in Proceedings of the Eighth Inverse Problems in Engineering Seminar, A. Maniatty ed., Rensselaer Polytechnic Institute, 1997.

# INVERSE APPROACH TO DESIGN OF STEADY FORMING PROCESSES

ANTOINETTE M. MANIATTY AND MING-FA CHEN*

Department of Mechanical Engineering,
Aeronautical Engineering and Mechanics
Rensselaer Polytechnic Institute
Troy, NY 12180, U.S.A.

## SUMMARY

This paper presents a procedure for optimal process design of steady metal forming operations. The problem is formulated as a constrained optimization problem with inequality constraints. An augmented Lagrangian method is used to convert the constrained minimization problem to a sequence of unconstrained problems. The numerical algorithm applies a quasi-Newton method for minimizing the augmented Lagrangian and employs an adjoint method for formulating the sensitivity coefficients used in the minimization procedure. Two design problems are considered here. The first case involves minimizing the process power requirement, and the second case involves producing a specified mechanical property distribution in formed materials. The design parameters are the process geometry and speed. Numerical examples are presented for drawing, rolling, and extrusion to show the stability and convergence of the numerical algorithm.

## 1. INTRODUCTION

Using numerical simulation for designing forming processes in industry is becoming a common practice. In the modeling of bulk forming processes, generally a system of governing equations including the balance of linear momentum, kinematic equations, and constitutive relations for the material are specified along with a description of the boundary and initial conditions. Then a solution for the deformation, stress, and state variable fields throughout the domain of interest is found. This is referred to as a direct problem. Significant progress has been made in recent years using high powered computing facilities for the numerical solution of these problems involving complicated deformations, constitutive laws, and boundary conditions. The current design practice where numerical simulations are used generally is as follows: (i) propose a design; (ii) run a simulation of the process for the given design; (iii) examine the results of the simulation to determine if the design is acceptable; (iv) if the design is acceptable, use it, if not, go back to step (i). This practice saves much time and expense because it allows a designer to "try out" a design on a computer and see how it fairs before actually building the design. However, this method of process design is still trial and error. In reality, what is desired is a numerical algorithm that can determine the optimal design parameters for the designer given certain design criteria and constraints. Such problems as these can be classified as either optimization or inverse problems and are the concern of this paper.

In formed material products, product quality generally depends on material characteristics such as hardness, surface finish, geometric tolerances, residual stresses, etc. These material characteristics, in turn, depend on the forming process parameters such as geometry, temperature, and speed of both the

---

* Currently at China Steel Corporation, Kaohsiung, Taiwan, Republic of China.

tool and the workpiece. The process cost depends on the production time and the energy required to carry out the process which also, in turn, depend on the forming process parameters. The process designer must carefully choose these parameters in order to produce the desired products at a low cost. This is a process design problem. Most studies in recent years formulate this design problem as a series of direct problems. A direct problem is mathematically based on a set of input information, including the domain, governing equations, constitutive relations, and boundary conditions. After this information is provided, a solution for the deformation, stress, and internal state variable fields through the deformation zone is calculated. In the forward solution procedure, different design parameters are tried and the direct problem is solved repeatedly until a satisfactory design is reached.

In fact, a solution formulation for such design problems should start with the product specifications as input, and then compute the process design parameters which will generate the desired product. This formulation for a design problem in forming is effectively the inverse of the forward or direct problem formulation. Hence, this class of problem is sometimes termed inverse problems (see for example Beck et al. [1]). In inverse problems unknown boundary conditions or parameters are determined based on given data regarding the solution (or desired solution). Such a problem can also be considered an optimization problem since the goal is to determine the optimal design parameters for generating a desired product. In this work, the problem is formulated as an optimization problem.

Only in recent years have inverse/optimization methodologies been used to design forming processes. Kusiak and Thompson [2] determined the optimal die shape for steady-state extrusion to minimize the ram force and to produce a uniform exit velocity during the extrusion of bimetallic rods. They treated the material as a viscous fluid with a constant viscosity. In their analysis, they compare the non-gradient Hooke-Jeeves, the steepest descent, and the gradient Fletcher-Powell methods for minimizing the objective functions. Legat and Marchal [3] used an implicit formulation and iterative Newton-Raphson method to solve the inverse steady-state extrusion problem for designing the optimal die shape that produces an extrudate of prescribed shape for a Newtonian flow. Han et al. [4] derived a formulation for the preform design of axisymmetric forging processes. Fourment and Chenot [5] and Fourment et al. [6] extended Han's method for designing the preform workpiece and tool optimization to multiple step forging processes. They have a velocity based finite element formulation which allows for h-refinement adaptivity. Badrinarayanan and Zabaras [7] proposed a direct differentiation method to calculate shape sensitivity coefficients and applied a gradient-based method for the optimal design of some simple transient forming processes in the context of a displacement based finite element procedure. Michaleris et al. [8] present a systematic approach for the design of weakly coupled thermoelastoplastic systems. This paper focuses on developing a design tool to determine the optimum process parameters for two dimensional steady forming operations such as drawing or rolling. The process parameters considered are the tool shape and the driving velocity.

Most optimization problems involve searching for the non-zero minimum of a function. For nonlinear problems developing an efficient algorithm that converges is difficult because the function may be nonconvex. Furthermore, since the problem under consideration is an inverse problem, it will in generally be ill-posed, i. e. the solution does not satisfy the general conditions of existence, uniqueness and stability.

In this work, a finite element model for solving the direct forming problem is used as a basis for formulating the inverse problem. The discretized form of the direct problem results in two coupled nonlinear systems of equations. Therefore, the optimization problem will also have a nonlinear structure. Hence, the consumption of the computational time and the accuracy of the derivatives used in a solution-searching formula have to be carefully considered in the numerical algorithm. The efficiency of the algorithm is in turn related to the nonlinearity of the problem. Finally, due to the ill-posedness of the problem, uniqueness of the solution should be examine. This is still a topic of active research in this field.

This work proposes a solution procedure to solve for the nonlinear optimization/inverse problem applied to the design of steady forming processes. The problem is defined in terms of objective functions to be minimized subject to specified constraints, generally inequalities. The objective function represents either the input power or the deviation between the specified and simulated internal state variables. In this study, the constrained optimization problem is transformed to an unconstrained optimization problem by adding the constraint terms to the objective function by an augmented multiplier (Lagrangian) formulation [12]. Multiplier methods are widely used in optimization problems and have

been shown to aid in approaching the solution in a more stable manner. This study chooses a minimization procedure with the consideration of the numerical error and the efficiency in the calculation of the function gradients. The BFGS (Broyden-Fletcher-Goldfarb-Shanno) quasi-Newton method (Fletcher [12]) was selected because it was found to be superior for such function minimization.

A sensitivity analysis is needed to determine the design sensitivity parameters needed for the gradient based minimization procedure. The problem of shape design sensitivity analysis for a variety of problems, both linear and nonlinear, using the adjoint method is presented in Arora and Cardoso [13]. Zhang et al. [14] determine design sensitivity coefficients for finite deformation elasto-viscoplastic problems using the direct differentiation approach in a boundary element formulation context. Michaleris et al. [15] derived equations for tangent and design sensitivity parameters for steady-state and transient non-linear elastic-plastic problems. Vaz and Hinton [16] present a semi-analytic method for determining finite element shape sensitivity parameters for elasto-plastic analyses. Maniatty and Chen [17] presented an explicit matrix formulation of sensitivities for steady metal forming processes using the adjoint method. This study uses this explicit matrix formulation for the efficient calculation of design sensitivities.

In this work, the direct forming problem which is used as a basis for the optimization procedure is simulated by a coupled, Eulerian, viscoplastic finite element formulation neglecting thermal effects. The elastic part is also neglected for simplicity and the deformation is assumed incompressible. A unified isotropic flow theory is used for the constitutive relations. The viscoplastic material behavior is modeled using a scalar internal variable that represents the isotropic resistance to plastic flow. Some examples including a sheet drawing, a flat rolling, and a axisymmetric extrusion processes are used to investigate the stability and convergence of the algorithm for the process design problem. The formulation of the constraints and their effect on the convergence and stability are discussed.

## 2. PROBLEM DEFINITION

In the problem considered herein, the desired product characteristics are specified as input and the process parameters including geometry and speed are calculated as output. The solution method involves the Eulerian finite element method for direct problems and a modified formulation using inverse techniques for determining the unknown shape parameters and, in some cases, speed. The following two sections describe this process design problem.

