CP: Solution algorithms

A. J. Conejo, R. Sioshansi, 2017
THE OHIO STATE UNIVERSITY

What

1. Penalty
2. Multipliers

Penalty

Penalty

We use a generic equality- and inequality-constrained nonlinear optimization problem of the form:

Penalty

$$
\begin{aligned}
\min_{x \in \mathbb{R}^n} \quad & f(x) \\
\text{s.t.} \quad & h_1(x) = 0 \\
& h_2(x) = 0 \\
& \vdots \\
& h_m(x) = 0 \\
& g_1(x) \leq 0 \\
& g_2(x) \leq 0 \\
& \vdots \\
& g_r(x) \leq 0
\end{aligned}
$$

Penalty

The first step to applying the penalty-based method is to convert the inequality constraints to equalities.

This is done by adding or subtracting non-negative slack or surplus variables from the left-hand side of the constraints.

If we introduce slack variables, $z_1, z_2, \ldots, z_r$ (i.e., one for each inequality constraint, our problem becomes:

Penalty

$$\begin{aligned} \min_{x\in\mathbb{R}^n, z\in\mathbb{R}^r, z\geq 0} \quad & f(x) \\ \text{s.t.} \quad & h_1(x) = 0 \\ & h_2(x) = 0 \\ & \vdots \\ & h_m(x) = 0 \\ & g_1(x) + z_1 = 0 \\ & g_2(x) + z_2 = 0 \\ & \vdots \\ & g_r(x) + z_r = 0 \end{aligned}$$

Penalty

We next add terms to the objective function to penalize constraint violations.

Because we have equal-to-zero constraints, we want to penalize having the left-hand sides of any of the constraints be negative or positive.

Thus, we add terms to the objective with the left-hand side of each constraint squared:

Penalty

$$f(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2 + \sum_{j=1}^{r} \phi_j \cdot (g_j(x) + z_j)^2,$$

where $\rho_1, \rho_2, \ldots, \rho_m, \phi_1, \phi_2, \ldots, \phi_r > 0$ are fixed coefficients that determine how much weight is placed on violating a particular constraint.

Penalty

Instead of solving the original constrained problem, we solve the problem:

$$\min_{x \in \mathbb{R}^n, z \in \mathbb{R}^r, z \geq 0} f(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2 + \sum_{j=1}^{r} \phi_j \cdot (g_j(x) + z_j)^2,$$

where we drop the original equality and inequality constraints.

Penalty

We solve this problem in two steps:

Penalty

$$\min_{x\in\mathbb{R}^n, z\in\mathbb{R}^r, z\geq 0} \; f(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2 + \sum_{j=1}^{r} \phi_j \cdot (g_j(x) + z_j)^2$$

$$= \min_{x\in\mathbb{R}^n} \left\{ \min_{z\in\mathbb{R}^r, z\geq 0} \left\{ f(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2 + \sum_{j=1}^{r} \phi_j \cdot (g_j(x) + z_j)^2 \right\} \right\}$$

$$= \min_{x\in\mathbb{R}^n} \left\{ f(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2 + \min_{z\in\mathbb{R}^r, z\geq 0} \left\{ \sum_{j=1}^{r} \phi_j \cdot (g_j(x) + z_j)^2 \right\} \right\}$$

Penalty

We solve this!

$$\min_{x \in \mathbb{R}^n} \left\{ f(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2 + \min_{z \in \mathbb{R}^r, z \geq 0} \left\{ \sum_{j=1}^{r} \phi_j \cdot (g_j(x) + z_j)^2 \right\} \right\}$$

Penalty

If we examine the third term:

$$\min_{z \in \mathbb{R}^r, z \geq 0} \sum_{j=1}^{r} \phi_j \cdot (g_j(x) + z_j)^2,$$

we note that this is a quadratic function of z and can minimize it with respect to z using the KKT condition, which is:

Penalty

$$2\phi_1 \cdot (g_1(x) + z_1) - \mu_1 = 0$$

$$\vdots$$

$$2\phi_r \cdot (g_r(x) + z_r) - \mu_r = 0$$

$$0 \leq z_1 \perp \mu_1 \geq 0$$

$$\vdots$$

$$0 \leq z_r \perp \mu_r \geq 0,$$

where μ_j is the Lagrange multiplier on the jth non-negativity constraint ($z_j \geq 0$).

