Constrained Optimization: Indirect Methods

AE 6310: Optimization for the Design of Engineered Systems

Spring 2017

Dr. Glenn Lightsey

Lecture Notes Developed By Dr. Brian German

Georgia Institute of Technology

Daniel Guggenheim School of Aerospace Engineering

Indirect Methods

Indirect methods for constrained optimization "modify" the problem to allow solution with unconstrained optimization techniques.

The influence of the constraints is captured by adding a *penalty function* to the objective function to create a *pseudo-objective function* that is then minimized*:*

$$\underbrace{\phi(\boldsymbol{x}, r_p)}_{\text{Pseudo-objective Function}} = \underbrace{f(\boldsymbol{x})}_{\text{Objective Function}} + \underbrace{r_p}_{\text{Scalar Multiplier}} \underbrace{p(\boldsymbol{x})}_{\text{Penalty Function}}$$

Indirect Methods

The penalty function increases the value of the pseudo-objective to represent the degree to which the constraints are violated.

The result is that infeasible designs have very large values of the pseudo-objective function.

The penalty function is necessarily “steep” in the region at which the constraints become active, i.e. the boundary between feasible and infeasible regions of the design space.

The steepness of the penalty function may cause numerical ill-conditioning when unconstrained algorithms are applied.

Indirect Methods

To deal with this issue, the scalar parameter r_p is successively changed during each iteration p.

A complete unconstrained optimization is conducted for each iteration p, with the next optimization using the optimum from the prior optimization as a starting point.

Because of this approach, indirect methods for constrained optimization are sometimes called *Sequential Unconstrained Minimization Techniques (SUMTs).*

Types of Indirect Methods

Indirect methods can be classified into three general types:

- Exterior penalty function methods
- Interior penalty function methods
- Augmented Lagrangian methods

We will examine each of these methods.

Exterior Penalty Function Method

In the exterior penalty function method, the penalty function $p(\boldsymbol{x})$ is formulated as,

$$p(\boldsymbol{x}) = \sum_{j=1}^{m} \max\left[0, g_j(\boldsymbol{x})\right]^2 + \sum_{k=1}^{l} [h_k(\boldsymbol{x})]^2$$

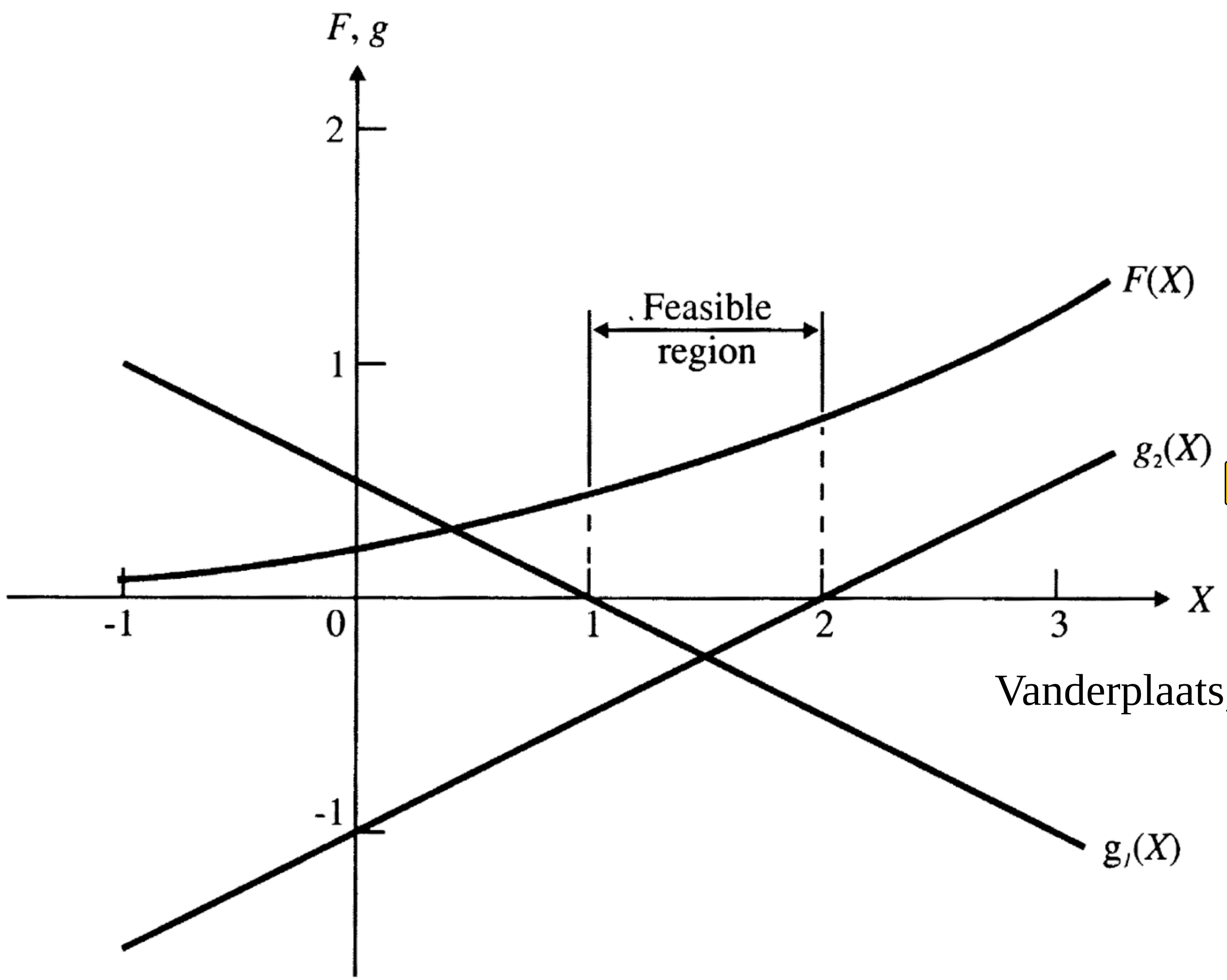
and the corresponding pseudo-objective is,

$$\phi(\boldsymbol{x}, r_p) = f(\boldsymbol{x}) + r_p \left\{ \sum_{j=1}^{m} \max\left[0, g_j(\boldsymbol{x})\right]^2 + \sum_{k=1}^{l} [h_k(\boldsymbol{x})]^2 \right\}$$

