Inverse Problems in Engineering: Theory and Practice
3rd Int. Conference on Inverse Problems in Engineering
June 13-18, 1999, Port Ludlow, WA, USA

ID07

WEIGHTLESS REGULARISED IDENTIFICATION USING MULTI-OBJECTIVE OPTIMISATION METHOD

Tomonari Furukawa
Dept. of Quantum Engineering and Systems Science
University of Tokyo

Shinobu Yoshimura
Dept. of Quantum Engineering and Systems Science
University of Tokyo

Genki Yagawa
Dept. of Quantum Engineering and Systems Science
University of Tokyo

ABSTRACT

Although the regularisation increased the popularity of inverse analysis due to its capability of deriving a stable solution, the significant problem is that the solution depends upon the regularisation parameters chosen. This paper presents a technique for deriving solutions without the use of the parameters, and further an optimisation method, which can work efficiently for problems of concern. Numerical examples show that the technique can efficiently search for appropriate solutions.

INTRODUCTION

It often becomes difficult to solve inverse problems if measurement data are not sufficiently available and/or if measurement data and/or the direct model contains large errors [1]. One of the approaches for overcoming this problem is to introduce a regularisation term to a functional to be minimised [2,3], which normally consists of a function multiplied with weighting factors. The term makes the functional smooth, so that a conventional calculus-based optimisation can obtain an appropriate parameter set without divergence or vibration. The problem of the regularisation is however the selection of its weighting factors as the solution obtained depends upon the selection. Most of the research work thereby shows results with a couple of selections and leaves the selection for further studies.

Finding the best value of the weighting factors has not yet been much studied and can been found only in several papers to the best of the authors' knowledge [4-6]. In some techniques, the best weighting factors are found after a single solution is obtained. In this case, additional parameters are however introduced to find it, and the solution is again dependent on these parameters. The other techniques find solutions each with a different set of weighting factors by a step size before finding a single solution by some criteria, thereby the solutions not depending on the weighting factors. Deriving a number of solutions with a single optimisor is however time-consuming, and, in addition, the solutions are governed by the step size of each weighting factor.

On the other hand, multi-objective optimisation methods, which optimise a vector functional thereby giving a set of admissible solutions rather than a single solution, have been proposed, mostly by the evolutionary computation community, and have received remarkable attention [7-9]. The most popular evolutionary algorithm (EA) is the genetic algorithm (GA) [10], which incorporates binary strings and their reproduction. Despite, GA is too inefficient for the minimisation of continuous functions with continuous parameters, which is a typical inverse problem and the problem of the authors' concern.

In this paper, a technique for solving a regularised inverse problem without weighting factors is first proposed. In this technique, regularisation terms are each formulated as another objective function, and the multi-objective optimisation problem is solved by a multi-objective optimisation method. Then, a multi-objective optimisation method termed Multi-objective Continuous Evolutionary Algorithm (MCEA) specifically formulated for this class of inverse problem, is further proposed. The next section deals with the overview of the inverse analysis, and the proposed weightless regularised identification technique and the multi-objective optimisation method are presented in the third section. Numerical results showing its effectiveness and superiority to a conventional technique are dealt with in the fourth section, and the final section summarises conclusions.

INVERSE ANALYSIS

Problem formulation

Inverse analysis in industry is typically defined to identify the continuous vector, $\mathbf{x} \in R^n$, given a set of continuous experimental data. In order to solve it, an inverse problem is often converted to the minimisation of a functional:

$$f(\mathbf{x}) \rightarrow \min_{\mathbf{x}}, \tag{1}$$

where $f : R^n \rightarrow R$. The parameter set minimising such an objective function is to be found within a search space:

$$\mathbf{x}_{\min} \leq \mathbf{x} \leq \mathbf{x}_{\max}, \tag{2}$$

where $[\mathbf{x}_{\min}, \mathbf{x}_{\max}] \subseteq R^n$.

As an example for the objective function, consider the popular method of least squares, the objective function of which is represented as

$$f(\mathbf{x}) = \left\| \mathbf{K}(\hat{\mathbf{v}}(\mathbf{u}^*, \mathbf{x}) - \mathbf{v}^*) \right\|^2, \tag{3}$$

where $\mathbf{K}$ and $[u_i^*, v_i^*]$ are the weighting matrix and the set of measured data respectively, and $\hat{\mathbf{v}}$ is the computational model. It is clearly seen that the objective function consists of the model and the measured data, thus the shape of the objective function depending upon them. The difficulty of the inverse analysis is therefore that the objective function can become complex if the model and measured data contain considerable errors. It is more apparent when the number of measured data is small.

