CIVE 546 Structural Design Optimization

(3 units)

Sensitivity Analysis

Instructor: Prof. Yi Shao

Winter 2025

McGill Poll
Or hands up!

slido

Please download and install the Slido app on all computers you use

Which of the following is a convex set (choose ALL that apply)

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

Which of the following is a convex function when x1<=x<=x2 (select ALL that apply)

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

Which of the following is NOT a necessary condition for optimality?(select ALL that apply)

ⓘ Start presenting to display the poll results on this slide.

Karush–Kuhn–Tucker (KKT) optimality conditions w/o slack

Problem: minimize $f(\boldsymbol{x})$, where the design variable vector $\boldsymbol{x} = (x_1, \cdots, x_n)$, subjected to $(s.t.)$ $h_i(\boldsymbol{x}) = 0,\ i = 1 \cdots m;\ g_j(\boldsymbol{x}) \leq 0,\ j = 1 \cdots p.$

KKT 1) Lagrangian function definition

$$L(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i h_i(\boldsymbol{x}) + \sum_{j=1}^{p} \lambda_j g_j(\boldsymbol{x})$$

KKT 2) Gradient conditions

$$\frac{\partial L}{\partial x_k} = 0;\ \frac{\partial f}{\partial x_k} + \sum_{i=1}^{m} \lambda_i^* \frac{\partial h_i}{\partial x_k} + \sum_{j=1}^{p} \lambda_j^* \frac{\partial g_j}{\partial x_k} = 0,\ k = 1 \cdots n.$$

Karush–Kuhn–Tucker (KKT) optimality conditions w/o slack

KKT 3) Feasibility check

$$h_i(\boldsymbol{x}^*) = 0; i = 1 \cdots m;\ g_j(\boldsymbol{x}^*) \le 0, j = 1 \cdots p.$$

KKT 4) Switching conditions

$$\lambda_j^* g_j(\boldsymbol{x}^*) = 0, j = 1 \cdots p.$$

KKT 5) Non-negativity of LMs for inequalities

$$\lambda_j^* \ge 0;\ j = 1 \cdots p.$$

KKT 6) Regularity check

Gradients of active constraints must be linearly independent. In such case, the LMs for the constraints are unique.

slido

Please download and install the Slido app on all computers you use

When are the necessary conditions also sufficient conditions for local optimum being global optimum?

ⓘ Start presenting to display the poll results on this slide.

slido

Please download and install the Slido app on all computers you use

Which of the statement is true?

ⓘ Start presenting to display the poll results on this slide.

Recall the definition of positive definite

If for any non-zero real $\underline{x}$

$\underline{x}^T \underline{H}\, \underline{x} > 0$, $\underline{H}$ is positive definite

$\underline{x}^T \underline{H}\, \underline{x} \geq 0$, $\underline{H}$ is positive semidefinite

Prove that **<u>a diagonal matrix</u>** is positive definite if and only if all its diagonal entries are positive.

What are the conditions for a diagonal matrix to be

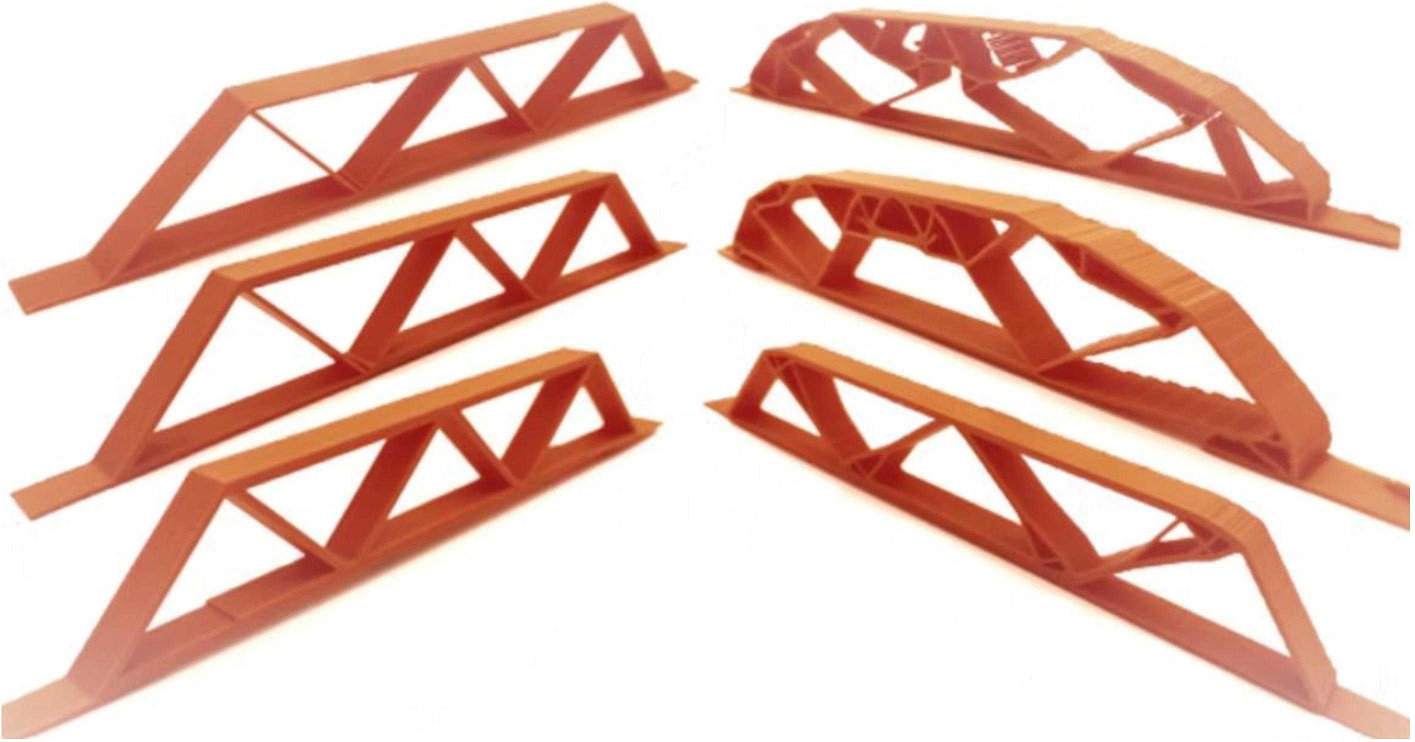
- positive semi-definite $H_i \geq 0$
- negative definite $H_i < 0$
- negative semi-definite $H_i \leq 0$
- indefinite H ?

Administrative announcement

Do you have classes on Monday afternoon?

Can we have class from 1:30pm-3:30pm on Monday (Mar 17)?

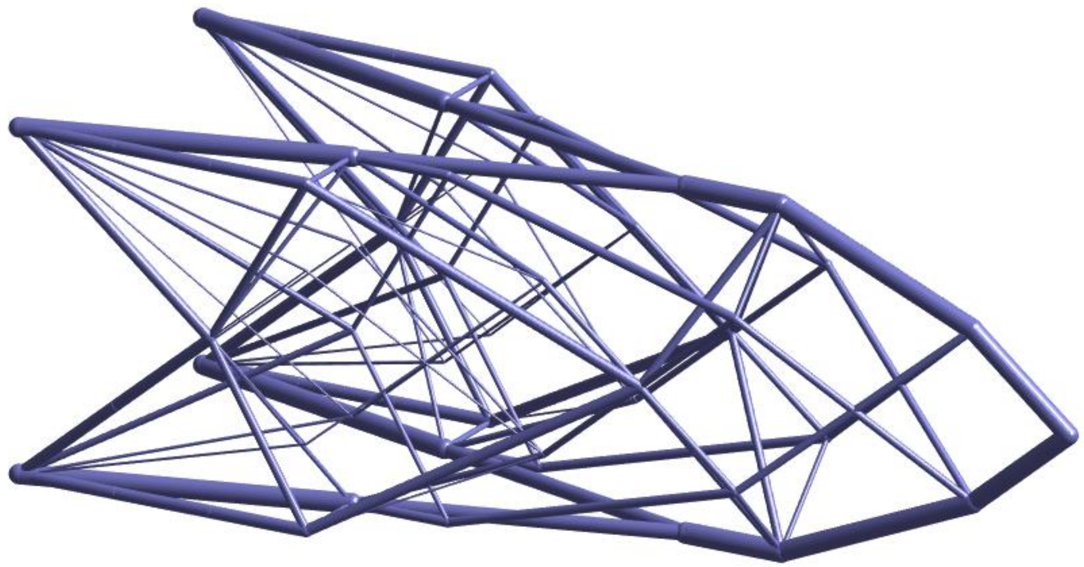
Guest lecture from **Prof. Josephine Carstensen (MIT)** on Structural Design Optimization

