Integrated genetic search for identification of material properties and dynamic excitations

Wang, Chao

2005

Wang, C. (2005). Integrated genetic search for identification of material properties and dynamic excitations. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/12085>

<https://doi.org/10.32657/10356/12085>

Nanyang Technological University

Downloaded on 26 Apr 2023 22:50:56 SGT

**INTEGRATED GENETIC SEARCH FOR
IDENTIFICATION OF MATERIAL PROPERTIES
AND DYNAMIC EXCITATIONS**

WANG CHAO

**SCHOOL OF CIVIL & ENVIRONMENTAL ENGINEERING
NANYANG TECHNOLOGICAL UNIVERSITY**

2005

**Integrated Genetic Search for
Identification of Material Properties
and Dynamic Excitations**

**Wang Chao
B. Eng., M. Eng.**

School of Civil & Environmental Engineering

**A thesis submitted to the Nanyang Technological University
in fulfillment of the requirements for the degree of
Doctor of Philosophy**

2005

This thesis is dedicated
to my parents,
and to my wife.

ACKNOWLEDGMENTS

It is difficult to use a single page to express my appreciation to all those who have contributed to the success of my research over the last three years. Herein, I would like to take this opportunity to thank those whose contributions to the research were indispensable.

First and foremost, I would like to express my sincere gratitude to my supervisor, Professor Soh Chee Kiong for his profound vision, his excellent guidance, patience and encouragement in conducting this research. His unwavering enthusiasm and interest in scientific research is much appreciated and unforgettable. It has been a great privilege and honor to work with him.

I would like to acknowledge and thank the colleagues working in the same research group for helpful discussions and suggestions. Specifically, I would like to thank Doctor Yang Yaowen and Doctor Ma Guowei for their invaluable advice, helpful ideas, constant encouragement and strong supports.

The Research Scholarship offered by Nanyang Technological University is gratefully acknowledged.

Finally, I must acknowledge the continual support and encouragement provided by my family and friends, near and distant. Most of all, I would like to express my deepest appreciation to my beloved wife, Zhang Yun, for her love, encouragement, understanding and many sacrifices, all of which made this effort possible.

TABLE OF CONTENTS

ACKNOWLEDGMENTS	ii
TABLE OF CONTENTS	iii
SUMMARY	vii
LIST OF FIGURES	ix
LIST OF TABLES	xiv
LIST OF SYMBOLS	xv
CHAPTER 1 INTRODUCTION	1
1.1 BACKGROUND	1
1.2 OBJECTIVES AND CONTRIBUTIONS.....	5
1.3 SCOPE OF WORK	7
1.4 ORGANIZATION OF THESIS	8
CHAPTER 2 GENETIC ALGORITHMS AND GENETIC PROGRAMMING	11
2.1 EVOLUTIONARY COMPUTATION METHODS	11
2.2 GENETIC ALGORITHMS.....	17
2.2.1 General Structure of GAs.....	18
2.2.2 Coding.....	19
2.2.3 Operators of GAs	20
2.2.4 Features of GAs	22
2.3 GENETIC PROGRAMMING	23
2.3.1 Preparatory Steps for GP	24
2.3.2 General Structure of GP.....	27
2.3.3 Operators of GP	29

2.3.4	Features of GP.....	33
2.4	GA AND GP APPLICATIONS IN ENGINEERING.....	34

CHAPTER 3 PARAMETER IDENTIFICATION FOR HARD ROCKS IN

	SHPB TEST	45
3.1	INTRODUCTION.....	45
3.2	BASIC THEORY OF SHPB TEST	46
3.3	SIMPLIFIED ROCK MATERIAL MODEL	50
3.4	NUMERICAL SIMULATION OF SHPB TEST.....	51
3.5	PARAMETER IDENTIFICATION.....	54
3.5.1	Parameter Identificaiton Using GA	54
3.5.2	Parameter Identification Using Non-linear Mathematical Programming	56
3.5.3	Parameter Identification Using SA	58
3.6	ILLUSTRATIVE EXAMPLES	60
3.6.1	Example 1: Conventional SHPB Configuration	60
3.6.2	Example 2: Shorter SHPB Configuration	69
3.7	CONCLUSIONS	74

CHAPTER 4 IDENTIFICATION OF DYNAMIC PROPERTIES OF

	ALUMINUM FOAMS.....	75
4.1	INTRODUCTION.....	75
4.2	DYANAMIC PROPERTIES OF ALUMINUM FOAMS	77
4.3	TRADITIONAL SHPB TECHNIQUE TO RECONSTITUTE COMPRESSION BEHAVIORS OF ALUMINUM FOAMS.....	79
4.4	GENETIC ALGORITHM (GA): CHROMOSOME.....	82
4.5	TWO MODIFIED SHPB SETUPS	83
4.6	PROBLEM DESCRIPTION AND SOLUTION PROCEDURE.....	85
4.7	ILLUSTRATIVE EXAMPLES	86
4.7.1	Modified SHPB Setup 1.....	88
4.7.2	Modified SHPB Setup 2.....	90
4.8	CONCLUSIONS	94

CHAPTER 5 FORCE IDENTIFICATION OF DYNAMIC SYSTEMS USING GENETIC PROGRAMMING..... 95

5.1	INTRODUCTION	95
5.2	PROBLEM DESCRIPTION	98
5.3	FUNCTION AND TERMINAL SETS	99
5.4	CROSSOVER AND MUTATION	100
5.5	FITNESS EVALUATION	103
5.6	SOLUTION PROCEDURE	104
5.7	ILLUSTRATIVE EXAMPLES	105
5.7.1	Example 1: Single-Degree-Of-Freedom Dynamic System.....	106
5.7.2	Example 2: Three-Degree-Of-Freedom Dynamic System	109
5.7.3	Example 3: Another Three-Degree-Of-Freedom Dynamic System	119
5.7.4	Example 4: Frame Structure	121
5.8	CONCLUSIONS	128

CHAPTER 6 GP FOR GROUND MOTION IDENTIFICATION OF DYNAMIC SYSTEMS 129

6.1	INTRODUCTION	129
6.2	PROBLEM DEFINITION	131
6.3	GP-BASED METHOD FOR GROUND MOTION IDENTIFICATION..	133
6.3.1	Overview	133
6.3.2	Initialization	134
6.3.3	Fitness Evaluation	134
6.3.4	Solution Procedure	136
6.4	ILLUSTRATIVE EXAMPLES	137
6.4.1	Example 1: Three-Degree-Of-Freedom Dynamic System	137
6.4.2	Example 2: Frame Structure	152
6.4.3	Example 3: Multi-storey Building	157
6.5	CONCLUSIONS	160

CHAPTER 7 INTEGRATED LOCAL SEARCH OPERATORS AND GP FOR	
 DYNAMIC EXCITATIONS IDENTIFICATION.....	161
7.1 INTRODUCTION	161
7.2 LOCAL SEARCH OPERATORS.....	163
7.2.1 GA.....	163
7.2.2 LBS	163
7.2.3 NLP	164
7.3 INTEGRATION OF LSOS AND GP	164
7.3.1 General Description	164
7.3.2 Encoding between LSOs and GP	168
7.4 APPLICATION TO FORCE IDENTIFICATION OF DYNAMIC	
SYSTEMS	170
7.4.1 Problem Description	170
7.4.2 Solution Procedure.....	170
7.4.3 Illustrative Examples	172
7.5 APPLICATION TO GROUND MOTION IDENTIFICATION OF	
DYNAMIC SYSTEMS	181
7.5.1 Problem Description	181
7.5.2 Solution Procedure.....	181
7.5.3 Illustrative Example	181
7.6 CONCLUSIONS	184
 CHAPTER 8 CONCLUSIONS AND FUTURE WORK	 186
8.1 SUMMARY	186
8.2 CONCLUSIONS	188
8.3 RECOMMENDATIONS AND FUTURE WORK.....	190
 REFERENCES.....	 196

SUMMARY

In the last decade, genetic algorithms (GAs) have been widely used as search techniques. Their success is attributed to their little mathematical requirements about the problems and high effectiveness at performing global search. Recent research has shown that genetic programming (GP) is powerful in various problems solving, as well as having certain advantages over GAs.

To improve the effectiveness of GP, three general search techniques, namely the inGAP, the GP+LBS and the GP+NLP, which integrate the merits of different local search operators (LSOs) (i.e., the GA, the linear bisection search (LBS) and the nonlinear programming (NLP)) and GP, are developed. In this study, GA, GP and the proposed inGAP/GP+LBS/GP+NLP methods are employed as global search techniques to solve the problems of material dynamic properties identification and dynamic excitations identification.

Although the traditional split Hopkinson pressure bar (SHPB) applications have been extended from metallic materials to various nonmetallic materials, the traditional SHPB setup and its corresponding data processing technique still have some problems when they are employed to measure the dynamic properties of very hard/soft materials. To reduce the effect of wave dispersion in the pressure bars and the size effect of the specimen during the traditional SHPB testing, a GA based method is proposed to identify the rock dynamic properties based on the traditional and modified SHPB setups. Based on two new modified SHPB setups, the GA based method is also extended to identify the entire stress-strain curve of aluminum foams.

One obvious limitation of the traditional force identification techniques is that they are unable to obtain the explicit expression of the force. Moreover, some techniques need both the displacement and velocity data of all degrees-of-freedom, and some need the Markov parameters from numerical calculation or experimental test before the force identification can be carried out. This study presents a GP based method for excitation force identification of dynamic systems to overcome these traditional methods' disadvantages. One obvious merit of the proposed method is that it can obtain the explicit expression of the unknown force. Another advantage is that it only needs the dynamic response data at one point, i.e., displacement or velocity or acceleration of one degree-of-freedom. As little agreement has been reached in the past 30 years on studies concerned with the estimation of ground motion relations, the proposed GP based method is also extended to ground motion identification.

To improve the effectiveness of the standard GP, three kinds of phenotype integration method are proposed. One main obstacle in integrating the LSOs with GP is how to encode a GP individual into an expression that can be recognized by the LSOs. In this study, an encoding strategy is presented to map between the GP individuals and the numerical string that can be recognized by the LSOs. It is hoped that these novel integration methods will have higher effectiveness and efficiency than either of the original methods for certain problems, and thus could be used as alternative global search and optimization tools for engineering problems solving.

LIST OF FIGURES

Figure No.	Description	Page
Figure 2.1	Structure of EC	15
Figure 2.2	Illustration of basic procedure of GAs	18
Figure 2.3	One-point crossover scheme.....	21
Figure 2.4	Mutation operator for binary code.....	22
Figure 2.5	A tree representation of an individual with equation $(\sqrt{2a} - b)/2a$	26
Figure 2.6	Schematical view of GP	29
Figure 2.7	Typical crossover operation for GP.....	30
Figure 2.8	Crossover operation with identical parents	31
Figure 2.9	Two different types of mutation	32
Figure 3.1	Conventional SHPB setup	47
Figure 3.2	Incident, reflected and transmitted pulses	47
Figure 3.3	Stress-strain relations with respect to different strain rates.....	51
Figure 3.4	Typical configuration of conventional SHPB test.....	52
Figure 3.5	GA-based parameter identification method.....	55
Figure 3.6	Parameter identification using Hooke and Jeeves' method.....	57
Figure 3.7	Algorithm for accepting or rejecting a change in SA.....	59
Figure 3.8	Strains at gauges a and b for conventional SHPB	61
Figure 3.9	Target curve and the upper and lower bounds curves for GA.....	62
Figure 3.10	Fitness evolution for conventional SHPB.	63
Figure 3.11	Comparison of the performance of different methods.	64
Figure 3.12	Comparison of the target and the reconstituted (by binary code GA) stress-strain curves from gauges a and b in conventional SHPB	65

Figure 3.13	Comparison of the target and the reconstituted (by real code GA) stress-strain curves from gauges a and b in conventional SHPB...	66
Figure 3.14	Strain signals at gauges a1 and b1 for conventional SHPB.....	67
Figure 3.15	The target and the reconstituted (by binary code GA) stress-strain curves from gauges a1 and b1 for conventional SHPB	68
Figure 3.16	The target and the reconstituted (by real code GA) stress-strain curves from gauges a1 and b1 for conventional SHPB	68
Figure 3.17	Configuration of modified SHPB	69
Figure 3.18	Strains at gauges a1 and b1 for modified SHPB	70
Figure 3.19	Fitness evolution for modified SHPB.....	70
Figure 3.20	The target and the reconstituted (by binary code GA) stress-strain curves for modified SHPB test	71
Figure 3.21	The target and the reconstituted (by real code GA) stress-strain curves for modified SHPB test	72
Figure 4.1	The real stress-strain curves and nonlinear fitted curves.....	79
Figure 4.2	Typical configuration of conventional SHPB test.....	80
Figure 4.3	Input and reconstituted stress-strain curves (thickness of the specimen: 4mm and 40mm).....	81
Figure 4.4	Typical GA chromosome and represented characteristic	82
Figure 4.5	Two modified SHPB setups	84
Figure 4.6	Target curve and the upper and lower bounds.....	87
Figure 4.7	Strain signals at gauges a1 and b1 for the modified SHPB setup 1	88
Figure 4.8	Fitness evolution for the modified SHPB setup 1	89
Figure 4.9	Comparison of the target and the reconstituted (by GA) stress-strain curves based on modified SHPB setup 1	90
Figure 4.10	Strain signal at gauge for the modified SHPB setup 2	91
Figure 4.11	Fitness evolution for modified SHPB setup 2	92
Figure 4.12	Comparison of the target and the reconstituted (by GA) stress-strain curves based on the modified SHPB setup 2	92
Figure 4.13	Comparison of strain history of specimen of modified SHPB setups 1 and 2.....	93
Figure 5.1	A typical GP parse tree.....	100

Figure 5.2	GP crossover operation.....	101
Figure 5.3	GP mutation operation.....	102
Figure 5.4	Solution procedure for GP-based force identification method.....	105
Figure 5.5	A single-degree-of-freedom vibration system.....	106
Figure 5.6	Fitness evolution for Example 1	107
Figure 5.7	Evolutionary process of estimated force and displacement for Example 1	109
Figure 5.8	A three-degree-of-freedom dynamic system	109
Figure 5.9	Fitness evolution for Example 2.....	112
Figure 5.10	Evolutionary process of estimated force and displacement	113
Figure 5.11	Evolutionary process of estimated force and velocity.....	114
Figure 5.12	Evolutionary process of estimated force and acceleration	115
Figure 5.13	Actual and estimated forces of Example 2	118
Figure 5.14	Actual and estimated velocities of the second freedom of Example 3	120
Figure 5.15	Actual and estimated forces of Example 3	120
Figure 5.16	The frame model.....	121
Figure 5.17	Comparison of the actual and estimated forces and displacements of point 1 of Example 4.....	124
Figure 5.18	Comparison of the actual and estimated forces and velocities of point 2 of Example 4.....	125
Figure 5.19	Comparison of the actual and estimated forces and accelerations of point 3 of Example 4.....	126
Figure 6.1	A typical GP parse tree and its represented expression.....	134
Figure 6.2	Model of three-degree-of-freedom system and ground acceleration	137
Figure 6.3	Fitness evolution for Case 1	140
Figure 6.4	Optimal GP pare tree and LISP S-expression for Case 1	140
Figure 6.5	Evolutionary process of estimated ground acceleration and displacement at point 3 for Case 1 of Example 1	142
Figure 6.6	Fitness evolution for Case 2	145

Figure 6.7	Evolutionary process of estimated ground acceleration and acceleration at point 1 for Case 2.1	146
Figure 6.8	Evolutionary process of estimated ground acceleration and velocity at point 2 for Case 2.5	147
Figure 6.9	Evolutionary process of estimated ground acceleration and displacement at point 3 for Case 2.9	148
Figure 6.10	Comparison of actual and estimated ground accelerations and system responses for Cases 2.2, 2.3, 2.4, 2.6, 2.7 and 2.8	151
Figure 6.11	The frame model.....	153
Figure 6.12	Actual and estimated displacements of measurement point 1 of Example 2	154
Figure 6.13	Actual and estimated (by measurement point 1) ground accelerations of Example 2	155
Figure 6.14	Actual and estimated velocities of measurement point 2 of Example 2.....	155
Figure 6.15	Actual and estimated (by measurement point 2) ground accelerations of Example 2	156
Figure 6.16	The 10-storey building model	157
Figure 6.17	Actual and estimated velocities of the 10th story.....	158
Figure 6.18	Actual and estimated ground accelerations of Example 3.....	159
Figure 7.1	Flowchart of integration method	166
Figure 7.2	Encoding from GP individual to LSO's numerical string	169
Figure 7.3	Solution procedure for dynamic force identification.....	171
Figure 7.4	Comparison of the actual and estimated forces in Example 1.....	175
Figure 7.5	Comparison of the actual and estimated accelerations of point 2 in Example 1	176
Figure 7.6	The 10-storey building model	177
Figure 7.7	Comparison of the actual and estimated forces in Example 2.....	179
Figure 7.8	Comparison of the actual and estimated velocities of the 9th storey in Example 2	180
Figure 7.9	Comparison of the actual and estimated ground accelerations	183

Figure 7.10	Comparison of the actual and estimated velocities of the 10th storey	184
Figure 8.1	Flowchart of alternate inGAP method.....	193
Figure 8.2	A typical parse tree for genotype integration of GA and GP	194

LIST OF TABLES

Table No.	Description	Page
Table 2.1	Main differences between GAs and ESs.....	14
Table 3.1	Parameters used in the GA.....	56
Table 3.2	Comparison of target and reconstituted coefficients (by binary code GA) of hard rock.....	73
Table 3.3	Comparison of target and reconstituted coefficients (by real code GA) of hard rock.....	73
Table 5.1	Cases studied in Example 2	110
Table 5.2	The actual forces	122
Table 6.1	Tableau for GP parameters	138
Table 6.2	Sub-cases studied in Case 2	143
Table 7.1	Control parameters for GP and GA	173
Table 7.2	Comparison of standard GP and integrated GPs for dynamic force identification for Example 1	174
Table 7.3	Comparison of standard GP and integrated GPs for dynamic force identification for Example 2	178
Table 7.4	Comparison of standard GP and integrated GPs for ground motion identification	182

LIST OF SYMBOLS

A	=	the parameter for the Lorentz curve
A_0	=	the area of the specimen
A_p	=	the cross section area of the pressure bar
C	=	the constant defined by the user for a specific problem
C_0	=	the longitudinal elastic wave speed of the bar material
D	=	the isotropic damage scalar
E	=	the elastic modulus of a bar material
E_0	=	the elastic modulus of the rock material
f	=	the interface force
F_r	=	the fitness function
F	=	the function set
Gen	=	the number of the current generation
$gene[m]$	=	the binary value
i	=	the index referring to an individual in the population
l	=	the penetration value
L_0	=	the length of the specimen
M	=	the population size
P_1	=	the force acting on the interface between the specimen and the input bar
P_2	=	the force acting on the interface between the specimen and the output bar
r	=	the aspect ratio
R	=	the random number
$round\{\}$	=	the function to round the value to the nearest integer
T	=	the terminal set

u	=	the particle displacement aligned with the wave propagation
v	=	the floating point-valued vectors
V	=	the impact velocity of the striker
V_1	=	the particle velocities at the end of the input bar
V_2	=	the particle velocities at the end of the output bar
W	=	the error function
w	=	the parameter for the Lorentz curve
x	=	the point in the solution space
σ	=	the corresponding standard deviation
$N(0, s)$	=	the vector of independent identically normally distributed random numbers with a mean zero and standard deviation σ .
x_i	=	the i th parameter
$\bar{x}_i$	=	the normalized i th parameter;
x_i^L	=	the minimum value of the i th parameter
x_i^U	=	the maximum value of the i th parameters
N_{gene}	=	the number of bits in the gene
x_i^{qn}	=	the quantized value of $\bar{x}_i$
x_i^q	=	the quantized value of x_i
$\varepsilon_I(t)$	=	the input stress wave
$\varepsilon_R(t)$	=	the reflected stress wave
$\varepsilon_T(t)$	=	the transmitted stress wave
ρ	=	the density of the bar material
σ_y	=	the uniaxial yield stress of the bar material
ε_0	=	the threshold strain
α	=	an independent material coefficient
β	=	another independent material coefficient
ρ_r	=	the density of the rock specimen
σ_r	=	the normal stress in the rock specimen
k_f	=	the stiffness factor

W'	=	the revised fitness function
$\varepsilon_I^m(t)$	=	the measured strain signals at the incident bar
$\varepsilon_I^e(t)$	=	the estimated strain signals at the incident bar
$\varepsilon_O^m(t)$	=	the measured strain signals at the transmitted bar
$\varepsilon_O^e(t)$	=	the estimated strain signals at the transmitted bar
ΔT	=	the measuring duration of a classical SHPB set-up
Δl	=	the total relative displacement between the two bar/specimen interfaces
$\varepsilon_{\max}$	=	the maximum measurable strain
$\dot{\varepsilon}$	=	the given average strain rate
E_{0a}	=	the elastic modulus of aluminum foam
ε_{0a}	=	the maximum elastic strain of aluminum foam
ε_c	=	a parameter for the Lorentz curve
σ_{0a}	=	another parameter for the Lorentz curve
$[M]$	=	the mass matrix
$[C]$	=	the damping matrix
$[K]$	=	the stiffness matrix
$\{f(t)\}$	=	the input force vector
$\{\ddot{x}(t)\}$	=	the vector of acceleration
$\{\dot{x}(t)\}$	=	the vector of velocity
$\{x(t)\}$	=	the vector of displacement
$\{x_f^m(t)\}$	=	the measured displacement of freedom f
$\{x_f^e(t)\}$	=	the estimated displacement of freedom f
$\{\dot{x}_f^m(t)\}$	=	the measured velocity of freedom f
$\{\dot{x}_f^e(t)\}$	=	the estimated velocity of freedom f
$\{\ddot{x}_f^m(t)\}$	=	the measured acceleration of freedom f
$\{\ddot{x}_f^e(t)\}$	=	the estimated acceleration of freedom f

α_d	=	the coefficient of damping model
β_d	=	the coefficient of damping model
$\{I\}$	=	the identity vector.
$a(t)$	=	the ground acceleration
$x_f^m(t_i)$	=	the actual displacement response of freedom f at time t_i
$x_f^e(t_i)$	=	the estimated displacement response of freedom f at time t_i
$\dot{x}_f^m(t_i)$	=	the actual velocity response of freedom f at time t_i
$\dot{x}_f^e(t_i)$	=	the estimated velocity response of freedom f at time t_i
$\ddot{x}_f^m(t_i)$	=	the actual acceleration response of freedom f at time t_i
$\ddot{x}_f^e(t_i)$	=	the estimated acceleration response of freedom f at time t_i

CHAPTER 1 INTRODUCTION

1.1 BACKGROUND

Darwin was the first person who proposed a creative, coherent and tenable alternative evolutionary theory (Darwin 1859). Specifically, his theory of evolution states that:

- (1) Organisms reproduce and increase in number, given infinite resources;
- (2) Competition for the finite resources is inevitable. Individuals that compete effectively will survive and reproduce while those that do not will perish;
- (3) A reproducing organism passes many of its characteristics onto its offspring;
- (4) These characteristics are not always copied perfectly. As a result, variations are observed in the population; and
- (5) These variations may lead to further differences in ability to compete for the finite resources.

Evolutionary adaptation will occur in any system exhibiting these properties (Maynard-Smith 1994). The idea of applying these principles to artificial search and optimization algorithms is not new. In 1958, Friedberg (Friedberg 1958; Friedberg et al. 1959) developed the first example of what is now known as an evolutionary computation (EC) method (Baeck and Schwefel 1996; Baeck et al. 1997).

Subsequent variations included evolution strategies (ESs) (Rechenberg 1994; Schwefel 1995), evolutionary programming (EP) (Fogel et al. 1966; Fogel 1995) and classifier systems (Holland and Reitman 1978; Wilson 1995). Two further EC methods of particular interest are the genetic algorithms (GAs) (Holland 1975; Goldberg 1989) and genetic programming (GP) (Koza 1992, 1994; Koza et al. 1998). These algorithms all incorporate the following abstractions of Darwinian evolution:

- (1) A random initial population of candidate solutions;
- (2) Reproduction with variation to produce new solutions;
- (3) Evaluation of individuals (candidate solutions) against a cost function (fitness function); and
- (4) Probabilistic selection for survival and future reproduction based on fitness.

GAs are general-purpose search algorithms that use the principles inspired by natural population genetics to evolve solutions to problems (Holland 1975). The essence of this method is that by maintaining a population of knowledge structures that represent the candidate solutions to the problem of interest, the population evolves over time through a process of competition (i.e., survival of the fittest) and controlled variation (i.e., reproduction, crossover and mutation). This innovative technique has been demonstrated to be effective and robust in a variety of optimization problems (Goldberg 1989; Davis 1991; Soh and Yang 1996; Yang and Soh 1997; Gen and Cheng 1997; Liu and Han 2003; Wang et al. 2004).

GP has been empirically shown to be a powerful program-induction methodology and has been successfully applied to a large number of difficult problems like structural optimization and design, process modeling, auto control, signal processing, hydrological modeling, ecological modeling, pattern recognition, and data mining (Spencer 1994; Sharman et al. 1995; Sen and Stoffa 1996; Shao and Yoshisada 1996; McKay et al. 1997; Gray et al. 1998; Babovic 2000; Cao et al. 2000; Coello 2000; Soh and Yang 2000; Yang and Soh 2000; Brameier and

Banzhaf 2001; Khu et al. 2001; Ross et al. 2001; Wigham and Recknagel 2001). Although GP is an extension of GAs, it is in some sense more powerful than GAs. Compared to GAs' one-dimensional encoding of individuals, GP's encoding of two-dimensional parse-trees varying in size and complexity is more natural and flexible in representing the candidate solutions. When solving a problem for which we do not know in advance the size and model of its best solution, the ability of GP to examine different size solutions is very important. Generally, GP is good at global search for the model of the solution of the given problem, however, for the problems that need "fine-tuning" of the solution parameters, GP is not as good as GAs (Yang and Soh 2002). These inspire the attempt to integrate GA with GP. It is hoped that this novel integration may have higher effectiveness and efficiency than either of the original methods for certain problems, and thus could be used as an alternative global search and optimization tool for engineering problem solving. In this kind of integration, GA is adopted as a local search operator (LSO). This inspires the attempt to integrate other two LSOs (i.e., local linear bisection search (LBS) and nonlinear programming (NLP)) with GP.

It is well known that the mechanical properties of most engineering materials are affected by their strain rates, which may further influence the structural behaviors when the structure is subjected to rapidly changing loads, such as impact and blast. Thus, it is necessary to take the strain rate effect on material constitutive equations into account when impact loading is involved in structural design. There are several experimental methods to measure the material dynamic properties, depending on the strain rate range (Bai and Dodd 1992; Meyers 1994). Among these experimental methods, split Hopkinson pressure bar (SHPB), or Kolsky's apparatus are very popular test methods for strain rate in the range of 10^1 - 10^3 /s.

Historically, Hopkinson (1914) was the first to use a long thin bar to measure the pulse shape induced by an impact. This method was further established by the critical work of Davies (1948). The experimental setup with two long bars and a short specimen was introduced by Kolsky (1949a, 1963). The SHPB

technique, which has been initially used in compression, has been extended to tension (Harding et al. 1960), torsion (Duffy et al. 1971), simultaneous torsion compression (Lewis and Goldsmith 1973), and simultaneous compression torsion (Chichili and Ramesh 1999). Its applications have also been extended from metallic materials to various nonmetallic materials, including polymer, foam, wood, concrete, rock, rubber, ceramic and composite (Zhao 1998; Li et al. 2000; Gary and Zhao 2000; Svante 2002; Meng 2002; Lee 2003). However, the traditional SHPB setup and its corresponding data processing technique still have some problems when they are adopted to measure the dynamic properties of very hard materials (e.g. hard rocks) or very soft materials (e.g. aluminum foams). To overcome these difficulties, GA will be adopted as the data processing method to identify the dynamic properties of these special materials in this study.

The identification of dynamic forces acting on a structure is an old problem, but has only been treated with partial success (Stelzner et al. 2001). It is very important for dynamic design, analysis and evaluation of structures, especially for structures with low reliability. Nevertheless, direct measurements of the excitation forces are not feasible in most actual physical and mechanical systems. Instead, vibration responses can often be conveniently measured. An indirect estimation of excitation forces with some accessible dynamic responses is frequently employed to solve the problem. With the process of determining the applied forces from the available responses, the related force sensors are not necessary. Finding an effective, practicable and easy to use method to solve this problem becomes more and more important. In this study, a GP-based method will be developed to identify the unknown excitation forces.

The identification of ground motions from measured structural responses is an important and typical type of inverse problem in structural mechanics. In certain situations, the ground motion time history may not be recorded or retrieved. But if the structural responses are available, how to reconstruct the unknown ground motions from the known dynamic responses becomes the critical issue. It is found that little agreement has been reached in the past 30 years on ground motion

estimation relation studies (Douglas 2003). So, how to solve this problem effectively becomes one of the objectives of this research.

1.2 OBJECTIVES AND CONTRIBUTIONS

The main objective of this study is to develop new general approaches based on GA and GP as backcalculation algorithms for solving identification problems. In order to overcome the inherent disadvantages of the traditional SHPB setup and its corresponding data processing technique, the GA-based approach will be employed to identify the key parameters of the dynamic properties of hard rocks and aluminum foams. Due to its effectiveness and efficiency, the GP-based backcalculation method will be adopted to identify the excitation forces and ground motions of the dynamic systems.

Another objective of this study is to improve the effectiveness and efficiency of the standard GP. Three general search techniques, namely the inGAP, the GP+LBS and the GP+NLP, which integrate the merits of different LSOs (i.e., GA, LBS and NLP) and GP, are developed. The applicability and effectiveness of the proposed inGAP/GP+LBS/GP+NLP methods will be shown by their applications to the identification of dynamic excitations.

In this thesis, based on the traditional SHPB setup and proposed modified SHPB setups, the key parameters of hard rocks and aluminum foams are identified by using the GA-based backcalculation algorithm. The GP-based identification methods are employed to the identification of excitation forces and ground motions of dynamic systems. The inGAP/GP+LBS/GP+NLP methods are developed and applied to the identification of dynamic excitations. Numerical results are also used to verify the effectiveness and efficiency of these proposed methods.

The originalities and major contributions of the present work are:

- (1) A parameter identification technique for identifying the rock dynamic properties is presented by integrating GA into the traditional SHPB and modified SHPB setups. The proposed method can be extended for testing of other brittle materials.
- (2) GA is adopted as a backcalculation algorithm for identifying the dynamic properties of aluminum foams. In order to obtain the entire stress-strain curve of aluminum foams, including the densification property, which can not be obtained from the traditional SHPB setup and its corresponding data processing procedure, two new modified SHPB setups are proposed. The identification method based on these two new modified SHPB setups and GA is able to obtain the entire stress-strain curve of aluminum foams.
- (3) The GP-based identification method is proposed to identify the excitation forces of dynamic systems. The excitation force can be obtained from any known dynamic response, such as displacement, velocity or acceleration of a dynamic system.
- (4) How to reconstitute the ground motion acceleration history from the known dynamic response of the system is a big problem which has to be faced at some situations. The GP search method is extended to solve this problem. Numerical simulations are used to verify the practicality and effectiveness of the GP search method for ground motion identification.
- (5) The inGAP/GP+LBS/GP+NLP methods are developed to improve the effectiveness and efficiency of the standard GP. The identification methods based on these developed methods are used to identify the dynamic excitations of dynamic systems. The results demonstrate that

these developed integration methods have higher effectiveness and efficiency than the standard GP.

