Introduction to shape optimization

Frédéric de Gournay

Institut de Mathématiques de Toulouse.

14 mai 2017

Scope

Shape optimization : optimize a problem where the unknwon is the domain and the objective function depends on the solution of a PDE.

$$\min_{\mathcal{A}_\Omega u = f, \Omega \in \mathcal{U}} \mathcal{J}(u, \Omega)$$

Scope

Shape optimization : optimize a problem where the unknwon is the domain and the objective function depends on the solution of a PDE.

$$\min_{\mathcal{A}_\Omega u = f, \Omega \in \mathcal{U}} \mathcal{J}(u, \Omega)$$

Choice of PDE / objective function

- Optimization : static elasticity (compliance, the Von Misses stress), Eigenvalue / eigenvectors (frequency gap problem), buckling, time-dependent
- Inverse problem : is an optimization problem via the least-square method (where $\mathcal{O}$ is the observation operator)

$$J(u, \Omega) = \|\mathcal{O}(u) - data\|^2$$

Scope

Shape optimization : optimize a problem where the unknwon is the domain and the objective function depends on the solution of a PDE.

$$\min_{\mathcal{A}_\Omega u=f,\Omega\in\mathcal{U}} \mathcal{J}(u,\Omega)$$

Choice of PDE / objective function

- Optimization : static elasticity (compliance, the Von Misses stress), Eigenvalue / eigenvectors (frequency gap problem), buckling, time-dependent
- Inverse problem : is an optimization problem via the least-square method (where $\mathcal{O}$ is the observation operator)

$$J(u,\Omega) = \|\mathcal{O}(u) - data\|^2$$

Choice of Ω : three main techniques

- Parametric optimization
- Move the boundary
- Drill holes
- Perform mixture at a microscopic level of two different materials (homogenization).

Main objectives

$$\min_{\mathcal{A}_\Omega u = f, \Omega \in \mathcal{U}} \mathcal{J}(u, \Omega)$$

1. Prove existence of minimizers (modify $\mathcal{U}$ or $\mathcal{J}$ if needed).
2. Prove uniqueness of minimizer (almost impossible).
3. Find $\nabla_\Omega u$ (doable) and D, the gradient of $\Omega \mapsto J(u_\Omega, \Omega)$.
4. Perform first order minimisation procedure.

$$\Omega_{k+1} = \Omega_k - sD$$

5. Prove convergence of algorithm.
6. Implement the algorithm.

Shape sensitivity

From an initial domain Ω_k, the set of admissible domains $\mathcal{U}$ is the set

$$\Omega \in \mathcal{U} \Leftrightarrow \exists T \text{ diffeomorphism s.t. } \Omega = T(\Omega_k) = \{T(x), x \in \Omega\}$$

Advantages

- Endows $\mathcal{U}$ with a vector space structure (manifold), the tangent space is the space of vector fields θ, since for T small

$$T(x) = Id(x) + \theta(x)$$

- Perform a change of variable and study the mapping $\theta \mapsto \mathcal{J}(u_\theta, \theta)$.

Disadvantages

- The structure is the one of a manifold and it is difficult to perform optimization on a manifold.
- The structure does not support changes of topology (creating or removing holes).

Derivation of the solution of a PDE

Idea

Use the implicit function theorem that states if $u(\theta)$ is the unique solution of

$$B(\theta)u(\theta) = f(\theta)$$

then denoting by a prime the derivative in 0 wrto θ

$$Bu' + B'u = f'$$

Derivation of the solution of a PDE

Idea

Use the implicit function theorem that states if $u(\theta)$ is the unique solution of

$$B(\theta)u(\theta) = f(\theta)$$

then denoting by a prime the derivative in 0 wrto θ

$$Bu' + B'u = f'$$

In the laplacian setting

if

$$-\Delta u_T = f \text{ on } T(\Omega) \text{ and } u_T = 0 \text{ on } \partial T(\Omega),$$

Derivation of the solution of a PDE

Idea

Use the implicit function theorem that states if $u(\theta)$ is the unique solution of

$$B(\theta)u(\theta) = f(\theta)$$

then denoting by a prime the derivative in 0 wrto θ

$$Bu' + B'u = f'$$

In the laplacian setting

if

$$-\Delta u_T = f \text{ on } T(\Omega) \text{ and } u_T = 0 \text{ on } \partial T(\Omega),$$

set $\bar{u}_T = u \circ T$. Then

$$-divA(T)\nabla \bar{u}_T = (-\Delta u_T) \circ T = f \circ T \text{ on } \Omega \text{ and } \bar{u}_T = 0 \text{ on } \partial\Omega$$

Derivation of the solution of a PDE

Idea

Use the implicit function theorem that states if $u(\theta)$ is the unique solution of

$$B(\theta)u(\theta) = f(\theta)$$

then denoting by a prime the derivative in 0 wrto θ

$$Bu' + B'u = f'$$

In the laplacian setting

if

$$-\Delta u_T = f \text{ on } T(\Omega) \text{ and } u_T = 0 \text{ on } \partial T(\Omega),$$

set $\bar{u}_T = u \circ T$. Then

$$-divA(T)\nabla \bar{u}_T = (-\Delta u_T) \circ T = f \circ T \text{ on } \Omega \text{ and } \bar{u}_T = 0 \text{ on } \partial\Omega$$

and

$$-\Delta \bar{u}' - divA'\nabla u = f'$$

So that

$$-\Delta \bar{u}' = f' + divA'\nabla u \text{ and } \bar{u}' = 0 \text{ on } \partial\Omega$$