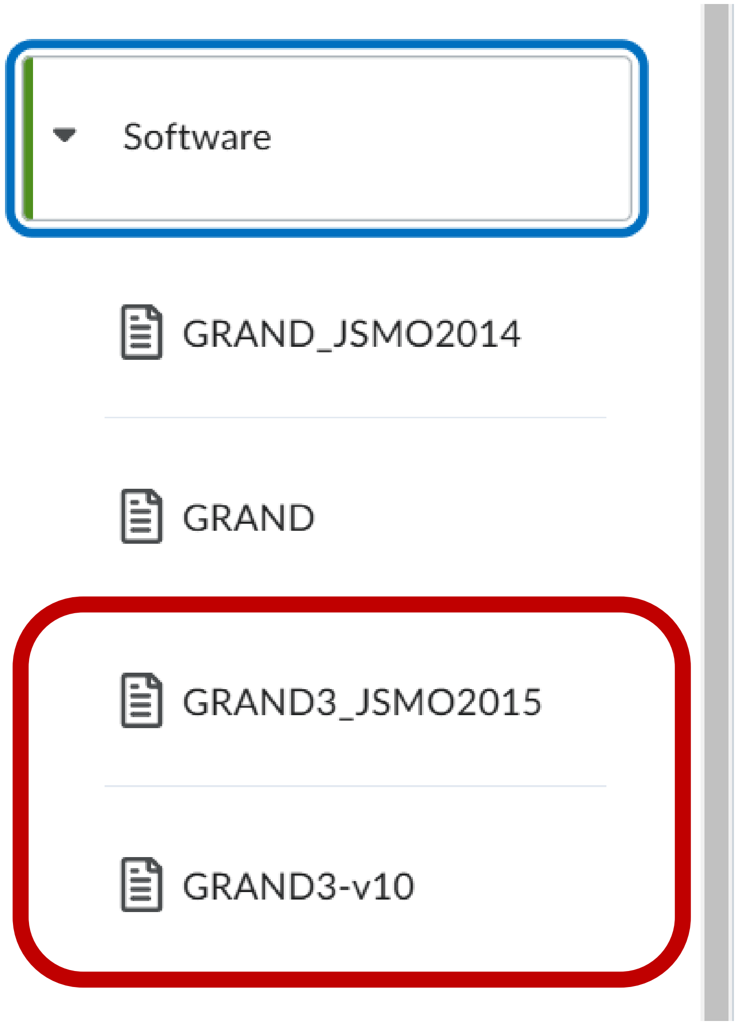


Administrative announcement

Homework 3 assigned

Read GRAND3 paper and play with the software





GRAND3

MATLAB

```
%GRAND3 - 3D Ground Structure Analysis and Design Code.
%   Tomas Zegard, Glaucio H Paulino - Version 1.0, Nov-2014
clc,close,clear all
%% === MESH GENERATION LOADS/BCS ====================================
kappa = 1; ColTol = 0.999999;
Ff = 2; Cutoff = 0.005; % Plot: Facet factor & member Cutoff

% --- OPTION 1: STRUCTURED-ORTHOGONAL MESH GENERATION ---------------------
[NODE,ELEM,SUPP,LOAD]=StructDomain3(10,4,4,30,10,10,'Cantilever');
Lvl=3; RestrictDomain=[];
```

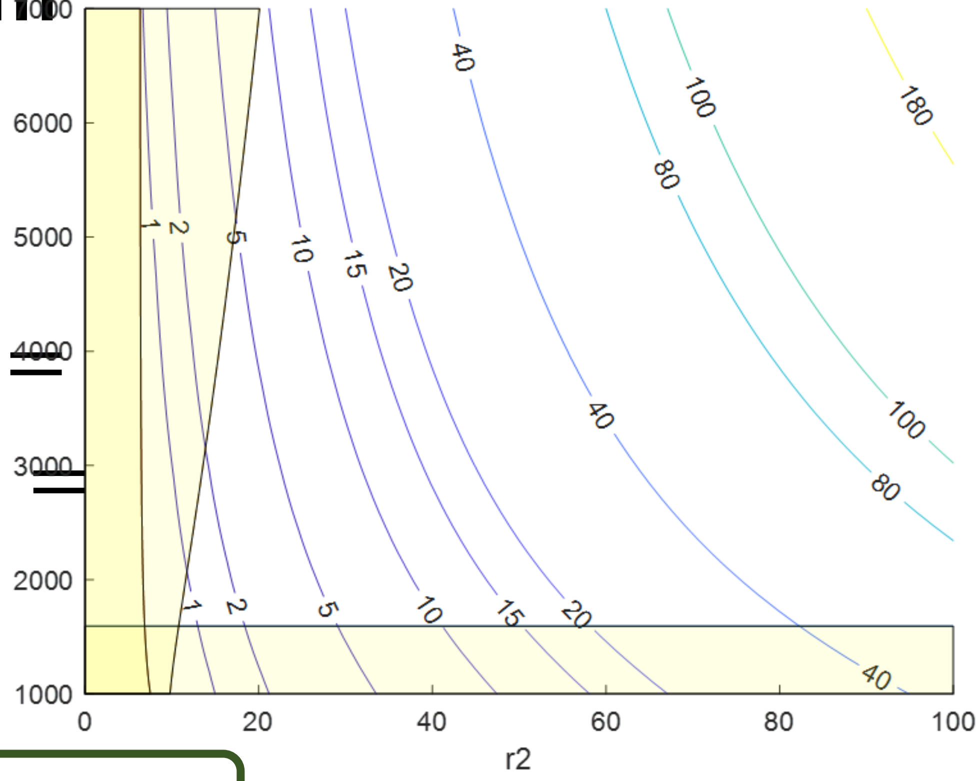
GRAND3

MATLAB

```
function [NODE,ELEM,SUPP,LOAD]=StructDomain3(Nx,Ny,Nz,Lx,Ly,Lz,ProblemID)
nargchk(nargin,6,7);

% Generate structured-orthogonal domains
[X,Y,Z] = meshgrid(linspace(0,Lx,Nx+1),linspace(0,Ly,Ny+1),linspace(0,Lz,Nz+1))
NODE = [reshape(X,numel(X),1) reshape(Y,numel(Y),1) reshape(Z,numel(Z),1)];
Nn = (Nx+1)*(Ny+1)*(Nz+1); Ne = Nx*Ny*Nz;

ELEM.V = cell(Ne,1);
aux = [1 Ny+2 Ny+3 2 (Nx+1)*(Ny+1)+[1 Ny+2 Ny+3 2]];
for k=1:Nz
    for j=1:Nx
        for i=1:Ny
            n = (k-1)*Ny*Nx + (j-1)*Ny + i;
            ELEM.V{n} = aux + (k-1)*(Ny+1)*(Nx+1) + (j-1)*(Ny+1) + i-1;
        end
    end
end

if (nargin<7 || isempty(ProblemID)), ProblemID = 1; end
switch ProblemID
    case {'Cantilever','cantilever',1}
        if rem(Ny,2)~=0, fprintf('INFO - Ideal Ny is EVEN.\n'), end
        SUPP = [ (1:(Nx+1)*(Ny+1):Nn)'   ones(Nz+1,3);
                 (Ny+1:(Nx+1)*(Ny+1):Nn)' ones(Nz+1,3)];
        LOAD = [round((Nz+1)/2)*(Nx+1)*(Ny+1)-round(Ny/2) 0 0 -1];
```

General form of an optimization problem

Objective function	$\min f(\underline{x})$	
	Subject to:	
Inequality constraints	$g_j(\underline{x}) \le 0$	$j =$
Equality constraints	$h_k(\underline{x}) = 0$	$k =$
Box constraints	$x_i^L \le x_i \le x_i^U$	

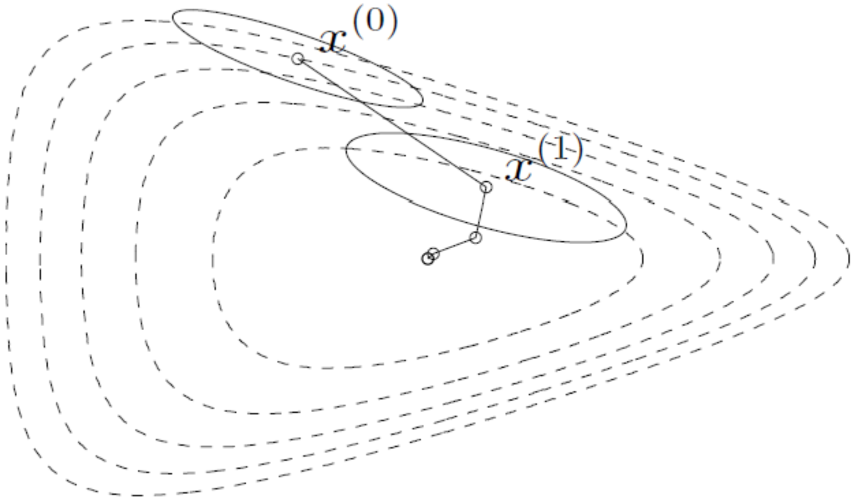


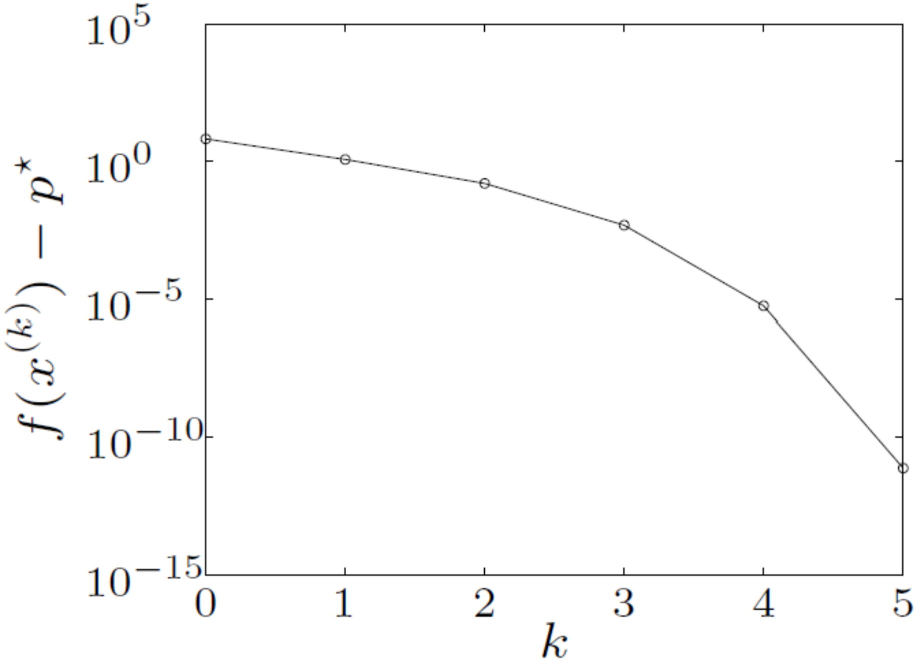
Optimization Method

- **Optimality Condition Method**
- **Search Method**

Sensitivity Analysis
Overview

1. General Optimization
2. Topology Optimization





Unconstrained Minimization

General Descent Method

$$\underline{x}^{k+1} = \underline{x}^{k} + \alpha^{k}\underline{\Delta x}^{k}$$

So that $f(\underline{x}^{k+1}) < f(\underline{x}^{k})$

Where, k is iteration number

$\underline{\Delta x}^{k}$ is the step or search direction

α^{k} is the step size or step length

Procedure:

given a starting point $\underline{x}^{0} \in$ **dom** f.

Repeat

1. Determine a descent direction $\underline{\Delta x}^{k}$.
2. Line search. Choose a step size α^{k}.
3. Update. $\underline{x}^{k+1} = \underline{x}^{k} + \alpha^{k}\underline{\Delta x}^{k}$.

Until stopping criterion is satisfied.

Procedure:

given a starting point $\underline{x}^{0} \in$ **dom**

Repeat

1. Determine a descent direction
2. Line search. Choose a step size
3. Update. $\underline{x}^{k+1} = \underline{x}^{k} + \alpha^{k}\underline{\Delta x}^{k}$.

Until stopping criterion is satisfi

Unconstrained Minimization
Descent Direction

To be a descent direction, $\underline{\Delta x}^k$ should meet the following condition

$$\nabla f\left(\underline{x}^k\right)^T \cdot \underline{\Delta x}^k < 0$$

$(e^x)' = e^x$

Example:

Given $f\left(\underline{x}\right) = x_1^2 - x_1 x_2 + 2x_2^2 - 2x_1 + e^{x_1+x_2}$

Check if $\underline{\Delta x} = [1,2]^T$ @ $\underline{x} = [0,0]^T$ is a descent direction for $f\left(\underline{x}\right)$

$$\nabla f(\underline{x}) = \begin{bmatrix} 2x_1 - x_2 - 2 + e^{x_1+x_2} \cdot (1) \\ -x_1 + 4x_2 + e^{x_1+x_2} \cdot (1) \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \quad \nabla f(\underline{x})^T \begin{bmatrix} 1 \\ 2 \end{bmatrix} = 1 > 0$$

Unconstrained Minimization

Line Search

12:45

Exact line search: $\alpha = argmin_{\alpha>0} f(\underline{x} + \alpha\underline{\Delta x})$

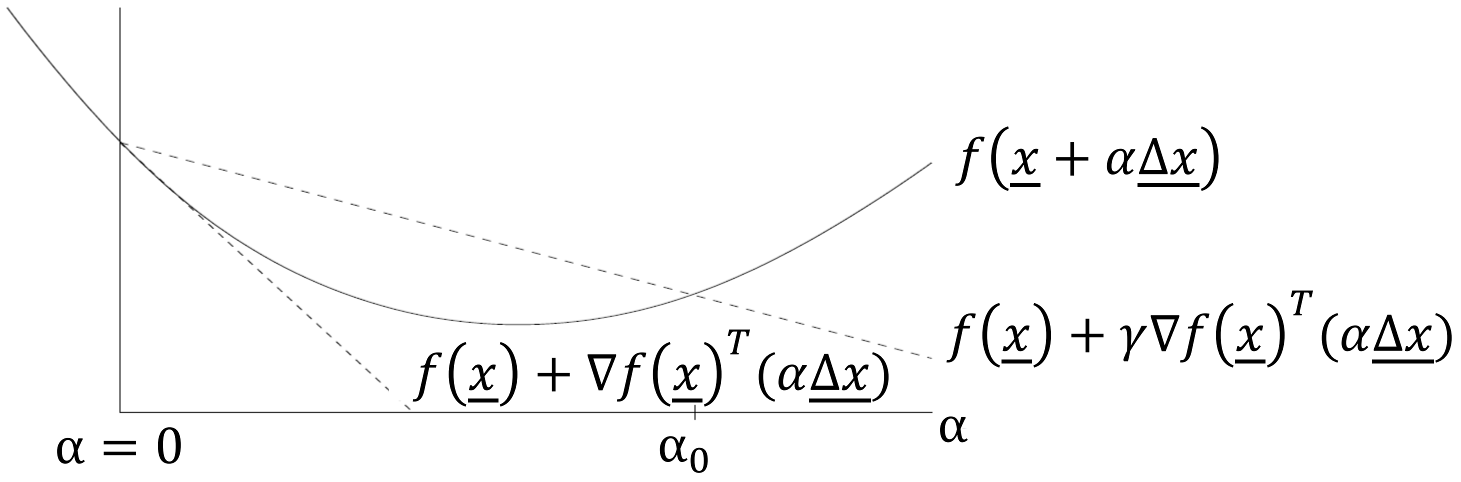
Backtracking line search:

with parameters $\gamma \in (0,0.5), \beta \in (0,1)$

- Starting at $\alpha = 1$, repeat $\alpha := \beta\alpha$ until

$$f(\underline{x} + \alpha\underline{\Delta x}) < f(\underline{x}) + \gamma\nabla f(\underline{x})^T(\alpha\underline{\Delta x})$$

- Graphical interpretation: backtracking until $\alpha \le \alpha_0$

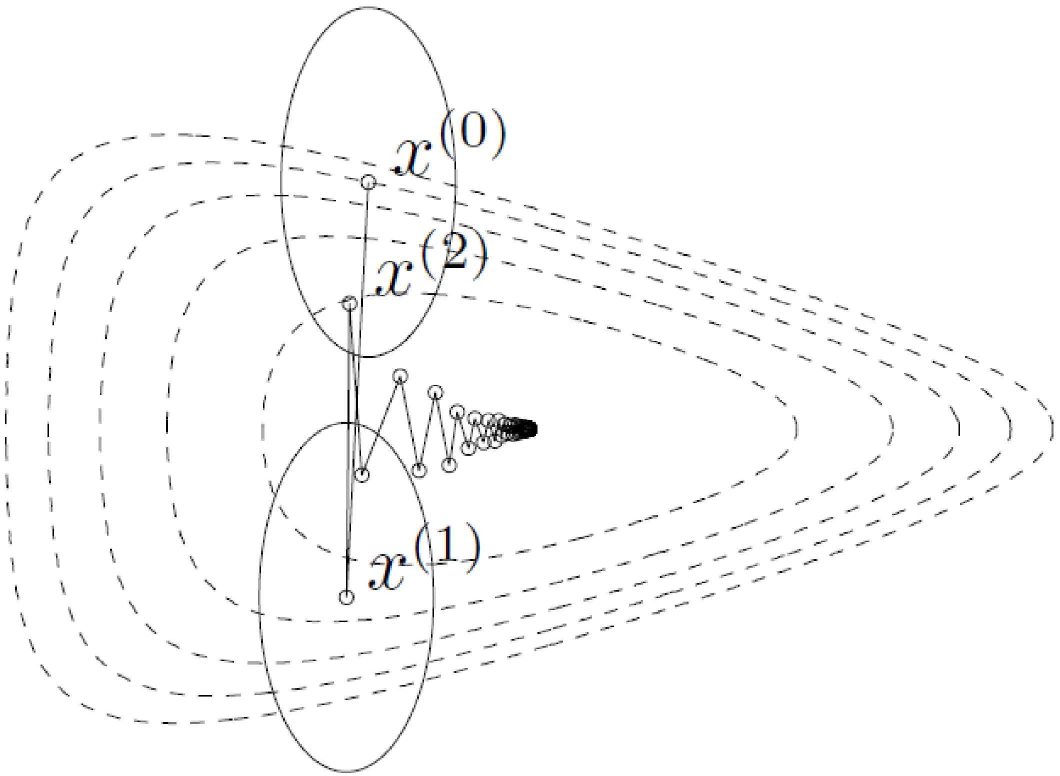


Unconstrained Minimization

Steepest Descent Method (First-order method)

$$\underline{x}^{k+1} = \underline{x}^{k} + \alpha^k \underline{\Delta x}^{k}$$

Where $\underline{\Delta x}^{k} = -\frac{\nabla f(\underline{x}^k)}{|\nabla f(\underline{x}^k)|}$: steepest descent direction



