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

The solution of LP($\mathbf{x}_1$, $\mathbf{\Delta}_1$) 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 $\mathbf{\Delta}_1 = (0.5, 0.5)^t$.

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 $\mathbf{\Delta}_2 = \mathbf{\Delta}_1/\beta = (1, 1)^t$. 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.

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

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

$$\begin{aligned} \nabla f(\mathbf{x}) + \sum_{i=1}^{l} v_i \nabla h_i(\mathbf{x}) &= \mathbf{0} \\ h_i(\mathbf{x}) = 0 \qquad i &= 1, \ldots, l \end{aligned} \tag{10.18}$$

Let us write this system of equations more compactly as $\mathbf{W}(\mathbf{x}, \mathbf{v}) = \mathbf{0}$. We now use the Newton–Raphson method to solve (10.18) or, equivalently, use Newton's method to minimize a function for which (10.18) represents the first-order condition that equates the gradient to zero. Hence, given an iterate $(\mathbf{x}_k, \mathbf{v}_k)$, we solve the first-order approximation

$$\mathbf{W}(\mathbf{x}_k, \mathbf{v}_k) + \nabla \mathbf{W}(\mathbf{x}_k, \mathbf{v}_k)\begin{bmatrix} \mathbf{x} - \mathbf{x}_k \\ \mathbf{v} - \mathbf{v}_k \end{bmatrix} = \mathbf{0} \tag{10.19}$$

to the given system to determine the next iterate $(\mathbf{x}, \mathbf{v}) = (\mathbf{x}_{k+1}, \mathbf{v}_{k+1})$, where $\nabla \mathbf{W}$ denotes the Jacobian of $\mathbf{W}$. Defining $\nabla^2 L(\mathbf{x}_k) = \nabla^2 f(\mathbf{x}_k) + \sum_{i=1}^{l} v_{ki} \nabla^2 \mathbf{h}(\mathbf{x}_k)$ to be the usual Hessian of the Lagrangian at $\mathbf{x}_k$ with the Lagrange multiplier vector $\mathbf{v}_k$, and letting $\nabla \mathbf{h}$ denote the Jacobian of $\mathbf{h}$ comprised of rows $\nabla h_i(\mathbf{x})^t$ for $i = 1, \ldots, l$, we have

$$\nabla \mathbf{W}(\mathbf{x}_k, \mathbf{v}_k) = \begin{bmatrix} \nabla^2 L(\mathbf{x}_k) & \nabla \mathbf{h}(\mathbf{x}_k)^t \\ \nabla \mathbf{h}(\mathbf{x}_k) & \mathbf{0} \end{bmatrix} \tag{10.20}$$

Using (10.18) and (10.20), we can rewrite (10.19) as

$$\begin{aligned} \nabla^2 L(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k) + \nabla \mathbf{h}(\mathbf{x}_k)^t(\mathbf{v} - \mathbf{v}_k) &= -\nabla f(\mathbf{x}_k) - \nabla \mathbf{h}(\mathbf{x}_k)^t \mathbf{v}_k \\ \nabla \mathbf{h}(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k) &= -\mathbf{h}(\mathbf{x}_k) \end{aligned}$$

Substituting $\mathbf{d} = \mathbf{x} - \mathbf{x}_k$, this in turn can be rewritten as

$$\begin{aligned} \nabla^2 L(\mathbf{x}_k)\mathbf{d} + \nabla \mathbf{h}(\mathbf{x}_k)^t \mathbf{v} &= -\nabla f(\mathbf{x}_k) \\ \nabla \mathbf{h}(\mathbf{x}_k)\mathbf{d} &= -\mathbf{h}(\mathbf{x}_k) \end{aligned} \tag{10.21}$$

We can now solve for $(\mathbf{d}, \mathbf{v}) = (\mathbf{d}_k, \mathbf{v}_{k+1})$, say, using this system, if a solution exists. (See the convergence analysis below and Exercise 10.24.) Setting $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k$, we then increment k by 1 and repeat this process until $\mathbf{d} = \mathbf{0}$ happens to solve (10.21). When this occurs, if at all, noting (10.18), we shall have found a KKT solution for Problem P.

Now, instead of adopting the foregoing process to find *any* KKT solution for P, we can instead employ a quadratic minimization subproblem whose optimality conditions duplicate (10.21), but which might tend to drive the process toward beneficial KKT solutions. Such a quadratic program is stated below, where the constant term $f(\mathbf{x}_k)$ has been inserted into the objective function for insight and convenience.

$$\begin{aligned} \boldsymbol{QP}(\mathbf{x}_k, \mathbf{v}_k)\text{:} \quad \text{Minimize} \quad & f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^t \mathbf{d} + \tfrac{1}{2}\mathbf{d}^t \nabla^2 L(\mathbf{x}_k)\mathbf{d} \\ \text{subject to} \quad & h_i(\mathbf{x}_k) + \nabla h_i(\mathbf{x}_k)^t \mathbf{d} = 0, \qquad i = 1 \ldots, l \end{aligned} \tag{10.22}$$

Several comments regarding the linearly constrained quadratic subproblem QP($\mathbf{x}_k$, $\mathbf{v}_k$), abbreviated QP wherever unambiguous, are in order at this point. First, note that an optimum to QP, if it exists, is a KKT point for QP and satisfies equations (10.21), where $\mathbf{v}$ is the set of Lagrange multipliers associated with the constraints of QP. However, the minimization process of QP drives the solution toward a desirable KKT point satisfying (10.21) whenever alternatives exist. Second, observe that by the foregoing derivation, the objective function of QP represents not just a quadratic approximation for