Regularisation

The complexity of the objective function in other words means that even a small change of the parameters may lead to a significant change to the functional to be minimised, and stabilisation techniques are often termed as the regularisation. In the Tikhonov regularisation [11], which is the most popular regularisation technique, the objective function is transformed into

$$\Pi(\mathbf{x}) = f(\mathbf{x}) + \alpha\Lambda(\mathbf{x}) \rightarrow \min_{\mathbf{x}}, \tag{4}$$

where α controls the total weighting factor [12]. Assume that the solution is known to be adjacent to $\mathbf{x}^*$, the regularisation term may be given by

$$\Lambda(\mathbf{x}) = \left\| \mathbf{K}(\mathbf{x} - \mathbf{x}^*) \right\|^2, \tag{5}$$

where $\mathbf{K}$ is a weighting matrix, which is normally set to the unity matrix without information. It is clear that the solution relies on the selection of α and $\mathbf{K}$.

WEIGHTLESS REGULARISATION BY MCEA

Problem formulation

The only way for finding solutions which do not depend upon the weighting factors is to remove them from the formulation, and we hereby propose a multi-objective formulation. If weighting matrix $\mathbf{K}$ is the unity matrix, Tikhonov regularisation parameter α is the only weighting factor and the objective of the problem is thus expressed as

$$\mathbf{f}(\mathbf{x})^T = \left\lfloor f(\mathbf{x}), \left\| \mathbf{x} - \mathbf{x}^* \right\|^2 \right\rfloor, \tag{6}$$

where $\mathbf{f}(\mathbf{x}) : R^n \rightarrow R^2$. If the matrix is diagonal,

$$\mathbf{f}(\mathbf{x}) = \left\lfloor f(\mathbf{x}), \left\| x_1 - x_1^* \right\|^2, \ldots, \left\| x_n - x_n^* \right\|^2 \right\rfloor, \tag{7}$$

where $\mathbf{f}(\mathbf{x}) : R^n \rightarrow R^{1+n}$. This formulation gives rise to the necessity for defining multi-objective optimisation problems and developing a method for solving such problems.

Multi-objective optimisation

While the single-objective optimisation tries to look for a single solution, multi-objective optimisation derives a set solutions, and this introduces the concept of Pareto-optimality. Consider a problem where we have m objective functions, $f_k : R^n \rightarrow R$, $k = 1, \ldots, m$:

$$\mathbf{f}(\mathbf{x})^T = \left[f_1(\mathbf{x}), \ldots, f_m(\mathbf{x}) \right] \rightarrow \min_{\mathbf{x}}. \tag{8}$$

A decision vector $\mathbf{x}_u \in R^n$ is said to be Pareto-optimal if and only if there is no vector $\mathbf{x}_v \in R^n$ for which $\mathbf{v} = \mathbf{f}(\mathbf{x}_v) = (v_1, \ldots, v_n)$ dominates $\mathbf{u} = \mathbf{f}(\mathbf{x}_u) = (u_1, \ldots, u_n)$, i.e., there is no vector $\mathbf{x}_v$ such that

$$v_i \leq u_i, \forall i \in \{1, \ldots, n\} \wedge v_i < u_i, \exists i \in \{1, \ldots, n\}. \tag{9}$$

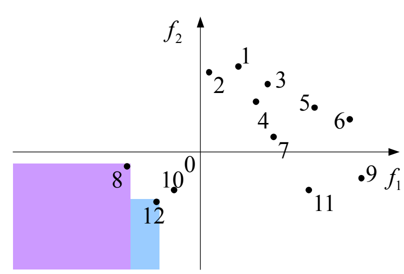

Fig. 1 Parato-optimal set

Figure 1 illustrates an example where $\mathbf{x}_8$ and $\mathbf{x}_{12}$ satisfy Eq. (10). The set of all Pareto-optimal decision vectors is called the Pareto-optimal, efficient, or admissible set of the problem. The corresponding set of objective vectors is called the non-dominated set. The Pareto-optimal can thus become the set of solutions for a multi-objective optimisation problem.

Multi-objective evolutionary algorithm for continuous problems

Capabilities of the method necessary for the multi-objective optimisation are the multi-point search method, as the multiple points can end up at a different set of solutions, and the equal evaluation to Pareto-optimal set. The characteristics of multi-point direct search thus renders evolutionary algorithms appropriate for multi-objective optimisation, provided that they can evaluate Pareto-optimal set equally.

Figure 2 shows the fundamental structure of MCEA proposed by the authors, which is efficient for problems with continuous search space. First, a population of individuals, each represented by a continuous vector, is initially (generation $t = 0$) generated at random, i.e.,

$$P^t = \{\mathbf{x}_1^t, ..., \mathbf{x}_\lambda^t\} \in (R^n)^\lambda , \tag{10}$$

where λ represents the population size of parental individuals [13]. Each vector thus represents a search point, which corresponds to the phenomenological representation of individual, unlike GAs.

```
t = 0;
Initialise P(t);
do{
        Recombinate P(t);
        Mutate P(t);
        Evaluate P(t);
        Group P(t);
        Select P(t);
        t++;
}while(condition);
```