Penalty

Solving the KKT conditions:

1. $\mu_i = 0 \Rightarrow z_i > 0 \quad \& \quad g_i(x) < 0$

   $\mu_i = 0 \Rightarrow g_i(x) + z_i = 0$, thus $z_i = -g_i(x)$

   thus

   $z_i = -g_i(x)$ if $g_i(x) < 0$

2. $\mu_i > 0 \Rightarrow z_i = 0$

   $\mu_i > 0 \Rightarrow 2\phi_i g_i(x) - \mu_i = 0$, thus, $\mu_i = 2\phi_i g_i(x)$

   thus

   $z_i = 0$, otherwise

Penalty

Thus

$$z_j^* = \begin{cases} -g_j(x), & \text{if } g_j(x) < 0; \\ 0, & \text{otherwise.} \end{cases}$$

We can write this more compactly as:

$$\boxed{z_j^* = \max\{0, -g_j(x)\}.}$$

Penalty

We further know that $z \geq 0$ defines a convex feasible region for the problem,

and its objective function has a Hessian that is positive definite:

$$\begin{bmatrix} 2\phi_1 & 0 & \cdots & 0 \\ 0 & 2\phi_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 2\phi_r \end{bmatrix}$$

Penalty

Thus, this third-term problem is convex and

$$z_j^* = \max\{0, -g_j(x)\}$$

is its global minimum.

The objective function:

Penalty

$$\left.\sum_{j=1}^{r} \phi_j \cdot (g_j(x) + z_j)^2 \right]_{z_j = z_j^* = \max\{0, -g_j(x)\}, \forall j} =$$

$$\sum_{j=1}^{r} \phi_j \cdot (g_j(x) + \max\{0, -g_j(x)\})^2 =$$

$$\boxed{\sum_{j=1}^{r} \phi_j \cdot (\max\{g_j(x), 0\})^2}$$

Penalty

Because we have an explicit closed-form value for z, we can substitute this into the original problem and instead solve the following unconstrained minimization problem:

$$\min_{x \in \mathbb{R}^n} \mathcal{F}_{\rho,\phi}(x) = f(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2 + \sum_{j=1}^{r} \phi_j \cdot (\max\{0, g_j(x)\})^2,$$

using any of the methods for UPs.

Penalty

We solve this!

$$\min_{x \in \mathbb{R}^n} \mathcal{F}_{\rho,\phi}(x) = f(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2 + \sum_{j=1}^{r} \phi_j \cdot (\max\{0, g_j(x)\})^2$$

Penalty

The penalized objective function, including the max term in it, is continuously differentiable. However, the Hessian is discontinuous at any x where $g_j(x) = 0$.

Thus, the steepest descent direction can be used to solve the penalized objective function without any problem.

Moreover, Newton's method can be used, except at these points of discontinuity.

Penalty

A question at this point is what values to set the ρ's and ϕ's equal to. A naïve answer is to set them to arbitrarily large values. By doing so, we ensure that we have enough weight on constraint violations to find a feasible solution.

The problem with setting the ρ's and ϕ's to very high values is that doing so can introduce scaling problems. Poorly scaled problems can be difficult to solve using our iterative algorithms.

Penalty

Thus, in practice, we begin with small values for the ρ's and ϕ's (setting them equal to one is a reasonable starting point) and begin using an iterative algorithm for UPs. As we conduct iterations, we test to see if the constraint violations are getting smaller or not.

Penalty

The ρ's and ϕ's associated with constraints that are having their violations reduced are kept the same. The ρ's and ϕ's associated with constraints that are not seeing an improvement in their violations are increased.

This is continued iteratively until all of the constraints are satisfied and the algorithm finds a stationary point of $\mathcal{F}(x)$, or another termination criteria (such as a limit on the number of iterations) is met.

Penalty: Solve: $\min_{x \in \mathbb{R}^n} \mathcal{F}_{\rho^k, \phi^k}(x)$, and get x^{k+1}.