Exterior Penalty Function Method

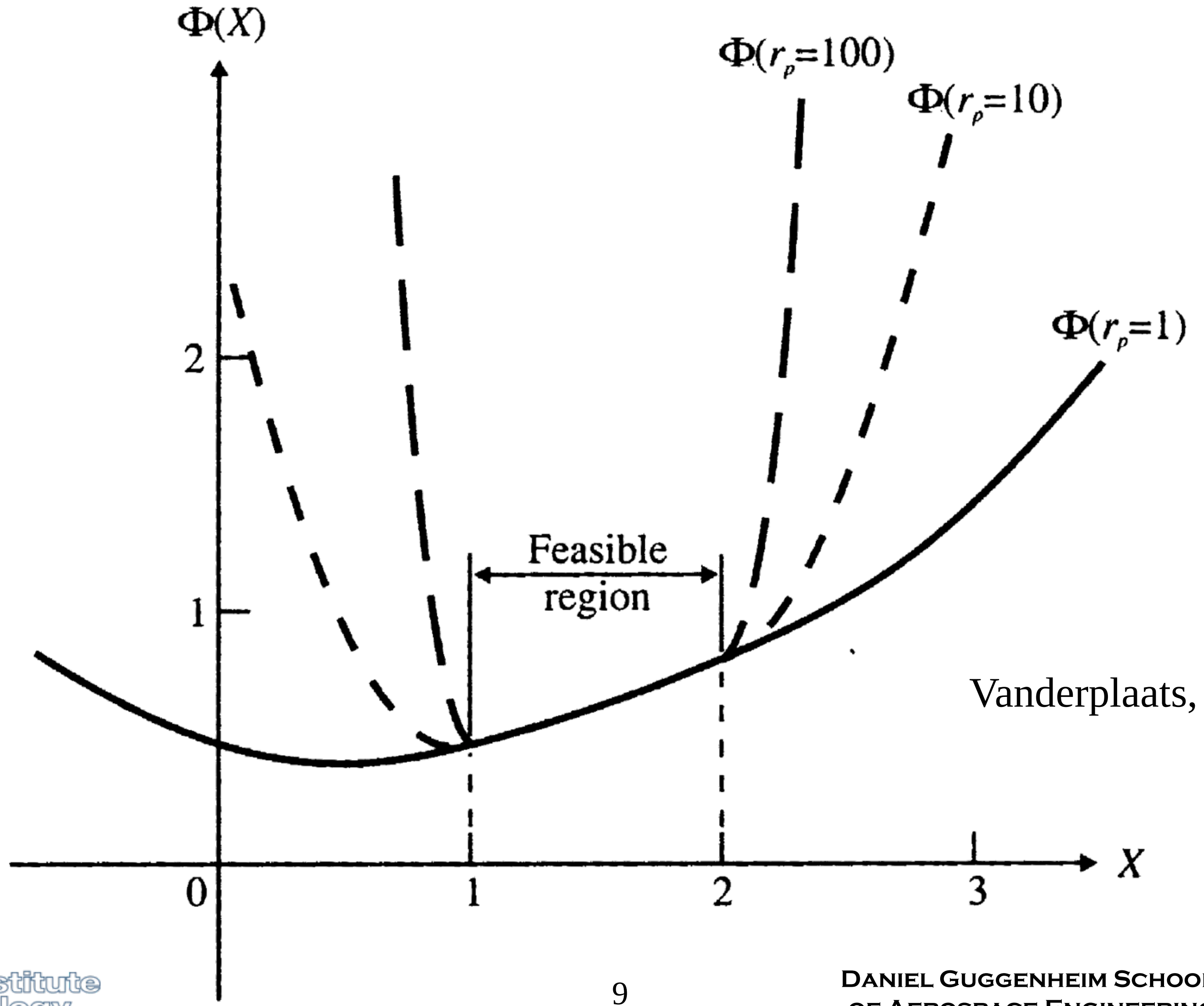
As its name implies, the exterior penalty function increases the objective only when the search proceeds in the region *exterior* to the feasible region, i.e. within the infeasible region.

Exterior Penalty Function Example



Vanderplaats, Fig. 5.2

Exterior Penalty Function Example



Vanderplaats, Fig. 5.3

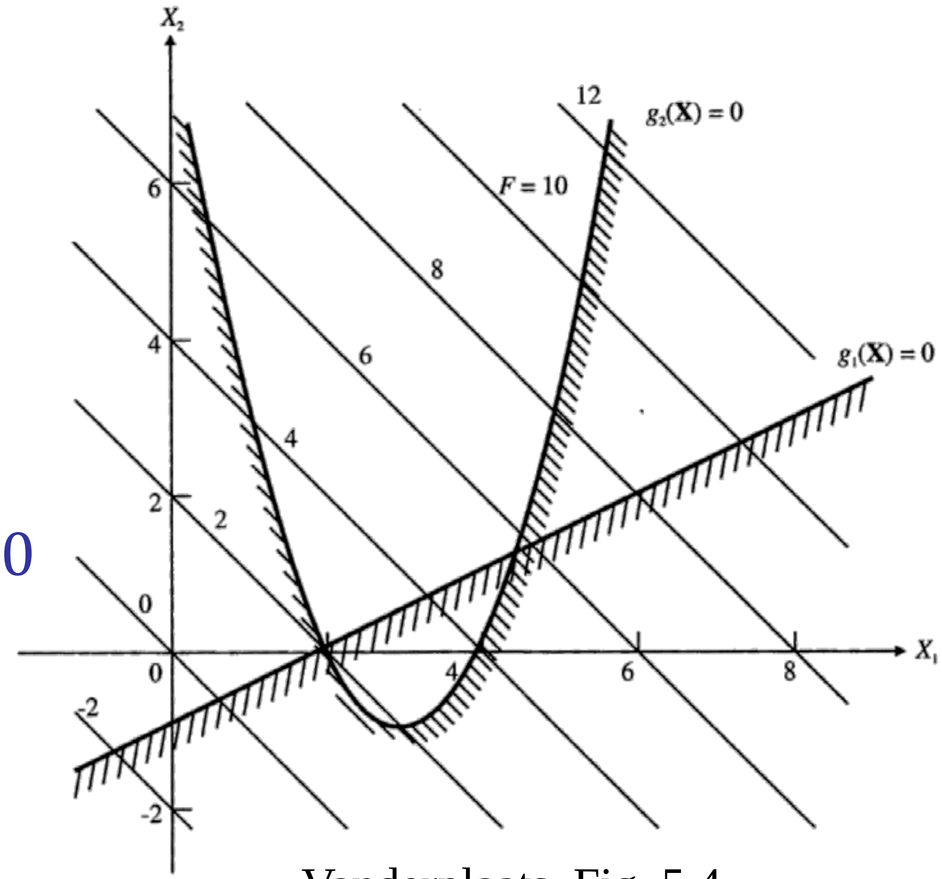
Exterior Penalty Function Example

Original problem:

$$\min f(x_1, x_2) = x_1 + x_2$$

subject to:

$$g_1(x_1, x_2) = -2 + x_1 - 2x_2 \le 0$$
$$g_2(x_1, x_2) = 8 - 6x_1 + x_1^2 + x_2 \le 0$$