Derivation of the objective function

Idea

Objective : find $\mathcal{J}'$ the derivative of $\mathcal{J}(u_T, T)$ with $Bu' = f' - B'u$, using

$$\mathcal{J}' = \partial_u \mathcal{J}.u' + \partial_T \mathcal{J}.$$

Derivation of the objective function

Idea

Objective : find $\mathcal{J}'$ the derivative of $\mathcal{J}(u_T, T)$ with $Bu' = f' - B'u$, using

$$\mathcal{J}' = \partial_u \mathcal{J}.u' + \partial_T \mathcal{J}.$$

Introduce p the adjoint that solves $(Bx, p) = \partial_u \mathcal{J}.x$ for all x, then

$$\mathcal{J}' = (f' - B'u, p) + \partial_T \mathcal{J}$$

Derivation of the objective function

Idea

Objective : find $\mathcal{J}'$ the derivative of $\mathcal{J}(u_T, T)$ with $Bu' = f' - B'u$, using

$$\mathcal{J}' = \partial_u \mathcal{J}.u' + \partial_T \mathcal{J}.$$

Introduce p the adjoint that solves $(Bx, p) = \partial_u \mathcal{J}.x$ for all x, then

$$\mathcal{J}' = (f' - B'u, p) + \partial_T \mathcal{J}$$

In the laplacian setting

$$\mathcal{J}(u_T, T) = \int_{T(\Omega)} h(x)|u_T(x) - d(x)|^2 dx$$

Derivation of the objective function

Idea

Objective : find $\mathcal{J}'$ the derivative of $\mathcal{J}(u_T, T)$ with $Bu' = f' - B'u$, using

$$\mathcal{J}' = \partial_u \mathcal{J}.u' + \partial_T \mathcal{J}.$$

Introduce p the adjoint that solves $(Bx, p) = \partial_u \mathcal{J}.x$ for all x, then

$$\mathcal{J}' = (f' - B'u, p) + \partial_T \mathcal{J}$$

In the laplacian setting

$$\mathcal{J}(u_T, T) = \int_{T(\Omega)} h(x)|u_T(x) - d(x)|^2 dx = \int_\Omega h \circ T|\bar{u}_T - d \circ T|^2 |det \nabla T|$$

$$\partial_u \mathcal{J}.z = 2\int_\Omega huz$$

Taking the L^2 scalar product with $-\Delta \bar{u}' = f' + divA'\nabla u$

$$-\Delta p = 2huz \text{ and } p = 0 \text{ on } \partial\Omega$$

and

$$\mathcal{J}' = \int_\Omega (f' + divA'\nabla u)p + \partial_T \mathcal{J}$$

Actually there exists a pressure π such that, if $\mathcal{J}'$ is the derivative in direction θ

$$\mathcal{J}' = \int_{\partial\Omega} (\theta \cdot n)\pi dx$$

Actually there exists a pressure π such that, if $\mathcal{J}'$ is the derivative in direction θ

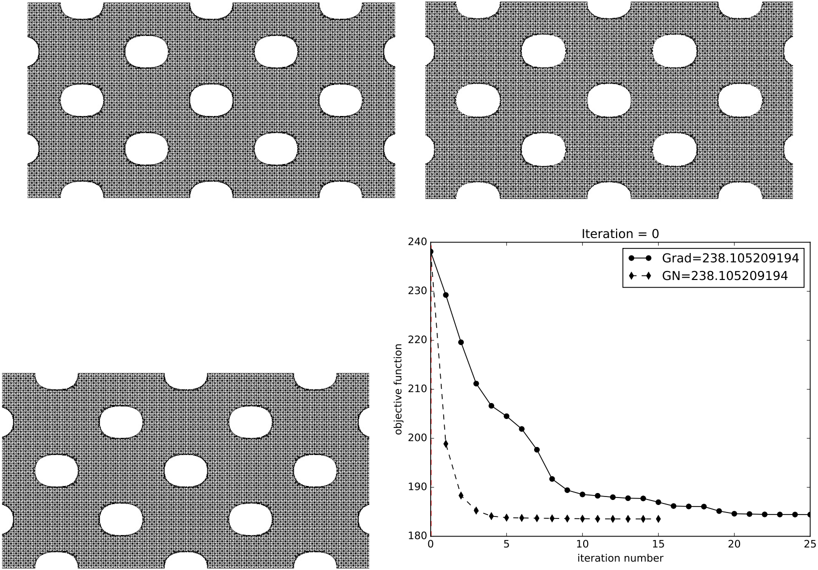
$$\mathcal{J}' = \int_{\partial\Omega} (\theta \cdot n)\pi dx$$

Exemples

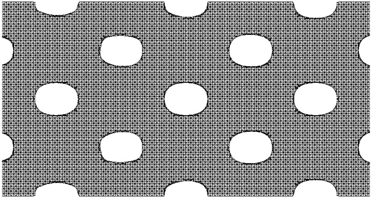
- $\mathcal{J}(\Omega) = \int_\Omega f(x)dx \quad \mathcal{J}' = \int_{\partial\Omega} (\theta \cdot n) f$
- $\mathcal{J}(\Omega) = \int_{\partial\Omega} f(x)dx \quad \mathcal{J}' = \int_{\partial\Omega} (\theta \cdot n)(\partial_n f + Hf)$
- u_Ω is solution of Laplacian with Dirichlet and $\mathcal{J}(\Omega) = \int_\Omega h|u_\Omega - d|^2$ then
$$\mathcal{J}' = \int_{\partial\Omega} (\theta \cdot n)(h|u_\Omega - d|^2 - \nabla p \cdot \nabla u)$$
- u_Ω is solution of linear elasticity with Dirichlet and $\mathcal{J}(\Omega) = \int_\Omega h|u_\Omega - d|^2$ then
$$\mathcal{J}' = \int_{\partial\Omega} (\theta \cdot n)(h|u_\Omega - d|^2 - Ae(p) : e(u))$$
- u_Ω is solution of linear elasticity with Dirichlet and $\mathcal{J}(\Omega) = \int_\Omega Ae(u) : e(u)$ then
$$\mathcal{J}' = \int_{\partial\Omega} (\theta \cdot n)(Ae(u) : e(u))$$
- u_Ω is solution of linear elasticity with Neumann + no source term close to the boundary and $\mathcal{J}(\Omega) = \int_\Omega Ae(u) : e(u)$ then
$$\mathcal{J}' = \int_{\partial\Omega} (\theta \cdot n)(-Ae(u) : e(u))$$

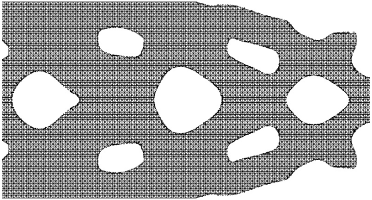
Numerical implementation

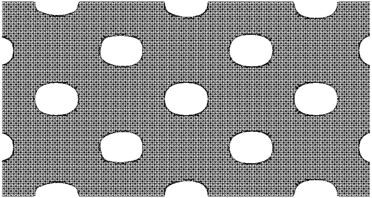
- For a given domain, compute the solution, compute the pressure and transform the domain according to the pressure π.
- Or introduce a level-set technique if you want to allow for changes.

