Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 33 (2015) 173 – 178

www.elsevier.com/locate/procedia

9th CIRP Conference on Intelligent Computation in Manufacturing Engineering

Design of a high performance predictive tool for forging operation

Claudio Ciancio*, Teresa Citrea, Giuseppina Ambrogio, Luigi Filice, Roberto Musmanno

Dept. of Mechanical, Energy and Management Engineering, University of Calabria, Rende (87036), Italy

*Tel.: +393407950544; *e-mail address*: claudio.ciancio@unical.it

Abstract

This paper presents a comparative study of different artificial intelligence techniques to map a input-output relationship of a manufacturing process and optimize the desired responses. More in detail, these techniques were tested to model and optimize the impression die forging process. The present work aims to reduce energy, load and material consumption satisfying at the same time product quality constraints. A flywheel is considered as specific case study for the investigation. The size of the billet used in the forging process will be optimized so that the molds are correctly filled, and waste, forging load and energy absorbed by the process are minimized. The shape of the initial billet is a hollow cylinder and the parameters to be optimized are the billet dimensions (inner diameter, outer diameter and height) and the friction coefficient. The analytical relationship between input and output parameters was identified in order to choose the optimal process configuration to obtain the desired output. The input-output relation was mapped with different techniques. First of all a Genetic Algorithm-Neural Network and a Taguchi-Neural Network approach are described where genetic algorithm and Taguchi are used to optimize the neural network architecture. The other techniques are support vector regression, fuzzy logic and response surface. Finally, a support vector machine approach was used to check the final product quality.

Selection and peer-review under responsibility of the International Scientific Committee of "9th CIRP ICME Conference"

Keywords: Forging; Energy; Load; Optimization; Machine learning

1. Introduction

The forging process is a massive forming process, characterised by the application of a high compressive load which generates plastic strain of the billet [1]. This process is used to manufacture crankshafts, connecting rods, gears, turbine blades, disks and other components for mechanical industry. In this study, the attention is focused on the impression-die forging variant [2] (Fig. 1).

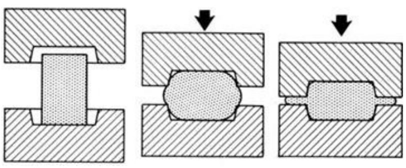

Fig. 1. Impression-die forging process

In this process the billet assumes the dies cavity shape, due to the compressive stresses on the billet during the closing phase. At the end of this phase, the dies are not mutually in contact; consequently, part of the material may flow radially outwardly of the cavity, forming flash. This flow must be hampered to guarantee the complete dies cavity filling and to obtain the desired shape. The friction at the die-workpiece interface plays a fundamental role in the final quality of the forged part. The design of the flash channel typically requires that the gap between the dies is 3% of the maximum thickness of the forged part. The flash channel length is five times larger than its thickness [3].Other important parameters are the radii of the die that must be large enough to facilitate the plastic flow of the material avoiding stress concentration that may reduce the useful life of the tools. The impression-die forging processes are aimed at the production of components with complex geometric shapes. This process still appears as one of the most applicable processing methods in the machine building industry due to the high quality of the forging parts and low production costs. Several studies have

Selection and peer-review under responsibility of the International Scientific Committee of "9th CIRP ICME Conference"

doi:10.1016/j.procir.2015.06.032

been executed to investigate and validate the Finite Element Method (FEM) capability as predictive tool for the considered process [4-5]. Starting by this result, the FEM analysis was used for the study purposes. The model is implemented by a commercial finite element code, DEFORM 2D. Naturally, implementation of periodic remesh to produce accurate plasticity solutions and good convergence is set into the model. The simulations will be axisymmetric exploiting the shape of the investigated geometry. The software package DEFORM 2D is applied here aimed at improving the process and billet design that ensure the complete dies cavity filling minimizing flash waste, energy and material consumption. In the specific case the manufactured component is a flywheel, which results an industrially interesting case.