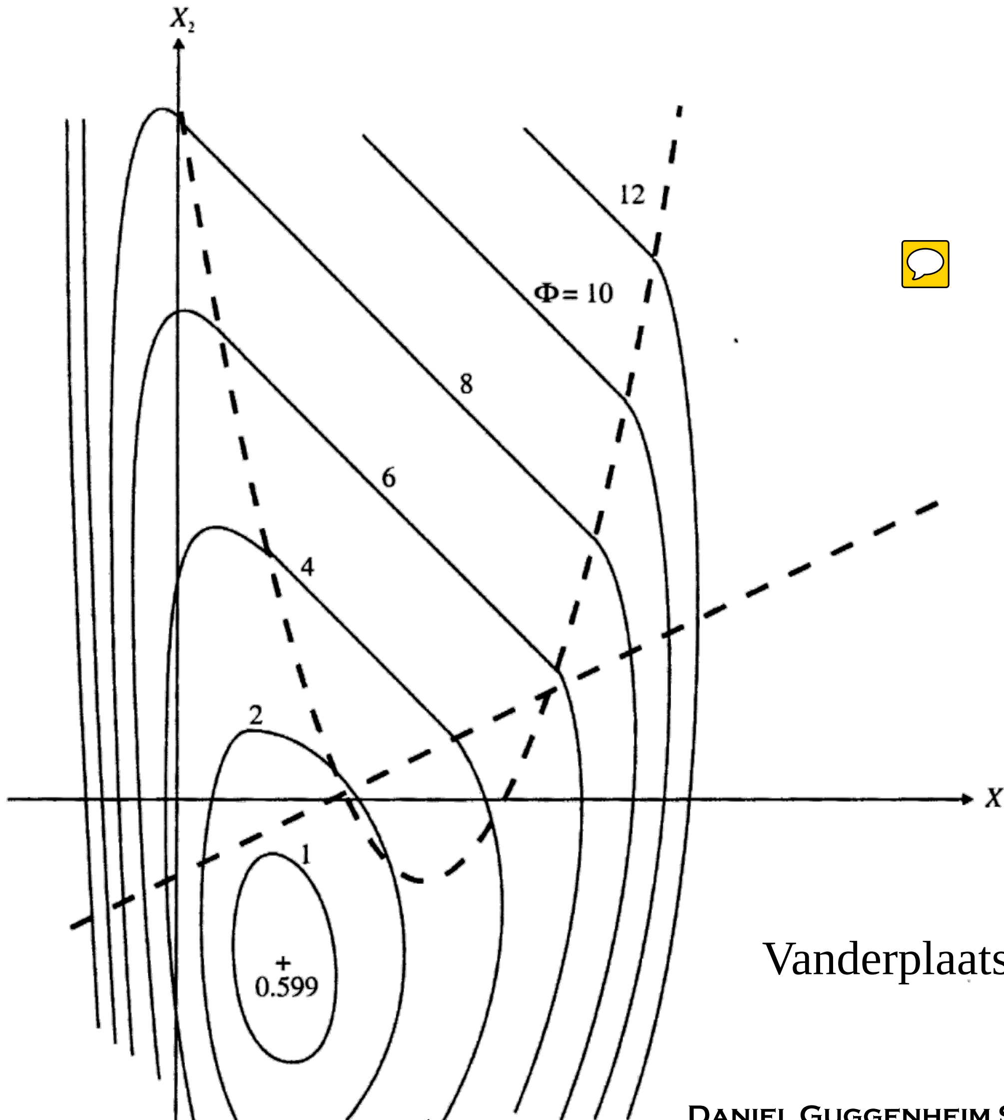


Vanderplaats, Fig. 5.4

Exterior Penalty Function Example

Contours of pseudo-obj.

$r_p = 0.05$

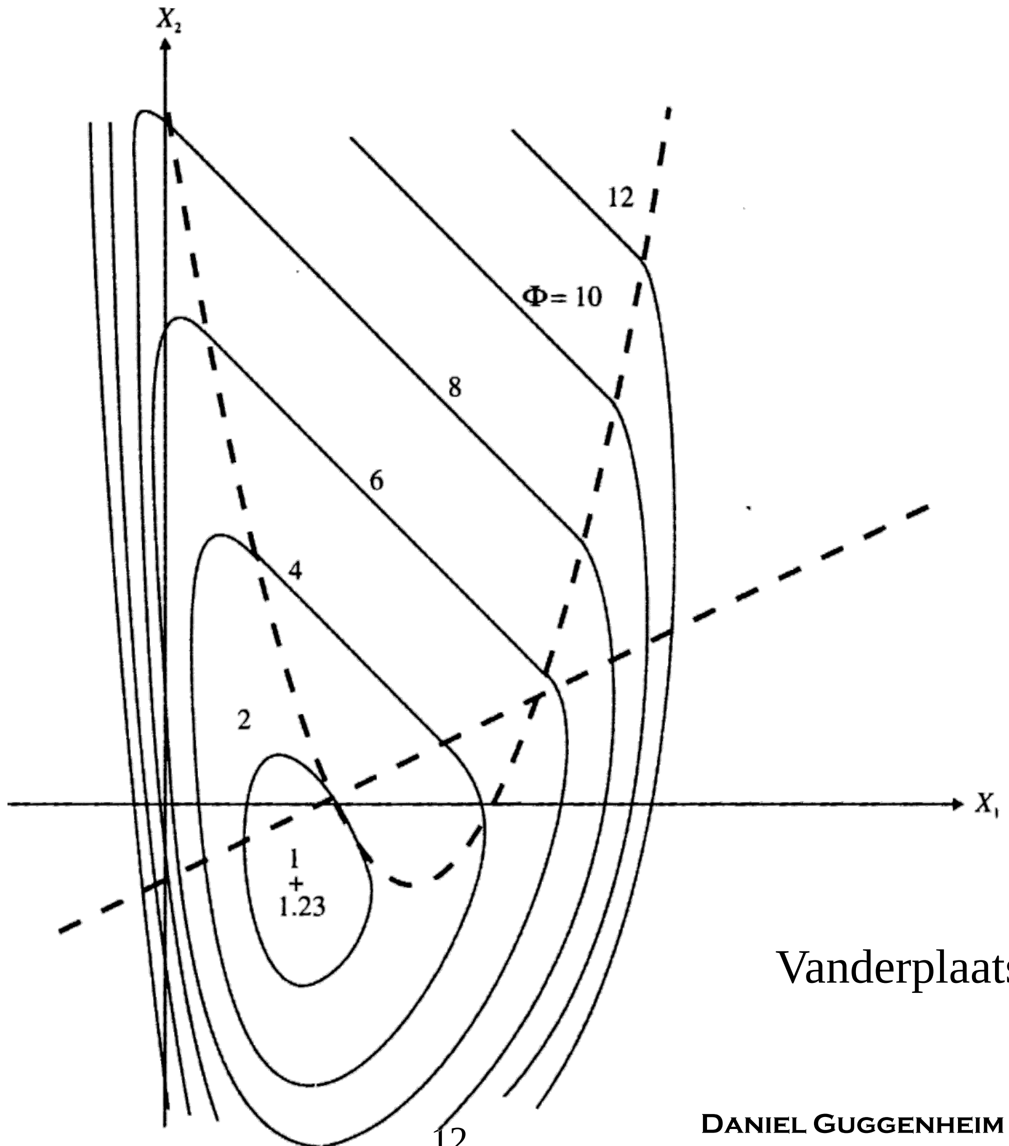


Vanderplaats, Fig. 5.4

Exterior Penalty Function Example

Contours of pseudo-obj.