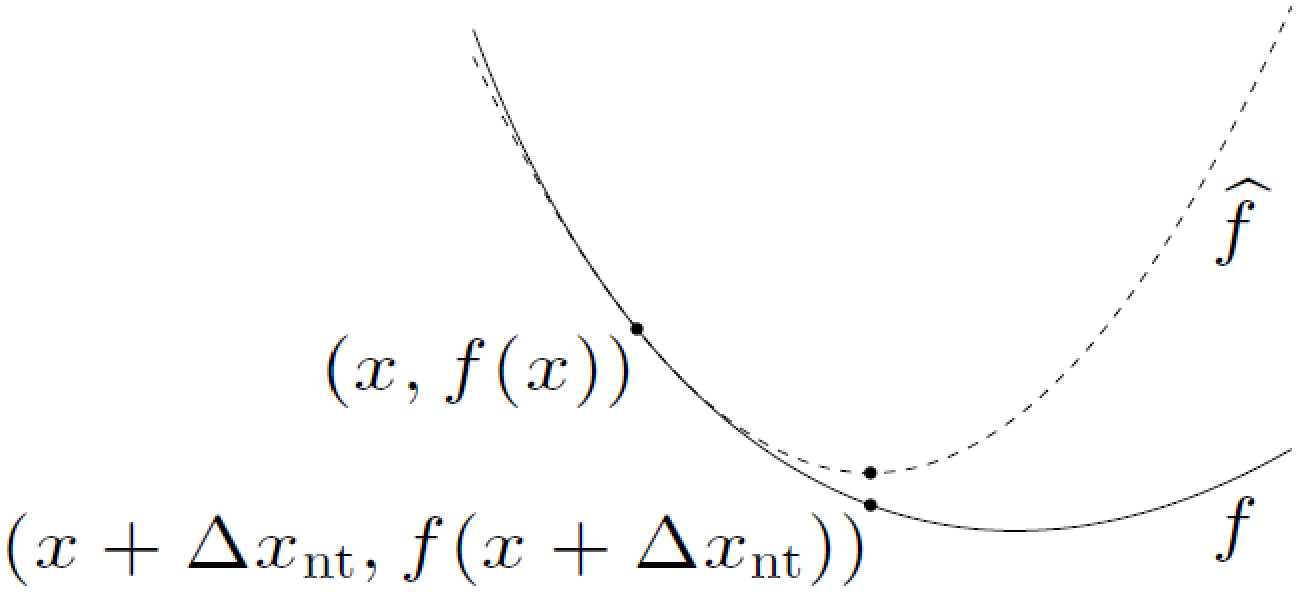
Challenging if the condition number of the hessian matrix is large, which indicates an elongated design space.

Condition number is the ratio of the largest eigenvalue to the smallest eigenvalue

Unconstrained Minimization
Newton's Method (Second-order method)

$$\underline{x}^{k+1} = \underline{x}^k + \alpha^k \underline{\Delta x}^k$$

Where $\underline{\Delta x}^k = -\nabla^2 f\left(\underline{x}^k\right)^{-1} \nabla f(\underline{x}^k)$



Unconstrained Minimization
Newton's Method (Second-order method)

$$\underline{x^{k+1}} = \underline{x^k} + \alpha^k \underline{\Delta x^k}$$

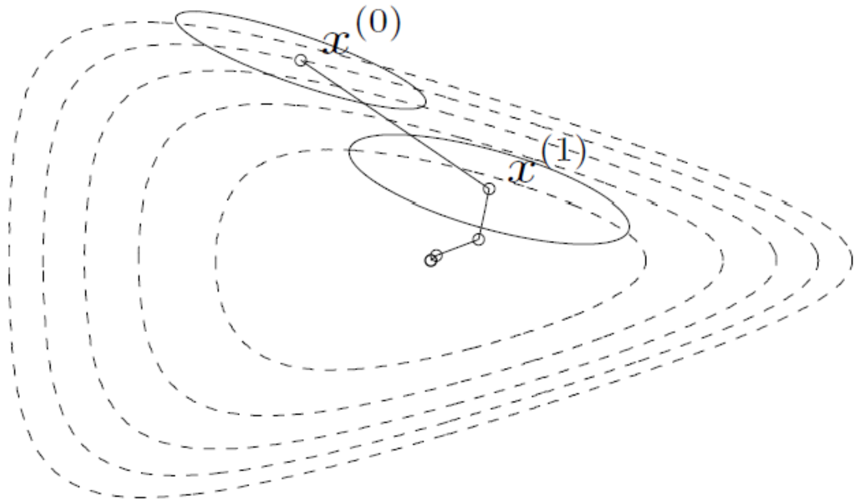
Where $\underline{\Delta x^k} = -\nabla^2 f(\underline{x^k})^{-1} \nabla f(\underline{x^k})$

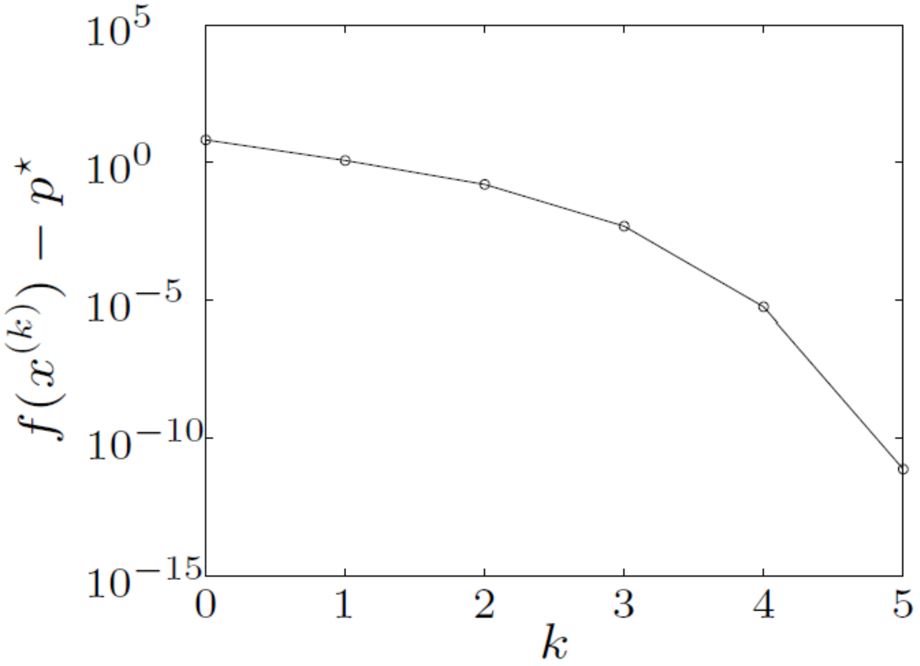
Advantage:

- Fast convergence

Disadvantage:

Computing $\nabla^2 f(\underline{x^k})^{-1}$ can be highly computational expensive