2. Predictive tools (Input-Output problem)

To map the input-output relation of a manufacturing process several artificial intelligence techniques such as neural network, support vector regression or fuzzy logic can be used. Moreover, these approaches could be combined to obtain better performances. Neural networks have powerful pattern classification and pattern recognition capabilities [6]. However, even if the correct design of a neural network is a crucial problem it is commonly solved by trial and error procedures. To overcome this issue an automated method based on genetic algorithm or Taguchi [7] is here proposed. Support vector regression (SVR) is one of the most used regression techniques. Both SVR and SVM (support vector machine) are often used with the kernel trick but recently, in some applications, linear SVM without kernel has been shown to give good performances reducing the computational time. The neuro-fuzzy method is a rather way to create a fuzzy model from data by some kind of learning methods that is motivated by learning procedures used in neural networks. The first approach of this method was developed by Jang [8] with the adaptive network-based fuzzy *inference system (ANFIS)*. This paper is organized in the following way: in this section different techniques to map input-output relationships are described; then in the following sections an algorithm to identify the optimal process configuration and its application on the forging process are reported.

2.1 Genetic algorithm - Neural network

Neural networks are powerful tools for classification and regression problems. A neural network based on back propagation is a multilayered architecture made up of one or more hidden layers placed between the input and the output layer. The performances of a neural network are affected by many factors such as the network architecture, the type of activation function and the training algorithm. The proposed methodology uses genetic algorithm to determine the optimal value of these factors [9]. The first step of this method is to encode the features of the neural network into specific chromosomes. A chromosome is a sequence of bits with value 0 or 1. In this work we considered five input variables using chromosomes with 11 genes. The genetic algorithm, during its execution, undertakes to evolve the solution according to the following basic pattern:

1. random generation of the first population;
2. selection of the best solutions;
3. generation of a new set of solutions using crossover and mutation;
4. repetition of Step 2-3 for *n* iterations;
5. selection of the best found solution.

The selection of the chromosomes to product a new generation is a crucial step of the algorithm. The most promising chromosomes will be included in the next generation and will be used as ‘parents’ in the crossover operation. A chromosome will be selected if the value of its correspondent fitness function is low. The fitness function used in this work consists of two terms. The first term is the sum of the absolute error on the training set while the second term has the same format but it is measured on another set of data (test set) that is not used to get convergence. Both terms are then multiplied by appropriate weights. After the selection phase, a new population is generated using the crossover technique. One-point crossover is the most used and probably the simplest crossover approach. In this crossover operator two chromosomes (called parents) are chosen randomly and a binary string from the beginning of the chromosome to the crossover point is copied from the first parent whereas the rest from the second parent. Mutation is another genetic algorithm operator used to maintain genetic diversity from one generation of a population to the next. This operation consists of randomly altering the value of one element of the chromosome according to a mutation probability. Selection, crossover and mutation are repeated iteratively until the processing time exceeds a maximum time or the fitness function of the best found solution is smaller than a given threshold. Then the chromosome with the best fitness function is decoded and a neural network with those features is built.

2.2 Support Vector Regression

Support vector machines are an analytical tool that can be used for both classification and regression but their use with kernel is often time consuming [10]. In fact, classical kernel based algorithms typically have memory and computational requirements of $O(N^2)$. The basic idea behind constructing nonlinear SVR is to map the training data from the original space into a higher dimensional space called feature space, and compute an optimal linear regression function in this feature space.

Given a set of training data, a linear SVR tries to find a model, so that the quantity $w^t x_i + b$ (predicted value of the configuration *i*) is close enough to the target value y_i for all the training data. The optimal value of w^T is found by solving the following optimization model:

$$\min_{w} f(w) = \frac{1}{2} w^T w$$
$$subject\ to \quad |w^T x_i + b - y_i| \leq \epsilon$$

The assumption of this problem is the existence of a function f that approximates all pairs (x_i, y_i). However, sometimes this problem could be infeasible. Therefore in some cases it is necessary to use a different formulation of this problem that allows errors larger than ϵ:

$$\min_{w} f(w) = \frac{1}{2} w^T w + C \sum_{i=1}^{n} \xi_\epsilon(w; b; x_i; y_i)$$

where C is a penalty coefficient and $\xi_\epsilon(w; b; x_i; y_i)$ is called ϵ-intensive loss function and its value is equal to:

$\xi_\epsilon(w; b; x_i; y_i) = \max(|w^T x_i + b - y_i| - \epsilon, 0)$.