$r_p = 0.1$

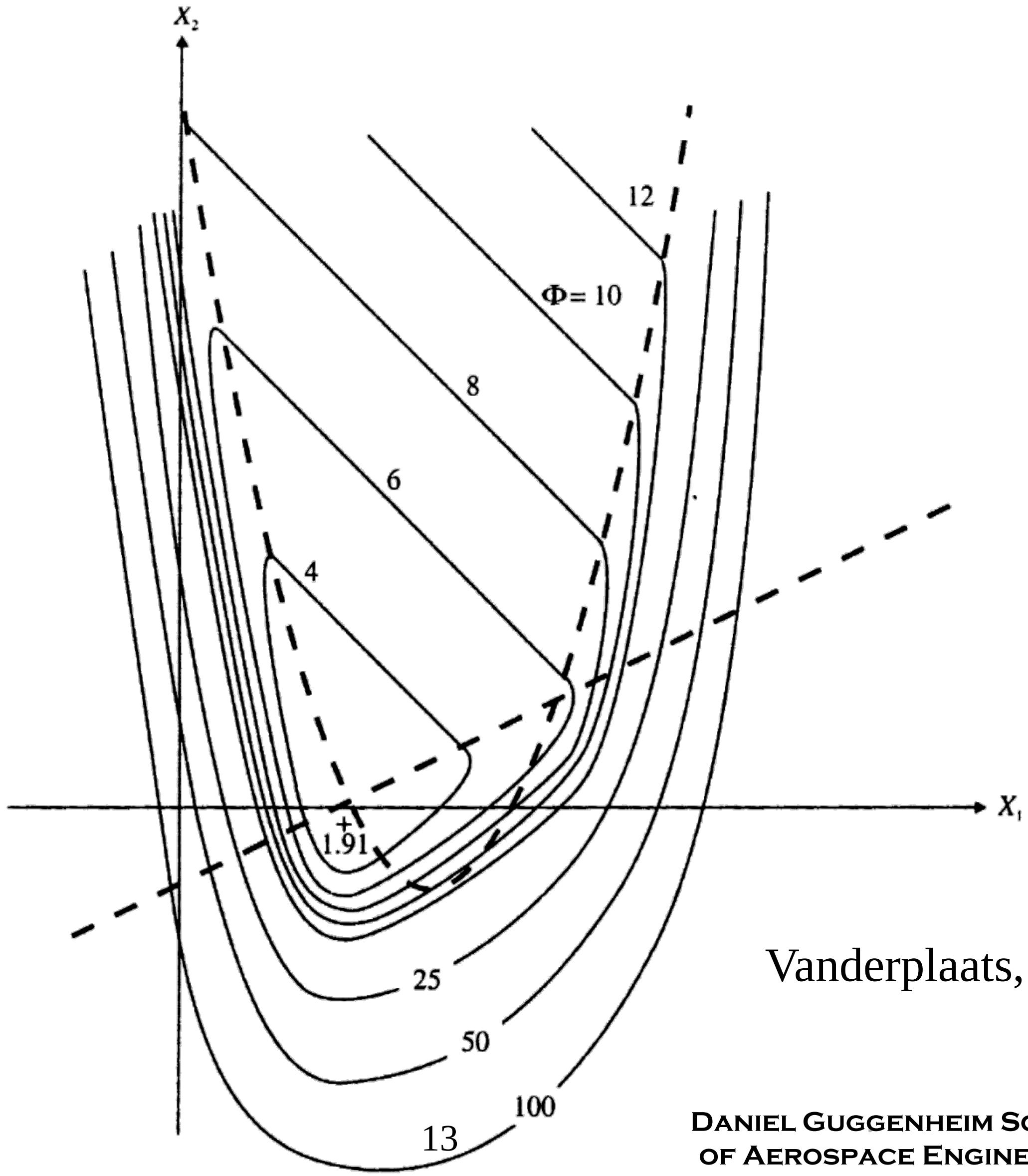


Vanderplaats, Fig. 5.4

Exterior Penalty Function Example

Contours of pseudo-obj.

$r_p = 1.0$



Vanderplaats, Fig. 5.4

Pros/Cons of Exterior Penalty Method

Advantages:

- Simple to implement
- Allows application of unconstrained algorithms

Disadvantages:

- As r_p is increased, the function becomes increasingly nonlinear and may cause difficulty with line searches
- The constrained optimum is approached from the infeasible region. If the iteration is stopped prematurely, the result will be an infeasible design.

Interior Penalty Function Method

As the name implies, the interior penalty function begins to increase the objective function inside the feasible region as the search nears the constraint boundary.

The intent is to ensure that, if iteration is stopped early, the resulting design will at least be feasible, unlike the case for exterior penalty methods.

Consequently, interior penalty functions are applied only for inequality constraints as there is no “interior” to the feasible region for equality constraints.

Interior Penalty Function Method

A common form of interior penalty function is,

$$p(\boldsymbol{x}) = \sum_{j=1}^{m} \frac{-1}{g_j(\boldsymbol{x})}$$

An interior penalty function for inequality constraints is then combined with an exterior penalty function for equality constraints to obtain the pseudo-objective as,

$$\phi(\boldsymbol{x}, r_p) = f(\boldsymbol{x}) + r_p' \sum_{j=1}^{m} \frac{-1}{g_j(\boldsymbol{x})} + r_p \sum_{\boldsymbol{k=1}}^{\boldsymbol{l}} [h_k(\boldsymbol{x})]^{\boldsymbol{2}}$$

Interior Penalty Function Method

A alternate form of interior penalty function is,

$$p(\boldsymbol{x}) = \sum_{j=1}^{m} -\log\left[-g_j(\boldsymbol{x})\right]$$

The pseudo-objective is then,

$$\phi\left(\boldsymbol{x}, r_p\right) = f(\boldsymbol{x}) + r_p' \sum_{j=1}^{m} -\log\left[-g_j(\boldsymbol{x})\right] + r_p \sum_{\boldsymbol{k=1}}^{\boldsymbol{l}} [h_k(\boldsymbol{x})]^{\boldsymbol{2}}$$