$f(\mathbf{x})$ but also incorporates an additional term $\frac{1}{2}\sum_{i=1}^{l} v_{ki}\mathbf{d}^t\nabla^2\mathbf{h}(\mathbf{x}_k)\mathbf{d}$ to represent the curvature of the constraints. In fact, defining the Lagrangian function $L(\mathbf{x}) = f(\mathbf{x}) + \sum_{i=1}^{l} v_{ki}h_i(\mathbf{x})$, the objective function of QP($\mathbf{x}_k$, $\mathbf{v}_k$) can alternatively be written as follows, noting the constraints:

$$\text{Minimize} \quad L(\mathbf{x}_k) + \nabla_x L(\mathbf{x}_k)^t\mathbf{d} + \frac{1}{2}\mathbf{d}^t\nabla^2 L(\mathbf{x}_k)\mathbf{d} \tag{10.23}$$

Observe that (10.23) represents a second-order Taylor series approximation for the Lagrangian function L. In particular, this supports the quadratic convergence rate behavior in the presence of nonlinear constraints (see also Exercise 10.25). Third, note that the constraints of QP represent a first-order linearization at the current iterate $\mathbf{x}_k$. Fourth, observe that QP might be unbounded or infeasible, whereas P is not. Although the first of these unfavorable events can be managed by bounding the variation in $\mathbf{d}$, for instance, the second is more disconcerting. For example, if we have a constraint $x_1^2 + x_2^2 = 1$ and we linearize this at the origin, we obtain an inconsistent restriction requiring $-1 = 0$. We later present a variant of the above scheme which overcomes this difficulty (see also Exercise 10.23). Notwithstanding this problem, and assuming a well-behaved QP subproblem, we are now ready to state a rudimentary SQP algorithm.

Rudimentary SQP Algorithm (RSQP)

Initialization Put the iteration counter $k = 1$ and select a (suitable) starting primal–dual solution ($\mathbf{x}_k$, $\mathbf{v}_k$).

Main Step Solve the quadratic subproblem QP($\mathbf{x}_k$, $\mathbf{v}_k$) to obtain a solution $\mathbf{d}_k$ along with a set of Lagrange multipliers $\mathbf{v}_{k+1}$. If $\mathbf{d}_k = \mathbf{0}$, then, from (10.21), ($\mathbf{x}_k$, $\mathbf{v}_{k+1}$) satisfies the KKT conditions (10.18) for Problem P; stop. Otherwise, put $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k$, increment k by 1, and repeat the main step.

Convergence Rate Analysis

Under appropriate conditions, we can argue a quadratic convergence behavior for the foregoing algorithm. Specifically, suppose that $\bar{\mathbf{x}}$ is a regular KKT solution for Problem P which, together with a set of Lagrange multipliers $\bar{\mathbf{v}}$, satisfies the second-order sufficiency conditions of Theorem 4.4.2. Then, $\nabla\mathbf{W}(\bar{\mathbf{x}}, \bar{\mathbf{v}}) \equiv \overline{\nabla\mathbf{W}}$, say, defined by (10.20) is nonsingular. To see this, let us show that the system

$$\nabla\mathbf{W}(\bar{\mathbf{x}}, \bar{\mathbf{v}})\begin{bmatrix}\mathbf{d}_1\\ \mathbf{d}_2\end{bmatrix} = \mathbf{0}$$

has a unique solution given by $(\mathbf{d}_1^t, \mathbf{d}_2^t) = \mathbf{0}$. Consider any solution $(\mathbf{d}_1^t, \mathbf{d}_2^t)$. Since $\bar{\mathbf{x}}$ is a regular solution, $\nabla\mathbf{h}(\bar{\mathbf{x}})^t$ has full column rank; and so, if $\mathbf{d}_1 = \mathbf{0}$, then $\mathbf{d}_2 = \mathbf{0}$ as well. If $\mathbf{d}_1 \neq \mathbf{0}$, then, since $\nabla\mathbf{h}(\bar{\mathbf{x}})\mathbf{d}_1 = \mathbf{0}$, by the second-order sufficiency conditions we have $\mathbf{d}_1^t\nabla^2 L(\bar{\mathbf{x}})\mathbf{d}_1 > 0$. However, since $\nabla^2 L(\bar{\mathbf{x}})\mathbf{d}_1 + \underline{\nabla\mathbf{h}(\bar{\mathbf{x}})\mathbf{d}_2} = \mathbf{0}$, we have $\mathbf{d}_1^t\nabla^2 L(\bar{\mathbf{x}})\mathbf{d}_1 = -\mathbf{d}_2^t\nabla\mathbf{h}(\bar{\mathbf{x}})\mathbf{d}_1 = 0$, a contradiction. Hence, $\overline{\nabla\mathbf{W}}$ is nonsingular; and thus, for ($\mathbf{x}_k$, $\mathbf{v}_k$) sufficienty close to ($\bar{\mathbf{x}}$, $\bar{\mathbf{v}}$), $\nabla\mathbf{W}(\mathbf{x}_k, \mathbf{v}_k)$ is nonsingular. Therefore, the system (10.21), and so Problem QP($\mathbf{x}_k$, $\mathbf{v}_k$), has a well-defined (unique) solution. Consequently, in the spirit of Theorem 8.6.5, when ($\mathbf{x}_k$, $\mathbf{v}_k$) is sufficiently close to ($\bar{\mathbf{x}}$, $\bar{\mathbf{v}}$), a quadratic rate of convergence to ($\bar{\mathbf{x}}$, $\bar{\mathbf{v}}$) is obtained.

Actually, the closeness of $\mathbf{x}_k$ alone to $\bar{\mathbf{x}}$ is sufficient to establish convergence. It can be shown (see the Notes and References section) that, if $\mathbf{x}_1$ is sufficiently close to $\bar{\mathbf{x}}$ and if $\nabla \mathbf{W}(\mathbf{x}_1, \mathbf{v}_1)$ is nonsingular, then the algorithm SQPR converges quadratically to $(\bar{\mathbf{x}}, \bar{\mathbf{v}})$. In this respect, the Lagrange multipliers $\mathbf{v}$, appearing only in the second-order term in QP, do not play as important a role as they do in augmented Lagrangian (ALAG) penalty methods, for example, and inaccuracies in their estimation can be more flexibly tolerated.

