$$\nabla f(\mathbf{x}_k)^t\mathbf{d}_k \le z_k < -\varepsilon \qquad \text{for } k \in \mathcal{K}' \text{ sufficiently large} \tag{10.4}$$

$$g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^t\mathbf{d}_k \le z_k < -\varepsilon \qquad \text{for } k \in \mathcal{K}' \text{ sufficiently large, for } i = 1, \ldots, m \tag{10.5}$$

By continuous differentiability of f, (10.4) implies that $\nabla f(\mathbf{x})^t\mathbf{d} < 0$.

Since g_i is continuously differentiable, from (10.5), there exists a $\delta > 0$ such that the following inequality holds for each $\lambda \in [0, \delta]$:

$$g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k + \lambda\mathbf{d}_k)^t\mathbf{d}_k < -\frac{\varepsilon}{2} \qquad \text{for } k \in \mathcal{K}' \text{ sufficiently large, for } i = 1, \ldots, m \tag{10.6}$$

Now let $\lambda \in [0, \delta]$. By the mean value theorem, and since $g_i(\mathbf{x}_k) \le 0$ for each k and each i, we get

$$\begin{aligned} g_i(\mathbf{x}_k + \lambda\mathbf{d}_k) &= g_i(\mathbf{x}_k) + \lambda\nabla g_i(\mathbf{x}_k + \alpha_{ik}\lambda\mathbf{d}_k)^t\mathbf{d}_k \\ &= (1 - \lambda)g_i(\mathbf{x}_k) + \lambda[g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k + \alpha_{ik}\lambda\mathbf{d}_k)^t\mathbf{d}_k] \\ &\le \lambda[g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k + \alpha_{ik}\lambda\mathbf{d}_k)^t\mathbf{d}_k] \end{aligned} \tag{10.7}$$

where $\alpha_{ik} \in [0, 1]$. Since $\alpha_{ik}\lambda \in [0, \delta]$, from (10.6) and (10.7), it follows that $g_i(\mathbf{x}_k + \lambda\mathbf{d}_k) \le -\lambda\varepsilon/2 \le 0$ for $k \in \mathcal{K}'$ sufficiently large and for $i = 1, \ldots, m$. This shows that $\mathbf{x}_k + \lambda\mathbf{d}_k$ is feasible for each $\lambda \in [0, \delta]$, for all $k \in \mathcal{K}'$ sufficiently large.

To summarize, we have exhibited a sequence $\{(\mathbf{x}_k, \mathbf{d}_k)\}_{\mathcal{K}'}$ that satisfies conditions 1 through 4 of Lemma 10.2.6. By the lemma, however, the existence of such a sequence is not possible. This contradiction shows that $\mathbf{x}$ is a Fritz John point, and the proof is complete.

10.3 Successive Linear Programming Approach

In our foregoing discussion of Zoutendijk's algorithm and its convergent variant as proposed by Topkis and Veinott, we have learned that at each iteration of this method, we solve a direction-finding linear programming problem based on first-order functional approximations in a minimax framework and then conduct a line search along this direction. Conceptually, this is similar to *successive linear programming (SLP)* approaches, also known as *sequential*, or *recursive*, linear programming. Here, at each iteration k, a direction-finding linear program is formulated based on first-order Taylor series approximations to the objective and constraint functons, in addition to appropriate *step bounds* or trust region restrictions on the direction components. If $\mathbf{d}_k = \mathbf{0}$ solves this problem, then the current iterate $\mathbf{x}_k$ is optimal to the first-order approximation, and so, from Theorem 4.2.15, this solution is a KKT point and we terminate the procedure. Otherwise, the procedure either accepts the new iterate $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k$, or rejects this iterate and reduces the step bounds, and then repeats this process. The decision as to whether to accept or reject the new iterate is typically made based on a *merit function* fashioned around l_1, or the absolute value penalty function [see equation (9.8)].

