

$$\alpha(t+1) = \begin{cases} \frac{\alpha(t)}{\beta} & \text{if } \tau(t) > \tau_0 \\ \beta\alpha(t) & \text{if } \tau(t) < \tau_0 \\ \alpha(t) & \text{otherwise} \end{cases}$$

τ0 given threshold, typically 50%

- Based on the current state of the search
- Does not guarantee feasibility
- Searches the neighborhood of the feasible region

Agenda

Evolutionary Algorithms

Topological Optimum Design

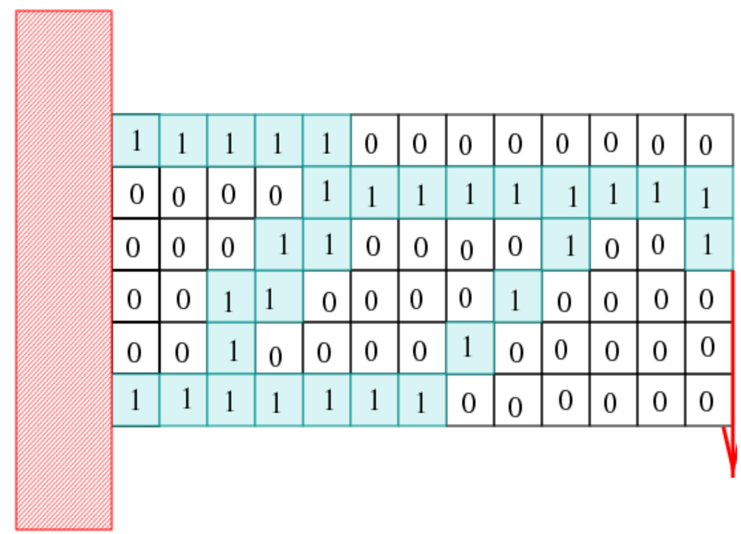
- The fitness function
- The bitarray representation
- The Voronoi representation
- Multi-objective optimization
- Modularity and Scalability

Representation issues

- Search space: bi-partitions of the design domain
  - with some regularity
- Fitness computed using a Finite Element solver
  - Need to mesh all shapes
- Re-meshing introduces numerical errors
  - use the same mesh for the whole population

Bitarrays

- Given a mesh of the whole design domain,
- An element can be made of material (1) or void (0)



- Natural from FE point of view
- Used in all pioneering works

The **complexity** of the representation is that of the mesh

Bitarrays ...

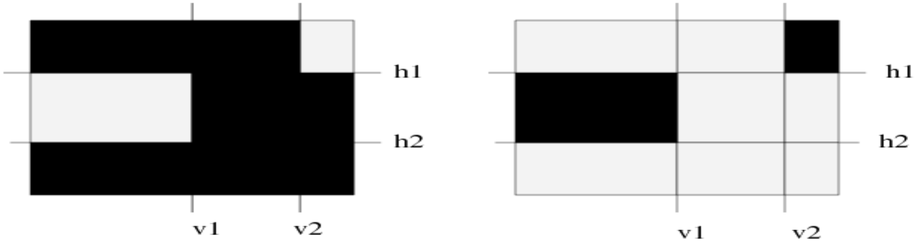
- … are not bitstrings,
- even though an n by m array is formally equivalent to an n.m bitstring.
- Using standard bistring crossover operators introduces a geometrical bias

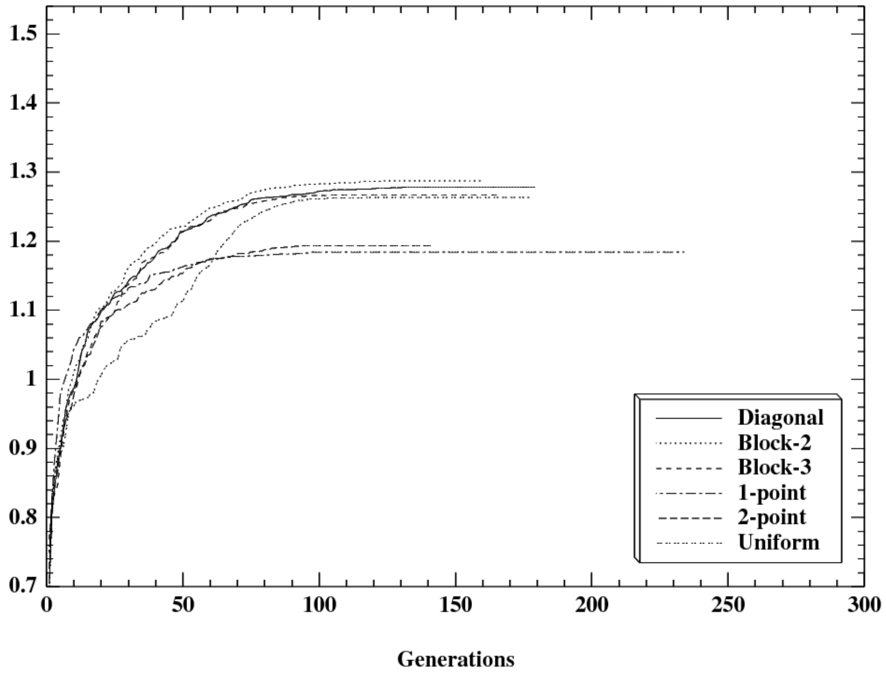
1-point crossover   2-point crossover

Specific 2D crossover

- Diagonal-crossover

- Bloc-crossover





Sample experimental results

Mutation

- No geometrical bias for the standard bit-flip mutation
- But difficulties for adjusting the final bits

Problem-specific mutation

- Start with standard mutation
- As evolution proceeds, increase the probability to mutate the border elements

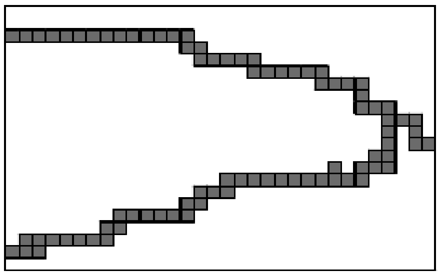
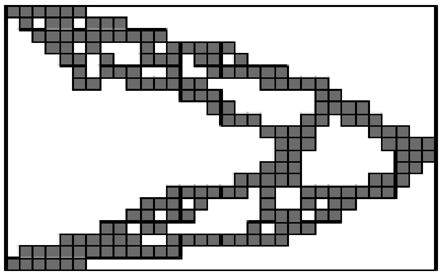
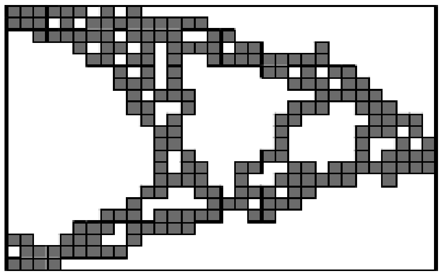
Bitarrays: results

C. Kane, 1997

Experimental conditions

- Population size 125
- Block crossover with probability 0.6
- Mutation with probability 0.2
- Stop after 1000 generations
- Around 80 000 FE computations

Linear elasticity

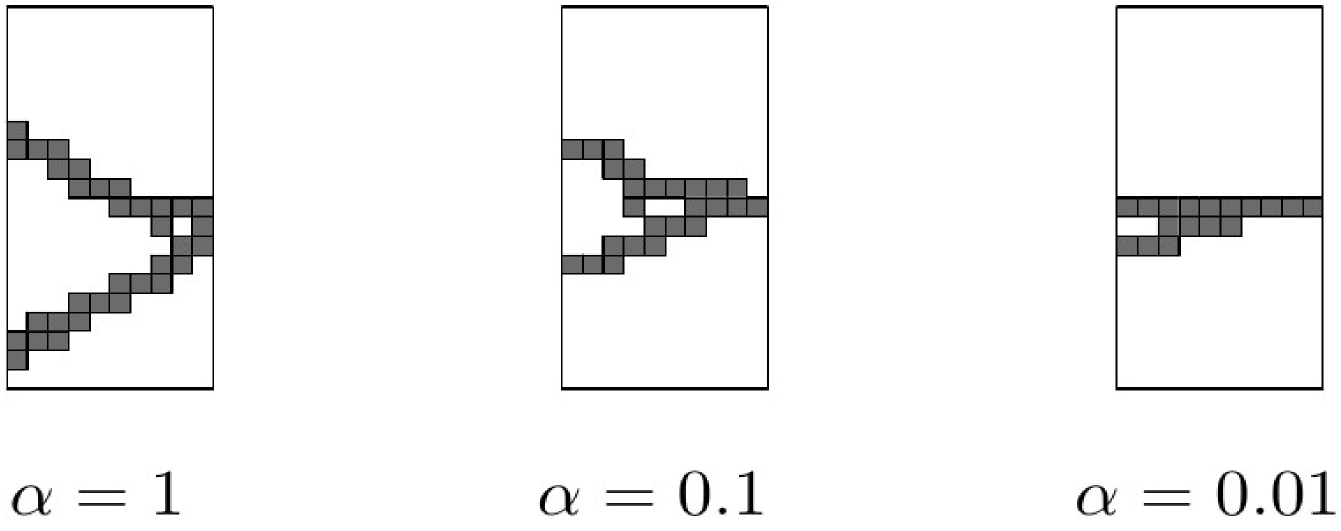
			
D_{Lim}	2.463	0.1293	0.133
D_{Max}	2.439	0.129286	0.133
$Area$	0.2075	0.4675	0.49

Typical results for different values of D_{lim}

Compliance minimization

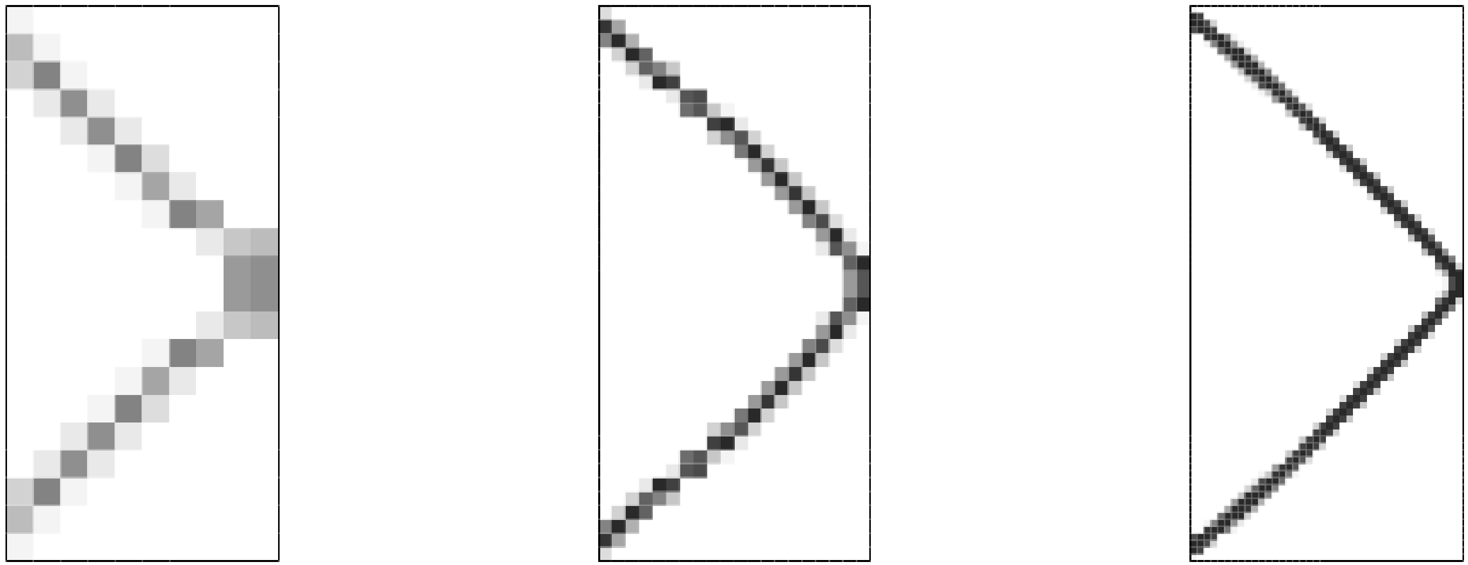
Homogenization minimizes the compliance = ∫Fu

$$Fitness_{compliance} = \frac{1}{Area + \alpha C}$$



Evolutionary optimization of the compliance for different values of α

Homogenization vs EAs



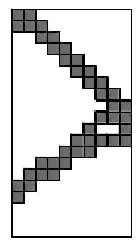
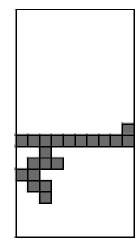
10×20 20×40 40×80

Compliance optimization by homogenization for $\alpha = 1$

- EAs more flexible
- But 2 orders of magnitude slower!