1. Set $k = 0$ and initialize x^k, ρ^k and ϕ^k.
2. Solve: $\min_{x \in \mathbb{R}^n} \mathcal{F}_{\rho^k, \phi^k}(x)$, and get x^{k+1}.
3. If

   $|x^{k+1} - x^k| \leq \epsilon_1$ and

   $|h(x^{k+1})| \leq \epsilon_2$ and

   $g(x^{k+1}) \leq \epsilon_3$, stop.
4. Otherwise update ρ and ϕ as appropriate and continue in 2.

We don't need to solve it to optimality. One iteration is fine.

procedure PENALTY-BASED ALGORITHM
$k \leftarrow 0$ ▷ Set iteration counter to 0
Fix $\rho_1, \rho_2, \ldots, \rho_m, \phi_1, \phi_2, \ldots, \phi_r$ to positive values ▷ Initialize penalty weights
Fix x^0 ▷ Fix a starting point
while Termination criteria are not met **do**
Find direction, d^k, to move in
Determine step size, α^k
$x^{k+1} \leftarrow x^k + \alpha^k d^k$ ▷ Update point
for $i \leftarrow 1, \ldots, m$ **do**
if $|h_i(x^{k+1})| \geq |h_i(x^k)|$ and $h_i(x^{k+1}) \neq 0$ **then**
Increase ρ_i ▷ Increase penalty weight if constraint does not improve
end if
end for
for $j \leftarrow 1, \ldots, r$ **do**
if $\max\{0, g_j(x^{k+1})\} \geq \max\{0, g_j(x^k)\}$ and $\max\{0, g_j(x^{k+1})\} \neq 0$ **then**
Increase ϕ_j ▷ Increase penalty weight if constraint does not improve
end if
end for
$k \leftarrow k + 1$ ▷ Update iteration counter
end while
end procedure

Penalty: Example

Penalty: Example

Consider the problem:

$$\min_{x} \; f(x) = (x_1 + 2)^2 + (x_2 - 3)^2$$

$$\text{s.t. } h_1(x) = x_1 + 2x_2 - 4 = 0$$

$$g_1(x) = -x_1 \leq 0.$$

Penalty: Example

We derive the penalized objective function:

$$\mathcal{F}_{1,1}(x) = (x_1+2)^2+(x_2-3)^2+(x_1+2x_2-4)^2+(\max\{0, -x_1\})^2,$$

with weights of one on both constraints.

Penalty: Example

Starting from the point $x^0 = (-1, -1)^\top$, we conduct one iteration of steepest descent using an approximate line search.

Penalty: Example

To conduct the iteration, we must compute the gradient of the objective function. To do so, we note that at $x^0 = (-1, -1)^\top$ the objective is equal to:

$$\begin{aligned}\mathcal{F}_{1,1}(x) &= (x_1 + 2)^2 + (x_2 - 3)^2 + (x_1 + 2x_2 - 4)^2 + (\max\{0, -x_1\})^2 \\ &= (x_1 + 2)^2 + (x_2 - 3)^2 + (x_1 + 2x_2 - 4)^2 + (-x_1)^2.\end{aligned}$$

Penalty: Example

This step of determining whether:

$$\max\{0, -x_1\},$$

is equal to 0 or $-x_1$ is vitally important, as the gradient that we compute depends on what we substitute for this term.

Penalty: Example

The gradient is equal to:

$$\nabla \mathcal{F}_{1,1}(x) = \begin{pmatrix} 2(x_1 + 2) + 2(x_1 + 2x_2 - 4) + 2x_1 \\ 2(x_2 - 3) + 4(x_1 + 2x_2 - 4) \end{pmatrix},$$

and the search direction is:

$$d^0 = -\nabla \mathcal{F}_{1,1}(x^0) = \begin{pmatrix} 14 \\ 36 \end{pmatrix}.$$

Penalty: Example

Approximate line search

If we set $\alpha^0 = 1/10$, our new point is $x^1 = (0.4, 2.6)^\top$.

Penalty: Example

We already know that the value of the penalized objective function improves from this one iteration. We further have that:

$$f(x^0) = 17,$$

and:

$$f(x^1) = 5.92,$$

meaning that the objective of the original problem improved in this one iteration.

