PROBLEM 2

OPTIMAL CONTROL OF A STEEL SLAB CASTER

1. INTRODUCTION

BHP Steel International (Slab and Plate Products Division) currently operates a continuous slab caster at Port Kembla, NSW. The company suspects that the present operating practice for this caster may not be optimal. In particular, a greater net production rate is thought to be achievable by better choices of casting speed and flow rates in the spray cooling system. As a first step towards investigating this possibility, the company has developed a simple one-dimensional heat transfer computer model of the caster. The problem posed to the 1985 MISG was twofold:

i. To outline an optimal design/control model, incorporating this existing heat transfer model, which could predict "optimal" settings for the casting speed, coolant flow rates etc.

ii. To indicate means of effectively solving the above model.

Three kinds of roles for an optimal design/control model can be identified:

a. optimal design of the steady state (set up) process,

b. optimal design of the changeover from one steady state process to another (e.g. a change in the grade of steel being cast),

c. optimal response to short term disturbances in the process. Such disturbances could be scheduled (e.g. regular tundish changes) or unscheduled

(e.g. system slow downs, cooling system blockages etc.).

Note that any analysis can be done off-line in the case of (a), (b) and scheduled instances of (c), whereas unscheduled disturbances are more or less unpredictable and so much of the analysis involved in (c) would need to be done in real time.

The Study Group decided to concentrate on (a) only, as this seemed the most basic application, and would provide a natural starting point for possible investigation of (b) and (c) at a later stage.

2. THE HEAT TRANSFER MODEL

The company has developed a one-dimensional slice model of the (steady state) heat transfer process in the caster. Only heat conduction through the thickness of the slab is considered. The neglect of conduction along the length of the slab is reasonable, given the relatively low longitudinal thermal gradient (15 Km^{-1}, compared to $3x10^3$ Km^{-1} through the half-thickness). Disregarding heat conduction across the width of the slab may be less realistic; however, for slabs of significant width/thickness ratio, the assumption may be justified, at least as a first approximation. For a billet caster though, a true two-dimensional slice model would probably be needed. Other physical inputs to the model include

i. a method of accounting for the liberation of latent heat of solidification,

ii. a heat transfer boundary condition on the faces of the slab, both in the mould and in the spray zones,

iii. temperature dependence of both specific heat and thermal conductivity.

To a large extent we shall treat the heat-transfer model as a "black box", and not enquire too much into its internal workings. It will be sufficient to take the following point of view:

Let x,z be coordinates measured through the thickness and along the length of the slab respectively. Let $0=x_1<..<x_k<..<x_{N_x} = T$ and $0=z_1<..<z_\ell<..<z_{N_z} = L$ be some points distributed along these axes. The output from the heat transfer model is a set of quantities $\theta_k^{[\ell]}$, $k=1,\ldots, N_x$, $\ell=1,..,N_z$, representing the temperature of the casting at the point $x=x_k$, $z=z_\ell$ (or equivalently, the temperature at $x=x_k$ of a typical slice as it passes $z=z_\ell$, (see Fig. 1). For notational convenience, let $\theta^{[\ell]}$ denote the vector with components $\theta_k^{[\ell]}$ and write θ for the matrix whose (k,ℓ) entry is $\theta_k^{[\ell]}$. From a purely formal point of view, the heat transfer model can be thought of as a system of algebraic equations to be solved for θ.

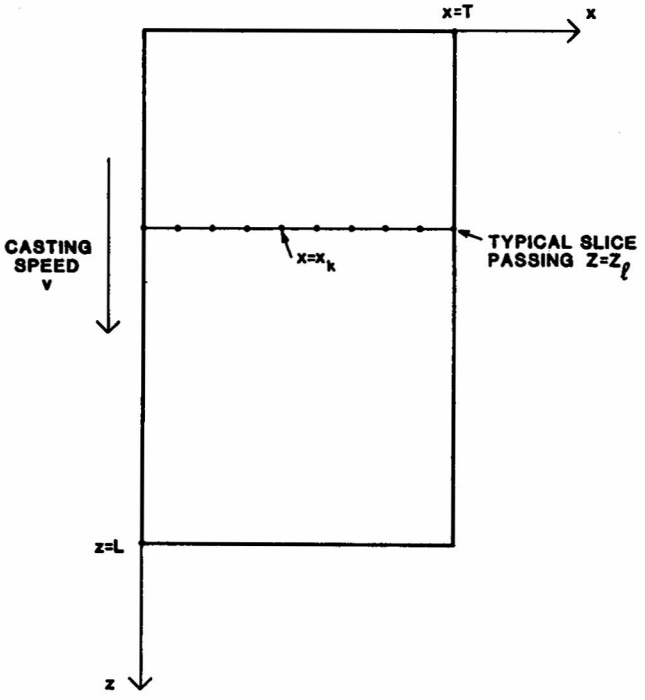


Figure 1. Definition sketch for (x_k, z_ℓ).

