Consider the nonlinear programming problem

\[
\begin{align*}
\text{Minimize } & \quad f(x_1, x_2, \ldots, x_n) \\
\text{subject to } & \quad h_i(x_1, x_2, \ldots, x_n) = 0, \ i = 1, 2, \ldots, m \\
& \quad a_j \leq x_j \leq b_j, \ j = 1, 2, \ldots, n
\end{align*}
\]
The GRG (Generalized Reduced Gradient) algorithm is similar in concept to the Simplex method for LP:

<table>
<thead>
<tr>
<th>SIMPLEX</th>
<th>GRG</th>
</tr>
</thead>
<tbody>
<tr>
<td>reduced costs</td>
<td>reduced gradient</td>
</tr>
<tr>
<td>basic variables</td>
<td>dependent variables</td>
</tr>
<tr>
<td>nonbasic variables</td>
<td>(state variables)</td>
</tr>
<tr>
<td></td>
<td>independent variables</td>
</tr>
<tr>
<td></td>
<td>(decision variables)</td>
</tr>
</tbody>
</table>

There are, however, several differences between the two algorithms:

In **GRG**, unlike the simplex method,

• nonbasic (independent) variables need not be at their bound (lower or upper)
• at each iteration, several nonbasic (independent) variables may have their values changed (increased or decreased)
• the basis need not change at each iteration
At the beginning of each iteration, the $n$ variables are partitioned into two sets:

- Dependent variables (one per equation)
- Independent variables

(after re-ordering the variables):

$$
x = \begin{bmatrix} x_D \\ x_I \end{bmatrix} \text{ where } \begin{cases} x_D = \text{vector of } m \text{ dependent variables} \\ x_I = \text{vector of } (n-m) \text{ independent variables} \end{cases}
$$

In the same manner, we partition the gradient of the objective and the bounds:

$$
a = \begin{bmatrix} a_D \\ a_I \end{bmatrix}, \quad b = \begin{bmatrix} b_D \\ b_I \end{bmatrix}, \quad \nabla f(x) = \begin{bmatrix} \nabla_D f(x) \\ \nabla_I f(x) \end{bmatrix}
$$

and the Jacobian matrix:

$$
J(x) = \begin{bmatrix} J_D(x) \\ J_I(x) \end{bmatrix} =
\begin{bmatrix}
\nabla_D h_1(x) & \nabla_I h_1(x) \\
\nabla_D h_2(x) & \nabla_I h_2(x) \\
\vdots & \vdots \\
\nabla_D h_m(x) & \nabla_I h_m(x)
\end{bmatrix}
$$
Suppose that we are given an initial point $X^0$ which satisfies:

1) $h_i(X^0) = 0 \ \forall i$

2) $a_D < X_D^0 < b_D \quad \text{(nondegeneracy)}$

3) $J_D(X^0)$ is nonsingular, i.e., $[J_D(X^0)]^{-1}$ exists

4) $a_I \leq X_I^0 \leq b_I$

Denote the change in $X$ by $\delta = \begin{bmatrix} \delta_D \\ \delta_I \end{bmatrix}$

For "small" $\delta$, the change in the objective is

$$\Delta f = (f(X^0 + \delta) - f(X^0)) \approx \begin{bmatrix} \nabla f(X^0) \end{bmatrix}^T \cdot \delta$$

i.e.,

$$\Delta f \approx \begin{bmatrix} \nabla_D f(X^0) \\ \nabla_I f(X^0) \end{bmatrix}^T \begin{bmatrix} \delta_D \\ \delta_I \end{bmatrix} = \nabla_D f(X^0) \cdot \delta_D + \nabla_I f(X^0) \cdot \delta_I$$
We want to choose $\delta$ so that we maintain feasibility:

$$\left( h_i(X^0 + \delta) - h_i(X^0) \right) = \Delta h_i \approx \left[ \nabla h_i(X^0) \right]^T \cdot \delta = 0 \quad \forall \ i$$

i.e.,

$$\Delta h_i \approx \nabla_D h_i(X^0) \cdot \delta_D + \nabla_I h_i(X^0) \cdot \delta_I = 0 \quad \forall \ i$$

This system of equations (linear in $\delta$) may be written:

$$\Delta h = J(X^0) \cdot \delta = J_D(X^0) \cdot \delta_D + J_I(X^0) \cdot \delta_I = 0$$

Since we assume that $J_D(X^0)$ is nonsingular,

$$J_D(X^0) \cdot \delta_D + J_I(X^0) \cdot \delta_I = 0$$

$$\Rightarrow \begin{bmatrix} \delta_D = - \left[ J_D(X^0) \right]^{-1} J_I(X^0) \cdot \delta_I \end{bmatrix}$$

This equation tells us the required changes in the dependent variables which are required to maintain feasibility when the independent variables are changed by the amount $\delta_I$. 