Extension to Include Inequality Constraints

We now consider the inclusion of inequality constraints $g_i(\mathbf{x}) \leq 0$, $i = 1, \ldots, m$, in Problem P, where g_i are continuously twice-differentiable for $i = 1, \ldots, m$. This revised problem is restated below:

$$P: \quad \begin{array}{lll} \text{Minimize} & f(\mathbf{x}) & \\ \text{subject to} & g_i(\mathbf{x}) \leq 0 & i = 1, \ldots, m \\ & h_i(\mathbf{x}) = 0 & i = 1, \ldots, l \end{array} \tag{10.24}$$

For this instance, given an iterate $(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k)$, where $\mathbf{u}_k \geq \mathbf{0}$ and $\mathbf{v}_k$ are, respectively, the Lagrange multiplier estimates for the inequality and the equality constraints, we consider the following quadratic programming subproblem as a direct extension of (10.22):

$$QP(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k): \quad \begin{array}{lll} \text{Minimize} & f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^t\mathbf{d} + \frac{1}{2}\mathbf{d}^t\nabla^2 L(\mathbf{x}_k)\mathbf{d} & \\ \text{subject to} & g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^t\mathbf{d} \leq 0 & i = 1, \ldots, m \\ & h_i(\mathbf{x}_k) + \nabla h_i(\mathbf{x}_k)^t\mathbf{d} = 0 & i = 1, \ldots, l \end{array} \tag{10.25}$$

where $\nabla^2 L(\mathbf{x}_k) = \nabla^2 f(\mathbf{x}_k) + \sum_{i=1}^m u_{ki}\nabla^2 g_i(\mathbf{x}_k) + \sum_{i=1}^l v_{ki}\nabla^2 h_i(\mathbf{x}_k)$. Note that the KKT conditions for this problem require that, in addition to primal feasibility, we find Lagrange multipliers $\mathbf{u}$ and $\mathbf{v}$ such that

$$\nabla f(\mathbf{x}_k) + \nabla^2 L(\mathbf{x}_k)\mathbf{d} + \sum_{i=1}^m u_i \nabla g_i(\mathbf{x}_k) + \sum_{i=1}^l v_i \nabla h_i(\mathbf{x}_k) = \mathbf{0} \tag{10.26a}$$

$$u_i[g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^t\mathbf{d}] = 0 \qquad i = 1, \ldots, m \tag{10.26b}$$

$$\mathbf{u} \geq \mathbf{0} \qquad \mathbf{v} \text{ unrestricted} \tag{10.26c}$$

Hence, if $\mathbf{d}_k$ solves QP$(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k)$ with Lagrange multipliers $\mathbf{u}_{k+1}$ and $\mathbf{v}_{k+1}$, and if $\mathbf{d}_k = \mathbf{0}$, then $\mathbf{x}_k$ along with $(\mathbf{u}_{k+1}, \mathbf{v}_{k+1})$ yields a KKT solution for the original Problem P. Otherwise, we set $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k$ as before, increment k by 1, and repeat the process. In a likewise manner, it can be shown that if $\bar{\mathbf{x}}$ is a regular KKT solution which, together with $(\bar{\mathbf{u}}, \bar{\mathbf{v}})$, satisfies the second-order sufficiency conditions, and if $(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k)$ is initialized sufficiently close to $(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})$, then the foregoing iterative process will converge quadratically to $(\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\mathbf{v}})$.

Quasi-Newton Approximations

One disadvantage of the SQP method discussed thus far is that we require second-order derivatives to be calculated and, besides, that $\nabla^2 L(\mathbf{x}_k)$ might not be positive definite. This can be overcome by employing quasi-Newton positive definite approximations for $\nabla^2 L$. For example, given a positive definite approximation $\mathbf{B}_k$ for $\nabla^2 L(\mathbf{x}_k)$ in the algorithm SQPR described above, we can solve the system (10.21) with $\nabla^2 L(\mathbf{x}_k)$ replaced by $\mathbf{B}_k$, to obtain the unique solution $\mathbf{d}_k$ and $\mathbf{v}_{k+1}$, and then set $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k$. This is equivalent to the iterative step given by

$$\begin{bmatrix} \mathbf{x}_{k+1} \\ \mathbf{v}_{k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_k \\ \mathbf{v}_k \end{bmatrix} - \begin{bmatrix} \mathbf{B}_k & \nabla \mathbf{h}(\mathbf{x}_k)^t \\ \nabla \mathbf{h}(\mathbf{x}_k) & \mathbf{0} \end{bmatrix}^{-1} \begin{bmatrix} \nabla L(\mathbf{x}_k) \\ \mathbf{h}(\mathbf{x}_k) \end{bmatrix}$$

where $\nabla L(\mathbf{x}_k) = \nabla f(\mathbf{x}_k) + \nabla \mathbf{h}(\mathbf{x}_k)^t \mathbf{v}_k$. Then, adopting the popular BFGS update for the Hessian as defined by (8.63), we can compute

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \frac{\mathbf{q}_k \mathbf{q}_k^t}{\mathbf{q}_k^t \mathbf{p}_k} - \frac{\mathbf{B}_k \mathbf{p}_k \mathbf{p}_k^t \mathbf{B}_k}{\mathbf{p}_k^t \mathbf{B}_k \mathbf{p}_k} \tag{10.27a}$$

where

$$\mathbf{p}_k = \mathbf{x}_{k+1} - \mathbf{x}_k \qquad \mathbf{q}_k = \nabla L'(\mathbf{x}_{k+1}) - \nabla L'(\mathbf{x}_k)$$

and where

$$\nabla L'(\mathbf{x}) \equiv \nabla f(\mathbf{x}) + \sum_{i=1}^{l} v_{(k+1)i} \nabla h_i(\mathbf{x}) \tag{10.27b}$$

It can be shown that this modification in the rudimentary process, similar to the quasi-Newton modification of Newton's algorithm, converges superlinearly when initialized sufficiently close to a solution $(\bar{\mathbf{x}}, \bar{\mathbf{v}})$ that satisfies the foregoing regularity and second-order sufficiency conditions. However, this superlinear convergence rate is strongly based on the use of unit step sizes.