Nonlinear elasticity

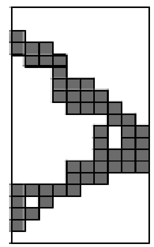
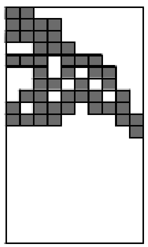
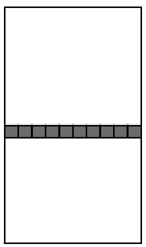
EAs only need a solver for the direct problem: can adapt to any mechanical model (e.g. large strains)

		
D_{Max}	0.022607	0.0199
σ_{Max}	0.076	0.77
$Area$	0.41	0.20
	(a): Small strains.	(b): Large strains.

Disastrous results F = 0.009 and D_{Lim} = 0.02285

Nonlinear elasticity revisited

$$\min\left[Area + \alpha(D_{Max} - D_{Lim})^+ + \beta(\sigma_{Max} - \sigma_{Lim})^+\right]$$

			
F	0.009	0.018	0.09
D_{Lim}	0.22856	0.457	2.2856
σ_{Lim}	0.53	1.0622	5.3
D_{Max}	0.2143	0.4504	1.687
σ_{Max}	0.550	0.9835	4.379
$Area$	0.21	0.47	0.1

Optimal results for F/FLim = Cst

Bitarrays: Conclusions

EAs are flexible

- Any mechanical model — e.g. large strains
- Loading on the unknown boundary — not shown

But

- Representation complexity = size of the mesh
- Accurate results require fine mesh — not to mention 3D
- Empirical and theoretical results suggest that pop. size should be proportional to number of bits

Need mesh-independent representations

Agenda

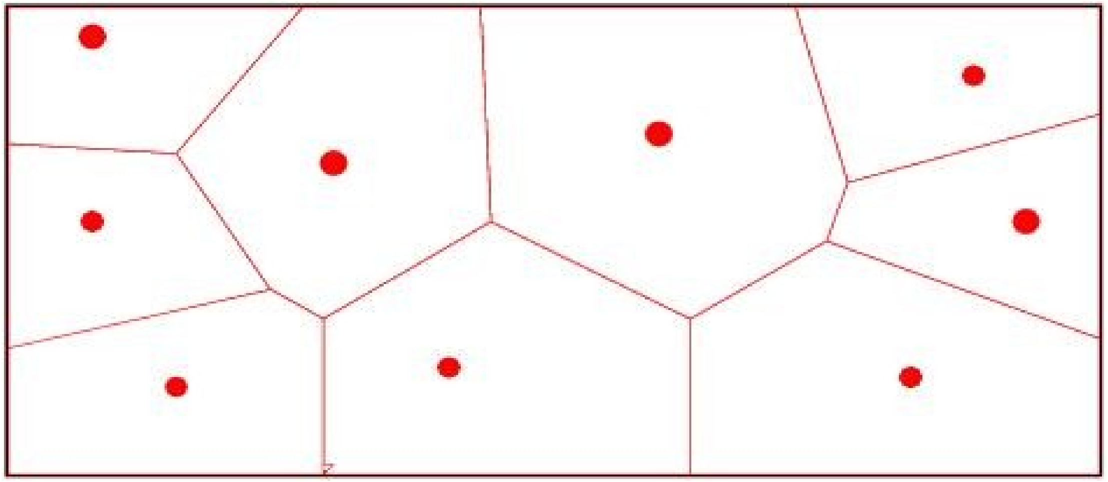
Evolutionary Algorithms

Topological Optimum Design

- The fitness function
- The bitarray representation
- The Voronoi representation
- Multi-objective optimization
- Modularity and Scalability

Diagrammes de Voronoi

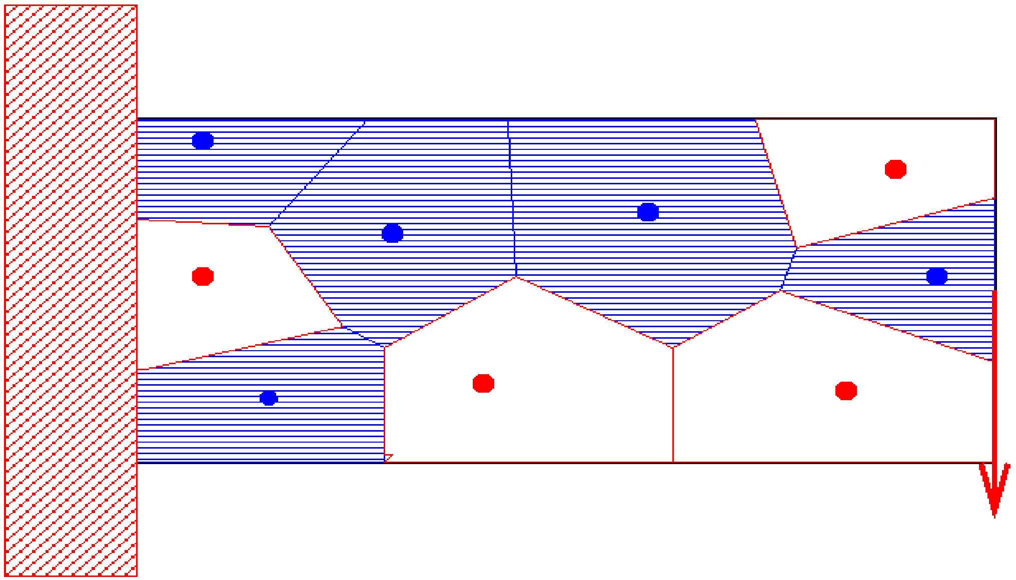
- Set of Voronoi sites $S_1, \ldots, S_n$ in the design domain
- A Voronoi cell is associated to each site:
  $Cell(S_i) = \{M; d(M,S_i) = \min_j d(M, S_j)\}$