Penalty: Example

We further have that:

$$h_1(x^0) = -7,$$

and:

$$h_1(x^1) = 1.6,$$

and:

$$g_1(x^0) = 1,$$

and:

$$g_1(x^1) = -0.4.$$

Penalty: Example

This means that we have reduced the extent to which the two constraints are violated in this one iteration.

The $h_1(\cdot)$ constraint is still violated, but by less than before. The $g_1(\cdot)$ constraint is satisfied by the new point that we have found.

Penalty: Example

We keep going...

Multipliers

Multipliers

Just equalities for simplicity

Extension do inequalities is straightforward

Multipliers

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \ & f(x) && (1) \\ \text{s.t.} \ & h_1(x) = 0 \\ & h_2(x) = 0 \\ & \vdots \\ & h_m(x) = 0 \end{aligned}$$

Multipliers

$$\min_{x \in \mathbb{R}^n} \; f(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2 \qquad (2)$$

Multipliers

$$\min_{x \in \mathbb{R}^n} f(x) + \sum_{i=1}^{m} \lambda_i^* \cdot h_i(x) \tag{3}$$

$$\begin{aligned} \text{s.t.} \quad & h_1(x) = 0 \\ & h_2(x) = 0 \\ & \vdots \\ & h_m(x) = 0 \end{aligned}$$

Multipliers

$$\min_{x \in \mathbb{R}^n} \; f(x) + \sum_{i=1}^{m} \lambda_i^* \cdot h_i(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2 \qquad (4)$$

Multipliers

(1), (2), (3) and (4) are equivalent!

Multipliers

$$\min_{x \in \mathbb{R}^n} \; f(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2$$

2

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & h_1(x) = 0 \\ & h_2(x) = 0 \\ & \vdots \\ & h_m(x) = 0 \end{aligned}$$

1

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & f(x) + \sum_{i=1}^{m} \lambda_i^* \cdot h_i(x) \\ \text{s.t.} \quad & h_1(x) = 0 \\ & h_2(x) = 0 \\ & \vdots \\ & h_m(x) = 0 \end{aligned}$$

3

$$\min_{x \in \mathbb{R}^n} \; f(x) + \sum_{i=1}^{m} \lambda_i^* \cdot h_i(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2$$

4

Multipliers

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & f(x) \\ \text{s.t.} \quad & h_1(x) = 0 \\ & h_2(x) = 0 \\ & \vdots \\ & h_m(x) = 0 \end{aligned}$$

KKTs of (1) are:

$$\nabla f(x) + \sum_{i=1}^{m} \lambda_i^* \cdot \nabla h_i(x) = 0 \qquad (5)$$

Multipliers

$$\min_{x \in \mathbb{R}^n} \ f(x) + \sum_{i=1}^{m} \lambda_i^* \cdot h_i(x)$$
$$\text{s.t.} \quad h_1(x) = 0$$
$$h_2(x) = 0$$
$$\vdots$$
$$h_m(x) = 0$$

Considering approximate multiplier values $\lambda_1^k, \cdots, \lambda_m^k$, problem (3) becomes:

$$\min_{x \in \mathbb{R}^n} \ f(x) + \sum_{i=1}^{m} \lambda_i^k \cdot h_i(x) \tag{6}$$
$$\text{s.t.} \quad h_1(x) = 0$$
$$h_2(x) = 0$$
$$\vdots$$
$$h_m(x) = 0$$

Multipliers

$$\min_{x \in \mathbb{R}^n} \; f(x) + \sum_{i=1}^{m} \lambda_i^k \cdot h_i(x)$$
$$\text{s.t.} \quad h_1(x) = 0$$
$$h_2(x) = 0$$
$$\vdots$$
$$h_m(x) = 0$$

KKTs of (6) are:

$$\nabla f(x) + \sum_{i=1}^{m} \lambda_i^k \cdot \nabla h_i(x) + \sum_{i=1}^{m} \hat{\lambda}_i \cdot \nabla h_i(x) = 0 \quad (7)$$

Multipliers

Considering (5) and (7):

$$\nabla f(x) + \sum_{i=1}^{m} \lambda_i^* \cdot \nabla h_i(x) = 0$$

$$\nabla f(x) + \sum_{i=1}^{m} \lambda_i^k \cdot \nabla h_i(x) + \sum_{i=1}^{m} \hat{\lambda}_i \cdot \nabla h_i(x) = 0$$