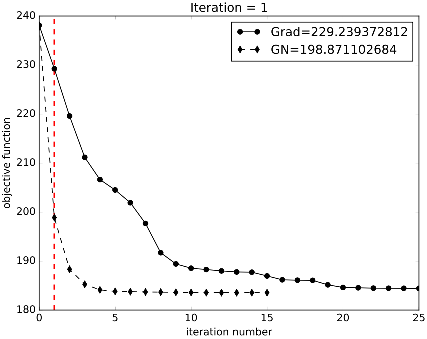


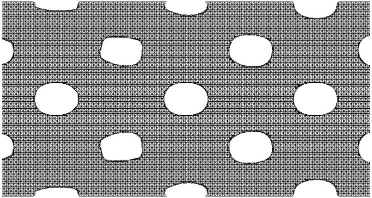
Iteration = 0
Grad=238.105209194
GN=238.105209194
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

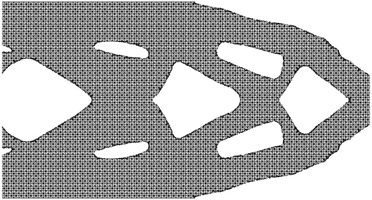


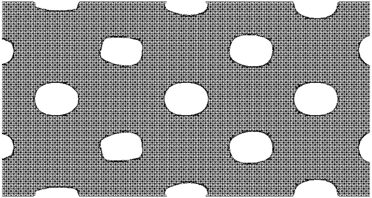


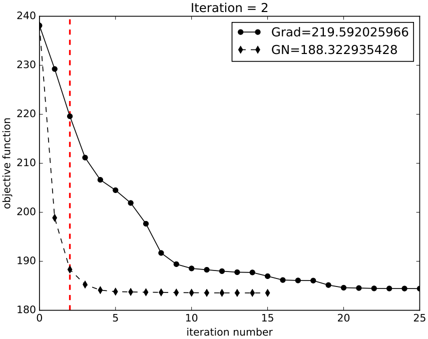


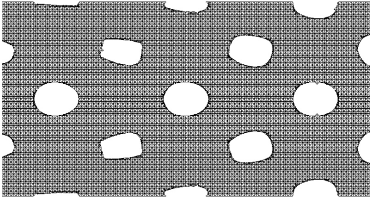

Iteration = 1
Grad=229.239372812
GN=198.871102684
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

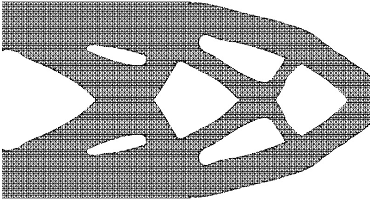


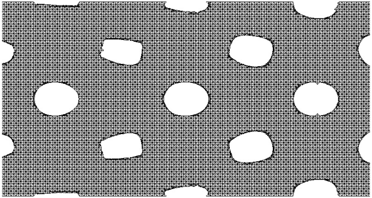


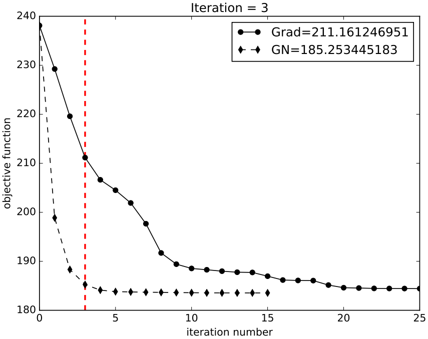


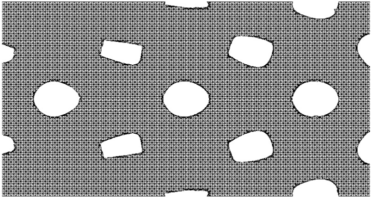

Iteration = 2
Grad=219.592025966
GN=188.322935428
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

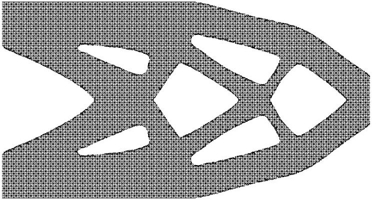


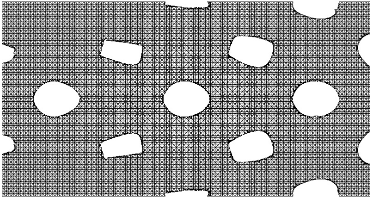


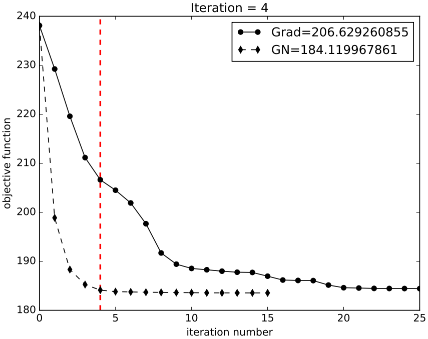


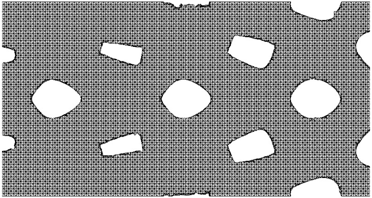

Iteration = 3
Grad=211.161246951
GN=185.253445183
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