The philosophy of this approach was introduced by Griffith and Stewart of the Shell Development Company in 1961, and has been widely used since then, particularly in the oil and chemical industries (see Exercise 10.55). The principal advantage of this type of method is its ease and robustness in implementation for large-scale problems, given an efficient and stable linear programming solver. As can be expected, if the optimum is a vertex of the (linearized) feasible region, then a rapid convergence is obtained. Indeed, once the algorithm enters a relatively close neighborhood of such a solution, it essentially behaves like Newton's algorithm applied to the binding constraints (under suitable regularity assumptions), with the Newton iterate being the (unique) linear programming solution, and a quadratic convergence rate obtains. Hence, highly constrained nonlinear programming problems that have nearly as many linearly independent active constraints as variables are very suitable for this class of algorithms. Real-world nonlinear refinery models tend to be of this nature, and problems of up to 1000 rows have been successfully solved. On the negative side, SLP algorithms exhibit slow convergence to nonvertex solutions, and they also have the disadvantage of violating nonlinear constraints en route to optimality.

Below, we describe an SLP algorithm, called the *penalty successive linear programming (PSLP)* algorithm, which employs the l_1 penalty function more actively in the direction-finding problem, itself, rather than as only a merit function, and enjoys good robustness and convergence properties. The problem we consider is of the form:

$$\begin{array}{llll} P: & \text{Minimize} & f(\mathbf{x}) & \\ & \text{subject to} & g_i(\mathbf{x}) \leq 0 & \text{for } i = 1, \ldots, m \\ & & h_i(\mathbf{x}) = 0 & \text{for } i = 1, \ldots, l \\ & & \mathbf{x} \in X = \{\mathbf{x} : \mathbf{A}\mathbf{x} \leq \mathbf{b}\} & \end{array}$$

where all functions are assumed to be continuously differentiable, where $\mathbf{x} \in E_n$, and where the linear constraints defining the problem have all been accommodated into the set X.

Now let $F_E(\mathbf{x})$ be the l_1, or absolute value, exact penalty function of equation (9.8), restated below for a penalty parameter $\mu > 0$:

$$F_E(\mathbf{x}) = f(\mathbf{x}) + \mu\left[\sum_{i=1}^{m} \max\{0, g_i(\mathbf{x})\} + \sum_{i=1}^{l} |h_i(\mathbf{x})|\right] \tag{10.8}$$

Accordingly, consider the following (linearly constrained) penalty problem PP

$$PP: \quad \text{Minimize}\{F_E(\mathbf{x}) : \mathbf{x} \in X\} \tag{10.9a}$$

Substituting y_i for $\max\{0, g_i(\mathbf{x})\}$, $i = 1, \ldots, m$, and writing $h_i(\mathbf{x})$ as the difference $z_i^+ - z_i^-$ of two nonnegative variables, where $|h_i(\mathbf{x})| = z_i^+ + z_i^-$, for $i = 1, \ldots, l$, we can equivalently rewrite (10.9a) without the nondifferentiable terms as follows:

$$\begin{array}{llll} PP: & \text{Minimize} & f(\mathbf{x}) + \mu\left[\sum_{i=1}^{m} y_i + \sum_{i=1}^{l} (z_i^+ + z_i^-)\right] & \\ & \text{subject to} & y_i \geq g_i(\mathbf{x}) & i = 1, \ldots, m \\ & & z_i^+ - z_i^- = h_i(\mathbf{x}) & i = 1, \ldots, l \\ & & \mathbf{x} \in X, y_i \geq 0 & \text{for } i = 1, \ldots, m \\ & & z_i^+ \text{ and } z_i^- \geq 0 & \text{for } i = 1, \ldots, l \end{array} \tag{10.9b}$$

Note that, given any $\mathbf{x} \in X$, since $\mu > 0$, the optimal completion $(\mathbf{y}, \mathbf{z}^+, \mathbf{z}^-) = (y_1, \ldots, y_m, z_1^+, \ldots, z_l^+, z_1^- \ldots, z_l^-)$ is determined by letting

$$\begin{array}{lll} y_i = \max\{0, g_i(\mathbf{x})\} & & i = 1, \ldots, m \\ z_i^+ = \max\{0, h_i(\mathbf{x})\} & z_i^- = \max\{0, -h_i(\mathbf{x})\} & i = 1, \ldots, l \\ \text{so that } (z_i^+ + z_i^-) = |h_i(\mathbf{x})| & & i = 1, \ldots, l \end{array} \tag{10.10}$$