Fig. 2 Fundamental continuous evolutionary algorithms

The definition of the recombination and mutation becomes the probabilistic distribution of the phenomenological measures accordingly. In the recombination, parental individuals breed offspring individuals by combining part of the information from the parental individuals, thereby creating new points inheriting some information from the old points. The recombination operation is then defined as

$$\begin{cases} \mathbf{x}'_\alpha = (1-\mu)\mathbf{x}_\alpha + \mu\mathbf{x}_\beta \\ \mathbf{x}'_\beta = \mu\mathbf{x}_\alpha + (1-\mu)\mathbf{x}_\beta \end{cases} , \tag{11}$$

where parameter μ may be defined by the normal distribution with mean 0 and standard deviation σ:

$$\mu = N(0, \sigma^2) \tag{12}$$

or simply a uniform distribution:

$$\mu = \mathrm{rand}(\mu_{\min}, \mu_{\max}) . \tag{13}$$

The mutation can also be achieved simply by

$$\mathbf{x}'' = \mathrm{rand}(\mathbf{x}_{\min}, \mathbf{x}_{\max}) . \tag{14}$$

Note that the mutation is not necessary for parameter μ with normal distribution since it can allow individuals to alter largely with small possibility, when the coefficient μ is large.

The grouping process of the individuals is illustrated in Fig. 3. First, all the points are concerned and the points satisfying Eq. (8) are grouped No. 1. The points in group No. 1 is then eliminated, and the points in No. 2 and later are grouped in the same fashion [8].

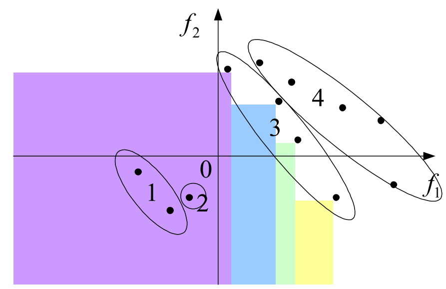

Fig. 3 Grouping of individuals. Shadowed areas represent search areas of the points in group No. 3.

With the understanding of the grouping, let the set of points in group No. k be $G(k)$ for further convenience:

$$G(k) = \{\mathbf{x}_i \mid \mathrm{group}(\mathbf{x}_i) = k, \forall i \in \{1, ..., n\}\} . \tag{15}$$

Figure 5 illustrates the evaluation of the fitness of each individual. The evaluation can be conducted with a linear scaling:

$$\Phi(\mathbf{x}_i^t) = f_{\mathrm{worst}} - f_{\mathrm{best}}(\mathbf{x}_i^t) , \tag{16}$$

where f_{worst} is given by

$$f_{\mathrm{worst}} = \sum_{j=1}^{m} f_{\mathrm{worst}j} , \tag{17}$$

$$f_{\mathrm{worst}j} = \max\{f_j(\mathbf{x}_i) \mid \forall i \in \{1, ..., n\}\}, \tag{18}$$

and

$$f_{\mathrm{best}}(\mathbf{x}_i) = \min\left\{ \sum_j f_j(\mathbf{x}_i) \mid \mathbf{x}_i \in G(k) \right\} . \tag{19}$$

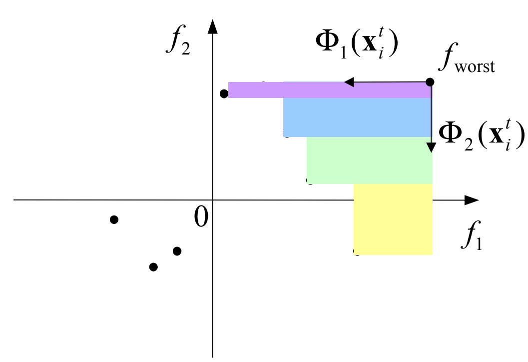

Fig. 4 Evaluation of individuals. Shadowed areas each represent search areas of the points in group No. 3.

The selection operator favourably selects individuals of higher fitness to produce more often than those of lower fitness. As $\Phi(\mathbf{x}_i^t) \geq 0$ is satisfied by this equation, the proportional selection [7], which is the most popular selection operation, can also be directly used in the proposed algorithm. In this selection, the reproduction probabilities of individuals are given by their relative fitness:

$$P_s(\mathbf{x}_i^t) = \frac{\Phi(\mathbf{x}_i^t)}{\sum_{j=1}^{\lambda} \Phi(\mathbf{x}_j^t)}. \tag{20}$$

These reproductive operations form one generation of the evolutionary process, which corresponds to one iteration in the algorithm, and the iteration is repeated until a given terminal criterion is satisfied.

System configuration

Figure 5 depicts the multi-objective optimisation system, named MCEA, developed based on the algorithms in the last subsection. The system requires three input files, optimisation file, search space file, and objective function file, and returns one output file.