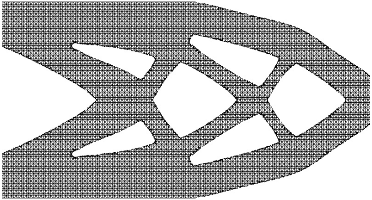


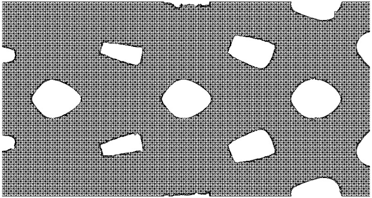


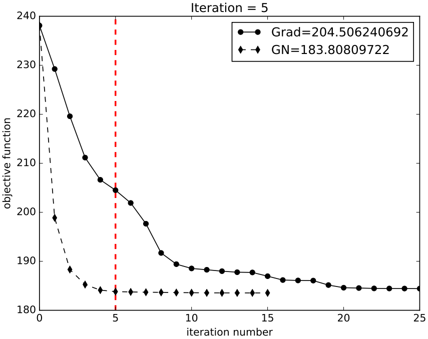


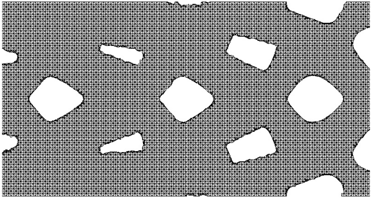

Iteration = 4
Grad=206.629260855
GN=184.119967861
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

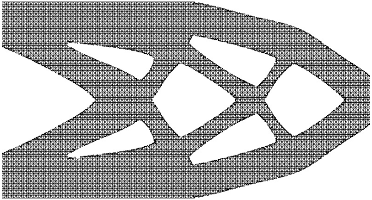


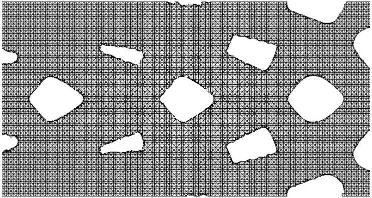


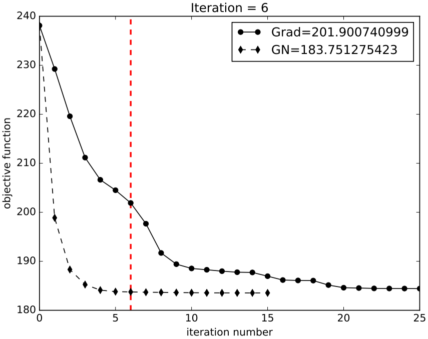


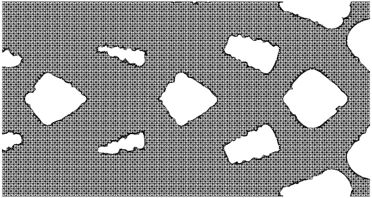

Iteration = 5
Grad=204.506240692
GN=183.80809722
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

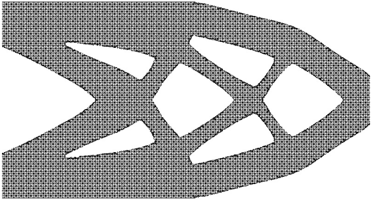


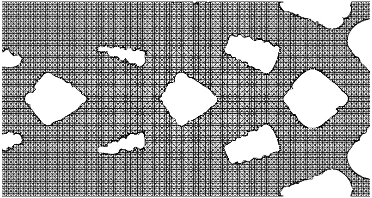


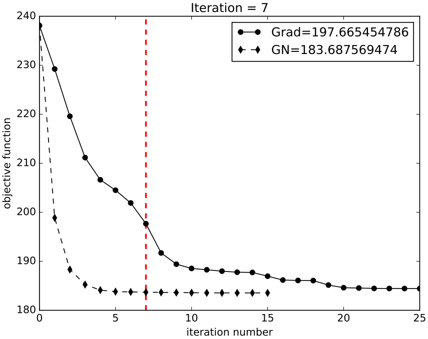



Iteration = 6
Grad=201.900740999
GN=183.751275423
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

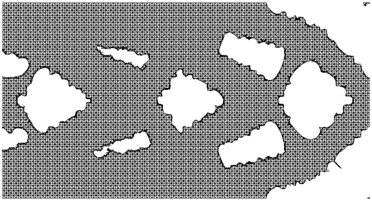


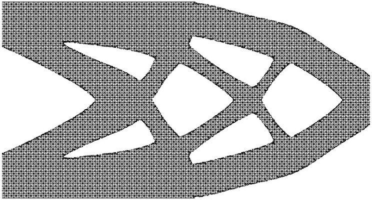


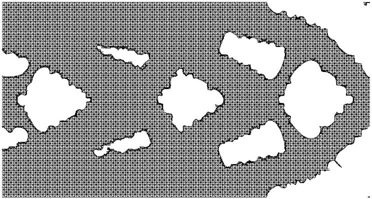


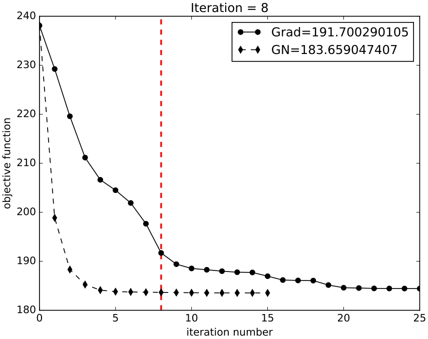


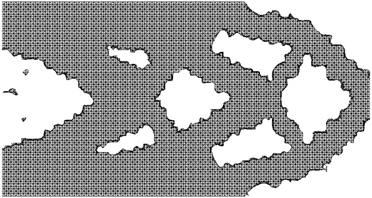
Iteration = 7
Grad=197.665454786
GN=183.687569474
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

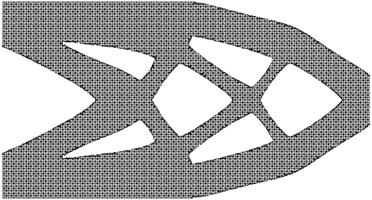


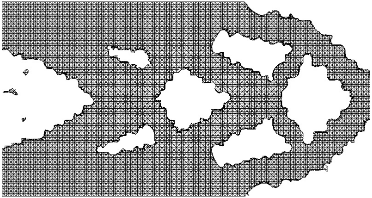


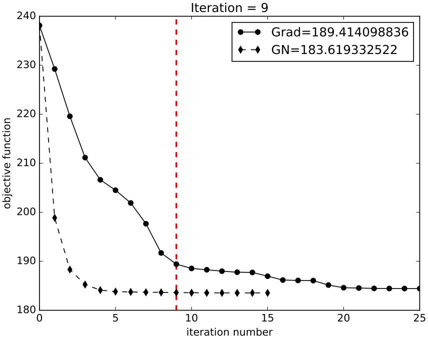



Iteration = 8
Grad=191.700290105
GN=183.659047407
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