Consequently, (10.9b) is equivalent to (10.9a) and may be essentially viewed as also being a problem in the **x**-variable space. Moreover, under the condition of Theorem 9.3.1, if μ is sufficiently large and if $\bar{\mathbf{x}}$ is an optimum for P, then $\bar{\mathbf{x}}$ solves PP. Alternatively, as in Exercise 9.12, if μ is sufficiently large and if $\bar{\mathbf{x}}$ satisfies the second-order sufficiency conditions for P, then $\bar{\mathbf{x}}$ is a strict local minimum for PP. In either case, μ must be, at least, as large as the absolute value of any Lagrange multiplier associated with the constraints $\mathbf{g}(\mathbf{x}) \leq \mathbf{0}$ and $\mathbf{h}(\mathbf{x}) = \mathbf{0}$ in P. Note that instead of using a single penalty parameter μ, we can employ a set of parameters $\mu_1, \ldots, \mu_{m+l}$, one associated with each of the penalized constraints. Selecting some reasonably large values for these parameters (assuming a well-scaled problem), we can solve PP; and if an infeasible solution results, then these parameters can be manually increased and the process repeated. We shall assume, however, that we have selected some suitably large, admissible value of a single penalty parameter μ. With this motivation, the algorithm PSLP seeks to solve Problem PP, using a box step or hypercube first-order trust region approach as introduced in Section 8.7.

Specifically, this approach proceeds as follows. Given a current iterate $\mathbf{x}_k \in X$ and a trust region or step bound vector $\boldsymbol{\Delta}_k \in E_n$, consider the following linearization of PP, given by (10.9a), where we have also imposed a trust region bound based on the l_∞ or sup-norm.

$$
\begin{aligned}
LP(\mathbf{x}_k,\boldsymbol{\Delta}_k):\quad \text{Minimize}\quad & F_{EL_k}(\mathbf{x}) \equiv f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^t(\mathbf{x}-\mathbf{x}_k) \\
& + \mu\left[\sum_{i=1}^{m} \max\{0, g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^t(\mathbf{x}-\mathbf{x}_k)\}\right. \\
& \left. + \sum_{i=1}^{l} |h_i(\mathbf{x}_k) + \nabla h_i(\mathbf{x}_k)^t(\mathbf{x}-\mathbf{x}_k)|\right] \\
\text{subject to}\quad & \mathbf{x} \in X \equiv \{\mathbf{x}: \mathbf{A}\mathbf{x} \leq \mathbf{b}\} \\
& -\boldsymbol{\Delta}_k \leq \mathbf{x}-\mathbf{x}_k \leq \boldsymbol{\Delta}_k
\end{aligned}
\qquad (10.11a)
$$

Similar to (10.9) and (10.10), this can be equivalently restated as the following *linear programming problem*, where we have also used the substitution $\mathbf{x} = \mathbf{x}_k + \mathbf{d}$ and have dropped the constant $f(\mathbf{x}_k)$ from the objective function:

$$
\begin{aligned}
LP(\mathbf{x}_k,\boldsymbol{\Delta}_k):\quad \text{Minimize}\quad & \nabla f(\mathbf{x}_k)^t\mathbf{d} + \mu\left[\sum_{i=1}^{m} y_i + \sum_{i=1}^{l} (z_i^+ + z_i^-)\right] \\
\text{subject to}\quad & y_i \geq g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^t\mathbf{d} && i = 1,\ldots,m \\
& (z_i^+ - z_i^-) = h_i(\mathbf{x}_k) + \nabla h_i(\mathbf{x}_k)^t\mathbf{d} && i = 1,\ldots,l \\
& \mathbf{A}(\mathbf{x}_k + \mathbf{d}) \leq \mathbf{b} \\
& -\Delta_{ki} \leq d_i \leq \Delta_{ki} && i = 1,\ldots,n \\
& \mathbf{y} \geq \mathbf{0}, \mathbf{z}^+ \geq \mathbf{0}, \mathbf{z}^- \geq \mathbf{0}
\end{aligned}
\qquad (10.11b)
$$