### *2.1 Formulation of the direct problem*

The direct forming problem of interest here is defined by the domain of interest, governing equations and complete boundary conditions given a priori. The formulation and solution procedure for the direct problem follows that given in Dawson [18]. Summarizing, consider a two dimensional domain $B$ with boundary $\partial B$ where the material being deformed is flowing steadily through the domain. The boundary value problem for equilibrium on $B$ is regarded as a control volume problem and is expressed in the following manner:

$$
\begin{aligned}
&div\mathbf{T} = \mathbf{0} && \text{on } B && \\
&\mathbf{e}_i \cdot \mathbf{u} = \hat{u}_i && \text{on } \partial B_{1i} && \text{i=1, 2} \\
&\mathbf{e}_i \cdot (\mathbf{Tn}) = \hat{\mathrm{t}}_i && \text{on } \partial B_{2i} && \text{i=1, 2} \\
&\mathbf{e}_i \cdot (\mathbf{Tn}) = \beta(u_{0i} - u_i) && \text{on } \partial B_{3i} && \text{i=1, 2}
\end{aligned}
$$

where $\mathbf{T}$ is the Cauchy stress tensor, $\hat{u}_i$ is the velocity specified on $\partial B_{1i}$, $\hat{t}_i$ is the traction specified on $\partial B_{2i}$, $\mathbf{n}$ is the unit normal vector on $\partial B$, and $\mathbf{e}_i$ form an orthonormal basis for the two dimensional space with i=1, 2. Furthermore, the fourth equation represents a hydrodynamic friction law where $u_{0i}$ is the tangential velocity of the tool on $\partial B_{3i}$ and $\beta$ is the coefficient of hydrodynamic friction. In general,

$\beta$ is a function of the temperature and the normal traction, but in this analysis, it will be taken as constant for simplicity. The boundary conditions must be specified on the entire boundary for each degree of freedom without overlap, so $\partial B_{1i} \cup \partial B_{2i} \cup \partial B_{3i} = \partial B$ and $\partial B_{ji} \cap \partial B_{ki} = \phi$ for i=1, 2 and j≠k.

The material model in this study uses an internal state variable to represent the deformation resistance of isotropic viscoplastic flow. The deviatoric stress tensor is proportional to the plastic rate of deformation tensor with the coefficient depending on the rate of deformation acting as an effective nonlinear viscosity. Elasticity will be neglected for simplicity. This is reasonable since the elastic deformations are in general very small relative to the viscoplastic deformations and since the residual stresses are not being considered.

In addition, an evolution equation for the state variable, $s$, is required to describe the evolution of the material state that arises during deformation. This can be expressed as

$$\dot{s} = g(\dot{\bar{\varepsilon}}, s)$$

where dot $(\dot{})$ denotes a material time derivative and $\dot{\bar{\varepsilon}}$ is the effective rate of deformation. Since steady problems are being considered, $\dot{s} = \nabla s \cdot \mathbf{u}$. Now a boundary value problem for the state variable can be expressed as

$$\nabla s \cdot \mathbf{u} = g(\dot{\bar{\varepsilon}}, s) \qquad \text{on } B$$
$$s = \hat{s} \qquad \text{on } \partial B_s$$

where $\hat{s}$ is the initial value of the state variable on the entrance boundary $\partial B_s$.

Thus, the direct problem consists of two nonlinear systems of equations in the finite element formulation, where the unknown state fields are nodal velocities and nodal state variables. These resulting equations are:

$$\underline{\mathbf{k}}(\underline{\mathbf{u}}, \underline{\mathbf{s}}) = \underline{\mathbf{f}} \tag{1}$$
$$\underline{\mathbf{h}}(\underline{\mathbf{u}}, \underline{\mathbf{s}}) = \mathbf{0} \quad . \tag{2}$$

where $\underline{\mathbf{u}}$ and $\underline{\mathbf{s}}$ are vectors of the nodal velocities and state variables, respectively, and $\underline{\mathbf{f}}$ is the force vector.

A consistent penalty method is employed in equation (1) to enforce the incompressibility constraint. A staggered solution procedure is used for solving the nonlinear system. First, keeping $\underline{\mathbf{s}}$ constant, solve equation (1) for $\underline{\mathbf{u}}$ by using a combined successive substitution and Newton-Raphson method; second, keeping $\underline{\mathbf{u}}$ constant, solve equation (2) for $\underline{\mathbf{s}}$ by the Newton-Raphson method. Repeat the two steps until the solution reaches the convergence criteria.

*2.2 Formulation of process design problem*

The process design problem herein involves an incomplete domain and unknown boundary conditions on part of the boundaries. In this study the goal is to determine the process parameters such that the forming process produces two types of desired design results. One design obtains the lowest input power to perform the required deformation. The other design generates a state variable distribution in the final product that is as close as possible to a specified distribution. The two process design problems can be defined in the same way as the direct problem, but now the shape of part of the boundary $\partial\hat{B}$ which represents the tool contact zone is not known and the driving velocity, for example the drawing rate, $u_d$ may also not be known. Let the shape of $\partial\hat{B}$ defined by position vector $\hat{\mathbf{x}}(\xi)$ be parameterized by:

$$\mathbf{e}_i \cdot \hat{\mathbf{x}}(\xi) = b_{i\alpha}\varphi_\alpha(\xi) \quad \hat{\mathbf{x}} \in \partial\hat{B} \qquad i = 1,2 \quad \alpha = 1, N_{\hat{B}} \tag{3}$$

where $b_{i\alpha}$ (or $\underline{\mathbf{b}}$ in direct notation) are the shape design parameters, $\varphi_\alpha$ are shape functions, x is a measure of distance along $\partial\hat{B}$ , and $N_{\hat{B}}$ is the number of discrete points used to parameterize the boundary $\partial\hat{B}$. Summation is assumed on the repeated index $\alpha$. Now the nonlinear system of governing equations (1) and (2) becomes

$$\underline{\mathbf{k}}(\underline{\mathbf{u}},\underline{\mathbf{s}},\underline{\mathbf{b}}) = \underline{\mathbf{f}}(\underline{\mathbf{b}},u_d) \tag{4}$$

$$\underline{\mathbf{h}}(\underline{\mathbf{u}},\underline{\mathbf{s}},\underline{\mathbf{b}}) = \mathbf{0} \quad . \tag{5}$$

where $\underline{\mathbf{b}}$ and $u_d$ are added to the list of variables (velocity boundary condition only affects forcing term in discretized equation). It should be noted that in many cases only one component of the position vector $\hat{\mathbf{x}}(\xi)$ will be variable, i.e. $i$ = 1 or 2. So to be more general, let $N_d$ be the total number of design variables.

This work constructs the formulation of the two process design problems for the solutions by minimizing the discretized functions with respect to the design variables, $\underline{\mathbf{b}}$ and $u_d$, as follows:

$$\min\ (\underline{\mathbf{f}}'^{T}\ \underline{\mathbf{u}}') \qquad \text{: for lowest input power} \tag{6}$$

$$\min\ (\underline{\mathbf{s}}' - \hat{\underline{\mathbf{s}}}')^T(\underline{\mathbf{s}}' - \hat{\underline{\mathbf{s}}}') \qquad \text{: for desired state variables} \tag{7}$$

where $\underline{\mathbf{f}}'$ is the nodal force vector, $\underline{\mathbf{u}}'$ is the vector of velocities in the direction of $\underline{\mathbf{f}}'$ on the boundary $\partial B'$ on which the load is applied, $\underline{\mathbf{s}}'$ is the vector of nodal state variables at the discrete locations in the exit region $\partial B_e$, and $\hat{\underline{\mathbf{s}}}'$ is the corresponding vector of desired state variables. In fact, it does not make sense to consider the driving velocity in minimizing the power because the minimum power consumption will be zero when the driving speed is zero, so the result would always be driven to a zero velocity. So $u_d$ will only be considered a design variable for the second case defined by equation (7) which involved obtaining a desired state variable field. The above objective functions must be minimized subject to a set of constraints on the geometry and possibly process speed:

$$G_i(\underline{\mathbf{b}},u_d) \le 0 \qquad i = 1, N_c \tag{8}$$

where $N_c$ is the number of constraint equations. The constraints are generally inequality constraints.