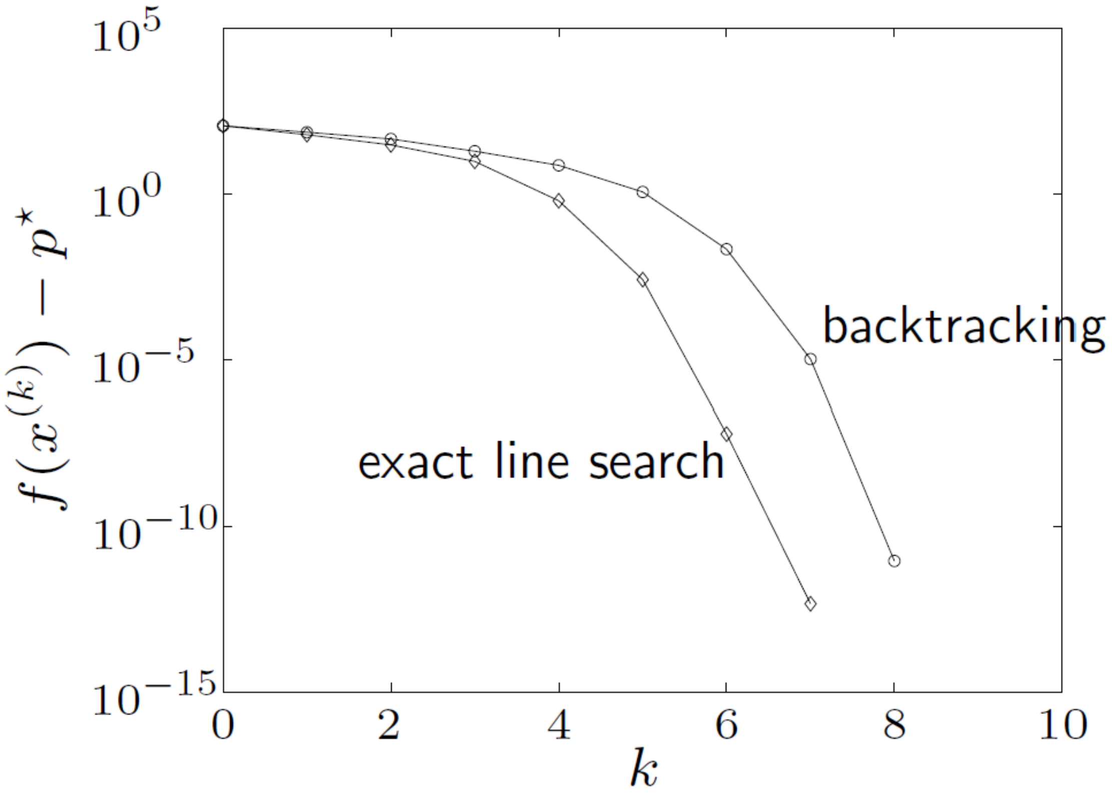


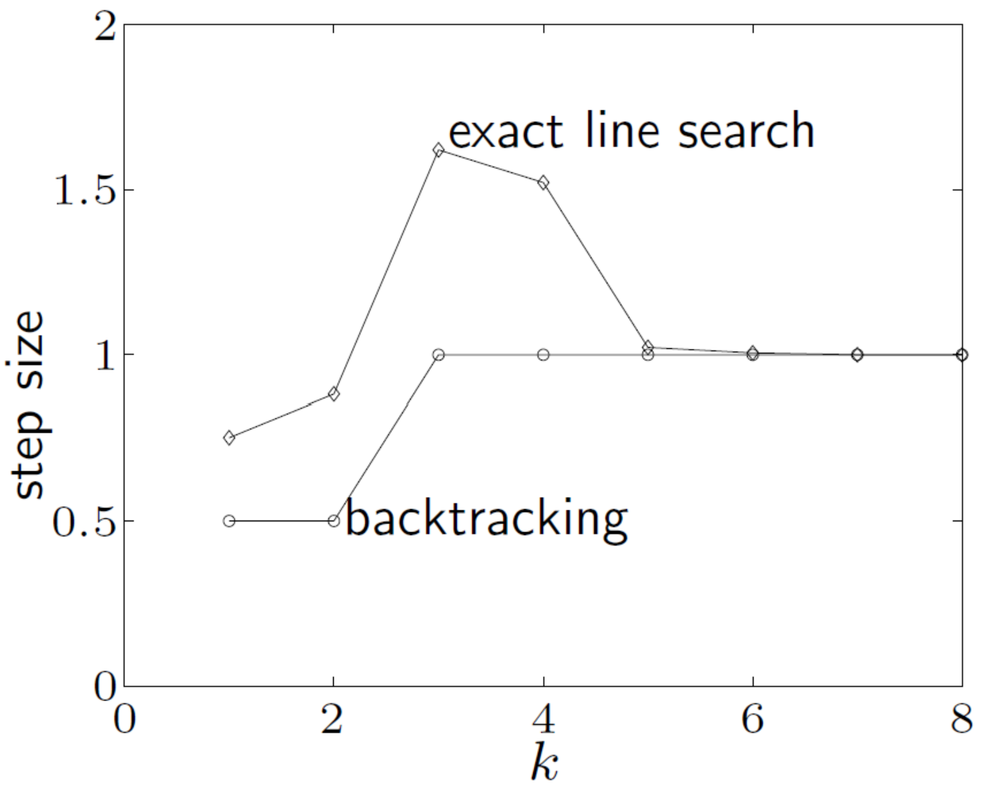


backtracking parameters $\gamma = 0.1$, $\beta = 0.7$

Unconstrained Minimization
Newton's Method (Second-order method)

Backtracking line search is almost as fast as exact line search (and much simpler)





backtracking parameters $\gamma = 0.1$, $\beta = 0.7$

Unconstrained Minimization

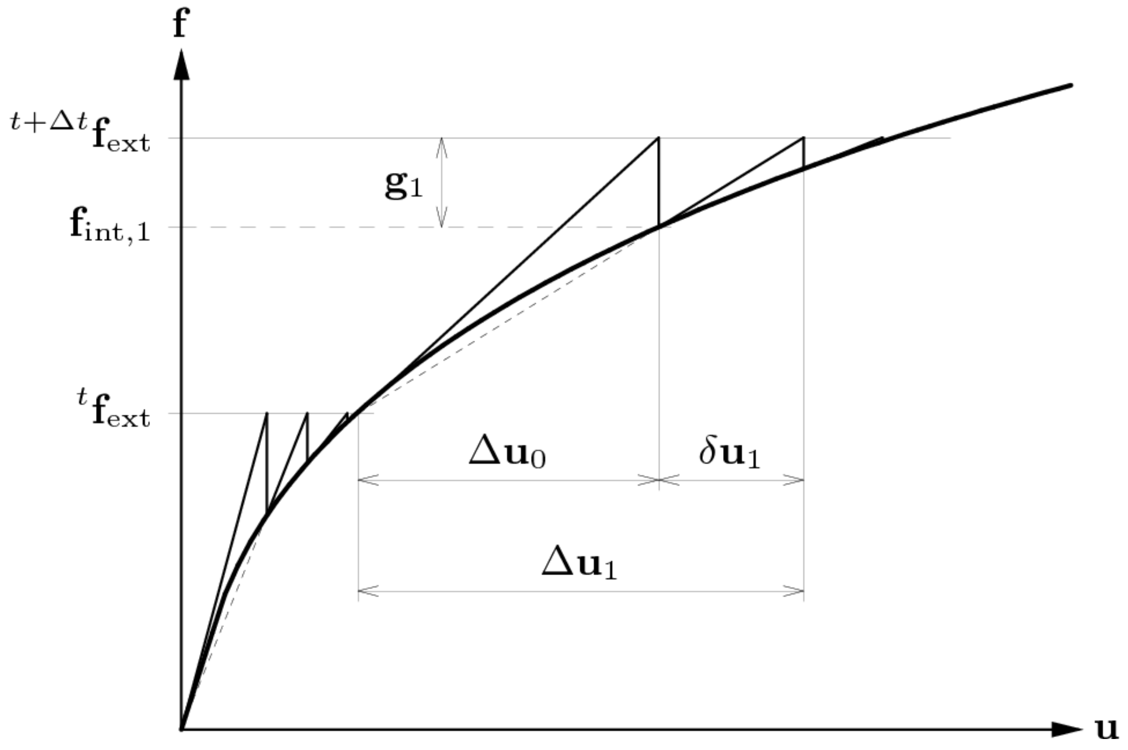
Quasi-Newton's Method (Second-order method)

$$\underline{x}^{k+1} = \underline{x}^k + \alpha^k \underline{\Delta x}^k$$

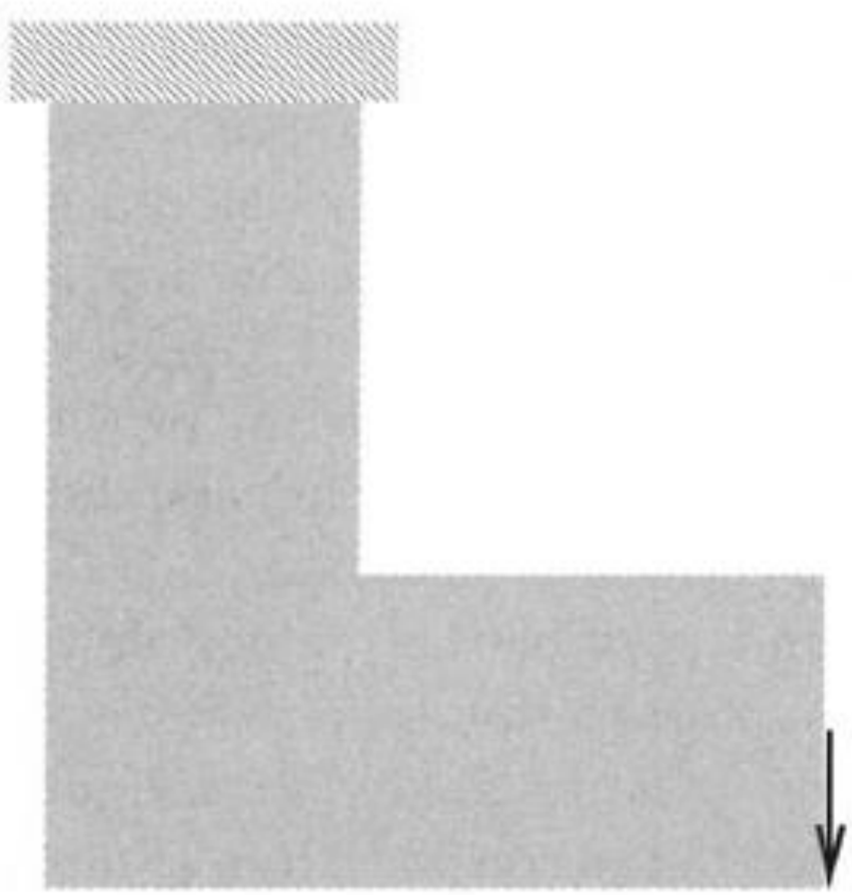
Where $\underline{\Delta x}^k = -\underline{Q}\,\nabla f(\underline{x}^k)$

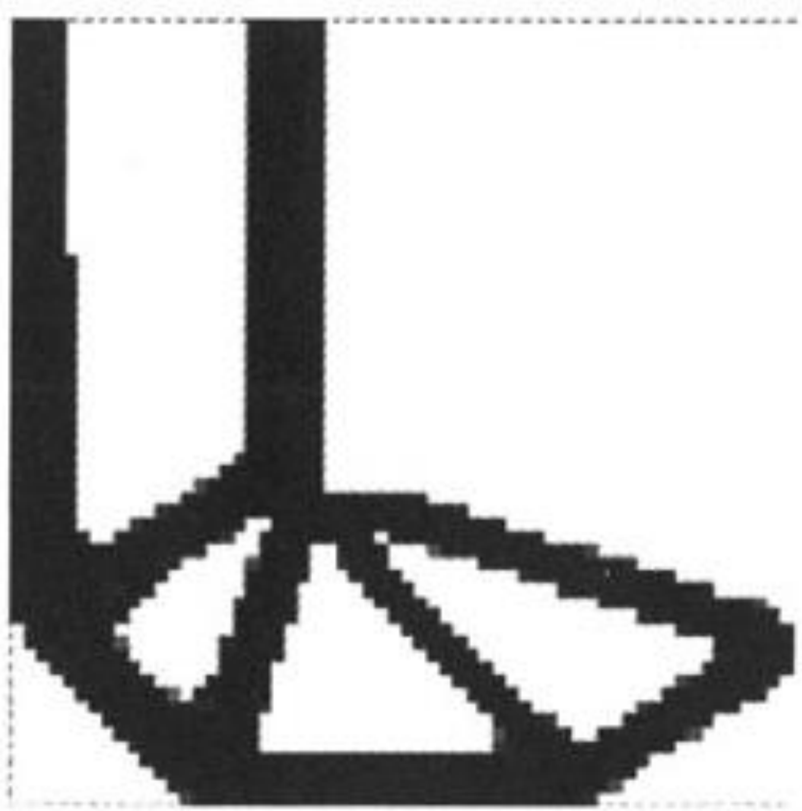
$\underline{Q}$ approximates $\nabla^2 f\left(\underline{x}^k\right)^{-1}$

Many different methods proposed for the approximation.