The linear program LP($\mathbf{x}_k$, $\boldsymbol{\Delta}_k$) given by (10.11b) is the *direction-finding subproblem* that yields an optimal solution $\mathbf{d}_k$, say, along with the accompanying values of $\mathbf{y}$, $\mathbf{z}^+$, and $\mathbf{z}^-$, which are given as follows, similar to (10.10):

$$
\begin{aligned}
& y_i = \max\{0, g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^t\mathbf{d}_k\} && i = 1,\ldots,m \\
& z_i^+ = \max\{0, h_i(\mathbf{x}_k) + \nabla h_i(\mathbf{x}_k)^t\mathbf{d}_k\} && z_i^- = \max\{0, -[h_i(\mathbf{x}_k) + \nabla h_i(\mathbf{x}_k)^t\mathbf{d}_k]\} \\
& \text{so that } (z_i^+ + z_i^-) \equiv |h_i(\mathbf{x}_k) + \nabla h_i(\mathbf{x}_k)^t\mathbf{d}_k| && i = 1,\ldots,l
\end{aligned}
\qquad (10.12)
$$

As with trust region methods described in Section 8.7, the decision whether to accept or to reject the new iterate $\mathbf{x}_k + \mathbf{d}_k$ and the adjustment of the step bounds $\mathbf{\Delta}_k$ is made based on the ratio R_k of the actual decrease ΔF_{E_k} in the l_1 penalty function F_E, and the decrease ΔF_{EL_k} as predicted by its linearized version F_{EL_k}, provided that the latter is nonzero. These quantities are given as follows from (10.8) and (10.11a):

$$\Delta F_{E_k} = F_E(\mathbf{x}_k) - F_E(\mathbf{x}_k + \mathbf{d}_k) \qquad \Delta F_{EL_k} = F_{EL_k}(\mathbf{x}_k) - F_{EL_k}(\mathbf{x}_k + \mathbf{d}_k) \tag{10.13}$$

The principal concepts tying in the development presented thus far are encapsulated by the following result.

10.3.1 Theorem

Consider Problem P and the absolute value (l_1) penalty function (10.8), where μ is assumed to be large enough as in Theorem 9.3.1.

a. If the conditions of Theorem 9.3.1 hold and if $\bar{\mathbf{x}}$ solves Problem P, then $\bar{\mathbf{x}}$ also solves PP of equation (10.9a). Alternatively, if $\bar{\mathbf{x}}$ is a regular point that satisfies the second-order sufficiency conditions for P, then $\bar{\mathbf{x}}$ is a strict local minimum for PP.
b. Consider Problem PP given by (10.9b), where $(\mathbf{y}, \mathbf{z}^+, \mathbf{z}^-)$ are given by (10.10) for any $\mathbf{x} \in X$. If $\bar{\mathbf{x}}$ is a KKT solution for Problem P, then, for μ large enough, as in Theorem 9.3.1, $\bar{\mathbf{x}}$ is a KKT solution for Problem PP. Conversely, if $\bar{\mathbf{x}}$ is a KKT solution for PP and if $\bar{\mathbf{x}}$ is feasible to P, then $\bar{\mathbf{x}}$ is a KKT solution for P.
c. The solution $\mathbf{d}_k = \mathbf{0}$ is optimal for LP($\mathbf{x}_k$, $\mathbf{\Delta}_k$) defined by (10.11b) and (10.12) if and only if $\mathbf{x}_k$ is a KKT solution for PP.
d. The predicted decrease ΔF_{EL_k} in the linearized penalty function, as given by (10.13), is nonnegative, and is zero if and only if $\mathbf{d}_k = \mathbf{0}$ solves Problem LP($\mathbf{x}_k$, $\mathbf{\Delta}_k$).

Proof

The proof for part a is similar to that of Theorem 9.3.1 and of Exercise 9.12, and is left to the reader in Exercise 10.18. Next, consider part b. The KKT conditions for P require a primal feasible solution $\bar{\mathbf{x}}$, along with Lagrange multipliers $\bar{\mathbf{u}}$, $\bar{\mathbf{v}}$, and $\bar{\mathbf{w}}$ satisfying

$$\begin{gathered}\sum_{i=1}^{m} \bar{u}_i \nabla g_i(\bar{\mathbf{x}}) + \sum_{i=1}^{l} \bar{v}_i \nabla h_i(\bar{\mathbf{x}}) + \mathbf{A}^t\bar{\mathbf{w}} = -\nabla f(\bar{\mathbf{x}}) \\ \bar{\mathbf{u}} \ge \mathbf{0} \qquad \bar{\mathbf{v}} \text{ unrestricted} \qquad \bar{\mathbf{w}} \ge \mathbf{0} \\ \bar{\mathbf{u}}^t\mathbf{g}(\bar{\mathbf{x}}) = 0 \qquad \bar{\mathbf{w}}^t(\mathbf{A}\bar{\mathbf{x}} - \mathbf{b}) = 0\end{gathered} \tag{10.14}$$