## 3. SOLUTION PROCEDURE

The objective functions of the process design problem shown in the previous section are implicit and nonlinear with respect to the design variables. In addition, the application of the constraints affects the stability and needs to be considered carefully. This study develops a solution procedure for the process design problem, which considers an efficient algorithm to update the solution and to perform a stable convergence for the solution. An augmented Lagrangian is formulated for converting the constrained minimization to an unconstrained minimization problem. Then a BFGS procedure with a line search is used to perform the minimization. A procedure for updating the constraint parameters is also used. The next sections describe these procedures.

*3.1 Augmented Lagrangian Formulation*

In this paper, the solution of the process design problem requires the minimization of an objective function, for which the forming process obtains the lowest power requirement or produces the desired material state in the final product, subject to prescribed constraints. Forming augmented Lagrangians to be minimized with respect to $\underline{\mathbf{b}}$ and $u_d$ for the objective functions in equations (6) and (7) incorporating the constraints in equation (8) gives:

$$\Lambda^p = (\underline{\mathbf{f}}'^{T}\ \underline{\mathbf{u}}') + \Omega \tag{9}$$

$$\Lambda^s = (\underline{\mathbf{s}}' - \hat{\underline{\mathbf{s}}}')^T(\underline{\mathbf{s}}' - \hat{\underline{\mathbf{s}}}') + \Omega \tag{10}$$

where $\Omega$ is a generalized penalty function which imposes the constraints. The generalized penalty function may be either explicit or implicit in form. If the problem used the discretized position coordinates as design variables, the function can be expressed in explicit form to restrict the domain of the design variables. For example, the constraint may only specify that the nodal coordinates fall between some maximum and minimum values. This is a relatively weak constraint. If the problem defines a

smooth function to fit the tool shape where the design parameters are related to the coefficients of the function, then the generalized penalty function is implicit. This is generally a stronger constraint than the explicit constraint. The construction of the generalized penalty function affects the convergence and efficiency of the minimization solution. Examples of how to construct the penalty function in equations (9) and (10) are discussed in Arora et al. [9]. After considering numerous methods for applying the constraints, the following form was found to give the most robust algorithm for the problems considered here in terms of stability:

$$\Omega = \frac{1}{2}\sum_{i=1}^{N_c} \lambda_i^c \left( \left\langle G_i(\underline{\mathbf{b}}, u_d) + \omega_i \right\rangle^2 + \hat{c} \left\langle G_i(\underline{\mathbf{b}}, u_d) + \eta \right\rangle \left\langle \lambda_i^c - \hat{\lambda}^c \right\rangle \right) \tag{11}$$

where $\lambda_i^c$ and $\omega_i$ are penalty parameters and Lagrange multipliers, respectively, for each constraint. These parameters need to be updated in the minimization procedure. The angle brackets $\langle \cdot \rangle$ denote a singularity function, i. e.,

$$\langle x \rangle = \begin{cases} 0 \text{ if } x \le 0 \\ x \text{ if } x > 0 \end{cases}$$

The parameters $\eta$ and $\hat{\lambda}^c$ remain fixed, and the parameter $\hat{c}$ is initially set to zero and if convergence problems occur because of the constraint equation, $\hat{c}$ is set to a large number forcing the solution to move towards the feasible region away from the constraint boundaries.

*3.2 Minimization Procedure*

Minimizing the augmented Lagrangians in equations (9) and (10) where the generalized penalty function, $\Omega$, is defined in equation (11) has been found to be very difficult because these function are generally highly nonlinear and the design variables are bounded by physical constraints. To be able to solve this problem, the numerical algorithm should employ an efficient minimization procedure and a suitable procedure to update the constraint parameters. In this work the minimization of the augmented Lagrangians is performed by the BFGS method with a line search scheme. The basic formula of this method is:

$$\underline{\mathbf{q}}^{(r+1)} = \underline{\mathbf{q}}^{(r)} + \alpha \underline{\mathbf{d}}^{(r)} \tag{12}$$

where $\underline{\mathbf{q}}^{(r)} = \left( \underline{\mathbf{b}}^{(r)}, u^{(r)} \right)$ is the vector of design variables, $\underline{\mathbf{d}}^{(r)}$ is the search direction, $\alpha$ is the line search parameter, and superscripts $(r)$ and $(r + 1)$ indicate the iteration number. The design variables, $\underline{\mathbf{q}}^{(r)}$, are updated at each iteration until some convergence criteria are satisfied.

The gradient-based BFGS method is based on searching the solution in a descent direct obtained from Taylor's expansion. The derivative of the function is calculated by the sensitivity matrix that is evaluated from the sensitivity analysis. In this framework, the multi-dimensional domain of the problem is converted into a one-dimensional variable, $\alpha$. Then a line search is performed for the minimization of the function with respect to the scalar variable $\alpha$. In this study, it is assumed that the minimum can be bracketed and uses a modified Brent method [17].

In the minimization considered here, the constraints affect the well-posedness of the problem. This work uses a procedure modified from the Powell's multiplier algorithm (Powell [18]) to update these parameters. At each iteration of the minimization procedure, a check is made for a constraint violation. If a constraint violation exists, the Lagrange multipliers and the penalty parameter are updated. This procedure guarantees global convergence (not a global minimum).

The algorithm can be summarized as follows:

(1) Initialize the design parameters, $\underline{\mathbf{q}}^{(0)} = \left( \underline{\mathbf{b}}^{(0)}, u^{(0)} \right)$, the Lagrange multipliers and penalty parameters, $\lambda_i^{c(0)}$ and $\omega_i^{(0)}$, and set $\hat{c} = 0$. Initialize the iteration number for the minimization algorithm, $r = 0$, and for the multiplier update procedure, $r' = 0$.

(2) Solved the direct problem, equations (4) and (5), for the nodal velocities, $\underline{\mathbf{u}}^{(0)}$, and state variables, $\underline{\mathbf{s}}^{(0)}$.

(3) Compute the gradient of the augmented Lagrangian

$$\underline{\mathbf{g}}_1^{(0)} = \frac{d\Lambda^{p\,(0)}}{d\underline{\mathbf{q}}} = \underline{\mathbf{B}}^{(0)^T}\underline{\mathbf{u}}'^{(0)} + \sum_{i=1}^{N_c}\lambda_i^{c(0)}\left\langle G_i^{(0)} + \omega_i\right\rangle\frac{dG_i^{(0)}}{d\underline{\mathbf{q}}} \tag{13}$$

$$\underline{\mathbf{g}}_2^{(0)} = \frac{d\Lambda^{s\,(0)}}{d\underline{\mathbf{q}}} = \underline{\mathbf{C}}^{(0)^T}\left(\underline{\mathbf{s}}'^{(0)} - \underline{\hat{\mathbf{s}}}'\right) + \sum_{i=1}^{N_c}\lambda_i^{c(0)}\left\langle G_i^{(0)} + \omega_i^{(0)}\right\rangle\frac{dG_i^{(0)}}{d\underline{\mathbf{q}}} \tag{14}$$

where $\underline{\mathbf{g}}_1^{(0)}$ is the gradient for the case concerning minimizing the power required and $\underline{\mathbf{g}}_2^{(0)}$ is the gradient for the case concerning minimizing the difference between the model state variables and the desired state variables. The matrices $\underline{\mathbf{B}}^{(0)} = \frac{d\underline{\mathbf{f}}'^{(0)}}{d\underline{\mathbf{q}}}$ and $\underline{\mathbf{C}}^{(0)} = \frac{d\underline{\mathbf{s}}'^{(0)}}{d\underline{\mathbf{q}}}$ are the design sensitivity matrices. Note that since $u_d$ is not a design variable for the case of minimizing power, $\underline{\mathbf{q}} = \underline{\mathbf{b}}$ in equation (13). The details for computing the sensitivity matrices is presented in [15] and summarized in section 3.3.

(4) Initialize approximation to the inverse Hessian with identity matrix, $\underline{\mathbf{H}}^{(0)} = \underline{\mathbf{I}}$.

(5) Compute the search direction, $\underline{\mathbf{d}}^{(r)} = -\underline{\mathbf{H}}^{(r)}\underline{\mathbf{g}}^{(r)}$.