Topology Optimization
General Setup for Density Based Approach





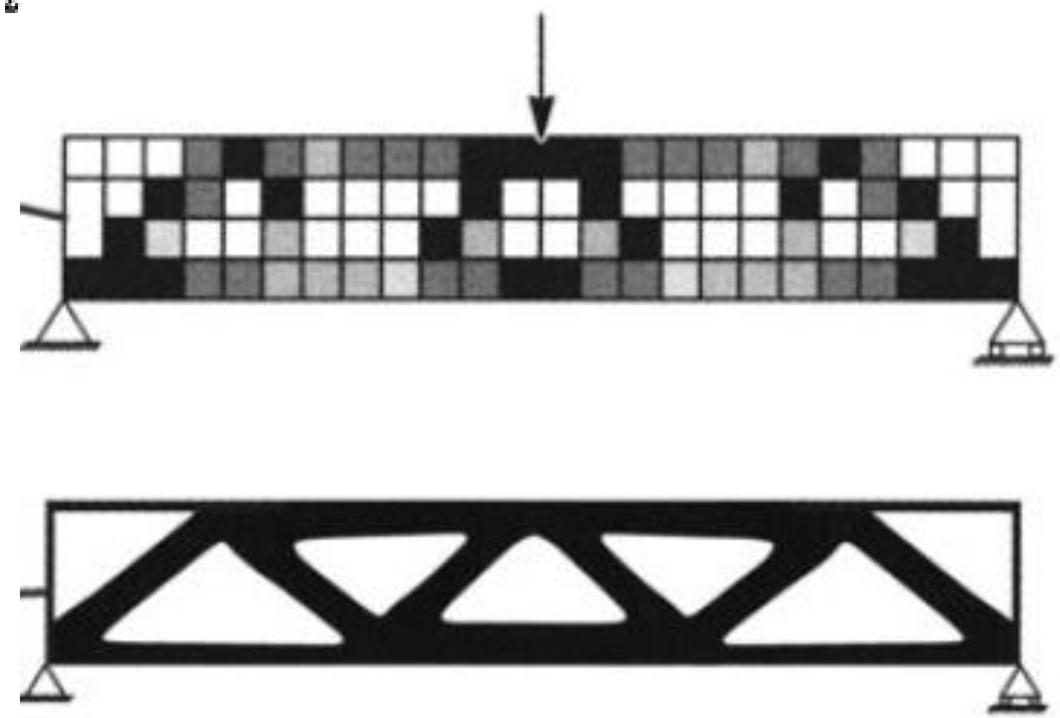
Topology Optimization
General Setup for Density Based Approach

$\boxed{K} u = f$

SIMP: Solid Isotropic Material with Penalization.

$$E_{ijkl}(x) = \rho(x)^p E^0_{ijkl}, \quad p > 1\,,$$

$$\int_\Omega \rho(x) \mathrm{d}\Omega \leq V; \quad 0 \leq \rho(x) \leq 1, \quad x \in \Omega\,,$$



Topology Optimization

General Setup for Density Based Approach

$$\min_{\mathbf{u},\rho_e} \mathbf{f}^T\mathbf{u}$$

Compliance $:= \frac{1}{\text{Stiffness}}$

1:20

$$\text{s.t.}: \left(\sum_{e=1}^{N} \rho_e^p \mathbf{K}_e\right)\mathbf{u} = \mathbf{f},$$

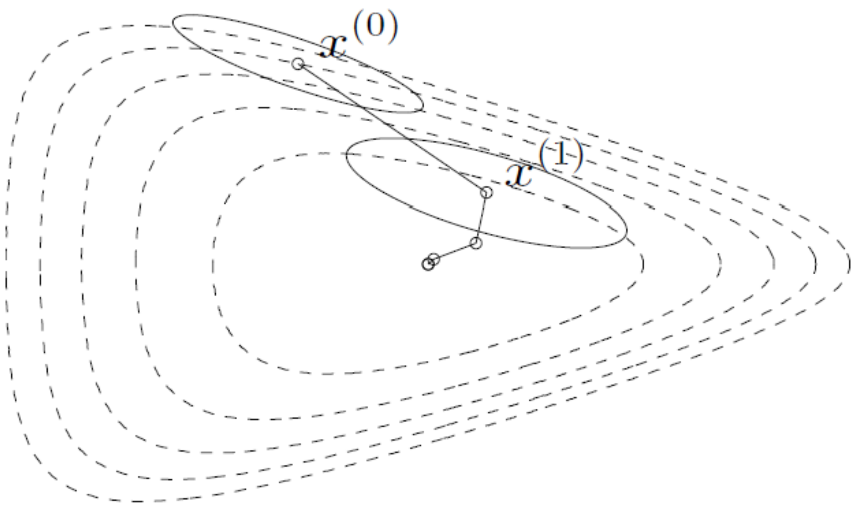
$$\sum_{e=1}^{N} v_e\rho_e \le V, \quad 0 < \rho_{\min} \le \rho_e \le 1, \quad e = 1, \ldots, N.$$

Given material volume, how can we distribute materials to minimize compliance (i.e., maximize stiffness) ?

Sensitivity Analysis
Overview

Secant Search

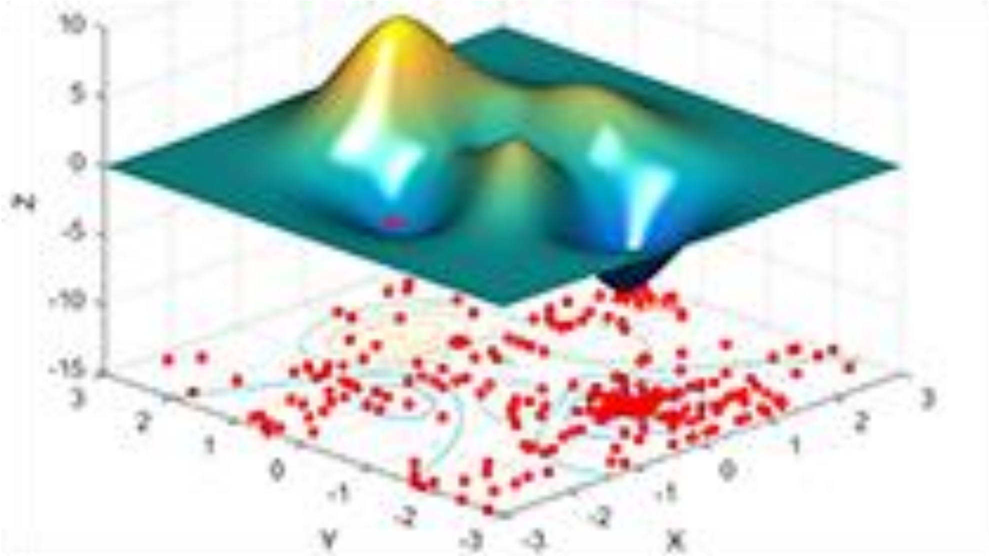
(Line search/gradient-based)