1.3 SCOPE OF WORK

In this work, GA, GP, the proposed inGAP/GP+LBS/GP+NLP methods are employed as global search techniques to solve different inverse problems. Firstly, to illustrate its feasibility as backcalculation algorithm for parameter identification, GA is employed to identify the critical parameters of hard rocks based on the traditional SHPB setup and the modified SHPB setups. In order to obtain the entire compression behavior of aluminum foams, GA is integrated into the two modified SHPB setups to identify the critical parameters. These GA-based identification methods can be extended to identify the dynamic properties of other materials. GP is then integrated with the finite element analysis to search for the excitation forces and ground motions from the dynamic responses of dynamic systems. Finally, the inGAP/GP+LBS/GP+NLP methods, which integrate the merits of GA/LBS/NLP and GP, are used for the identification of dynamic excitations. Since the proposed integration methods are general search techniques, they have the potential to be conveniently applied to other types of identification and optimization problems.

To validate the correctness of the proposed methods, experimental validation would be the best way. However, due to time constraint and resource limitation, all methods in this thesis are validated by numerical simulations. After the dynamic response is obtained by numerical simulation for the dynamic systems under certain excitations, the proposed method is used to back-calculate the excitations. If the back-calculated results match the numerical simulation results, the proposed methods are considered to be validated. The advantage of using numerical validation is that the accuracy and efficiency of the proposed methods can be better demonstrated.

The scope of this project includes:

- (1) Integration of GA into the traditional SHPB setup and the modified SHPB setup to identify the dynamic properties of the hard rocks;
- (2) Proposal of two new modified SHPB setups and using GA as the backcalculation method to reconstitute the entire stress-strain curve of aluminum foams which can not be obtained from the traditional SHPB setup and its corresponding data processing procedure;
- (3) Implementation of GP algorithm for force identification from the measured responses of dynamic systems;
- (4) Extension of the GP-based force identification method to ground motion identification;
- (5) Development of the inGAP/GP+LBS/GP+NLP methods to improve the search effectiveness and efficiency of the standard GP; and
- (6) Illustration of the process and the efficiency of these proposed integration methods by applying them to the identification of dynamic excitations.

1.4 ORGANIZATION OF THESIS

This thesis consists of eight chapters.

Chapter 1 gives a brief introduction of the background, objectives, originality, contributions and scope of this study.

In Chapter 2, a brief introduction of EC is presented, followed by the general descriptions of GA and GP. A review of the applications of GA and GP in civil engineering and other areas is also included in this chapter.

Chapter 3 describes the development of a GA-based method to identify rock dynamic properties in the SHPB test. The rock material properties can be accurately identified by adopting numerical simulation of the test and applying GA for parameter optimization. Two modified SHPB configurations are suggested to reduce the effects of wave dispersion and the specimen length. Based on the modified configurations, the strain gauges can be freely cemented on the incident and the transmission bars. In addition, the data collected from only a single strain gauge at the incident/transmission bar is sufficient for the identification of the rock dynamic properties.

Usually, the densification strain of aluminum foams is from 50% to 80%, and the real specimen for the aluminum foam should be 30-50mm in length to include a few cells in the longitudinal direction, so the traditional SHPB setup and its corresponding data processing method are invalid. In Chapter 4, two new modified SHPB setups are proposed to solve this problem. The numerical simulation results show that after GA is integrated into these two new modified SHPB setups, the entire stress-strain curve of aluminum foams can be reconstituted correctly.

One obvious limitation of the traditional force identification techniques is that they are unable to obtain the explicit expression of the excitation force. Moreover, some techniques need both the displacement and velocity data of all degrees-of-freedom, and some need the Markov parameters from numerical calculation or experimental test before the excitation force can be identified. Chapter 5 presents a GP-based method for excitation force identification of dynamic systems to overcome these traditional methods' disadvantages. GP is employed as a search and optimization method to obtain the optimal, if not the best, force expression from the known dynamic responses.

In Chapter 6, the GP method for ground motion identification of dynamic systems is presented. GP is employed as a search and optimization method to obtain the optimal ground motion expression from the known dynamic responses. The

numerical simulations show that this method can identify the unknown ground motion history correctly, based on any response of any degree-of-freedom of the dynamic system.

Chapter 7 proposes three kinds of phenotype integration method to improve the effectiveness and efficiency of the standard GP. These integration techniques incorporate GA, LBS and NLP as LSOs of GP, respectively. One main obstacle in integrating the LSOs with GP is how to encode a GP individual into an expression that can be recognized by the LSOs. The key of the integration is how to establish the effective encoding approach. An encoding strategy is presented to map between GP individuals and the numerical string that can be recognized by the LSOs, and the encoding approach is illustrated by a simple example. These proposed integration methods are applied to identify the dynamic excitations of dynamic systems. The numerical results show that local learning can be efficiently included into GP and local search by using GA, LBS or NLP can significantly enhance the performance of GP.

Finally, Chapter 8 summarizes the present work and discusses the conclusions and the areas for future work.

CHAPTER 2 GENETIC ALGORITHMS AND GENETIC PROGRAMMING

2.1 EVOLUTIONARY COMPUTATION METHODS

Evolutionary computation (EC) incorporates algorithms that are inspired by the evolution principles in nature (Fogel 1997). EC can be applied to problems where the traditional methods are difficult to apply (e.g., gradients are not available) or lead to unsatisfactory solutions (i.e., local optima). Essentially, EC is stochastic in nature and its search imitates and models two primary natural phenomena: i) the survival of the fittest and ii) the genetic inheritance.

For over three decades, the basic principles of evolution have been applied to the solution of problems in a variety of domains. This work was begun independently by Rechenberg (1964), Fogel et al. (1966), and Holland (1975). Till now, there are four mainstream methods in the field of EC. They are Genetic Algorithms (GAs), developed by Holland (1975) and with broad extension contributed by Goldberg (1989), Genetic Programming (GP), developed by Koza (1992) and sometimes considered as an extension of GAs, Evolution Strategies (ESs), developed by Rechenberg (1973) and Schwefel (1981), and Evolutionary Programming (EP), initiated by Fogel (1962) and brought to the modern type by Fogel (1992). Although there are many close similarities between these evolutionary computing paradigms, there are also profound differences between them (De Jong 2001). These differences generally concern the level in the hierarchy

Chapter 2: Genetic Algorithms and Genetic Programming

of evolution being modeled: the chromosome, the individual, or the species (Fogel 1997). There are also many hybrid systems which incorporate various features of the above paradigms and consequently are hard to classify; but, they can all be generally referred to as EC methods.

Genetic Algorithms (GAs)

GAs are part of a collection of stochastic optimization algorithms inspired by natural genetic and the theory of biological evolution. The idea behind GAs is to simulate natural evolution to optimize a particular objective function. In the last three decades, GAs have emerged as a practical, robust optimization and search method (Fogel 1997). A detailed review of GAs is presented in Section 2.2.

Genetic Programming (GP)

Koza viewed many different problems in artificial intelligence, symbolic processing, and machine learning as requiring the discovery of a computer program that produces certain desired output for particular input (Koza 1992). The process of solving these problems is equivalent to searching a space of possible computer programs for a most fit individual computer program. Koza (1992) developed GP, which provides a way to search for a most fit program for the given problem. A detailed review of GP is given in Section 2.3.

Evolutionary Programming (EP)

EP was developed by Fogel in the late 1960s. Although EP techniques originally aimed at evolving artificial intelligence in the sense of developing the ability to predict changes in the environment (Fogel 1995; Dasgupta and Michalewicz 1997), it is often used as an optimizer. EP differs philosophically from other EC techniques such as GA in a crucial manner. EP is a top-down versus GA's bottom-up approach to optimization. It is important to note that (according to neo-Darwinism) selection only operates on the phenotypic expressions of genotype; and the underlying code of the phenotype is only affected indirectly. The realization that a sum of optimal parts sometimes does not lead to an optimal overall solution is the key to this philosophical difference. GA's successive iterating local optimization

Chapter 2: Genetic Algorithms and Genetic Programming

process is different from EP, which is an entirely global approach to optimization. Solutions of an EP algorithm are judged solely on their fitness with respect to the given environment. No attempt is made to partition credit to the individual component of the solutions. In EP (and in ESs), the variation operator allows for simultaneous modification of all variables at the same time. Fitness, described in terms of the behavior of each population member, is evaluated directly, and is the sole basis for survival of an individual in the population. Thus, a crossover operation designed to recombine building blocks is not utilized in the general forms of EP (Porto 1997). EP strongly resembles ES and the major distinguishing features between the two algorithms occur at the representational level.

Evolution Strategies (ESs)

ESs are algorithms that mimic the principles of natural evolution to solve parametric optimization problem. They were developed and used in Germany at about the same time as GAs emergence in the USA during the late 1960s (Schwefel 1995). Early ESs were based on a population consisting of one individual and one genetic operator (mutation) only. This method is called the two-member ES. In ESs, an individual is represented as a pair of floating point-valued vectors, i.e., $v = (x, \sigma)$. Here, the first vector x represents a point in the solution space; the second vector σ represents the corresponding standard deviation. The individual is altered by mutation by $x_{t+1} = x_t + N(0, s)$, where $N(0, s)$ is a vector of independent, identical normally distributed random numbers with a mean zero and standard deviation σ . This is consistent with the biological observation that smaller changes occur more often than larger ones (Michalewicz 1996). The mutated individual replaces its parent if and only if the mutated individual has a better fitness and all existing constraints are satisfied. Otherwise, the offspring vanishes and the population remains unchanged. Many differences between ESs and GAs directly or indirectly stem from a substantial difference in the amount of ‘genetic’ information and its representation. The main important differences between ESs and GAs are summarized in Table 2.1 (Hoffmeister and Bäck 1990, Bäck and Hoffmeister 1992).

Chapter 2: Genetic Algorithms and Genetic Programming

Table 2.1 Main differences between GAs and ESs.

GAs	ESs
Genotype level of individuals	Phenotype level of individuals
No knowledge about the properties of the objective function	Knowledge of the dimension of the objective function
Parameter space restrictions for decoding purpose	No parameter restrictions apart from machine-dependencies
Proportional selection, ranking	(μ, λ) -selection, $(\mu + \lambda)$ -selection
Recombination serves as the main search operator	Mutation serves as the main search operator
Secondary role of mutation	Several recombination schemes
No collective self-learning of parameter settings	Collective self-learning of strategy parameters

An EC method typically first initializes a random population of possible solutions, although domain-specific knowledge can be used to bias the search. Then, the individuals are altered according to different operators such as crossover and mutation. Each individual in the population receives a measure of its fitness in the environment. Evaluation may be as simple as computing a fitness function or as complex as running an elaborate simulation. Selection focuses attention on high performance individuals, thus exploiting the available fitness information. Crossover and mutation perturb those individuals, providing general heuristics for exploration. The main loop of the algorithm, including alteration, evaluation and selection, is iterated until a pre-described termination criterion is satisfied. Darwinian principle of natural selection, i.e., survival-of-the-fittest, is the driving force of the evolution of individuals. Such a procedure, although simplistic from a biologist's viewpoint, is sufficiently complex to provide robust and powerful adaptive search mechanisms.

Chapter 2: Genetic Algorithms and Genetic Programming

A typical EC algorithm is outlined in Figure 2.1.

```

// start with an initial time
t := 0;
// initialize a usually random population of individuals
initial population P (t);
// evaluate fitness of all initial individuals in population
evaluate P (t);
// test for termination criterion (time, fitness, etc.)
while not done do
    // increase the time counter
    t := t + 1;
    // select sub-population for offspring production
    P' := select parents P (t);
    // recombine the "genes" of selected parents
    recombine P' (t);
    // perturb the mated population stochastically
    mutate P' (t);
    // evaluate its new fitness
    evaluate P' (t);
    // select the survivors from actual fitness
    P := survive P, P' (t);
end
end EC.

```

Figure 2.1 Structure of EC

EC can offer several advantages for solving difficult real world optimization problems. These advantages are related to its (Fogel 1997):

- Conceptual simplicity;
- Broad applicability;

Chapter 2: Genetic Algorithms and Genetic Programming

- Higher performance than the classic methods for real life problems;
- Can be easily hybridized with other methods;
- Suitability for parallel processing (computing);
- Adaptive solutions to changing circumstances;
- Capability to optimize its exogenous parameters; and
- Gradient information is not essential.

The “No-Free-Lunch” (NFL) theorem (Wolpert and Macready 1995) shows that all algorithms that search for the optima of a cost function perform exactly the same, when averaged over all possible cost functions. As a result, there cannot exist a single algorithm for solving all optimization problems that is consistently better than any other algorithms. The question of whether EC is inferior or superior to the other optimization methods does not make sense. It could be claimed that EC behaves better than other methods with respect to solving a specific class of problems (Schwefel 2000). Many classical methods are more efficient in solving linear, quadratic, convex, unimodal, separable, and many other special problems. On the other hand, EC is more efficient in solving discontinuous, nondifferentiable, multimodal, noisy problems (Schwefel 2000).

However, EC has some disadvantages. For instance, there is no guarantee of finding the global optimal solution and there are no reliable termination criteria. Often, EC is computationally expensive. There is also not enough theoretical basis. It is very difficult to carry out a comparison between different EC algorithms except experimentally. It is not possible to know how far the solution obtained by the algorithm is from the global optimum. Finally, and probably the worst characteristic, is the fact that EC depends on a set of parameters that have to be tuned experimentally for the problem at hand, with this tuning itself being a difficult optimization problem for some cases.

2.2 GENETIC ALGORITHMS

The basic principles of GAs were first introduced by John Holland in his pioneering book *Adaptation in Natural and Artificial Systems* (Holland 1975). The main idea is that to make a population of individuals adapt to certain special environment, it should behave like a natural system; survival, and therefore reproduction, which is promoted by the elimination of useless traits and by rewarding useful behavior. Holland's insight was in abstracting the fundamental biological mechanisms that permit system adaptation into a mathematically well-specified algorithm.

GAs are stochastic search techniques based on the mechanism of natural selection and genetics inheritance. They attempt to find a very good, if not the best, solution to the given problem. They have received considerable attention due to their potential as a search and optimization technique for complex problems and have been successfully applied in the areas of industrial engineering (Goldberg 1989). The well known applications include scheduling and sequencing, vehicle routing and scheduling, group technology, facility layout and location, transportation, structural optimization, inverse problem, and many others (Blanton and Wainwright 1993; Grefenstette 1994; Cheng and Gen 1996; Yao 1996; Soh and Yang 1996; Yang and Soh 1997; Gen and Cheng 1997; Chou and Ghaboussi 2001; Liu and Han 2003; Ma et al. 2004).

The conventional form of GAs, as described by Goldberg (1989), transforms a population of individuals, called chromosomes, each with an associated fitness value, into a new generation of population by using the Darwinian theory of evolution and biologically inspired operations. Each chromosome in the population represents a possible solution to the given problem. In the conventional GAs, the chromosomes are fixed-length strings of symbols, and usually, but not necessarily, binary bit strings. Variable-length GAs and real-coded GAs have also been proposed (Janikow and Michalewicz 1991; Srikanth et al. 1995; Yoon and Shoemaker 2001; Maulik and Bandyopadhyay 2003).

2.2.1 General Structure of GAs

Differing from the conventional search techniques, GAs start with a random population with each chromosome in the population representing a solution of the given problem. The chromosomes evolve through successive iterations, called generations. During each generation, the chromosomes are evaluated, using certain measures of fitness. To create the next generation, new chromosomes, called offspring, are formed by (a) merging two chromosomes from the current generation using a crossover operator and (b) modifying existing chromosomes using a mutation operator. A new generation is formed by (a) selecting, according to the fitness values, some of the parents and the offspring and (b) rejecting others so as to keep the population size constant. Fitter chromosomes have higher probabilities to be selected. After several generations, the algorithms converge to the best chromosome, which hopefully represents the optimum or sub-optimal solution to the problem.

The basic procedure of GAs is shown in Fig. 2.2.

Figure 2.2 Illustration of basic procedure of GAs

Chapter 2: Genetic Algorithms and Genetic Programming

As the basic procedure of GAs is very general, there are many different implementations according to the given problems.

2.2.2 Coding

In a GA program, each parameter, $x_i = (1, 2, \dots, N)$, of a given problem is encoded in a finite-length string according to the coding method. The most popular and simplest coding method is the binary coding method. It should be noted that binary code of the parameter is not absolutely necessary. Of course, there are many other ways of encoding. This mainly depends on the given problem and the solving procedure. For example, one can directly encode integer, real number, gray code, and so on. The chromosome is formed as a super string that combines all these finite-length strings to represent an individual. After the optimal solution is found, it is then decoded into real physical parameters, which are recognizable by human beings.

The objective function (fitness) is often defined using continuous variables of parameters. However, GAs operate on the discrete parameters (e.g. binary code). Therefore, these parameters in a continuous space should be first discretized, and then encoded in binary forms. The mathematical formulation for the binary encoding and decoding method of the i th parameter is given as (Haupt and Haupt, 1998)

Encoding

$$\bar{x}_i = \frac{x_i - x_i^L}{x_i^U - x_i^L} \quad (2.1)$$

$$gene[m] = round \left\{ \bar{x}_i - 2^{-m} - \sum_{k=1}^{m-1} gene[k] 2^{-k} \right\} \quad (2.2)$$

Decoding

$$x_i^{qn} = \frac{\sum_{m=1}^{N_{gene}} gene[m] 2^{N_{gene}-m}}{2^{N_{gene}} - 1} \quad (2.3)$$

Chapter 2: Genetic Algorithms and Genetic Programming

$$x_i^q = x_i^{qn} (x_i^U - x_i^L) + x_i^L \quad (2.4)$$

where: $\bar{x}_i$ is the normalized *ith* parameter; x_i^L is the minimum value of the *ith* parameter; x_i^U is the maximum value of the *ith* parameters; $gene[m]$ is the binary value of x_i , $round\{\}$ is the function to round the value to the nearest integer; N_{gene} is the number of bits in the gene; and x_i^{qn} is the quantized value of $\bar{x}_i$ and x_i^q is the quantized value of x_i .

2.2.3 Operators of GAs

Three basic genetic operators (i.e., selection, crossover and mutation) are performed on the chromosomes of the current generation to produce the next generation which is fitter in the simulated evolution process.

Selection is a process in which a mating pool of individual chromosomes of the current generation is chosen in a certain way for reproduction of the next generation according to the fitness value of the chromosomes of the current generation. This operator is designed to improve the average fitness of the population by giving higher fitness individual higher possibility to be copied to the next generation. A number of selection schemes, such as proportionate selection, ranking selection and tournament selection have been widely used in GA programs. Obviously, the selection operator is an artificial emulation of natural selection of the Darwinian nature selection theory.

Crossover is an operator to exchange part of the genes in the chromosomes of two parents in the mating pool to create new individuals for the next generation; it is the most important operator in GAs. A common crossover proceeds in two steps. First, members of the chromosomes in the mating pool are mated at random. Next, each pair of the randomly selected chromosomes undergoes a crossover using specific schemes to generate new offspring. One-point, multipoint and uniform crossover schemes (Lawrence 1987; Goldberg 1989; Davis 1991) are the most

Chapter 2: Genetic Algorithms and Genetic Programming

popular crossover schemes adopted for GA programs. The most common one-point crossover scheme is shown in Fig. 2.3.

Figure 2.3 One-point crossover scheme

Crossover can be rather complicated and highly dependent on the encoding method of the chromosome. Matching specific crossover scheme to specific problem can improve the performance of GAs.

Mutation is a background operator which produces spontaneous random changes in various chromosomes. A simple way to achieve mutation would be to alter one or more genes. In GAs, mutation serves the crucial role of either (i) replacing the genes lost from the population during the selection process so that they can be tried in a new context or (ii) providing the genes that were not present in the initial population. This is to prevent trapping all solutions of the population into a local optimum of the given problem and introduce some new possible solutions to the generation. Without mutation, the population would rapidly become uniform under the so called conjugated effect of selection and crossover mutation. For binary encoding, we can switch a few randomly chosen bits from 1 to 0 or vice versa. A typical mutation is shown in Fig 2.4:

Figure 2.4 Mutation operator for binary code

The mutation depends on the encoding as well as the crossover. For example, when we are encoding permutations, mutation could be exchanging two genes.

2.2.4 Features of GAs

GAs are stochastic global search methods and differ in fundamental concepts from the conventional gradient-based optimization and search procedures. Goldberg (1989) summarized these differences as follows:

- (1) GAs work with the coding of a solution set rather than the solution itself;
- (2) GAs search from a population of solutions rather than from a single solution;
- (3) GAs use payoff information (fitness function) rather than derivatives or other auxiliary knowledge; and
- (4) GAs use probabilistic transition rules rather than deterministic rules.

There are three major advantages when applying GAs to optimization problems (Gen and Cheng 1997):

- (1) GAs do not have much mathematical requirements about the optimization problems. GAs can handle any kind of objective functions and any kind of constraints (i.e., linear or nonlinear) defined in the discrete, continuous, or mixed search spaces;

Chapter 2: Genetic Algorithms and Genetic Programming

- (2) The ergodicity of evolution operators makes GAs very effective at performing global search (in probability). The traditional approaches perform local search by a convergent stepwise procedure, which compares the values of nearby points and moves to the relative optimal points; and
- (3) GAs provide great flexibility to hybridize with domain-dependent heuristics to make an efficient implementation for a specific problem.

One major disadvantage of GAs is their higher computational cost, generally, more evaluations of the objective functions are required by GAs than the traditional gradient-based search method. Another major disadvantage of GAs is their deficiency for problems with too many variables because of the exponential growth rate of the search space with respect to the increase of the number of variables (Liu and Han 2003).

2.3 GENETIC PROGRAMMING

GP is an attempt to deal with one of the promising questions in computer science (Samuel 1959), namely,

“How can computers learn to solve problems without being explicitly programmed? In other words, how can computers be made to do what needs to be done, without being told exactly how to do it?”

GP is an extension of the conventional GAs in which each individual in the population is a computer program. The book *Genetic Programming: On the Programming of Computers by Means of Natural Selection* (Koza 1992) demonstrated a result that many found surprising and counterintuitive, namely that as an automatic, domain-independent method, GP can genetically breed computer

Chapter 2: Genetic Algorithms and Genetic Programming

programs capable of solving, or approximately solving, a wide variety of problems from a wide variety of fields.

One of the central challenges of computer science is to let a computer do what needs to be done, without telling it how to do it. GP addresses this challenge by providing a method for automatically creating a working computer program from a high-level problem statement of the problem at hand. GP achieves this goal of automatic programming (sometimes called program synthesis or program induction) by genetically breeding a population of computer programs using the principles of Darwinian natural selection and biologically inspired operations. The operations include reproduction, crossover, mutation, and architecture-altering operations patterned after gene duplication and deletion in nature.

2.3.1 Preparatory Steps for GP

GP starts from a high-level statement of the requirements of a problem and attempts to produce a computer program that solves the problem. The human user communicates the high-level statement of the problem to the GP system by performing certain well-defined preparatory steps.

The five major preparatory steps for the basic version of GP require the human user to specify

- (1) the set of terminals for each branch of the to-be-evolved program;
- (2) the set of primitive functions for each branch of the to-be-evolved program;
- (3) the fitness measure (fitness function);
- (4) certain parameters for controlling the run; and
- (5) the termination criterion and method for designating the result of the run.

Chapter 2: Genetic Algorithms and Genetic Programming

The definition of the function set and terminal set for a particular problem (or category of problems) is usually a straightforward process. For some problems, the function set may consist of merely the arithmetic functions of addition, subtraction, multiplication, and division as well as a conditional branching operator. The terminal set may consist of the program's external inputs (independent variables) and numerical constants. This function set and terminal set is useful for a wide variety of problems (and corresponds to the basic operations found in virtually every general-purpose digital computer).

To obtain a legal and useful representation with regard to solving a given problem, the function set and terminal set must fulfill the following two important requirements (Fraser 1994):

- *Closure property:* The symbolic expressions will be operated on by the GP mechanism. It is therefore necessary that all possible arrangements of the expressions will lead to a program, which can be evaluated without any error. This property of the expressions is termed the closure property. For example: mathematical functions should be 'protected' from division by zero, negative logarithms, negative square root, etc.
- *Sufficiency property:* The problem must be solvable using the proposed selection of functions and terminals. In other words, the human operator/programmer must choose a set of functions and terminals, which are relevant (and could usefully be applied by the GP) to the given problem. When choosing the symbolic expressions, it is important that they can be expressed in some combination a solution to the given problem, and it must be possible to describe the final solution adequately by the selected functions/ terminals.

The terminal and function sets are important components of GP. After the terminal and function sets are defined properly; an individual can then be depicted as a rooted, point-labeled tree with ordered branches, using operators (internal

Chapter 2: Genetic Algorithms and Genetic Programming

points of the tree) from the function set and arguments (leaves of the tree) from the terminal set.

An example of such a typical tree is given in Fig. 2.5. It is called a GP parse tree.

Figure 2.5 A tree representation of an individual with equation $(\sqrt{2a} - b)/2a$

Using a LISP S-expression, the aforementioned example can be written as:

$(\div(-(\sqrt{(*2\ a))}\ b)\ (*2\ a))$

This expression, using a prefix notation, is read from left to right applying recursively each function to the next one or two arguments or sub-S-expression. Usually, this kind of expression is used for replacing the chromosome at the real computer program level.

The third preparatory step concerns the fitness measure for the problem. The fitness measure specifies what needs to be done. The fitness measure is the primary mechanism for communicating the high-level statement of the problem's requirements to the GP system. The first two preparatory steps define the search space whereas the fitness measure implicitly specifies the search's desired goal.

The fourth and fifth preparatory steps are administrative. The fourth preparatory step entails specifying the control parameters for the run. The most important control parameter is the population size. In practice, the user may choose a population size that will produce a reasonably large number of generations in the amount of computer time the user is willing to devote to a problem (as opposed to,

Chapter 2: Genetic Algorithms and Genetic Programming

say, analytically choosing the population size by somehow analyzing a problem's fitness landscape). Other control parameters include the probabilities of performing the genetic operations, the maximum size for programs, and other details of the run.

The fifth preparatory step consists of specifying the termination criterion and the method of designating the result of the run. The termination criterion may include a maximum number of generations to be run as well as a problem-specific success predicate. In practice, one may manually monitor and manually terminate the run when the values of fitness for numerous successive best-of-generation individuals appear to have reached a plateau. The single best-so-far individual is then harvested and designated as the result of the run.

2.3.2 General Structure of GP

The main generational loop of a GP run consists of the fitness evaluation, the Darwinian selection, and the genetic operations. Each individual program in the population is evaluated to determine how fit it is at solving the given problem. Programs are then probabilistically selected from the population based on their fitness to participate in the various genetic operations, with reselection allowed. While a fitter program has a larger possibility to be selected, even individuals known to be unfit are allocated some trials in a mathematically principled way.

The individuals in the random initial population and the offspring produced by each genetic operation are all syntactically valid executable programs. After many generations, a program may emerge that solves, or approximately solves, the given problem.

The steps of genetic programming are as follows:

- (1) Randomly create an initial population (generation 0) of individual computer programs composed of the available functions and terminals.

Chapter 2: Genetic Algorithms and Genetic Programming

- (2) Iteratively perform the following sub-steps (called a *generation*) on the population until the termination criterion is satisfied:
 - (a) Execute each program in the population and ascertain its fitness (explicitly or implicitly) using the problem's fitness measure.
 - (b) Select one or two individual program(s) from the population with a probability based on fitness (with reselection allowed) to participate in the genetic operations in (c).
 - (c) Create new individual program(s) for the population by applying the following genetic operations with specified probabilities:
 - (i) *Reproduction*: Copy the selected individual program to the new population.
 - (ii) *Crossover*: Create new offspring program(s) for the new population by recombining randomly chosen parts from two selected programs.
 - (iii) *Mutation*: Create one new offspring program for the new population by mutating a randomly chosen part of one selected program.
- (3) After the termination criterion is satisfied, the single best program in the population produced during the run (the best-so-far individual) is harvested and designated as the result of the run. If the run is successful, the result may be a solution (or approximate solution) to the problem.

A schematical view of GP is given in Fig. 2.6 (Koza 1992).

Chapter 2: Genetic Algorithms and Genetic Programming

Figure 2.6 Schematical view of GP

2.3.3 Operators of GP

Two primary operations exist for modifying the individuals in GP. The most important is the crossover operation. In the crossover operation, two individuals are

Chapter 2: Genetic Algorithms and Genetic Programming

combined to form two new individuals or offspring. The parental individuals are chosen from the population according to the fitness.

The creation of the first offspring from the crossover operation is accomplished by deleting the crossover fragment of the first parent and then inserting the crossover fragment of the second parent. The second offspring is produced in a symmetric manner. For example, consider the two parse trees depicted in Fig. 2.7. By exchanging the highlighted sub-trees between them, two offspring are generated.

Figure 2.7 Typical crossover operation for GP

Chapter 2: Genetic Algorithms and Genetic Programming

An important improvement that GP displays over GAs is its ability to create two new solutions (offspring) from the same solution (parent). This is critical for GP to maintain its population diversity when the search is approaching the local optima where the population may be dominated by many identical good solutions. In Fig. 2.8, the same parent solution is used twice to create two new different offspring solutions.

Figure 2.8 Crossover operation with identical parents

Chapter 2: Genetic Algorithms and Genetic Programming

Mutation is another important operation of GP. In the mutation operation, a single parental program is probabilistically selected from the population based on fitness. A mutation point is randomly chosen, the sub-tree rooted at that point is deleted, and a new sub-tree is grown there using the same random growth process that was used to generate the initial population. Two types of mutations are possible. In the first type, a function can only replace a function or a terminal can only replace a terminal. In the second type, an entire sub-tree can replace another sub-tree. Fig. 2.9 illustrates the concept of mutation of GP.

Figure 2.9 Two different types of mutation

The top parse tree is the original agent. The bottom left parse tree illustrates a mutation of a single terminal (2) for another single terminal (a). It also illustrates a mutation of a single function (-) for another single function (+). The parse tree on the bottom right illustrates the replacement of a sub-tree by another sub-tree.

2.3.4 Features of GP

GP has proven its worth in a broad spectrum of real-world problems with remarkable flexibility as a machine learning technique. The use of machine learning is able to help formalize the knowledge and produce the necessary rules (Arciszewski and Ziarko 1990, 1992). A survey of machine learning applications to civil engineering was done by Melhem and Nagaraja (1996). A number of learning systems and algorithms were reviewed and a brief summary of each had been presented. GP is also unique in its combination of symbolic and subsymbolic application areas. Some of GP's flexibility may be attributed to the freedom in choosing an arbitrary function and terminal sets. Another reason for the broad spectrum of GP application domain is its generally very robust evolutionary search. If it is tractable, evolving a way to do things is more useful than evolving the things. GP also has certain disadvantages, such as been extremely slow and needed very large initial population. Due to GP's inherent parallel mechanism (Salhi et al. 1998) and with the development of the high speed computer, these disadvantages may be overcome soon.

Although GP is an extension of GA, it is more powerful than GA in certain sense. Compared with GAs, in which solutions are represented by chromosomes, GP's dynamic variability in parse-trees are more natural and adaptive in representing and solving the problems (Yang 2001). Furthermore, the output of GAs is a quantity, while the output of GP is a computer program. In essence, this is the beginning of computer programs which are able to program themselves.

2.4 GA AND GP APPLICATIONS IN ENGINEERING

Research on EC in engineering has a relatively long history, which was initiated in Europe in the early sixties by Rechenberg (1965). The state-of-the-art review is provided by Bentley and Wakefield (1996), and Gen and Cheng (1997). Among existing computational paradigms, EC is recognized as particularly appropriate for various traditional and novel computational applications in civil engineering. Arciszewski and Jong (2001) provided an overview of the state-of-the-art of EC in civil engineering, and they also provided various conclusions and recommendations for further research.