Partition of the design domain in convex polygons

Shape representation

- Each site is labelled (0/1)
- Each cell receives its site label

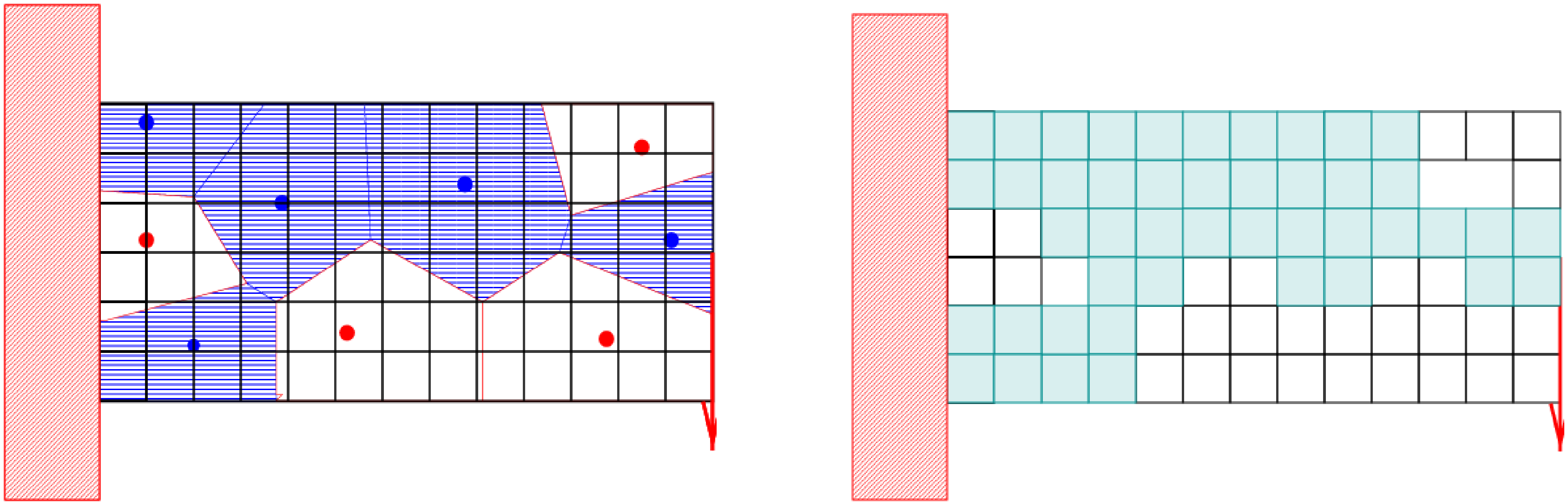


- Genotype: Variable length unordered list of labeled sites

$$\{n, (S_1,c_1), \ldots, (S_n,c_n)\}$$

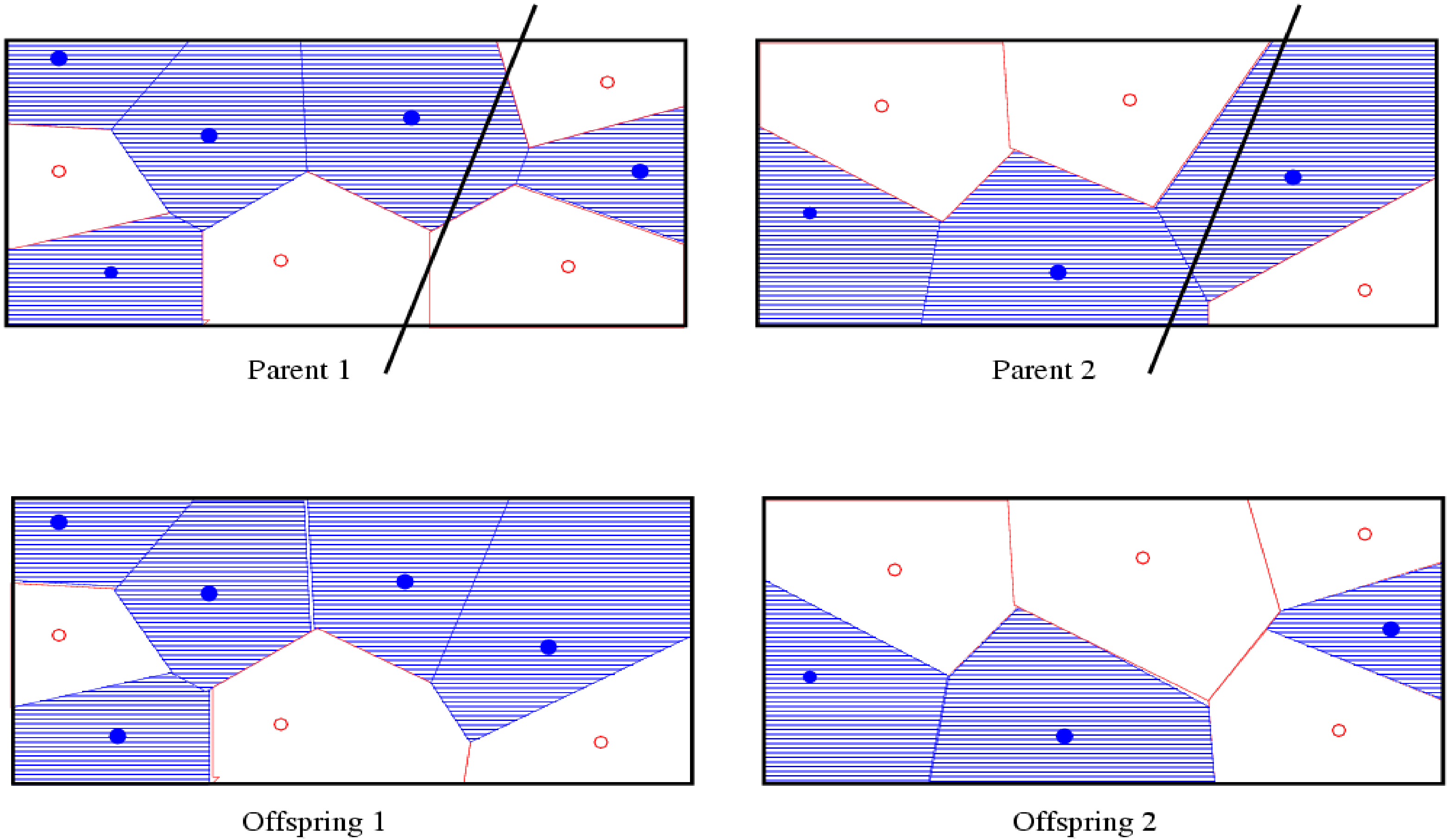
Morphogenesis

- Still need to use the same mesh for a whole generation



Projection on a given mesh

Variation operators

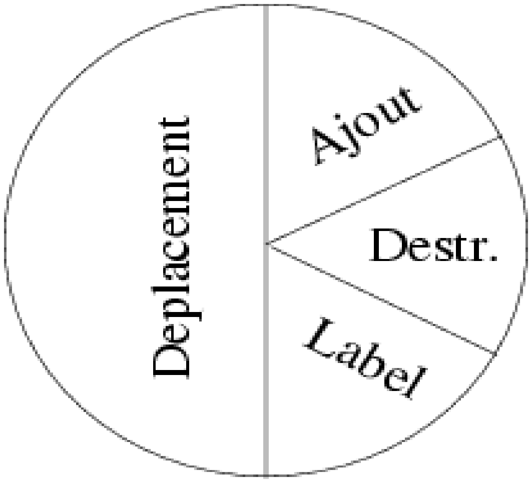


Geometrical exchange of Voronoi sites

Mutations

- Gaussian mutation of site coordinates
  possibly adaptive
- Label flip
- Addition of a Voronoi site — with biased label
- Deletion of a Voronoi site — biased toward redundant sites

- Random choice of mutation
  from user-defined weights



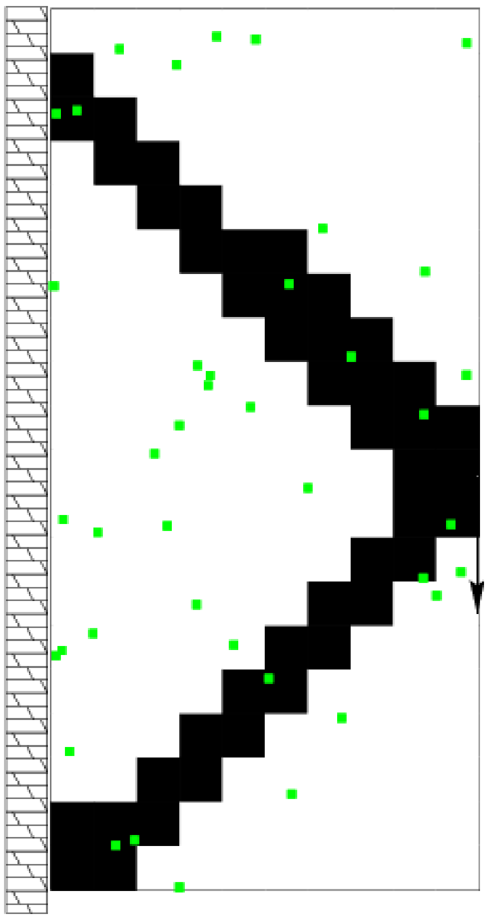
Experimental conditions

Cantilever 1 x 2 and 2 x 1

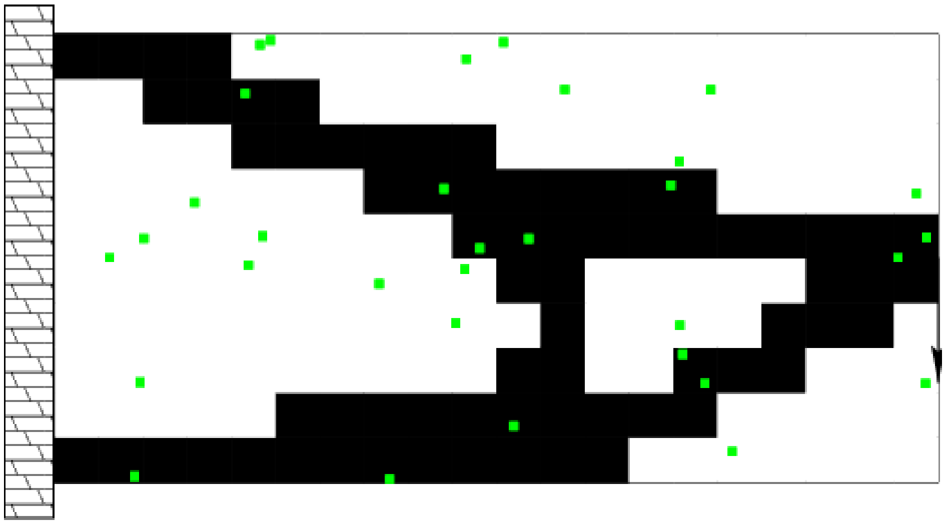
- Tournament(2) selection in (P+P)-ES engine
- P   80-120 → around 100 000 evaluations
- (0.6, 0.3, 0.1) weights for crossover, mutation, copy
- $(\frac{1}{2},\frac{1}{6},\frac{1}{6},\frac{1}{6})$ weights for the mutations
- 21 independent runs for each test
- Averages (and standard deviations)

Typical results

- 10 x 20 and 20 x 10 meshes
- Less than 1mn per run (today!)