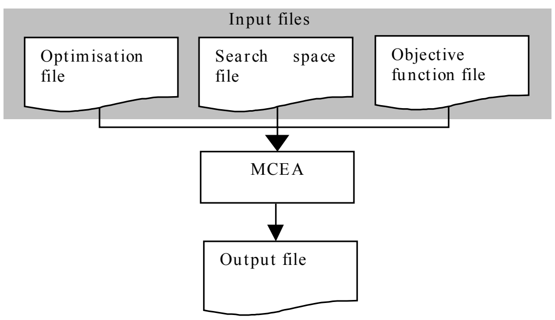

Fig. 5 Overview of MCEA

```
#include "../genetic/lib/genetic.h"

size_t geneSize = 5;
size_t perfSize = 2;

static double z[5] = {0.3, 0.4, 0.5, 0.6, 0.7};

status Target(size_t geneSize, size_t perfSize, Point *value)
{
        size_t i;

        value->performance[0] = 0.0;
        for(i = 0; i < geneSize; i++)
                value->performance[0] += value->gene[i] * value->gene[i];
        value->performance[0] /= 3.0;

        value->performance[1] = 0.0;
        for(i = 0; i < geneSize; i++)
                value->performance[1] += (value->gene[i] - z[i]) * (value->gene[i] - z[i]);
        value->performance[1] /= 3.0;

        return Ok;
}
```

Fig. 6 Function file

```
Parameters = 5
Parameter No.1
min = -5
max = 5
Parameter No.2
min = -5
max = 5
Parameter No.3
```

Fig. 7 Search space file

```
PointNum = 10
MaxHold = 500
Iteration = 500
ErrorRate = 0.02
Resolution = 0.000001
PerfResolution = 0.000001
IsFIFO = 1
RankingBestRecord = 1
Selection = p
Display = 1
DisplayFreq = 100
MaxWrite = 100
RandomSeed = 1
RankingBestRecord = 1
```

Fig. 8 Optimisation file

```
generation =   1001, best =     37 Points, left      0 Points
no    gen   :p0       :p1       :g0       :g1       :g2       :g3       :g4
  1:    72 9.49e-01 7.22e-02 3.32e-01 3.11e-01 5.38e-01 4.40e-01 5.10e-01
  2:   163 4.50e-01 2.53e-01 1.47e-01 2.32e-01 3.40e-01 2.89e-01 4.18e-01
  3:   173 9.01e-01 8.58e-02 8.29e-02 2.98e-01 4.67e-01 4.60e-01 6.13e-01
  4:   174 5.97e-01 1.88e-01 1.06e-01 1.67e-01 4.02e-01 3.93e-01 4.92e-01
  5:   213 3.50e-01 3.91e-01 5.83e-02 9.80e-02 3.41e-01 3.60e-01 3.02e-01
  6:   214 3.80e-01 3.37e-01 1.03e-01 2.07e-01 2.98e-01 2.09e-01 4.40e-01
  7:   252 5.63e-01 1.93e-01 2.83e-01 2.95e-01 3.39e-01 3.71e-01 3.79e-01
  8:   254 4.14e-01 3.31e-01 2.43e-01 6.45e-02 2.30e-01 3.38e-01 4.29e-01
  9:   301 1.60e-01 6.27e-01 1.35e-01 1.30e-01 5.47e-02 2.39e-01 2.55e-01
 10:   329 1.52e-01 6.38e-01 8.04e-02 2.09e-01 1.92e-01 2.06e-01 1.50e-01
 11:   353 1.07e-01 8.56e-01 2.16e-01 9.29e-02 1.95e-01 3.75e-02 1.13e-01
 12:   376 5.58e-01 2.02e-01 8.88e-02 2.86e-01 2.74e-01 3.56e-01 5.16e-01
 13:   377 2.48e-01 5.03e-01 1.87e-01 1.48e-01 1.27e-01 3.55e-01 2.23e-01
 14:   410 3.17e-01 3.92e-01 1.08e-01 9.01e-02 2.26e-01 2.95e-01 3.98e-01
 15:   424 9.80e-01 4.38e-02 2.30e-01 2.59e-01 4.92e-01 5.28e-01 5.83e-01
 16:   443 8.40e-01 8.60e-02 1.64e-01 4.03e-01 3.54e-01 4.12e-01 5.97e-01
 17:   452 4.90e-01 2.42e-01 2.14e-01 2.81e-01 2.57e-01 4.32e-01 3.36e-01
```

Fig. 9 Ouput file

Figures 6-8 shows examples for the input files whilst a typical output file is shown in Fig. 9. The function file has to be written with small knowledge of C language, but other input files require one to input the values mostly explained in the last subsection. The user can get therefore accustomed to them very easily.