The solution of the optimal design model will require many calls to the heat transfer model. It is therefore important to have a computationally efficient means of solving the heat transfer model. In this respect an implicit timestepping method would almost certainly be superior to any explicit method, particularly as the heat transfer model has only one space dimension which will make the implicit system of equations relatively easy to solve at each time level.

3. MATHEMATICAL FORMULATION OF THE OPTIMAL DESIGN MODEL

3.1 Control parameters

The casting process is controlled by a number of parameters. These can be varied, within certain limits, by the operator. The optimal design problem is to determine the "best" settings of these parameters. The only parameters that shall be explicitly considered in this report are

v = casting speed

$u_1, u_2, .., u_m$ = flow rates in each of the m (independently controllable) spray cooling zones,

although additional parameters can easily be fitted into the framework that will be described. Typical constraints on these parameters could be

$$a_j \leq u_j \leq A_j, \qquad j=1,..,m \tag{3.1}$$

where a_j, A_j are the minimum and maximum possible flow rates in each of the spray zones. In addition, there may be a restriction on the total flow rate through all sprays

$$b \leq \sum_{j=1}^{m} u_j \leq B. \tag{3.2}$$

Although the size of m is usually only of the order of 10 or so, it may still be desirable in the interests of simplicity to reduce the number of

independent control parameters that appear in the optimal design model. One rather natural way of achieving this is to make an assumption on how the flow rates vary along the length of the caster. For instance, if the jth spray zone is located at $z = \omega_j$, we could assume that

$$u_j = \alpha_1 \omega_j + \alpha_2, \qquad j=1,..,m \tag{3.3}$$

thus reducing the number of independent parameters to three: α_1, α_2 and the casting speed v. In terms of these parameters, the constraints (3.1) and (3.2) become

$$\begin{aligned} &a_j \leq \alpha_1 \omega_j + \alpha_2 \leq A_j \qquad j=1,..,m, \\ &b \leq \alpha_1 \sum_{j=1}^{m} \omega_j + m\alpha_2 \leq B. \end{aligned} \tag{3.4}$$

In particular, note that the constraints are linear in α_1, α_2. If the assumption (3.3) is thought too restrictive, one could instead assume a quadratic relation

$$u_j = \alpha_1 \omega_j^2 + \alpha_2 \omega_j + \alpha_3, \qquad j=1,..,m. \tag{3.5}$$

This would give four independent parameters: $\alpha_1, \alpha_2, \alpha_3$ and v. The constraints would now become

$$\begin{aligned} &a_j \leq \alpha_1 \omega_j^2 + \alpha_2 \omega_j + \alpha_3 \leq A_j, \qquad j=1,..,m, \\ &b \leq \alpha_1 \sum_{j=1}^{m} \omega_j^2 + \alpha_2 \sum_{j=1}^{m} \omega_j + m\alpha_3 \leq B \end{aligned}$$

which are again linear in α_1, α_2 and α_3. Clearly other functional forms can be used in place of (3.3) or (3.5) - cubic, exponential etc. Although (3.3) and (3.5) restrict a priori the range of permissible parameter settings to less than that which would be physically possible, they do not seem unreasonable assumptions, as it is unlikely that the flow rates should vary erratically along the caster.

To be able to handle all the possibilities discussed above, we shall sup-

pose that there are n independent control parameters

$$\alpha_1, \ldots . \alpha_n \tag{3.6}$$

and that the casting speed v and flow rates u_j can be expressed in terms of these

$$v = v(\alpha_1, .., \alpha_n), \; u_j = u_j(\alpha_1, .., \alpha_n), \quad j=1, .., m. \tag{3.7}$$

Furthermore, we shall suppose that (3.1) and (3.2) become m+1 linear constraints

$$g_i \leq \sum_{j=1}^{m} L_{ij}\alpha_j \leq G_i \qquad (i=1, .., m+1).$$

3.2 The objective (net production rate) function

Optimal operation of the caster involves a trade-off between high gross production rates on the one hand, and the cost of treating any resulting product defects on the other. Perhaps the most crucial part of any optimization model is to develop a realistic means of quantifying this trade off, given the usually rather limited information that is available. We shall sketch a possible framework for doing this.