Quick convergence

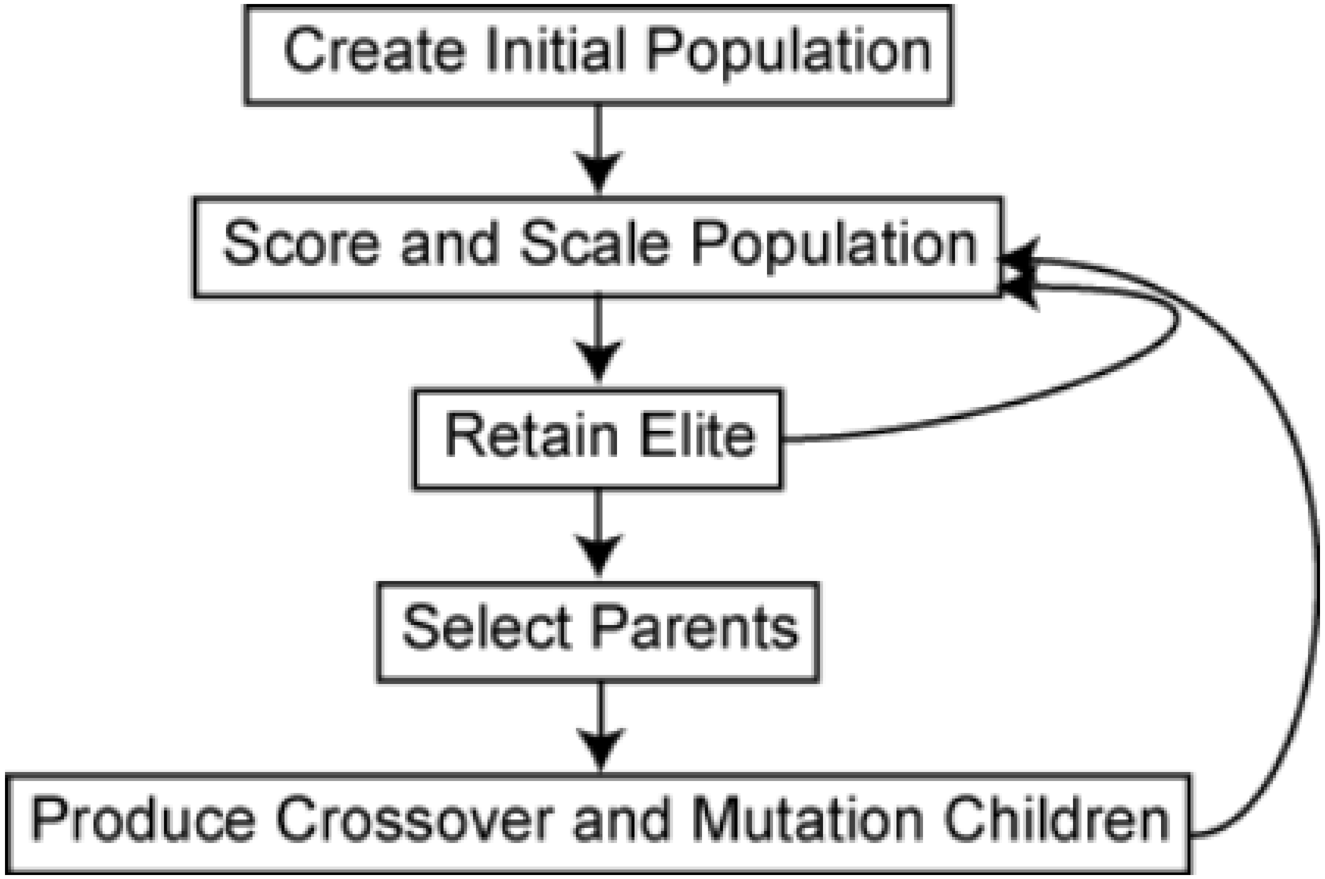
Probabilistic Search

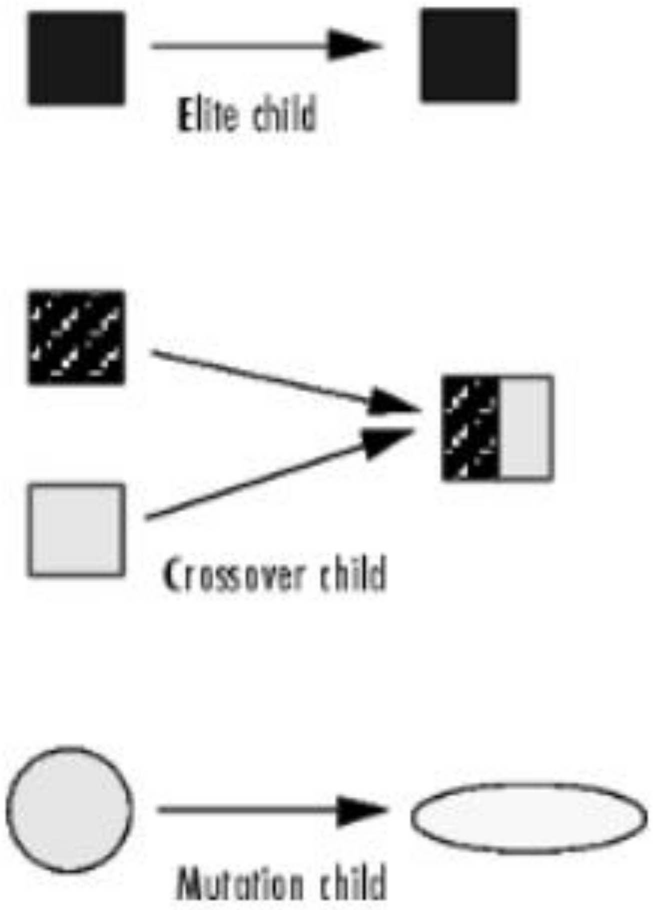
(Genetic Optimization/evolutionary)



Objective function is discontinuous, nondifferentiable, stochastic, or highly nonlinear