NUMERICAL EXAMPLE

Optimisation with quadrilateral functions

First, the capability of MCEA was investigated with a simple multi-objective problem with quadrilateral objective functions

$$f_1(\mathbf{x}) = \|\mathbf{x}\|^2, \quad (21)$$

where $\mathbf{x} \in R^5$ is subject to inequality constraint (4) with $\mathbf{x}^T_{\min} = [-5,-5,-5,-5,-5]$ and $\mathbf{x}^T_{\max} = [5,5,5,5,5]$, and we assume that some information is known on its solution, and add a Tikhonov regularisation term as another objective function:

$$f_2(\mathbf{x}) = \|\mathbf{x} - \mathbf{z}\|^2 \quad (22)$$

where $\mathbf{z}^T = [0.3,0.4,0.5,0.6,0.7] \in R^5$. The problem therefore becomes to minimise functions (21) and (22). The efficient set in this problem can be determined analytically and it is given by

$$X = \{\mathbf{x} \mid \mathbf{x} = r\mathbf{z}, r \in [0,1]\} \quad (23)$$

and we can thus investigate the performance of the proposed technique with the exact set of solutions. Values of major parameters for MCEA are listed in Table 1.

Figures 10-12 show the Pareto-optimal set in $x_2 - x_4$ space at 50, 500 and 2500 generations respectively. In accordance with Eq. (23), the exact Pareto-optimal solutions are known to be on the lines shown in the figures. It is easily seen that the computed solutions at larger generations are closer to the exact solutions, and this indicates that the proposed method is appropriately finding the exact solutions. In addition, the number of computed solutions increases with respect to the number of generations as shown in Fig. 13, and this helps one to imagine the shape of the solution space.