A Globally Convergent Variant Using the l_1 Penalty as a Merit Function

A principal disadvantage of the SQP method described thus far is that convergence is guaranteed only when the algorithm is initialized sufficiently close to a desirable solution whereas, in practice, this condition is usually difficult to realize. To remedy this situation and to ensure global convergence, we introduce the idea of a *merit function*. This is a function that, along with the objective function, is simultaneously minimized at the solution of the problem, but one that also serves as a descent function, guiding the iterates and providing a measure of progress. Preferably, it should be easy to evaluate this function, and it should not impair the convergence rate of the algorithm. We describe the use of the popular l_1, or absolute value, penalty function (9.8), restated below, as a merit function for Problem P given in (10.24):

$$F_E(\mathbf{x}) = f(\mathbf{x}) + \mu\left[\sum_{i=1}^{m} \max\{0, g_i(\mathbf{x})\} + \sum_{i=1}^{l} |h_i(\mathbf{x})|\right] \tag{10.28}$$

The following lemma establishes the role of F_E as a merit function. The Notes and References section points out other quadratic and ALAG penalty functions that can be used as merit functions in a similar context.

10.4.1 Lemma

Given an iterate $\mathbf{x}_k$, consider the quadratic subproblem QP given by (10.25), where $\nabla^2 L(\mathbf{x}_k)$ is replaced by any positive definite approximation $\mathbf{B}_k$. Let $\mathbf{d}$ solve this problem with Lagrange multipliers $\mathbf{u}$ and $\mathbf{v}$ associated with the inequality and the equality constraints, respectively. If $\mathbf{d} \neq \mathbf{0}$, and if $\mu \geq \max\{u_1, \ldots, u_m, |v_1|, \ldots, |v_l|\}$, then $\mathbf{d}$ is a descent direction at $\mathbf{x} = \mathbf{x}_k$ for the l_1 penalty function F_E given by (10.28).

Proof

Using the primal feasibility, the dual feasibility, and the complementary slackness conditions (10.25), (10.26a), and (10.26b) for QP, we have

$$\begin{aligned}\nabla f(\mathbf{x}_k)^t\mathbf{d} &= -\mathbf{d}^t\mathbf{B}_k\mathbf{d} - \sum_{i=1}^{m} u_i\nabla g_i(\mathbf{x}_k)^t\mathbf{d} - \sum_{i=1}^{l} v_i\nabla h_i(\mathbf{x}_k)^t\mathbf{d} \\ &= -\mathbf{d}^t\mathbf{B}_k\mathbf{d} + \sum_{i=1}^{m} u_i g_i(\mathbf{x}_k) + \sum_{i=1}^{l} v_i h_i(\mathbf{x}_k) \\ &\leq -\mathbf{d}^t\mathbf{B}_k\mathbf{d} + \sum_{i=1}^{m} u_i \max\{0, g_i(\mathbf{x}_k)\} + \sum_{i=1}^{l} |v_i|\,|h_i(\mathbf{x}_k)| \\ &\leq -\mathbf{d}^t\mathbf{B}_k\mathbf{d} + \mu\left[\sum_{i=1}^{m} \max\{0, g_i(\mathbf{x}_k)\} + \sum_{i=1}^{l} |h_i(\mathbf{x}_k)|\right]\end{aligned} \tag{10.29}$$

Now we have, from (10.28), that for a step length $\lambda \geq 0$,

$$\begin{aligned}F_E(\mathbf{x}_k) - F_E(\mathbf{x}_k + \lambda\mathbf{d}) &= [f(\mathbf{x}_k) - f(\mathbf{x}_k + \lambda\mathbf{d})] \\ &+ \mu\left\{\sum_{i=1}^{m} [\max\{0, g_i(\mathbf{x}_k)\} - \max\{0, g_i(\mathbf{x}_k + \lambda\mathbf{d})\}]\right. \\ &\left. + \sum_{i=1}^{l} [|h_i(\mathbf{x}_k)| - |h_i(\mathbf{x}_k + \lambda\mathbf{d})|]\right\}\end{aligned} \tag{10.30}$$

Letting $O_i(\lambda)$ denote an appropriate function that approaches zero as $\lambda \to 0$, for $i = 0, 1, \ldots, m + l$, we have, for $\lambda > 0$ and sufficiently small,

$$f(\mathbf{x}_k + \lambda\mathbf{d}) = f(\mathbf{x}_k) + \lambda\nabla f(\mathbf{x}_k)^t\mathbf{d} + \lambda O_0(\lambda) \tag{10.31a}$$

Also, $g_i(\mathbf{x}_k + \lambda\mathbf{d}) = g_i(\mathbf{x}_k) + \lambda\nabla g_i(\mathbf{x}_k)^t\mathbf{d} + \lambda O_i(\lambda) \leq g_i(\mathbf{x}_k) - \lambda g_i(\mathbf{x}_k) + \lambda O_i(\lambda)$ from (10.25). Hence,

$$\max\{0, g_i(\mathbf{x}_k + \lambda\mathbf{d})\} \leq (1 - \lambda)\max\{0, g_i(\mathbf{x}_k)\} + \lambda|O_i(\lambda)| \tag{10.31b}$$

Similarly, from (10.25),

$$h_i(\mathbf{x}_k + \lambda\mathbf{d}) = h_i(\mathbf{x}_k) + \lambda\nabla h_i(\mathbf{x}_k) + \lambda O_{m+i}(\lambda) = (1 - \lambda)h_i(\mathbf{x}_k) + \lambda O_{m+i}(\lambda)$$

and hence

$$|h_i(\mathbf{x}_k + \lambda\mathbf{d})| \leq (1 - \lambda)|h_i(\mathbf{x}_k)| + \lambda|O_{m+i}(\lambda)| \tag{10.31c}$$