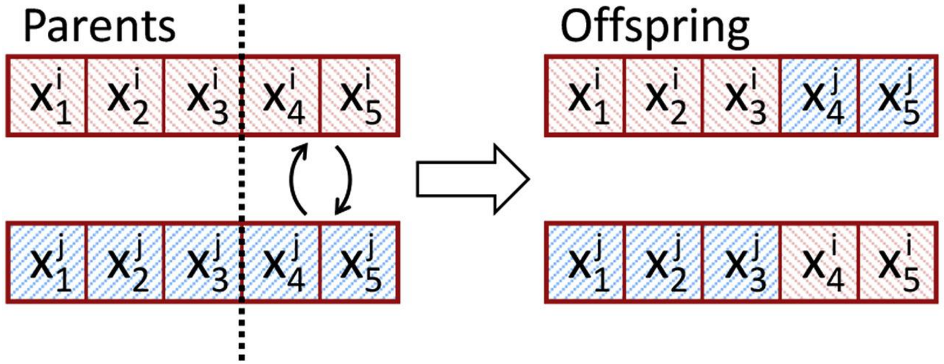
Genetic Optimization
Overview



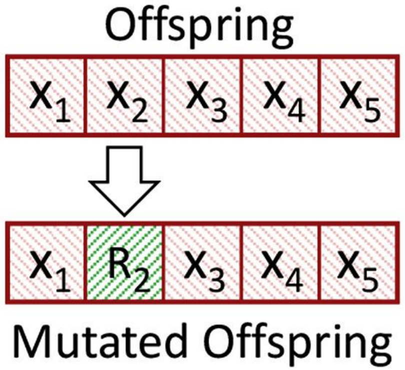


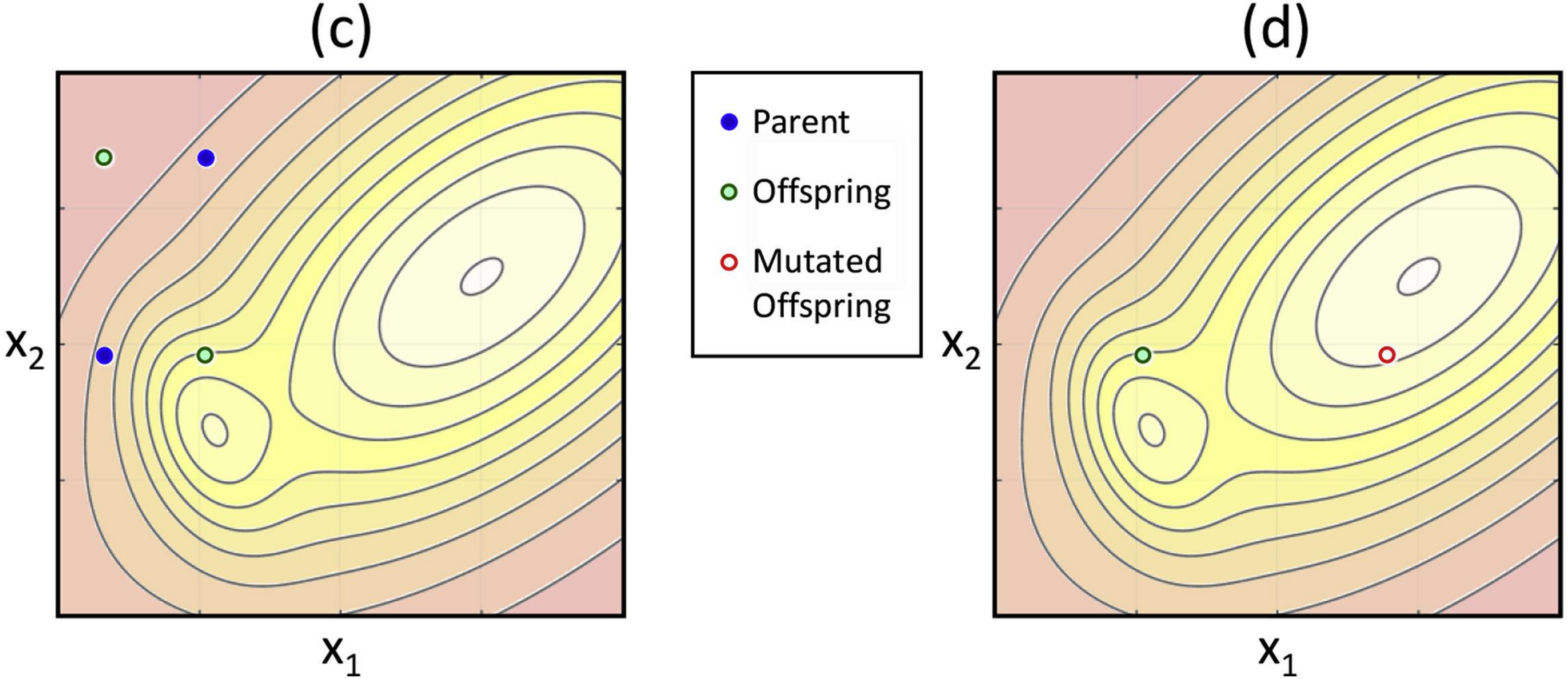
Genetic Optimization
Overview

Crossover



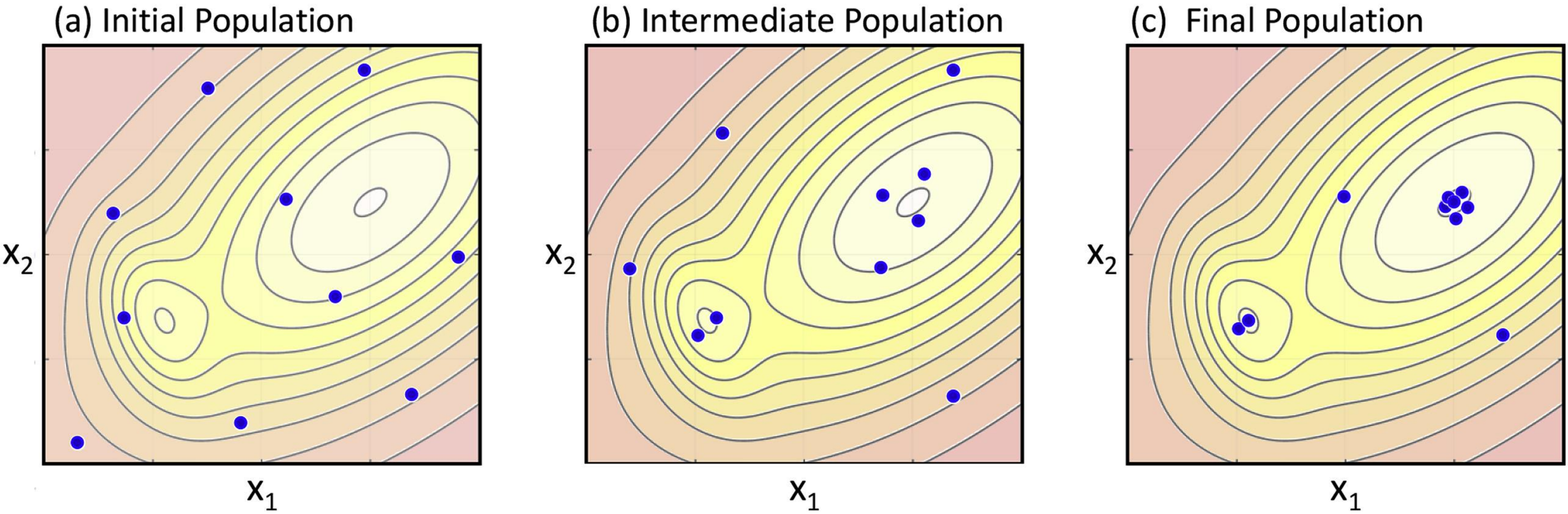
Mutation





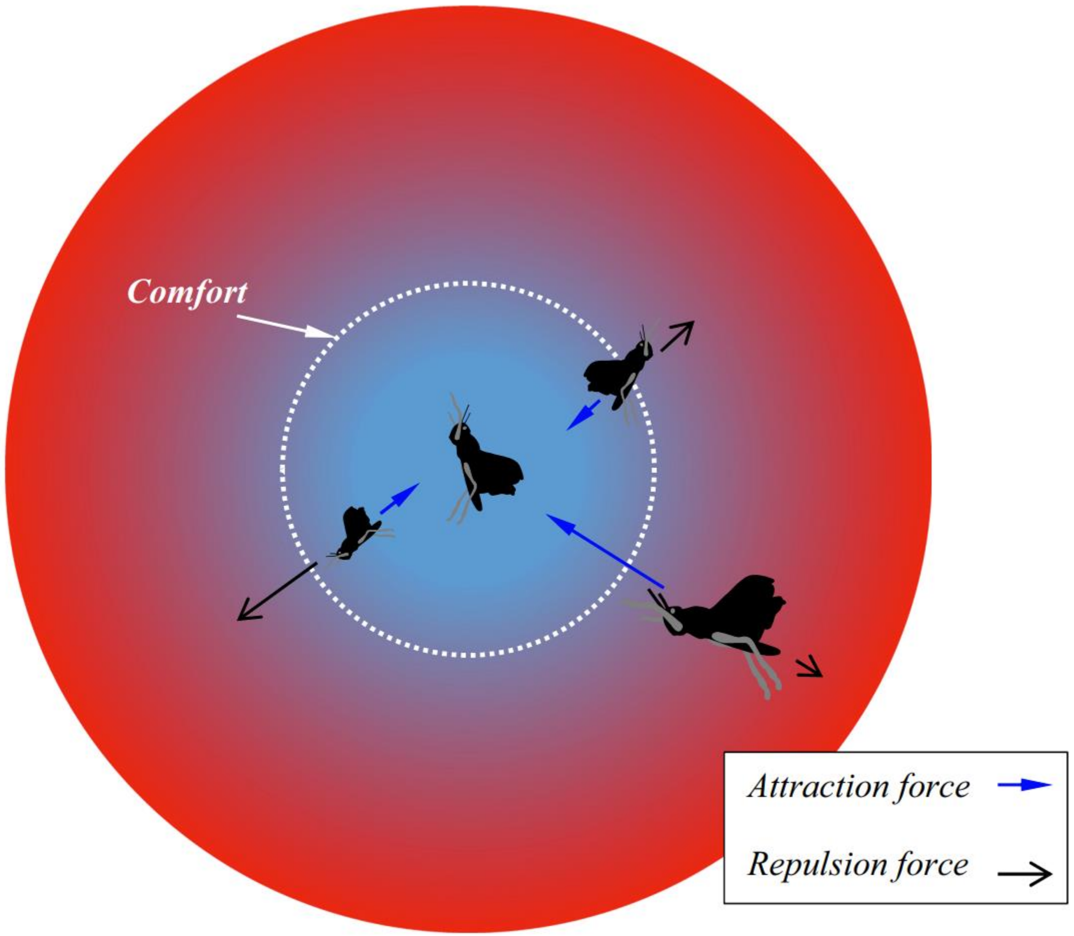
https://doi.org/10.1016/j.envsoft.2018.11.018

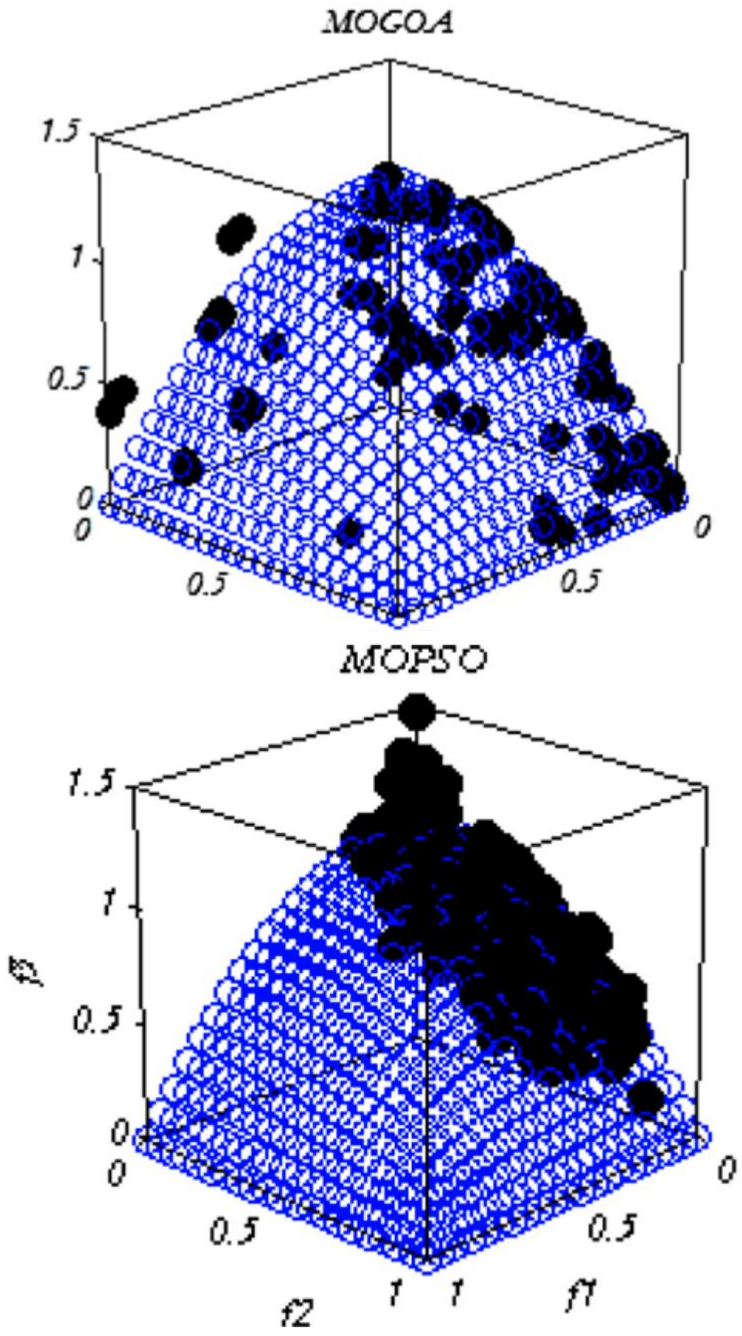
Genetic Optimization
Overview



https://doi.org/10.1016/j.envsoft.2018.11.018

Genetic Optimization
Swarm-intelligence: Grasshopper





DOI 10.1007/s10489-017-1019-8