Interior Penalty Function Method

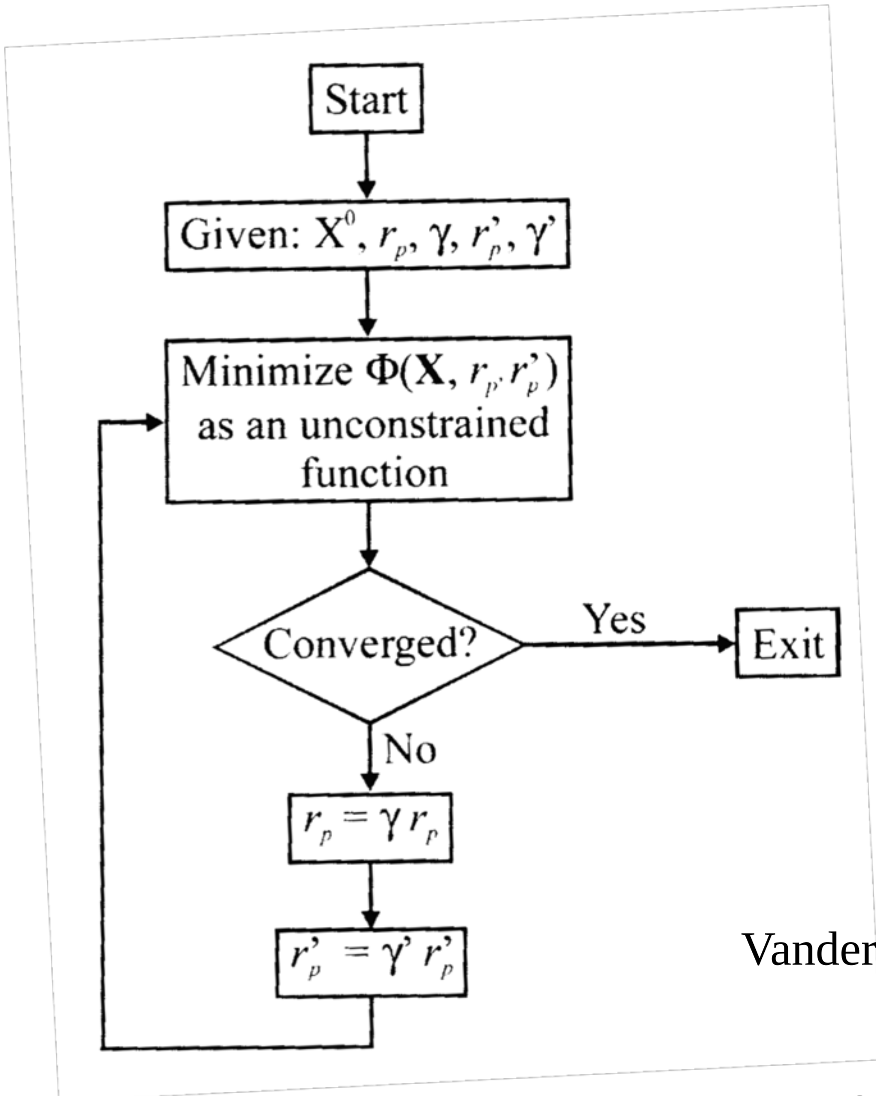
r_p' is initialized as a "large" (+) number and *decreased* toward zero.

r_p' is updated by the relation, $r_p' = \gamma' r_{p-1}'$, where $0 < \gamma' < 1$.