(6) Update the vector of design parameters with a line search, i. e. solve equation (12) for the best value of $\alpha$. The details of this procedure are given in section 3.4. It should be noted that in the course of the line search, the parameters $\underline{\mathbf{u}}^{(r+1)}$, $\underline{\mathbf{s}}^{(r+1)}$, $\Lambda^{(r+1)}$ and $\underline{\mathbf{g}}^{(r+1)}$ are computed.

(7) Check for convergence using the following criteria:

$$\Lambda^{(r)} \le \varepsilon_1$$
$$\left|\Lambda^{(r)} - \Lambda^{(r-1)}\right| \le \varepsilon_2\Lambda^{(r)}$$
$$\sqrt{\underline{\mathbf{g}}^{(r)^T}\underline{\mathbf{g}}^{(r)}} \le \varepsilon_3$$

where $\varepsilon_1$, $\varepsilon_2$ and $\varepsilon_3$ are tolerances on the objective function, the change in the objective function over the last iteration, and the magnitude of the gradient of the objective function, respectively. If all the convergence criteria are met, then the algorithm is finished, otherwise, continue.

(8) Check the generalized penalty function, $\Omega^{(r+1)}$, for a constraint violation, i.e. check if $\Omega^{(r+1)} > 0$. If a constraint violation exists, update the Lagrange multipliers and penalty parameters according to the algorithm described in section 3.5, reinitialize $\underline{\mathbf{H}}^{(r+1)} = \underline{\mathbf{I}}$, update $\underline{\mathbf{g}}^{(r+1)}$, and go back to step (5), otherwise continue.

(9) Update the approximation to the inverse Hessian with the BFGS formula.

$$\underline{\mathbf{H}}^{(r+1)} = \underline{\mathbf{H}}^{(r)} + \left(1 + \frac{\underline{\mathbf{w}}^{(r)^T}\underline{\mathbf{H}}^{(r)}\underline{\mathbf{w}}^{(r)}}{\underline{\mathbf{v}}^{(r)^T}\underline{\mathbf{w}}^{(r)}}\right)\frac{\underline{\mathbf{v}}^{(r)}\underline{\mathbf{v}}^{(r)^T}}{\underline{\mathbf{v}}^{(r)^T}\underline{\mathbf{w}}^{(r)}} - \left(\frac{\underline{\mathbf{v}}^{(r)}\underline{\mathbf{w}}^{(r)^T}\underline{\mathbf{H}}^{(r)} + \underline{\mathbf{H}}^{(r)}\underline{\mathbf{w}}^{(r)}\underline{\mathbf{v}}^{(r)^T}}{\underline{\mathbf{v}}^{(r)^T}\underline{\mathbf{w}}^{(r)}}\right)$$

where $\underline{\mathbf{v}}^{(r)} = \underline{\mathbf{q}}^{(r+1)} - \underline{\mathbf{q}}^{(r)}$ and $\underline{\mathbf{w}}^{(r)} = \underline{\mathbf{g}}^{(r+1)} - \underline{\mathbf{g}}^{(r)}$. Go back to step (5).

*3.3 Nonlinear Sensitivity Analysis*

An sensitivity analysis measures how sensitive the design state field is with respect to design parameters and provides state gradient information that can be used in the minimization algorithm resulting from the process design problem.

The first case considered is the problem of determining the optimal process geometry to minimizing the power required for the forming process. The discrete sensitivities which arise in the gradient equation (13) are expressed by the following matrix:

$$\underline{\mathbf{B}} = \frac{d\underline{\mathbf{f}}'}{d\underline{\mathbf{b}}}$$

where recall $\underline{\mathbf{q}} = \underline{\mathbf{b}}$ since in this case the process driving speed, $u_d$, is not considered a design parameter, and $\underline{\mathbf{f}}'$ is the vector of nodal forces on the boundary $\partial B'$ where the tool drives the workpiece. The elements of this matrix define locally how the nodal forces $\underline{\mathbf{f}}'$ vary with the geometry parameters $\underline{\mathbf{b}}$. It should be noted that it is assumed that the tool velocity is constant on this boundary so the problem of minimizing the power reduces to a problem of minimizing the sum of the nodal forces on $\partial B'$.

The second case considered is the problem of determining the optimum process geometry and possibly driving velocity to generate a state variable field in the final product that is as closed as possible to a prescribed state variable field. In this case, the discrete sensitivities which arise in equation (14) are expressed by the matrix

$$\underline{\mathbf{C}} = \frac{d\underline{\mathbf{s}}'}{d\underline{\mathbf{q}}}$$

where $\underline{\mathbf{s}}'$ is the vector of nodal state variables on the exit boundary $\partial B_e$. The elements of this matrix define locally how the nodal state variables $\underline{\mathbf{s}}'$ vary with the geometric parameters $\underline{\mathbf{b}}$ and the driving speed $u_d$.

The adjoint method is used to determine the matrices B and C. For details of this method, see Maniatty and Chen [15]. The resulting equations are

$$\underline{\mathbf{B}} = \frac{d\underline{\mathbf{f}}'}{d\underline{\mathbf{b}}} = \underline{\mathbf{K}}_{\mathbf{b}}^{*} + \underline{\mathbf{K}}_{\mathbf{u}}^{*}\left(\underline{\mathbf{K}}_{\mathbf{u}} - \underline{\mathbf{K}}_{\mathbf{s}}\underline{\mathbf{H}}_{\mathbf{s}}^{-1}\underline{\mathbf{H}}_{\mathbf{u}}\right)^{-1}\left(\underline{\mathbf{F}}_{\mathbf{b}} - \underline{\mathbf{K}}_{\mathbf{b}}\right) + \underline{\mathbf{K}}_{\mathbf{s}}^{*}\left(\underline{\mathbf{K}}_{\mathbf{s}} - \underline{\mathbf{K}}_{\mathbf{u}}\underline{\mathbf{H}}_{\mathbf{u}}^{-1}\underline{\mathbf{H}}_{\mathbf{s}}\right)^{-1}\left(\underline{\mathbf{F}}_{\mathbf{b}} - \underline{\mathbf{K}}_{\mathbf{b}}\right)$$
$$+\underline{\mathbf{K}}_{\mathbf{u}}^{*}\left(\underline{\mathbf{H}}_{\mathbf{s}}\underline{\mathbf{K}}_{\mathbf{s}}^{-1}\underline{\mathbf{K}}_{\mathbf{u}} - \underline{\mathbf{H}}_{\mathbf{u}}\right)^{-1}\underline{\mathbf{H}}_{\mathbf{b}} + \underline{\mathbf{K}}_{\mathbf{s}}^{*}\left(\underline{\mathbf{H}}_{\mathbf{u}}\underline{\mathbf{K}}_{\mathbf{u}}^{-1}\underline{\mathbf{K}}_{\mathbf{s}} - \underline{\mathbf{H}}_{\mathbf{s}}\right)^{-1}\underline{\mathbf{H}}_{\mathbf{b}} \tag{15}$$

and

$$\underline{\mathbf{C}} = \left(\underline{\mathbf{C}}_{\mathbf{b}}, \underline{\mathbf{C}}_{u_d}\right) \tag{16a}$$

$$\underline{\mathbf{C}}_{\mathbf{b}} = \frac{d\underline{\mathbf{s}}'}{d\underline{\mathbf{b}}} = \underline{\mathbf{R}}\left[\left(\underline{\mathbf{K}}_{\mathbf{s}} - \underline{\mathbf{K}}_{\mathbf{u}}\underline{\mathbf{H}}_{\mathbf{u}}^{-1}\underline{\mathbf{H}}_{\mathbf{s}}\right)^{-1}\left(\underline{\mathbf{F}}_{\mathbf{b}} - \underline{\mathbf{K}}_{\mathbf{b}}\right) + \left(\underline{\mathbf{H}}_{\mathbf{u}}\underline{\mathbf{K}}_{\mathbf{u}}^{-1}\underline{\mathbf{K}}_{\mathbf{s}} - \underline{\mathbf{H}}_{\mathbf{s}}\right)^{-1}\underline{\mathbf{H}}_{\mathbf{b}}\right] \tag{16b}$$

$$\underline{\mathbf{C}}_{u_d} = \frac{d\underline{\mathbf{s}}'}{du_d} = \underline{\mathbf{R}}\left(\underline{\mathbf{K}}_{\mathbf{s}} - \underline{\mathbf{K}}_{\mathbf{u}}\underline{\mathbf{H}}_{\mathbf{u}}^{-1}\underline{\mathbf{H}}_{\mathbf{s}}\right)^{-1}\underline{\mathbf{f}}_{u_d} \tag{16c}$$