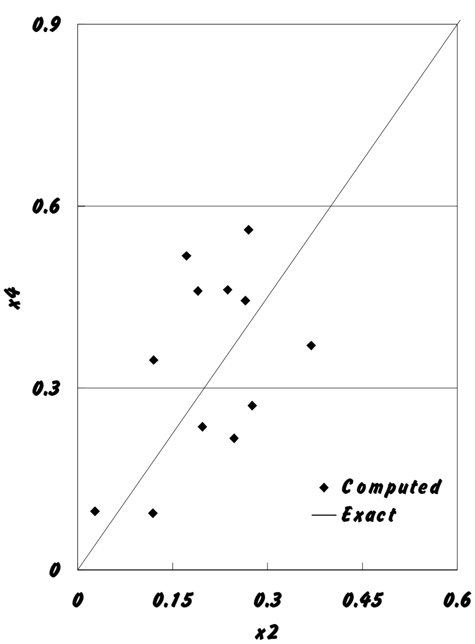

Fig. 10 Pareto-optimal set at 50th generation

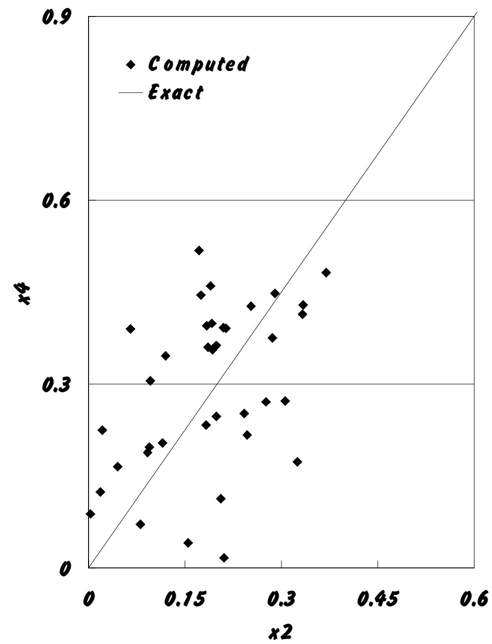

Fig. 11 Pareto-optimal set at 500th generations

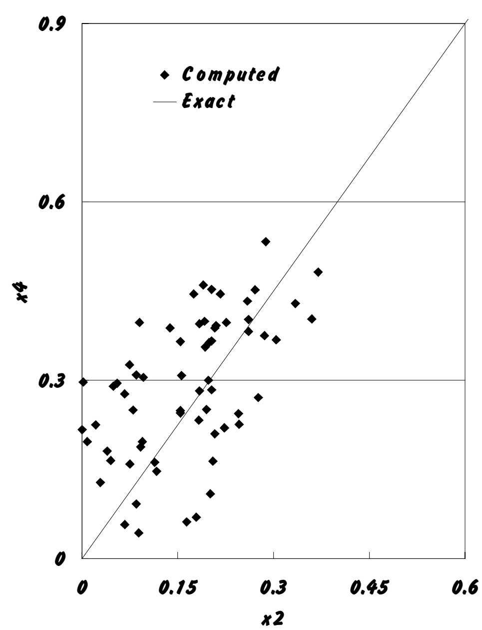

Fig. 12 Pareto-optimal set at 2500th generations

Table 1 Parameters for MCEA

Parameter	Value
No. of generations	2500
Population	10
Mutation rate	0.02

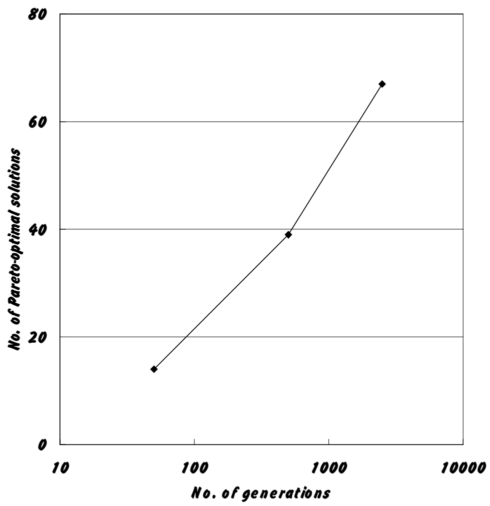

Fig. 13 No. of solutions with respect to no. of generations

Figure 14 shows the resultant Pareto-optimal solutions in function space. One can easily see that the solution space is settling down to a smooth curve with the increase of the number of generations. The final solution can be chosen subjectively from the Pareto-solutions, by considering how much the regularisation term should be taken into account.

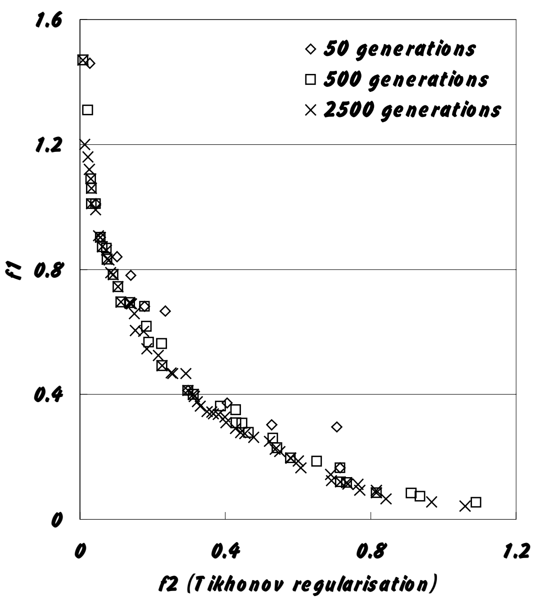

Fig. 14 Pareto-optimal set in function space

In order to investigate its efficiency of the proposed multi-objective formulation with MCEA compared to others, only objective function (22) was minimised with a single-objective optimisation method. MCEA can be used as a single-objective optimiser simply by implementing only one function in the function file, so that MCEA was used for this optimisation. All the algorithms at the programming level are therefore the same, and the direct comparison is hence possible. Note that the use of MCEA for single-objective optimisation results in Continuous Evolutionary Algorithm (CEA) proposed by Furukawa and Dissanayake [14], which was reported to be ten times faster than conventional GAs in convergence [15].

Figure 15 shows the minimal value of objective function (22) of both the multi- and single-objective optimisation. The figure clearly indicates that there is only small difference between both the optimisations. This may be caused by the fact that the individuals having the same best fitness often occupies in single-objective optimisation while multi-objective optimisation keeps variety over generations. In addition to finding the best value of objective function (22) comparable to single-objective optimisation, multi-objective optimisation searches other Pareto-optimal solutions with various states of Tikhonov regularisation, and we may conclude that the multi-objective optimisation is superior to single-objective optimisation.

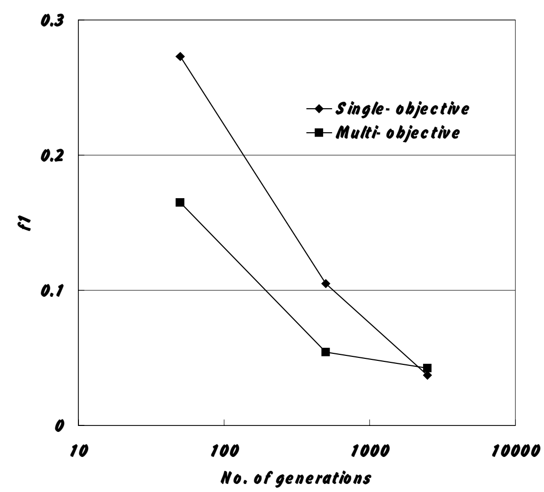

Fig. 15 Multi- and single-optimisation

Optimisation with complex function

With the understanding of the appropriate performance of the proposed technique for identification with a simple objective, the identification with a complext function, which is more realistic to engineering problems, has been investigated. The set of objective functions has an additional term to Eq. (22) and is given by

$$f_1(\mathbf{x}) = \|\mathbf{x}\|^2 + 50 - \sum_{i=1}^{5} 10\cos(\omega x_i) \tag{24}$$

Again, Eq. (23) was used as the Tikhonov regularisation term, and Table 1 as MCEA parameters.