Using (10.31) in (10.30), we obtain, for $\lambda \geq 0$ and sufficiently small, $F_E(\mathbf{x}_k) - F_E(\mathbf{x}_k + \lambda\mathbf{d}) \geq \lambda[-\nabla f(\mathbf{x}_k)^t\mathbf{d} + \mu\{\sum_{i=1}^{m} \max\{0, g_i(\mathbf{x}_k)\} + \sum_{i=1}^{l} |h_i(\mathbf{x}_k)| + O(\lambda)]$, where $O(\lambda) \to 0$ as $\lambda \to 0$. Hence, by (10.29), this gives $F_E(\mathbf{x}_k) - F_E(\mathbf{x}_k + \lambda\mathbf{d}) \geq \lambda[\mathbf{d}^t\mathbf{B}_k\mathbf{d} + O(\lambda)] > 0$ for all $\lambda \in (0, \delta)$ for some $\delta > 0$ by the positive definiteness of $\mathbf{B}_k$, and this completes the proof.

Lemma 10.4.1 exhibits the flexibility in the choice of $\mathbf{B}_k$ for the resulting direction to be a descent direction for the exact penalty function. This matrix only needs to be positive definite, and may be updated by using any quasi-Newton strategy such as an extension of (10.27), or may even be held constant throughout the algorithm. This descent feature enables us to obtain a globally convergent algorithm under mild assumptions, as shown below.

Summary of the Merit Function SQP Algorithm (MSQP)

Initialization Put the iteration counter at $k = 1$ and select a (suitable) starting solution $\mathbf{x}_k$. Also, select a positive definite approximation $\mathbf{B}_k$ to the Hessian $\nabla^2 L(\mathbf{x}_k)$ defined with respect to some Lagrange multipliers $\mathbf{u}_k \geq \mathbf{0}$ and $\mathbf{v}_k$ associated with the inequality and the equality constraints, respectively, of Problem (10.24). (Note that $\mathbf{B}_k$ might be arbitrary and need not necessarily bear any relationship to $\nabla^2 L(\mathbf{x}_k)$.)

Main Step Solve the quadratic programming subproblem QP given by (10.25) with $\nabla^2 L(\mathbf{x}_k)$ replaced by $\mathbf{B}_k$ and obtain a solution $\mathbf{d}_k$ along with Lagrange multipliers $(\mathbf{u}_{k+1}, \mathbf{v}_{k+1})$. If $\mathbf{d}_k = \mathbf{0}$, then stop with $\mathbf{x}_k$ as a KKT solution for Problem P of (10.24), having Lagrange multipliers $(\mathbf{u}_{k+1}, \mathbf{v}_{k+1})$. Otherwise, find $\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{d}_k$, where λ_k minimizes $F_E(\mathbf{x}_k + \lambda \mathbf{d}_k)$ over $\lambda \in E_1$, $\lambda \geq 0$. Update $\mathbf{B}_k$ to a positive definite matrix $\mathbf{B}_{k+1}$ [which might be $\mathbf{B}_k$, itself, or $\nabla^2 L(\mathbf{x}_{k+1})$ defined with respect to $(\mathbf{u}_{k+1}, \mathbf{v}_{k+1})$, or some approximation thereof updated according to a quasi-Newton scheme]. Increment k by 1 and repeat the main step.

The reader may note that the line search above is to be performed with respect to a nondifferentiable function, which obviates the use of the techniques in Sections 8.2 and 8.3, including the popular curve fitting approaches. Below, we sketch the proof of convergence for algorithm MSQP. In Exercise 10.26, we ask the reader to provide a detailed argument.

10.4.2 Theorem

Algorithm MSQP either terminates finitely with a KKT solution to Problem P defined in (10.24), or else an infinite sequence of iterates $\{\mathbf{x}_k\}$ is generated. In the latter case, assume that $\{\mathbf{x}_k\} \subseteq X$, a compact subset of E_n, and that for any point $\mathbf{x} \in X$ and any positive definite matrix $\mathbf{B}$, the quadratic programming subproblem QP (with $\nabla^2 L$ replaced by $\mathbf{B}$) has a unique solution $\mathbf{d}$ (and so this problem is feasible), and has unique Lagrange multipliers $\mathbf{u}$ and $\mathbf{v}$ satisfying $\mu \geq \max\{u_1, \ldots, u_m, |v_1|, \ldots, |v_l|\}$, where μ is the penalty parameter for F_E defined in (10.28). Furthermore, assume that the accompanying sequence $\{\mathbf{B}_k\}$ of positive definite matrices generated lies in a compact subspace, with all accumulation points being positive definite (or with $\{\mathbf{B}_k^{-1}\}$ also being bounded). Then, every accumulation point of $\{\mathbf{x}_k\}$ is a KKT solution for P.

Proof

Let the solution set Ω be composed of all points $\mathbf{x}$ such that the corresponding subproblem QP produces $\mathbf{d} = \mathbf{0}$ at optimality. Note from (10.26) that, given any positive definite matrix $\mathbf{B}$, $\mathbf{x}$ is a KKT solution for P if and only if $\mathbf{d} = \mathbf{0}$ is optimal for QP, that is, $\mathbf{x} \in \Omega$. Now the algorithm MSQP can be viewed as a map **UMD**, where **D** is the direction-finding map that determines the direction $\mathbf{d}_k$ via the subproblem QP defined with respect to $\mathbf{x}_k$ and $\mathbf{B}_k$, **M** is the usual line search map, and **U** is a map that updates $\mathbf{B}_k$ to $\mathbf{B}_{k+1}$. Since the optimality conditions of QP are continuous in the data, the output of QP can readily be seen to be a continuous function of the input. By Theorem 8.4.1, the line search map **M** is also closed, since F_E is continuous. Since the conditions of

Theorem 7.3.2 hold, **MD** is therefore closed. Moreover, by Lemma 10.4.1, if $\mathbf{x}_k \notin \Omega$, then $F_E(\mathbf{x}_{k+1}) < F_E(\mathbf{x}_k)$, thus providing a strict descent function. Since the map **U** does not disturb this descent feature; and since $\{\mathbf{x}_k\}$ and $\{\mathbf{B}_k\}$ are contained within compact sets, with any accumulation point of $\mathbf{B}_k$ being positive definite, the argument of Theorem 7.3.4 holds. This completes the proof.