GAs have received considerable attention regarding their potential as a search and optimization technique. It has been widely applied in many fields. The following brief review outlines some developments and applications of GAs, and its applications in engineering fields, especially in civil engineering.

Renders and Bersini (1994) investigated two ways of hybridizing GA with hill-climbing methods for global optimization. The first way involved two interwoven levels of optimization, with evolution (GA) and individual learning (hill climbing) cooperating in the global optimization process. The second way consisted of modifying GA by the introduction of new genetic operators or by the alteration of the traditional ones in such a way that these new operators captured the basic mechanisms of hill-climbing method. The results showed that the performances were improved with respect to the two isolated hybrid GAs.

Lis and Eiben (1997) proposed a new method for solving multicriteria optimization problems by multi-sexual genetic algorithm (MSGGA). In this method, individuals had an additional feature, their sex or gender, which was used in the recombination process. The fitness function was chosen according to the sex of the

Chapter 2: Genetic Algorithms and Genetic Programming

individual. The numerical experiments confirmed the MSGA's ability to find Pareto-optimal solutions.

One of the main obstacles in applying GAs to complex problems has been the high computational cost due to their slow convergence rate. To alleviate this difficulty, Yen and Lee (1997) developed a hybrid approach that combined GA with a stochastic variant of the Simplex method in function optimization. Their motivation for developing the stochastic Simplex method was to introduce a cost-effective exploration component into the conventional Simplex method. The numerical results showed that the hybrid approach was an effective and robust optimization technique.

Soh and Yang (1996) investigated a fuzzy controlled genetic-based search technique for structural shape optimization. The method was used in the least-weight design of truss structures, which included their geometry as design variables to be optimized. A fuzzy rule-based GA was implemented to control the optimization process. In order to improve their GA's search efficiency, Yang and Soh (1997) employed a tournament selection strategy as a replacement for the commonly used fitness-proportional strategy. The numerical results revealed that a significant reduction of computation cost could be achieved through the proposed method.

Friswell et al. (1998) applied GA to damage detection by using vibration data. The objective was to identify the position of one or more damage sites in a structure, and to estimate the extent of the damage at these sites. The GA was used to optimize the discrete damage location variables. For a given damage location site or sites, a standard eigensensitivity method was used to optimize the damage extent. This two-level approach incorporated the advantages of both the GA and the eigensensitivity method. The method was validated by a simulated beam and an experimental plate.

Chapter 2: Genetic Algorithms and Genetic Programming

Zhang and Leung (1999) proposed an orthogonal GA for multimedia multicast routing. Its salient feature was to incorporate an experimental design method called orthogonal design into the crossover operation. As a result, it could search the solution space in a statistically sound manner and it was suited for parallel implementation and execution. The results indicated that for practical problem sizes, the orthogonal GA could find near-optimal solutions within moderate numbers of generations. Leung and Wang (2001) designed another orthogonal GA with quantization for global numerical optimization with continuous variables. The results showed that the proposed algorithm could find the optimal or close-to-optimal solutions.

Magyar et al. (2000) presented a hybrid GA with an adaptive application of genetic operators for solving the three-matching problem (3MP), which was a nondeterministic polynomial (NP)-complete graphic problem. They introduced several general/heuristic crossover and local hill-climbing operators for the 3MP, and applied adaptation at the level of choosing among the operators. Their hybrid GA combined these operators to form an effective problem solver. The algorithm was hybridized as it incorporated local search heuristics, and it was adaptive as the individual recombination/improvement operators were fired according to their on-line performance.

Sarma and Adeli (2000) presented a fuzzy augmented Lagrangian GA for optimization of steel structures subjected to the constraints of the AISC allowable stress design specifications, taking into account the fuzziness in the constraints. The membership function for the fuzzy domain was found by the intersection of the fuzzy membership function for objective function and constraints using the max-min procedure. Nonlinear quadratic fuzzy membership functions were used for objective function and constraints. The features and advantages of the fuzzy GA included acknowledging the imprecision and fuzziness in the code-based design constraints, increased likelihood of obtaining the global optimum solution, improved convergence, and reduced total computer processing time.

Chapter 2: Genetic Algorithms and Genetic Programming

Burczynski et al. (2000) investigated the application of GA in engineering mechanics. A floating-point GA was employed to identify voids and cracks in structures. The satisfactory results obtained showed that evolutionary techniques were very effective and promising. However, they also pointed out that one disadvantage of the application of GA in these mechanical problems was the time consuming computation.

Kazarlis et al. (2001) investigated the potential of a micro GA (MGA) with small population and short evolution as a generalized hill-climbing operator. Combining a standard GA with the suggested MGA operator leads to a hybrid genetic scheme GA-MGA, with enhanced searching qualities. The main GA performed the global search while the MGA explored the neighborhood of the current solution provided by the main GA, looking for better solutions. In contrast to the conventional hill climbers that attempted independent steps along each axis, the MGA operator performed genetic local search. The major advantage of MGA was its ability to identify and follow narrow ridges of arbitrary direction leading to the global optimum. The results showed that GA-MGA exhibited a better performance in terms of solution accuracy, feasibility percentage of the attained solution, and robustness.

Lingras and Davies (2001) proposed the use of rough GAs based on the notion of interval values. A gene in a rough GA is represented by an interval. They explained how this generalization facilitated development of new genetic operators and evaluation measures. Various operations of rough GAs were illustrated using a simple document retrieval example. Rough GAs seemed to provide useful extensions for practical application.

Detection of structural damage is an inverse problem in structural engineering. There are three main questions in damage detection: the existence, the location and the extent of the damage. The problem was formulated as an optimization problem by Chou and Ghaboussi (2001), which was then solved using GAs. Static measurements of displacements at few degrees of freedom (DOFs) were

Chapter 2: Genetic Algorithms and Genetic Programming

used to identify the changes of the characteristic properties of structural members such as Young's modulus and cross-sectional area, which were indicated by the difference of measured and computed response. In order to avoid structural analyses in fitness evaluation, the displacements at unmeasured DOFs were also determined by GAs.

Krishnamoorthy et al. (2002) discussed the object-oriented design and implementation of a GA core library. Object-oriented design, apart from giving a more natural representation of information, also facilitated better memory management and code reusability. It was shown how classes derived from the implemented libraries could be used for the practical size optimization of large space trusses, where several constructibility aspects had been incorporated to simulate real-world design constraints.

Although GAs have been proven to be powerful and have been widely used, extensive experimentation and experience from a large number of applications have revealed certain limitations and shortcoming of GAs. For example, GAs are not good at identifying the model of the solution when the size of the solution is not known in advance (Yang 2001). In an attempt to enhance the qualities of GAs, the research community has added heuristic and hybridizations with other optimization techniques, which are usually problem specific.

An investigation by Chakraborty et al. (2002) on stiffened isotropic and composite plates has been conducted to determine the geometric and material parameters for the plate, as well as the stiffener, from experimental modal data and finite element predictions using a GA. The problem was formulated as a global minimization of the error function defined by the difference in undamped eigenvalues and eigenvectors, as predicted from the finite-element modeling to those obtained experimentally.

The vibration-based methods have also been applied to detect structural damage. Hao and Xia (2002) proposed a method which used a GA with real number

Chapter 2: Genetic Algorithms and Genetic Programming

encoding to identify structural damage by minimizing the objective function, which directly compares the changes in the measurements before and after damage.

Li and Kwan (2003) presented a hybrid GA for the bi-objective public transport driver scheduling problem. A driver schedule contained a set of shifts that covered all the required work, and each shift had an associated cost. The driver scheduling problem was to find a cover with the minimum cost using the minimum number of shifts. These two objectives were formulated as a fuzzified criterion. GA was used to derive a near-optimal weight distribution amongst the fuzzified criteria, so that a single valued weighted evaluation could be computed for each potential schedule. The benchmark experimental results demonstrated that this method is capable of solving large scale real-world driver scheduling problem.

Composite panel structure optimization is commonly decomposed into panel optimization subproblems, with specified local loads, resulting in manufacturing incompatibilities between adjacent panel designs. Adams et al. (2003) used a GA to exclusively generate and evaluate valid globally blended designs, utilizing a simple master-slave parallel implementation, implicitly reducing the size of the problem design space and increasing the quality of discovered local optima.

For testing rock- and concrete-like materials, a larger bar diameter of split Hopkinson pressure bar (SHPB) is required to satisfy the requirement of stress uniformity in the specimen. However, with a larger diameter bar, it was found that the stress-strain response is strongly influenced by wave dispersion. A new modified SHPB set has been proposed to reduce the wave dispersion in the conventional SHPB by Wang et al. (2004), where a GA was used to identify the rock dynamic property by adopting both the conventional and modified SHPB setups. The results showed that GA could identify the hard rock dynamic property with satisfactory accuracy for both the SHPB sets.

The traditional SHPB technique fails in the study of the dynamic behavior of polymeric and metallic foams for the desired maximum strain of these materials is

Chapter 2: Genetic Algorithms and Genetic Programming

often up to 80%. Based on the modified SHPB setup, Ma et al. (2004) adopted GA as the data processing technique to overcome this inherent disadvantage of the traditional SHPB technique. The illustrative example showed that the GA is able to identify the entire compression behavior of the aluminum foam correctly based on the modified SHPB setup.

Faribairn et al. (2004) presented a procedure to optimize the construction of mass concrete structures using GAs. The optimization criterion is the construction cost and the decision variables are the material types, placing temperature, height of lifts and time intervals between lifts. The constraint imposed on the decision variables is cracking due to thermal stresses. The results indicated that the procedure can be successfully used in the design of massive concrete structures.

GP has received considerable attention regarding its potential as a search and optimization technique. In recent years, it has been widely applied to many engineering fields, including civil engineering.

Walsh and Ryan (1996) reported an application in the domain of software engineering. Considering the huge body of serial software, an automatic parallelization of serial programs was highly relevant. Hence, they presented a GP-based system for auto-parallelization of serial software. The system's goal was to transform a serial program into a functionally equivalent highly parallel program.

It is a known fact that the selection and crossover operators contribute to generating solutions in GP. Traditionally, the crossover points are selected randomly by a normal selection method. However, the traditional method has several difficulties as the building blocks are broken because of blind application of the normal crossover. Takuya et al. (1998) proposed a depth-dependent crossover for GP, in which the depth selection ratio was varied according to the depth of a node. This method accumulates building blocks via the encapsulation of the depth-dependent crossover. The results demonstrated the superiority of the depth-dependent crossover to the normal crossover.

Chapter 2: Genetic Algorithms and Genetic Programming

Salhi et al. (1998) investigated parallel implementation of a GP based tool for symbolic regression. The implementation relied on a random island model, which combined both the conventional island model where migration of individuals between islands occurred periodically and niching where no migration took place. The system was designed for the algorithm to be synergistic with parallel/distributed architectures, and worked to maximise use of the processor time and minimise use of the network bandwidth without complicating the sequential algorithm.

System modeling is highly relevant in the automation and simulation processes. Until now, there are two main ways to manage the problem. The first is to collect the equations, normally differential, which direct the dynamics of the system and to solve them using, most of the time, the *S*-transform. The other way is to collect enough data from the process, and, based on a predefined structure of the model, use a method for the parameter adjustment such as least mean square technique. López et al. (1999) proposed an alternate method. Based on GP, a tray for the induction of models in the block diagram representation using simple discretized systems was made. A GA was then used to adjust the parameters. But, this was not an integration of GA and GP because in their work, GP and GA were used separately.

Sean and Lee (1997) presented a large and systematic body of data on the relative effectiveness of mutation, crossover and combination of mutation and crossover in GP. The GP programs commonly in form of trees representing LISP *S*-expressions, and a typical evolutionary run produces a great number of these trees. Due to this reason, a good tree-generation algorithm is very important to GP. Sean (2000) presented two new tree-generation algorithms for GP. These algorithms were fast, allowed the user to request specific tree sizes, and guaranteed probabilities of certain nodes appearing in the trees.

Chapter 2: Genetic Algorithms and Genetic Programming

GP usually has a wide search space and high flexibility, hence GP is able to search for global optimum solution. But, in general, GP's learning speed is not so fast. Niimi and Tazaki (2000) proposed a rule generation technique from a database using GP combined with associated rule algorithm. It took the rules generated by the associated rule algorithm as the initial individual of GP. The learning speed of GP was improved by the combined algorithm.

Soh and Yang (2000) introduced GP into civil engineering problem solving. A GP-based approach was proposed for the simultaneous sizing, geometry, and topology optimization of structures. An encoding strategy was presented to map the real structures and the GP parse trees. Numerical results showed that this approach could potentially be a powerful search and optimization technique in civil engineering problem solving. In order to improve their GP's performance, Yang and Soh (2000) integrated a fuzzy logical controller into GP for structural optimization and design. The numerical results showed that when comparing the proposed fuzzy logic controlled GP approach with the pure GP method, the proposed new approach had higher search efficiency.

Kojima et al. (2001) dealt with a quantitative nondestructive evaluation in eddy current testing for steam generator tubes of nuclear power plants using GP and fuzzy inference system. In their work, defects were detected as a probe impedance trajectory by scanning a pancake type probe coil. An inference system was proposed for identifying the defect shape inside and/or outside of the tubes. GP was applied to extract and select effective features from a probe impedance trajectory. Using the extracted features, a fuzzy inference system detected the presence, position, and size of a defect of the test sample. The effectiveness of this method was demonstrated through computer simulation studies.

Ahluwalia and Bull (2001) introduced a new approach to the use of automatically defined functions (ADFs) within GP. The technique consisted of evolving a number of separate sub-populations of functions, which could be used by a population of evolving main programs. They presented and refined a set of

Chapter 2: Genetic Algorithms and Genetic Programming

mechanisms by which the number and constitution of the function sub-populations could be defined, and compared their performance on two well-known classification tasks. The comparison showed that in all cases the co-evolutionary approach performed better than the traditional GP with and without ADFs.

Chen (2003) demonstrated a method, which used the macroevolutionary algorithm (MA) combined with GP to estimate the compressive strength of high-performance concrete (HPC). The results showed that MAGP was better than the traditional proportional selection GP for HPC strength estimation.

Ashour et al. (2003) investigated the feasibility of using GP to create an empirical model for the complicated non-linear relationship between various input parameters associated with reinforced concrete (RC) deep beams and their ultimate shear strength. The size and structural complexity of the empirical model were not specified in advance, but these characteristics evolved as part of the prediction. The engineering knowledge on RC deep beams was also included in the search process through the use of appropriate mathematical functions. Good agreement between the model predictions and experiments was achieved.

The structure of the non-linear constitutive models is a key to control non-linear behaviours of materials. Because the non-linear mechanical mechanism is not clearly understood in most cases, it is very difficult to assume the structure of the model in advance. Feng and Yang (2004) proposed a hybrid evolutionary method to solve this problem. GP was used to recognize the structure of the non-linear stress-strain relationship without any assumption in advance, and GA was then used to recognize its coefficients. The non-linear stress-strain relationship thus found not only satisfied the dynamic change in its structure but also its variables and coefficients. The results indicated that the coupling non-linear constitutive model of the structure and its coefficients could identify the model which the traditional constitutive model theory was unable to recognize.

Howard and D'Angelo (1995) proposed a new hybrid of GA and GP, GA-P, for symbolic regression. In their method, a GA string, representing the coefficients

Chapter 2: Genetic Algorithms and Genetic Programming

of an expression, was attached to a standard GP parse tree to form a chromosome. The numerical results showed that GA-P was capable of evolving complex mathematical expressions. Sanchez (2000) extended GA-P to interval GA-P to search for the algebraic expression that best approximated the experiment data.

Various applications of GP have shown that GP is a highly flexible machine learning technique. Generally, GP is powerful at identifying the model of the best solution of a given problem. However, GP is not as good as GAs at optimizing the parameters of the solution, i.e., fine-tuning the solution (Yang and Soh 2002).

CHAPTER 3 PARAMETER IDENTIFICATION FOR HARD ROCKS IN SHPB TEST

3.1 INTRODUCTION

Split Hopkinson Pressure Bar (SHPB) is recognized as one of the most effective methods for testing the stress-strain relation of various materials under high strain rate condition in the range of 10^1 - 10^3 s⁻¹. The principle of the SHPB test is to measure the stress-strain curve of the specimens based on one-dimensional wave propagation theory in aligned pressure bars (Hopkinson 1914; Davies 1948; Kolsky 1949b). The conventional SHPB test scheme has some obvious limitations, such as wave dispersion in the pressure bars, specimen size and incident pulse length requirements, uniformity of the stress wave, etc. Numerous efforts have been dedicated to improving the data acquisition and analysis methods of the SHPB tests (Bertholf and Karnes 1975; Lifshitz and Leber 1994; Bussac et al. 2002; Lee and Kim 2003; Meng and Li 2003). These methods are usually developed to overcome a specific limitation. However, they are sometimes not very effective in measuring the strain-rate-dependent material properties for brittle materials.

On the other hand, dynamic properties of hard rocks are required in many applications in rock engineering such as excavation, tunneling, earthquake and blast design, etc. The dynamic constitutive relation of rocks is indispensable in numerical modeling of these areas of applications. Some research works for obtaining the dynamic properties of rocks using SHPB apparatus (Yu and Jin 1990; Ross et al.

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

1990; Ross et al. 1995; Shan et al. 2000; Gomez et al. 2001; Li and Meng 2003) have been done. For rock- and concrete-like materials, a large-diameter SHPB is necessary to achieve relatively uniform stress wave in the aggregative materials. However, for large-diameter pressure bars, wave dispersion may be significantly introduced. Some researchers (Meng and Li 2003) have suggested shorter pressure bars to reduce the wave dispersion effect. However, overlapping of the incident and reflected signals may occur in the shorter pressure bars. To improve the SHPB test, wave separation method (Bussac et al. 2002; Meng and Li 2003; Zhao 2003), dispersion correction method (Follansbee and Frantz 1983; Lifshitz and Leber 1994; Zhao and Gray 1996), two-point strain gauge measurement, etc. have been suggested. A back-analysis method for processing of the measured signals in the SHPB test has also been adopted (Zhao 2003).

In order to reduce the effect of wave dispersion in the pressure bars and the size effect of the specimen during the traditional SHPB testing, a genetic algorithm (GA) (Wang et al. 2004) based method is proposed to identify the rock dynamic properties in the SHPB tests. The Hooke and Jeeves's method and simulated annealing (SA) based methods are proposed to identify the rock dynamic properties in SHPB test also. The advantages of these proposed parameter identification methods include, 1) the effect of the pressure bar length and the specimen size can be eliminated; 2) short SHPB configuration is applicable; 3) the strain gauge can be freely positioned on any location along the incident or transmission bars; and 4) it is sufficient to identify the rock material properties using the signal from only one strain gauge.

3.2 BASIC THEORY OF SHPB TEST

It has been noticed that the mechanical properties of most engineering materials under rapidly applied loads, such as impact and blast loads, differ from

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

those under static or dead loads, which might influence the structural behavior when subjected to dynamic loads.

The SHPB apparatus is used to test the uniaxial stress-strain curve under different strain rates. A typical SHPB set-up is shown in Fig. 3.1

Figure 3.1 Conventional SHPB setup

The SHPB set-up consists of an input (incident) bar and an output (transmission) bar with a short specimen sandwiched between them. To execute the experiment, a pressurized gas chamber fires the striker bar to impact the input bar, generating a compressive stress wave $\varepsilon_I(t)$ that travels over the input bar to the input bar/specimen interface. A portion of the wave, $\varepsilon_R(t)$ is reflected back the input bar while the remaining transmits through the specimen to the output bar. The stress wave in the specimen is approximately uniform. The transmitted wave $\varepsilon_T(t)$ is recorded at the output bar. The three waves in the pressure bars are shown in Fig. 3.2.

Figure 3.2 Incident, reflected and transmitted pulses

The wave signals are measured by strain gauges cemented on the input and output bars. Usually, the strain rate for the SHPB is lower than 10^3 s^{-1} .

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

In order to keep all the pressure bars within the elastic range, the impact velocity of the striker V must satisfy

$$V < \frac{2\sigma_y}{\rho C_0} \quad (3.1)$$

where ρ , C_0 and σ_y are respectively the density, the longitudinal elastic wave speed, and the uniaxial yield stress of the bar material.

The engineering stress, strain rate and strain of the specimen can be defined as following,

$$\begin{aligned} \sigma(t) &= \frac{P_1(t) + P_2(t)}{2A_0} \\ \dot{\varepsilon}(t) &= \frac{V_1(t) - V_2(t)}{L_0} \\ \varepsilon &= \int_0^t \dot{\varepsilon} dt \end{aligned} \quad (3.2)$$

where P_1 and P_2 are the forces acting on the two interfaces between the specimen and the input and output bars, respectively; V_1 and V_2 are the particle velocities at the ends of the input and output bars at the two interfaces; A_0 and L_0 are the area and the length of the specimen, respectively.

The elastic pressure bars satisfy the following one-dimensional wave propagation equation,

$$E \frac{\partial^2 u}{\partial x^2} = \rho \frac{\partial^2 u}{\partial t^2} \quad (3.3)$$

where E and ρ are the elastic modulus and the mass density of the bar material, respectively; and u is the particle displacement aligned with the wave propagation.

After applying the one-dimensional wave propagation theory and considering the equilibrium of the specimen, Eqn. (3.2) can be expressed by the corresponding incident, reflected and transmitted pressure strains as following

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

$$\begin{aligned}
\sigma(t) &= \frac{A_p}{2A_0} E(\varepsilon_I(t) + \varepsilon_R(t) + \varepsilon_T(t)) \\
\dot{\varepsilon}(t) &= -\frac{C_0}{L_0} (\varepsilon_T(t) - \varepsilon_I(t) + \varepsilon_R(t)) \\
\varepsilon(t) &= -\frac{C_0}{L_0} \int_0^t (\varepsilon_T(t) - \varepsilon_I(t) + \varepsilon_R(t)) dt
\end{aligned} \tag{3.4}$$

where A_p , E and C_0 are the cross section area, the Young's modulus and the elastic wave speed of the pressure bars, respectively; L_0 is the length of the specimen; ε_I , ε_R and ε_T are the recorded incident, reflected and transmitted strain pulses, respectively.

The stress in the specimen is assumed to be uniaxial compression and uniform along the specimen. For uniform uniaxial deformation, conservation of momentum requires that the total strain in the input bar $(\varepsilon_I(t) + \varepsilon_R(t))$ is equal to the total strain in the output bar $(\varepsilon_T(t))$. Thus

$$\varepsilon_I(t) + \varepsilon_R(t) = \varepsilon_T(t) \tag{3.5}$$

Then, Eqn. (3.4) can be simplified to

$$\begin{aligned}
\sigma(t) &= \frac{A_p}{A_0} E \varepsilon_T(t) \\
\dot{\varepsilon}(t) &= -\frac{2C_0}{L_0} \varepsilon_R(t) \\
\varepsilon(t) &= -\frac{2C_0}{L_0} \int_0^t \varepsilon_R(t) dt
\end{aligned} \tag{3.6}$$

Eqn. (3.6) is commonly used in the SHPB tests of different materials.

The traditional SHPB data analysis is mainly based on the following three assumptions:

- 1) one-dimensional wave propagation in the pressure bars and the specimen;

- 2) the pressure bars are elastic; and
- 3) the specimen inertia effect and the friction effect in the compressive test are negligible.

3.3 SIMPLIFIED ROCK MATERIAL MODEL

In rock damage mechanics, the uniaxial stress-strain relation of the rock material can be expressed as,

$$\sigma = E\varepsilon \quad (3.7)$$

where $E = E_0(1 - D)$, E_0 is the elastic modulus of the rock material; D is an isotropic damage scalar which can be represented by a three-parameter function,

$$D = 1 - e^{-\alpha \left(\frac{\varepsilon - \varepsilon_0}{\varepsilon_0} \right)^\beta} \quad (3.8)$$

in which ε_0 is a threshold strain; $D = 0$ when $\varepsilon < \varepsilon_0$; and α and β are two independent material coefficients (Mazars 1986; Lemaitre 1992). The damage scalar varies from 0 to 1 during the failure process of the material.

The uniaxial stress-strain curve of the rock material can thus be determined by the four material parameters, i.e. E_0 , ε_0 , α and β . The four parameters are versatile in modeling both the strain hardening and the strain softening of rock material. Fig. 3.3 shows some measured stress-strain curves of granite specimens in different strain rate conditions (Li et al. 2000). Assuming that $E_0 = 50$ GPa, $\varepsilon_0 = 0.003$, $\alpha = 0.9$ and $\beta = 1.4$, as shown in Fig. 3.3, the stress-strain curve based on the above damage model represents favorably the profile of the experimental results. Since the SHPB test is developed based on one-dimensional wave propagation, the uniaxial stress-strain curve given in Eqns. (3.7) and (3.8) is sufficient to simulate the measured curves.

Figure 3.3 Stress-strain relations with respect to different strain rates

In this study, identification of rock dynamic properties is characterized as an inverse problem for finding the four parameters from the measured responses of the SHPB test. However, in this study, the measured responses are obtained by numerically simulated SHPB tests instead of physical test. The advantage of using numerically simulated SHPB tests is that the accuracy and efficiency of the proposed parameter identification method can be better demonstrated.

3.4 NUMERICAL SIMULATION OF SHPB TEST

A typical configuration of the conventional SHPB test is shown in Fig. 3.4. The numerical SHPB test simulates the entire impact process, starting from the launch of the striker and terminating after the stress wave is completely transferred over both of the strain gauges on the two pressure bars. There are three types of solutions to the wave propagation equation, Eqn. (3.3), i) separated solution; ii) D'Alembert's solution; and iii) numerical solution. For some particular initial and boundary conditions, the separated and D'Alembert's solutions can be adopted to obtain the analytical solutions to Eqn. (3.3). However, for the pressure bars of the SHPB setup, the boundary conditions of the pressure bars are very complex impact boundary conditions. This means that the boundary conditions vary with time. Thus,

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

it is very difficult, and even impossible to obtain analytical solutions for Eqn. (3.3). In this study, finite difference method is employed to simulate the wave propagation in the system.

Figure 3.4 Typical configuration of conventional SHPB test

A finite difference equation of Eqn. (3.3) can be written as

$$E \frac{u(i+1, j) - 2u(i, j) + u(i-1, j))}{(\Delta x)^2} = \rho \frac{u(i, j+1) - 2u(i, j) + u(i, j-1))}{(\Delta t)^2} \quad (3.9)$$

where $x = i\Delta x$, $t = j\Delta t$, $i, j = 0, 1, 2, \dots$.

Eqn. (3.9) can be rewritten as

$$u(i, j+1) = 2(1-r)u(i, j) + r[u(i+1, j) + u(i-1, j)] - u(i, j-1) \quad (3.10)$$

where r is an aspect ratio given by

$$r = \frac{E}{\rho} \left(\frac{\Delta t}{\Delta x} \right)^2 \quad (3.11)$$

Eqn. (3.10) is an explicit finite difference formula for the wave equation. To be stable, the aspect ratio must satisfy $r \leq 1$.

The pressure bars are in the elastic state throughout the whole process, so the finite difference equations for the pressure bars can be represented by,

$$u_k(i, j+1) = 2(1-r_k)u_k(i, j) + r_k[u_k(i+1, j) + u_k(i-1, j)] - u_k(i, j-1) \quad (3.12)$$

where k represents the pressure bar material.

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

Since the rock material follows the nonlinear stress-strain curve, as described in Eqns. (3.7) and (3.8), the finite difference equations for the specimen can be represented by

$$u_r(i, j+1) = 2u_r(i, j) + \frac{(\Delta t)^2}{\rho_r \Delta x} (\sigma_r(i+1, j) - \sigma_r(i, j)) \quad (3.13)$$

where the subscript r represents the rock specimen material; ρ_r is the density of the rock specimen; and σ_r is the normal stress in the rock specimen, which can be determined from Eqns. (3.7) and (3.8) at a certain strain level.

An initial velocity is applied to the striker to activate the motion of the SHPB bar system. The initial velocity and displacement of the incident and transmission bars and the specimen are zero. The interface of the striker and the incident bar is modeled by a contact model. The sliding with closure and separation theory (John 1998) is adopted to solve the contact problem between the striker and the input bar. If the striker bar has penetrated the incident bar, an interface force will be added as follows,

$$f = -lk_f \text{ if } l < 0 \quad (3.14)$$

where f is the interface force; l is the penetration value; and k_f is the stiffness factor.

The interfaces of the pressure bars and the rock specimen are assumed to be in perfect contact during the impact process. It is a reasonable assumption because this study investigates the compressive stress-strain curve of the rock material, and the simulation is terminated as soon as the stress waves completely pass through the strain gauges attached on the pressure bars. Separation of the pressure bar and the material specimen will not happen during this response period.

Based on the conventional data analysis of the SHPB test, the engineering stress and strain along the specimen can be obtained based on Eqn. (3.4). These values are obtained from the two strain gauges located respectively at the middle of the incident and the transmission bars. The middle of the input bar is selected as the optimal strain gauge station because the incident and reflected pulses, propagating

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

in the positive and negative directions, are measured by one strain gauge on the input bar to avoid wave superposition.

In the numerical simulations, the aforementioned assumptions for the SHPB test are valid. The pressure bars are made of high-strength steel with Young's modulus $E = 200 \text{ GPa}$ and mass density $\rho = 7850 \text{ kg/m}^3$. The granite specimen has a mass density of $\rho = 2650 \text{ kg/m}^3$. From Fig. 3.3, it is apparent that the four-parameter model is capable of representing the stress-strain curve of rock materials in a certain strain rate.

3.5 PARAMETER IDENTIFICATION

The parameter identification methods developed in this chapter are error-driven process. They can be formulated as

$$W(E_0, \varepsilon_0, \alpha, \beta) = W((\varepsilon_I^m(t) - \varepsilon_I^e(t)), (\varepsilon_O^m(t) - \varepsilon_O^e(t))) \rightarrow \text{Min} \quad (3.15)$$

where $\varepsilon_I^m(t)$ and $\varepsilon_I^e(t)$ are respectively the measured and the estimated strain signals at the incident bar; and $\varepsilon_O^m(t)$ and $\varepsilon_O^e(t)$ are the corresponding strain signals at the transmission bar. W is an error function in terms of the differences between the measured and the estimated strain signals at the incident and the transmission bars.

3.5.1 Parameter Identification Using GA

GA is good at parameter identification and optimization (Soh and Yang 1996; Yang and Soh 1997; Wang et al. 2004). In this chapter, one chromosome is divided into four parts to decode the four coefficients needed to identify the rock dynamic properties. Both binary code GA and real code GA will be adopted in this paper. With proper dynamic analysis of one-dimensional wave propagation in the pressure bars and the specimen, we are able to reconstitute the dynamic stress-strain curve based on the measured strain pulses on the input and the output bars. As

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

mentioned before, what we used is a numerical SHPB test and the finite difference method based on one-dimensional wave propagation theory is employed in the numerical simulation. The numerical simulation is a critical step in the GA-based identification procedure, which updates the strain pulses at the middle of the incident and the transmission bars from the stress-strain curve decoded from the GA chromosome.

In this study, an alternative error function is used, i.e.

$$W' = C - W(E_0, \varepsilon_0, \alpha, \beta) \rightarrow \text{Max} \quad (3.16)$$

where C is a constant defined by the user for a specific problem. W' is a revised fitness function for the GA. The details for the definition of the fitness functions for different examples are given in the following section.

Fig. 3.5 depicts the flowchart of the dynamic material properties identification using the implemented GA. The necessary parameters and description of the binary code GA and the real code GA are given in Table 3.1.