In most cases this optimization problem can be solved more easily in its dual formulation using a Kernel function [11]. The main advantage of this technique is that it can be used to solve nonlinear problems preprocessing the training data x_i by a map K into a features space $\mathfrak{F}$ $(\mathcal{X} \rightarrow \mathfrak{F})$ in which is possible to use the standard SVR algorithm. The dual formulation of the problem is reported as follows

$$\max_{\alpha,\alpha^*} f(\alpha) = \frac{1}{2}(\alpha - \alpha^*)^T Q (\alpha - \alpha^*)$$
$$+\epsilon \sum_{i=1}^{Tr} (\alpha_i + \alpha_i^*) + \sum_{i=1}^{Tr} z_i (\alpha_i - \alpha_i^*)$$
$$subject\ to\ e^T(\alpha - \alpha^*) = 0$$
$$0 \leq \alpha_i, \alpha_i^* \leq C, \qquad i = 1 \ldots .Tr$$

where $Q_{ij} = K(x_i, x_j) = \emptyset(x_i)^T \emptyset(x_j)$.

2.3 Taguchi method

The Taguchi method was developed as a process optimization technique by Genuchi Taguchi during the 1950's. The method is based on statistical analysis of data and offers a simple mean of analysis and optimization of complex systems [12]. The Taguchi method for experimental design is straightforward and easy to apply to many engineering situations and can be a powerful tool to reduce time and cost for experiments. Normally the number of experiments grows quickly with the number of process parameters. To solve this problem, the Taguchi method uses a special design of orthogonal arrays to study the entire process parameters space with only a small number of experiments. The idea of the method is to maximize the S/N ratio in that, S stands for mean and is called signal and N stands for standard deviation and is called noise. S/N ratio is a performance criteria, that is used to minimize the effect of random noise factors that could have a significant impact on the process performance. The S/N ratio is measured with the following equation:

$$S/N = -10 Log \left(\frac{\sum_{i=1}^{n} y_i^2}{n} \right)$$

This method has been used to optimize the performances searching the best architecture of the neural network. Generally the number of experiments required is equal to:

$$N = L^F$$

where F is the number of factor and L the number of levels for each factor. Five factors and three levels for each factor are considered in this work. The number of required experiments is so 3^5 to investigate all the possible solutions, however, with Taguchi it is possible to use a reduced set of 3^3experiments. For each feature of the neural network the value with the small average S/N ratio is selected.

2.4 Response surface method

Response surface methodology is a collection of mathematical and statistical techniques for empirical model building [13-14]. The goal of this method is to optimize a response (output variables) that is influenced by many independent variables (input variables). Generally the research of the response function *f* starts with a low-order polynomial function. If the output variables can be defined using a linear function of input variables *f* is a first order model. A first order model with n different input variable can be written as:

$$y = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \epsilon_i$$

where β is a vector of n unknown constant coefficients referred to the input variables, β_0 is a bias term and ϵ is a random experimental error assumed to have zero mean. The second order model can be written, instead, as:

$$y = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \sum_{j=i+1}^{n} \sum_{i=1}^{n} \beta_{ij} x_{ij} + \sum_{i=1}^{n} \beta_i x_i^2 + \epsilon_i$$

Models with a higher degree are not recommendable because they have a high probability of overfitting. An important aspect of this method is the design of experiments. The objective of this phase is to choose the input configurations to evaluate. To find a model that takes into account all the interactions between the input parameters, a full factorial research is often necessary but if the number of input variables is elevated this is not possible, so to reduce the number of experiments is carried out only a fraction of the factorial plan. The last step is to analyze the model adequacy. The most used methods are analysis of variance, regression and lack of fit test.

2.5 Neuro-Fuzzy

Fuzzy logic is a tool that uses human experience and it is based on three concepts: fuzzy sets, linguistic variables and possibility distributions [15].