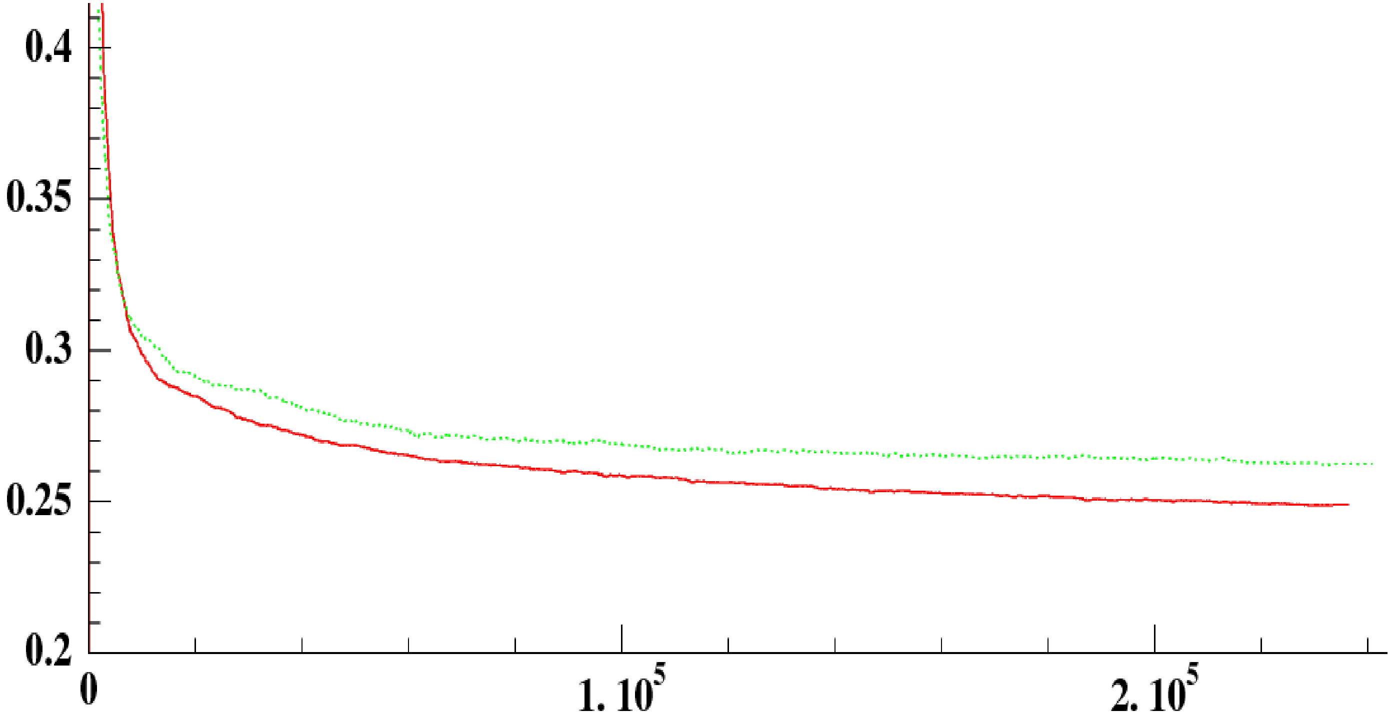
DLim = 20, weight=0.215, 35 sites



DLim = 220, weight=0.35, 32 sites

Complexity

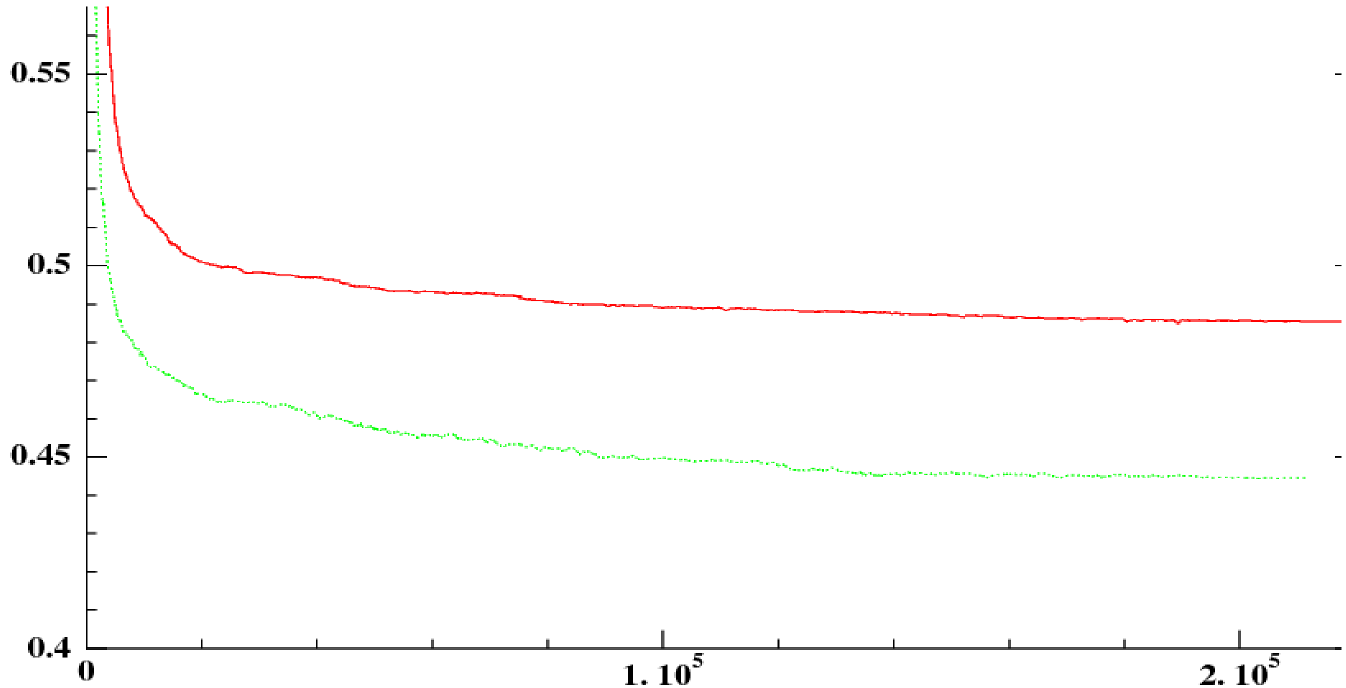
- Cantilever 1x2, D_{lim}=20,
- Two meshes: 20 x 10 and 40 x 20
- CPU cost x 3.5



Fitness vs # fitness evaluations (FEAs)

Complexity (2)

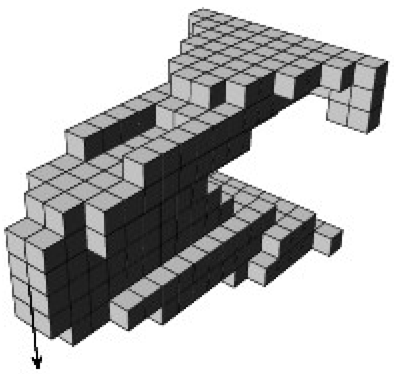
- Same conditions, except D_{lim}=10



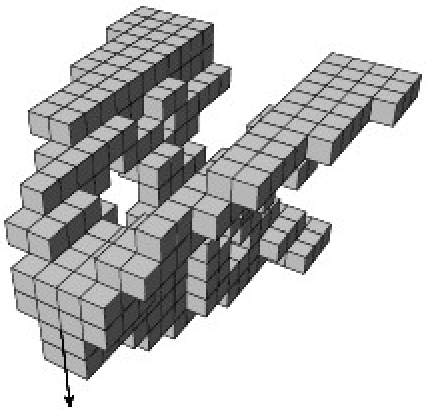
- Best sol. on 20 x 10: W = 0.44, D_{Max} = 9.99738
- Projected on the 40 x 20 mesh: W = 0.43125, D_{Max} = 11.2649

3D cantilever

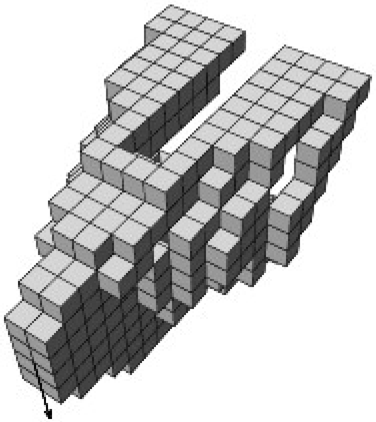
- 10 x 10 x 16 mesh
- Out of reach of bitarray representation (even today :-)
- Multiple quasi-optimal solutions



weight=0.152,
103 sites



weight=0.166,
109 sites



weight=0.157,
112 sites

Exploratory results
Coll. EZCT

Centre Georges Pompidou,
Collection permanente

Concours Serousi, Nov. 2007

Voronoi Representation

- Outperforms bitarray by far
  - Independence w.r.t. mesh complexity
- 3D, elongated cantilever (see later), …
- Opens the way toward Exploratory Design

But

- The problems are actually multi-objective
  - Minimize weight **and** maximize stiffness
  - … and those objectives are contradictory

Agenda

Evolutionary Algorithms

Topological Optimum Design

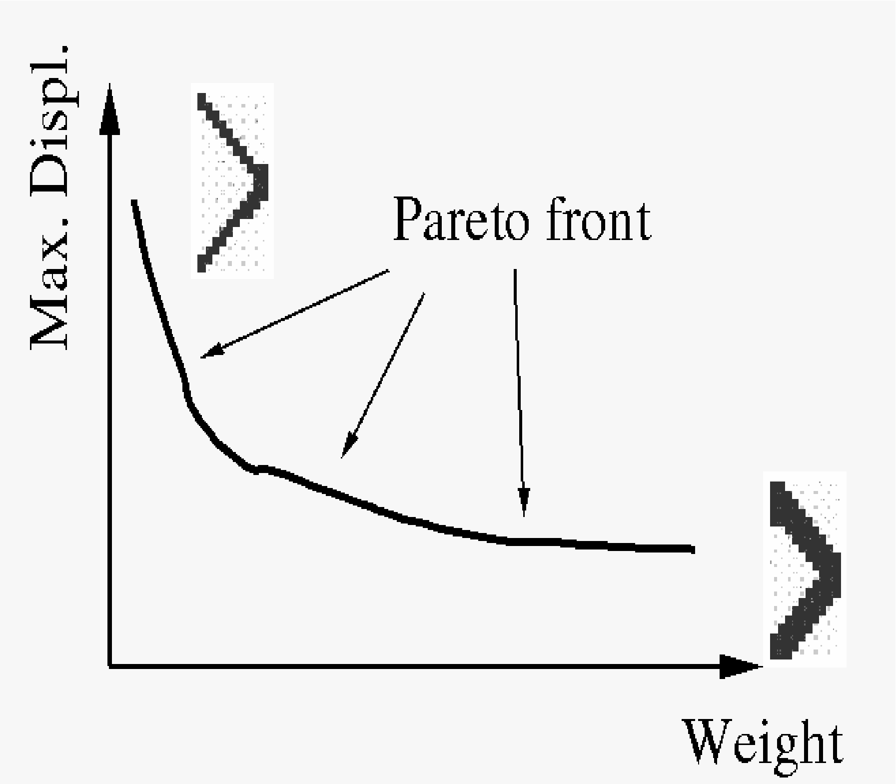
- The fitness function
- The bitarray representation
- The Voronoi representation
- Multi-objective optimization
- Modularity and Scalability

Multi-objective Optimization

- Several objectives to minimize (F_1, …, F_K)
- that are contradictory
- Need to re-define the idea of optimality
  - Nash equilibrium: each variable takes the best value given the other variables values
  - Pareto optimization: optimal trade-offs, based on the idea of Pareto dominance

Pareto optimization

- Pareto dominance: x dominates y if
  - $F_i(x) \leq F_i(y)$ for all i
  - $F_j(x) < F_j(y)$ for at least one j
- Pareto set: non-dominated points in search space
- Pareto front: same in objective space



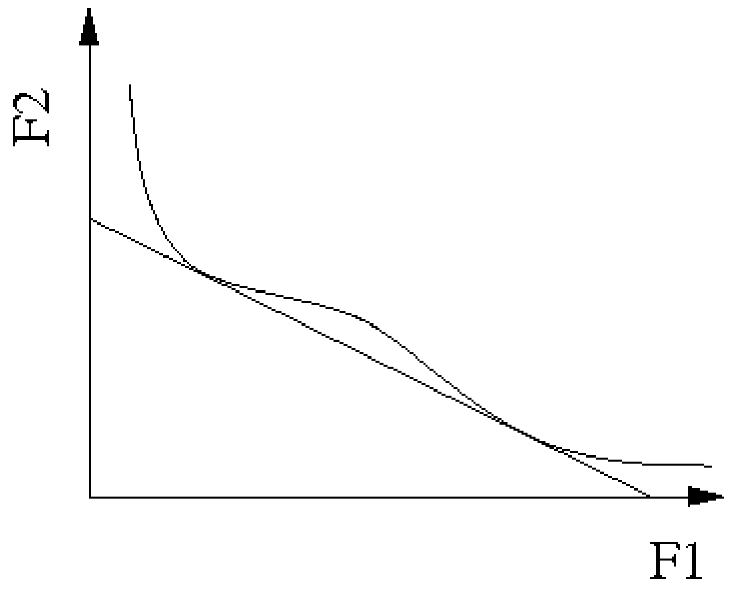
Goal

- Identify Pareto Front
- Make an informed decision

A classical approach

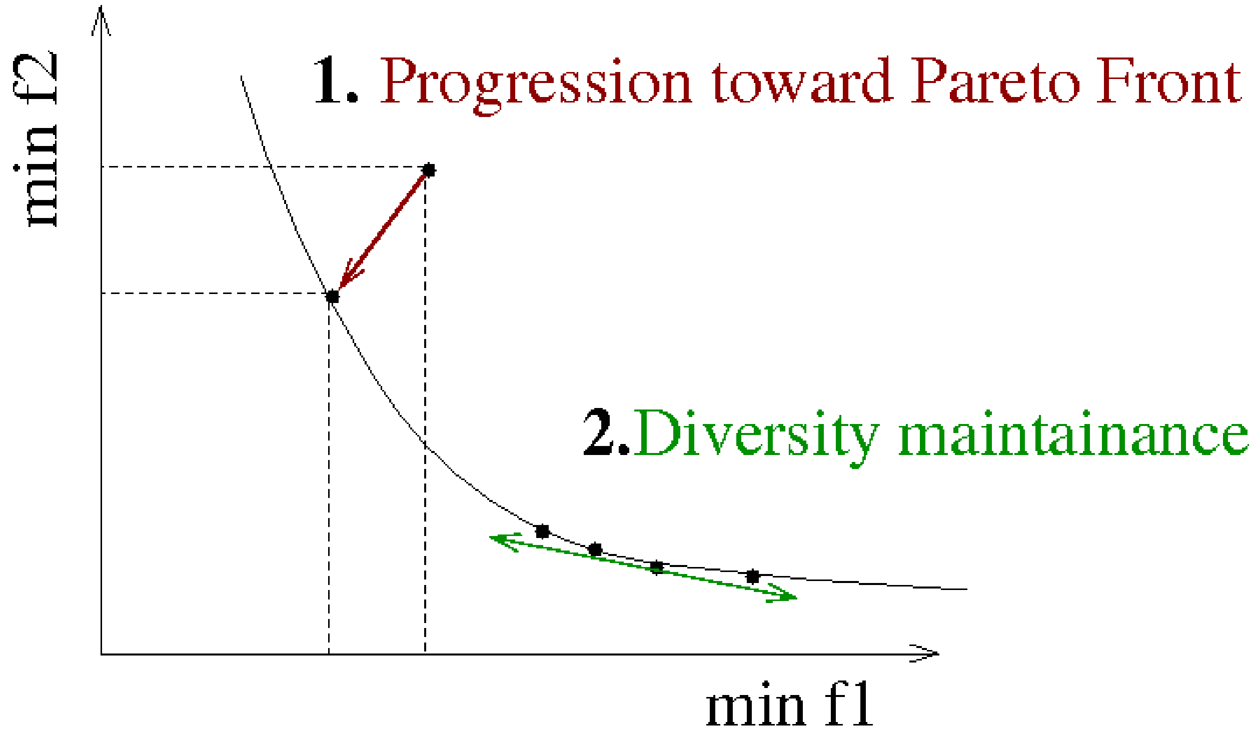
Aggregation of objectives

- Minimize $\Sigma_i \lambda_i F_i$
  - $\lambda i > 0$ iff Fi to be minimized
- Need to a priori fix λ_i
- One optimization per (λ_i)
- Concave parts of Pareto Front unreachable