Figure 3.5 GA-based parameter identification method

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

Table 3.1 Parameters used in the GA

Code type	Binary code	Real code
Selection type	Tournament selection	Tournament selection
Crossover type	Two point crossover	One point crossover
Chromosome length	40	4
Population size	300	300
Number of generation	50	50
Possibility of reproduction	20%	20%
Possibility of crossover	60%	60%
Possibility of mutation	20%	20%
Termination criterion	Generation=50	Generation=50
Maximum fitness	50.0	50.0

3.5.2 Parameter Identification Using Non-linear Mathematical Programming

Eqn. (3.16) can also be treated as non-linear mathematical programming problem. A number of algorithms have been developed over the years for the solution of nonlinear programming problems. Only a few have been demonstrated to be effective when applied to large-scale nonlinear programming problems, and none has proved to be so superior that it can be classified as a universal algorithm (Himmelblau, 1972).

Hooke and Jeeves' method (Hooke and Jeeves, 1961) is one of the most commonly used direct search methods for nonlinear mathematical programming problem solving. It is assumed that there is only one maximum in the region of search. In this method, an initial step size is chosen and the search direction is initiated from a given starting point. A combination of exploratory and pattern

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

moves is made iteratively to find the most profitable search directions. An exploratory move is employed first to find the best point around the initial point. If the exploratory move leads to an increase in the value of function, it is regarded as a success; otherwise, it is considered a failure. Then a pattern move is made to find the next point. The pattern move attempts to speed up the search performance.

The flowchart of using Hooke and Jeeves' method to identify the dynamic material properties is shown in Fig. 3.6.

Figure 3.6 Parameter identification using Hooke and Jeeves' method

The initial configuration is randomly generated within all boundaries. The parameters used in the Hooke and Jeeves' method are described as follows.

$\Delta^{(0)} = \max \left(\left| (E_0, \varepsilon_0, \alpha, \beta)^{(\max)} - (E_0, \varepsilon_0, \alpha, \beta)^{(0)} \right|, \left| (E_0, \varepsilon_0, \alpha, \beta)^{(0)} - (E_0, \varepsilon_0, \alpha, \beta)^{(\min)} \right| \right)$,
 and $\xi = 0.1\% * \left| (E_0, \varepsilon_0, \alpha, \beta)^{(\max)} - (E_0, \varepsilon_0, \alpha, \beta)^{(\min)} \right|$, where $(E_0, \varepsilon_0, \alpha, \beta)^{(\max)}$ and $(E_0, \varepsilon_0, \alpha, \beta)^{(\min)}$ are the upper and lower boundaries for the possible configuration, respectively. And η equals to 1.2. If the new configuration is beyond the boundaries, this new configuration will be reset at the nearest boundary of this configuration.

3.5.3 Parameter Identification Using SA

SA (Kirkpatrick et al. 1982) is based on the analogy between the simulation of the annealing of solids and the problem of solving large combinatorial optimization problems. For this reason the algorithm became known as “simulated annealing”. The SA can be viewed as such an algorithm: it is a general optimization technique for solving combinatorial optimization problems. The algorithm is based on randomization techniques. However, it also incorporates a number of aspects related to iterative improvement algorithms. The details about SA can be found at Ref (Laarhoven and Aarts 1987).

The error function is same as Eqn. (3.16). SA will be used to identify the dynamic material properties by finding the maximum of the revised error function.

Let randomly generated but valid material properties as the initial configuration with the cost as its energy, the general algorithm of applying SA to search for the best material properties can be outlined as follows:

1. randomly generate feasible material properties $(E_0, \varepsilon_0, \alpha, \beta)$, called the *current-configuration*.
2. start from the initial temperature $T = T_0$, while not reaching the final temperature T_{lowest} .
 {
 a) make a random change to the *current-configuration*, let *temp-configuration* be the configuration after the change;

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

- b) check to make sure that *temp-configuration* is valid.
Otherwise, go back to a);
 - c) calculated the costs of *current-configuration* (W_1') and *temp-configuration* (W_2') ;
 - d) determine whether to replace the *current-configuration* with the *temp-configuration*;
 - e) repeat a) to d) until a criterion is satisfied
 - f) reduce the temperature to a new T according to a cooling schedule.
- }

For implementing this generic algorithm, the most critical step is to develop a criterion for determining whether to accept or reject a change. The costs of the *current-configuration* (W_1') and *temp-configuration* (W_2') are first calculated based on the defined objective function, Eqn. (3.16). The algorithm for accepting or rejecting a change is shown in Fig. 3.7:

Input: *current-configuration* (W_1'), *temp-configuration* (W_2')

Output: *current-configuration* (by accepting or rejecting a change)

IF $W_2' > W_1'$

Accept the change (let the *temp-configuration* be the *current-configuration*)

ELSE

Generate a random number, X ($0 < X < 1$)

IF $X < \exp\left(\frac{(W_1' - W_2') * 10.0}{T}\right)$

Accept the change (let the *temp-configuration* be the *current-configuration*)

ELSE

Reject the change (let the *current-configuration* remain)

END IF

END IF

Figure 3.7 Algorithm for accepting or rejecting a change in SA

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

The following parameters will be adopted for the SA based method used in this chapter. The initial and final temperatures are 100.0 and 1.0, respectively. And the cooling schedule is $T = 0.9 * T$. At each temperature, 400 possible changes were checked one by one.

3.6 ILLUSTRATIVE EXAMPLES

In order to illustrate the capability and effectiveness of the proposed parameter identification methods, two numerical examples based on a conventional SHPB setup and a modified SHPB setup are investigated. Each program was run 10 times with the same parameters and different random seeds, and the result of the best run will be considered as the final result of the program. The ‘number of hit’ is defined as the number of successful run (the maximum fitness of the run is larger than or equals to 49.98) of 10 runs.

3.6.1 Example 1: Conventional SHPB Configuration

The first example used the conventional SHPB configuration as shown in Fig. 3.4. The traditional data analysis method reconstituted the stress-strain curve by using the strain signals collected at two gauges, i.e., gauges a and b. For verification purpose, the GA-based method is firstly applied to identify the stress-strain curve by using the signals at both gauge a and gauge b. Assuming that $E_0=50\text{GPa}$, $\varepsilon_0 = 0.003$, $\alpha = 0.9$ and $\beta = 1.4$, the signals obtained at the two gauges from the numerical SHPB test are shown in Fig. 3.8.

Figure 3.8 Strains at gauges a and b for conventional SHPB

Before the GA is used, some critical initial parameters and fitness function need to be defined. The target curve and the maximum and minimum stress-strain curves selected for GA are shown in Fig. 3.9. The domain within the upper and lower bounds contains the stress-strain curve of any kind of hard rock. It is also crucial to the GA to define a proper fitness function for the given problem. In order to identify the rock dynamic properties, the sum error of the measured signals collected from the gauges at the incident and transmission bars and the signals calculated from the chromosome of GA at the corresponding gauges is adopted to determine the fitness function. In this study, the following fitness function is selected,

$$F_r = 50.0 - \sum_{i=1}^{5000} \left(\left| \varepsilon_I^m(t_i) - \varepsilon_I^e(t_i) \right| + \left| \varepsilon_O^m(t_i) - \varepsilon_O^e(t_i) \right| \right) \quad (3.17)$$

in which

$$\left| \varepsilon_I^m(t_i) - \varepsilon_I^e(t_i) \right| + \left| \varepsilon_O^m(t_i) - \varepsilon_O^e(t_i) \right| = 0.01,$$

$$\text{if } \left| \varepsilon_I^m(t_i) - \varepsilon_I^e(t_i) \right| + \left| \varepsilon_O^m(t_i) - \varepsilon_O^e(t_i) \right| > 0.01,$$

where, $\varepsilon_I^m(t_i)$, $\varepsilon_I^e(t_i)$, $\varepsilon_O^m(t_i)$ and $\varepsilon_O^e(t_i)$ are the measured and the estimated strains at the gauges a and b at time t_i , respectively.

Figure 3.9 Target curve and the upper and lower bounds curves for GA

For the parameters listed in Table 3.1, the evolution of both the best and the mean fitness of the population with the increase of the generation number of the best run of real code GA and binary code GA is shown in Fig. 3.10. It can be observed that both the best fitness and the mean fitness rapidly increase in the initial generations, and the best fitness stabilizes at its optimal value after 10 generations while the mean fitness slightly fluctuates in the later generations. This fluctuation may be induced by the randomness of the GA search. It is worth mentioning that this small fluctuation of the mean fitness does not hinder the GA from finding the optimum solution; conversely, it may help to maintain the diversity of the population and thus preventing the GA search from trapping in the local optima.

Figure 3.10 Fitness evolution for conventional SHPB

The objective function and the boundaries of the stress-strain curve for the Hooke and Jeeves' method and the SA method are same as those defined for GAs. The parameters for Hooke and Jeeves' method and SA method are listed in Section 3.5.2 and 3.5.3, respectively. After each program was run 10 times, comparison of the performance of these methods is shown in Fig. 3.11. The 'number of hit' of 10 runs for the Hooke and Jeeves's methods, the SA method, the binary code GA and the real code GA are 2, 3, 7 and 9, respectively.

From Fig. 3.11, it can be observed that GA (real code GA and binary code GA) outperforms the Hooke and Jeeves's method and the SA method. And the real code GA based method outperforms the binary code GA based method. Thus, only real code GA and binary code GA will be adopted to identify the parameters for hard rocks at following examples.

Figure 3.11 Comparison of the performance of different methods

One of the outstanding advantages of the GA-based identification method is that the strain signal collected at only one gauge is sufficient for identifying the rock material properties. In the following, pulses from gauge a or gauge b are adopted separately in the identification.

Using the strain signal collected from gauge *a*, the fitness function is modified as follows,

$$F_r = 50.0 - \sum_{i=1}^{5000} \left(\left| \varepsilon_I^m(t_i) - \varepsilon_I^e(t_i) \right| \right) \quad (3.18)$$

in which $\left| \varepsilon_I^m(t_i) - \varepsilon_I^e(t_i) \right| = 0.01$ if $\left| \varepsilon_I^m(t_i) - \varepsilon_I^e(t_i) \right| > 0.01$, where $\varepsilon_I^m(t_i)$ and $\varepsilon_I^e(t_i)$ are the measured and the estimated strains at gauge *a* at time t_i , respectively. The ‘number of hit’ of 10 runs for the binary code GA and the real code GA are 8 and 9, respectively.

Using the strain signal collected from gauge b, the fitness function is modified as follows,

$$F_r = 50.0 - \sum_{i=1}^{5000} \left(\left| \varepsilon_o^m(t_i) - \varepsilon_o^e(t_i) \right| \right) \quad (3.19)$$

in which $\left| \varepsilon_o^m(t_i) - \varepsilon_o^e(t_i) \right| = 0.01$ if $\left| \varepsilon_o^m(t_i) - \varepsilon_o^e(t_i) \right| > 0.01$, where $\varepsilon_o^m(t_i)$ and $\varepsilon_o^e(t_i)$ are the measured and estimated strains at gauge b at time t_i , respectively. The ‘number of hit’ of 10 runs for the binary code GA and the real code GA are 7 and 10, respectively.

The target stress-strain curve and the reconstituted curves are shown in Figs. 3.12 (by binary code GA) and 3.13 (by real code GA), respectively. It is apparent that using the signals collected from the two gauges and using that from an individual gauge, the GA-based parameter identification method is able to accurately identify the rock material properties.

Figure 3.12 Comparison of the target and the reconstituted (by binary code GA) stress-strain curves from gauges a and b in conventional SHPB

Figure 3.13 Comparison of the target and the reconstituted (by real code GA) stress-strain curves from gauges a and b in conventional SHPB

Gauges a1 and b1 are located near the specimen/bar interfaces so as to reduce the wave dispersion effect. The strain signals collected from gauges a1 and b1 are supposed to be better in determining the stress and strain encountered in the specimen than the strain signals collected from gauges a and b. It is difficult to derive the stress-strain curve by using the traditional data analysis method because of the overlapping of the incident and reflected pulses. The GA-based identification method, however, can achieve very accurate stress-strain curve from the measured signals at gauges a1 and b1. The fitness function is the same as defined in Eqn. (3.17). The ‘number of hit’ of 10 runs for the binary code GA and the real code GA are 7 and 7, respectively.

In order to further demonstrate the advantages of the GA-based identification method, the signals collected from gauges a1 and b1 are also adopted respectively to identify the rock material properties. The strain signals collected at gauges a1 and b1 from the numerical SHPB test are shown in Fig. 3.14. The fitness functions are the same as defined in Eqns. (3.18) and (3.19) respectively. The

‘number of hit’ of 10 runs for the binary code GA and the real code GA using signal from gauges a1 are 7 and 8, respectively. The ‘number of hit’ of 10 runs for the binary code GA and the real code GA using signal from gauges b1 are 6 and 9, respectively.

Figure 3.14 Strain signals at gauges a1 and b1 for conventional SHPB

The target curve and the reconstituted curves from gauges a1 and b1 from the conventional SHPB configuration are shown in Figs. 3.15 (by binary code GA) and 3.16 (by real code GA), respectively. It is very clear that the GA-based method proposed in this chapter can accurately reconstitute the stress-strain curve of the rock specimen from the strain signals collected from gauges a1 and b1. It can also be found that the GA-based method can derive very accurate stress-strain curve of the rock specimen from only one gauge, i.e. either gauge a1 or gauge b1. This will significantly reduce the effect of wave dispersion.

From the obtained results, it is shown that the GA-based identification method can effectively and accurately identify the rock dynamic properties using the conventional SHPB configuration. No matter where the gauge location is, and whether two gauges or only one gauge signals are applied, the GA-based identification method is capable of reconstituting the stress-strain curve of the rock specimen accurately.

Figure 3.15 The target and the reconstituted (by binary code GA) stress-strain curves from gauges a1 and b1 for conventional SHPB

Figure 3.16 The target and the reconstituted (by real code GA) stress-strain curves from gauges a1 and b1 for conventional SHPB

3.6.2 Example 2: Shorter SHPB Configuration

It is well known that wave dispersion and attenuation increase with their travel distance in the pressure bars, especially when the bar diameter is relatively large. For the conventional SHPB configuration, wave dispersion is very critical to the final result of the stress-strain curve. If the length of the incident bar and the distance between the strain gauge station and the bar/specimen interface can be reduced, the effect of wave dispersion and attenuation can be significantly reduced (Meng and Li, 2003). Therefore, it is preferable to obtain strain records from locations close to the bar/specimen interface within minimum time-shift distance or to use shorter SHPB bars. However, it will result in overlapping between the incident and reflected pulses, and therefore the traditional data analysis method becomes invalid. In this chapter, the genetic identification method is incorporated which makes it feasible to use shorter SHPB to obtain more accurate compressive stress-strain behavior of rock material. A typical configuration is shown in Fig. 3.17. The incident and transmission bars are made of the same material as those used in the conventional SHPB. The target curve and the upper and lower bound stress-strain curves for the GA as shown in Fig. 3.9 are again adopted.

Figure 3.17 Configuration of modified SHPB

First, the strain signals at gauge a1 and gauge b1 are applied. The measured strain signals obtained from the numerical SHPB tests are shown in Fig. 3.18. The fitness function is the same as given in Eqn. (3.17). The process of evolution is

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

shown in Fig. 3.19. The ‘number of hit’ of 10 runs for the binary code GA and the real code GA are 5 and 7, respectively.

Figure 3.18 Strains at gauges a1 and b1 for modified SHPB

Figure 3.19 Fitness evolution for modified SHPB

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

Subsequently, only the strain signal collected at gauge a1 or b1 is used. Same as the previous example, the fitness functions are again defined by Eqns. (3.18) and (3.19) correspondingly. The ‘number of hit’ of 10 runs for the binary code GA and the real code GA using signal from gauges a1 are 6 and 8, respectively. And The ‘number of hit’ of 10 runs for the binary code GA and the real code GA using signal from gauges b1 are 7 and 7, respectively.

The target curve and the reconstituted stress-strain curves for the modified SHPB are shown in Figs. 3.20 (by binary code GA) and 3.21 (by real code GA), respectively. Again, it is shown that for the signals collected from two points at the incident and the transmission bars and for the signal collected from only one point located at either the incident or the transmission bar of the modified SHPB, the GA-based identification method is able to accurately identify the stress-strain curve.

Figure 3.20 The target and the reconstituted (by binary code GA) stress-strain curves for modified SHPB test

Figure 3.21 The target and the reconstituted (by real code GA) stress-strain curves for modified SHPB test

Comparison of the target coefficients and the reconstituted coefficients from the conventional SHPB (Example 1) and the modified SHPB (Example 2) by using binary code GA and real code GA are listed in Table 3.2 and 3.3, respectively. It can be observed that the real code GA based method outperforms the binary code GA based method for these two examples.

It is shown that the results obtained by GAs (binary code GA and real code GA) approximate the target coefficients. Although some minor errors exist for the identified parameters, the stress-strain curve reconstituted from the obtained parameters can match the target curve well, and the accuracy and efficiency of the GA-based method are acceptable in numerical simulations of rock-like materials.

Chapter 3: Parameter Identification for Hard Rocks in SHPB Test

Table 3.2 Comparison of target and reconstituted coefficients (by binary code GA) of hard rock

Example	Gauges	E_0 (Pa)	ε_0	α	β
Input		5e10	0.003	0.9	1.4
Example 1	a and b	4.97947e+010	0.00296383	0.849951	1.3998
	a	5.18964e+010	0.00268231	0.683773	1.4741
	b	5.0088e+010	0.00279765	0.717009	1.49951
	a1 and b1	5.02346e+010	0.0028739	0.78739	1.45259
	a1	4.83284e+010	0.00298338	0.80694	1.49267
	b1	4.91105e+010	0.00286413	0.745357	1.56305
Example 2	a1 and b1	4.95503e+010	0.00321408	1.09042	1.27957
	a1	4.97458e+010	0.003087	0.975073	1.35875
	b1	5.19941e+010	0.00276246	0.752199	1.40469
Average absolute error		8.53711e+008	0.0001526	0.135875	0.072054

Table 3.3 Comparison of target and reconstituted coefficients (by real code GA) of hard rock

Example	Gauges	E_0 (Pa)	ε_0	α	β
Input		5e10	0.003	0.9	1.4
Example 1	a and b	5.0008e+010	0.00299915	0.899524	1.40012
	a	5.0119e+010	0.00297856	0.883826	1.40475
	b	4.97903e+010	0.00308061	0.970636	1.35705
	a1 and b1	4.99477e+010	0.00301082	0.908553	1.39687
	a1	4.96556e+010	0.00309101	0.979234	1.37127
	b1	4.90691e+010	0.00320658	1.07221	1.32925
Example 2	a1 and b1	4.99577e+010	0.00300061	0.898639	1.40163
	a1	4.94025e+010	0.00313428	1.02168	1.38474
	b1	4.92063e+010	0.00313964	1.01014	1.37007
Average absolute error		3.44200e+008	0.00007620	0.064496	0.021917

3.7 CONCLUSIONS

Knowledge about the dynamic properties of rock is very important for practical reasons. In this chapter, identification of the rock dynamic properties by using a GA-based method is proposed. In order to reduce the effect of wave dispersion in the SHPB test, a modified SHPB configuration is presented. The advantages of the GA-based identification method in the conventional and the modified SHPB configurations are clearly illustrated. By using the proposed parameter identification method, the strain gauges can be freely positioned on the pressure bars. Overlapping of the incident and reflected signals is never burdened in the SHPB test. Single strain signal is also sufficient in identifying the material parameters when the developed GA is incorporated. The numerical results show that the GA based method outperforms the Hooke and Jeeves's based method and the SA based method, and the real code GA based method outperforms the binary code GA based method for identifying the parameters for hard rocks.

It should be highlighted that the GA-based method does not need further effort for wave separation. It can be conveniently extended to other brittle materials.

For the purpose of verifying the efficiency of the GA-based parameter identification method, the strain rate effect is not included in the present material model. The present model identifies the stress-strain curve in a certain strain rate level. It can be easily revised by multiplying a rate-dependent function, i.e., $\sigma = E\varepsilon \cdot f(\dot{\varepsilon})$, as shown by the many available empirical models. This study employed numerical simulation instead of physical SHPB test for the measured signals as it is able to verify the proposed identification method more precisely since the influence of noises by other factors can be avoided.

Future work will focus on using axisymmetrical wave propagation model to simulate the SHPB test. Other type of evolutionary computation method such as GP (Koza 1992) may also be considered to identify the stress-strain curve for its flexibility and capability to simulate the whole stress-strain curve of the materials without any preliminary assumptions for the curve.

CHAPTER 4 IDENTIFICATION OF DYNAMIC PROPERTIES OF ALUMINUM FOAMS

4.1 INTRODUCTION

The Split Hopkinson Pressure Bar (SHPB) technique, which has been initially used in compression, has been extended to tension (Harding et al. 1960), torsion (Duffy et al. 1971), simultaneous torsion compression (Lewis and Goldsmith 1973), and simultaneous compression torsion (Chichili and Ramesh 1999). Its application has also been extended from metallic materials to various nonmetallic materials, including polymer, foam, wood, concrete, rock, rubber, ceramic and composite (Zhao 1998; Li et al. 2000; Gary and Zhao 2000; Svante 2002; Meng 2002; Lee 2003). For the cases where the traditional SHPB analyses do not give acceptable results, an identification technique based on an inverse calculation method is necessary (Zhao 2003).

The measuring technique using bars relies on the knowledge of the two elementary waves propagating in opposite directions. Once they are known, they can be time shifted to the desired cross-sections (for example, bar specimen interfaces) to calculate all the mechanical variables for the specimen. The SHPB technique uses long bars with short loading pulse so that a cross-section exists where the total incident pulse and the first part of reflected waves (of the same duration) can be recorded without overlapping. A maximum observation duration then exists depending on the length of the bar. The measuring duration ΔT of a

Chapter 4: Identification of Dynamic Properties of Aluminum Foams

classical SHPB set-up is limited to $\Delta T \leq L/C$, where C and L are the wave speed and the length of the bar, respectively (Kolsky 1963). Consequently, the total relative displacement Δl between the two bar/specimen interfaces is limited for a given loading speed V ($\Delta l \leq V\Delta T$). For the material behavior testing at a given average strain rate, the maximum measurable strain is limited ($\varepsilon_{\max} \leq \dot{\varepsilon}\Delta T$). This limitation of the maximum measurable strain is one of the main inherent disadvantages of the traditional SHPB setup and its corresponding data processing procedure for testing soft material.

The SHPB technique fails in the study of the dynamic behavior of polymeric or metallic foams due to the fact that the desired maximum strain is often up to 80% of the densification region. To increase the measuring duration of the SHPB test, some researchers have recorded and analyzed multiple reflections in the pressure bars. Campbell and Duby (1956) have reported a method based on the one-dimensional elastic wave theory. Lundverg and Henchoz (1977) have proposed a simple explicit formula based on the one-dimensional wave propagation assumption to separate the two elementary waves and to measure the particle velocity after the observation window, using two signals recorded at two different cross-sections in a bar. Recently, this method has found new application in the prediction of wave propagation in a bar with a non-uniform impedance due to a temperature gradient (Lunberg et al. 1990), and has been successfully used in high temperature SHPB testing (Bacon et al. 1991, 1994; Lataillade et al. 1994). Zhao and Gary (1997) proposed a new method for the separation of waves. An iterative formula allowing for the calculation of the two virtually separate elementary waves for the total desired testing duration is presented. And this method was applied to the SHPB setup for an unlimited duration of measurement. Using bars of equal dimensions, it allows a quasi-unlimited measuring duration which can be up to 100 times longer than the classical one.

To test soft materials, such as polymeric or metallic foams, at high strain rates, the use of a SHPB setup made of low impedance bars (such as polymethyl methacrylate [PMMA] or nylon), which are mostly viscoelastic, is also suggested.

Chapter 4: Identification of Dynamic Properties of Aluminum Foams

The details of using viscoelastic bars have been given by Zhao et al. (1997). It is shown that the use of the viscoelastic SHPB has some advantages in testing soft materials such as foams. However, due to the viscoelastic nature of the bars, a proper correction of the wave dispersion is necessary to obtain reliable measurements.

The main objective of this study is to find an effective and practicable method which is able to obtain the entire stress-strain curve of the aluminum foams even though the length of the specimen is necessarily large. The traditional SHPB setup and its corresponding data processing method become invalid when the specimen is larger. Hence, modified SHPB setups and their corresponding data processing method will be proposed in this study to solve this problem. A similar genetic algorithm (GA), which is used in the former chapter, will again be adopted.

Based on the experience obtained from Chapter 3, in this chapter, the GAs (binary code GA and real code GA) will be integrated with the modified SHPB setups to reconstitute the entire stress-strain curve of the aluminum foam, including the densification region. The numerical simulation results show that the proposed GA-based methods are able to correctly identify the entire stress-strain curve of the aluminum foam. The outstanding advantages of the proposed GA-based methods are that only one strain gauge is needed and the strain gauge can be cemented on any location of the pressure bars. Furthermore, the modified SHPB setups can be implemented in laboratories.

4.2 DYNAMIC PROPERTIES OF ALUMINUM FOAMS

Aluminum foams are ultra-light solids which can absorb considerable energy by plastic dissipation in compression. Due to their porous structure, aluminum foams have high specific stiffness. The electrical and thermal conductivities of aluminum foams are considerably reduced, but still in the typical

Chapter 4: Identification of Dynamic Properties of Aluminum Foams

range of the metallic materials. Aluminum foams have good mechanical damping and sound insulation properties. They provide excellent energy absorption features at a higher strength level as compared to foamed polymers. Their cellular microstructure endows them with the ability to undergo large deformation at nearly constant nominal stress. It is expected that they will find application in absorbing impacts and shocks, such as in crash barriers or in blast mitigators (Deshpande and Fleck 2000). Such use of aluminum foams requires knowledge of the effect of impact velocity and strain rate on their compressive behavior. Knowledge of the compressive behavior of low impedance materials (like foams) under crash situations has been required in the automotive industry (Gary et al. 1995) and has appeared as an important goal.

After numerical analysis of the compression behavior of aluminum foams, it has been found that the Lorentz curve can fit the stress-strain curve of the aluminum foam beyond the elastic portion. It has the form,

$$\sigma = \sigma_{0a} + \frac{2A}{\pi} \frac{w}{4(\varepsilon - \varepsilon_c)^2 + w^2} \quad (4.1)$$

where, ε is a variable (strain) and σ is a function value (stress).

There are only a few critical parameters to determine for the entire stress-strain curve based on the Lorentz curve. Fig. 4.1 gives the measured stress-strain curves for different density aluminum foams. In order to simulate the three curves given in Fig. 4.1 (Baumeister et al. 1997), five parameters, i.e., the elastic modulus E_{0a} , the maximum elastic strain ε_{0a} , and A , w , ε_c , which are the parameters in Eqn. (4.1), should be determined for each curve. To keep the continuity of the curve, σ_{0a} can be calculated as

$$\sigma_{0a} = E_{0a} * \varepsilon_{0a} - \frac{2A}{\pi} \frac{w}{4(\varepsilon_0 - \varepsilon_c)^2 + w^2} \quad (4.2)$$

Chapter 4: Identification of Dynamic Properties of Aluminum Foams

The three nonlinear fitted curves are also shown in Fig. 4.1. The parameters for the middle curve are $E_{0a} = 327 \text{ MPa}$; $\varepsilon_{0a} = 0.023$; $\sigma_{0a} = 6.35 \text{ MPa}$; $A = 33.4 \text{ MPa}$; $w = 0.1636$; $\varepsilon_c = 0.8811$.

Figure 4.1 The real stress-strain curves and nonlinear fitted curves

4.3 TRADITIONAL SHPB TECHNIQUE TO RECONSTITUTE COMPRESSION BEHAVIORS OF ALUMINUM FOAMS

A typical configuration of the conventional SHPB test used in this study is shown in Fig. 4.2. The whole traditional SHPB setup is an aligned bar system comprising a striker bar, two long pressure bars (input and output bars) and the specimen.

Figure 4.2 Typical configuration of conventional SHPB test

The numerical SHPB test, which is similar to the one adopted in the previous chapter, is used to simulate the entire impact process, starting from the launch of the striker and terminating after the stress wave is completely transferred over both the strain gauges on the pressure bars. The finite difference method is again employed to model the wave propagation in the system.

For the traditional SHPB setup shown in Fig. 4.2, the pressure bar materials is made of high strength steel with Young's Modulus $E=200\text{GPa}$ and mass density $\rho=7850\text{kg/m}^3$. The pressure bars used in the modified SHPB setups in the following sections are also of the same material.

To examine the specimen size effect, two specimens of 4mm and 40mm thickness respectively are considered in the numerical simulations. The stress-strain relationship given in Fig. 4.1 is assumed. Numerical simulation of the entire traditional SHPB test for the two aluminum foams is then carried out. The strains at the middle point of the pressure bars are obtained from the numerical simulation. The reconstituted and input curves for the two specimens are shown in Fig. 4.3.

From Fig. 4.3, it can be found that the thickness of the aluminum foam specimen will influence the reconstituted stress-strain curve significantly. When the thickness of the specimen is 4mm , the traditional data processing method is able to

Chapter 4: Identification of Dynamic Properties of Aluminum Foams

reconstitute the stress-strain curve correctly based on the traditional SHPB setup. But when the thickness of the specimen increases to 40mm, the traditional SHPB setup and the corresponding data processing method can only obtain the very initial part of the entire stress-strain curve, the maximum strain which can be measured is only about 10%. This example also confirms that the limitation of the maximum desired strain of measurement is one of the inherent disadvantages of the traditional SHPB technique for testing soft materials.

Figure 4.3 Input and reconstituted stress-strain curves (thickness of the specimen: 4mm and 40mm)

Usually, the densification strain of aluminum foams is from 50% to 80%, and the real specimen for the aluminum foam should be 30-50mm in length to include a few cells in the longitude direction. Therefore, the traditional SHPB setup and its corresponding data processing method are invalid according to the above numerical simulation results. They are unable to reconstitute the entire stress-strain curve of the aluminum foams when the length of the specimen is necessarily large.

To practically solve this problem, two modified SHPB setups incorporated with a GA-based parameter identification method are proposed in the following sections to solve this problem effectively.

4.4 GENETIC ALGORITHM (GA): CHROMOSOME

The detailed description of GA can be found in Chapter 2. Hence, it will not be repeated here.

In this chapter, both binary code GA and real code GA will be adopted. The typical chromosome which is used in the following examples is shown in Fig. 4.4. The relationship between the chromosome and its represented characteristics is also shown in Fig. 4.4.

1101101110	1010101110	0011110001	1101101011	1010111010
$E_{0\alpha}$	$\varepsilon_{0\alpha}$	A	w	ε_c

(a) for binary code GA

3.2236e+08	0.0243122	2.32156e+07	0.167918	0.8446
$E_{0\alpha}$	$\varepsilon_{0\alpha}$	A	w	ε_c

(b) for real code GA

Figure 4.4 Typical GA chromosome and represented characteristic

4.5 TWO MODIFIED SHPB SETUPS

From the results of Section 4.3, it was found that the traditional SHPB setup and the traditional data processing method were unable to obtain the entire stress-strain curve of the aluminum foam when the thickness of the specimen is necessarily large. Hence, two modified SHPB setups, shown in Fig. 4.5, are proposed to overcome the maximum desired strain limitation problem. The effectiveness of these two modified SHPB setups and the proposed GA-based parameter identification method is numerically investigated. In practical applications, the dimensions of the modified SHPB can be slightly different from these two modified SHPB configurations. The strain gauges for these modified SHPB setups are still cemented at the pressure bars; however, the locations of these gauges are not very important for the proposed GA-based identification method. In the traditional SHPB setup, one of the strain gauges is required to be cemented at the middle of the incident bar to avoid the superposition of the incident and reflected waves. However, based on the proposed parameter identification method, only one strain gauge is needed.