10.4.3 Example

To illustrate algorithms RSQP and MSQP, consider the following problem:

$$\begin{aligned} \text{Minimize} \quad & 2x_1^2 + 2x_2^2 - 2x_1x_2 - 4x_1 - 6x_2 \\ \text{subject to} \quad & g_1(\mathbf{x}) = 2x_1^2 - x_2 \le 0 \\ & g_2(\mathbf{x}) = x_1 + 5x_2 - 5 \le 0 \\ & g_3(\mathbf{x}) = -x_1 \le 0 \\ & g_4(\mathbf{x}) = -x_2 \le 0 \end{aligned}$$

A graphical solution of this problem appears in Figure 10.13*a*. Following Example 10.3.2, let us use $\mu = 10$ in the l_1 penalty merit function F_E defined by (10.28). Let us also use $B_k = \nabla^2 L(\mathbf{x}_k)$ itself, and commence with $\mathbf{x}_1 = (0, 1)^t$ and with Lagrange multipliers $\mathbf{u}_1 = (0, 0, 0, 0)^t$. Hence, we have $f(\mathbf{x}_1) = -4 = F_E(\mathbf{x}_1)$, since $\mathbf{x}_1$ happens to be feasible. Also, $g_1(\mathbf{x}_1) = -1$, $g_2(\mathbf{x}_1) = 0$, $g_3(\mathbf{x}_1) = 0$, and $g_4(\mathbf{x}_1) = -1$. The function gradients are $\nabla f(\mathbf{x}_1) = (-6, -2)^t$, $\nabla g_1(\mathbf{x}_1) = (0, -1)^t$, $\nabla g_2(\mathbf{x}_1) = (1, 5)^t$, $\nabla g_3(\mathbf{x}_1) = (-1, 0)^t$, and $\nabla g_4(\mathbf{x}_1) = (0, -1)^t$. The Hessian of the Lagrangian is

$$\nabla^2 L(\mathbf{x}_1) = \nabla^2 f(\mathbf{x}_1) = \begin{bmatrix} 4 & -2 \\ -2 & 4 \end{bmatrix}$$

Accordingly, the quadratic programming subproblem QP defined in (10.25) is as follows:

$$\begin{aligned} \textbf{\textit{QP:}} \quad \text{Minimize} \quad & -6d_1 - 2d_2 + \tfrac{1}{2}[4d_1^2 + 4d_2^2 - 4d_1d_2] \\ \text{subject to} \quad & -1 - d_2 \le 0, \qquad d_1 + 5d_2 \le 0, \\ & -d_1 \le 0, \qquad\quad -1 - d_2 \le 0 \end{aligned}$$

Figure 10.14 depicts the graphical solution of this problem. At optimality, only the second constraint of QP is binding. Hence, the KKT system gives

$$4d_1 - 2d_2 - 6 + u_2 = 0 \qquad 4d_2 - 2d_1 - 2 + 5u_2 = 0 \qquad d_1 + 5d_2 = 0$$

Solving, we obtain $\mathbf{d}_1 = (\frac{35}{31}, -\frac{7}{31})^t$ and $\mathbf{u}_2 = (0, 1.032258, 0, 0)$ as the primal and dual optimal solutions, respectively, to QP.

Now for algorithm RSQP, we would take a unit step-size to obtain $\mathbf{x}_2 = \mathbf{x}_1 + \mathbf{d}_1 = (1.1290322, 0.7741936)^t$. This completes one iteration. We ask the reader in Exercise 10.27 to continue this process and to examine its convergence behavior.

On the other hand, for algorithm MSQP, we need to perform a line search, minimizing F_E from $\mathbf{x}_1$ along the direction $\mathbf{d}_1$. This line search problem is, from (10.28),

$$\begin{aligned} \underset{\lambda \ge 0}{\text{Minimize}} \quad & F_E(\mathbf{x}_1 + \lambda \mathbf{d}_1) \\ & = [3.1612897\lambda^2 - 6.3225804\lambda - 4] \\ & + 10[\max\{0, 2.5494274\lambda^2 + 0.2258064\lambda - 1\} \\ & + \max\{0, 0\} + \max\{0, -1.1290322\lambda\} \\ & + \max\{0, -1 + 0.2258064\lambda\}] \end{aligned}$$

Using the Golden Section Method, for example, we find the step length $\lambda_1 = 0.5835726$.

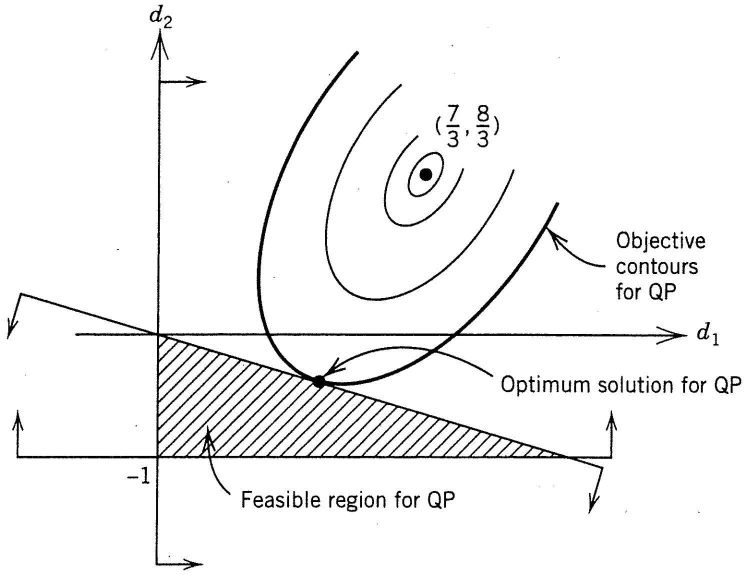

Figure 10.14 Solution of Subproblem QP.