© D. L. Bricker, U. of IA, 1999
We now make the substitution
\[
\delta_D = - \left[ J_D(X^0) \right]^{-1} J_I(X^0) \cdot \delta_I
\]
into the estimate of change in the objective function:
\[
\Delta f \approx \nabla_D f(X^0) \cdot \delta_D + \nabla_I f(X^0) \cdot \delta_I
\]
\[
\Delta f \approx \nabla_D f(X^0) \left[ - \left[ J_D(X^0) \right]^{-1} J_I(X^0) \delta_I \right] + \nabla_I f(X^0) \delta_I
\]
\[
\Delta f \approx \left[ \nabla_I f(X^0) - \nabla_D f(X^0) \left[ J_D(X^0) \right]^{-1} J_I(X^0) \right] \delta_I \equiv \Gamma_I \delta_I
\]

That is,
\[
\Delta f \approx \Gamma_I \delta_I
\]
where the "reduced gradient" \( \Gamma_I \) is defined as
\[
\Gamma_I \equiv \nabla_I f(X^0) - \nabla_D f(X^0) \left[ J_D(X^0) \right]^{-1} J_I(X^0)
\]
This gives us an estimate of the change in the objective when we change the independent variables \( X_I \) by the amount \( \delta_I \) and change the dependent variables \( X_D \) by the amount required to maintain feasibility!
\[ \Gamma_1 = \nabla f(X^0) - \nabla D f(X^0) \left[ J_D(X^0) \right]^{-1} J_I(X^0) \]

Compare the "reduced gradient" in GRG to the "reduced cost" in the Simplex method for LP:

\[ \bar{c}_j = c_j - z_j = c_j - \pi^A_j = c_j - c_B \left[ A^B \right]^{-1} A_j \]

\[ \{ \begin{align*}
\text{simplex multiplier vector} \\
\pi = c_B \left[ A^B \right]^{-1}
\end{align*} \]

Since the objective is to be minimized, we choose to move each independent variable in the negative of the direction given by the reduced gradient, taking into account the upper & lower bounds on \( X_i \):

\[ \delta_i = \begin{cases} 
0 & \text{if } \Gamma_i > 0 \text{ and } x_i^0 = a_i \\
0 & \text{if } \Gamma_i < 0 \text{ and } x_i^0 = b_i \\
- \Gamma_i & \text{otherwise}
\end{cases} \]

\[ \forall i \in I, \delta_i \]
Once the step direction $\delta_I$ (for the independent variables) is chosen, then the step direction for the dependent variables is determined by

$$\delta_D = -[J_D(X^0)]^{-1} J_I(X^0) \cdot \delta_I$$

(By the nondegeneracy assumption, i.e.,

$$a_D < X^0_D < b_D$$

some positive step can always be made in the dependent variables.)

\[
\delta_i = \begin{cases} 
0 & \text{if } \Gamma_i > 0 \text{ and } x_i^0 = a_i \\
0 & \text{if } \Gamma_i < 0 \text{ and } x_i^0 = b_i \\
-\Gamma_i & \text{otherwise}
\end{cases} \quad \forall i \in I
\]

$$\delta_D = -[J_D(X^0)]^{-1} J_I(X^0) \cdot \delta_I$$

Note that, unlike the Simplex LP method, which chooses a single nonbasic ($\approx$ independent) variable to be changed, GRG simultaneously changes many of the independent variables!
Having found the direction \( \delta \) in which to move, we next do a one-dimensional search along this direction in order to

\[
\begin{align*}
\text{Minimize } & f(x^0 + \lambda \delta) \\
\text{subject to } & a \leq x^0 + \lambda \delta \leq b \\
\text{i.e., } & a - x^0 \leq \lambda \delta \leq b - x^0
\end{align*}
\]

This can be done by any of several one-dimensional search methods, e.g., golden section search, cubic interpolation, etc.