(a) Modified SHPB setup 1

(b) Modified SHPB setup 2

Figure 4.5 Two modified SHPB setups

The pressure bars for the two modified SHPB setups are shorter than those of the traditional SHPB setup. Shorter pressure bars are preferable for multiple reflections of wave than the longer pressure bars. Instead of using the transmission bar, the modified SHPB setup 2 restrains the deformation of the specimen at the rear end. These modified SHPB setups are able to be implemented in laboratories.

The traditional data processing method is not applicable any more for these two modified SHPB setups because of the overlaps between the incident and reflected waves. Based on GA's powerful ability to solve optimization and identification problems, it is adopted as the data processing method in this study for the two modified SHPB setups. The original idea to employ the modified SHPB setup 2 is that only one strain history at any location of the pressure bar is sufficient to identify the entire stress-strain curve of the aluminum foam when the GA was adopted as the data processing method. This is also one of the important advantages of the proposed method. The advantages of using the GA-based method are illustrated by the numerical examples in Section 4.7.

4.6 PROBLEM DESCRIPTION AND SOLUTION PROCEDURE

In this study, identification of the dynamic properties of aluminum foams is a process to identify the critical parameters, which can be used to reconstitute the entire stress-strain curve of the aluminum foam. Identification of the dynamic properties of aluminum foams is characterized as an inverse problem for finding the five parameters from the measured responses of the SHPB test. The aim of this process is to minimize the sum error of the measured and simulated responses. The measured responses are obtained by numerical SHPB tests instead of physical test. The advantage of using numerical SHPB test is that the accuracy and efficiency of the proposed parameter identification method can be better demonstrated.

The developed GA-based identification method is an error-driven evolutionary process. It is also an optimization problem. It can be formulated as

$$W(E_{0a}, \varepsilon_{0a}, A, w, \varepsilon_c) = W(\varepsilon^m(t) - \varepsilon^e(t)) \rightarrow \min \quad (4.3)$$

where $\varepsilon^m(t)$ and $\varepsilon^e(t)$ are respectively the measured and the estimated strains at the incident/transmission bars. W is an error function in terms of the differences between the measured and the estimated strain signals at the incident/transmission bars.

The maximum and minimum optimization problems can be similarly treated after some proper transformation. For the convenience of using GA, the following formula is applied,

$$W' = C - W(E_{0a}, \varepsilon_{0a}, A, w, \varepsilon_c) \rightarrow \text{Max} \quad (4.4)$$

where C is a constant defined by the user for a specific problem. W' is a fitness function for the GA. The details for the definition of the fitness functions for different examples are given in the following section.

The solution procedure is similar to that shown in Fig. 3.5. The process starts with an initial population of randomly generated GA chromosome. Each

Chapter 4: Identification of Dynamic Properties of Aluminum Foams

chromosome in the initial population is then decoded into the five critical parameters. The encoding method is shown in Fig. 4.4. These parameters are substituted into the finite difference calculation to obtain the strains of the strain gauges which are cemented on the pressure bars. The initial population is subsequently evaluated according to the calculated strain; that is, the fitness of each individual is computed from the fitness function, i.e., Eqn. (4.4). The termination criterion is then checked to determine whether the desired parameters is found or the GA run has reached its allowed maximum number of generation. If the termination criterion is satisfied, the GA run will stop; otherwise, the performance of the GA genetic operations, i.e., reproduction, crossover and mutation will continue. The new population is created after this step, and the individuals are decoded and analyzed. Again, the individuals are evaluated according to the objective function to compare their fitness values. The process continues until the termination criterion is satisfied. The so-far-best chromosome obtained in this run will be used as the final result, and then be decoded to the five critical parameters which can be recognized for practical use.

4.7 ILLUSTRATIVE EXAMPLES

To illustrate the effectiveness of the GA-based parameter identification of dynamic material properties of aluminum foams, two numerical examples based on the two modified SHPB setups are investigated. Same as that in Chapter 3, each program will be run 10 time with same parameters, the results of the best run will be consider as the final results for this program. The ‘number of hit’ is defined as the number of successful run (the maximum fitness of the run is larger than or equals to 299.98) of 10 runs.

Before the GA is executed, the possible maximum and minimum values for the five critical parameters are needed. The lower and upper bounds of the stress-strain curves and the input stress-strain curve shown in Fig. 4.6 are obtained after

Chapter 4: Identification of Dynamic Properties of Aluminum Foams

careful definition of the maximum and minimum values of the parameters to include all possible stress-strain curves of the aluminum foams.

The control parameters for the GA run are as follows: the population size is 500; the maximum number of generation is 50; each subsequent generation of the run is created from the population of the preceding generation by performing 20% reproduction operation, 70% crossover operation and 10% mutation operation. The maximum number of generation is set as the termination criterion.

In order to identify the critical parameters of the dynamic properties of aluminum foams, the sum error of the measured signals collected from the gauges at the incident/transmission bars and the signals calculated from the chromosome of GA at the corresponding gauges is adopted to determine the fitness function. The fitness function for these two examples is as follows

$$W'(E, \varepsilon_0, A, w, \varepsilon_c) = 300 - \sum_{i=1}^{30000} |\varepsilon^m(t_i) - \varepsilon^e(t_i)| \quad (4.5)$$

Figure 4.6 Target curve and the upper and lower bounds

4.7.1 Modified SHPB Setup 1

Short pressure bars are preferred for the multiple reflection of the wave because the wave dispersion will become less when comparing that of the longer bars. So the modified SHPB setup uses short pressure bars. The details of the modified SHPB setup 1 are shown in Fig. 4.5(a). The material property of the pressure bars is the same as that used in Section 4.3, and the velocity of the striker bar is the same as that used in the traditional SHPB setup given in Section 4.3. The strain history of the gauges on the input bar and output bar are shown in Fig 4.7. Based on the strain history, the GA runs with the given parameters and fitness function. The evolution of both the best and the mean fitness of the population with increase of the generation number of the best run of the binary code GA and real code GA is shown in Fig. 4.8. The ‘number of hit’ of 10 runs for the binary code GA and the real code GA are 3 and 5, respectively.

Figure 4.7 Strain signals at gauges a1 and b1 for the modified SHPB setup 1

Figure 4.8 Fitness evolution for the modified SHPB setup 1

From Fig. 4.8, it can be observed that both the best fitness and the mean fitness rapidly increase in the initial generations, and the best fitness stabilizes at its optimal value after about 25 generations while the mean fitness slightly fluctuates in the later generations. This fluctuation may be induced by the randomness of the GA search. It is worth mentioning that this small fluctuation of the mean fitness does not hinder the GA from finding the optimum solution; conversely, it may help to maintain the diversity of the population and thus preventing the GA search from trapping in the local optima.

The target and the reconstituted (by binary code GA and real code GA) stress-strain curves are shown in Fig. 4.9. It can be observed that the reconstituted stress-strain curve is able to fit the target stress-strain curve well. It verifies that the proposed GA-based method is capable of identifying the entire stress-strain curve of the aluminum foam using the modified SHPB setup 1.

Figure 4.9 Comparison of the target and the reconstituted (by GA) stress-strain curves based on modified SHPB setup 1

4.7.2 Modified SHPB Setup 2

The configuration of modified SHPB setup 2 is shown in Fig. 4.5(b). The material property of the input bar is the same as the property of the pressure bar in modified SHPB setup 1. In order to compare the results obtained from these two modified SHPB setups, the striker with the same velocity of that used in the modified SHPB setup 1 is adopted for numerical simulation.

This new modified SHPB setup is extended from the modified SHPB setup 1. The outstanding advantage of using GA is that the data collected from only one strain gauge is enough for identification of the parameters. The strain history of the strain gauge cemented at the input bar is shown in Fig. 4.10.

Figure 4.10 Strain signal at gauge for the modified SHPB setup 2

The evolution of both the best and the mean fitness of the population with increase of the generation number of the best run of the binary code GA and real code GA is shown in Fig. 4.11. The ‘number of hit’ of 10 runs for the binary code GA and the real code GA are 5 and 6, respectively. Apparently, same conclusions as those drawn from Fig. 4.8 can be obtained.

The target and the reconstituted (by binary code GA and real code GA) stress-strain curves are shown in Fig. 4.12.

From Fig. 4.12, it is found that the reconstituted stress-strain curve fits the target stress-strain curve very well. Therefore, it is concluded that the modified SHPB setup 2 can be used for testing the compression behavior of aluminum foams, and the proposed GA-based method can identify the entire stress-strain curve of aluminum foams correctly.

Figure 4.11 Fitness evolution for modified SHPB setup 2

Figure 4.12 Comparison of the target and the reconstituted (by GA) stress-strain curves based on the modified SHPB setup 2

Chapter 4: Identification of Dynamic Properties of Aluminum Foams

It is also observed that with the same initial striker velocity, the two modified SHPB setups can obtain the entire stress-strain curve of aluminum foams as compared to the traditional SHPB setup. The obtained stress-strain curve is able to effectively show the densification process of aluminum foams, which is very important for practical use.

The strain history of the whole specimen for these two numerical examples is shown in Fig. 4.13.

Figure 4.13 Comparison of strain history of specimen of modified SHPB setups 1 and 2

It can be observed from Fig. 4.13 that with the same initial velocity and length of the striker bar, the modified SHPB setup 2 gives a larger maximum strain of the specimen than the modified SHPB setup 1. This is due to the fact that no energy is transmitted to the transmission bar when the modified SHPB setup 2 is adopted as the rear end of the specimen is fixed.

4.8 CONCLUSIONS

Two modified SHPB setups are proposed in this chapter. GA is incorporated with these two modified SHPB setups to identify the critical parameters for the determination of the entire stress-strain curve of aluminum foams. The numerical examples show that the proposed GA-based identification method can correctly reconstitute the entire stress-strain curve of aluminum foams, which includes the densification process. It can overcome the inherent disadvantage (maximum strain limitation) of the traditional SHPB setup and its corresponding data processing method. From the numerical results of these two examples, it can also be found that the real code GA outperforms the binary code GA for identifying the dynamic properties of aluminum foams.

The advantages of the proposed modified SHPB setups include that, first, the strain gauges can be cemented at any location of the pressure bars, and second, only one strain gauge is sufficient for identifying the critical parameters of the aluminum foam. Overlapping of the incident and reflected signals is never a burden in the modified SHPB tests. In addition, easy implementation of the modified SHPB setups in the laboratory with traditional SHPB setups is another merit of the proposed method.

Since aluminum foams are often used in impact absorption applications, their strain rate dependent properties are necessary for predicting their responses under such situations. Therefore, modeling and identification of the mechanical behavior of aluminum foams under rapidly changed loading, such as their loading rate sensitivity, will be one of the future works.

CHAPTER 5 FORCE IDENTIFICATION OF DYNAMIC SYSTEMS USING GENETIC PROGRAMMING

5.1 INTRODUCTION

An important type of inverse problem in structural mechanics is the identification of excitation forces from the measured structural responses. The identification of dynamic forces acting on a structure is an old problem, but has so far only been treated with partial success (Stelzner et al. 2001). Methods for such identification can be divided into two categories, direct methods and indirect methods. The direct methods use force transducers in the load paths at the point of force application. However, the introduction of force transducers can often greatly change the structural characteristics, resulting in inaccurate force prediction. Moreover, in many practical situations, direct measurements of excitation forces are not feasible, and it is difficult, if not impossible, to perform direct measurements or calculations of the external forces acting on the vibrating structures. Instead, vibrational responses can often be conveniently measured. Therefore, the indirect methods use other sensors, such as displacement transducers, velocity transducers and acceleration transducers, to measure the structural responses. These sensors can be placed at locations on the structure that may not necessarily correspond to the force input locations. Once the structural responses are obtained, the indirect methods employ certain back-calculation or system identification techniques to calculate the excitation force.

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

Many situations require indirect approach for force identification, which is more difficult (Starkey and Merrill 1989). As such it is often ill-posed and ill-conditioned (Groetsch 1993), making the work in this area slow and the gains modest. An overview of force identification techniques for the discrete and the continuous linear vibration systems was presented by Stevens (1987). The techniques were either frequency domain based or time domain based. Ory et al. (1986) used the Williams' method (Williams 1948) with a time integration scheme to estimate shock loading, and obtained good results for some discrete and distributed loading on a beam. Law et al. (1997) proposed a method based on modal superposition to identify the forces on a simply supported beam in the time domain. Law and Fang (2001) also presented a dynamic programming technique for the identification of moving forces on a simply supported beam. Liang et al. (2001) applied the fuzzy adaptive back-propagation algorithm, which combined fuzzy theory with artificial neural network technique, to the identification of restoring forces in non-linear vibration systems. Tao et al. (2001) described how to reconstruct the excitation force of a typical engine from the measured velocity data at mounting points using a combination of genetic and gradient-based algorithms. Huang (2001) used an algorithm based on the conjugate gradient method to estimate the unknown non-linear external forces in a damped system. Kammer (1998) presented a time domain method for estimating the discrete structural input force based on the measured structural response, and Stelzner et al. (2001) presented a modified time domain technique for the estimation of forces applied to an unrestrained structure. Both methods needed a vibration test of the structure with the given input force data and the sensor response data, to obtain the Markov parameters for the structure before the force estimation. Ma and his co-workers presented the applications of the Kalman filter with a recursive estimator to determine the impulsive loads (Ma et al. 1998), and the input force on a cantilevered plate (Liu et al. 2000). They also reported an on-line recursive method to estimate the input force on beam structures (Ma et al. 2003). All these applications needed to measure the dynamic responses of the system, including both the displacement and velocity of all degrees-of-freedom.

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

However, all the above force identification methods can not obtain the explicit force expression as some techniques needed a large quantity of dynamic response data for force identification and others needed extra experimental tests before the force identification procedure. Thus, the genetic programming (GP) based method is proposed to overcome these limitations.

GP is a technique devised for the automatic generation of computer programs by means of natural selection (Koza 1992). Although GP is an extension of genetic algorithm (GA) (Holland 1975), it is in some sense more powerful than GA. Compared to GA's encoding of individuals as fix model, GP's encoding of two-dimensional parse trees varying in size and complexity is more natural and flexible in representing the candidate solutions. When solving the problem which we do not know in advance the size and model of its best solution, the ability of GP to examine different size solutions is very important (Yang and Soh 2002). Generally, GP is good at global search for the model of the solution of the given problem. It has been empirically shown to be a powerful program-induction methodology and has been successfully applied to a large number of difficult problems, such as automatic design, pattern recognition, robotic control, synthesis on neural architectures, symbolic regression, factory job scheduling, electronic circuit design, signal processing, music and picture generation, and forecasting (Sheta and Mahmoud 2001; Tull et al. 2002). Soh and Yang (2000) introduced GP into civil engineering problem solving. They proposed a GP-based approach for simultaneous sizing, geometry and topology optimization of structures. In order to improve their GP's performance, Yang and Soh (2000) integrated a fuzzy logic controller into GP. They also presented a GP-based methodology for the automated optimum design of truss structures using an approach which is free from ground structures (Yang and Soh 2002).

In this chapter, due to its earlier mentioned advantages, GP is used to identify the excitation forces of dynamic systems. Compared to the conventional methods, the GP system can not only obtain the explicit mathematical expression of the force, but can also be used to forecast the force. This GP-based force

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

identification method does not require much knowledge about the mathematical transform nor further mechanics analysis of dynamic systems.

5.2 PROBLEM DESCRIPTION

In the present study, the finite element method (FEM) is adopted to construct the state-space model of dynamic systems. For FEM, a dynamic system can be considered as an n -degree-of-freedom system. Therefore, the differential equations of motion of the system in terms of mass, stiffness and damping matrices are

$$[M]\{\ddot{x}(t)\} + [C]\{\dot{x}(t)\} + [K]\{x(t)\} = \{f(t)\} \quad (5.1)$$

where $[M]$, $[C]$ and $[K]$ denote the $n \times n$ mass, damping and stiffness matrices, respectively; $\{f(t)\}$ is the $n \times 1$ input force vector; and $\{\ddot{x}(t)\}$, $\{\dot{x}(t)\}$ and $\{x(t)\}$ are the $n \times 1$ vectors of acceleration, velocity and displacement, respectively. The matrices $[M]$ and $[K]$ can be obtained from the FEM discretization. The matrix $[C]$ is obtained by assembling the matrices $[M]$ and $[K]$ as a proportional damping model.

Input force identification is a process of identifying the applied loadings from the measurements of system responses. The present input force identification method consists of two steps: determination of the system state-space model, i.e., Eqn. (5.1), and identification of the unknown forces based on the measured responses using the GP-based method. It is in fact an error-driven evolutionary process. The measured response used in this study is the displacement, velocity or acceleration of any degree-of-freedom.

The problem of force identification can be formulated as follows.

Find $\{f(t)\}$, such that

$$W(\{\dot{x}_f^m(t)\} - \{\dot{x}_f^e(t)\}, \{\ddot{x}_f^m(t)\} - \{\ddot{x}_f^e(t)\}, \{\dot{x}_f^m(t)\} - \{\dot{x}_f^e(t)\}) \rightarrow \min \quad (5.2)$$

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

where $\{x_f^m(t)\}$ and $\{x_f^e(t)\}$ denote the measured and estimated displacement of freedom f , respectively; $\{\dot{x}_f^m(t)\}$ and $\{\dot{x}_f^e(t)\}$ are the measured and estimated velocity of freedom f , respectively; and $\{\ddot{x}_f^m(t)\}$ and $\{\ddot{x}_f^e(t)\}$ are the measured and estimated acceleration of freedom f , respectively. The estimated displacement, velocity and acceleration can be obtained from Eqn. (5.1) for a trial solution of force.

5.3 FUNCTION AND TERMINAL SETS

In order to identify the excitation force from the dynamic response such as the displacement or the velocity or the acceleration of a structure, it is crucial to properly define the function and terminal sets with which the parse trees of GP are built. The terminal set can comprise the input to the GP program, as well as the constants supplied to the GP program. The function set can comprise the statements, operators, and functions available to the GP system. To obtain a useful representation with regard to solving a given problem, the function set and terminal set must fulfill two important requirements, *closure property* and *sufficiency property* (Fraser 1994).

In this study, the function set, F , contains the following mathematical operators.

$$F = \{+, -, *, \%, \sin, \cos\} \quad (5.3)$$

with % the protected division operator defined as: If (denominator=0) then (result=0) else (result=numerator/denominator).

The terminal set, T , for the connecting functions consists of

$$T = \{t, R\} \quad (5.4)$$

where t is time and R is a random number. In this study, R is defined as $R \in [-10.0, 10.0]$.

5.4 CROSSOVER AND MUTATION

A typical GP parse tree, according to the above function and terminal sets, is given in Fig. 5.1 in which the inner nodes represent the functions and the external nodes(leaves) represent the terminals. Using a LISP S-expression (Koza 1992), the aforementioned GP parse tree can be written as follows:

$(+ (* t 5.0) (\sin (\% t 2.0)))$

Figure 5.1 A typical GP parse tree

This expression, using a prefix notation, is read from left to right applying recursively each function to the next one or two arguments or sub-S-expression. It is equivalent to the following mathematical expression: $5.0 * t + \sin\left(\frac{t}{2.0}\right)$.

The GP crossover operation in this study is illustrated in Fig. 5.2, in which the highlighted sub-trees of the parents are swapped to generate two offspring. Since the entire sub-trees are swapped, and because of the closure property of the functions themselves, this genetic crossover operation always produce syntactically legal LISP S-expression as offspring regardless of the selection of parents or crossover points.

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

Parents

$$1.0 * (t + 2.0) + \sin\left(\frac{t}{2.0}\right)$$

$$5.0 * t * \left(t - \frac{t}{2.0}\right)$$

Crossover

Offspring

$$1.0 * (t + 2.0) + \frac{t}{2.0}$$

$$5.0 * t * \left(t - \sin\left(\frac{t}{2.0}\right)\right)$$

Figure 5.2 GP crossover operation

The GP mutation operation used in this chapter is illustrated in Fig. 5.3, in which the highlighted two inner nodes (functions) and one leave (terminal) of the parent are replaced by two other functions and one terminal, respectively. Usually,

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

two kinds of mutations are possible. In the first type, a function can only be replaced by a function and a terminal can only be replaced by a terminal. In the second type, an entire sub-tree can be replaced by another sub-tree. In this study, only the first type of mutation is used to avoid the code growth problem. The mutation has to satisfy the closure property, such as, the “*sin*” can only be replaced by “*cos*” if it has been selected to mutate.

Figure 5.3 GP mutation operation

5.5 FITNESS EVALUATION

In order to find the best explicit expression of the unknown force, it is necessary to establish a suitable evaluation criterion of the force estimated by the GP system. This criterion is called the objective function. In this chapter, the objective function is defined as the sum of all errors between the measured system responses and the estimated system responses derived from the estimated force that is decoded from the GP parse tree at each time point. This objective function W , can thus be written as

$$W = \sum_{i=1}^n \sum_{f=1}^l \left(\alpha_f |x_f^m(t_i) - x_f^e(t_i)| + \beta_f |\dot{x}_f^m(t_i) - \dot{x}_f^e(t_i)| + \gamma_f |\ddot{x}_f^m(t_i) - \ddot{x}_f^e(t_i)| \right) \quad (5.5)$$

in which $x_f^m(t_i)$, $x_f^e(t_i)$, $\dot{x}_f^m(t_i)$, $\dot{x}_f^e(t_i)$, $\ddot{x}_f^m(t_i)$, $\ddot{x}_f^e(t_i)$ are the actual and the estimated displacement, velocity and acceleration responses of freedom f at time t_i , respectively; n is the number of time points; and α_f , β_f and γ_f are the weight coefficients.

The force identification formulation in Eqn. (5.2) can then be expressed as follows:

Find the particular force expression $\{f(t)\}$ from T and F , such that the objective function $W \rightarrow \min$. (5.6)

In this study, the following equivalent form is used:

$$W' = \text{Const} - W \rightarrow \max \quad (5.7)$$

where Const is a constant number defined by the user for the specific problem.

5.6 SOLUTION PROCEDURE

Fig. 5.4 gives an overview of the GP-based search process for excitation force identification of dynamic systems. The process starts with an initial population of GP parse trees, which are generated by randomly selecting functions from the function set, i.e., Eqn. (5.3) and terminals from the terminal set, i.e., Eqn. (5.4), and using the ramped half-and-half method (Koza 1992). Each individual (GP parse tree) in the initial population is then decoded into force expression. The force expression is substituted into Eqn. (5.1) and the dynamic analysis is performed to obtain the system responses using the Wilson- θ method (Bathe 1982). The initial population is subsequently evaluated according to the calculated responses of the dynamic system; that is, the fitness value of each individual is computed from the objective function, i.e., Eqn. (5.7). The termination criterion is then checked to determine whether the desired force expression is found or the GP run has reached its allowed maximum number of generation. If the termination criterion is satisfied, the GP run will stop; otherwise, the next step is the performance of GP genetic operations, i.e., reproduction, crossover and mutation. The new population is created after this step, and the individuals are decoded and analysed. Again, the individuals are evaluated according to the objective function to compare their fitness values. The process keeps moving ahead until the termination criterion is satisfied.

Figure 5.4 Solution procedure for GP-based force identification method

5.7 ILLUSTRATIVE EXAMPLES

To illustrate the practicability and effectiveness of the GP-based force identification method in identifying unknown excitation forces, four numerical examples are investigated. In these examples, the function set and terminal set listed in Eqns. (5.3) and (5.4) are used. Based on the past experience of our research group (Soh and Yang 2000, Yang and Soh 2000, Yang 2001, Yang et al. 2005, Wang et al. 2005), the control parameters for the GP run are selected as follows: the population size is 1000 for Examples 1 and 2; 3000 for Examples 3 and 4; the maximum number of generation is 30 for Example 1 and 100 for Examples 2, 3 and 4; each subsequent generation of the run is created from the population of the

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

preceding generation by performing 20% reproduction operation, 60% crossover operation and 20% mutation operation. After the individual was selected to attend the mutation, 10% terminals and functions will be mutated. In this study, the depth of the GP parse tree of the initial population is limited to 17 and the length of the GP parse tree is limited to 100 at any generation. The maximum number of generation is set as the termination criterion. For each example, 20 runs of the same program is performed, the best result of these runs will be shown as the final result.

5.7.1 Example 1: Single-Degree-Of-Freedom Dynamic System

The first example is a single-degree-of-freedom dynamic system as shown in Fig. 5.5. The parameters in the dynamic equation of this simple vibration system are as follows: $[M]=100\text{kg}$, $[C]=85\text{N.s/m}$ and $[K]=140\text{N/m}$. The initial conditions are: $\{\ddot{x}(0)\} = \{\dot{x}(0)\} = \{x(0)\} = 0$. The actual excitation force $f(t) = 6.0\sin t$.

Figure 5.5 A single-degree-of-freedom vibration system

In this example, the measured system responses are simulated by the actual responses calculated from the actual force applied on the dynamic system. In practice, the measured responses may be obtained from sensor measurements. In this example, only the displacement response is used to evaluate the fitness function as the following,

$$W' = 100 - \sum_{i=1}^{1000} |x^m(t_i) - x^e(t_i)|, \quad (5.8)$$

$$\text{if } |x^m(t_i) - x^e(t_i)| > 0.1, \text{ then } |x^m(t_i) - x^e(t_i)| = 0.1.$$

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

Evolution of both the best and the mean fitness of the population with the increase of generation number is shown in Fig. 5.6. It can be observed that the best fitness steadily increases in the initial generations and remains at its optimal value throughout the later generations. However, although the mean fitness has the similar trend as the best fitness in the initial generations, it slightly fluctuates in the later generations rather than remains at a constant value. This may be attributed to the characteristic of randomness of the GP search. It is worth mentioning that this small fluctuation of the mean fitness does not hinder the GP from finding the optimum solution; conversely, it may help to maintain the diversity of the population and thus preventing the GP search from trapping in the local optima.

Figure 5.6 Fitness evolution for Example 1

The optimal force expression is found at Generation 6. Its LISP S-expression is $(+ t (- (- (- (- (+ t 2.8) (+ t 2.8)) (\sin t)) (\sin t)) (- t (* (\sin t) 8))))$. This LISP S-expression can be simplified as $6\sin t$, which is the same as the actual excitation force expression. The ‘number of hit’ (maximum fitness=100.0) of 20 GP runs is 14, that is, 14 of 20 runs can find this optimal expressions, i.e., $6\sin t$, their LISP S-expression are very different from each other in terms of length and complexity.

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

The comparisons of the time histories of the actual force and the estimated force decoded from the best individual in various generations, and the actual displacement and the estimated displacement are shown in Fig. 5.7. Fig. 5.7 can be viewed as the evolutionary process of the estimated force and the displacement. It is evident that with the increase of generation number, both the estimated force and displacement evolve to their respective optima.

(a)

(b)

(c)

(d)

(e)

(f)

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

(g)

(h)

Figure 5.7 Evolutionary process of estimated force and displacement for Example 1

5.7.2 Example 2: Three-Degree-Of-Freedom Dynamic System

The second example is a three-degree-of-freedom dynamic system as shown in Fig. 5.8. It is presented to illustrate the capability of GP in solving the force identification problem of multi-degree dynamic systems.

Figure 5.8 A three-degree-of-freedom dynamic system

The system parameters of this example are:

$$[M] = \begin{bmatrix} 100.2 & 0 & 0 \\ 0 & 100.2 & 0 \\ 0 & 0 & 100.2 \end{bmatrix} \text{ kg}, \quad [C] = \begin{bmatrix} 170 & -85 & 0 \\ -85 & 170 & -85 \\ 0 & -85 & 85 \end{bmatrix} \text{ N.s/m}$$

$$[K] = \begin{bmatrix} 280 & -140 & 0 \\ -140 & 280 & -140 \\ 0 & -140 & 140 \end{bmatrix} \text{ N/m},$$

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

The initial conditions are $\{\ddot{x}(0)\} = \{\dot{x}(0)\} = \{x(0)\} = 0$. The excitation force is assumed to apply at point 3, i.e., the third degree-of-freedom of the system, as

$f_3(t) = \frac{8.0}{1+t} \sin t$. Thus, the actual excitation force vector can be expressed as

$$\{f(t)\} = \begin{Bmatrix} 0 \\ 0 \\ \frac{8.0}{1+t} \sin t \end{Bmatrix}$$

Again, the measured system responses are simulated by the actual responses calculated from the actual excitation force.

The nine cases listed in Table 5.1 are investigated to show the effectiveness of the GP-based method. Table 5.1 is self-explanatory. For example, for Case 1, the displacement of the third degree-of-freedom is used as the measured system response to compute the fitness of the GP individuals; for Case 5, the velocity of the second degree-of-freedom is used; and for Case 9, the acceleration of the first degree-of-freedom is used.

Table 5.1 Cases studied in Example 2

Degree of freedom	Measured system response		
	Displacement	Velocity	Acceleration
3rd	Case 1	Case 2	Case 3
2nd	Case 4	Case 5	Case 6
1st	Case 7	Case 8	Case 9

For the cases using displacement as the system response, i.e., Cases 1, 4 and 7, the fitness function is the same as the one defined in Eqn. (5.8). For the cases using velocity, i.e., Cases 2, 5 and 8, and acceleration, i.e., Cases 3, 6 and 9, the fitness functions are defined in Eqns. (5.9) and (5.10), respectively.

$$W' = 100 - \sum_{i=1}^{1000} \left| \dot{x}^m(t_i) - \dot{x}^e(t_i) \right|, \quad (5.9)$$

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

$$\text{if } \left| \dot{x}^m(t_i) - \dot{x}^e(t_i) \right| > 0.1, \text{ then } \left| \dot{x}^m(t_i) - \dot{x}^e(t_i) \right| = 0.1.$$

$$W' = 100 - \sum_{i=1}^{1000} \left| \ddot{x}^m(t_i) - \ddot{x}^e(t_i) \right|, \quad (5.10)$$

$$\text{if } \left| \ddot{x}^m(t_i) - \ddot{x}^e(t_i) \right| > 0.1, \text{ then } \left| \ddot{x}^m(t_i) - \ddot{x}^e(t_i) \right| = 0.1.$$

The fitness evolution for each case of Example 2 is shown in Fig. 5.9. Similar variation trends of the fitness curves as depicted in Fig. 5.6 can be observed.

(Case 1)

(Case 2)

(Case 3)

(Case 4)

(Case 5)

(Case 6)

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

(Case 7)

(Case 8)

(Case 9)

Figure 5.9 Fitness evolution for Example 2

The evolutionary processes of the estimated force and displacement for Case 1, the estimated force and velocity for Case 5, and the estimated force and acceleration for Case 9 are shown in Figs. 5.10, 5.11 and 5.12, respectively. Again, similar to the trends shown in Fig. 5.7, the estimated force and displacement/velocity/acceleration evolve to their respective optima with the increase of generation number.

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

**Figure 5.10 Evolutionary process of estimated force and displacement
for Case 1 of Example 2**

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

**Figure 5.11 Evolutionary process of estimated force and velocity
for Case 5 of Example 2**

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

**Figure 5.12 Evolutionary process of estimated force and acceleration
for Case 9 of Example 2**

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

Another observation from Figs. 5.10, 5.11 and 5.12 is that although the optimal dynamic responses which are calculated from the corresponding optimal estimated forces are very close to the actual responses, the optimal estimated forces still have discriminable differences with the actual force. This implies that slightly different excitation forces may generate very similar, even non-discriminable, system responses. This is one difficulty in the force identification problem, which exists not only for the GP-based method, but also for the other conventional techniques.

The optimal force expressions for Case 1, 5 and 9 are listed as follows.