Figure 16-18 show the resultant Pareto-optimal solutions in $x_2 - x_4$ space at 50, 500 and 2500 generations respectively. Three groups of solution are seen at 50th generations where one group consists of only one solution, and then converge to two of them. The solution space is getting clear with the increase of the number of Pareto-optimal solutions over generations.

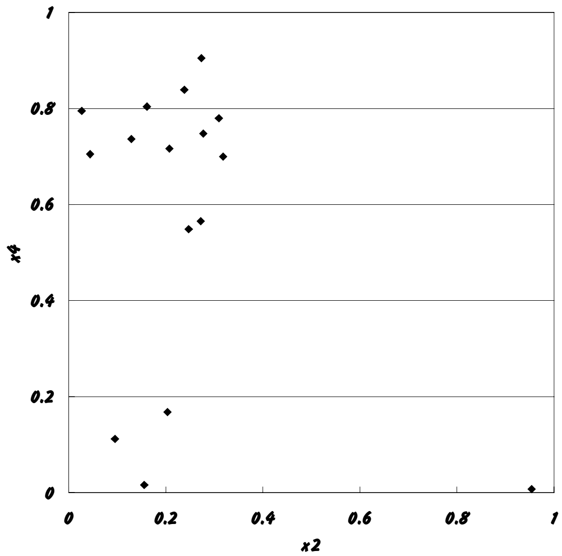

Fig. 16 Parato-optimal set at 50th generations

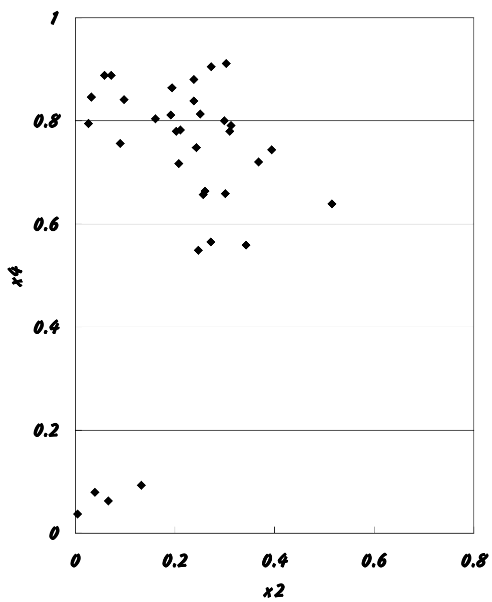

Fig. 17 Pareto-optimal set at 500th generations

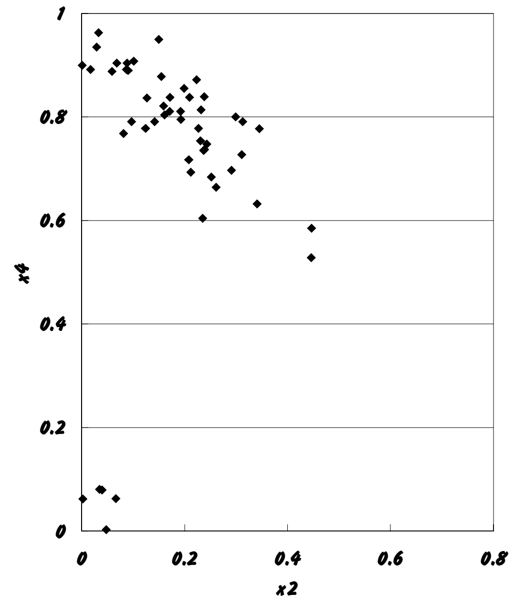

Fig. 18 Pareto-optimal set at 2500th generations

The resultant Pareto-optimal set each at 50th, 500th and 2500th generations are shown in Fig. 19. It is again seen that the function is becoming smoother as the number of generations increases.

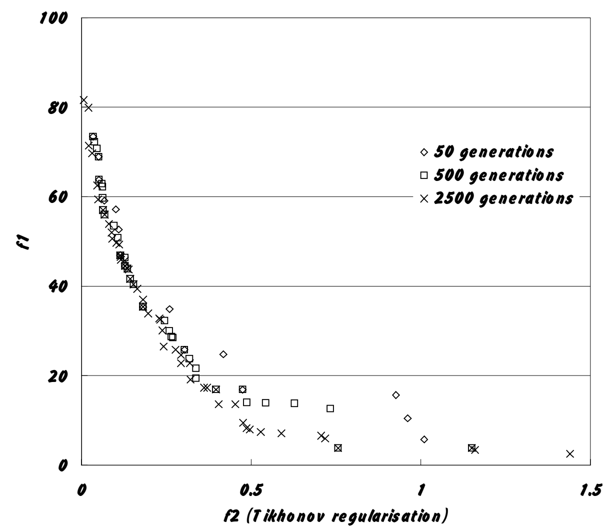

Fig. 19 Objective function values

Finally, the searching capability of MCEA was compared to that of CEA for single optimisation in the same manner, and the result of the comparison is shown in Fig. 20. There is also little difference between both the optimisations.