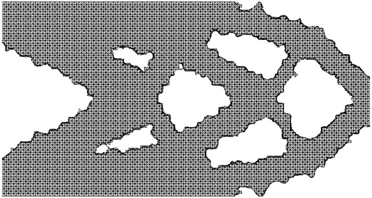


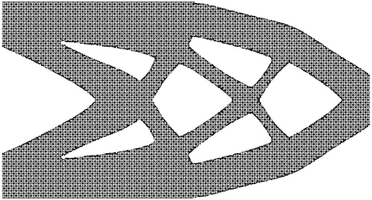


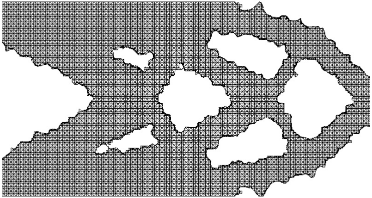


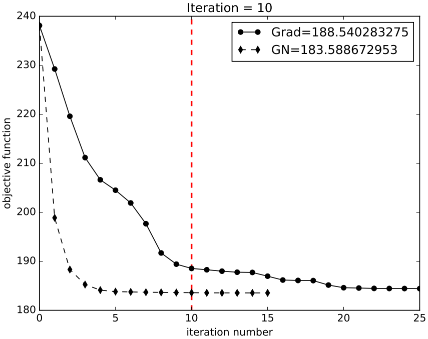


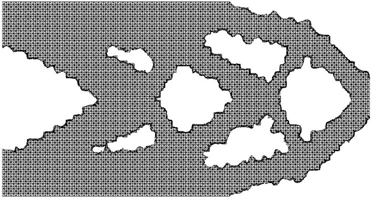
Iteration = 9
Grad=189.414098836
GN=183.619332522
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

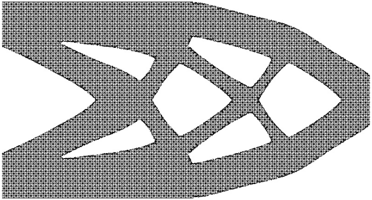


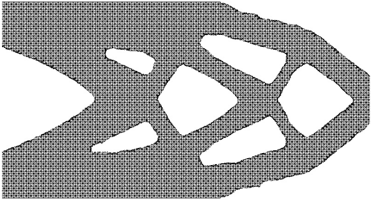


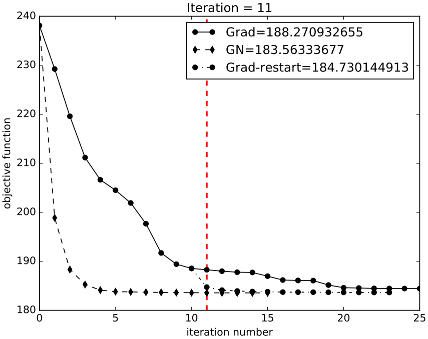


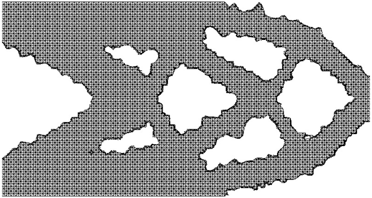

Iteration = 10
Grad=188.540283275
GN=183.588672953
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

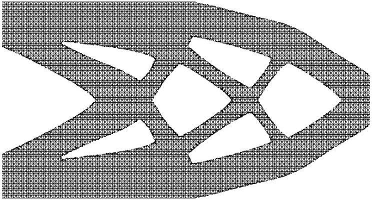


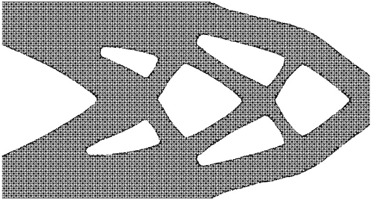


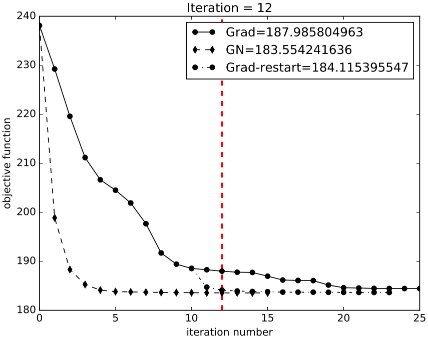



Iteration = 11
Grad=188.270932655
GN=183.56333677
Grad-restart=184.730144913
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

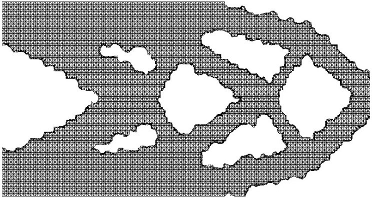


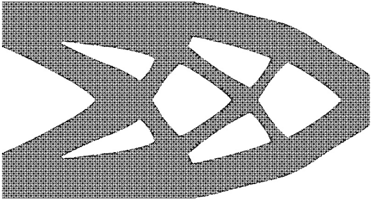


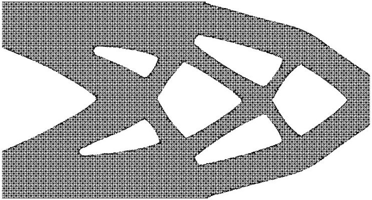


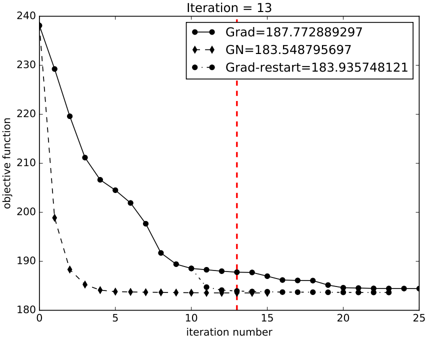


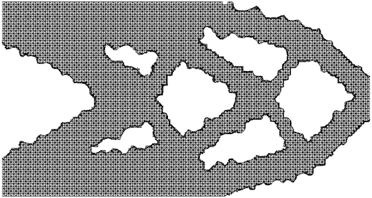
Iteration = 12
Grad=187.985804963
GN=183.554241636
Grad-restart=184.115395547
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