(A) Decide upon a number M of defect types that are to be considered. For each of these defects, develop a criteria for its occurrence which is expressable in terms of the output of the heat transfer model (that is, in terms of θ, and also maybe v). The task of formulating such defect criteria is chiefly of a metallurgical nature.

(B) For each defect type define a "degree of defect" measure $d_j, j=1, .., M$ where

i. $d_j = d_j(\theta, v)$ is a smooth function of the temperature array θ and the casting speed v.

ii. $d_j > 0$, if a defect of type j is present according to the criteria in (A)

$d_j = 0$, if the defect is not present according to (A).

Example 1. A simple defect criteria could be: Defect j will occur if the surface temperature of the slab exceeds τ^*. In this case a possible degree of defect measure could be

$$d_j(\theta,v) = \sum_{k,\ell}{}^{o} \rho(\theta_k^{[\ell]} - \tau^*) \tag{3.8}$$

where $\sum_{(k,\ell)}{}^{o}$ denotes a summation over all indices (k,ℓ) for which (x_k,z_ℓ) lies on the surface of the slab, and $\rho(.)$ is some smooth function with the property

$\rho(t) = 0$ if $t \leq 0$ and

$\rho(t) > 0$ if $t > 0$

For instance, (see Fig 2a)

$\rho(t) = 0$, if $t \leq 0$,

$\rho(t) = \beta t^n$, if $t > 0$

where $\beta > 0$ and $n = 2,3$, are appropriate constants.

The function d_j defined in (3.8) satisfies (i) and (ii) of (B). The larger the numerical value of d_j, the more serious is the violation of the defect criteria. As the defect criteria will probably only ever be approximate, it seems natural to talk in terms of such a continuously varying measure of the degree of defect, rather in terms of a simple, defect present/defect absent dichotomy.

Obviously there are many variants of (3.8). For instance, if it were thought more serious for the surface temperature to exceed τ^* at some locations along the caster than at others, then (3.8) could be changed to

$$d_j(\theta,v) = \sum_{k,\ell}{}^{o} \gamma_{k\ell}\, \rho(\theta_k^{[\ell]} - \tau^*)$$

where $\gamma_{k\ell} > 0$ are weighting factors.

Example 2. Another possible defect criteria could be: Defect j will occur if at $z=z_\ell$ the solidified shell of the slab is less than h* in thickness. For this criteria a possible degree of defect measure could be defined by

$$d_j(\theta,v) = \sum_k^* \rho(\theta_k^{[\ell]} - \tau_{SOL})$$

where τ_{SOL} is the solidus temperature, $\sum_k^*$ denotes the summation over all indices k for which x_k is within h* of the surface of the slab, and $\rho(.)$ is as in Example 1.

Example 3. As a final example, consider the defect criteria: Defect j will occur if the surface cooling/reheating rate lies outside the interval Q^-,Q^+. For this criteria we could define

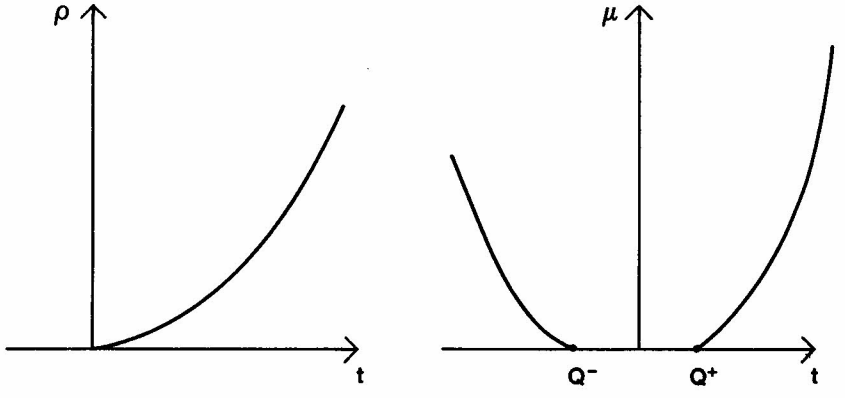
$$d_j(\theta,v) = \sum_{\ell=2}^{N_z} \sum_{k=1,N_x} \beta_{k\ell}\, \mu\left(v\, \frac{\theta_k^{[\ell]} - \theta_k^{[\ell-1]}}{z_\ell - z_{\ell-1}}\right)$$

where $\beta_{k\ell}$ are weighting factors used to emphasize some surface points (x_k,z_ℓ) relative to others, and $\mu(.)$ is a smooth function satisfying

$\mu(t) = 0$ if $Q^- \leq t \leq Q^+$,

$\mu(t) > 0$ if $\tau > Q^+$ or $\tau < Q^-$,

(see Fig. 2b).