Furthermore, $\bar{\mathbf{x}}$ is a KKT point for PP, with $\bar{\mathbf{x}} \in X$ and with $(\mathbf{y}, \mathbf{z}^+, \mathbf{z}^-)$ given accordingly by (10.10), provided Lagrange multipliers $\bar{\mathbf{u}}$, $\bar{\mathbf{v}}$, and $\bar{\mathbf{w}}$ exist satisfying

$$\sum_{i=1}^{m} \bar{u}_i \nabla g_i(\bar{\mathbf{x}}) + \sum_{i=1}^{l} \bar{v}_i \nabla h_i(\bar{\mathbf{x}}) + \mathbf{A}^t\bar{\mathbf{w}} = -\nabla f(\bar{\mathbf{x}}) \tag{10.15a}$$

$$0 \le \bar{u}_i \le \mu \quad (\bar{u}_i - \mu)y_i = 0 \quad \bar{u}_i[y_i - g_i(\bar{\mathbf{x}})] = 0 \quad \text{for } i = 1, \ldots, m \tag{10.15b}$$

$$|\bar{v}_i| \le \mu \quad z_i^+(\mu - \bar{v}_i) = 0 \quad z_i^-(\mu + \bar{v}_i) = 0 \quad \text{for } i = 1, \ldots, l \tag{10.15c}$$

$$\bar{\mathbf{w}}^t(\mathbf{A}\bar{\mathbf{x}} - \mathbf{b}) = 0 \qquad \bar{\mathbf{w}} \ge \mathbf{0} \tag{10.15d}$$

Now let $\bar{\mathbf{x}}$ be a KKT solution for Problem P, with Lagrange multipliers $\bar{\mathbf{u}}$, $\bar{\mathbf{v}}$, and $\bar{\mathbf{w}}$ satisfying (10.14). Defining $(\mathbf{y}, \mathbf{z}^+, \mathbf{z}^-)$ according to (10.10), we get $\mathbf{y} = \mathbf{0}$, $\mathbf{z}^+ = \mathbf{z}^- = \mathbf{0}$; and so, for μ large enough, as in Theorem 9.3.1, $\bar{\mathbf{x}}$ is a KKT solution for PP by (10.15). Conversely, let $\bar{\mathbf{x}}$ be a KKT solution for PP and suppose that $\bar{\mathbf{x}}$ is feasible to Problem P. Then, we again have $\mathbf{y} = \mathbf{0}$, $\mathbf{z}^+ = \mathbf{z}^- = \mathbf{0}$ by (10.10); and so, by (10.15) and (10.14), $\bar{\mathbf{x}}$ is a KKT solution for Problem P. This proves part b.

Part c follows from Theorem 4.2.15, noting that LP($\mathbf{x}_k$, $\boldsymbol{\Delta}_k$) represents a first-order linearization of PP at the point $\mathbf{x}_k$ and that the step bounds $-\boldsymbol{\Delta}_k \le \mathbf{d} \le \boldsymbol{\Delta}_k$ are nonbinding at $\mathbf{d}_k = \mathbf{0}$, that is, at $\mathbf{x} = \mathbf{x}_k$.

Finally, consider part d. Since $\mathbf{d}_k$ minimizes LP($\mathbf{x}_k$, $\boldsymbol{\Delta}_k$) in (10.11b), $\mathbf{x} = \mathbf{x}_k + \mathbf{d}_k$ minimizes (10.11a); and so, since $\mathbf{x}_k$ is feasible to (10.11a), we have $F_{EL_k}(\mathbf{x}_k) \ge F_{EL_k}(\mathbf{x}_k + \mathbf{d}_k)$, or that $\Delta F_{EL_k} \ge 0$. By the same token, this difference is zero if and only if $\mathbf{d}_k = \mathbf{0}$ is optimal for LP($\mathbf{x}_k$, $\boldsymbol{\Delta}_k$), and this completes the proof.