where

$$\underline{\mathbf{K}}_{\mathbf{b}}^{*} = \underline{\mathbf{Q}}\frac{\partial \underline{\mathbf{k}}^{*}}{\partial \underline{\mathbf{b}}} \qquad \underline{\mathbf{K}}_{\mathbf{u}}^{*} = \underline{\mathbf{Q}}\frac{\partial \underline{\mathbf{k}}^{*}}{\partial \underline{\mathbf{u}}} \qquad \underline{\mathbf{K}}_{\mathbf{s}}^{*} = \underline{\mathbf{Q}}\frac{\partial \underline{\mathbf{k}}^{*}}{\partial \underline{\mathbf{s}}}$$

$$\underline{\mathbf{K}}_{\mathbf{b}} = \frac{\partial \underline{\mathbf{k}}}{\partial \underline{\mathbf{b}}} \qquad \underline{\mathbf{K}}_{\mathbf{u}} = \frac{\partial \underline{\mathbf{k}}}{\partial \underline{\mathbf{u}}} \qquad \underline{\mathbf{K}}_{\mathbf{s}} = \frac{\partial \underline{\mathbf{k}}}{\partial \underline{\mathbf{s}}}$$

$$\underline{\mathbf{H}}_{\mathbf{b}} = \frac{\partial \underline{\mathbf{h}}}{\partial \underline{\mathbf{b}}} \qquad \underline{\mathbf{H}}_{\mathbf{u}} = \frac{\partial \underline{\mathbf{h}}}{\partial \underline{\mathbf{u}}} \qquad \underline{\mathbf{H}}_{\mathbf{s}} = \frac{\partial \underline{\mathbf{h}}}{\partial \underline{\mathbf{s}}}$$

$$\underline{\mathbf{F}}_{\mathbf{b}} = \frac{\partial \underline{\mathbf{f}}}{\partial \underline{\mathbf{b}}} \qquad \underline{\mathbf{f}}_{u_d} = \frac{\partial \underline{\mathbf{f}}}{\partial u_d}.$$

In the above equation, $\underline{\mathbf{Q}}$ and $\underline{\mathbf{R}}$ are rectangular matrices which extract only the components of the sensitivity matrices pertinent to the problems being considered and $\underline{\mathbf{k}}^*$ is the same as $\underline{\mathbf{k}}$ except no boundary condition is enforced on $\partial B'$.

*3.4 Line Search Procedure*

The line search algorithm is a modification of Brent's method [17] and is outlined as follows.

(1) Determine the initial bracketing interval $\alpha \subset [\alpha_l, \alpha_u]$. The lower bound is simply $\alpha_l = 0$ which in the update formula would return the result from the previous interation. The upper bound is given based on the upper and lower bound constraints on the design parameters. Let the lower and upper bound constraints be $q_i^l$ and $q_i^u$, $i = 1, N_d$. Then the upper bound is found by

$$\alpha_u = \min\left(\alpha_i^*\right) \qquad \alpha_i^* = \max\left(\alpha_i^l, \alpha_i^u\right) \qquad i = 1, N_d$$

where $\alpha_i^l = \dfrac{q_i^l - q_i^{(r)}}{d_i^{(r)}}$ and $\alpha_i^u = \dfrac{q_i^u - q_i^{(r)}}{d_i^{(r)}}$ .

(2) Initialize intermediate function points, $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_u$.

(3) Check for convergence using the convergence criterion

$$\frac{\alpha_1 - \alpha_l}{\alpha_1} \le \varepsilon_\alpha \ .$$

If the criterion is satisfied, then take $\alpha = \alpha_1$, otherwise continue.

(4) If this is not the first iteration in the line search, evaluate two trial points, $\alpha^{tr}$, by the secant method.

$$\alpha_{j-1}^{tr} = \frac{\alpha_j \Lambda'(\alpha_j) - \alpha_1 \Lambda'(\alpha_1)}{\Lambda'(\alpha_j) - \Lambda'(\alpha_1)} - \frac{2\left[\Lambda(\alpha_j) - \Lambda(\alpha_1)\right]}{\Lambda'(\alpha_j) - \Lambda'(\alpha_1)} \qquad j = 2,3$$

where

$$\Lambda'(\alpha_i) = \underline{\mathbf{g}}(\alpha_i)^T \underline{\mathbf{d}}(\alpha_i) \ .$$

An acceptable trial point must (i) be in the descent direction measured from $\alpha_1$ and (ii) be in the interval $[\alpha_l, \alpha_u]$. If both trial points are acceptable, choose the trial point closest to $\alpha_l$. If at least one of the trial points is acceptable, continue with step (6). If this is the first iteration or if neither trial point computed by the secant method is acceptable, continue with step (5).

(5) Determine the trial point by the bisection method:

$$\alpha^{tr} = \frac{\alpha_1 + \alpha_k}{2}$$

where $k = u$ if $\Lambda'(\alpha_1) < 0$, otherwise $k = l$.

(6) If $\Lambda(\alpha^{tr}) > \Lambda(\alpha_1)$ and the convergence parameter is such that

$$\frac{\alpha_1 - \alpha_l}{\alpha_1} \le 10^2 \varepsilon_\alpha \ ,$$

then compute the trial point by parabolic interpolation:

$$\alpha^{tr} = \alpha_1 + \frac{1}{2} \frac{(\alpha_2 - \alpha_1)^2\left[\Lambda(\alpha_3) - \Lambda(\alpha_1)\right] - (\alpha_3 - \alpha_1)^2\left[\Lambda(\alpha_2) - \Lambda(\alpha_1)\right]}{(\alpha_2 - \alpha_1)\left[\Lambda(\alpha_3) - \Lambda(\alpha_1)\right] - (\alpha_3 - \alpha_1)\left[\Lambda(\alpha_2) - \Lambda(\alpha_1)\right]} \ .$$

(7) Update the bracket bounds according to the following conditions:

(a) if $\Lambda(\alpha^{tr}) > \Lambda(\alpha_1)$, then

$\alpha_u = \alpha^{tr}$ if $\alpha_1 \le \alpha^{tr}$ or

$\alpha_l = \alpha^{tr}$ if $\alpha_1 > \alpha^{tr}$

(b) if $\Lambda(\alpha^{tr}) \leq \Lambda(\alpha_1)$, then

$\alpha_l = \alpha_1$ if $\alpha_1 \leq \alpha^{tr}$ or

$\alpha_u = \alpha_1$ if $\alpha_1 > \alpha^{tr}$ .

(8) The parameters $\alpha_1$, $\alpha_2$ and $\alpha_3$ are redefined to coincide with the three smallest function evaluations thus far and such that $\Lambda(\alpha_1) \leq \Lambda(\alpha_2) \leq \Lambda(\alpha_3)$. Return to step (3).

The computational costs of the line search algorithm outlined above are that it takes two to three function evaluations in the first iteration and one to three function evaluations in subsequent iterations. The advantages of this procedure are that it ensures global convergence, it generally takes only a few iterations (< 10) to converge, and the last function and gradient evaluation are used in the minimization procedure.

*3.5 Multiplier Update Procedure*

The algorithm used to update the Lagrange multipliers, $\omega_i$ and penalty parameters, $\lambda_i^c$ is executed only when a constraint violation exists. Although the line search procedure prevents bounding constraint violations, there are usually other types of constraints as well which may be violated after an update in the minimization procedure.

The goal of the penalty parameters and Lagrange multipliers is to enforce the constraints during the minimization without causing the system to become ill-conditioned which is a frequent problem with penalty methods. The penalty parameters need to be large enough to enforce the constraint, but not so large as to cause the system to be ill-conditioned. Thus the penalty parameters are initialized to relatively moderate values and gradually increased as needed. The Lagrange multipliers act to "push" the solution in the direction of the feasible region, i. e. region where constraints are satisfied, and are either increased, decreased or left unchanged as needed. The multiplier update algorithm used here is a modified version from Powell [18] and Arora et al. [9]. The procedure can be summarized as follows:

(1) Initialize parameters in generalized penalty function which have not already been initialized in the minimization procedure: $\hat{\eta}$, $\hat{e}^{(0)}$, $a$, $\bar{a}$ and $c$. Set $\hat{\lambda}^c = 10^9 \lambda_i^{c(0)}$ ($\lambda_i^{c(0)}$ are all set to the same value initially).