A fuzzy set F in X is defined by:

$$F = \{x, \mu_F(x) | x \in X\}$$

where X is called universe of discourse and can contain both discrete and continue values. $\mu_F(x)$ is the membership function of x in F .The core of a membership function for some fuzzy set $\underline{F}$ is defined as that region of the universe that is characterized by complete and full membership in the set F. This means that the core comprises those elements x of the universe such that $\mu_F(x) = 1$. The support of membership function is defined instead as the region of the universe such that $\mu_F(x) \geq 0$, while the region of the universe containing elements that have non zero elements but not a complete membership is called boundary. A fuzzy set can be also represented by:

$$F = \begin{cases} \sum_{x_i \in X} \frac{\mu_F(x_i)}{x_i}, if\ x\ is\ discrete \\ \int \frac{\mu_F(x)}{x}, if\ x\ is\ continuous \end{cases}$$

A fuzzy set is normally divided in many subsets. A collection of fuzzy subsets is called fuzzy partition. A linguistic variable is a variable whose values are expressed in linguistic terms. Typical values are big, small or medium. One benefit of fuzzy systems is that the basic rules can be created from expert knowledge used to specify fuzzy sets to partition all variables and a sufficient number of fuzzy rules to describe the input/output relation of the problem.

In the past few years, various neuro-fuzzy systems have been developed. The neuro-fuzzy is a technique that combines the natural language description of fuzzy systems and the learning properties of a neural network [16]. Neural fuzzy systems use neural networks to produce a fuzzy system. In the training process, a neural network adjusts its weights in order to minimize the mean square error between the output of the network and the desired output. The weights of the neural network represent the parameters of the fuzzification function, the fuzzy rules and the defuzzification function.

3. Predictive tools (Output-Input problem)

The techniques described previously are able to determine the value of output variables of the problem given an input configuration; however, often this is not the real aim of the research. Given a problem with n inputs $X = \{x_1, x_2 \dots x_n\}$ and m outputs $Y = \{y_1, y_2 \dots y_m\}$, generally the goal is to:

1. find all the possible solutions $\bar{X}$ that satisfy a set of constraints:

$$\bar{X} \in \Omega: \begin{cases} X_i \leq b_i \ \ i \subseteq n \\ X_j \geq b_j \ \ j \subseteq n \\ Y_z \leq b_z \ \ z \subseteq m \\ Y_k \geq b_k \ \ k \subseteq m \end{cases}$$

2. select the best solution x^* belongs to the set of $\bar{X}$

$$x^* = \arg\{\min_{x = \bar{X}} f(x)\}$$
$$f(x) = \sum_{i=1}^{n} a_i x_i + \sum_{j=1}^{m} b_j Y_j$$

To solve the first problem the feasible region is split in two parts X^+ and X^- with a SVM tool. The SVM searches a separating hyperplane between the two classes by maximizing the margin between the classes closest points, named support vectors. When a linear separation is not possible, the data points are projected into a different space by using the kernel technique.

A new point is then classified as follows:

$$\begin{cases} X \in X^+ \ if \sum_{x_k \in S_V} y_k a_k K(x_k, X) + b \geq 0 \\ X \in X^- \ if \sum_{x_k \in S_V} y_k a_k K(x_k, X) + b < 0 \end{cases}$$

where y_k is a binary parameter with value 1 if the configuration k satisfies all the constraints, -1 otherwise, and a_k is a weight parameter. In some cases, a given configuration can be misclassified so it might be useful to calculate the probability that a point belongs to a specific class taking into account the uncertainty of the parameters. Therefore a solution x can be considered feasible only if $Pr(x \in X^+) > p$ where p is an input parameter generally set to 0.9 or 0.95. With an high value of p the risk of misclassification is reduced but the quality of the optimal solution is generally worse.

4. Computational test

This section shows the results of the previously described algorithms for a forging process. Two different tests were carried out. In the first test only the billet dimensions were considered as input variables while in the second case also the friction between the billet and the molds was taken into account. For the first test a dataset of 25 input-output pairs was used. These data were collected through numerical simulations. The volume of the raw material is increased of 1% with respect to the 3D profile. The shape of the initial billet is a hollow cylinder and the parameters to optimize are the billet dimensions: inner diameter, outer diameter and height. In the second test-set, a variation of the friction coefficient is considered. The simulations are axisymmetric exploiting the shape of the investigated geometry. The height of the flash channel is 1.8mm and its length is 6.5mm. The material considered is a steel alloy, DIN-C35. The initial temperature of the billet is 1100°C and the dies temperature is 1000°C. The temperature is constant and the punch speed is 1mm/s. Constant-shear model is applied as boundary condition and a friction coefficient equal to 0.1 is fixed, according to the range quoted by Kalpakjian for hot forging [17]. First of all, a series of tests were performed to identify the ranges of the geometrical dimensions and to define the correct workability area. The following ranges were identified for the investigate dimensions:

- internal diameter 40-80mm;
- external diameter 180-260mm

The height is calculated as a function of the previous two parameters. The proper shape volume has been determined with software Pro-Engineer. The analysis of the results show that the maximum load reached increases during the process. In most cases, maximum load occurred at the end of the process but in other cases it occurs when the punch begins to compress the billet

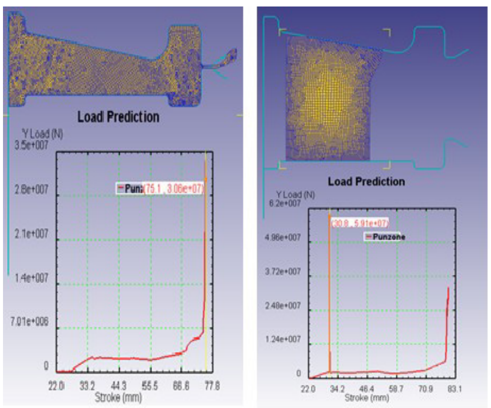

Fig. 2 a) Maximum forging load at the end of the process b) Maximum load when the punch begins to compress the billet

The dataset is split in two parts: 20 pairs are used as training set and 5 as test set. Furthermore, in order to verify the real performance of these techniques 5 input-output pairs are used as validation set. For each technique has been calculated the average percentage error for training, test and validation test.

Table 1 Test 1: Average percentage error (energy)

	GA-NN	Tag-NN	NN-Fuzzy	RSM	SVR
Training set	1.85%	4.21%	2.73%	3.20%	3.10%
Test set	2.83%	3.11%	3.25%	1.28%	1.36%
Validation set	1.23%	3.47%	1.97%	2.04%	1.67%

Table 2 Test 1: Average percentage error (load)

	GA-NN	Tag-NN	NN-Fuzzy	RSM	SVR
Training set	0.99%	1.12%	1.81%	1.99%	1.96%
Test set	1.98%	1.98%	3.28%	1.78%	1.56%
Validation set	2.12%	1.71%	0.53%	0.62%	0.28%

As shown in the tables, the best performances are obtained using the GA-NN, the SVR and the NN-Fuzzy techniques. However the average percentage error is less than 5% for each illustrated technique. Not all the input configurations are feasible according to quality constraints. The construction defects can be classified as microcrack and macrocrack, depending on the position of the billet on which they occur. The microcracks are welding defects that occur on the upper side of the material; the macrocracks are bending and welding defects that occur on the lateral side of the material when the dies are approaching. To identify the set of feasible solutions the Matlab library LIBSVM has been used. The probability to obtain a feasible solution (class 1) is high for configurations with small inner diameter and high outer diameter.

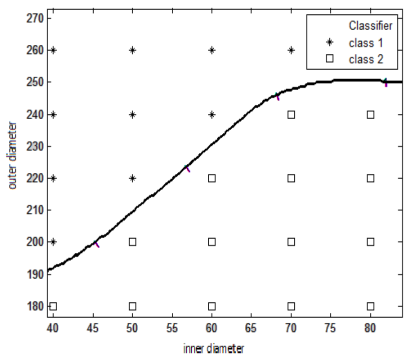

Fig. 3 Classification of the input configurations

This information is useful for the output-input problem. The aim of this problem is to identify a feasible solution that minimizes the energy absorbed by the process and the maximum load. The maximum forging load must not exceed a certain value otherwise the dies could break, while the energy must be minimal to reduce the cost and environmental impact.

Tab. 3 Test 1: Optimal process configurations

	Inner diameter	Outer diameter	Height
Min Load	59.75	260	32.84
Min Energy	72.25	249.25	36.95

To reduce the misclassification risk it is imposed that the probability that the chosen configuration is clear of defects is greater than 90%. In both cases the found solution (Tab. 3) is better than all the configurations without defects used for training test and validation. The two optimal configurations were tested to verify the accuracy of model and the presence of defects. The prediction error is less than 3% and both configurations satisfy the quality constraints. In the second test has been used a dataset of 27 input-output pairs, 21 for training and 6 for test. The tables below highlight the performance of the algorithms for different kind of data.