Multipliers

We conclude:

$$\lambda_i^* = \lambda_i^k + \hat{\lambda}_i, \quad i = 1, \ldots, m$$

or

$$\hat{\lambda}_i = \lambda_i^* - \lambda_i^k, \quad i = 1, \ldots, m$$

$$\min_{x \in \mathbb{R}^n} \; f(x) + \sum_{i=1}^{m} \lambda_i^* \cdot h_i(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2$$

Multipliers

Considering approximate multiplier values $\lambda_1^k, \cdots, \lambda_m^k$, problem (4) becomes:

$$\min_{x \in \mathbb{R}^n} \; f(x) + \sum_{i=1}^{m} \lambda_i^k \cdot h_i(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2 \qquad (8)$$

$$\min_{x \in \mathbb{R}^n} \; f(x) + \sum_{i=1}^{m} \lambda_i^k \cdot h_i(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2$$

Multipliers

KKTs of (8) are:

$$\nabla f(x) + \sum_{i=1}^{m} \lambda_i^k \cdot \nabla h_i(x) + \sum_{i=1}^{m} 2\rho_i h_i(x) \cdot \nabla h_i(x) = 0 \tag{9}$$

Multipliers

Considering (7) and (9):

$$\nabla f(x) + \sum_{i=1}^{m} \lambda_i^k \cdot \nabla h_i(x) + \sum_{i=1}^{m} \hat{\lambda}_i \cdot \nabla h_i(x) = 0$$

$$\nabla f(x) + \sum_{i=1}^{m} \lambda_i^k \cdot \nabla h_i(x) + \sum_{i=1}^{m} 2\rho_i h_i(x) \cdot \nabla h_i(x) = 0$$

Multipliers

We conclude:

$$\hat{\lambda}_i = 2\rho_i h_i(x) \quad i = 1, \ldots, m$$

or

$$\boxed{\hat{\lambda}_i = 2\rho_i h_i(x^k) \quad i = 1, \ldots, m}$$

Multipliers

Considering:

$$\hat{\lambda}_i = \lambda_i^* - \lambda_i^k, \quad i = 1, \ldots, m$$

$$\hat{\lambda}_i = 2\rho_i \cdot h_i(x^k) \quad i = 1, \ldots, m$$

Multipliers

We conclude:

$$\lambda_i^* - \lambda_i^k = 2\rho_i \cdot h_i(x^k) \quad i = 1, \ldots, m$$

and:

$$\boxed{\lambda_i^{k+1} \leftarrow \lambda_i^k + 2\rho_i \cdot h_i(x^k) \quad i = 1, \ldots, m}$$

Multipliers

$$\lambda_i^{k+1} \leftarrow \lambda_i^k + 2\rho_i \cdot h_i(x^k) \quad i = 1, \ldots, m$$

Multipliers

1. Set $k = 0$, choose $\rho_1, \cdots, \rho_m$ large enough, and initialize λ^k.

2. Solve:

$$\min_{x \in \mathbb{R}^n} \; f(x) + \sum_{i=1}^{m} \lambda_i^k \cdot h_i(x) + \sum_{i=1}^{m} \rho_i \cdot (h_i(x))^2$$

   and get x^k.

3. Update λ: $\lambda_i^{k+1} \leftarrow \lambda_i^k + 2\rho_i \cdot h_i(x^k) \quad i = 1, \dots, m$

4. Repeat 1-3 until $|\lambda_i^{k+1} - \lambda_i^k| \leq \epsilon \quad i = 1, \dots, m$

We don't need to solve it to optimality. One iteration is fine.