(2) Compute the parameter $e^{(r')}$ according to

$$\bar{e}_i^{(r')} = \max\left(G_i, -\omega_i^{(r')}\right) \qquad i = 1, N_c$$

$$e^{(r')} = \max\left(\left|\bar{e}_i^{(r')}\right|\right) \qquad i = 1, N_c$$

(3) If $e^{(r')} < \hat{e}^{(r')}$, then update the Lagrange multipliers according to

$$\omega_i^{(r'+1)} = \omega_i^{(r')} + \bar{e}_i^{(r')}$$

otherwise

$$\omega_i^{(r'+1)} = \omega_i^{(r')} .$$

It should be noted that in this step, if the constraint is satisfied, then the Lagrange multiplier is decreased; if the constraint is violated by a margin less than $\hat{e}^{(r')}$, then the Lagrange multiplier is increased; and if the constraint is violated by a margin greater than $\hat{e}^{(r')}$, then the Lagrange multiplier is left unchanged.

(4) If $e^{(r')} < \hat{e}^{(r')}/a$, then go to step (6), otherwise check each constraint. If $\left|\bar{e}_i^{(r')}\right| < \hat{e}^{(r')}/a$, then update the penalty parameters and scale the Lagrange multipliers appropriately for constraint $i$ as follows:

$$\lambda_i^{c(r'+1)} = c\lambda_i^{c(r')} \qquad \text{and} \qquad \omega_i^{(r'+1)^*} = \frac{\omega_i^{(r'+1)}}{c}$$

and then set $\omega_i^{(r'+1)} = \omega_i^{(r'+1)^*}$.

(5) If one or more constraint violations are not improving, i. e. the solution is not moving into the feasible region, set $\hat{c}$ to a big number and exit the multiplier update routine, otherwise continue.

(6) Update $\hat{e}^{(r'+1)} = \bar{a}e^{(r')}$ and exit the multiplier update routine.

*3.6 Examination of Results*

Inverse problems are often ill-posed and therefore do not satisfy the usual conditions of existence, uniqueness, and stability. The nature of the constraints has a significant effect on the well-posedness or ill-posedness of the problem. This is true for the problems considered herein. For this reason, it is important to examine the results for a convergent solution of the problem.

An easy way to test for uniqueness and stability is to check if the solution converge to the same result with different initial guesses. While this is certainly not a proof and does not guarantee uniqueness and stability, it does provide additional confidence in the solution and may be satisfactory for engineering purpose.

For the second class of problems considered, where a specified state variable distribution is sought in the solution, it is possible to have multiple solutions. Lack of uniqueness would not be considered a problem here because any solution that satisfies the constraints and is within the tolerance is assumed to be acceptable. It may also be possible that no forming geometry satisfying the geometric constraints gives a state variable distribution within the tolerance of the desired distribution. Then it may be necessary for the designer to change the geometric constraints imposed or reexamine what state variable field is necessary.

## 4. NUMERICAL EXAMPLES AND DISCUSSION

The design for drawing, extrusion, and rolling processes are presented to demonstrate the stability and convergence of the numerical algorithm. These designs involve two problems. One design problem determines the optimum die geometry to produce the lowest input power for drawing process; the other design problem determines the optimum die geometry for drawing and extrusion and the optimum speed and process geometry for rolling to generate the state variable distribution as close as possible to the specified distribution. In the first problem, this study compares the results of optimal die profiles with the experimental and numerical data in the literature [19,20], and discusses the convergence of the solution by a simple test. The second problem focuses on explaining the ill-posedness of the inverse problem. In particular, we show the stable solution for the drawing problem, the non-unique solution for the extrusion problem, and non-existing solution for the rolling problem.

The geometry and the design parameters of the design forming problems considered herein are shown in Figure 1, where $\partial\tilde{B} = \partial B_{1y} + \partial B_{2x}$; $\partial B' = \partial B_{1x} + \partial B_{2y}$; $\partial B^f = \partial B_{2x} + \partial B_{2y}$; and $\partial\hat{B} = \partial B_{3x'} + \partial B_{1y'}$. Symmetry is assumed about the centerline at $y = 0$. The coordinates $x'$ and $y'$ are tangent and normal coordinates on $\partial\hat{B}$, respectively. For drawing and extrusion processes, on surface $\partial\tilde{B}$ the velocity is constrained to be in the horizontal direction with no friction; on surface $\partial B'$ the forming speed is applied in the horizontal direction for drawing and extrusion processes or in the tangent direction for rolling process; surfaces $\partial B^f$ are free surfaces; and on surface $\partial\hat{B}$ the normal velocity is zero and a friction condition is applied in the tangential direction. The domain of interest is divided into either a coarse mesh or a refine mesh. The coarse mesh consists of 16 8-node elements for a total of 69 nodes for the velocity field interpolation; the pressure field is interpolated with 3 node discontinuous pressure elements; and the design curve of the die profile is interpolated by 9 nodal points. The refined mesh consists of 100 8-node element for a total of 351 nodes, and 17 nodal points on the design curve of

the die profile. For simplicity, this study uses the coarse mesh to demonstrate the algorithm in most example problems; and uses the refine mesh for verification. The constitutive model presented in Brown et al. [15] is assumed, and the material in these examples is taken to be 1100 Aluminum at 450°C. The friction coefficient $\beta$ is given to be 0.1, 0.5 and 1.d0 GPa for light, medium, and high friction, respectively; and the penalty parameter for imposing incompressibility is $10^{11}$.

*4.1 Drawing problem*

A simple example of a steady metal forming process, plane strain hot drawing, is used for two design problems: shape optimization for minimizing the input power and inverse problem for producing a specified state variable distribution.

For the case involving minimizing the input power, we assume the reduction and the length of the die are fixed parameters. The nodal points at both ends of the die curve are also fixed. The shape optimization problem is posed to determine the optimal shape design parameters $\underline{\mathbf{b}}$ which results in the lowest input power required for the drawing process. Figure 2 shows the optimum die profile is in a sigmoidal shape for the drawing process. The optimal solution of $\underline{\mathbf{b}}$ is unique in this case. The unique solution can be explained by a simple test where the solution is convergent from three different initial die profiles shown in Figure 2. This result also agrees with the experimental data by Richmond [19]. The minimized input power of the drawing processed is reduced by 8.62%, 5.56%, and 10.00% from the three different initial die profiles, straight, sigmoidal and convex, respectively. The friction on contact surface between die and workpiece affects the performance of the drawing process and changes the optimal die profile shown in Figure 3. The reduction of the input power is 6.71%, 8.58% and 19.49% for the small , middle and strong friction of drawing processes, respectively.

For the inverse drawing problem, an optimum die profile is shown in Figure 4. This die profile produces the most uniform distribution of the state variable at 39 *MPa*, shown in Figure 5. We show the uniqueness of the solution by giving three different initial die profiles. This examination does not provide a proof for the convergence of the solution, but the optimum die profile obtained from the most possible solution regions should be a more confident result.

Figure 6 shows the optimal die profiles to produce the uniform state variable distribution for the drawing processes with no, light, medium and high friction. Note that for high friction the drawing process can not produce the uniform state variable at 39 *MPa*, and the optimal die profile is similar as that of the shape optimization problem in Figure 3.

*4.2 Extrusion problem*

A round-to-round extrusion process is simulated as an axisymmetric problem in Figure 1. We investigate the uniqueness of the inverse problem in this example. In the same manner, the reduction and the length of die is assumed constant in the design. Light friction is considered in the extrusion process. The die profile does not have any constraint used to fit for a smooth function. For examination in convergence, there are four different initial die profiles used to start the numerical algorithm for the solution. Figure 7 shows this extrusion process has two optimal die profiles from those initial die profiles. The two optimal die profiles generate the exact uniform distribution of the state variable at 45 *MPa*. This inverse problem is non-unique! The solution of the nonlinear inverse problem is dependent on the initial guess. The state variable distributions for those die profiles are shown in Figure 8.