Summary of the Penalty Successive Linear Programming (PSLP) Algorithm

Initialization Put the iteration counter $k = 1$, and select a starting solution $\mathbf{x}_k \in X$ feasible to the linear constraints, along with a step bound or trust region vector $\boldsymbol{\Delta}_k > \mathbf{0}$ in E_n. Let $\boldsymbol{\Delta}_{\text{LB}} > \mathbf{0}$ be some small lower bound tolerance on $\boldsymbol{\Delta}_k$. (Sometimes, $\boldsymbol{\Delta}_{\text{LB}} = \mathbf{0}$ is also used.) Also, select a suitable value of the penalty parameter μ (or values for penalty parameters $\mu_1, \ldots, \mu_{m+l}$ as discussed above). Choose values for the scalars $0 < \rho_0 < \rho_1 < \rho_2 < 1$ to be used in the trust region ratio test, and for the step bound adjustment multiplier $\beta \in (0, 1)$. (Typically, $\rho_0 = 10^{-6}$, $\rho_1 = 0.25$, $\rho_2 = 0.75$, and $\beta = 0.5$.)

Step 1 Linear Programming Subproblem Solve the linear program LP($\mathbf{x}_k$, $\boldsymbol{\Delta}_k$) to obtain an optimum $\mathbf{d}_k$. Compute the actual and the predicted decreases ΔF_{E_k} and ΔF_{EL_k}, respectively, in the penalty function as given by (10.13). If $\Delta F_{EL_k} = 0$ (equivalently, by Theorem 10.3.1d, if $\mathbf{d}_k = \mathbf{0}$), then stop. Otherwise, compute the ratio $R_k = \Delta F_{E_k}/\Delta F_{EL_k}$. If $R_k < \rho_0$, then, since $\Delta F_{EL_k} \ge 0$ by Theorem 10.3.1d, the penalty function has either worsened or its improvement is insufficient. Hence, reject the current solution, shrink $\boldsymbol{\Delta}_k$ to $\beta\boldsymbol{\Delta}_k$, and repeat this step. (Zhang, Kim, and Lasdon [1985] show that within a finite number of such reductions, we will have $R_k \ge \rho_0$. Note that, while R_k remains less than ρ_0, components of $\boldsymbol{\Delta}_k$ may shrink below those of $\boldsymbol{\Delta}_{\text{LB}}$.) On the other hand, if $R_k \ge \rho_0$, proceed to step 2.

Step 2 New Iterate and Adjustment of Step Bounds Let $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k$. If $0 \le R_k < \rho_1$, then shrink $\boldsymbol{\Delta}_k$ to $\boldsymbol{\Delta}_{k+1} = \beta\boldsymbol{\Delta}_k$, since the penalty function has not improved sufficiently. If $\rho_1 \le R_k \le \rho_2$, then retain $\boldsymbol{\Delta}_{k+1} = \boldsymbol{\Delta}_k$. On the other hand, if $R_k > \rho_2$, then amplify the trust region by letting $\boldsymbol{\Delta}_{k+1} = \boldsymbol{\Delta}_k/\beta$. In all cases, replace $\boldsymbol{\Delta}_{k+1}$ by $\max\{\boldsymbol{\Delta}_{k+1}, \boldsymbol{\Delta}_{\text{LB}}\}$, where the max $\{\cdot\}$ is taken componentwise. Increment k by 1 and return to step 1.

A few comments are in order at this point. First, note that the linear program (10.11b) is feasible and bounded ($\mathbf{d} = \mathbf{0}$ is a feasible solution) and that it preserves any sparsity structure of the original problem. Second, if there are any variables that appear linearly in the objective function as well as in the constraints of P, then the corresponding step bounds for such variables can be taken as some arbitrarily large value M and can be retained at that value throughout the procedure. Third, when termination occurs at step 1, then, by Theorem 10.3.1, $\mathbf{x}_k$ is a KKT solution for PP; and if $\mathbf{x}_k$ is feasible to P, then it is also a KKT solution for P. (Otherwise, the penalty parameters may need to be increased as discussed earlier.) Fourth, it can be shown that either the algorithm terminates finitely or, else, an infinite sequence $\{\mathbf{x}_k\}$ is generated such that if the level set $\{\mathbf{x} \in X : F_E(\mathbf{x}) \le F_E(\mathbf{x}_1)\}$ is bounded, then $\{\mathbf{x}_k\}$ has an accumulation point, and every such accumulation point is a KKT solution for Problem PP. Finally, the stopping criterion of step 1 is usually replaced by several practical termination criteria. For example, if the fractional change in the l_1 penalty function is less than a tolerance ($\varepsilon = 10^{-4}$) for some c ($= 3$) consecutive iterations, or if the iterate is ε-feasible and either the KKT conditions are satisfied within an ε-tolerance, or if the fractional change in the objective function value for Problem P is less than ε for c consecutive iterations, then the procedure can be terminated. Also, the amplification or reduction in the step bounds are often modified in implementations so as to treat deviations of R_k from unity symmetrically. For example, at step 2, if $|1 - R_k| < 0.25$, then all step bounds are amplified by dividing by $\beta = 0.5$; and if $|1 - R_k| > 0.75$, then all step bounds are reduced by multiplying by β. In addition, if any nonlinear variable remains at the same step bound for c ($= 3$) consecutive iterations, then its step bound is amplified by dividing by β.