2 (a) 2 (b)

Figures 2(a,b). Illustrative sketches of the functions $\rho(t),\mu(t)$.

(C) Penalize each defect by assigning a "cost" c_j to each defect type $j=1,..,m$, so that $c_j d_j$ is the cost per unit length of cast slab if defect j is present with the degree of defect d_j. This cost is measured in units of equivalent "ideal" (defect free) production. Strictly speaking, the c_j could be incorporated into the definition of the d_j; however, we separate the two concepts since, in some sense, d_j is a metallurgical quantity while c_j is a commercial one.

(D) Define a net production rate function $J = J(d,v)$. This should be a smooth function of $d = d_1,d_2,..,d_m$ and v. The reason for requiring J, as well as d_j in (B) to be smooth is that most optimization techniques work best with smooth objective functions. Given that there is considerable uncertainty in the definitions of J and d_j anyway, it should not be too much of an imposition to require some degree of smoothness for J and d_j.

Example 4. Perhaps the simplest net production rate function would be

$$J = J(d,v) = (Y - \sum_{j=1}^{M} c_j d_j)v \tag{3.9}$$

where Y is the "ideal" yield per unit length, assuming no defects. This must be discounted for the second term in (3.9). This assumes that the "costs" of the different defects are additive. After multiplying by the casting speed v, (3.9) therefore represents the nett production per unit time.

3.3 The optimization problem

The nett production rate J is, of course, ultimately dependent on the control parameters $\alpha = (\alpha_1,..,\alpha_n)$. More precisely,

$J = J(d,v)$ (see (D) of §3.2),

but

$d = d(\theta,v)$ (see (ii) of (B) of §3.2),

$v = v(\alpha)$ (see (3.7)),

and of course through the solution of the heat transfer model we obtain the temperature array θ as a function of α

$$\theta = \theta(\alpha).$$

Let us denote by $J(\alpha)$ this dependence of J on α.

The optimal design problem can then be formally stated as:

$$\text{maximize } J(\alpha) \tag{3.10a}$$

subject to the linear constraints (see §3.1)

$$g_i \leq \sum_{j=1}^{n} L_{ij}\alpha_j \leq G_i \quad ,(i=1,..,m+1) \tag{3.10b}$$

As has been hinted at before, the definitions of the net production rate function J and the degree of defect measures d_j are perhaps the weakest links in the approach that has been outlined so far. Therefore one should not accept any purely mathematical solution of (3.10) uncritically. Furthermore, there is probably little to be gained by solving (3.10) "exactly", rather than only approximately.

4. SOLUTION OF THE OPTIMIZATION PROBLEM

Solution of optimization problems by a single, all purpose, method is cumbersome and inefficient. Optimization algorithms are designed for particular categories of problems, where each category is determined by properties of the objective and constraint functions. Also, within each category, a range of algorithms is often available depending upon the information that is available about the derivatives of the objective and constraint functions. More information (function values, gradients, Hessians) generally means that a more efficient algorithm is available. The size (number of variables) of the problem is also an important factor in the choice of a suitable algorithm.

There are a number of general purpose optimization packages available (see the NAG Library, Chapter E04, Gill et al. (1984b), and the IMSL Library for example). These are libraries of Fortran subroutines, and are designed for use on minicomputers (a VAX for example) or larger computers. Linear algebra packages are available (through IMSL and MATLAB for example) on personal computers such as the IBM PC and PC-AT. However, software for optimization problems is not widely available on personal computers. In principle there is no reason why algorithms for small optimization problems cannot be implemented on one of the large personal computers.