*4.3 Rolling problem*

The design variables for the flat rolling problem are roll radius, reduction, and rolling speed as shown in Figure 1. For dimensionless, the reference values for the design variables are the initial values. For this rolling problem, the optimum roll radius is 6.11 cm, reduction 39.02%, and rolling speed 7.70 cm/sec. The rolling process with these parameters produces the most uniform distribution at 45 *MPa* for the state variable, as shown in Figure 8. For the linear state variable distribution in Figure 9, the optimum roll radius is 2.24 cm, reduction 72.22 %, and rolling speed 19.94 cm/sec.

## 5. CONCLUSION

Solution procedures and numerical algorithms for determining optimum process parameters in steady forming operation have been presented. This solution procedure has been found to give a convergent solution (confidently saying global) for the nonlinear inverse problem.

Two process design problems were examined. The first case involves determining the process geometry that minimized the power requirement. The second case involves determining the process parameters that generates a desired internal state variable field in the final product. These examples were used to demonstrate the effect of the process parameters on the resulting state variable field and on the power requirement, to verify the sensitivity analysis, and to demonstrate the inverse algorithm. The future work will extend the algorithm for the problem involving more complex geometries and thermal effects.

## ACKNOWLEDGMENTS

This work was supported by the Henry Luce Foundation, Inc. and by the National Science Foundation through Grant DMI-9358123.

## REFERENCES

1. J. V. Beck, B. Blackwell, and C. R. ST. Clair, Jr, *Inverse heat conduction: ill-posed problems*, John Wiley and Sons, Ltd., New York (1985).
2. J. Kusiak and E. G. Thompson, Optimization techniques for extrusion die shape design. *NUMIFORM 89*, E. G. Thompson, R. D. Wood, O. C. Zienkiewicz and A. Samuelsson, eds., A. A. Balkema, Rotterdam/Brookfield, pp. 569-574, 1989.
3. V. Legat and J. M. Marchal, Die design: an implicit formulation for the inverse problem. *International Journal for Numerical Methods in Engineering*, vol. 16, pp. 29-42, 1993.
4. C. S. Han, R. V. Grandhi and R. Srinivasan, Optimum design of forging die shapes using nonlinear finite element analysis. *AIAA Journal*, vol. 31, pp. 774-781, 1993.
5. L. Fourment and J. L. Chenot, Optimal design for non-steady-state metal forming processes — I. Shape optimization method. *International Journal for Numerical Methods in Engineering*, vol. 39, pp. 33-50, 1996.
6. L. Fourment, T. Balan and J. L. Chenot, Optimal design for non-steady-state metal forming processes — II. Application of shape optimization in forging. *International Journal for Numerical Methods in Engineering*, vol. 39, pp. 51-65, 1996.
7. S. Badrinarayanan and N. Zabaras, A sensitivity analysis for the optimal design of metal forming processes. *Computer Methods in Applied Mechanics and Engineering* vol. 129, pp. 319-348, 1996.
8. P. Michaleris, D. A. Tortorelli and C. Vidal, Analysis and optimization of weakly coupled thermoelastoplastic systems with applications to weldment design. *International Journal for Numerical Methods in Engineering*, vol. 38, pp. 1259-1285, 1995.
9. J. S. Arora, A. I. Chahande and J. K. Paeng, Multiplier methods for engineering optimization. *International Journal for Numerical Methods in Engineering*, vol. 32, pp. 1485-1525, 1991
10. .R. Fletcher, *Practical Methods of Optimization*, John Wiley and Sons, Ltd., New York (1987).
11. J. S. Arora and J. B. Cardoso, 'Variational principle for shape design sensitivity analysis', *AIAA Journal* **30**, 538-547 (1992).
12. Q. Zhang, S. Mukherjee and A. Chandra, Shape design sensitivity analysis for geometrically and materially nonlinear problems by the boundary element method. *International Journal of Solids and Structures*, vol. 29, pp. 2503-2525, 1992.
13. P. Michaleris, D. A. Tortorelli and C. A. Vidal, Tangent operators and design sensitivity formulations for transient non-linear coupled problems with applications to elastoplasticity. *International Journal for Numerical Methods in Engineering*, vol. 37, pp. 2471-2499, 1994.
14. L. E. Vaz, E. Hinton and J. Sienz, Formulations for the shape optimization considering material nonlinear behavior. *Computational Plasticity: Fundamentals and Applications*, D. R. J. Owen and E. Oñate, eds., Pineridge Press, Swansea, U.K., pp. 57-70, 1995.
15. A. M. Maniatty and M.-F. Chen, Shape sensitivity analysis for steady metal forming processes. *International Journal for Numerical Methods in Engineering*, vol. 39, pp. 1199-1217, 1996.
16. P. R. Dawson, On modeling of mechanical property changes during flat rolling of aluminum. *International Journal of Solids and Structures*, vol. 23, pp. 947-986, 1987.
17. R. P. Brent, *Algorithms for Minimization without Derivatives*, Prentice-Hall, New Jersey (1973).
18. M. J. D. Powell, 'A method for nonlinear constraints in minimization problems', in R. Fletcher Ed., *Optimization*, Academic Press, New York (1969).
19. O. Richmond, 'Theory of streamlined dies for drawing and extrusion', in F. P. J. Rimrott and J. Schwaighofer Ed., *Mechanics of the Solid State*, University of Toronto Press, Canada (1968).

20. P. A. Balaji, T. Sundararajan, and G. K. Lal, 'Viscoplastic deformation analysis and extrusion die design by FEM', *J. of Applied Mechanics* 58, 645-650 (1991).

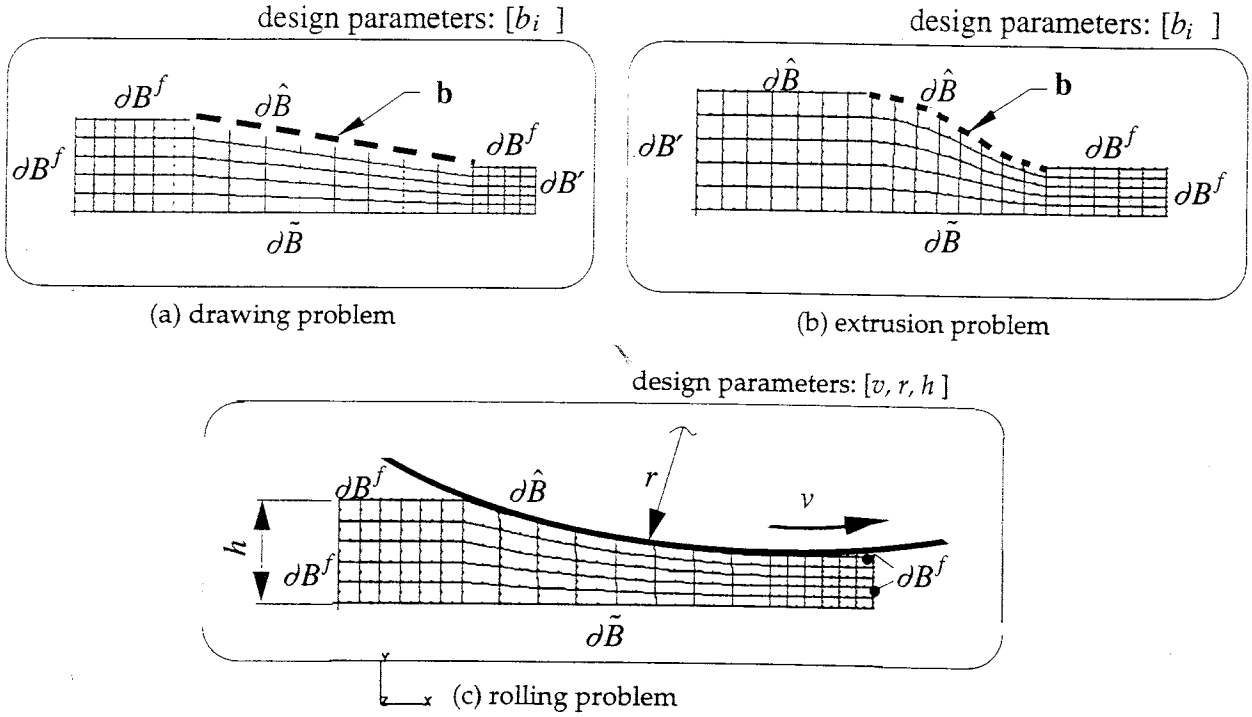


Figure 1: Geometries for process design.