r_p is initialized as a "small" (+) number and *increased* as

$r_p = \gamma r_{p-1}$ where $\gamma > 1$

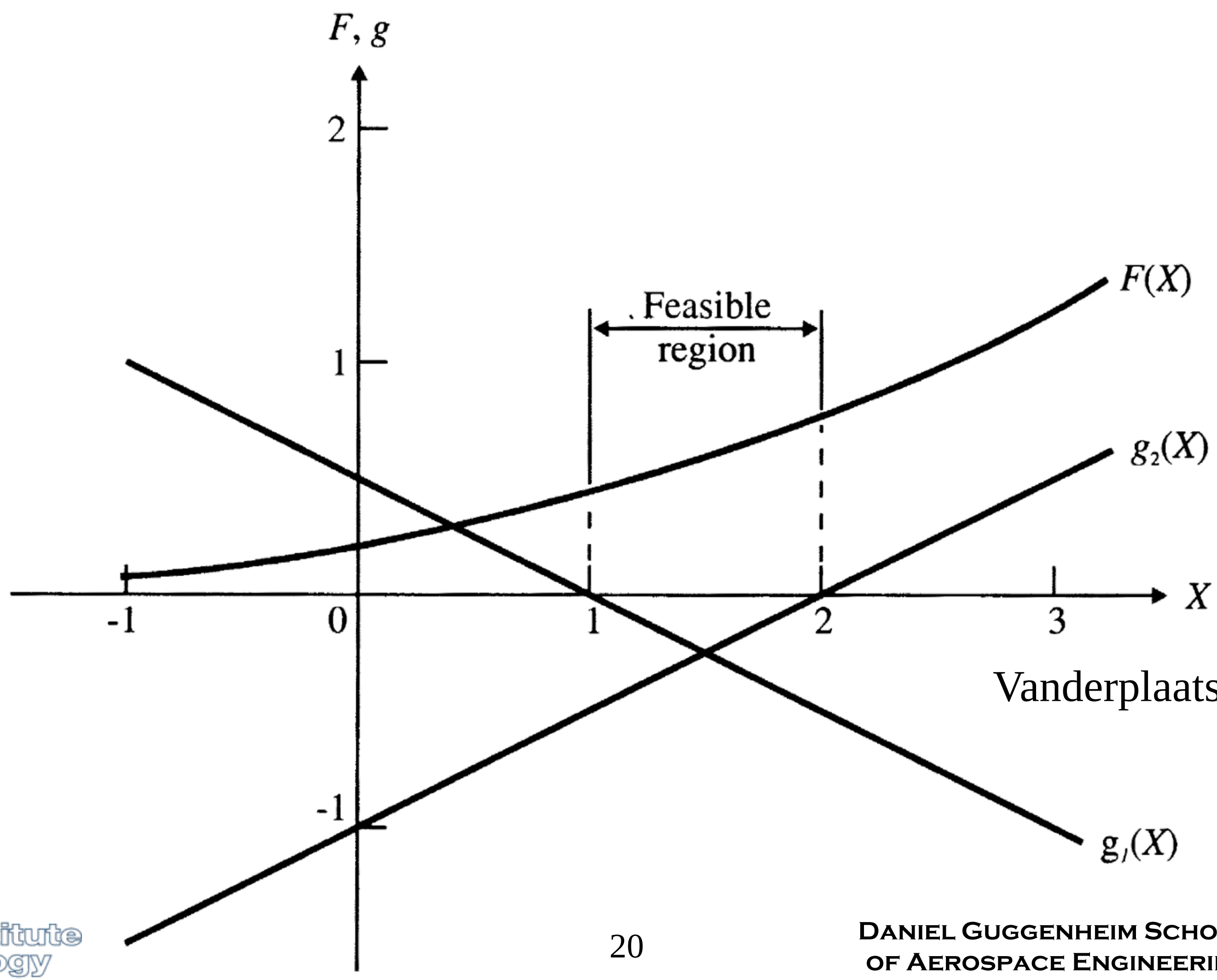
Interior Penalty Function Method



Vanderplaats, Fig. 5.5

Interior Penalty Function Example

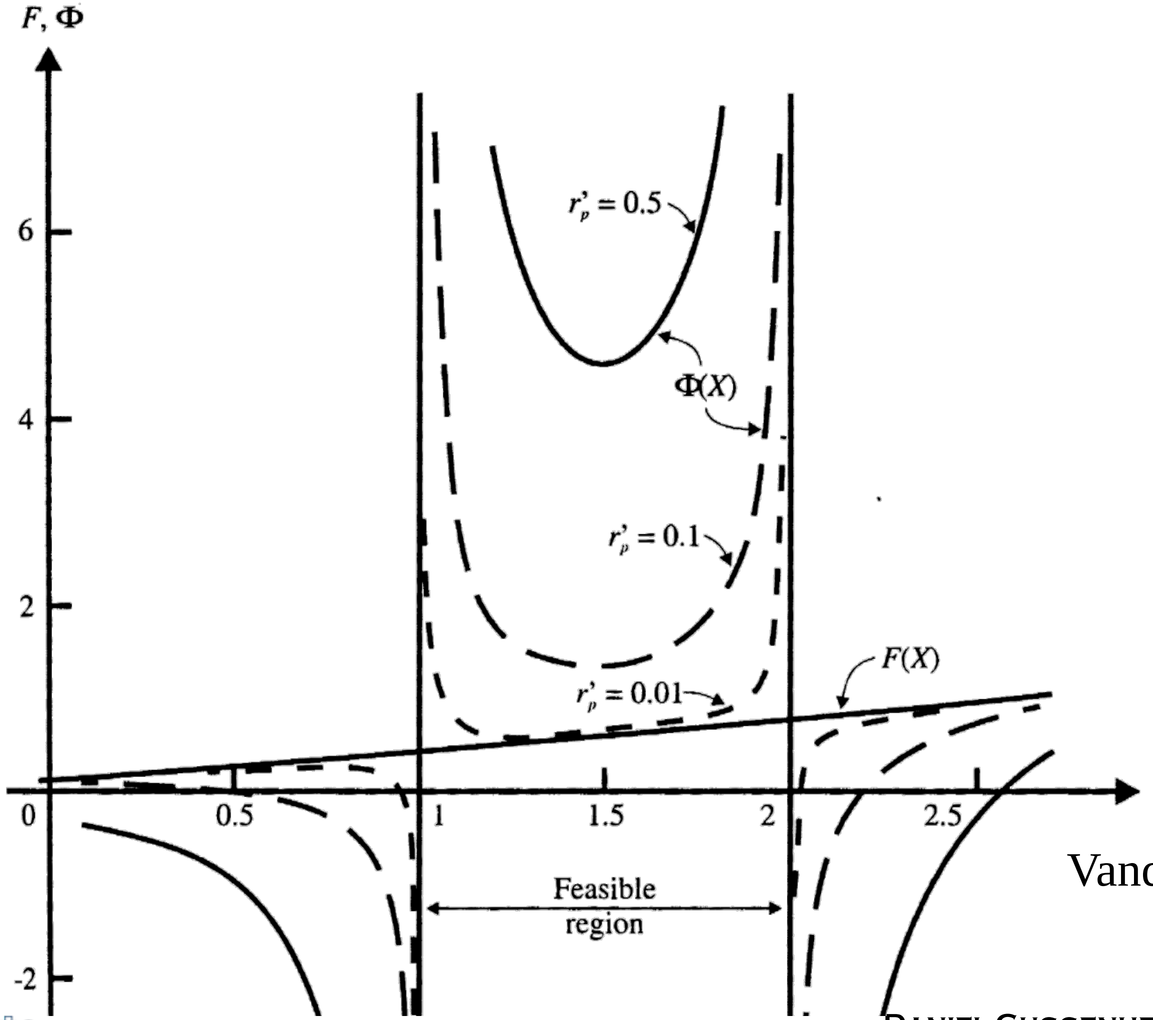
Recall the 1-D problem we examined for the exterior penalty method:



Vanderplaats, Fig. 5.5

Interior Penalty Function Example

Result of application of an interior penalty function:



Vanderplaats, Fig. 5.6

Pros/Cons of Interior Penalty Method

Advantages:

- A feasible design is obtained even if the algorithm stops early (presuming that the initial search was begun in the feasible region)

Disadvantages:

- Slightly more complicated than the exterior penalty method
- The pseudo-objective function is discontinuous at the constraint boundaries, which may cause considerable problems for line search methods

Extended Interior Penalty Function Method

The intent of the extended interior penalty function method is to create a pseudo-objective that is continuous everywhere, even at the constraint boundaries.

The penalty function is of the form,

$$p(\boldsymbol{x}) = \sum_{j=1}^{m} \widetilde{g_j}(\boldsymbol{x})$$

where $\widetilde{g_j}(\boldsymbol{x})$ is defined as…

Linear Extended Interior Penalty Function

For $g_j(\boldsymbol{x}) \leq \varepsilon$,

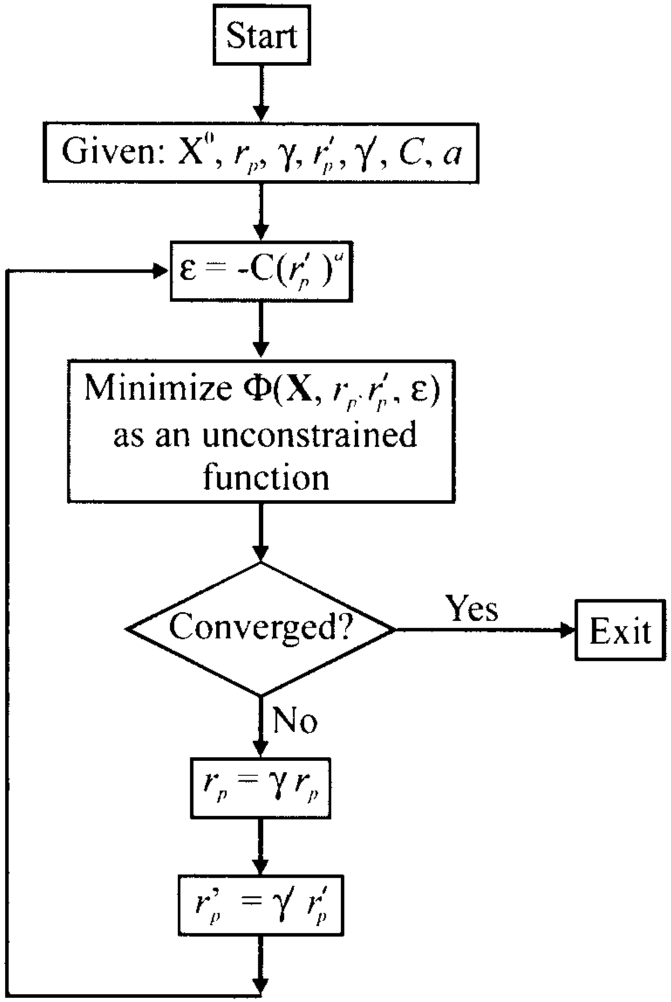
$$\widetilde{g_j}(\boldsymbol{x}) = \frac{-1}{g_j(\boldsymbol{x})}$$

For $g_j(\boldsymbol{x}) > \varepsilon$,

$$\widetilde{g_j}(\boldsymbol{x}) = -\frac{2\varepsilon - g_j(\boldsymbol{x})}{\varepsilon^2}$$