(Note that the unconstrained minimum of $f(\mathbf{x}_1 + \lambda \mathbf{d}_1)$ occurs at $\lambda = 1$; but beyond $\lambda = \lambda_1$, the first max $\{0, \cdot\}$ term starts to become positive and increases the value of F_E, hence, giving λ_1 as the desired step size.) This produces the new iterate $\mathbf{x}_2 = \mathbf{x}_1 + \lambda_1 \mathbf{d}_1 = (0.6588722, 0.8682256)^t$. Observe that, because the generated direction $\mathbf{d}_1$ happened to be leading toward the optimum for P, the minimization of the exact l_1 penalty function (with μ sufficiently large) produced this optimum. We ask the reader in Exercise 10.27 to verify the optimality of $\mathbf{x}_2$ via the corresponding quadratic programming subproblem.

The Maratos Effect

Consider the equality constrained Problem P defined in (10.17). (A similar phenomenon holds for Problem (10.24).) Note that the rudimentary SQP algorithm adopts a unit step size and converges quadratically when $(\mathbf{x}_k, \mathbf{v}_k)$ is initialized close to a regular solution $(\bar{\mathbf{x}}, \bar{\mathbf{v}})$ satisfying the second-order sufficiency conditions. The merit function-based algorithm, however, performs a line search at each iteration to minimize the exact penalty function F_E of (10.28), given that the conditions of Lemma 10.4.1 hold true. Assuming all of the foregoing conditions, one might think that when $(\mathbf{x}_k, \mathbf{v}_k)$ is sufficiently close to $(\bar{\mathbf{x}}, \bar{\mathbf{v}})$, a unit step size would decrease the value of F_E. This statement is incorrect, and its violation is known as the *Maratos effect,* after N. Maratos, who discovered this in relation to Powell's algorithm in 1978.

10.4.4. Example (Maratos Effect)

Consider the following example discussed in Powell (1986).

$$\begin{aligned} &\text{Minimize} && f(\mathbf{x}) = -x_1 + 2(x_1^2 + x_2^2 - 1) \\ &\text{subject to} && h(\mathbf{x}) = x_1^2 + x_2^2 - 1 = 0 \end{aligned}$$

Clearly, the optimum occurs at $\bar{\mathbf{x}} = (1, 0)^t$. The Lagrange multiplier at this solution is readily obtained from the KKT conditions to be $\bar{v} = -\frac{3}{2}$, and, so, $\nabla^2 L(\bar{\mathbf{x}}) =$

$\nabla^2 f(\bar{\mathbf{x}}) + \bar{v}\nabla^2 h(\bar{\mathbf{x}}) = \mathbf{I}$. Let us take the approximations $\mathbf{B}_k$ to be equal to $\mathbf{I}$ throughout the algorithm.

Now let us select $\mathbf{x}_k$ to be sufficiently close to $\bar{\mathbf{x}}$, but lying on the unit ball defining the constraint. Hence, we can let $\mathbf{x}_k = (\cos\theta, \sin\theta)^t$, where $|\theta|$ is small. The quadratic program (10.22) is given by

$$\text{Minimize} \quad f(\mathbf{x}_k) + (-1 + 4\cos\theta)d_1 + (4\sin\theta)d_2 + \tfrac{1}{2}[d_1^2 + d_2^2]$$
$$\text{subject to} \quad 2\cos\theta d_1 + 2\sin\theta d_2 = 0\}$$
$$\equiv \text{Minimize} \quad \{ f(\mathbf{x}_k) - d_1 + \tfrac{1}{2}(d_1^2 + d_2^2) : \cos\theta d_1 + \sin\theta d_2 = 0\}$$

Writing the KKT conditions for this problem and solving, we readily obtain the optimal solution $\mathbf{d}_k = (\sin^2\theta, \ -\sin\theta\cos\theta)^t$. Hence, $\mathbf{x}_{k+1} = (\mathbf{x}_k + \mathbf{d}_k) = (\cos\theta + \sin^2\theta, \sin\theta - \sin\theta\cos\theta)^t$. Note that $\|\mathbf{x}_k - \mathbf{x}\|^2 = \sqrt{2(1-\cos\theta)} \simeq \theta$, adopting a second-order Taylor series approximation while, similarly, $\|(\mathbf{x}_k + \mathbf{d}_k) - \bar{\mathbf{x}}\| \simeq \theta^2/2$, thereby attesting to the rapid convergence behavior. However, it is readily verified that $f(\mathbf{x}_k + \mathbf{d}_k) = -\cos\theta + \sin^2\theta$ while $f(\mathbf{x}_k) = -\cos\theta$ and, also, that $h(\mathbf{x}_k + \mathbf{d}_k) = 2\sin^2\theta$ while $h(\mathbf{x}_k) = 0$. Hence, although a unit step makes $\|\mathbf{x}_k + \mathbf{d}_k - \bar{\mathbf{x}}\|$ considerably smaller than $\|\mathbf{x}_k - \bar{\mathbf{x}}\|$, it results in an increase in both f and in the constraint violation, and therefore would increase the value of F_E for any $\mu \geq 0$ or, for that matter, it would increase the value of any merit function.

Several suggestions have been proposed for overcoming the Maratos effect based on tolerating an increase in both f and the constraint violations, or recalculating the step length after correcting for second-order effects, or altering the search direction via modifications in second-order approximations to the objective and the constraint functions. We direct the reader to the Notes and References section for further reading on this subject.