For Case 1, $f_3(t) = (* (-(\cos (\% (\%(\sin (* (* <-4.5> t)(\sin (\% <1.6> t)))) (+ (+ t <3.1>)(\sin t))) (\%(\cos(\cos(\sin(\sin(\cos (\% (\% (\% (\% <-3> t) (\% (- t(\cos t))(\cos(\sin(\sin (+ t t))))) (-(\sin(\sin t))(\cos t))) (- (+(\cos (\% (\% t t)(\cos t)))(\sin t)) (\% t t))))))) t))) (-(\cos(\sin (\% (* <-4.5> t) (+ t <3.8>)))) (+ (-(\sin (\% <1.6> t)) (* (\% <-3> t)(\sin <2.5>)))(\sin(\cos(\sin (\%(\sin <-0.7>)(\sin t)))))))(\sin t)))$;

For Case 5, $f_3(t) = ((+ (-(\sin t)(\sin (* (*(\sin t) (\% (* <4.1>(\sin t)) t))(\cos (*(\sin t) (+ (\% (+ t <-6.7>) (- <-6.2> <-3.5>))(\sin t)))))) (-(\cos (\%(\cos (+(\cos(\sin (+ t <-6.7>))) (- (\% t t) (\% t <1.9>)))) (- (- <-6.2> <-3.5>) (\% (+ t <-6.7>) (- <-6.2> <-3.5>))))(\cos (-(\cos (+(\cos(\sin (*(\sin t) (\% (* <4.1>(\sin t)) t)))) (- (\% t t) (\% t <1.9>)))) (+ (\% (+ t <-6.7>) (- <-6.2> <-3.5>))(\sin(\sin(\sin t)))))))$; and

For Case 9, $f_3(t) = ((+ (* (-(\cos t) (+ (-(\cos t) (+ <6.8> <-0.5>))(\sin t))) (\%(\cos (+(\cos <1.7>) (+(\cos <1.7>) (+ t <4.9>)))) (- (+ (+ <2.1> t)(\sin <2.4>)) (+(\sin t) (\% <0.4> t))))) (* (-(\cos(\cos t)) (+ (-(\sin t) (+ <6.8> <-0.5>)) (+ (+ (+(\cos <5.2>)(\sin(\cos t)))(\cos <1.7>))(\cos(\cos (+ <2.1> t)))))) (\%(\cos (+(\cos <1.7>) (+ t <4.9>))) (- (+ (* <2.1> t)(\sin <2.4>)) (+ (+(\cos (+(\cos <1.7>) (+ t <4.9>))) (\% <-5.5> <9.2>))(\cos <1.7>))))))$.

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

The evolutionary processes of the estimated force and force expression for the other cases are not presented here. Only, the time behaviors of the optimal estimated force for all cases, as compared to the actual force, are shown in Fig. 5.13. The number of hit (maximum fitness > 99.0) of 20 GP runs for these 9 cases is from 10 to 16.

(Case 1)

(Case 2)

(Case 3)

(Case 4)

(Case 5)

(Case 6)

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

(Case 7)

(Case 8)

(Case 9)

Figure 5.13 Actual and estimated forces of Example 2

It can be observed from Fig. 5.13 that although fluctuations of some estimated forces exist at the initial time stage, at the later time stage, all the estimated forces are very close to the actual force. Fig. 5.13 demonstrates that despite the aforementioned difficulty of the force identification problem, GP is able to obtain acceptable force expression for all cases. No matter what kind of system response, i.e., displacement, velocity or acceleration, is used as the measured response and no matter which point is used as the measurement point, i.e., point 1, point 2 or point 3, the GP-based force identification method can find satisfactory force expression.

Another finding from Fig. 5.13 is that there are no significant discrepancies between the qualities of the estimated forces obtained from using displacement,

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

velocity or acceleration as the measured system response for fitness evaluation. In other words, none of these three methods of using displacement, velocity and acceleration has shown superiority to the others. This observation is useful in practical force identification. In addition, results from the response data of point 3 (Cases 1, 2 and 3) are comparable to those from the data of point 1 (Cases 7, 8 and 9) and point 2 (Cases 4, 5 and 6). This means that the GP-based force identification method can use the response data from any point of the dynamic system, making it much more practical for actual application.

Finally, any single point data as the measured data is adequate for GP to identify the unknown excitation force of the dynamic system. This is an outstanding advantage of GP-based force identification method when compared to the conventional force identification techniques.

5.7.3 Example 3: Another Three-Degree-Of-Freedom Dynamic System

The third example is the same as Example 2, except for the mass matrix and the force expression. This time, we do not know the exact force expression in advance; and as an example, only the velocity of the second degree-of-freedom will be used as the measured value. The fitness function is the same as Eqn (5.9). The mass matrix is

$$[M] = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 5 \end{bmatrix}$$

At generation 98, the optimal result was found. The maximum fitness equals to 92.7145, and the length of the optimal force expression equals to 99. The actual and estimated responses and forces are shown in Figs. 5.14 and 5.15. The number of hit (maximum fitness > 98.0) of this example for 20 GP runs is 9.

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

As can be observed from Fig. 5.15, the actual force curve is much more complex than those in the previous examples. In practice, this excitation force curve could be plotted from the measured data.

Figure 5.14 Actual and estimated velocities of the second freedom of Example 3

Figure 5.15 Actual and estimated forces of Example 3

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

The optimal force expression obtained is as following,

$$f_3(t) = ((+ (* (- (-(\sin <-2.3>) (-(\sin t)(\sin (-(\cos t)(\cos(\sin(\cos t))))))) (- (\% <0.7> t) (-(\sin <-2.3>) (* (-(\cos (- (\% <0.7> t)(\sin t)))(\cos t)) (- (\%(\cos <-4.9>) (\% <0.7> t)) (- (-(\cos t)(\sin t))(\sin t)))))(\cos (+ (* (-(\sin (* <4.9> <8.1>)) (- (- (\% <0.7> t)(\sin (* <-6.8> <-1.9>))) (\% <0.7> t))) (- (\% t <-0.1>)(\cos t))) (\%(\cos (- (\% <0.7> t)(\cos (\% <0.7> t)))) (\%(\sin <-2.3>)(\sin(\sin t)))))) (\%(\cos (\% t <-0.1>))(\cos <-4.9>))))$$

From the result, it can be concluded that the GP-based force identification method proposed in this chapter can also identify complex forces.

5.7.4 Example 4: Frame Structure

In this example, a more complex structure is considered as the dynamic system, which is shown in Fig. 5.16. The parameters are: $m=3000$ Kg; $EI=3 \times 10^6$ N.m² for the columns; $EA=1 \times 10^9$ N for the beams; and proportional damping model is employed, $[C] = \alpha_d [M] + \beta_d [K]$, where $\alpha_d = 0$, and $\beta_d = 0.1$.

Figure 5.16 The frame model

Again, the exact force expression is not known in advance. For the three cases considered in this example, the displacement of point 1, velocity of point 2 and acceleration of point 3 are used as the measured values, respectively. One

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

thousand measured response data point are used in the fitness calculation. The fitness functions are shown in Eqns. (5.8), (5.9) and (5.10), respectively.

The actual forces are shown in Table 5.2. In the numerical simulation, time interval of 0.01s is used, but to save pages, time interval of 0.05s is shown in this table.

Table 5.2 The actual forces

Time(s)	Force(kN)	Time(s)	Force(kN)	Time(s)	Force(kN)	Time(s)	Force(kN)
0	0	2.5	0.358446	5	-3.33469	7.5	-5.20419
0.05	4.79237	2.55	1.81128	5.05	-4.59072	7.55	-6.24508
0.1	8.31609	2.6	2.69674	5.1	-5.22982	7.6	-6.32496
0.15	9.73006	2.65	2.73995	5.15	-5.10434	7.65	-5.19446
0.2	8.72342	2.7	1.90326	5.2	-4.16632	7.7	-2.83723
0.25	5.58145	2.75	0.396688	5.25	-2.49378	7.75	0.476969
0.3	1.10531	2.8	-1.37723	5.3	-0.29851	7.8	4.20562
0.35	-3.59206	2.85	-2.92959	5.35	2.09153	7.85	7.61912
0.4	-7.3619	2.9	-3.81413	5.4	4.27723	7.9	9.94652
0.45	-9.29907	2.95	-3.74599	5.45	5.85574	7.95	10.5515
0.5	-8.95714	3	-2.67959	5.5	6.49889	8	9.09819
0.55	-6.44834	3.05	-0.826	5.55	6.02626	8.05	5.66469
0.6	-2.40636	3.1	1.39462	5.6	4.4557	8.1	0.770794
0.65	2.17608	3.15	3.45358	5.65	2.01702	8.15	-4.69477
0.7	6.19298	3.2	4.84212	5.7	-0.8779	8.2	-9.63949
0.75	8.69313	3.25	5.19603	5.75	-3.70631	8.25	-12.9933
0.8	9.10516	3.3	4.38606	5.8	-5.93416	8.3	-13.9475
0.85	7.36643	3.35	2.55211	5.85	-7.12422	8.35	-12.1518
0.9	3.92689	3.4	0.073583	5.9	-7.02905	8.4	-7.82055
0.95	-0.37003	3.45	-2.51666	5.95	-5.64759	8.45	-1.71925
1	-4.49767	3.5	-4.65367	6	-3.23124	8.5	4.97375
1.05	-7.49485	3.55	-5.86816	6.05	-0.23653	8.55	10.9037
1.1	-8.69213	3.6	-5.89094	6.1	2.76669	8.6	14.8256
1.15	-7.86205	3.65	-4.71039	6.15	5.21176	8.65	15.8777
1.2	-5.25958	3.7	-2.57112	6.2	6.65107	8.7	13.7783
1.25	-1.54625	3.75	0.083946	6.25	6.84553	8.75	8.90199
1.3	2.37996	3.8	2.71325	6.3	5.80796	8.8	2.21221

*Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming***Table 5.3 The actual forces (Continued)**

1.35	5.60528	3.85	4.79386	6.35	3.79153	8.85	-4.93663
1.4	7.41658	3.9	5.92979	6.4	1.22685	8.9	-11.0856
1.45	7.46177	3.95	5.92793	6.45	-1.37641	8.95	-14.9839
1.5	5.8178	4	4.82712	6.5	-3.53993	9	-15.8572
1.55	2.95322	4.05	2.87711	6.55	-4.91336	9.05	-13.572
1.6	-0.40121	4.1	0.475602	6.6	-5.33575	9.1	-8.65844
1.65	-3.43925	4.15	-1.91976	6.65	-4.84949	9.15	-2.18682
1.7	-5.47533	4.2	-3.88375	6.7	-3.66693	9.2	4.47317
1.75	-6.10228	4.25	-5.10055	6.75	-2.10127	9.25	9.95561
1.8	-5.27399	4.3	-5.41125	6.8	-0.48191	9.3	13.1939
1.85	-3.29603	4.35	-4.8257	6.85	0.922858	9.35	13.6424
1.9	-0.73086	4.4	-3.50145	6.9	1.9563	9.4	11.3741
1.95	1.75515	4.45	-1.69878	6.95	2.58568	9.45	7.03724
2	3.56235	4.5	0.275067	7	2.87698	9.5	1.68179
2.05	4.3029	4.55	2.11974	7.05	2.94283	9.55	-3.49651
2.1	3.88518	4.6	3.58404	7.1	2.8825	9.6	-7.43413
2.15	2.51918	4.65	4.49167	7.15	2.73438	9.65	-9.43228
2.2	0.645069	4.7	4.75108	7.2	2.45737	9.7	-9.2907
2.25	-1.19375	4.75	4.35295	7.25	1.94844	9.75	-7.31476
2.3	-2.49812	4.8	3.36119	7.3	1.092	9.8	-4.1999
2.35	-2.94272	4.85	1.90177	7.35	-0.17439	9.85	-0.82749
2.4	-2.45488	4.9	0.151521	7.4	-1.79965	9.9	1.97242
2.45	-1.22403	4.95	-1.67373	7.45	-3.58679	9.95	3.62949

The actual and estimated responses and forces are shown in Figs. 5.17, 5.18 and 5.19, respectively. The number of hit (maximum fitness>98.0) of these three cases for 20 GP runs are 15, 14 and 11, respectively.

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

(a)

(b)

Figure 5.17 Comparison of the actual and estimated forces and displacements of point 1 of Example 4

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

(a)

(b)

Figure 5.18 Comparison of the actual and estimated forces and velocities of point 2 of Example 4

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

(a)

(b)

Figure 5.19 Comparison of the actual and estimated forces and accelerations of point 3 of Example 4

Chapter 5: Force Identification of Dynamic Systems Using Genetic Programming

The three obtained optimal force expressions are as following,

From displacement of point 1, $f(t) = ((-(-(\sin(-(t < -6.7>))(* < -9.4> t)))(-(-(-(-(\sin(* < -8.2> t))(\sin(-(t < -0.5>))(* < -9.4> t))))(-(\sin(-(\cos < -5.1>)(* < -9.6> t))(-(\sin(* < -8.2> t))(\sin(* t < -7.8>)))))(\sin(* t < -7.9>))(-(\sin(-(* t < -8.6>))(\% < 9.5> < -10>))(-(\sin(* < -8.2> t))(\sin(-(\cos < -5.1>)(* < -9.6> t)))))(\sin(-(\cos < -5.2>)(* < -9.6> t))))(-(-(\sin(* < -8.2> t))(\sin(-(* t < -7.9>))(\% < -8.3> < -1.4>)))(\sin(-(t < -0.4>))(* < -9.4> t)))));$

From velocity of point 2, $f(t) = ((+(-(\sin(* (t t) (- < -3.8> < -9>)))(\sin(-(\cos(\sin(* (* < 1.1> < -10>))(\cos(\cos < -4.5>)))))(* (t t) (- < -3.8> < -9>))))((+(-(-(\cos(* t < 9.5>))(\sin(-(\cos(\cos(* t < 9.5>)))(* (t t) (- < -3.8> < -9>)))))(\sin(-(\cos(% t < 5.1>))(-(* t < 9.5>))(- < -3.8> < -9>)))))(-(\cos(-(\cos(% t < 7.8>))(* t < 9.5>)))(* (-(* (%(\cos(\cos < -4.5>))(* < 1.4> < -9.3>))(* t < 9.5>))(\cos(\cos(% t < 5.9>)))))(%(\cos(* t < 8>))(\cos(\cos(\cos(\cos < -4.5>)))))))); and$

From acceleration of point 3, $f(t) = ((+(+(\cos(* < -8> t))(+(\cos(-(* t < -9.5>))(-t < -9.9>)))(+(+(\cos(* t < -9.3>))(+(\cos(-(* t < -9.5>))(-t < -10>)))(\cos(* < -8> t))))((+(* (-(-(* (% t < -0.9>)) (t t)))(\cos(\cos(-t < -3.8>)))))(\cos(\cos(-t < -3.8>)))(% (+(\cos(* < -8> t))(\cos(* t < -9.5>)))(* < 10> < -3.1>)))(\cos(* < -8> t))))((+((\cos(-(* t < -9.5>))(-t < -10>)))(+(\cos(-(* t < -10>))(-t < -4.1>)))(\cos(-(* t < -9.3>))(\cos(% < 6.6> < 0.3>)))))))).$

From these results, it is found that the proposed GP-based force identification method can also correctly identify the complex force acting on the complex structure. Again, no matter what kind of system response is used as the measured response, and no matter which point is used as the measurement point, the GP-based force identification method can find satisfactory force expression.

5.8 CONCLUSIONS

This chapter presents a GP-based method for the identification of unknown excitation forces of dynamic systems. The numerical examples show that the GP system is able to identify the excitation forces of dynamic systems. Comparisons between the measured and the estimated forces validated the proposed GP-based method. One advantage of this method is that no matter which point is used as the measurement point, and no matter what kind of system response is used as the measured response, it is able to identify the unknown force with good accuracy. More importantly, only single point data is needed. Another merit of the proposed method is that it can obtain the explicit force expression.

It is hoped that the GP-based force identification method could provide a useful tool for quantifying discrete input force experienced during structural operation, which can be used in many different subsequent analyses, such as structural load analysis and health monitoring.

CHAPTER 6 GP FOR GROUND MOTION IDENTIFICATION OF DYNAMIC SYSTEMS

6.1 INTRODUCTION

How to identify the ground motion from the measured structural response is an important and typical type of inverse problem in structural dynamics. In certain situations, the ground motion time history may not be recorded or retrieved. But if the structural responses are available, how to reconstruct the unknown ground motion from the known dynamic responses becomes the critical issue. It is found that little agreement has been reached in the past 30 years on ground motion estimation relation studies (Douglas 2003). Rofooei et al. (2001) used the generalized nonstationary Kanai-Tajimi model to describe and simulate the ground motion time histories, with the moving time-window technique embedded to evaluate the time varying parameters of the model. Choi and Seo (2002) performed a series of tests and dynamic analyses on the traditional Korean wooden houses for the intensity estimation of typical large historical earthquake records. Typical earthquake time histories recorded on soil and rock sites were used as the input for the tests. A prototypical wooden house was analyzed for multiple time histories which matched Ohsaki's ground response spectra. Thrainsson and Kiremidjian (2002) proposed a method to obtain the acceleration time histories of the horizontal earthquake ground motion by inverting the discrete Fourier transform, which was defined by modeling the probability distribution of the Fourier phase differences conditional on the Fourier amplitude. Cahouet et al. (2002) presented a method

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

utilizing all available imperfect position and force measurements to extract the optimal estimated joint acceleration. A weighted least-square optimization approach was employed to provide the optimal acceleration distribution which was most consistent with the position and force data. Lee and Han (2002) developed efficient neural-network-based models for the generation of artificial earthquake and response spectra. Five neural-network-based models were proposed for replacing the traditional processes. Cao et al. (1998) developed an approach to the identification of loads acting on aircraft wings, which used an artificial neural network to model the load-strain relationship in structural analysis. Paola and Zingales (2000) presented a digital simulation procedure for a n -variate vector process representing the ground acceleration in different spatial point locations. Zerva and Beck (2003) proposed a methodology for the identification of random field models for the description of seismic ground motions recorded over extended areas. It was shown that parametric coherency estimates identified by means of this approach were stable and insensitive to the amount of smoothing performed on the empirical data.

So far, little research has been carried out on using the measured structural response to identify the unknown ground motions of dynamic systems. It is hoped that the genetic search method proposed in this chapter could be useful for this type of problem solving.

GP (Koza 1992) is a branch of GAs (Holland 1975), and it is in some sense, more powerful than GAs. The main difference between GP and GA is the representation of the solution. GP creates computer programs as the solutions. GA creates a string of numbers that represent the solution. GP has been empirically shown to be a powerful program-induction methodology and has been successfully applied to a large number of difficult problems, such as automatic design, pattern recognition, robotic control, synthesis on neural architectures, symbolic regression, factory job scheduling, electronic circuit design, signal processing, and music and picture generation (Gritz and Hahn 1997; Koza et al. 1997). Combined with a hybrid simplex/simulated annealing algorithm, GP was applied to the identification

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

of nonlinear dynamic models from the simulated experimental data by Gray et al. (1996a). They also used GP to evolve an algebraic expression as part of an equation representing the measured input-output response data (Gray et al. 1996b), in which a GP-based method was proposed to identify the parts of the nonlinear differential equations describing a dynamic system along with their numerical parameters.

This hierarchical search and optimization method has been introduced into civil engineering problem solving by Soh and Yang (2000). They proposed a GP-based approach for the simultaneous sizing, geometry, and topology optimization of structures. In order to improve their GP's performance, Yang and Soh (2000) integrated a fuzzy logic controller into GP. They also presented a GP-based methodology for the automated optimum design of truss structures using an approach which is free of ground structures (Yang and Soh 2002). When solving a problem which we do not know in advance the size and model of its best solution, the ability of GP to examine different size solutions is very important (Yang and Soh 2002).

In this chapter, GP is used to identify the ground motion of dynamic systems as it is a robust global search technique. It is to be highlighted that this GP-based ground motion identification method does not require much knowledge about the mathematical transform and further mechanics analysis of dynamic systems. This GP system can not only obtain the explicit mathematical expression of the ground motion, but can also be used for ground motion prediction.

6.2 PROBLEM DEFINITION

In this study, the finite element method (FEM) is adopted to construct the state-space model of dynamic systems. After discretization, a dynamic system can be considered as an n -degree-of-freedom system. Therefore, the differential equation of motion of the system under ground acceleration is

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

$$[M]\{\ddot{x}(t)\} + [C]\{\dot{x}(t)\} + [K]\{x(t)\} = -[M]\{I\}a(t) \quad (6.1)$$

where $[M]$, $[C]$, $[K]$ denote the $n \times n$ mass, damping and stiffness matrices, respectively; $\{\ddot{x}(t)\}$, $\{\dot{x}(t)\}$ and $\{x(t)\}$ are the $n \times 1$ vectors of acceleration, velocity and displacement, respectively; $\{I\}$ is the identity vector; and $a(t)$ is the ground acceleration. The matrices $[M]$ and $[K]$ can be obtained from the FEM discretization. The matrix $[C]$ is obtained by assembling the matrices $[M]$ and $[K]$ as a proportional damping model.

In this chapter, ground motion identification is defined as a process of identifying the ground acceleration from the measured system responses. The measured response used in this study is the displacement, velocity or acceleration of any degree-of-freedom. The proposed GP-based ground motion identification method, which is in fact an error-driven evolutionary process, consists of the following two steps:

- 1) determination of the system state-space model, i.e., Eqn. (6.1); and
- 2) identification of the unknown ground acceleration using the GP-based method.

Therefore, the problem of ground motion identification can be formulated as follows

Find $a(t)$, such that the defined sum error

$$W(x_f^m(t), x_f^e(t), \dot{x}_f^m(t), \dot{x}_f^e(t), \ddot{x}_f^m(t), \ddot{x}_f^e(t)) \rightarrow \min \quad (6.2)$$

where $\{x_f^m(t)\}$ and $\{x_f^e(t)\}$ denote the measured and estimated displacements of freedom f , respectively; $\{\dot{x}_f^m(t)\}$ and $\{\dot{x}_f^e(t)\}$ the measured and estimated velocities of freedom f , respectively; and $\{\ddot{x}_f^m(t)\}$ and $\{\ddot{x}_f^e(t)\}$ the measured and estimated accelerations of freedom f , respectively. The estimated displacement, velocity and acceleration can be calculated from Eqn. (6.1) for the trial solution of ground acceleration.

6.3 GP-BASED METHOD FOR GROUND MOTION IDENTIFICATION

6.3.1 Overview

The proposed method uses GP to identify the ground motion of the dynamic systems. After choosing the proper terminal set and function set, a population of randomly generated ground motion expression are represented in GP parse-trees, using the proper encoding method. The genetic operators, consisting of reproduction, crossover and mutation, act on them over a number of generations, evolving increasingly fitter ground motion expression. The evolution of fitter ground motion proceeds under selection pressure, which depends on the relative fitness of the individual designs, calculated using the fitness function.

The preparatory steps for GP runs entail determining (1) the set of terminals; (2) the set of functions; (3) the fitness measure; (4) the parameters for controlling the run; and (5) the termination criterion and the method of result designation for the run. The first two steps define the search space of GP, and the fitness measure determines the outcome of the search. The parameters for controlling the run are the population size, the maximum number of generations, and the probability of performing the various generic operators. Completing a maximum number of generations is taken as the termination criterion, and the best-so-far individual is designated as the result of the run.

The terminal set and function set for the GP parse tree building are the same as that employed in Section 5.3. The crossover and mutation operators are the same as that employed in Section 5.4.

6.3.2 Initialization

The generation of each individual in the initial population is done by randomly generating a GP parse tree with ramped half and half method (Koza 1992). A typical GP parse tree and its represented mathematical expression are given in Fig. 6.1, in which the inner nodes represent the functions and the external nodes (leaves) represent the terminals.

Figure 6.1 A typical GP parse tree and its represented expression

Diversity is valuable in GP populations. The individual parse-trees in the initial population generated with ramped half and half method will have a wide variety and the generating process itself is stochastic. The variety of the initial population is very important for preventing the genetic search from any premature convergence (Goldberg 1989; Koza 1992).

6.3.3 Fitness Evaluation

In order to find the explicit expression of the unknown ground motion, it is necessary to establish a suitable fitness evaluation criterion for all GP individuals (parse trees). This criterion is called the objective function. In this chapter, for Example 1, the objective function is defined as the sum of relative errors between the measured system responses and the estimated system responses derived from

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

the estimated ground acceleration that is decoded from the GP parse tree at each time point, and for Examples 2 and 3, the objective function is defined as the sum of the square of errors between the measured system responses and the estimated systems responses. These two kinds of objective functions W , can thus be written as

$$W = \sum_{i=1}^n \sum_{f=1}^l \left(\alpha_f \left| \frac{x_f^m(t_i) - x_f^e(t_i)}{x_f^m(t_i)} \right| + \beta_f \left| \frac{\dot{x}_f^m(t_i) - \dot{x}_f^e(t_i)}{\dot{x}_f^m(t_i)} \right| + \gamma_f \left| \frac{\ddot{x}_f^m(t_i) - \ddot{x}_f^e(t_i)}{\ddot{x}_f^m(t_i)} \right| \right) \quad (6.3a)$$

$$W = \sum_{i=1}^n \sum_{f=1}^l \left(\alpha_f (x_f^m(t_i) - x_f^e(t_i))^2 + \beta_f (\dot{x}_f^m(t_i) - \dot{x}_f^e(t_i))^2 + \gamma_f (\ddot{x}_f^m(t_i) - \ddot{x}_f^e(t_i))^2 \right) \quad (6.3b)$$

in which $x_f^m(t_i)$, $x_f^e(t_i)$, $\dot{x}_f^m(t_i)$, $\dot{x}_f^e(t_i)$, $\ddot{x}_f^m(t_i)$, $\ddot{x}_f^e(t_i)$ are the measured and the estimated displacement, velocity and acceleration responses of freedom f at time t_i , respectively; n is the number of time points; and α_f , β_f and γ_f are the weight coefficients.

After defining the proper objective function, Eqn. (6.2) can be expressed as follows

Find the particular ground acceleration expression $a(t)$ from function set F and terminal set T , such that the objective function $W \rightarrow \min$. (6.4)

The following equivalent expression is adopted in this study:

$$W' = \text{Const} - W \rightarrow \max \quad (6.5)$$

where Const is a constant number defined by the user for the specific problem.

In order to calculate the dynamic response of the dynamic system under the known ground acceleration, the standard Wilson- θ method (Bathe 1982) is used after proper transformation of Eqn. (6.1).

Multiplying Eqn. (6.1) by matrix $[M]^{-1}$, we obtained

$$[I]\{\ddot{x}(t)\} + [M]^{-1}[C]\{\dot{x}(t)\} + [M]^{-1}[K]\{x(t)\} = -\{I\}a(t) \quad (6.6)$$

where $[I]$ is the unit matrix.

Eqn. (6.6) is the standard expression for the Wilson- θ method. Dynamic responses can then be calculated after the ground acceleration is obtained, which can be decoded from the GP parse tree.

6.3.4 Solution Procedure

The solution procedure can be described as following

1. Generate the initial population;
GP parse trees of the initial population are generated by randomly selecting functions from the function set, and terminals from the terminal set, and using the ramped half-and-half method (Koza 1992).
2. Iteratively perform the following sub-steps until the termination criterion has been satisfied.

- a. Execute each individual in the population and assign it a fitness value according to how well it solves the problem;

The ground acceleration expression is substituted into Eqn. (6.6) and the dynamic analysis is performed to obtain the system responses using the Wilson- θ method (Bathe 1982). The population is subsequently evaluated according to the calculated responses of the dynamic system, that is, the fitness value of each individual is computed from the objective function, i.e., Eqn. (6.5).

- b. Create the new population of individuals by applying the genetic operators. The operators are applied to individuals in the population with a probability based on fitness.

6.4 ILLUSTRATIVE EXAMPLES

To illustrate the practicality and effectiveness of the GP based ground motion identification method, three numerical examples are investigated. All parameters selections are based on the past experience of our research group (Soh and Yang 2000, Yang and Soh 2000, Yang 2001, Yang et al. 2005, Wang et al. 2005). It should be noted that GP is a stochastic search by nature. Given the inherent randomness of all stochastic search methods, GP may arrive at different optimal solutions for different runs. Therefore, it is necessary to perform multiple runs to obtain the best result. Multiple runs of GP can also enable us to examine how often GP can obtain the optimal or near-optimal solution.

For each case or sub-case studied in the following two examples, 20 runs of GP search are performed. The best result of these runs will be shown as the final result.

6.4.1 Example 1: Three-Degree-Of-Freedom Dynamic System

In the first example, two cases of a three-degree-of-freedom dynamic system are investigated. Consider a three-story building as shown in Fig. 6.2(a), with a simplified model as shown in Fig. 6.2(b).

Figure 6.2 Model of three-degree-of-freedom system and ground acceleration

The dynamic matrices are:

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

$$[M] = \begin{bmatrix} 71.57 & 0 & 0 \\ 0 & 71.57 & 0 \\ 0 & 0 & 71.57 \end{bmatrix} \text{ Kg} \quad [C] = \begin{bmatrix} 121.43 & -60.71 & 0 \\ -60.71 & 121.43 & -60.71 \\ 0 & -60.71 & 60.71 \end{bmatrix} \text{ N.s/m}$$

$$[K] = \begin{bmatrix} 200 & -100 & 0 \\ -100 & 200 & -100 \\ 0 & -100 & 100 \end{bmatrix} \text{ N/m}$$

At the instant the ground motion occurs, the structure is assumed to be at rest. Therefore, the initial conditions of the dynamic structure are: $\{\ddot{x}(0)\} = \{\dot{x}(0)\} = \{x(0)\} = 0$. The necessary parameters and descriptions of the GP are shown in Table 6.1.

Table 6.1 Tableau for GP parameters

Function set	+, -, *, %, sin, cos
Terminal set	t (the independent variable), R (random number)
Population size	1000
Number of generation	100
Possibility of reproduction	20%
Possibility of crossover	60%
Possibility of mutation	20%
Creation type of initial population	Ramped half and half
Parse tree depth of initial population	≤ 10
Parse tree length of all generations	≤ 100
Termination criterion	Generation=100
Maximum fitness	1000

Case 1: Ground acceleration $a(t) = 8.0 \sin t$

In the first case, the actual ground acceleration is $a(t) = 8.0 \sin t$ and the displacement of point 3 is used as the measured system response.

The measured system responses are simulated by the actual responses calculated from the actual ground acceleration applied on the dynamic structure, as the measured responses can only be obtained from sensor measurements. Therefore, for illustrative examples, the actual responses will be referred to as the measured responses. In this case, only the displacement response of point 3 is used to evaluate the fitness function as follows,

$$W' = 1000 - \sum_{i=1}^{1000} \left| \frac{x^m(t_i) - x^e(t_i)}{x^m(t_i)} \right|, \quad (6.7)$$

$$\text{if } x^m(t_i) \leq 0.1, \text{ use } |x^m(t_i) - x^e(t_i)| \text{ to replace } \left| \frac{x^m(t_i) - x^e(t_i)}{x^m(t_i)} \right|$$

$$\text{if } \left| \frac{x^m(t_i) - x^e(t_i)}{x^m(t_i)} \right| > 1.0, \text{ then } \left| \frac{x^m(t_i) - x^e(t_i)}{x^m(t_i)} \right| = 1.0.$$

Evolution of both the best and the mean fitness of the population with the increase of generation number is shown in Fig. 6.3. It can be observed that both the best fitness and the mean fitness rapidly increase in the initial generations, and the best fitness steadily remains at its optimal value while the mean fitness slightly fluctuates in the later generations. This fluctuation may be attributed to the characteristic of randomness of the GP search. It is worth mentioning that this small fluctuation of the mean fitness does not hinder the GP from finding the optimum solution; conversely, it may help to maintain the diversity of the population and thus preventing the GP search from trapping in the local optima.