The problem (3.10) is an optimization problem almost in a standard form, with a nonlinear objective function, simple bounds (3.1) on the variables and linear constraints (3.2). The standard form for optimization problems is that of a minimization problem. A maximization problem, such as (3.10), can be converted into a minimization problem simply by changing the sign of the objective function. To be consistent with the references we shall discuss optimization procedures in terms of minimizing $\tilde{J}(\alpha) = -J(\alpha)$. Typically n (the number of variables α) and m+1 (the number of constraints) are of the order of 10. This size of problem does not require an algorithm designed for large sparse problems. As there are no nonlinear constraints, the key feature in determining a suitable algorithm is the amount of derivative information available for the objective function. This is especially true for problem (3.10) as each evaluation of $J(\alpha)$ requires a solution of the heat transfer model, which is likely to be computationally expensive. Optimization algorithms which use both the objective value $\tilde{J}(\alpha)$ and the gradient $\nabla_\alpha \tilde{J}$ ($(\nabla_\alpha \tilde{J})_i = \frac{\partial \tilde{J}}{\partial \alpha_i}, i=1,\ldots,n$) are considerably more efficient than those which only use $J(\alpha)$. Thus it is very important to be able to evaluate $\nabla J(\alpha)$. The quantities needed for the evaluation of $\tilde{J}(\alpha)$ are outlined in section 3.2. However the evaluation of $\nabla_\alpha \tilde{J}(\alpha)$ is considerably more difficult, as one of the quanti-

ties required is $\nabla_\alpha \theta$. To obtain $\nabla_\alpha \theta$, access to the internal working, or even modification, of the heat conduction model may be required. For now we shall assume routines for evaluating $\tilde{J}(\alpha)$ and $\nabla_\alpha J(\alpha)$ are available, and discuss some of the methods for solving (3.10).

4.1 Methods for linearly constrained problems

Optimization algorithms generate a sequence of points $\alpha^{(k)}$ that converges to the solution α^* in the limit. A convergence test (see Fletcher (1980), Gill et al. (1981) and the NAG library) is used to terminate the computation when the current estimate of the solution is adequate. The sequence $\{\alpha^{(k)}\}$ is generated by

$$\alpha^{(k+1)} = \alpha^{(k)} + \xi^{(k)} s^{(k)} \tag{4.1}$$

where the vector $s^{(k)}$ is the direction of search and $\xi^{(k)}$ is the steplength. The steplength is computed so $\tilde{J}(\alpha^{(k+1)}) < \tilde{J}(\alpha^{(k)})$, using a technique for one-dimensional minimization. This step is called the line search.

The search direction $s^{(k)}$ is obtained using a quadratic model of the objective function, namely

$$q_k(s) = \tilde{J}(\alpha^k) + s^T \nabla_\alpha \tilde{J}(\alpha^{(k)}) + \frac{1}{2} s^T B^{(k)} s. \tag{4.2}$$

If the Hessian $\nabla_\alpha^2 \tilde{J}(\alpha)$ $([\nabla_\alpha^2 \tilde{J}(\alpha)]_{ij} = \frac{\partial^2 \tilde{J}(\alpha)}{\partial \alpha_i \partial \alpha_j})$ can be evaluated then $B^{(k)} = \nabla_\alpha^2 \tilde{J}(\alpha^{(k)})$ in (4.2). If $\nabla_\alpha^2 \tilde{J}(\alpha)$ cannot be evaluated, then $B^{(k)}$ is a positive definite quasi-Newton approximation to $\nabla_\alpha^2 \tilde{J}(\alpha)$, which is updated using the gradient differences $\nabla_\alpha \tilde{J}(\alpha^{(k+1)}) - \nabla_\alpha \tilde{J}(\alpha^{(k)})$ (see Fletcher (1980,1981) or Gill et al. (1981) for more details). It is very important that second order (curvature) information is used, either by evaluating or approximating $\nabla_\alpha^2 \tilde{J}(\alpha)$, if an algorithm is to be efficient.

The search direction $s^{(k)}$ is obtained by minimizing (4.2) subject to the linear constraints

$$a_j - \alpha_j^{(k)} \leq s_i \leq A_j - \alpha_j^{(k)}, \quad j=2,\ldots,m+1, \tag{4.3}$$

and

$$b - \sum_{j=2}^{m+1} \alpha_j^{(k)} \leq \sum_{j=2}^{m+1} s_j \leq B - \sum_{j=2}^{m+1} \alpha_j^{(k)} \tag{4.4}$$

obtained from (3.1) and (3.2). This quadratic programming subproblem can be solved in a number of ways, either using a quadratic programming routine or by using an active set method directly (Fletcher (1981, Chapt. 11), Gill et al. (1984a)). A feature of such methods is that if $\alpha^{(1)}$ is feasible and $\xi^{(k)} \leq 1$ for all k in (4.1), then $\alpha^{(k)}$ is a feasible point on every iteration. Thus even if the optimal solution has not been reached, a point, with lower function value than the starting point and satisfying all the constraints, is always available.