In general, when the constraints are nonlinear, for the optimal stepsize \( \lambda^* \), \( h(x^0 + \lambda^* \delta) \neq 0 \)

Then we need to move back onto the feasible surface by solving \( h(x)=0 \), using \( x^0 + \lambda^* \delta \) as an initial "guess" (e.g., using the Newton-Raphson method).
EXAMPLE

Minimize $f(x) = x_1^2 - x_1 - x_2$

subject to

\[
\begin{align*}
  g_1(x) &= 2x_1 + x_2 \leq 1 \\
  g_2(x) &= x_1 + 2x_2 \leq 1 \\
  x_j &\geq 0, j=1,2
\end{align*}
\]

We first write the inequality constraints as equations:

\[
\begin{align*}
  h_1(x) &= 2x_1 + x_2 + x_3 - 1 = 0 \\
  h_2(x) &= x_1 + 2x_2 + x_4 - 1 = 0
\end{align*}
\]
For standard GRG form, we need both upper & lower bounds on the variables, which we deduce:

\[ 2x_1 + x_2 \leq 1 \implies x_2 \leq 1 \]
\[ x_1 + 2x_2 \leq 1 \implies x_1 \leq 1 \]

\[ \begin{align*}
  x_3 &= 1 - (2x_1 + x_2) \\
  2x_1 + x_2 &\geq 0
\end{align*} \implies x_3 \leq 1 \]

\[ \begin{align*}
  x_4 &= 1 - (x_1 + 2x_2) \\
  x_1 + 2x_2 &\geq 0
\end{align*} \implies x_4 \leq 1 \]

**Standard Form**

Minimize \( f(x) = x_1^2 - x_1 - x_2 \)
subject to
\[ h_1(x) = 2x_1 + x_2 + x_3 - 1 = 0 \]
\[ h_2(x) = x_1 + 2x_2 + x_4 - 1 = 0 \]
\[ 0 \leq x_j \leq 1, \ j=1,2,3,4 \]

We will use as feasible starting points

\[ \begin{align*}
  x^0 &= \left( \frac{1}{4}, 0, \frac{1}{2}, \frac{3}{4} \right) \\
  x^0 &= \left( \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4} \right)
\end{align*} \]
\[ x^0 = \left( \frac{1}{4}, 0, \frac{1}{2}, \frac{3}{4} \right) \]

- at lower bound

To avoid degeneracy in the initial partition, we cannot allow \( x_2 \) to be dependent ("basic"), and so our choice of two dependent variables is limited to \( x_1, x_3, \) and \( x_4. \)

For the starting partition of the variables, let's define (arbitrarily)

\[ x_I = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad x_D = \begin{bmatrix} x_3 \\ x_4 \end{bmatrix}, \quad D = \{3, 4\} \quad \text{and} \quad I = \{1, 2\} \]

\[ \nabla f(x) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \frac{\partial f}{\partial x_3} \\ \frac{\partial f}{\partial x_4} \end{bmatrix} = \begin{bmatrix} 2x_1 - 1 \\ -1 \\ 0 \\ 0 \end{bmatrix} \quad \nabla f(x^0) = \begin{bmatrix} -\frac{1}{2} \\ -1 \end{bmatrix} \quad \nabla_D f(x^0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \]
\[ \mathbf{J}(x) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{J}_1(x) = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \]

\[ \Gamma_1 = \nabla f(x^0) - \nabla Df(x^0) \left[ J_D \right]^{-1} J_1 \]

\[ \Rightarrow \begin{bmatrix} \Gamma_1 \\ \Gamma_2 \end{bmatrix} = \begin{bmatrix} -1/2 \\ -1 \end{bmatrix} \]

\[ \begin{bmatrix} \Gamma_1 \\ \Gamma_2 \end{bmatrix} = \begin{bmatrix} -1/2 \\ -1 \end{bmatrix} \quad \& \quad x^0 = \left( \frac{1}{4}, 0, \frac{1}{2}, \frac{3}{4} \right) \]

Computing the step direction:

\[ 0 < x_1^0 < 1 \]
\[ 0 = x_2^0 < 1 \]

\[ \delta_1 = -\Gamma_1 = 1/2 \]
\[ \delta_2 = -\Gamma_2 = 1 \]