Evolutionary approaches

- “Only” need to modify selection
- But Pareto dominance is only a partial order

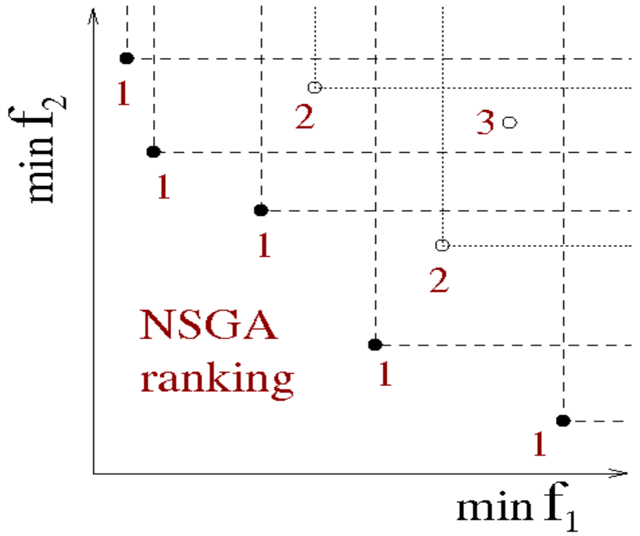


- Main criterion: Pareto Dominance
- Secondary criterion: diversty preserving measure

An example: NSGA-II

K. Deb, 2000

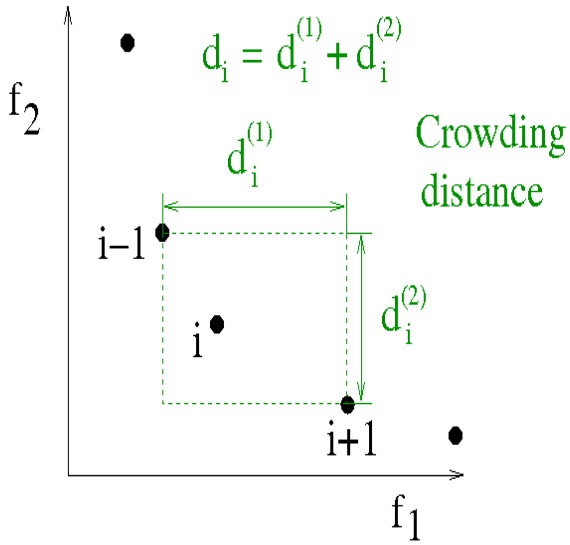
- Pareto ranking
  - Non-dominated: rank 1
  - Remove and loop



- Crowding distance

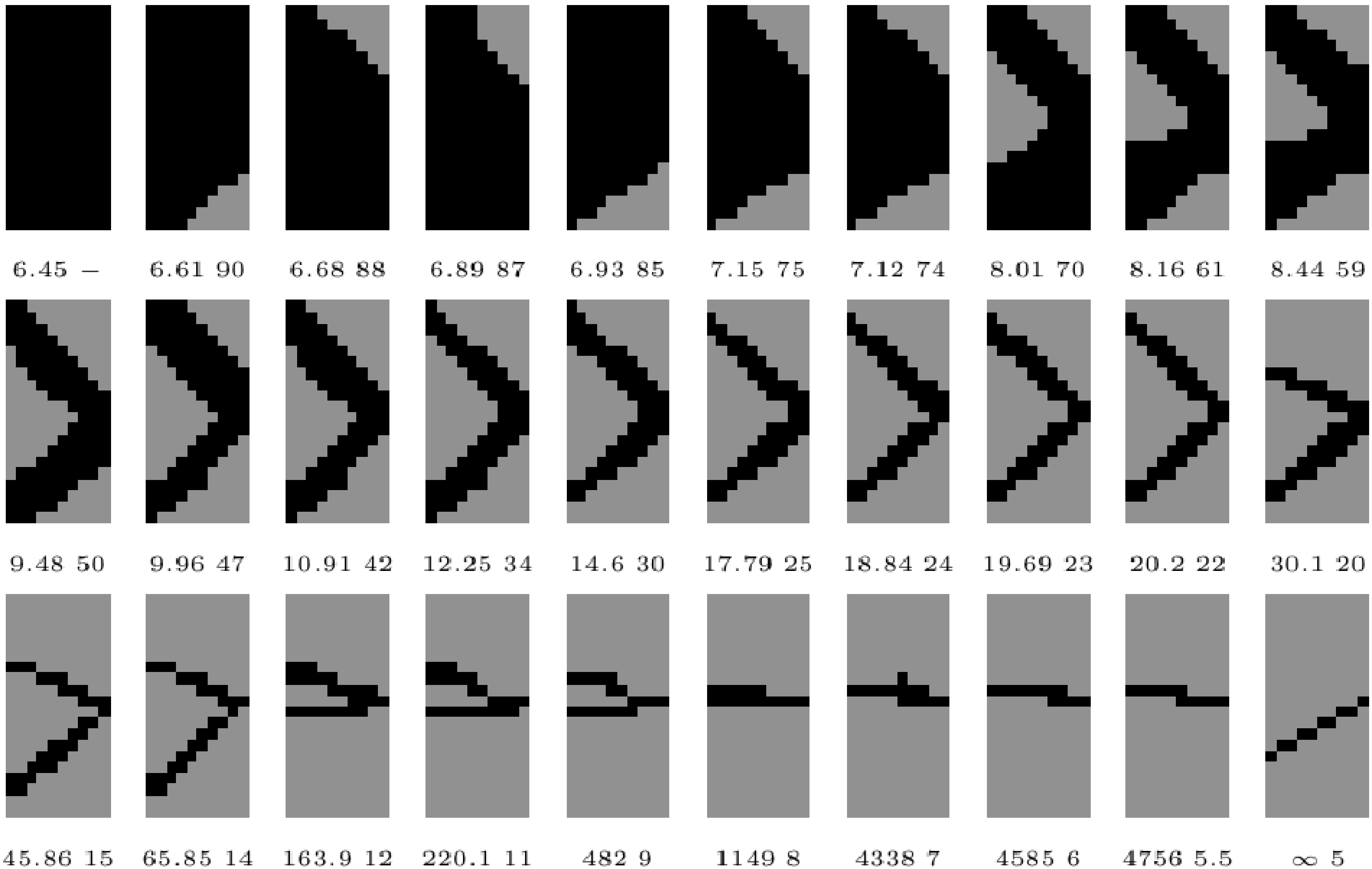
  for each criterion c
  - Sort according to F_i
  - $d_c(xi) = d(x_i,x_{i-1})+d(x_i,x_{i+1})$

$d_{crowding}(x) = \Sigma_c\ dc(x)$



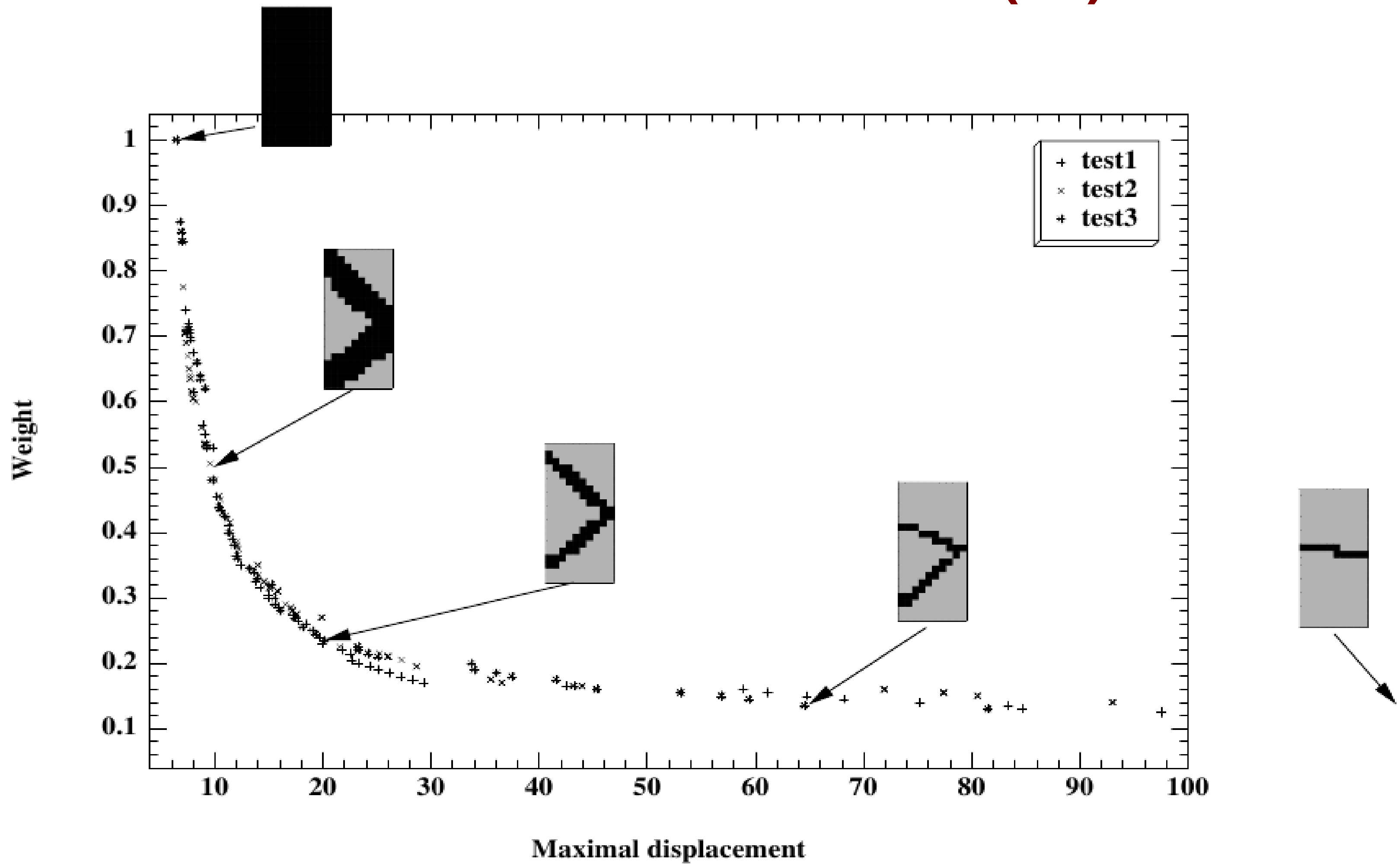
Cantilever 10 x 20

CPU cost ≈ 1.2 single objective run



Cantilever 10 x 20 (2)