Figure 6.3 Fitness evolution for Case 1

The optimal acceleration expression is found at Generation 6. The optimal GP parse tree and its LISP S-expression are shown in Fig. 6.4. The LISP S-expression can be simplified as $8\sin t$, which is the same as the actual ground acceleration expression.

Figure 6.4 Optimal GP parse tree and LISP S-expression for Case 1

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

The comparisons of the time histories of the actual (measured) ground acceleration and the estimated ground acceleration, and the actual (measured) displacement and the estimated displacement are shown in Fig. 6.5. The estimations are decoded from the best individual of the GP population in Generations 0, 2, 4, and 6; hence, Fig. 6.5 can be viewed as the evolutionary process of the estimated ground acceleration and the estimated displacement. It is evident that both the estimated ground acceleration and displacement evolve to their respective optima with the increase of generation number (from Generation 0 to 2 to 4), and reach the optima at Generation 6. The “number of hit” (maximum fitness = 1000.0) for 20 GP runs is 8, that is, 8 out of 20 runs found this optimal result.

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

Figure 6.5 Evolutionary process of estimated ground acceleration and displacement at point 3 for Case 1 of Example 1

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

Case 2: Ground acceleration $a(t) = \frac{2.0}{1+t} \sin t$

The second case is to identify a complex ground acceleration $a(t) = \frac{2.0}{1+t} \sin t$ using various dynamic responses at points 1, 2 and 3. The purpose is to show the robustness of the proposed GP method.

Same as Case 1, the measured system responses are simulated by the actual responses calculated from the actual ground acceleration. The following nine sub-cases listed in Table 6.2 are investigated to show the effectiveness of the GP-based ground motion identification method. Table 6.2 is self-explanatory. For example, for Case 2.1, the acceleration of the first degree-of-freedom is used as the actual (measured) system response to compute the fitness of the GP individuals; and for Case 2.5 and 2.9, the velocity of the second degree-of-freedom and the displacement of the third degree-of-freedom are used, respectively.

Table 6.2 Sub-cases studied in Case 2

Degree of freedom	Measured dynamic system response		
	Acceleration	Velocity	Displacement
1st	Case 2.1	Case 2.2	Case 2.3
2nd	Case 2.4	Case 2.5	Case 2.6
3rd	Case 2.7	Case 2.8	Case 2.9

For the sub-cases using displacement as the system response, i.e., Cases 2.3, 2.6 and 2.9, the fitness function is the same as the one defined in Eqn. (6.7). For the sub-cases using velocity, i.e., Cases 2.2, 2.5 and 2.8, and acceleration, i.e., Cases 2.1, 2.4 and 2.7, as the system response, the fitness functions are similar to Eqn. (6.7), except with the velocity and the acceleration replacing the displacement in Eqn. (6.7), respectively.

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

The fitness evolution for each sub-case of Case 2 is shown in Fig. 6.6. It can also be observed that both the best fitness and the mean fitness for each sub-case rapidly increase in the initial generations, and the best fitness steadily remains at its optimal value while the mean fitness slightly fluctuates in the later generations. Besides, the steady best fitness for each sub-case almost reaches the defined maximum fitness (1000.0). This indicates that the proposed GP-based ground motion identification method can accurately identify the unknown ground motion from any actual (measured) response of the dynamic system.

(Case 2.1)

(Case 2.2)

(Case 2.3)

(Case 2.4)

(Case 2.5)

(Case 2.6)

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

(Case 2.7)

(Case 2.8)

(Case 2.9)

Figure 6.6 Fitness evolution for Case 2

The evolutionary processes of the estimated ground acceleration and the respective system response for Cases 2.1, 2.5 and 2.9 are shown in Figs. 6.7, 6.8 and 6.9, respectively. Similar conclusion as that drawn from Fig. 6.5 can be drawn from Figs. 6.7, 6.8 and 6.9, i.e., the estimated ground acceleration and displacement(Case 2.1)/velocity(Case 2.5)/acceleration(Case 2.9) approach their respective optima with increase of generation number, and at last, satisfactory results (but not the exact solution) are obtained for all cases.

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

Figure 6.7 Evolutionary process of estimated ground acceleration and acceleration at point 1 for Case 2.1

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

Figure 6.8 Evolutionary process of estimated ground acceleration and velocity at point 2 for Case 2.5

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

Figure 6.9 Evolutionary process of estimated ground acceleration and displacement at point 3 for Case 2.9

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

The optimal ground acceleration expressions for Cases 2.1, 2.5 and 2.9 are listed as follows.

For Case 2.1,

$$a(t) = (*(\sin t) (\% <6.5> (-(\cos (\sin (\% t t))) (- (* (- (\% <-3.3> <-2.9>) (- t (\% t t))) (\% <6.5> (- (\% t t) (- (\% (+ t <7.8>) <-9.7>) (\cos (\% (- (\% (- (\% (+ t <7.8>) <-9.7>) (- t (\% (- t (+ (* t (\% (-(\sin t) (* t (\% (-(\sin t) (* t (\% (* (\% (+ t <7.8>) <-9.7>) (* t (\% t <7.8>))) <-9.7>))) <-9.7>))) <-9.7>)) <-9.7>)) <-9.7>)) <2>) (- t (\cos (- (\% t <7.8>) (- t (\% t <7.8>))))) <-9.7>))))) (+ (+ t <7.8>) <-0.4>))))) ;$$

For Case 2.5,

$$a(t) = (\sin (\%(\sin t) (*(\sin(\cos(\cos(\sin (\%(\sin(\cos t)) (+(\sin t) (*(\sin(\cos(\cos(\sin (\% (\%(\sin t)(\sin (\%(\sin(\cos t))(\cos(\cos (*(\cos (\%(\cos t) (* (* (+ t <0.6>) (* (+ t <0.6>) (+ t <0.6>))) (+ t <0.6>))))(\sin (\%(\sin t) (*(\cos (+ t <0.6>)) (+ t <0.6>)))))))) (*(\sin(\cos(\cos (\%(\sin (+ t <0.6>)) (* (+ t <0.6>) (\sin (\sin (\sin (\cos t))))))))(\sin (\%(\sin t) (*(\cos (+ t <0.6>)) (+ t <0.6>)))))))) (+ t <0.6>))))))) (+ t <0.6>))))) ; and$$

For Case 2.9,

$$a(t) = (\sin(\sin (-(\sin (- (\%(\sin t) t) (-(\sin (\%(\sin (- (\%(\sin t) t) (\%(\sin t) (+ t t)))) t)) (\%(\sin t) t)))) (-(\sin (\% (\% (\% (- (-(\sin(\sin t)) (\%(\sin t) t))(\sin (\%(\sin t) t))) t) t) (\%(\sin t)(\sin t)))) (-(\sin (\% (*(\sin (- (- t t) (\%(\sin t)(\sin t)))) (\% (-(\sin(\sin t)) (\%(\sin t) t)) t)) (+ t t))) (- t t))))) .$$

The evolutionary processes of the estimated ground acceleration expression for the other cases are not presented. Only the time behavior of the optimal estimated ground acceleration, as compared to the actual (measured) ground acceleration, and behavior of the optimal estimated system response, as compared to the actual (measured) values, for Cases 2.2, 2.3, 2.4, 2.6, 2.7 and 2.8 are depicted in Fig. 6.10. The numbers of hit (maximum fitness > 980) for 20 runs for all 9 sub-cases 2.1-2.9 are 3, 6, 5, 6, 6, 4, 5, 7, and 6, respectively.

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

(Case 2.2)

(Case 2.3)

(Case 2.4)

(Case 2.6)

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

(Case 2.7)

(Case 2.8)

Figure 6.10 Comparison of actual and estimated ground accelerations and system responses for Cases 2.2, 2.3, 2.4, 2.6, 2.7 and 2.8

It can be observed from Figs. 6.7, 6.8, 6.9 and 6.10 that although the optimal dynamic responses which are calculated from the corresponding optimal estimated ground acceleration are very close to the actual (measured) responses, the optimal estimated ground accelerations still have discriminable differences with the actual (measured) ground acceleration, especially at the initial time stage. This implies that slightly different ground accelerations may generate very similar, even non-discriminable, system responses. But, it is also apparent from Figs. 6.7, 6.8, 6.9 and 6.10 that although fluctuations of some estimated ground motions exist at the very short initial time stage, at the later time stage, all the estimated ground motions are very close to the actual (measured) ground acceleration. These results demonstrate that GP is able to obtain acceptable ground acceleration expression for all cases. It

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

can also be seen from Figs. 6.7, 6.8, 6.9 and 6.10 that no matter which system response, i.e., displacement, velocity or acceleration, is used as the measured response and no matter which point is used for measurement, i.e., point 1, point 2 or point 3, the GP-based ground motion identification method can find satisfactory ground motion expression.

From Figs. 6.7, 6.8, 6.9 and 6.10, it can also be seen that results from displacement as the actual (measured) data (i.e., Cases 2.3, 2.6 and 2.9) is not as good as those from velocity (i.e., Cases 2.2, 2.5 and 2.8) and acceleration (i.e., Cases 2.1, 2.4, 2.7) as the actual (measured) data at the very short initial time stage. However, none of these three methods of using displacement, velocity and acceleration as the actual (measured) data is superior over the others at the later time stage. All of these three methods can obtain similar satisfactory results, so there is no special restriction for what kind of actual (measured) data is used. This observation is useful and meaningful in ground acceleration identification practices. In addition, results from the response data of point 1 (i.e., Cases 2.1, 2.2 and 2.3) are comparable to those from the data of point 2 (i.e., Cases 2.4, 2.5 and 2.6) and point 3 (i.e., Cases 2.7, 2.8 and 2.9). This means that the GP-based ground motion identification method can use the response data from any point of the dynamic system, making it much more practical for actual application.

Finally, it is to be highlighted that any single point data as the measured data is adequate for GP to identify the ground acceleration of the dynamic system. This is another merit of the proposed GP search method.

6.4.2 Example 2: Frame Structure

In this example, a more complex structure is considered as the dynamic system, which is shown in Fig. 6.11. The parameters are: $m=3000$ Kg; $EI=3\times 10^6$ N.m² for the columns; $EA=1\times 10^9$ N for the beams; and proportional damping model is employed, $[C]=\alpha_d[M]+\beta_d[K]$, where $\alpha_d=0$, and $\beta_d=0.1$.

Figure 6.11 The frame model

This time, the exact ground acceleration expression is not known in advance. The GP control parameters are the same as that shown in Table 6.1 except that the population size is now 3000. In this example, the displacement of measurement point 1 and the velocity of measurement point 2 are used as the measured values, respectively. The two fitness functions for identifying the ground acceleration by using the value measured from points 1 and 2 are defined as follows,

$$W' = 100 - \sum_{i=1}^{1000} \left(x_1^m(t_i) - x_1^e(t_i) \right)^2, \quad (6.8)$$

if $\left(x_1^m(t_i) - x_1^e(t_i) \right)^2 > 0.1$, then $\left(x_1^m(t_i) - x_1^e(t_i) \right)^2 = 0.1$.

$$W' = 100 - \sum_{i=1}^{1000} \left(\dot{x}_2^m(t_i) - \dot{x}_2^e(t_i) \right)^2, \quad (6.9)$$

if $\left(\dot{x}_2^m(t_i) - \dot{x}_2^e(t_i) \right)^2 > 0.1$, then $\left(\dot{x}_2^m(t_i) - \dot{x}_2^e(t_i) \right)^2 = 0.1$.

where, $x_1^m(t_i)$ and $x_1^e(t_i)$ are the measured and estimated displacement of measurement point 1, respectively; and $\dot{x}_2^m(t_i)$ and $\dot{x}_2^e(t_i)$ are the measured and estimated velocity of measurement point 2, respectively.

When the displacement of point 1 is adopted as the measured dynamic response, at generation 97, the optimal ground acceleration is found. The maximum fitness equals to 99.3860, and the length of the optimal ground motion expression equals to 100. The actual and estimated displacements of measurement point 1 and

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

ground accelerations are shown in Figs. 6.12 and 6.13, respectively. When the velocity of point 2 is adopted, at generation 92, the optimal result is found. The maximum fitness equals to 99.9394, and the length of the optimal ground motion expression equals to 100. The actual and estimated velocities of point 2 and ground accelerations are shown in Figs 6.14 and 6.15, respectively. The numbers of hit (maximum fitness > 99.0) of these two cases for 20 GP runs are 12 and 8, respectively.

**Figure 6.12 Actual and estimated displacements of measurement point 1 of
Example 2**

Figure 6.13 Actual and estimated (by measurement point 1) ground accelerations of Example 2

Figure 6.14 Actual and estimated velocities of measurement point 2 of Example

Figure 6.15 Actual and estimated (by measurement point 2) ground accelerations of Example 2

The optimal ground acceleration expression obtained is as following,

From the displacement of measurement point 1, $a(t) = ((+((+(*(<2>t)(\cos(*t<-10>)))((+((+(*(-<10>t)(\cos(+(-(*t<-10>)(*(<1.8>t))(\cos<-8.3>)))(\%t\ t)))(\cos(-(-(*t<-10>)(*(<2.3>t)(\cos(+(\%t<-9>)(\cos(\sin(\cos(+(*<9.7>t)(*(<2>t)))))))(\cos<-3.3>)))(\cos(+(-(-(*t<-10>)(*(<2.4>t)(\cos<-9.4>)))(\%t\ t)))))(\cos(-(-(*t<-10>)(*(<2.3>t)(\cos(+(\%t<-9.4>)(\cos(\sin(\cos(+(*<9.7>t)(*(<2.4>t)))))))(\cos<3.6>))))))));$ and

From the velocity of measurement point 2, $a(t) = ((+((+(-(+(\cos(*(<-10><-1.7>)(-<-1.5>t)))(\cos(*(<-10><-1.7>)(-<-1.5>t)))(\%(*(<-1.7>t)(\cos(\%t<0.1>)))(\cos<5.4>)))(+(-(\cos(*(<-8.8><-3.6>)(-<2.2>t)))(-(\cos(-(-<-1.1>t)(*(<-1.3><-10>)(-<6.1>t)))(\cos(*$

$$\begin{aligned}
& (+ <-10> <-1.7>) (- <-1.5> t)))) (-(\cos (* (+ <-8.8> <-3.5>) (- <-2.2> t)))) \\
& (-(\cos (- (- <-1.1> t) (* (+ <-1.3> <-10>) (- <-1.1> t)))) (\cos (* (+ <-10> <- \\
& 1.6>) (- <-5.5> t)))))) (\cos (* (+ <-8.8> <-3.6>) (- <-2.2> t))))))
\end{aligned}$$

From these results, it can be concluded that the proposed GP-based search method is able to obtain satisfactory ground motion acting on the complex structure from any known dynamic response (i.e., displacement of measurement point 1, or velocity of measurement point 2). The advantages of the proposed GP search method which are verified by Example 1 are also verified.

6.4.3 Example 3: Multi-storey building

In this example, a 10-storey tall building (Li et al. 2004) shown in Fig. 6.16 will be considered as the dynamic system. This building is modeled as a uniform shear beam as described in Ref. (Clough and Penzien 1993). The flexural rigidities of all columns within one storey $\sum EI = 1.62 \times 10^8 \text{ N.m}^2$, and the storey mass is $3.6 \times 10^4 \text{ Kg}$. The damping model is same as that adopted in Example 2 at Section 6.4.2.

Figure 6.16 The 10-storey building model

Chapter 6: GP for Ground Motion Identification of Dynamic Systems

This time, the exact ground acceleration expression is not known in advance. The GP control parameters are the same as that used in Example 2 at Section 6.4.2. In this example, the velocity of the 10th storey is used as the measured value. The fitness function for identifying the ground acceleration by using the value measured from the 10th storey is defined as follows,

$$W' = 1000.0 - \sum_{i=1}^{1000} (\dot{x}_{10}^m(t_i) - \dot{x}_{10}^e(t_i))^2, \quad (6.10)$$

if $(\dot{x}_{10}^m(t_i) - \dot{x}_{10}^e(t_i))^2 > 1.0$, then $(\dot{x}_{10}^m(t_i) - \dot{x}_{10}^e(t_i))^2 = 1.0$.

where, $\dot{x}_{10}^m(t_i)$ and $\dot{x}_{10}^e(t_i)$ are the measured and estimated velocity of the 10th storey, respectively.

At generation 88, the optimal ground acceleration is found. The maximum fitness equals to 989.8383, and the length of the optimal ground motion expression equals to 100. The actual and estimated velocities of the 10th storey and ground accelerations are shown in Figs 6.17 and 6.18, respectively. The numbers of hit (maximum fitness > 980.0) for 20 GP runs is 3.

Figure 6.17 Actual and estimated velocities of the 10th story

Figure 6.18 Actual and estimated ground accelerations of Example 3

The optimal ground acceleration expression obtained is, $a(t) = ((+(-(-(+(-(-(-(-(+(\sin(*<8.6>t))(\sin(\sin(*<10>t)))(-(+(\sin(*t(\sin(\sin(\sin(\sin(*<2.1>(\sin(*X<-9.8>)))))))(\%<5.7><8.4>))(-(*t(\sin(*t<-9.8>)))(*(-3.9>(\sin(+(\sin(\sin(\sin(\sin(*t<9.2>)))))(\sin(\sin(*<8.7>t)))))))(\sin(\sin(*t(\sin(\sin(+(\%t<-5.8>)(*<8.7>t))))))<-8.2>)(\sin(*<8.7>t))(\cos(-(*t(\sin(+(\%t<-9>)(\%<10><7.2>)))))(*<1.9>(\sin(\sin(*t<-9.8>))))))<6.9>)(\sin(*<8.1>t))))$

From this example, it can be found that the proposed GP-based search method is able to correctly reconstitute the ground motion acting on the 10-storey building from the known dynamic response. This indicates that the proposed GP-based search method is potentially a promising technique for ground motion identification of multi-storey building using known dynamic response.

6.5 CONCLUSIONS

Accurate knowledge of the discrete ground motion acting on a structure during its operation is vital for correspondingly accurate numerical simulation of its response. This chapter presents a genetic search for ground motion of dynamic systems based on FEM. The numerical examples show that the GP system is able to identify the ground motion of a three-degree-of-freedom dynamic system, a frame structure and a multi-storey building. Comparisons between the actual (measured) and the estimated ground accelerations validated the proposed GP-based ground motion identification method. One advantage of this method is that no matter which point is used for measurement, and no matter which system response is used as the measured response, it is able to identify the unknown ground motion with good accuracy. More importantly, only single point data is needed. Another merit of the proposed method is that it can obtain the explicit ground motion expression, depicting its potential for ground motion prediction. From the problem solving procedure, it is apparent that there is no need for any further mathematical knowledge on solving inverse problems and for mechanics analysis. This shows that it is an easy tool for solving ground motion identification problems.

It is believed that the proposed genetic search for ground motion of dynamic systems provides a useful and easy tool for quantifying discrete ground motion experienced during structural operation, which can be used in many different subsequent analyses, such as structural failure analysis, earthquake estimation and structural health monitoring. Computational time is currently one drawback in using the proposed method. Future research will be to resolve this problem.

CHAPTER 7 INTEGRATED LOCAL SEARCH OPERATORS AND GP FOR DYNAMIC EXCITATIONS IDENTIFICATION

7.1 INTRODUCTION

GP is one of the most useful, general-purpose problem solving techniques. It has been used to solve a wide range of engineering problems, such as symbolic regression, data mining, and structural optimization. It is one instant of the class of techniques called evolutionary computation, which are based on insights from the study of natural selection and evolution. GP is an extension of the conventional GA, but it is in certain sense more powerful than GA. GA involves encoded strings that represent particular problem solutions, while GP breeds executable computer programs.

The standard GP is inspired by the Darwinian natural selection theory. In the 19th century, the Darwinian theory was challenged by Lamarck, who proposed that environmental changes throughout an organism's life cause the internal variations of the organism that are transmitted to the offspring. Lamarckian theory hypothesizes how the organisms pass down the knowledge and experience they acquired and is essentially a theory of directed variation. Although Lamarckian theory has not been observed in biological history, its convincing power is illustrated by the evolution of our society whereby knowledge and idea are passed from one generation to another through culture and language (Gen and Cheng 1997). However, GP, the artificial organisms, can certainly benefit from the advantages of

*Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations
Identification*

the Lamarckian theory. By letting some individuals' good "experience" pass down to the offspring, GP's ability to focus on the most promising areas can be improved. The Lamarckian theory has been introduced into GAs by Grefenstette (1991), and Kennedy (1993) gave an explanation for this hybrid. In this chapter, the proposed integrated GP methods incorporating three local search operators (LSOs), i.e., the GA, the linear bisection search (LBS) and the nonlinear programming (NLP), are based on both the Darwinian and Lamarckian theories. In these proposed methods, the standard GP, based on the Darwinian theory, performs the global search in the problem domain, while the LSOs vary a number of selected good GP individuals before they are injected into the next generation, imitating the directed variation in the Lamarckian theory. It is directed variation because the LSOs improve the fitness of the selected individuals by moving them to the local optima.

When solving problems that we do not know in advance the size and structure of its best solution, the ability of GP to examine different size solutions is very important. When standard GP is used, the generated GP individuals may include a number of internal parameters. These randomly generated parameters can have significant effects on the performance of the GP individuals. Generally, GP is good at global search for the structure of the solution of the given problem, however, for the problems that need "fine-tuning" of the solution parameters, GP is not as good as GA (Yang and Soh 2002). These inspire our attempt to integrate GA with GP. It is hoped that this novel integration may have higher effectiveness and efficiency than either of the original methods for certain problems, and thus could be used as an alternative global search and optimization tool for engineering problem solving. There are several methods of integration of the standard GP and GA search. Since the LSOs can significantly improve the search efficiency, even the effectiveness of the genetic search methods, in the proposed inGAP method, GA was used as a local search method of the standard GP. Another promising integration of the standard GP and GA search is described in the section on future works. The other two LSOs, i.e., LBS and NLP, are also studied to examine the performance of integrating GP and different LSOs.

Chellapilla et al. (1998) and Birru et al. (1999) embedded various LSOs into the self-adaptive evolutionary programming (EP) and the fast EP to investigate the effectiveness of LSOs in EP. They observed that the LSOs can statistically and

*Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations
Identification*

significantly enhance the performance of EP both in terms of the rate of convergence and the quality of the final solutions obtained. Niimi and Tazaki (2000) proposed a rule generation technique from a database using GP combined with an associated rule algorithm. Howard and D'Angelo (1995) proposed a hybrid of GA and GP (GA-P) for symbolic regression. In their method, a GA string, representing the coefficients of an expression, was attached to a standard GP parse tree to form a chromosome (GP individual). Sanchez (2000) extended the GA-P to interval GA-P to search for the algebraic expression that best approximated the experiment data. The GA-P integration is different from ours since the GA was not used as the LSO, but to perform the global search together with the GP. Generally, the GA-P can be treated as a genotype combination of GA and GP, i.e., a GA-P chromosome consists of a standard GP parse tree and a GA string. In the proposed methods, the integration of LSOs, including the GA, with the GP is a phenotype combination, i.e., LSOs operate on the selected chromosomes based on their fitness. The GP performs the global search while the LSOs explore the neighborhood of the current solution provided by the GP, looking for better solutions.

7.2 LOCAL SEARCH OPERATORS

Three kinds of variation operators, namely the GA, the LBS and the NLP are adopted as the LSOs for the standard GP in this Chapter.

7.2.1 GA

The detailed description of GA can be found in Chapter 2. Hence, it will not be repeated here. The integration of standard GP and GA is named as inGAP.

7.2.2 LBS

Yang et al. (2005) proposed a hybrid EP algorithm, which uses the macro mutation operator, the local LBS operator and the crossover operator to improve the

Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations Identification

capability of the standard EP algorithm. The results show that the employment of LBS operator can improve the search efficiency of EP for problems with large search space.

In the integrated GP method, the LBS operator is applied to each parameter in the GP individual. For the j -th parameter x_j , a variation is applied as $x'_j = x_j \pm s_j$, where $\pm$ is selected according to the gradient g_j . For example, for a maximum problem, if g_j is greater than zero, $x'_j = x_j + s_j$, otherwise, subtraction is adopted. If the mutated chromosome is not better than its parent, s_j reduces to $s_j/2$ until a better solution is found or the mutation step s_j is less than a preset small number which depends on the expected accuracy. It is expected that a point closer to a local optimum can be found by the LBS. Bisection method is employed to adjust the mutation step because it requires no specific knowledge about the fitness landscape. Thus, the LBS operator is a general method to be applied to any optimization problem. The integration of standard GP and local LBS is named as GP+LBS.

7.2.3 NLP

The detailed description of NLP (Hooke and Jeeves' method) can be found in Section 3.5.2. Hence, it will not be repeated here. The integration of standard GP and NLP is named as GP+NLP.

7.3 INTEGRATION OF LSOS AND GP

7.3.1 General Description

The proposed integration method incorporates LSOs as an add-on to the standard GP loop of recombination and selection. The LSO was implemented as an evolutionary operator that operates on a parent to generate an offspring. With this

Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations Identification

integration method, LSO is applied to some selected GP individuals to move them to the local optima before injecting them into the new population. GP is used to perform global exploration among a population, while LSO is used to perform local exploitation of selected individuals. GP is more powerful at identifying the model of the solution and GA is better at optimizing the parameters of the solution (Yang 2001, Yang and Soh 2002). So, the integration of GA and GP will combine the advantages of both GA and GP. LBS and NLP are good at local search, so they will be adopted as LSOs to fine-tune the parameters. The proposed integration method is thus an attempt to combine the advantages of both LSOs and GP.

The proposed integration method breeds computer programs to solve problems by executing the following three main steps:

- 1) Generate an initial population of random compositions of functions and terminals of the problem (i.e., computer programs).
- 2) Iteratively perform the following sub-steps until the termination criterion is satisfied:
 - a) Execute each program in the population and assign it a fitness value according to how well it solves the problem.
 - b) Create a new population of computer programs by applying the following four genetic operations. The operations are applied to the computer program(s) in the population chosen with a probability based on fitness.
 - i) Reproduction: Copying an existing computer program into the new population.
 - ii) Crossover: Creating two new computer programs by genetically recombining randomly chosen parts of two existing programs.
 - iii) Mutation: Creating a new computer program by changing one or several sub-tree(s) in an existing computer program.
 - iv) GA/LBS/NLP: Creating a better computer program by tuning the numerical terminals of the selected GP individual through GA/LBS/NLP.

Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations Identification

- 3) The best computer program that appeared in any generation (i.e., the best-so-far individual) is designated as the result of integrated system. This result may be a solution (or an approximate solution) to the problem.

Fig. 7.1 shows the flowchart of the integration method, where the index i refers to an individual in the population of size M , and the variable Gen is the number of the current generation.

Figure 7.1 Flowchart of integration method

Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations Identification

Comparing the proposed integration method with the standard GP, it can be observed that an LSO actually performs as an extra genetic operation in GP. This LSO operation is only performed on some GP individuals, probabilistically selected based on the fitness, i.e., the performance of the GP individuals. Therefore this integration method can be viewed as phenotype integration of LSO and GP.

In the inGAP, different from the conventional GAs, the initial population of GA in the hybrid GP is not completely randomly generated. One GA chromosome will include all correctly arranged numerical terminals of the selected GP individual, other GA chromosomes will be randomly generated within the range and accuracy of the numerical terminals adopted in the GP. In the GA operation, the elitism scheme, by which the best chromosome in the current generation is always copied into the next generation, is employed. The adopted initialization of the population and elitism strategy can ensure that the GA operation can generate new solution which is better or at least not worse than the selected GP individual in the current generation.

In the GP+LBS and GP+NLP, the LBS and NLP operators are respectively performed on the parameter strings encoded from the GP individuals selected based on their fitness values. After the LBS and NLP operators are performed, the generated new individual will be better or at least not worse than the selected GP individual in the current generation. This optimized individual will be copied into in next generation.

A well-known problem in GP is the tendency to generate larger and larger programs over time (Tackett 1993; Nordin and Banzhaf 1995; Soule and Foster 1998), called the bloat or code growth problem. This is harmful since it results in larger solutions than necessary. Moreover, it increasingly slows down the rate at which new individuals can be evaluated. Thus, keeping the size of trees small is generally an implicit objective of GP. So in the proposed integrated system, the size of the individual is controlled by adapting the size of crossover until the size of

Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations Identification

the individual is not larger than the preset size. When the mutation operator is applied to an individual, as a function can only be replaced by a similar function and a terminal can only be replaced by a similar terminal, the size of the individual can also be controlled.

7.3.2 Encoding between LSOs and GP

Before the proposed integration method is adopted, it is crucial to properly define the function and terminal sets with which the parse trees of GP are built. To obtain a correct and useful representation with regard to solving a given problem, the function set and terminal set must fulfill two important requirements: closure property and sufficiency property.

In this chapter, the function set, F , contains the following mathematical operators.

$$F = \{+, -, *, \%, \sin, \cos\} \quad (7.1)$$

with % the protected division operator defined as: If (denominator=0) then (result=0) else (result=numerator/denominator).

The terminal set, T , for the connecting functions consists of

$$T = \{t, R\} \quad (7.2)$$

where t is time and R is a random number. In this study, R is defined as $R \in [-10.0, 10.0]$ with 0.1 accuracy.

One crucial issue in the integration system implementation is how to encode a GP individual into a numerical string, which can be recognized by these LSOs. The following encoding method is proposed to solve this problem. A typical GP parse tree according to the above function and terminal sets is given in Fig.7.2, in which the inner nodes represent the functions and the external nodes(leaves) represent the terminals. Using a LISP S-expression, the aforementioned GP parse tree can be written as follows:

Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations
Identification

$$(-(*((+0.3\ t)\ 2.8)\ (\cos(*(-1.1\ 6.0)))))(\sin(\%(t\ 9.6))))$$

$$((0.3 + t) + 2.8) * (\cos(-1.1 * 6.0)) - \sin\left(\frac{t}{9.6}\right)$$

GP individual

Encoding

0.3	2.8	-1.1	6.0	9.6
-----	-----	------	-----	-----

LSO's numerical string

Figure 7.2 Encoding from GP individual to LSO's numerical string

This expression, using a prefix notation, is read from left to right applying recursively each function to the next one or two arguments or sub-S-expression. It is equivalent to the following mathematical expression:

$$((0.3 + t) + 2.8) * (\cos(-1.1 * 6.0)) - \sin\left(\frac{t}{9.6}\right).$$

As shown in Fig. 7.2, a typical GP individual consists of functions, +, -, *, %, *sin* and *cos* and terminals, *t*, 0.3, 2.8, -1.1, 6.0 and 9.6. For the numerical terminals, the encoding is straightforward. The encoding method between the GP individual and the LSO's numerical string is shown in Fig. 7.2, where the numerical strings consists of all the numerical terminals in a GP parse tree. In this chapter, LSOs will not be used to fine-tune the symbolic terminals in GP as the GP mutation operation can be employed to alter the

*Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations
Identification*

symbolic terminals. Thus, in this chapter, numerical strings only consist of the numerical terminals in the selected GP individuals.

7.4 APPLICATION TO FORCE IDENTIFICATION OF DYNAMIC SYSTEMS

7.4.1 Problem Description

To show the flexibility and the capability of the InGAP, the GP+LBS and the GP+NLP methods, they were used to solve the force identification problems described earlier in Chapter 5. The problem description, terminal set and function set, genetic operators and fitness evaluation have been described in Section 5.3, 5.4, 5.5 and 5.6 respectively; hence, they will not be repeated here.