Table 4 Test 2: Average percentage error (load)

	GA-NN	Tag-NN	NN-Fuzzy	RSM	SVR
Training set	0.87%	0.78%	0.13%	0.28%	2.24%
Test set	1.57%	2.34%	2.24%	2.38%	2.76%
Validation set	1.80%	2.29%	2.29%	3.53%	4.67%

Table 5 Test 2: Average percentage error (energy)

	GA-NN	Tag-NN	NN-Fuzzy	RSM	SVR
Training set	1.25%	1.88%	0.27%	0.43%	2.62%
Test set	3.07%	3.94%	3.88%	4.53%	1.72%
Validation set	2.94%	3.10%	2.42%	3.47%	2.60%

In this second test the best performance are obtained using the NN-Fuzzy or the GA-NN approach. Also in this case the two configurations that minimize the output variables were found (Table 6). In this case the solution that minimizes the value of the energy absorbed by the process is one of the configurations used for the training set.

Table 6 Test 2: Optimal process configurations

	Inner diameter	Outer diameter	Height	Friction
Min Load	60	260	32.85	0.1
Min Energy	70	253.5	35.41	0.15

6. Conclusion

As expected, the maximum load reached during the process depends on the billet's size. The load in fact increases when the height of the billet is small. Despite, the energy reduces his value in the same conditions. The change in friction factor affected the maximum load in the impression die-forging process. This was because during the flash formation, which can be thought of as extrusion of metal through a constriction that circumscribes the workpiece, the dominant mechanical process is friction. All the artificial intelligence techniques described have shown good results and can therefore be a useful tool to assist business users to optimize a mechanical process.

References

[1] T. Altan, Cold and hot forging: fundamentals and applications. ASM international, 2005.

[2] G. Snape, S. Clift, A. Bramley. "Parametric sensitivity analyses for FEA of hot steel forging." Journal of materials processing technology 125 (2002): 353-360.

[3] B. I. Tomov, V. I. Gagov, R. H. Radev. "Numerical simulations of hot die forging processes using finite element method." Journal of materials processing technology 153 (2004): 352-358.

[4] H. Kim, T. Yagi, M. Yamanaka. "FE simulation as a must tool in cold/warm forging process and tool design." Journal of Materials Processing Technology 98.2 (2000): 143-149.

[5] W. Wilson, S. Schmid, J. Liu. "Advanced simulations for hot forging: heat transfer model for use with the finite element method." Journal of materials processing technology 155 (2004): 1912-1917.

[6] M. Rocha, P. Cortez, J. Neves. "Evolution of neural networks for classification and regression." Neurocomputing 70.16 (2007): 2809-2816.

[7] J. Tsai, J. Chou, T. Liu. "Tuning the structure and parameters of a neural network by using hybrid Taguchi-genetic algorithm." Neural Networks, IEEE Transactions on 17.1 (2006): 69-80.

[8] J. Jang,. "ANFIS: adaptive-network-based fuzzy inference system." Systems, Man and Cybernetics, IEEE Transactions on 23.3 (1993): 665-685.

[9] S. Sexton, R. Dorsey, D. Johnson. "Optimization of neural networks: A comparative analysis of the genetic algorithm and simulated annealing." European Journal of Operational Research 114.3 (1999): 589-601.

[10] Support Vector Regression Basics, G.Nalbantov, Patrick J.F. Groenen and Jan C. Bioch 2005

[11] Smola, B. Schölkopf. "A tutorial on support vector regression." Statistics and computing 14.3 (2004): 199-222.

[12] M. Phadke. Quality engineering using robust design. Prentice Hall PTR, 1995.

[13] N. Bradley. The response surface methodology. Diss. Indiana University South Bend, 2007.

[14] M. Giovanni. "Response surface methodology and product optimization." Food technology (1983).

[15] Zhang, Y. Wu, J. Lu, K. Du . "Evolutionary computation and its applications in neural and fuzzy systems." Applied Computational Intelligence and Soft Computing 2011 (2011): 7.

[16] Azar. "Adaptive neuro-fuzzy systems." Fuzzy systems (2010): 85-110.

[17] S. Kalpakjian. "Survey of the Feasibility of an Analytical Approach to Die Design in Closed-die Forging". No. DMIC Memo-217. Battelle Memorial Inst Colombus Ohio Defense Metals Information Center, 1966.