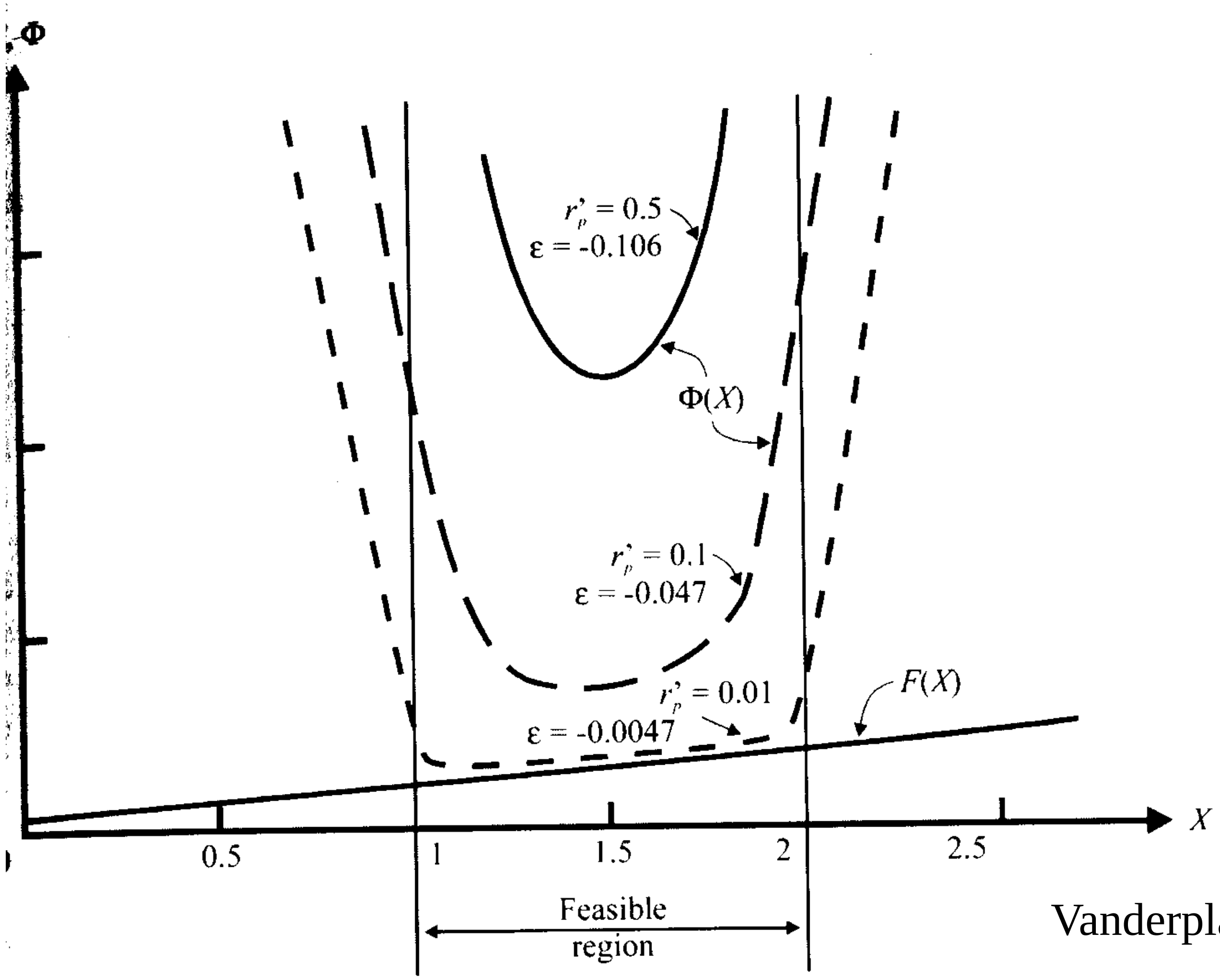
With $\varepsilon = -c\left(r_p'\right)^{\mathrm{a}}$ and $-\frac{1}{3} \leq a \leq \frac{1}{2}$. This choice results in a positive slope of the pseudo-objective at the constraint boundary. See Vanderplaats section 5-3.2 for details. c is a positive const.

Extended Interior Penalty Function Method



Vanderplaats, Fig. 5.7

Extended Interior Penalty Function Method



Vanderplaats, Fig. 5.8

Scaling of the Constraints

In penalty function methods, it is advantageous to scale the constraints such that the gradient of the constraints is of the same order of magnitude as the gradient of the objective function.

This has two advantages:

- Ensure that the curvature of the pseudo-objective is not dominated by a single constraint with much steeper gradients than the other constraints
- Make the methods less sensitive to initial choice of r_p and $r_p{}'$

Scaling of the Constraints

The recommended approach is to scale each constraint as follows. Let's say the un-scaled constraints are denoted as $\bar{g}_j$. Create the scaled constraints as,

$$g_j = c_j \bar{g}_j$$

where,

$$c_j = \frac{|\nabla f(\boldsymbol{x_0})|}{|\nabla \bar{g}_j(\boldsymbol{x_0})|}$$

and work with the scaled constraints in the formulating penalty functions.

Picking the Initial Penalty Parameters

With constraints scaled as described in the previous slide, good initial values of r_p and r_p' are simply:

$$r_0 = 1$$

$$r_0' = 1$$

Augmented Lagrangian Method

To understand the augmented Lagrangian method, first compare the Lagrangian to the pseudo-objective from the exterior penalty function.

Lagrangian:

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_{j=1}^{m} \lambda_j g_j(\boldsymbol{x}) + \sum_{k=1}^{l} \lambda_{m+k} h_k(\boldsymbol{x})$$

Pseudo-objective for exterior penalty function:

$$\phi(\boldsymbol{x}, r_p) = f(\boldsymbol{x}) + r_p \left\{ \sum_{j=1}^{m} \max\left[0, g_j(\boldsymbol{x})\right]^2 + \sum_{k=1}^{l} [h_k(\boldsymbol{x})]^2 \right\}$$

Augmented Lagrangian Method

Next, take the gradients of both and evaluate them at a potential optimal point $\boldsymbol{x}^*$:

$$\nabla\mathcal{L}(\boldsymbol{x}^*,\boldsymbol{\lambda}) = \nabla f^* + \sum_{j=1}^{m}\lambda_j\nabla g_j^* + \sum_{k=1}^{l}\lambda_{m+k}\nabla h_k^*$$

$$\nabla\phi\left(\boldsymbol{x}^*, r_p\right) = \nabla f^* + r_p\left\{\sum_{\substack{j=1,\\ j\ active}}^{m} 2g_j^*\nabla g_j^* + \sum_{k=1}^{\boldsymbol{l}} 2h_k^*\nabla h_k^*\right\}$$

Augmented Lagrangian Method

The KKT conditions require that at the optimum, $\nabla\mathcal{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}) = 0$. If we require that $\nabla\mathcal{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}) \to \mathbf{0}$ implies $\nabla\phi\left(\boldsymbol{x}^*, r_p\right) \to \mathbf{0}$ without placing restrictions on the constraint functions, then,

$$\lambda_j = 2r_p g_j^*, \quad j = 1, \ldots, m, \ j \text{ active}$$

and

$$\lambda_{m+k} = 2r_p h_k^*, \qquad k = 1, \ldots, l$$

Augmented Lagrangian Method

Rearranging gives,

$$g_j^* = \frac{\lambda_j}{2r_p}$$

and

$$h_k^* = \frac{\lambda_{m+k}}{2r_p}$$

Since the Lagrange multipliers are nonzero for the active constraints at $\boldsymbol{x}^*$, we must have $r_p \to \infty$ such that $g_j^* = 0$ and $h_k^* = 0$ to achieve feasibility.