10.3.2 Example

Consider the problem

$$\begin{aligned} \text{Minimize} \quad & f(\mathbf{x}) = 2x_1^2 + 2x_2^2 - 2x_1x_2 - 4x_1 - 6x_2 \\ \text{subject to} \quad & g_1(\mathbf{x}) = 2x_1^2 - x_2 \le 0 \\ & \mathbf{x} \in X = \{\mathbf{x} = (x_1, x_2) : x_1 + 5x_2 \le 5, \mathbf{x} \ge \mathbf{0}\} \end{aligned}$$

Figure 10.13*a* provides a sketch for the graphical solution of this problem. Note that this problem has a "vertex" solution, and thus we might expect a rapid convergence behavior. Let us begin with the solution $\mathbf{x}_1 = (0, 1)^t \in X$ and use $\mu = 10$ (which can be verified to be sufficiently large—see Exercise 10.19). Let us also select $\mathbf{\Delta}_1 = (1, 1)^t$, $\mathbf{\Delta}_{LB} = (10^{-6}, 10^{-6})^t$, $\rho_0 = 10^{-6}$, $\rho_1 = 0.25$, $\rho_2 = 0.75$, and $\beta = 0.5$.

The linear program LP($\mathbf{x}_1$, $\mathbf{\Delta}_1$) given by (10.11b) now needs to be solved, as, for example, by the simplex method. To graphically illustrate the process, consider the equivalent problem (10.11a). Noting that $\mathbf{x}_1 = (0, 1)^t$, $\mu = 10$, $f(\mathbf{x}_1) = -4$, $\nabla f(\mathbf{x}_1) = (-6, -2)^t$, $g_1(\mathbf{x}_1) = -1$, and $\nabla g_1(\mathbf{x}_1) = (0, -1)^t$, we have

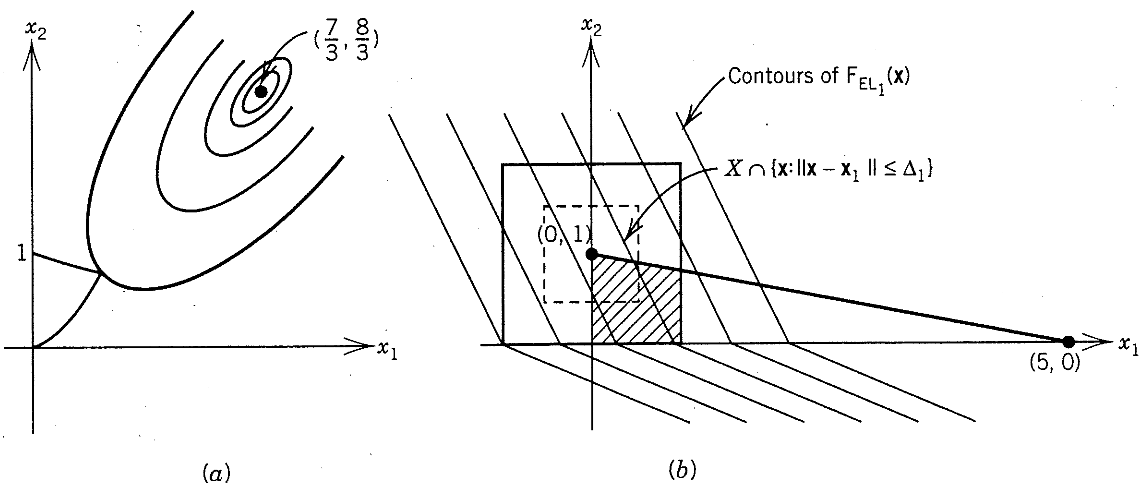

Figure 10.13 Illustration for Example 10.3.2.