(Neither independent variable is at its upper bound, and so \( \delta_1 = -\Gamma_1 \))
\[
\delta_I = \begin{bmatrix}
1/2 \\
1
\end{bmatrix}
\]
\[
\delta_D = -\left[ J_D(X^0) \right]^{-1} J_I(X^0) \cdot \delta_I
\]
\[
\Rightarrow \delta_D = -\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1/2 \\ 1 \end{bmatrix} = \begin{bmatrix} -2 \\ -5/2 \end{bmatrix}
\]
<table>
<thead>
<tr>
<th>i</th>
<th>Lower End</th>
<th>Upper End</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Gradient for objective function

```
G←GRADIENT X;Q;C
Q←2 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
C←-1 -1
G←C+2×Q+,×21X
G←G,0 0
```

Jacobian of Equality Constraints

```
J←JACOBIAN X;COEF
COEF←2 4 2 1 1 0 1 2 0 1
J←COEF
```

©D.L. Bricker, U. of IA, 1999
Iteration 1

\[ x = 0.25 \ 0 \ 0.5 \ 0.75 \]
\[ F(x) = -0.1875 \]
Dependent Index Set: 3 4
Independent Index Set: 1 2
\[ h(x) = 0 \ 0 \]
Gradient = -0.5 -1 0 0

Negative of Reduced Gradient = 0.5 1
Search Direction = 0.5 1 -2 -2.5
(Normalized Search Direction = 0.2 0.4 -0.8 -1)

\[ \delta \] was normalized by scaling so that
\[ \max |\delta_i| = 1 \]

Computing Maximum Step Size

Based upon the lower & upper bounds:

\[
\begin{cases}
0 \leq x_1 + \lambda \delta_1 = \frac{1}{4} + \frac{4}{5} \lambda \leq 1 \Rightarrow \lambda \leq \frac{5}{4} \times \frac{3}{4} = \frac{15}{16} \\
0 \leq x_2 + \lambda \delta_2 = 0 + \frac{2}{5} \lambda \leq 1 \Rightarrow \lambda \leq \frac{5}{2} \\
0 \leq x_3 + \lambda \delta_3 = \frac{1}{2} - \frac{4}{5} \lambda \leq 1 \Rightarrow \lambda \leq \frac{5}{4} \times \frac{1}{2} = \frac{5}{8} \\
0 \leq x_4 + \lambda \delta_4 = \frac{3}{4} - \lambda \leq 1 \Rightarrow \lambda \leq \frac{3}{4}
\end{cases}
\]

\{ lowest upper bound \}

©D.L. Brickor, U. of IA, 1999
Max Step Size = 0.625 = \frac{5}{8} \begin{cases} \text{lowest upper bound!} \\
\end{cases}

Optimal Step Size = 0.625

x = 0.375 0.25 0 0.125

h(x) = 0 0, F(x) = -0.484375

Note that x (which contributed the maximum stepsize) has reached its lower bound!

X_3 is the slack in the inequality constraint, and so GRG has moved to the boundary of that constraint as X_3 decreases to 0.
Dependent variables cannot be at either lower or upper bound, and so $X_3$ must become independent, and replaced by either $X_1$ or $X_2$. 

($X_4$ is already dependent.)

Variable(s) 3 has reached a bound and must be removed from D

Variables 1 2 are candidates to enter D

Try entering variable 1

Determinant of $J[;D]$ = 2 checking that the Jacobian submatrix is nonsingular!

3 is replaced by 1 in Dependent Variable Set.

$h(x)=0$ 0, $F(x)= -0.484375$

---

**Iteration 2**

$x = 0.375 0.25 0 0.125$

$F(x)= -0.484375$

Dependent Index Set: 1 4

Independent Index Set: 3 2

$h(x) = 0$ 0

Gradient = $-0.25$ $-1$ 0 0

Negative of Reduced Gradient = $-0.125$ 0.875

Search Direction = $-0.4375$ 0.875 0 $-1.3125$

(Normalized Search Direction = $-0.333333$ 0.866667 0 $-$

Max Step Size = 0.125

Optimal Step Size = 0.125

$x = 0.333333 0.333333 0 0$

$h(x) = 0$ 0, $F(x)= -0.555556$
Variable(s) 4 has reached a bound
and must be removed from D