- O

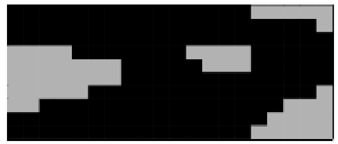


3 independent Pareto Fronts

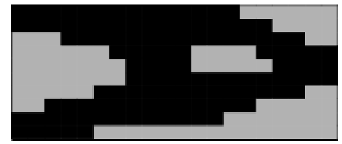
300 individuals, 400 generations

Cantilever 20 x 10

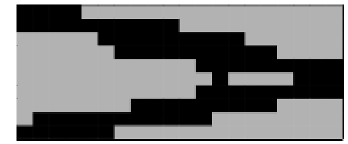
71.26 100



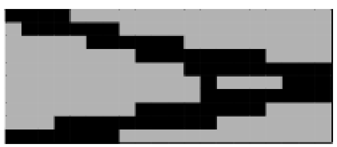
87.37 74.5



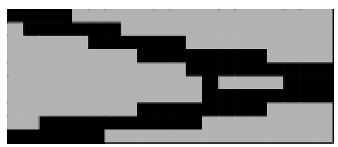
110.06 60



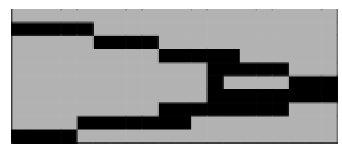
188.8 0.54



218.1 35

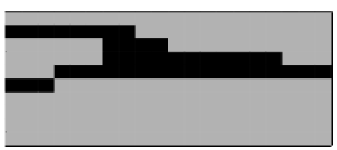


224 34.5

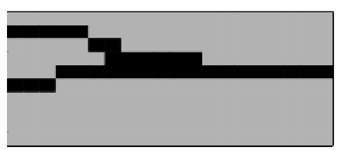


465.11 25

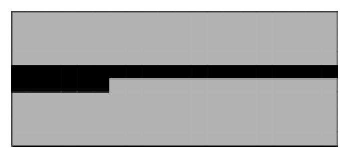
584.7 24



2112.6 21.5



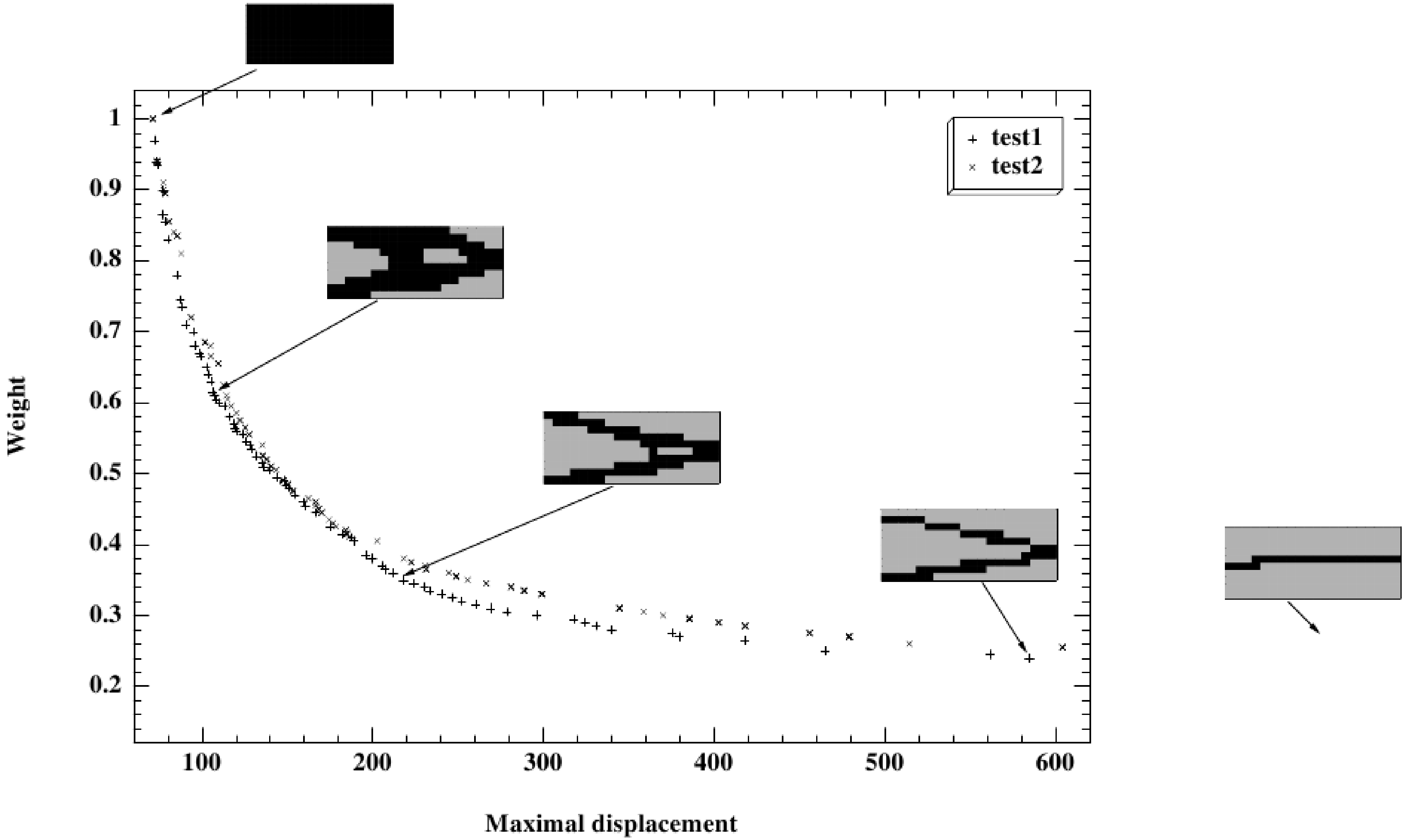
5810 16.5



18487 13

39050 10.5

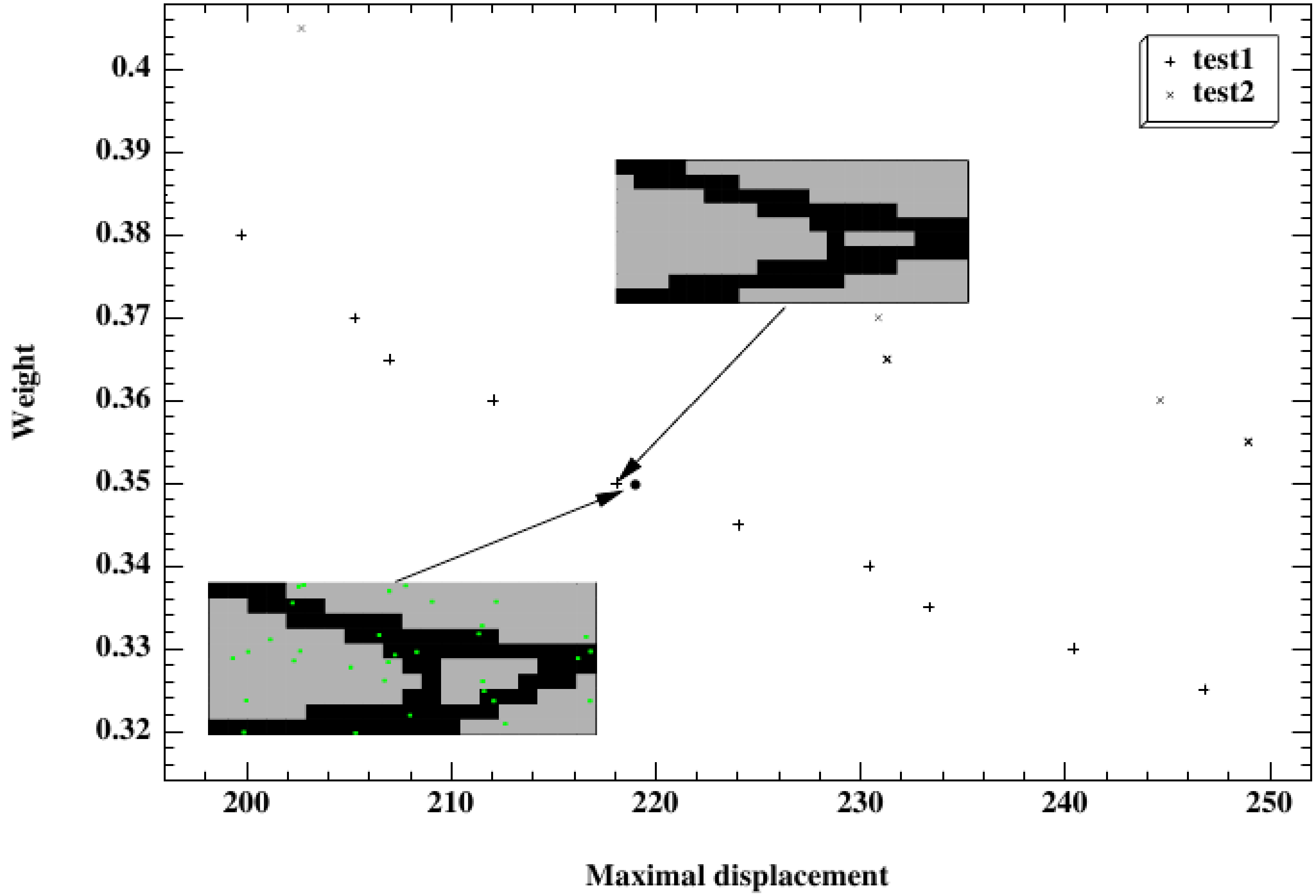
Cantilever 20 x 10 (2)



2 independent Pareto Fronts

300 individuals, 400 generations

Multi-objective vs single-objective



Zoom on Pareto Front, around D_{Max}=220

Top: multi-objective – Bottom: single-objective

Voronoi Representation

Pros

- More compact than enumerative bitarray
- Complexity is evolvable
  - Not imposed by technical considerations