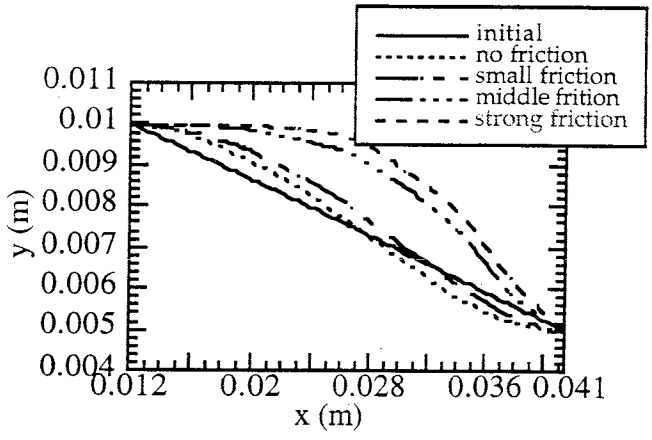


Figure 2: Optimal die profile and initial guesses for minimizing power in drawing (no friction).

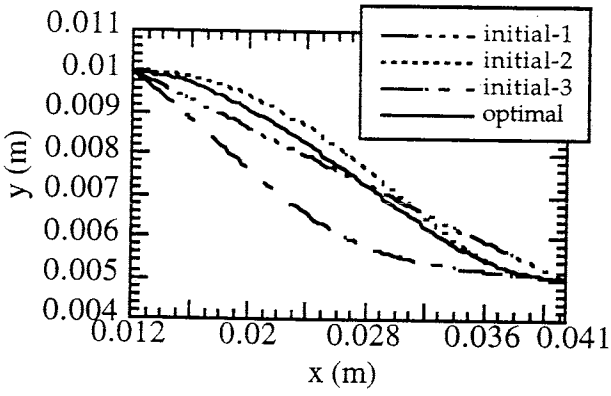


Figure 3: Optimal die profile for different levels of friction in drawing for minimizing power.

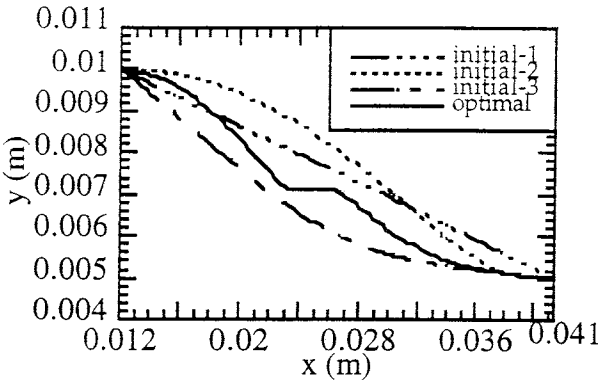


Figure 4: Optimal die profile and initial guesses for desired state variable distribution in drawing (no friction).

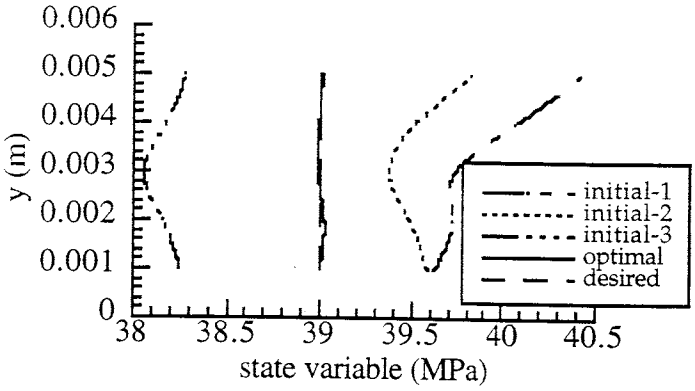


Figure 5: Resulting state variable field for dies i n Figure 4.

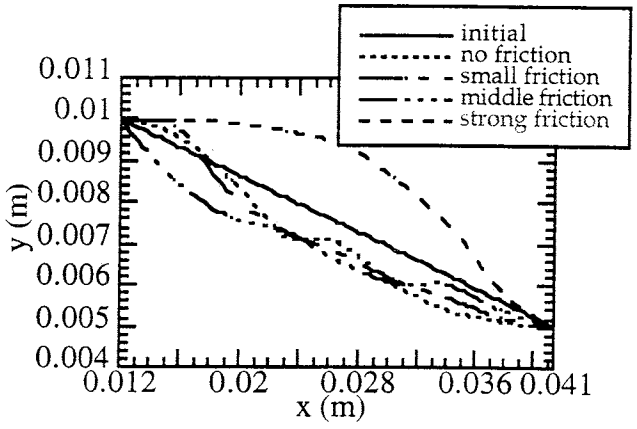


Figure 6: Effect of friction on optimal die profile in drawing.

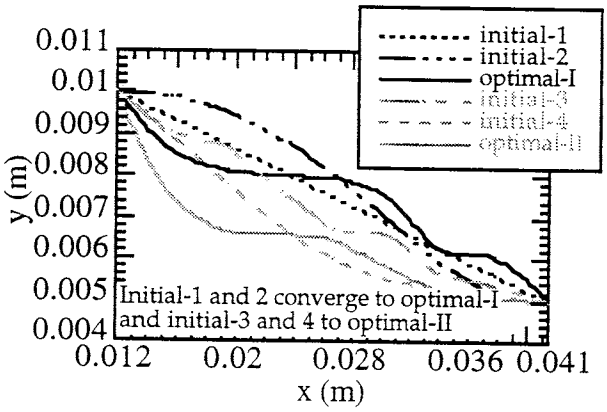


Figure 7: Two solutions for optimal die profile for extrusion problem.

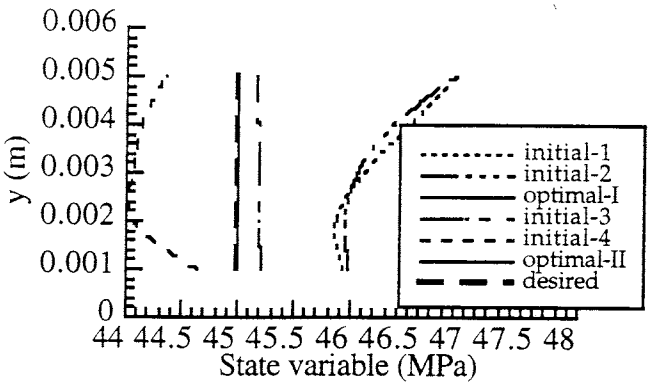


Figure 8: State variable distributions for dies shown in Figure 7.

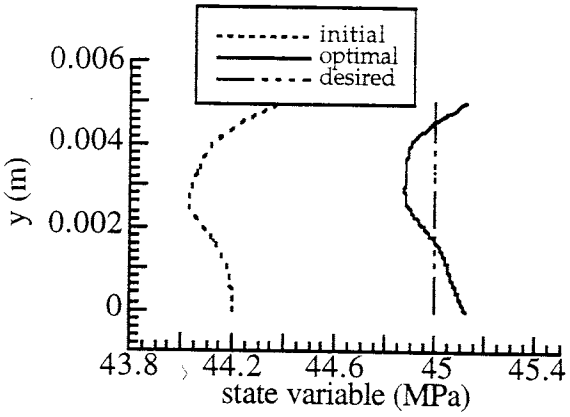


Figure 9: Optimal state variable distribution for rolling problem where uniform distribution is desired.

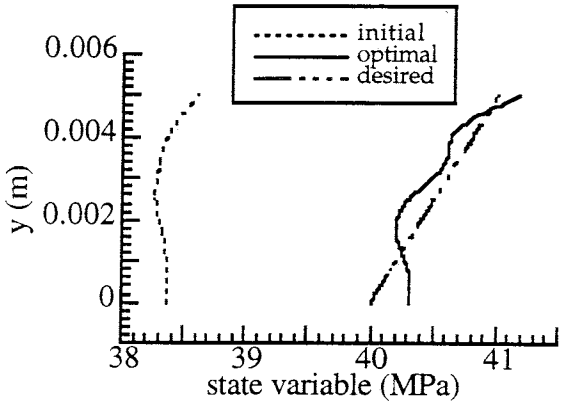


Figure 10: Optimal state variable distribution for rolling problem where linearly varying state variable distribution is desired.