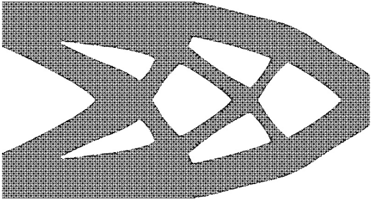


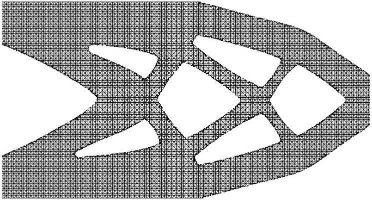


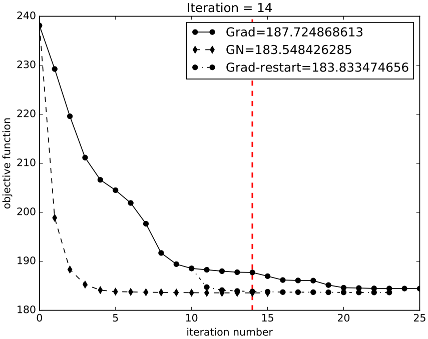


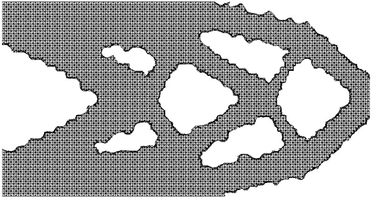

Iteration = 13
Grad=187.772889297
GN=183.548795697
Grad-restart=183.935748121
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

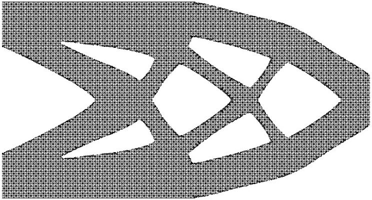


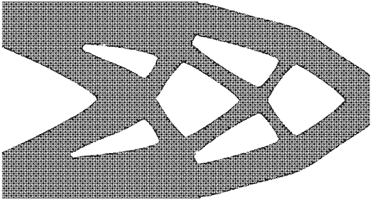


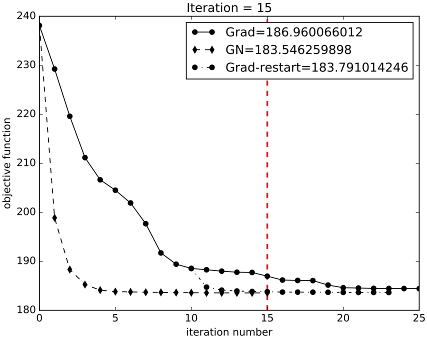


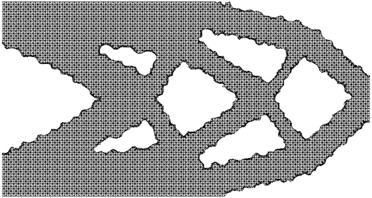

Iteration = 14
Grad=187.724868613
GN=183.548426285
Grad-restart=183.833474656
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

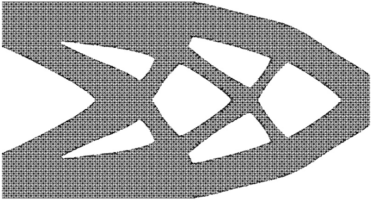


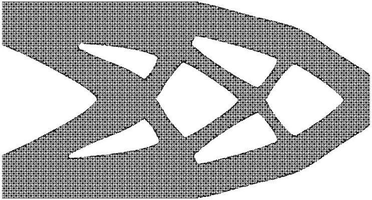


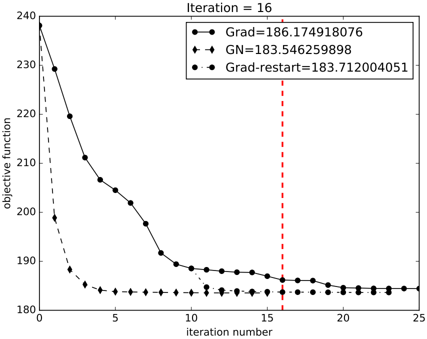


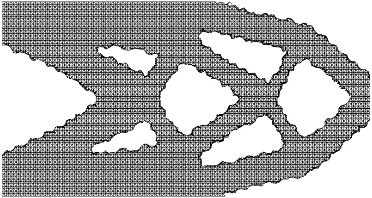

Iteration = 15
Grad=186.960066012
GN=183.546259898
Grad-restart=183.791014246
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

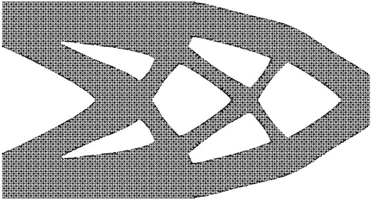


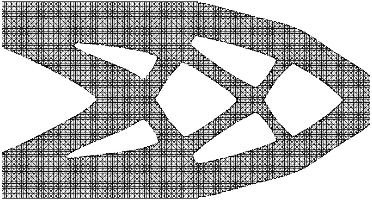


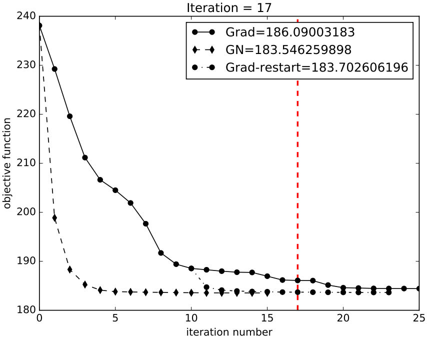


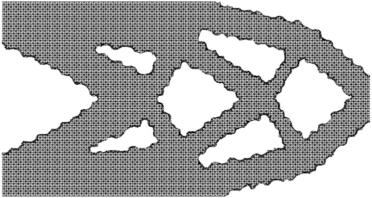

Iteration = 16
Grad=186.174918076
GN=183.546259898
Grad-restart=183.712004051
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