Cons: lacks

- Scalability and modularity

Evolve large structures

Re-use parts

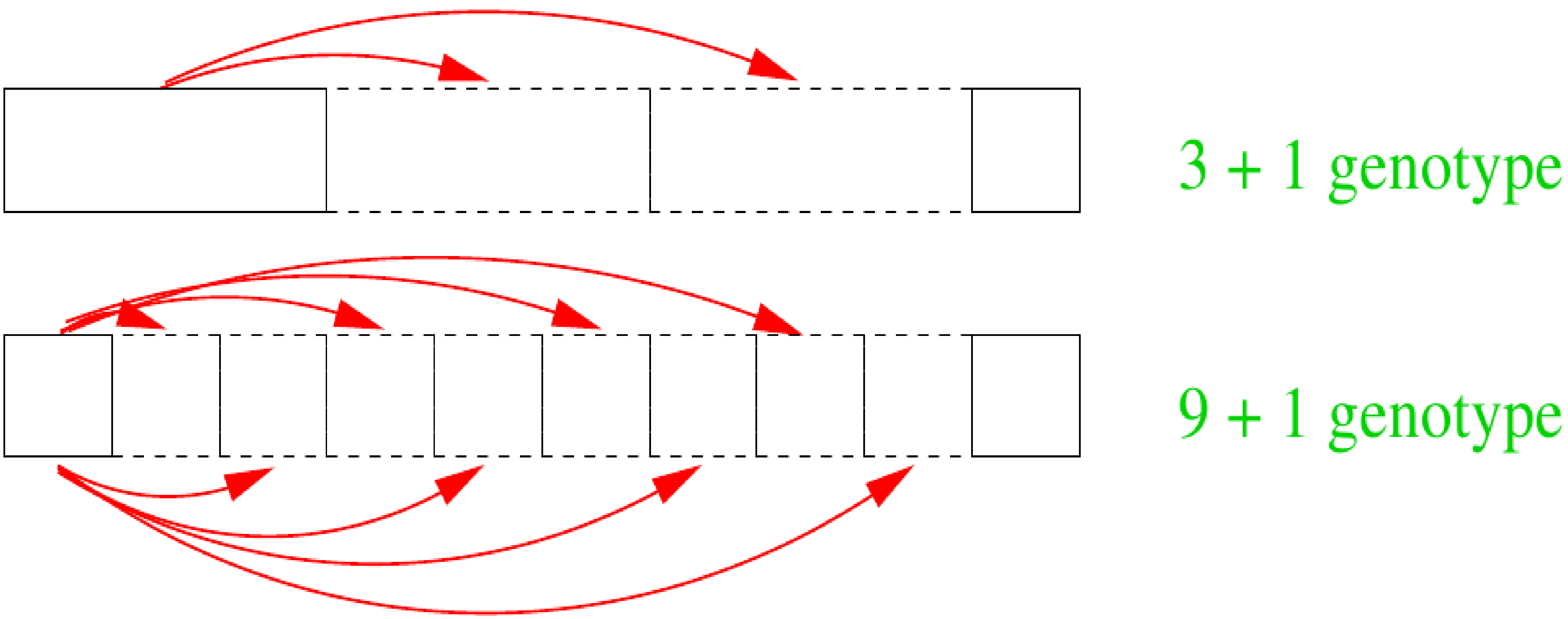
Agenda

Evolutionary Algorithms

Topological Optimum Design

- The fitness function
- The bitarray representation
- The Voronoi representation
- Multi-objective optimization
- Modularity and Scalability

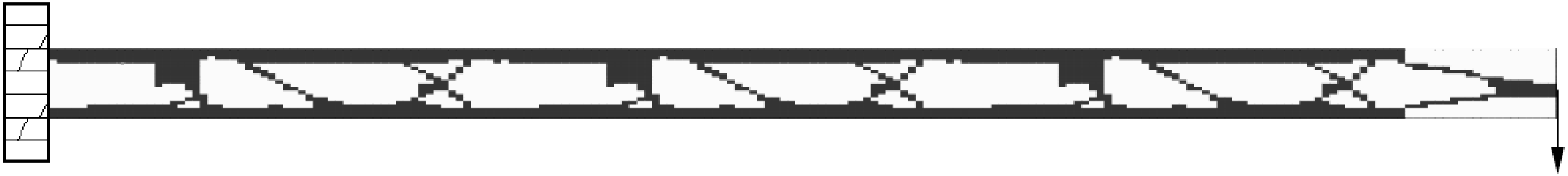
Manual Modularity



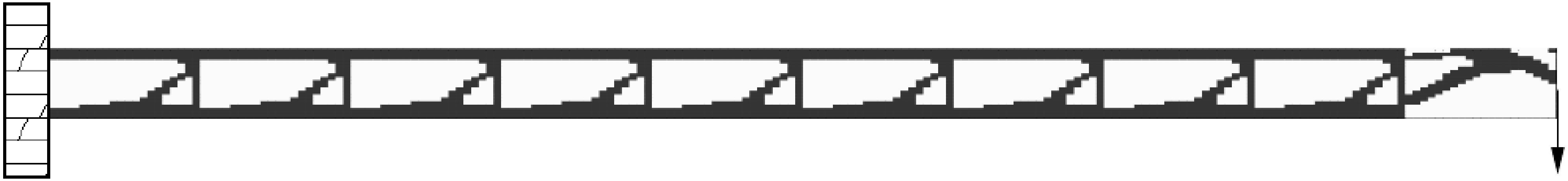
Best results

200 x 20 mesh, Dlim=12

1-genotype: Weight = 0.445, D_{max} = 11.99, 105 sites



3+1 genotype: Weight = 0.428, D_{max} = 11.98, 60 sites



9+1 genotype: Weight = 0.432, D_{max} = 11.99, 40 sites

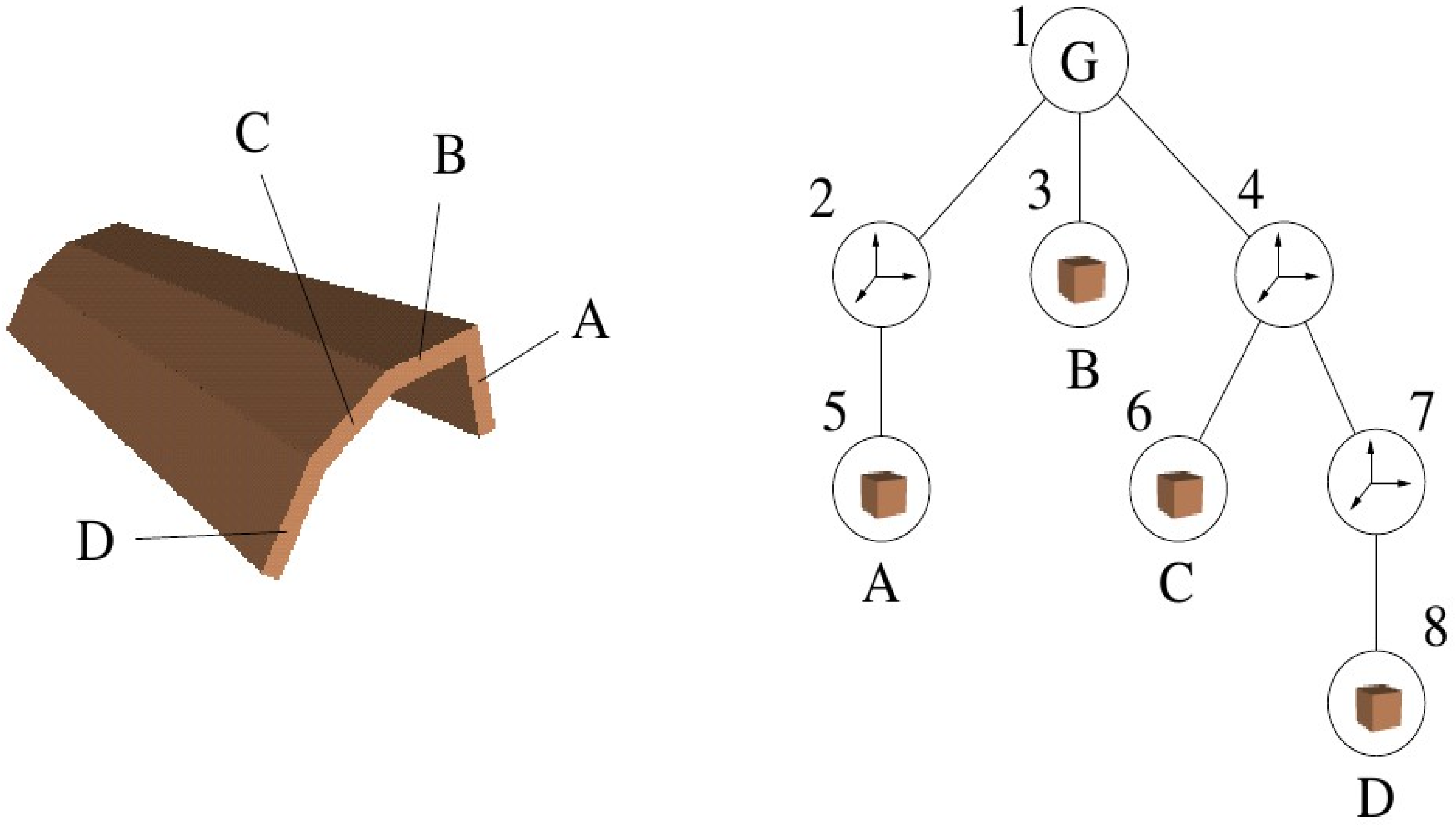
Evolution of Scene Graphs

Marc Ebner, Univ. Würzburg - 2003

- VRML: Virtual Reality Markup Language
- A scene is a hiearchical list of nodes
  - i.e., a tree, similar to Genetic Programming trees
- Nodes are
  - Elementary shapes
  - Geometrical transformations
  - Grouping of elements
- Evolved turbine shapes using GP techniques

Evolution of Scene Graphs

Example of a VRML Scene Graph



Agenda

Evolutionary Algorithms

Topological Optimum Design

- The fitness function
- The bitarray representation
- The Voronoi representation
- Multi-objective optimization
- Modularity and Scalability

Artificial Embryogeny

- Evolve the program that computes the solution rather than the solution itself

Most popular approaches

- Genetic Programming applied to some embryo
  e.g., to evolve digital circuits (Koza, 1998)
- Cellular automata (e.g., Conway's game-of-life) to mimick cell growth
  - Different cell types
  - Evolution modifies the update rules

Embryogeny for planar trusses

T. Kowaliw et al., Concordia U., Montreal - 2007

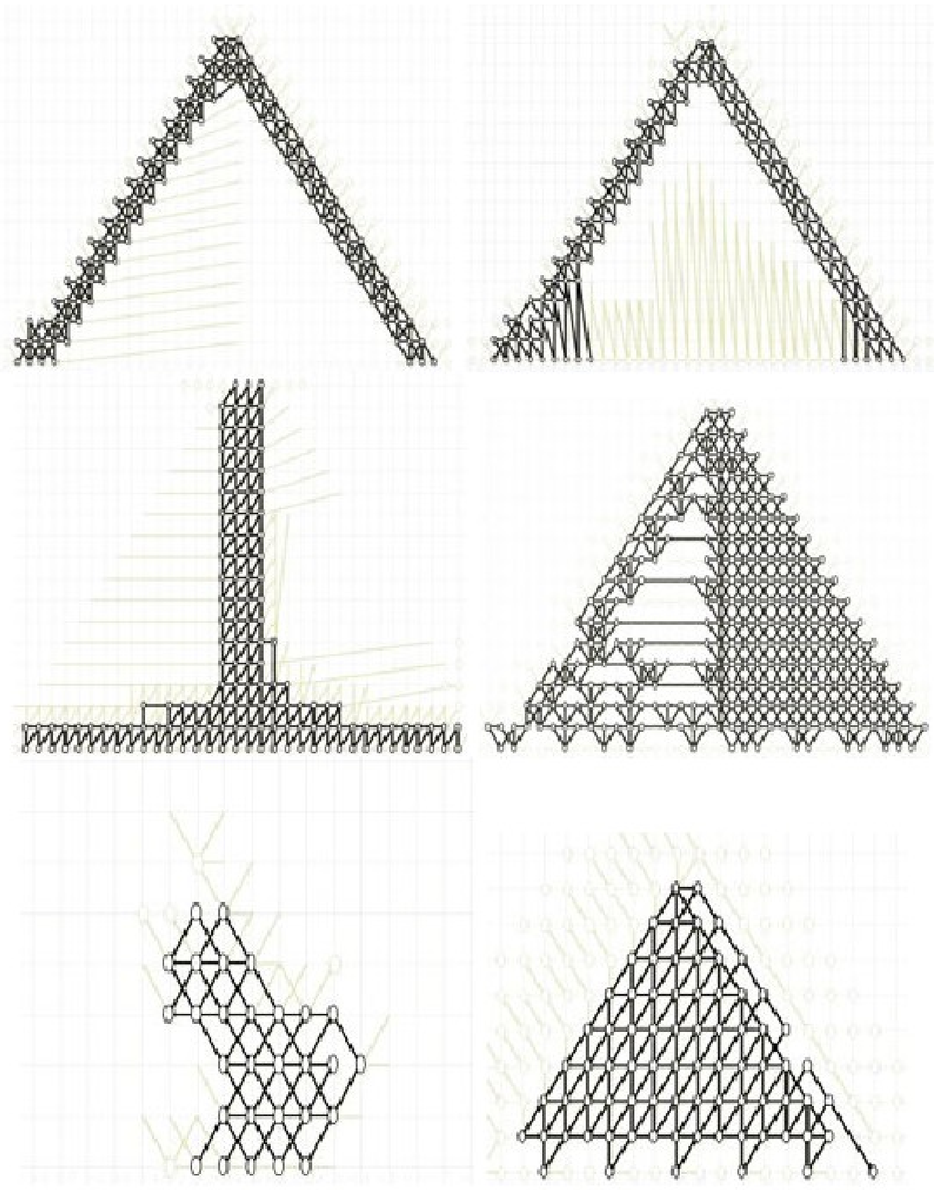
- Space of cells, originally empty except the central one
- All cells share update rules (c, h_1, …, h_{nc}, a) Evolved
  - c is a color
  - h_1, …, h_{nc} are “hormone levels”
  - Action a: *Nothing, Die, Divide, Elongate, Specialize(x)*