If, in equation (4.2), $B^{(k)} = \nabla_\alpha^2 \tilde{J}(\alpha^{(k)})$, another strategy is to replace the line search (4.1) by a trust region algorithm, in which the constraint

$$||s||_p \leq r^{(k)} \tag{4.5}$$

is added to the constraints (4.3) and (4.4) in the subproblem determining $s^{(k)}$. If $p = 2$, so $||s||_2^2 = \sum_{i=1}^{n} s_i^2$, then (4.5) represents a spherical trust region of radius $r^{(k)}$, while if $p = \infty$, so $||s||_\infty = \max\{|s_i|, i=1,\ldots,n\}$, then (4.5) represents a box-like trust region. The size of the trust region is governed by $r^{(k)}$ which is updated according to rules based on the agreement between $\tilde{J}(\alpha^{(k)} + s^{(k)})$ and the quadratic model $q_k(s^{(k)})$ (see Fletcher (1980), Gill et al. (1981) for more details).

Trust region methods are some of the most efficient methods for nonlinear optimization. However, they have the major disadvantage of requiring $\nabla_\alpha^2 \tilde{J}(\alpha)$ to be evaluated. When only the gradient $\nabla_\alpha \tilde{J}(\alpha)$ is available, the most efficient method is to use a quasi-Newton approximation to $\nabla_\alpha^2 \tilde{J}(\alpha)$. If only the function value $\tilde{J}(\alpha)$ can be evaluated, then the usual strategy is to use finite

differences to approximate $\nabla_\alpha \tilde{J}(\alpha)$ (Gill et al. (1983)) and then use this approximation in a quasi-Newton method. This can be very expensive as at least n extra function evaluations are required to evaluate $\nabla_\alpha \tilde{J}(\alpha)$. Thus it is essential that $\nabla_\alpha \tilde{J}(\alpha)$ is evaluated if possible.

5. SUMMARY

The suggested steps for developing a model for the steady state problem are:

1. develop criteria for the occurrence of defects in terms of the independent parameters (section 3.1).

2. develop a net production rate function in terms of the independent parameters (section 3.2).

3. use optimization software to provide good values of the parameters (section 4).

At each stage the models must be verified by comparison with industrial experience and current operating practices, before they are used to investigate the performance of the system under new conditions.

Note that initially considerable advantage can be gained by reducing the number of independent variables as discussed in section 3.1. The very small number of parameters (3 or 4) so obtained would allow graphical investigation of the net production rate function $J(\alpha)$. As a first step one could just look at values of $J(\alpha)$ obtained for different parameter values, without using any optimization software. The very small number of parameters would also make finite difference approximations to $\nabla_\alpha J(\alpha)$ practical, so the current heat conduction model could be used in conjunction with efficient optimization software. Further developments may well require modifications to the current heat conduction model.

REFERENCES

Fletcher, R., Practical methods of optimization, Vol. 1 : Unconstrained Optimization (John Wiley, Chichester, 1980).

Fletcher, R., Practical methods of optimization, Vol. 2 : Constrained Optimization (John Wiley, Chichester, 1981).

Gill, P.E., Murray, W., Saunders, M.A. and Wright, M.H., "Computing forward difference intervals for numerical optimization", SIAM Journal on Scientific and Statistical Computing 4 (1983) 310-321.

Gill, P.E., Murray, W., Saunders, M.A. and Wright, M.H., "Procedures for optimization problems with a mixture of bounds and general linear constraints", ACM Transactions on Mathematical Software 10 (1984) 282-298.

Gill, P.E., Murray, W., Saunders, M.A. and Wright, M.H., "User's guide for SOL/NPSOL : A Fortran package for Nonlinear Programming", Technical Report 84-7, Systems Optimization Laboratory, Stanford University, (1984).

Gill, P.E., Murray, W. and Wright, M.H., Practical Optimization (Academic Press, London and New York, 1981).

IMSL Library and IMSL MATH/PC Library Manuals, IMSL, Houston, Texas.

NAG Fortran Library Manual Mark 11, Numerical Algorithms Group, Oxford (1984).

PC-MATLAB Manual, The Math Works Inc., Portola Valley, California.