Using the l_1 Penalty in the QP Subproblem—The L_1SQP Approach

In Section 10.3, we presented a superior penalty-based SLP algorithm that adopts trust region concepts and affords a robust and efficient scheme. A similar procedure has been proposed by Fletcher [1981] in the SQP framework, which exhibits a relatively superior computational behavior. Here, given an iterate $\mathbf{x}_k$ and a positive definite approximation $\mathbf{B}_k$ to the Hessian of the Lagrangian function, analogous to (10.11a), this procedure solves the following quadratic subproblem:

$$\begin{aligned} QP: \quad \text{Minimize} \quad & [f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^t\mathbf{d} + \tfrac{1}{2}\mathbf{d}^t\mathbf{B}_k\mathbf{d}] \\ & + \mu\left[\sum_{i=1}^{m} \max\{0, g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^t\mathbf{d}\}\right. \\ & + \left.\sum_{i=1}^{l} |h_i(\mathbf{x}_k) + \nabla h_i(\mathbf{x}_k)^t\mathbf{d}|\right] \\ \text{subject to} \quad & -\mathbf{\Delta}_k \leq \mathbf{d} \leq \mathbf{\Delta}_k \end{aligned} \tag{10.32}$$

where $\mathbf{\Delta}_k$ is a trust region step bound and, as before, μ is a suitably large penalty parameter. Note that in comparison with Problem (10.25), the constraints have been accommodated into the objective function via an l_1 penalty term and have been replaced by a trust region constraint. Hence, the subproblem QP is always feasible and bounded and has an optimum. To contend with the nondifferentiability of the objective function, the l_1 terms can be retransferred into the constraints as in (10.11b). Similar to the PSLP algorithm, if $\mathbf{d}_k$ solves this problem along with Lagrange multiplier estimates $(\mathbf{u}_{k+1}, \mathbf{v}_{k+1})$, and if $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{d}_k$ is ε-feasible and satisfies the KKT conditions within a tolerance, or if the fractional improvement in the original objective function is not better than a given tolerance over some c consecutive iterations, the algorithm can be terminated. Otherwise, the process is iteratively repeated. This type of procedure enjoys the asymptotic local convergence properties of SQP methods but also achieves global convergence owing to the l_1 penalty function and the trust region features. However, it is also prone to the Maratos effect, and corrective measures are necessary to avoid this phenomenon. The reader should refer to the Notes and References section for further reading on this topic.

10.5 The Gradient Projection Method of Rosen

As we learned in Chapter 8, the direction of steepest descent is that of the negative gradient. In the presence of constraints, however, moving along the steepest descent direction may lead to infeasible points. The gradient projection method of Rosen [1960] projects the negative gradient in such a way that improves the objective function and meanwhile maintains feasibility.

First, consider the following definition of a projection matrix.

10.5.1 Definition

An $n \times n$ matrix $\mathbf{P}$ is called a *projection matrix* if $\mathbf{P} = \mathbf{P}^t$ and $\mathbf{PP} = \mathbf{P}$.

10.5.2 Lemma

Let $\mathbf{P}$ be an $n \times n$ matrix. Then, the following statements are true:

1. If $\mathbf{P}$ is a projection matrix, then $\mathbf{P}$ is positive semidefinite.
2. $\mathbf{P}$ is a projection matrix if and only if $\mathbf{I} - \mathbf{P}$ is a projection matrix.
3. Let $\mathbf{P}$ be a projection matrix and let $\mathbf{Q} = \mathbf{I} - \mathbf{P}$. Then, $L = \{\mathbf{Px}:\mathbf{x} \in E_n\}$ and $\mathbf{L}^\perp = \{\mathbf{Qx}:\mathbf{x} \in E_n\}$ are orthogonal linear subspaces. Furthermore, any point $\mathbf{x} \in E_n$ can be represented uniquely as $\mathbf{p} + \mathbf{q}$, where $\mathbf{p} \in L$ and $\mathbf{q} \in L^\perp$.

Proof

Let $\mathbf{P}$ be a projection matrix, and let $\mathbf{x} \in E_n$ be arbitrary. Then, $\mathbf{x}^t\mathbf{Px} = \mathbf{x}^t\mathbf{PPx} = \mathbf{x}^t\mathbf{P}^t\mathbf{Px} = \|\mathbf{Px}\|^2 \geq 0$, and, hence, $\mathbf{P}$ is positive semidefinite, and part 1 follows.

By Definition 10.5.1, part 2 is obvious. Clearly L and $L^\perp$ are linear subspaces. Note that $\mathbf{P}^t\mathbf{Q} = \mathbf{P}(\mathbf{I} - \mathbf{P}) = \mathbf{P} - \mathbf{PP} = \mathbf{0}$, and hence, L and $L^\perp$ are indeed orthogonal. Now let $\mathbf{x}$ be an arbitrary point in E_n. Then, $\mathbf{x} = \mathbf{Ix} = (\mathbf{P} + \mathbf{Q})\mathbf{x} = \mathbf{Px} + \mathbf{Qx} = \mathbf{p} + \mathbf{q}$, where $\mathbf{p} \in L$ and $\mathbf{q} \in L^\perp$. To show uniqueness, suppose that $\mathbf{x}$ can also be represented as $\mathbf{x} = \mathbf{p}' + \mathbf{q}'$, where $\mathbf{p}' \in L$ and $\mathbf{q}' \in L^\perp$. By subtraction, it follows that $\mathbf{p} - \mathbf{p}' = \mathbf{q}' - \mathbf{q}$. Since $\mathbf{p} - \mathbf{p}' \in L$ and $\mathbf{q}' - \mathbf{q} \in L^\perp$, and since the only point in the intersection of L and $L^\perp$ is the zero vector, it follows that $\mathbf{p} - \mathbf{p}' = \mathbf{q}' - \mathbf{q} = \mathbf{0}$. Thus the representation of $\mathbf{x}$ is unique, and the proof is complete.