Developement

- For a given number of time step, and for each non-empty cell
  - Find the best matching rule
  - Apply corresponding action
- Tranform cells into joints and beam according to their colors

Embryogeny for planar trusses

T. Kowaliw et al., Concordia U., Montreal - 2007

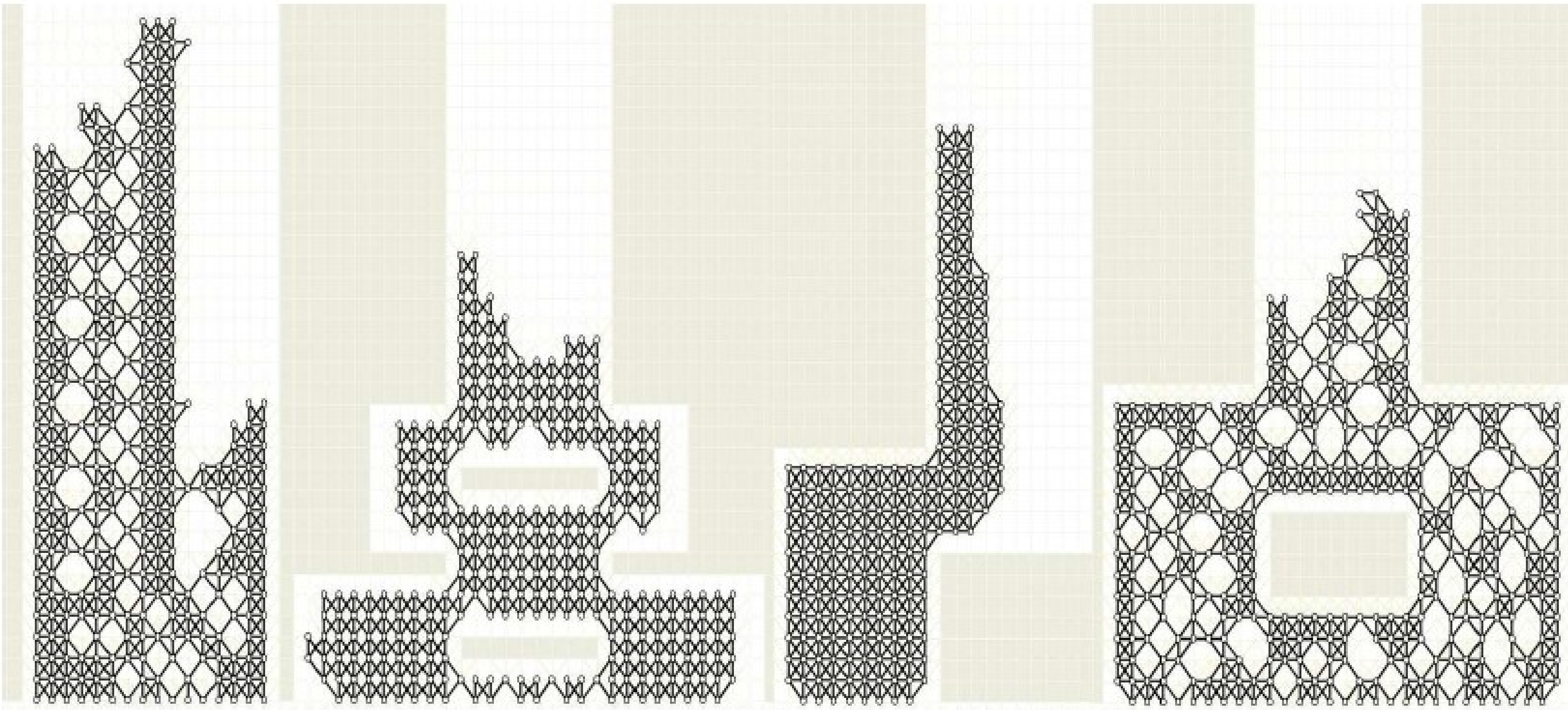


Optimized for

- height, weight, load on top
- height, weight, load at random locations
- height, weight, minimal base

Embryogeny with constraints

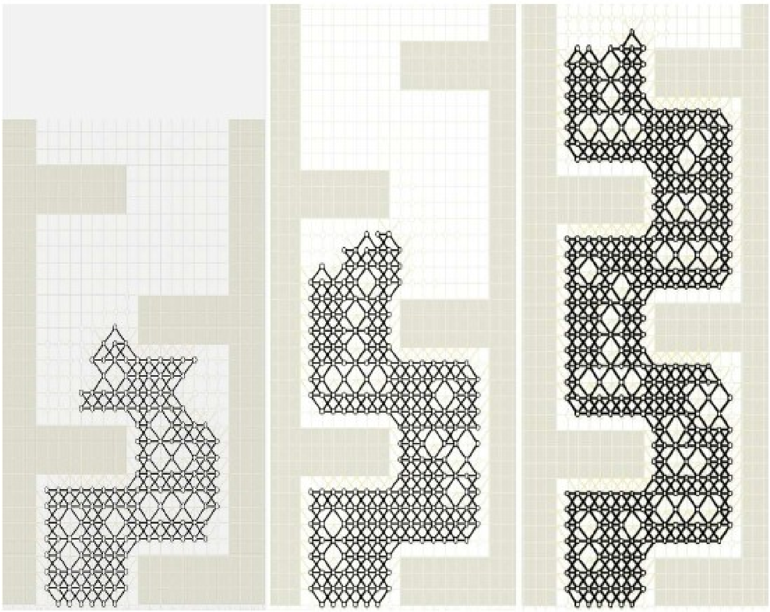
Kowaliw - 2008



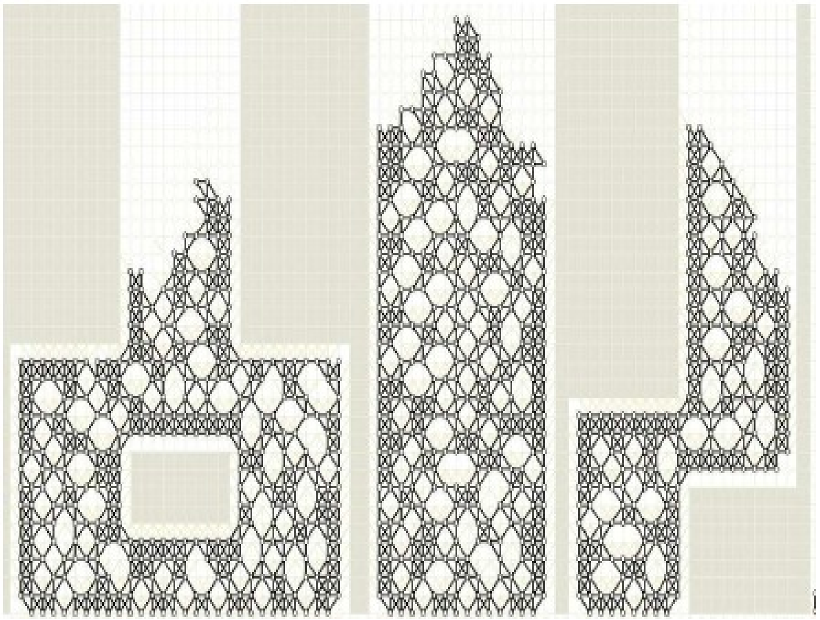
Similar objectives + geometrical constraints

Scalability and robustness

Kowaliw - 2008



Increasing development time after evolution

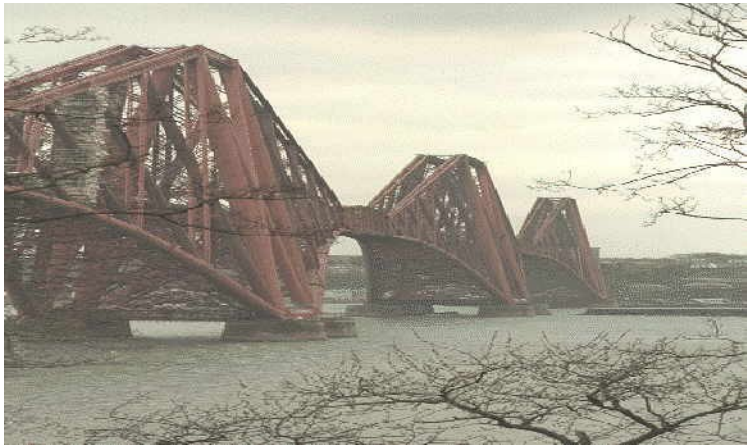


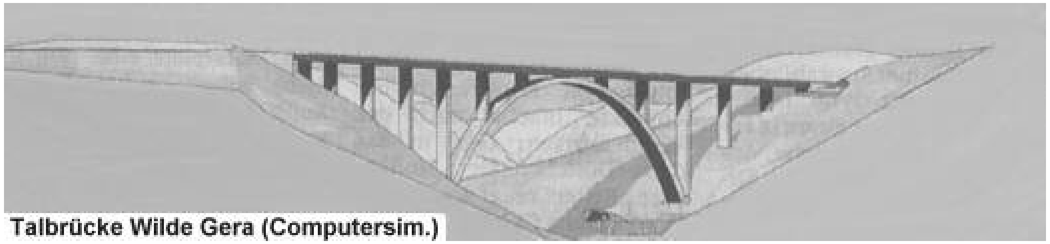
From the environment where evolution took place to an un-seen one

Conclusions

- EAs can solve hard optimization problems
  - Including Topological Optimum Design
- But EAs are also fantastic exploration tools
  - Giving hints toward surprising solutions
- Hybrids of EAs and classical methods are still to be built

Toward Artificial Creativity?





Talbrücke Wilde Gera (Computersim.)