Augmented Lagrangian Method

The augmented Lagrangian method attempts to achieve constrained optimal designs for finite values of r_p.

Let's first consider the method applied to an equality constrained problem with the Lagrangian,

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \sum_{k=1}^{l} \lambda_k h_k(\boldsymbol{x})$$

Augmented Lagrangian Method

The "augmented Lagrangian function" is defined by,

$$A(\boldsymbol{x}, \boldsymbol{\lambda_p}, r_p) = \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda_p}) + r_p \sum_{\boldsymbol{k=1}}^{\boldsymbol{l}} [h_k(\boldsymbol{x})]^2$$

or,

$$A(\boldsymbol{x}, \boldsymbol{\lambda_p}, r_p) = f(\boldsymbol{x}) + \sum_{k=1}^{l} \{\lambda_{p,k} h_k(\boldsymbol{x}) + r_p [h_k(\boldsymbol{x})]^2\}$$

Approximation to the Lagrange multiplier at iteration p

Augmented Lagrangian Method

Let's now take the gradient of the augmented Lagrangian:

$$\nabla A\left(\boldsymbol{x}, \boldsymbol{\lambda}_{\boldsymbol{p}}, r_p\right) = \nabla \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) + 2r_p \sum_{\boldsymbol{k}=\mathbf{1}}^{\boldsymbol{l}} h_k \nabla h_k$$

or,

$$\nabla A\left(\boldsymbol{x}, \boldsymbol{\lambda}_{\boldsymbol{p}}, r_p\right) = \nabla f + \sum_{k=1}^{l} \left\{\lambda_{p,k} \nabla h_k + 2r_p h_k \nabla h_k\right\}$$

Augmented Lagrangian Method

The gradient of both the true Lagrangian and the augmented Lagrangian should be equal to zero at the optimal point:

$$\nabla \mathcal{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}) = \nabla f^* + \sum_{k=1}^{l} \lambda_k \nabla h_k^* = \mathbf{0}$$

$$\nabla A\left(\boldsymbol{x}^*, \boldsymbol{\lambda}_p, r_p\right) = \nabla f^* + \sum_{k=1}^{l} \left\{\lambda_{p,k} \nabla h_k^* + 2 r_p h_k^* \nabla h_k^*\right\} = \mathbf{0}$$

Augmented Lagrangian Method

If we require that $\nabla\mathcal{L}(\boldsymbol{x}^*, \boldsymbol{\lambda}) \rightarrow \mathbf{0}$ implies $\nabla A\left(\boldsymbol{x}^*, \boldsymbol{\lambda_p}, r_p\right) \rightarrow \mathbf{0}$ without placing restrictions on the constraint functions, then,

$$\lambda_k = \lambda_{p,k} + 2r_p h_k^*$$

or,

$$h_k^* = \frac{\lambda_k - \lambda_{p,k}}{2r_p}$$

Augmented Lagrangian Method

If we compare the result from the augmented Lagrangian method,

$$h_k^* = \frac{\lambda_k - \lambda_{p,k}}{2r_p}$$

with our previous result from the exterior penalty function method,

$$h_k^* = \frac{\lambda_k}{2r_p}$$

we see that we no longer have to require $r_p \to \infty$ to satisfy the equality constraints ($h_k^* = 0$). Rather we can require that $\lambda_{p,k} \to \lambda_k$.

Augmented Lagrangian Method

This suggests that we can update our estimated value of the Lagrange multipliers, $\lambda_{p,k}$, during successive line searches as,

$$\lambda_{p+1,k} = \lambda_{p,k} + 2r_p h^*_{p,k}$$

Augmented Lagrangian Method

The augmented Lagrangian method for equality-constrained problems then proceeds as follows:

1. Set $p = 1$ and select initial values of r_p and all $\lambda_{p,k}$
2. Solve a complete unconstrained optimization to minimize the augmented Lagrangian with an algorithm of your choice.
3. Evaluate $h_{p,k}^*$ and set $p = p + 1$
4. Find $\lambda_{p+1,k}$ as $\lambda_{p+1,k} = \lambda_{p,k} + 2r_p h_{p,k}^*$
5. Increase r_p as $r_{p+1} = \gamma r_p$ with $\gamma > 1$
6. Repeat steps 2 through 4 until convergence criteria are met. Typically, $r_p < r_{p,\max}$ is set as an additional criterion.

Augmented Lagrangian Method

For a problem which also includes inequality constraints, the augmented Lagrangian generalizes as follows:

$$A(\boldsymbol{x}, \boldsymbol{\lambda_p}, r_p) = f(\boldsymbol{x}) + \sum_{j=1}^{m} \left[\lambda_j \psi_j + r_p \psi_j^2\right]$$

$$+ \sum_{k=1}^{l} \left\{\lambda_{p,k+m} h_k(\boldsymbol{x}) + r_p [h_k(\boldsymbol{x})]^2\right\}$$

where,

$$\psi_j = \max\left[g_j(\boldsymbol{x}), \frac{-\lambda_{p,j}}{2r_p}\right]$$

Augmented Lagrangian Method

The update equations for the Lagrange multipliers are then,

Equality constraints:

$$\lambda_{p+1,k+m} = \lambda_{p,k+m} + 2r_p h^*_{p,k}$$

Inequality constraints:

$$\lambda_{p+1,j} = \lambda_{p,j} + 2r_p \left\{ \max\left[g_j, \frac{-\lambda_{p,j}}{2r_p} \right] \right\}$$

The algorithm proceeds basically as previously described. See section 5-7.2 in Vanderplaats for details.

Augmented Lagrangian Method

The augmented Lagrangian method is considered to be a more "modern" form of penalty function method. Other methods, by comparison, are considered largely obsolete.

Advantages of the method (from Vanderplaats, Section 5-7.3):

- Relatively insensitive to the value of r_p. Not necessary for $r_p \to \infty$.
- Precise $g_j^* = 0$ (for active constraints) and $h_j^* = 0$ possible.
- Accelerated convergence is achieved compared to other methods by the process of updating the Lagrange multipliers.
- The starting point may be either feasible or infeasible.
- At the optimum, Lagrange multipliers satisfying $\lambda_j^* = 0$ will automatically identify the active constraint set.