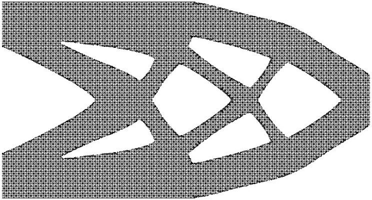


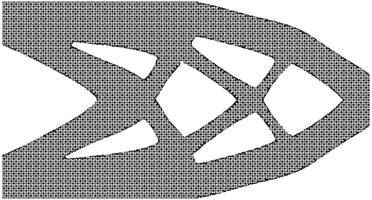


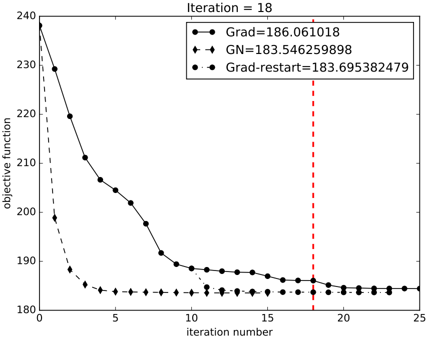


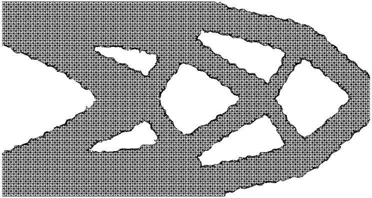

Iteration = 17
Grad=186.09003183
GN=183.546259898
Grad-restart=183.702606196
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

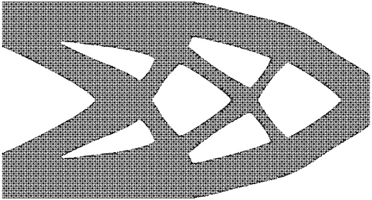


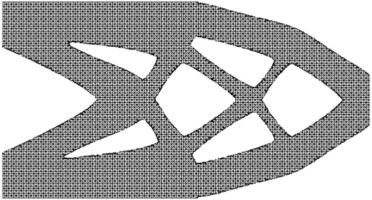


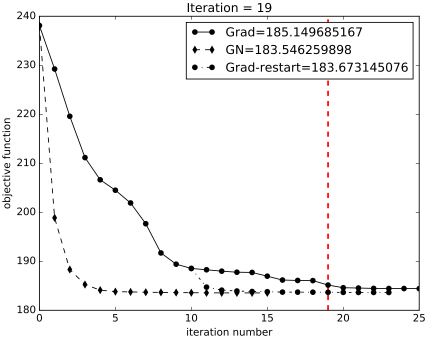


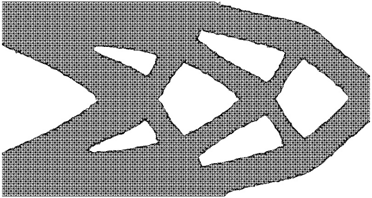

Iteration = 18
Grad=186.061018
GN=183.546259898
Grad-restart=183.695382479
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

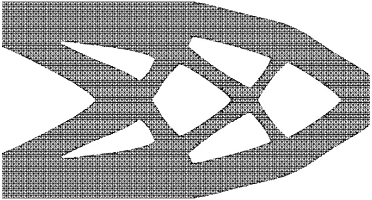


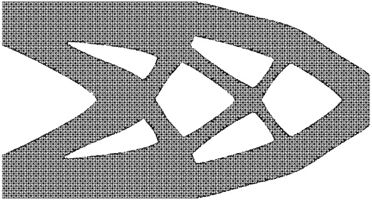


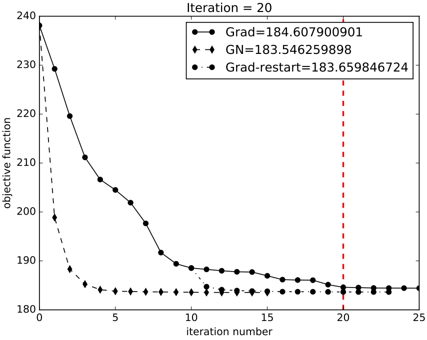


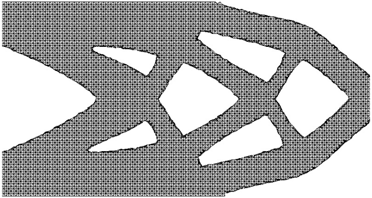

Iteration = 19
Grad=185.149685167
GN=183.546259898
Grad-restart=183.673145076
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

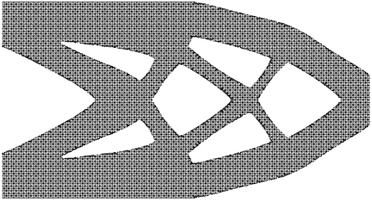


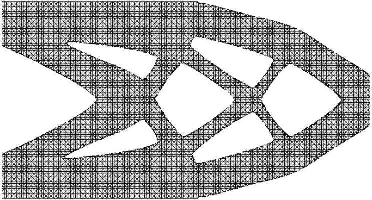


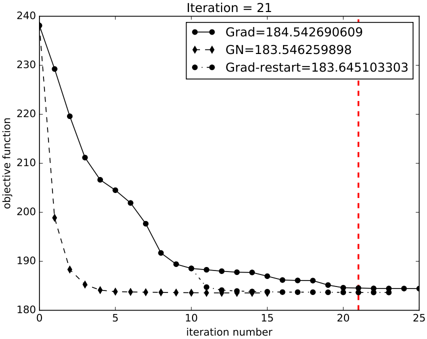


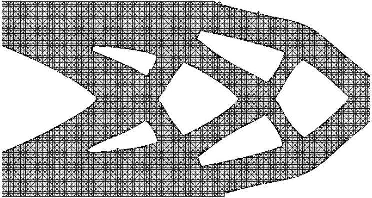

Iteration = 20
Grad=184.607900901
GN=183.546259898
Grad-restart=183.659846724
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