Multipliers

1. No need to solve the problem in step 2 to optimality.
2. Penalty terms can be dynamically updated.

See algorithm below:

procedure MULTIPLIER-BASED ALGORITHM

$k \leftarrow 0$ ▷ Set iteration counter to 0

Fix $\rho_1, \rho_2, \ldots, \rho_m$ to positive values ▷ Initialize penalty weights

Fix λ^0 ▷ Initialize Lagrange multipliers

Fix x^0 ▷ Fix a starting point

while Termination criteria are not met **do**

Find direction, d^k, to move in

Determine step size, α^k

$x^{k+1} \leftarrow x^k + \alpha^k d^k$ ▷ Update point

for $i \leftarrow 1, \ldots, m$ **do**

$\lambda_i^{k+1} \leftarrow \lambda_i^k + 2\rho_i h_i(x^{k+1})$ ▷ Update multiplier

if $|h_i(x^{k+1})| \geq |h_i(x^k)|$ and $h_i(x^{k+1}) \neq 0$ **then**

Increase ρ_i ▷ Increase penalty weight if constraint does not improve

end if

end for

$k \leftarrow k + 1$ ▷ Update iteration counter

end while

end procedure

Example

Example

Consider the problem:

$$\min_{x} \; f(x) = (x_1 + 2)^2 + (x_2 - 3)^2$$

$$\text{s.t. } h_1(x) = x_1 + 2x_2 - 4 = 0$$

$$g_1(x) = -x_1 \leq 0,$$

Example

To solve this problem using the multiplier method, we first convert it to the equality-constrained problem:

$$\min_{x} \; f(x) = (x_1 + 2)^2 + (x_2 - 3)^2$$
$$\text{s.t. } h_1(x) = x_1 + 2x_2 - 4 = 0$$
$$h_2(x) = -x_1 + x_3^2 = 0.$$

Example

We next derive the augmented Lagrangian function:

$$\mathcal{A}_{\lambda,\rho}(x) = (x_1 + 2)^2 + (x_2 - 3)^2 + \lambda_1 \cdot (x_1 + 2x_2 - 4)$$
$$+ \lambda_2 \cdot (x_3^2 - x_1) + \rho_1 \cdot (x_1 + 2x_2 - 4)^2 + \rho_2 \cdot (x_3^2 - x_1)^2,$$

Example

which becomes:

$$\mathcal{A}_{(1,1)^\top,(1,1)^\top}(x) = (x_1+2)^2+(x_2-3)^2+(x_1+2x_2-4) \\ + (x_3^2 - x_1) + (x_1 + 2x_2 - 4)^2 + (x_3^2 - x_1)^2,$$

if we assume starting Lagrange-multiplier and penalty-weight values of one.

Example

Starting from the point $x^0 = (-1, -1, -1)^\top$, we conduct one iteration of steepest descent. The gradient of the augmented Lagrangian function is:

$$\nabla \mathcal{A}_{(1,1)^\top,(1,1)^\top}(x) = \begin{pmatrix} 2(x_1 + 2) + 2(x_1 + 2x_2 - 4) - 2(x_3^2 - x_1) \\ 2(x_2 - 3) + 2 + 4(x_1 + 2x_2 - 4) \\ 2x_3 + 4x_3 \cdot (x_3^2 - x_1) \end{pmatrix},$$

Example

which gives us a search direction of:

$$d^0 = -\nabla \mathcal{A}_{(1,1)^\top,(1,1)^\top}(x^0) = \begin{pmatrix} 16 \\ 34 \\ 10 \end{pmatrix}.$$

Example

Applying a line search we get $\alpha^0 = 1/10$, meaning that our new point is $x^1 = (0.6, 2.4, 0)^\top$.

Example

We can now examine the changes in the objective- and constraint-function values of our original problem. We have that:

$$f(x^0) = 17,$$

$$h_1(x^0) = -7,$$

$$g_1(x^0) = 1,$$

Example

and:

$$f(x^1) = 7.12,$$
$$h_1(x^1) = 1.4,$$
$$g_1(x^1) = -0.6.$$

Example

Thus we see that in this one iteration the objective-function value of the original problem has improved and that the second constraint is no longer violated.

Although the first constraint is violated, the amount by which it is violated has come down.

We do not increase either penalty weight (i.e., neither of ρ_1 nor ρ_2) to conduct the next iteration.

Example

$$\lambda_1^1 \leftarrow \lambda_1^0 + 2\rho_1 h_1(x^1) = 3.8,$$

and:

$$\lambda_2^1 \leftarrow \lambda_2^0 + 2\rho_2 h_2(x^1) = -0.2.$$

Example

and... we keep going...

This is it!