Variables 2 are candidates to enter D

Try entering variable 2

Determinant of JT;D] = 3

4 is replaced by 2 in Dependent Variable Set.

h(x) = 0 0, F(x) = -0.555556

Iteration 3

x = 0.333333 0.333333 0 0
F(x) = -0.555556

Dependent Index Set: 1 2
Independent Index Set: 3 4

h(x) = 0 0
Gradient = -0.333333 -1 0 0

Negative of Reduced Gradient = 0.111111 -0.555556

Search Direction = -0.0740741 0.037037 0.111111 0
(Normalized Search Direction = -0.666667 0.333333 1 0

Max Step Size = 0.5
Optimal Step Size = 0.125

x = 0.25 0.375 0.125 0
h(x) = -1.11022E-16 -1.11022E-16, F(x) = -0.5625
Iteration 4

\( \mathbf{x} = 0.25 \ 0.375 \ 0.125 \ 0 \)

\( F(\mathbf{x}) = -0.5625 \)

Dependent Index Set: 1 2
Independent Index Set: 3 4

\( h(\mathbf{x}) = -1.11022E-16 \ -1.11022E-16 \)

Gradient = \(-0.5 \ -1 \ 0 \ 0 \)

Negative of Reduced Gradient = \(0 \ -0.5 \)

*** GRG HAS CONVERGED ***

Generalized Reduced Gradient Solution

\( \mathbf{x} = 0.25 \ 0.375 \ 0.125 \ 0 \)

\( F(\mathbf{x}) = -0.5625 \)

\( \mathbf{v} F(\mathbf{x}) = -0.5 \ -1 \ 0 \ 0 \)

\( h(\mathbf{x}) = -1.11022E-16 \ -1.11022E-16 \)

©D.L. Bricker, U. of IA, 1999
Iteration 1

\[ x = 0.25 \ 0.25 \ 0.25 \ 0.25 \ 0.25 \]
\[ F(x) = -0.4375 \]

Dependent Index Set: 3 4
Independent Index Set: 1 2

\[ h(x) = 0 \ 0 \]

Gradient = -0.5 -1 0 0

Negative of Reduced Gradient = 0.5 1

Search Direction = 0.5 1 -2 -2.5
(Normalized Search Direction = 0.2 0.4 -0.8 -1)

Max Step Size = 0.25
Optimal Step Size = 0.25
\[ x = 0.3 \ 0.35 \ 0.05 \ 0 \]
\[ h(x) = 0 \ 0, \ F(x) = -0.56 \]

Variable(s) 4 has reached a bound and must be removed from D

Variables 1 2 are candidates to enter D
Try entering variable 2
Determinant of J[^1][D] = 2

4 is replaced by 2 in Dependent Variable Set.
\[ h(x) = 0 \ 0, \ F(x) = -0.56 \]
Iteration 2

\[ x = 0.3 \ 0.35 \ 0.05 \ 0 \]
\[ F(x) = -0.56 \]

Dependent Index Set: 3 2
Independent Index Set: 1 4

\[ h(x) = 0 \ 0 \]
Gradient = -0.4 -1 0 0
Negative of Reduced Gradient = -0.1 -0.5

Search Direction = -0.1 0.05 0.15 0
(Normalized Search Direction = -0.666667 0.333333 1 0)

Max Step Size = 0.45
Optimal Step Size = 0.075

\[ x = 0.25 \ 0.375 \ 0.125 \ 0 \]
\[ h(x)=0 \ 0, \ F(x)= -0.5625 \]

---

Iteration 3

\[ x = 0.25 \ 0.375 \ 0.125 \ 0 \]
\[ F(x) = -0.5625 \]

Dependent Index Set: 3 2
Independent Index Set: 1 4

\[ h(x) = 0 \ 0 \]
Gradient = -0.5 -1 0 0
Negative of Reduced Gradient = 2.22045E-16 -0.5

*** GRG HAS CONVERGED ***

---

Generalized Reduced Gradient Solution

\[ x = 0.25 \ 0.375 \ 0.125 \ 0 \]
\[ F(x) = -0.5625 \]
\[ \nabla F(x) = -0.5 -1 0 0 \]
\[ h(x) = 0 \ 0 \]

@D.L. Bricker, U. of IA, 1999