$$F_{EL_1}(\mathbf{x}) = -2 - 6x_1 - 2x_2 + 10 \max\{0, -x_2\} \tag{10.16}$$

The solution of LP($\mathbf{x}_1$, $\boldsymbol{\Delta}_1$) via (10.11a) is depicted in Figure 10.13b. The optimum solution is $\mathbf{x} = (1, \frac{4}{5})^t$, so that $\mathbf{d}_1 = (1, \frac{4}{5})^t - (0, 1)^t = (1, -\frac{1}{5})^t$ solves (10.11b). From (10.13), using (10.8) and (10.16) along with $\mathbf{x}_1 = (0, 1)^t$ and $\mathbf{x}_1 + \mathbf{d}_1 = (1, \frac{4}{5})^t$, we get $\Delta F_{E_k} = -4.88$ and $\Delta F_{EL_k} = \frac{28}{5}$. Hence, the penalty function has worsened, and so we reduce the step bounds at step 1 itself and repeat this step with the revised $\boldsymbol{\Delta}_1 = (0.5, 0.5)^t$.

The revised step bound box is shown dotted in Figure 10.13*b*. The corresponding optimal solution is $\mathbf{x} = (0.5, 0.9)^t$, which corresponds to the optimum $\mathbf{d}_1 = (0.5, 0.9)^t - (0, 1)^t = (0.5, -0.1)^t$ for the problem (10.11b). From (10.13), using (10.8) and (10.16) along with $\mathbf{x}_1 = (0, 1)^t$ and $\mathbf{x}_1 + \mathbf{d}_1 = (0.5, 0.9)^t$, we get $\Delta F_{E_k} = 2.18$ and $\Delta F_{EL_k} = 2.8$, which gives $R_k = 2.18/2.8 = 0.7786$. We therefore accept this solution as the new iterate $\mathbf{x}_2 = (0.5, 0.9)^t$ and, since $R_k > \rho_2 = 0.75$, we amplify the trust region by letting $\boldsymbol{\Delta}_2 = \boldsymbol{\Delta}_1/\beta = (1, 1)^t$. We now ask the reader (see Exercise 10.19) to continue this process until a suitable termination criterion is satisfied, as discussed above.

10.4 Successive Quadratic Programming or Projected Lagrangian Approach

We have seen that Zoutendijk's algorithm as well as Topkis and Veinott's modification of this procedure are prone to zigzagging and slow convergence behavior because of the first-order approximations employed. The SLP approach enjoys a quadratic rate of convergence if the optimum occurs at a vertex of the feasible region, because then, the method begins to imitate Newton's method applied to the active constraints. However, for nonvertex solutions, this method again, being essentially a first-order approximation procedure, can succumb to a slow convergence process. To alleviate this behavior, we can employ second-order approximations and derive a *successive quadratic programming approach (SQP)*.

SQP methods, also known as *sequential*, or *recursive, quadratic programming*, employ Newton's method (or quasi-Newton methods) to directly solve the KKT conditions for the original problem. As a result, the accompanying subproblem turns out to be the minimization of a quadratic approximation to the Lagrangian function optimized over a linear approximation to the constraints. Hence, this type of process is also known as a *projected Lagrangian*, or the *Lagrange–Newton*, approach. By its nature, this method produces both primal and dual (Lagrange multiplier) solutions.

To present the concept of this method, consider the equality constrained nonlinear problem, where $\mathbf{x} \in E_n$, and all functions are assumed to be continuously twice-differentiable.

$$\begin{aligned} P:\ \ &\text{Minimize} && f(\mathbf{x}) \\ &\text{subject to} && h_i(\mathbf{x}) = 0 \qquad i = 1, \ldots, l \end{aligned} \tag{10.17}$$

The extension for including inequality constraints is motivated by the following analysis for the equality constrained case and is considered subsequently.

The KKT optimality conditions for Problem P require a primal solution $\mathbf{x} \in E_n$ and a Lagrange multiplier vector $\mathbf{v} \in E_l$ such that