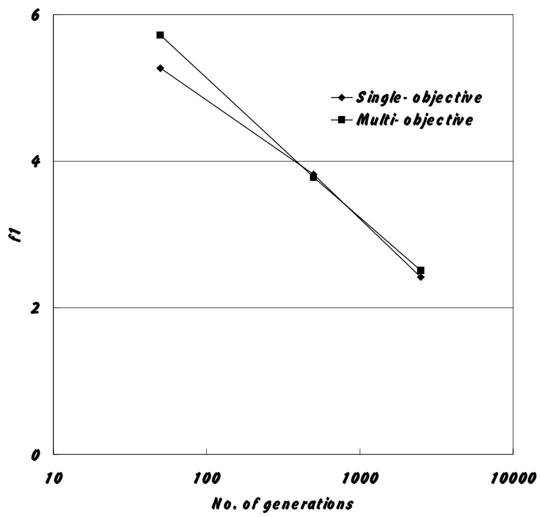

Fig. 20 Multi- and single-optimisation

CONCLUSIONS

A weightless regularised identification technique and further a multi-objective optimisation method of MCEA, which can search solutions efficiently for this class of problems have been proposed. The proposed technique was applied to two regularised identification problems as numerical examples, and the technique could find appropriate solutions in both the examples. Moreover, the searching capability of the technique was compared to the result of identification without regularisation solved by a single-objective optimisation method, and the comparison showed that a solution comparable to the solution by the single-objective optimisation was included in the set of solutions by the proposed technique. Conclusively, the effectiveness of the proposed technique has been confirmed.

Further studies include the application of the technique to actual engineering problems. The author is currently implementing the technique to the parameter identification of inelastic constitutive models [15,16]. The models contain 5-30 parameters, and its determination is above the human ability. The result of the identification will be reported in further papers.

REFERENCES

[1] Bard, Y., Nonlinear Parameter Estimation, Academic Press, New York, 1974.

[2] Baumeister, J., “Stable Solution of Inverse Problems”, Vieweg, Braunschweig, 1987.

[3] Morozov, V.A., “Methods for Solving Incorrectly Posed Problems,” Springer-Verlag, New York, 1984.

[4] Kitagawa, T., “A Numerical Method to Estimate the Optimal Regularization Parameter,” Journal of Information Processing, Vol. 11, pp. 263-270, 1988.

[5] Reginska, T., “A Regularization Parameters in Discrete Ill-posed Problems,” SIAM Journal of Scientific Computation, Vol. 17, No. 13, pp. 740-749, 1996.

[6] Kubo, S., Takahashi, T. and Ohji, K., “Characterization of the Tikhonov Regularization for Numerical Analysis of Inverse Boundary Value Problems by Using the Singular Value Decomposition,” Inverse Problems in Engineering Mechanics, Eds: Tanaka, M. and Dulikravich, G.S., pp. 337-344, 1998.

[7] Goldberg, D., “Genetic Algorithms in Search, Optimization and Machine Learning,” Addison-Wesley, Reading, MA, 1989.

[8] Fonseca, C.M. and Fleming, P.J., Genetic Algorithms for Multi-objective Optimisation: Formulation, Discussion and Generalisation,” Genetic Algorithms: Proceedings of the Fifth International Conference, Morgan Kaufmann, San Mateo, CA, Eds: Forrest, S., pp. 416-423, 1993.

[9] Fonseca, C.M. and Fleming, P.J, “An Overview of Evolutionary Algorithms in Multiobjective Optimization,” Evolutionary Computation, Vol. 3, No. 1, pp. 1-16, 1995.

[10] Holland, J.H., “Adaptation in Natural and Artificial Systems,” The University of Michigan Press, Michigan, 1975.

[11] Tikhonov, A.N. and Arsenin, V.Y., “Solutions to Ill-posed Problems,” John Willy Sons, New York, 1977.

[12] Kubo, S., “Inverse Problems,” Baifu-kan, 1993 (in Japanese).

[13] Furukawa, T. and Yagawa, G., “Parameter Identification of Inelastic Constitutive Equations Using an Evolutionary Algorithm,” Proceedings of 1995 ASME/JSME Pressure vessels and Piping Conference, PVP-Vol. 305, pp. 437-444, 1995.

[14] Furukawa, T. and Dissanayake, G., “Genetic Algorithms Using Real-valued Strings,” 71st JSME Annual Meeting, No. 930-71, pp. 509-510, 1993.

[15] Furukawa, T. and Yagawa, G., “Inelastic Constitutive Parameter Identification Using an Evolutionary Algorithm with Constitutive Individuals,” International Journal for Numerical Methods in Engineering, Vol. 40, pp. 1071-1090, 1997.

[16] Furukawa, T. and Yagawa, G., “Implicit Constitutive Modelling for Viscoplasticity Using Neural Networks,” International Journal for Numerical Methods in Engineering, Vol. 43, pp. 195-219, 1998.