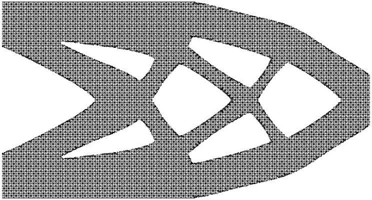


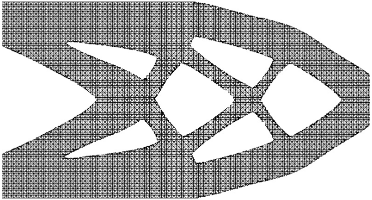


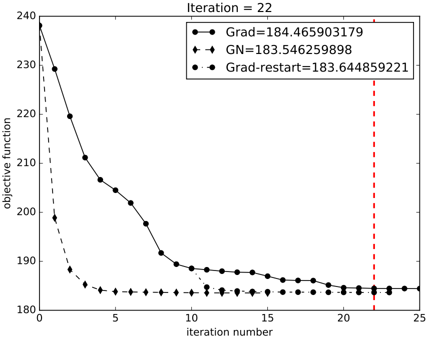


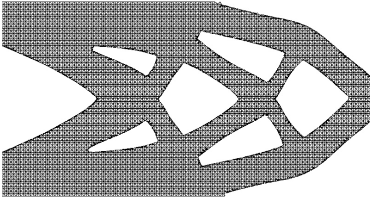

Iteration = 21
Grad=184.542690609
GN=183.546259898
Grad-restart=183.645103303
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

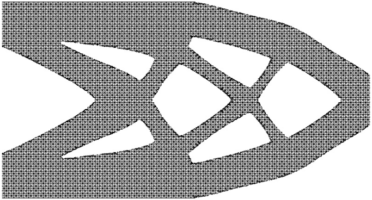


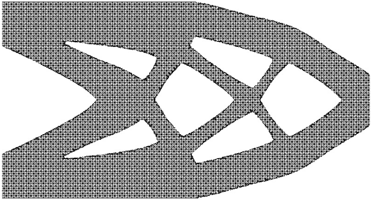


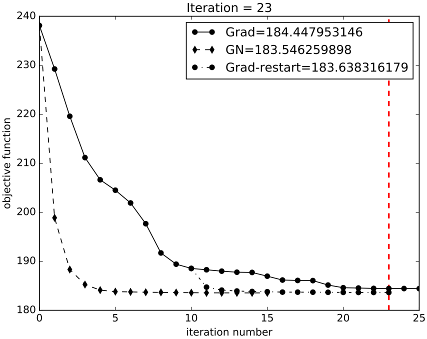


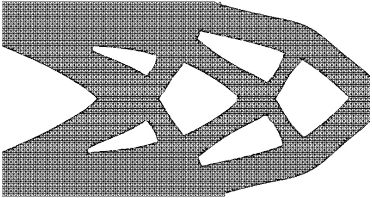

Iteration = 22
Grad=184.465903179
GN=183.546259898
Grad-restart=183.644859221
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

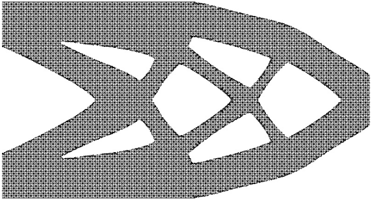


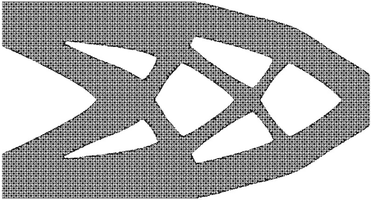


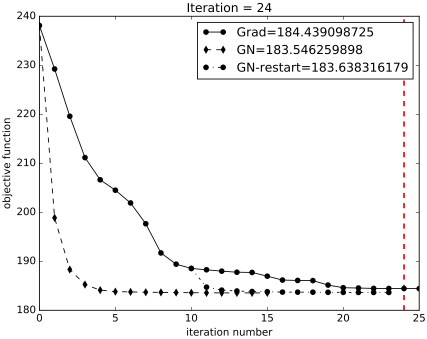


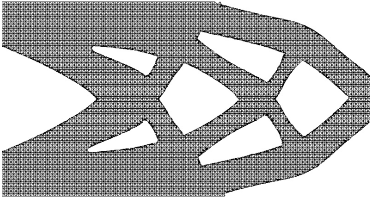

Iteration = 23
Grad=184.447953146
GN=183.546259898
Grad-restart=183.638316179
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25

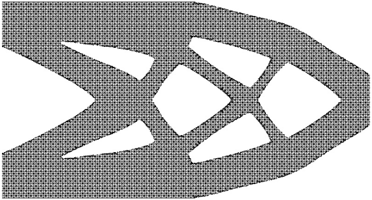


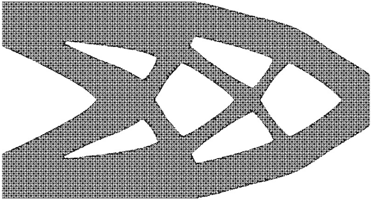


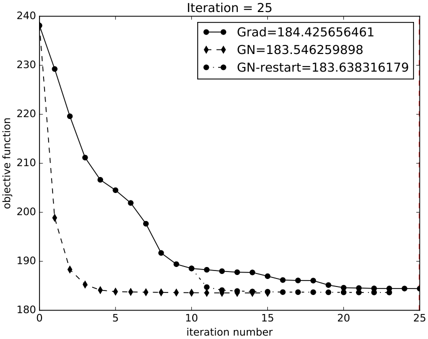



Iteration = 24
Grad=184.439098725
GN=183.546259898
GN-restart=183.638316179
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25








Iteration = 25
Grad=184.425656461
GN=183.546259898
GN-restart=183.638316179
objective function
iteration number
240
230
220
210
200
190
180
0
5
10
15
20
25