7.4.2 Solution Procedure

Fig. 7.3 gives an overview of the inGAP, the GP+LBS and the GP+NLP method-based search process for force identification. The process starts with an initial population of parse trees, which are randomly generated using the ramped half-and-half method (Koza 1992). Each individual in the initial population is then decoded into force expression and analysed by the Wilson- θ method (Bathe 1982) to obtain the system responses. Unless the force expression considered is extremely simple, such as a constant or a simple linear expression, or one is incredibly lucky, the initial population will not contain the desired optimum force expression. The initial population is then evaluated according to the given responses of the dynamic system. The next step is the performance of integration system's genetic operations, i.e., reproduction, crossover, mutation and GA/LBS/NLP. The new population is created after this step, and the individuals are decoded and analysed. Again, the individuals are evaluated according to the objective function to compare their

Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations
Identification

fitness values. The process keeps moving ahead until the desired force expression is found or the run reaches its allowed maximum number of generation.

Figure 7.3 Solution procedure for dynamic force identification

The process can be divided into the following steps:

- 1) Initialize the population of parse-trees.
- 2) Individuals of the initial population are generated by randomly selecting functions from the function set and terminals from the terminal set.
- 3) Decode each individual into real force expression and perform dynamic analysis.
- 4) Wilson- θ method is used to obtain the dynamic responses of the structure.
- 5) Evaluate the fitness value of each individual.

Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations Identification

- 6) The program is terminated if the termination criterion is satisfied. Here, the maximum number of generation is set as the termination criterion.
- 7) Perform genetic operations, i.e., reproduction, crossover, mutation and GA/LBS/NLP.
- 8) Return to step (2) and repeat the process until the termination criterion is satisfied.

7.4.3 Illustrative Examples

In order to investigate the capability and effectiveness of the proposed integration methods, two illustrative examples for dynamic force identification of dynamic systems are presented. The parameters for the proposed integration methods used in these two examples are shown in Table 7.1. The accuracy of LBS and NLP is set to 0.1 (which is the accuracy of the random number used in GP in this chapter) for searching the local optimum for the selected chromosome. The LSOs were adopted to the top 2 individuals of each generation in this chapter. In order to compare the performance of each program, same number of fitness evaluation (300,000) for each program (standard GP, inGAP, GP+LBS, GP+NLP) is run, and each program with different random seeds (which were automatically generated by the system time) is run for 20 times.

Example 1: A frame structure

In this example, a frame structure is considered as the dynamic system, which is shown in Fig. 5.16. The parameters and damping model are same as those used in Section 5.7.4.

The exact force expression is assumed to be unknown in advance but its time-history data are given. There are totally 1000 data points of the excitation force from 0 to 9.99s with time interval of 0.01s. The excitation force-time curve is

Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations Identification

plotted in solid line in Fig. 7.4. The acceleration of measurement point 2, is used as the measured value. The fitness function is shown in Eqn. (7.3).

$$W' = 1000 - \sum_{i=1}^{1000} (\ddot{x}_2^m(t_i) - \ddot{x}_2^e(t_i))^2, \quad (7.3)$$

$$\text{if } |\ddot{x}_2^m(t_i) - \ddot{x}_2^e(t_i)| > 1.0, \text{ then } |\ddot{x}_2^m(t_i) - \ddot{x}_2^e(t_i)| = 1.0.$$

where, $\ddot{x}_2^m(t_i)$ and $\ddot{x}_2^e(t_i)$ are the measured and estimated acceleration of measurement point 2, respectively.

Table 7.1 Control parameters for GP and GA

	GP	GA
Population size	3000	40
Possibility of reproduction	5%	5%
Possibility of crossover	85%	85%
Possibility of mutation	5%	5%
Creation type of initial population	Ramped half and half (Koza 1992)	Nil
Parse tree depth of initial population	≤ 7	Nil
Parse tree length of all generations	≤ 100	Nil
Selection methods	Tournament selection with tournament size of 10	Roulette wheel selection
Termination criterion	Number of fitness evaluation = 300,000	The fitness value of the best individual remained unchanged for three generations
Maximum fitness	1000.0	1000.0

The comparison of the capability of the standard GP and the proposed integration methods for Example 1 are shown in Table 7.2. The ‘number of hit’ is defined as the number of successful run (the maximum fitness of the run is larger than or equals to 980.0) of 20 runs. It can be observed from Table 7.2 that the average fitness, best fitness and the ‘number of hit’ are improved when the integration methods were used instead of the standard GP. This indicates that all the

*Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations
Identification*

proposed integration methods outperform the standard GP for force identification problems.

**Table 7.2 Comparison of standard GP and integrated GPs for dynamic force
identification for Example 1**

Run	GP	inGAP	GP+LBS	GP+NLP
1	956.351	994.163	930.831	927.821
2	970.671	979.038	956.987	928.084
3	961.61	991.584	925.713	980.058
4	921.198	960.425	921.559	965.975
5	988.423	965.076	989.928	946.14
6	979.775	935.94	947.003	964.974
7	909.137	950.587	967.146	985.474
8	912.752	998.842	982.336	939.027
9	911.396	964.988	955.348	974.634
10	907.827	993.257	947.587	990.992
11	945.088	993.196	987.809	952.31
12	982.653	988.676	966.951	989.053
13	881.61	955.134	992.316	927.08
14	901.887	972.06	976.938	959.756
15	891.762	985.716	956.397	931.393
16	929.927	938.798	963.034	994.381
17	953.829	948.115	994.588	957.508
18	903.275	987.968	986.208	992.042
19	902.842	980.586	937.868	988.322
20	989.723	952.593	973.274	926.295
Mean fitness	935.0868	971.8371	962.9911	961.066
Deviation	35.20577	20.08425	22.76754	25.0349
Maximum fitness	989.723	998.842	994.588	994.381
Minimum fitness	881.61	935.94	921.559	926.295
Number of hit	3	9	6	7

The actual and estimated forces and their corresponding accelerations of the measurement point of these four methods are given in Figs. 7.4 and 7.5, respectively. From these figures, it can be observed that both standard GP and the proposed integration methods can correctly identify the dynamic force from the known response of the frame structure.

Figure 7.4 Comparison of the actual and estimated forces in Example 1

Figure 7.5 Comparison of the actual and estimated accelerations of point 2 in Example 1

Example 2: A multi-storey building

In this example, a 10-storey tall building (Li et al. 2004) shown in Fig. 7.6 will be considered as the dynamic system. This building is modeled as a uniform shear beam as described in Ref. (Clough and Penzien 1993). The flexural rigidities of all columns within one storey $\sum EI = 1.62 \times 10^8 \text{ N.m}^2$, and the storey mass is $3.6 \times 10^4 \text{ Kg}$. The damping model is same as that used in Example 1.

Figure 7.6 The 10-storey building model

Again, the exact force expression is unknown in advance but its time-history data are given. There are totally 1000 data points of the excitation force from 0 to 10.989s with time interval of 0.011s. The excitation force-time curve is plotted in solid line in Fig. 7.7. In this example, the velocity of the 9th storey is used as the measured value. The fitness function for identifying the dynamic force acting on the 10th storey by using the value measured from the 9th storey is defined as follows,

$$W' = 1000.0 - \sum_{i=1}^{1000} \left(\dot{x}_9^m(t_i) - \dot{x}_9^e(t_i) \right)^2, \quad (7.4)$$

$$\text{if } \left(\dot{x}_9^m(t_i) - \dot{x}_9^e(t_i) \right)^2 > 1.0, \text{ then } \left(\dot{x}_9^m(t_i) - \dot{x}_9^e(t_i) \right)^2 = 1.0.$$

where, $\dot{x}_9^m(t_i)$ and $\dot{x}_9^e(t_i)$ are the measured and estimated velocity of the 9th storey, respectively.

The comparison of the capability of the standard GP and the proposed integration methods for Example 2 are shown in Table 7.3. The ‘number of hit’ is defined as the number of successful run (the maximum fitness of the run is larger than or equals to 980.0) of 20 runs. Similar conclusion as drawn from Table 7.2 can be observed. The example again indicated that all the proposed integration methods outperform the standard GP.

*Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations
Identification*

**Table 7.3 Comparison of standard GP and integrated GPs for dynamic force
identification for Example 2**

Run	GP	inGAP	GP+LBS	GP+NLP
1	911.634	973.63	945.989	997.334
2	990.993	989.966	997.362	988.225
3	924.163	994.896	961.282	955.636
4	969.407	979.841	994.328	987.841
5	988.104	997.882	955.862	967.709
6	954.245	986.38	998.681	982.725
7	934.587	988.384	981.696	975.602
8	981.587	957.785	995.904	934.557
9	941.499	982.272	968.822	944.16
10	967.244	980.23	968.791	974.3
11	975.037	999.734	992.626	994.374
12	964.419	987.796	984.445	956.076
13	943.474	997.807	943.481	995.97
14	994.753	996.115	937.003	978.337
15	996.785	996.102	992.894	992.134
16	953.319	988.738	985.077	996.053
17	929.144	968.794	995.153	991.103
18	972.685	998.934	984.471	998.696
19	982.291	939.04	987.853	983.766
20	991.642	990.486	978.352	966.007
Mean fitness	963.3506	984.7406	977.5036	978.0303
Deviation	25.66282	15.34892	19.45954	18.68016
Maximum fitness	996.785	999.734	998.681	998.696
Minimum fitness	911.634	939.04	937.003	934.557
Number of hit	7	15	12	11

Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations Identification

The actual and estimated forces and their corresponding velocities of the measurement point of these four methods are given in Figs. 7.7 and 7.8, respectively. From these figures, it can be observed that both standard GP and the proposed integration methods can correctly identify the dynamic force from the known response of the high building.

Figure 7.7 Comparison of the actual and estimated forces in Example 2

Figure 7.8 Comparison of the actual and estimated velocities of the 9th storey in Example 2

The numerical results of these two examples demonstrate that all proposed integration methods outperform the standard GP for force identification problems. These results also demonstrate that the inGAP method outperforms the GP+LBS and the GP+NLP methods for force identification problems. The successful force identification of these two numerical examples shows that the proposed inGAP/GP+LBS/GP+NLP based identification methods have the potential to be applied to the force identification problems of other type structures.

7.5 APPLICATION TO GROUND MOTION IDENTIFICATION OF DYNAMIC SYSTEMS

7.5.1 Problem Description

To show the flexibility and capability of the proposed inGAP, GP+LBS and GP+NLP methods, they were employed to solve the ground motion identification problem described earlier in Chapter 6. The problem and fitness evaluation have been described in Section 6.2, 6.3, respectively; hence, will not be repeated here.

7.5.2 Solution Procedure

The solution procedure is as described in Section 7.4.2, except for the dynamic analysis; hence, will not be repeated here.

7.5.3 Illustrative Example

The model is as shown in Fig 6.16, and the parameters are the same as those shown in Section 6.4.3. It is presented to illustrate the capability of inGAP in solving complex ground motion identification of the complex structure. The ground motion acceleration expression used in this example is not known in advance, but it is same as the one used in Section 6.4.3. The velocity of the measurement point 10 is also used as the measured value in this example. The control parameters are the same as those used in Section 7.4.3.

*Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations
Identification*

**Table 7.4 Comparison of standard GP and integrated GPs for ground motion
identification**

Run	GP	inGAP	GP+LBS	GP+NLP
1	917.039	917.046	984.422	982.646
2	888.044	928.562	995.873	988.078
3	894.703	948.305	989.244	909.695
4	895.055	979.984	938.046	916.882
5	884.171	969.847	903.912	911.643
6	939.918	997.944	971.798	993.01
7	928.018	995.987	987.835	989.546
8	969.288	994.02	957.218	920.574
9	991.879	958.656	996.971	975.72
10	966.518	995.879	989.654	957.725
11	937.424	971.927	948.562	927.974
12	887.593	994.806	977.964	923.468
13	986.986	989.785	962.382	995.283
14	975.559	990.885	952.976	915.536
15	923.741	985.833	926.032	984.606
16	942.523	984.319	993.286	974.686
17	878.824	905.912	979.874	967.662
18	985.073	947.103	983.313	927.263
19	954.146	993.468	925.284	974.331
20	933.223	986.361	983.042	983.091
Mean fitness	933.9863	971.8315	967.3844	955.971
Deviation	37.4195	28.29214	27.03666	32.23789
Maximum fitness	991.879	997.944	996.971	995.283
Minimum fitness	878.824	905.912	903.912	909.695
Number of hit	3	11	9	7

*Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations
Identification*

The comparison of the capability of the standard GP and the proposed integration methods for ground motion identification are shown in Table 7.4. The ‘number of hit’ is defined as the number of successful run (the maximum fitness of the run is larger than or equals to 980.0) of 20 runs. Similar conclusion drawn from Table 7.2 can be observed. This example indicated that all the proposed integration methods outperform the standard GP for identifying ground motion acceleration. This example also indicated that the inGAP method outperforms the GP+LBS and the GP+NLP methods for identifying found motion acceleration.

The actual and estimated ground acceleration and their corresponding velocities of the measurement point of these four methods are given in Figs. 7.9 and 7.10, respectively. From these figures, it can be observed that both standard GP and the proposed integration methods can correctly identify the ground motion from the known response of the high building.

Figure 7.9 Comparison of the actual and estimated ground accelerations

Figure 7.3 Comparison of the actual and estimated velocities of the 10th storey

The results of this example indicate that the proposed inGAP/GP+LBS/GP+NLP based identification methods are potentially promising techniques for ground motion identification of multi-storey building using known dynamic response.

7.6 CONCLUSIONS

GP addresses the problem of global optimization (minimization or maximization) in the presence of competing multiple local optima. Although GP is well suited for global optimization, its rate of convergence to the global optimum can be slow in some cases. As a result, when solving non-trivial problems, some

*Chapter 7: Integrated Local Search Operators and GP for Dynamic Excitations
Identification*

sort of local search is sometimes used in conjunction with GP to increase its rate of optimization. Local search can speed up the movement of individuals in the population over regions that contain few or no local optima. In an evolutionary algorithm, complete local optimization of an individual in the population could result in the optimized individual in the population becoming a copy of a local optimum close to the parent, making it difficult for subsequent offspring to escape the local optima's basin of attraction. This in turn would result in a loss in the algorithm's reliability to converge to the global optimum. So the proper possibility for local search is important.

In this chapter, GA/LBS/NLP is used as the LSO for enhancing the global optimization ability of GP. The numerical examples of dynamic excitations identification presented in this chapter also show that local learning in the form of GA/LBS/NLP can be efficiently included into GP and LSOs can significantly enhance the performance of GP. However, the issue of the probability of GP individuals using LSOs rises. How to address this issue is problem-dependent, and this will be one of the future works.

CHAPTER 8 CONCLUSIONS AND FUTURE WORK

8.1 SUMMARY

The identification problems in the present work are treated as inverse problems. For the identification of dynamic properties of materials, strain history at the pressure bars of the SHPB setup is measured and GA is employed as a data processing method to backcalculate the critical parameters, which can be used to reconstitute the dynamic properties of materials. For the identification of dynamic excitations, due to the numerous local optima and very large search space, the traditional local optimization strategies may not be able to find the correct global optimum. Thus, GP and the proposed inGAP/GP+LBS/GP+NLP methods are employed as search methods to backcalculate the unknown dynamic excitations history based on the measured structural response of one DOF of dynamic systems.

The main objective of this study is to develop a general approach based on GA, GP and the proposed inGAP/GP+LBS/GP+NLP methods as a backcalculation algorithm to identify the dynamic properties of materials and the dynamic excitations of dynamic systems. These proposed inGAP/GP+LBS/GP+NLP methods are general optimization and search solvers. In the present work, they are employed to search the dynamic excitations acting on dynamic systems. However, they have the potential of being applied to many other engineering applications.

The present research work emphasizes the following tasks:

- (1) the development of GA-based backcalculation algorithm to identify the critical parameters of hard rocks and aluminum foams based on the traditional and the proposed modified SHPB setups;
- (2) the systematic development and implementation of the GP-based technique for identifying the dynamic excitations of dynamic systems; and
- (3) the improvement of the proposed GP-based identification method using GA/LBS/NLP as the LSOs.

The GA-based backcalculation algorithm has been employed in Chapters 3 and 4 to identify the critical parameters of the dynamic properties of hard rocks and aluminum foams based on the traditional and modified SHPB setups. In order to overcome the inherent disadvantages of the traditional SHPB setup and its corresponding data processing procedure, two modified SHPB setups are proposed. The good results obtained show that GA-based backcalculation algorithm is promising for identifying the dynamic properties of other materials.

The GP-based dynamic force identification and ground motion acceleration identification presented in Chapters 5 and 6 included the selection of the function and terminal sets, the fitness measurement method, the initialization of population, the genetic operations, and the solution procedure which incorporates a finite element analysis process for dynamic structures.

In order to improve the search efficiency of the standard GP, GA/LBS/NLP are integrated with the standard GP as a LSO, respectively. This improvement enables the standard GP to find the optimal solution efficiently and effectively. This high search efficiency and effectiveness enhance the feasibility and practicality of applying the inGAP/GP+LBS/GP+NLP-based backcalculation algorithms for dynamic excitations identification. Again, these new inGAP/GP+LBS/GP+NLP-based backcalculation algorithms are promising for other inverse problem solving.

The proposed GA-based approach has been implemented in a parameter identification system which consists of two main modules, i.e., the GA kernel module and the finite difference analysis module. The proposed GP-based approach has also been implemented in dynamic excitations identification system which consists of two main modules, i.e., the GP kernel module and the finite element analysis module. The inGAP/GP+LBS/GP+NLP-based dynamic excitations identification system comprises three main modules, i.e., the GA/LBS/NLP kernel module, the GP kernel module and the finite element analysis module. All these systems are written in the C++ language and run on the *Microsoft Visual Studio. Net 2003*. As all these systems are using implicit parallel methodology, these systems are suitable for implementation on digital computers with parallel processing capability. With the rapid development of the digital computer technology, these kinds of EC-based backcalculation methodologies may possibly be widely applied to civil engineering and other areas.

8.2 CONCLUSIONS

This study demonstrated that GA, GP and the proposed inGAP/GP+LBS/GP+NLP methods are effective tools for inverse problem solving in civil engineering. The EC-based identification systems developed and implemented in this study are useful and powerful for the practical identification of material dynamic properties and dynamic excitations.

Based on the work in this study, the following conclusions are drawn:

- (1) The GA-based backcalculation algorithm is a feasible technique for inverse problem solving in civil engineering. With GA-based algorithm, the dynamic properties of hard rocks have been backcalculated from the measured strain history. The numerical simulation results demonstrate that the proposed GA-based backcalculation algorithm can be practically

applied to material dynamic properties identification problems. Only one strain gauge, which can be randomly cemented on the pressure bars, is enough for using GA to identify the critical parameters for the dynamic properties of hard rocks. In addition, this method can be extended to identify the dynamic properties of other materials that have similar properties as hard rocks.

- (2) Based on the two modified SHPB setups, GA-based backcalculation algorithm is able to identify the entire stress-strain curve of aluminum foams, which include the densification process. In addition, easy implementation of the modified SHPB setups makes the proposed method more practicable. In practical applications, the dimensions of the modified SHPB setups can be slightly different from these two modified SHPB configurations. It should also be highlighted that the proposed method can be extended to test other materials which have similar properties as aluminum foams, or which need larger than desired maximum strain.
- (3) The GP-based search is a possible alternative for inverse problem solving in civil engineering. As it is independent of the problem domains, simple in implementation and easy for parallel computing, the approach has great potential for application in wide variety of engineering optimization and identification problems, such as dynamic force identification, ground motion identification, structural optimization and design. The unknown dynamic excitation forces acting on dynamic systems can be identified by the proposed GP-based method. No matter which point is used as the measurement point, and no matter what kind of system response is used as the measured response, it is able to identify the unknown force with good accuracy. More importantly, only single point data is needed. These merits make the proposed GP-based method more powerful and practicable than the traditional methods.

- (4) Based on only one dynamic response from one DOF, the GP-based search method is able to correctly identify the ground motion acting on dynamic systems. The outstanding merit of the proposed method is that it can obtain the explicit ground motion expression, depicting its potential for ground motion prediction.
- (5) The proposed inGAP/GP+LBS/GP+NLP methods improve the search efficiency of the standard GP without loss of effectiveness in finding the global optimum. These integration methods employ GA/LBS/NLP as LSOs to enhance the capability of the standard GP. The encoding approach between the standard GP individual and the numerical string that can be recognized by the LSOs is the basis for integrating the LSOs into the standard GP. The proposed encoding approach is an ideal 1-to-1 mapping, which makes it easy to translate the numerical coefficients in the standard GP individual to the numerical string that can be recognized by the LSOs. The high search efficiency enhances the practicality of using the proposed integration methods for inverse problem solving in civil engineering. The application of the proposed integration methods to dynamic excitations identification problems has verified their high search efficiency. It should also be mentioned that the proposed integration methods are general optimization and search methods, which have the potential of being employed in other engineering applications.

8.3 RECOMMENDATIONS AND FUTURE WORK

In this study, GA, GP and the proposed inGAP/GP+LBS/GP+NLP methods were used as backcalculation algorithms to identify the material dynamic properties and the dynamic excitations of dynamic systems. The proposed methods can overcome many disadvantages of the existing methods. However, the proposed methods have higher computation costs; and it is also very important to select

proper critical sets (function and terminal sets), fitness function, terminal criterion and other control parameters (population size, generation, possibility of crossover, possibility of mutation, ect.). For the problems in which the traditional methods or other optimization and search methods can not obtain satisfactory results, these EC-based methods can be used as excellent alternatives.

The principal areas of future work can be divided into the following main parts:

- (1) Improve the computational model of the SHPB setup. The present work is only preliminary research on using GA as the backcalculation method to identify the critical parameters of the material dynamic properties. Hence, the one-dimensional wave propagation equation was adopted as the control equation. In practice, two dimensional or three dimensional simulation of the whole SHPB setup will be better.
- (2) Increase the efficiency of the GA-based backcalculation method. In the present work, the entire structure of the SHPB setup was analysed by numerical simulation. In the future work, the pressure bars can be assumed in the elastic status, thus some analytical solution can be employed. After the strain history at the pressure bars is measured, it can be shifted to the interface of the specimen and pressure bars by the analytical analysis. Thus, only numerical simulation of the specimen, rather than the entire SHPB setup, will be considered. This will greatly improve the efficiency of the GA-based backcalculation method to identify the critical parameters of the specimen.
- (3) Investigate the possibility of using the GP-based backcalculation method to identify the entire stress-strain curve of the specimen. In this study, the GA-based backcalculation method was adopted to identify some critical parameters of the material dynamic properties. For the use of GP method, functions which include these critical parameters can be defined,

so that the identification problem can be transformed into a symbolic regression problem. As GP has its inherent advantage to solve symbolic regression problems, the GP-based search method should be a promising method to identify the dynamic properties of materials.

- (4) Extend the application of the GP-based force identification method. The GP-based backcalculation system can be extended to identify moving forces. For this case, the variables in the force expression will include both the time and the positions of the moving forces. The difficulty is that the traditional dynamic response calculation methods may not be applicable to moving force systems. A solution to this difficulty needs to be found.
- (5) Extend the GP-based ground motion acceleration identification method to identify the broad-frequency-based ground motion. For this case, the LISP S-expression of the ground motion will be more complex than the usual one. How to select the proper function and terminal sets will be a challenge.
- (6) Extend the GP-based ground motion acceleration identification method to forecasting problems. A special merit of the GP-based method for ground motion identification is its ability to obtain the expression of the acceleration. Thus, it is very suited for the ground motion forecast problems.
- (7) Improve the inGAP algorithm.
 - First, both the functions and terminals of GP individuals can be encoded as the chromosomes of GA. This investigation will examine the suitability of different encoding methods for different applications.

- Second, the GA local search can be used after the standard GP genetic operations rather than in parallel (Fig. 7.1), as shown in Fig. 8.1. This is an alternate way to integrate GA and GP. Comparison between the results obtained from this new integration (Fig. 8.1) and the proposed integration (Fig. 7.1) will orientate our further research.

Figure 8.1 Flowchart of alternate inGAP method

- Third, a new genotype integration of GA and GP can be investigated. A typical parse tree of genotype integration of GA and GP is proposed in Fig. 8.2, where the functions and terminals are encoded into binary GA strings. However, binary string encoding is not compulsory; and other encoding methods such as real code can also be used. For this kind of integration, new crossover and mutation operators need to be redefined. These redefined crossover and mutation operators will have higher feasibility than the traditional crossover and mutation operators. The new defined crossover operators can not only change the structure of the parse tree, but also change the value of each node of the parse tree. This new genotype integration of GA and GP is very suitable for solving the optimization and search problems when nodes of the parse tree are numerical.

Figure 8.2 A typical parse tree for genotype integration of GA and GP

- Finally, to improve the ability of inGAP, the following detailed methods will be studied in the future: 1) using the dynamic probability (which varies with the GP generation) of GP individuals participating in the GA local search; 2) using GA to fully optimize the first generation to obtain good seeds; and 3) using the dynamic probability of the mutation and crossover.

- (8) Validate all proposed methods by experiments. All proposed GA, GP, inGAP/GP+LBS/GP+NLP based methods are validated by numerical simulation in this thesis, but experimental validation will be the best way. In order to make all proposed methods practicable, planning and executing proper experiments to validate the proposed methods will be a problem need to be faced first at future.

In various realistic engineering fields, especially in experiment work, there are large amount of data, in computer readable form, which requires examination, classification, and integration. Unfortunately, the interrelationships among the relevant variables are poorly understood (or as suspected that the current understanding may well be wrong). The merits of the inGAP system make it suitable for solving this kind of problems. The improved inGAP system is expected to be a powerful or at least, an alternate tool for large-scale data mining. Another potential application of the improved inGAP system is the tracking of moving objects, such as submarines and missiles. In essence, the tracking of moving objects is an inverse problem. Once the necessary signals (e.g., acoustic and electric signals) of the moving object are detected and analyzed, the inGAP system may be able to back calculate its kinetic and dynamic parameters, making it possible to identify and forecast its traveling route.

REFERENCES

- Adams, D. B., Watson, L. T., Gurdal, Z., and Anderson-Cook, C. M. (2003). "Genetic algorithm optimization and blending of composite laminates by locally reducing laminate thickness." *Advances in Engineering Software*, 35(1): 35-43.
- Ahluwalia, M., and Bull, L. (2001). "Coevolving functions in genetic programming." *Journal of Systems Architecture*, 47(7): 573-585.
- Arciszewski, T., and De Jong, K. (2001). "Evolutionary computation in civil engineering: Research frontiers." invited chapter, Topping, B. H. V., (editor), *Civil and Structural Engineering Computing 2001*, 161-185.
- Arciszewski, T., and Ziarko, W. (1990). "Inductive learning in civil engineering: Rough sets approach." *Microcomputers in Civil Engineering*, 5: 19-28.
- Arciszewski, T. and Ziarko, W. (1992). "Machine learning in knowledge acquisition." In Arciszewski, T., and Rossman, A. (editors), *Knowledge Acquisition in Civil Engineering*, ASCE, 50-68.
- Ashour, A. F., Alvarez, L. F., and Toropov, V. V. (2003). "Empirical modeling of shear strength of RC deep beams by genetic programming." *Computers & Structures*, 81: 331-338.
- Babovic, V. (2000). "Data mining and knowledge discovery in sediment transport." *Computer-Aided Civil and Infrastructure Engineering*, 15(5): 383-389.
- Bäck, T. and Hoffmeister, F. (1992). "Adaptive search by evolutionary algorithms." In Ebeling, W., Peschel, M. and Weidlich, W. (Editors), *Models of*

References

-
-
- Selforganization in Complex Systems (MOSES)*, PP: 156-163. Akademie-Verlag, Berlin, 1992.
- Bacon, C., Carlsson, J., and Lataillade, J. L. (1991). "Evaluation of force and particle velocity at the heated end of rod subjected to impact loading." *Journal of Physics IV*, C3: 395-402.
- Bacon, C., Färm, J., and lataillade, J. L. (1994). "Dynamic fracture toughness determined form load-point displacement." *Experimental Mechanics*, 34(9): 217-223.
- Baeck, T., and Schwefel, H. (1996). "Evolutionary computation: an overview." In *Proceedings of the 1996 IEEE 3rd International Conference on Evolutionary Computation*, Fogel, D. (editor), IEEE Press, 20-29.
- Baeck, T., Fogel, D., and Michalewicz, Z. (1997). *Handbook of Evolutionary Computation*, Oxford Press.
- Bai, Y., and Dodd, B. (1992). *Adiabatic Shear Localization, Occurrence, Theories and Applications*, Pergamon Press, Oxford.
- Bathe, K. J. (1982). *Finite Element Procedures in Engineering Analysis*, Englewood Cliffs, N. J.: Prentice-Hall.
- Baumeister, J., Banhart, J., and Weber, M. (1997). "Aluminum foams for transport industry." *Materials & Design*, 18: 217-220.
- Bentley, P. J., and Wakefield, J. P. (1996). "Overview of generic evolutionary design systems." In *Proceedings of 2nd On-Line World Conference on Evolutionary Computation (WEC2)*, 53-56.
- Bertholf, L. D., and Karnes, C. H. (1975). "Two dimensional analysis of split Hopkinson pressure bar system." *Journal of Mechanics and Physics of Solids*, 23: 1-19.
- Blanton, J., and Wainwright, R. (1993). "Multiple vehicles routing with time and capacity constraints using genetic algorithm." In *Proceedings of the 5th International Conference on Genetic Algorithms*, Forrest, S. (editor), Morgan Kaufmann Publishers, San Mateo, CA, 452-459.

References

-
- Brameier, M., and Banzhaf, W. (2001). "A comparison of linear genetic programming and neural networks in medical data mining." *IEEE Transactions on Evolutionary Computation*, 5(1): 17–26.
- Burczynski, T., Beluch, W., Dlugosz, A., Orantek, P., and Nowakowski, M. (2000). "Evolutionary methods in inverse problems of engineering mechanics." *Inverse Problems in Engineering Mechanics*, Tanaka, M., and Dulikravich, G. S. (editors), Elsevier Science Ltd., 553-562.
- Bussac, M. N., Collet, P., Gart, G., and Othman, R. (2002). "An optimization method for separating and rebuilding one-dimensional dispersive waves from multi-point measurements. Application to elastic or viscoelastic bars." *Journal of Mechanics and Physics of Solids*, 50: 321-349.
- Cahouet, V., Luc, M., and David, A. (2002). "Static optimal estimation of joint accelerations for inverse dynamics problem solution". *Journal of Biomechanics*, 35: 1507-1513.
- Campbell, J. D., and Duby, J. (1956). "The yield behaviour of mild steel in dynamic compression." *Proceedings of the Royal Society of London Series*, A236: 24-40.
- Cao, H. Q., Kang, L. S., and Chen, Y. P. (2000). "Evolutionary modeling of systems of ordinary differential equations with genetic programming." *Genetic Programming & Evolvable Machines*, 1(4): 309–373.
- Cao, X., Sugiyama, Y., and Mitsui, Y. (1998). "Application of artificial neural networks to load identification." *Computers & Structures*, 69: 63-78.
- Chakraborty, S., Mukhopadhyay, M., and Sha, O. P. (2002). "Determination of physical parameters of stiffened plates using genetic algorithm." *Journal of Computing in Civil Engineering, ASCE*, 16(3): 206-221.
- Chen, L. (2003). "Study of applying macroevolutionary genetic programming to concrete strength estimation." *Journal of Computing in Civil Engineering, ASCE*, 17(4): 290-294.

References

- Cheng, R., and Gen, M. (1996). "Fuzzy vehicle routing and scheduling problem using genetic algorithms." In *Genetic Algorithms and Soft Computing*, Herrera, F., and Verdegay, J. (editors), Springer-Verlag, 683-709.
- Chichili, D. R., and Ramesh, K. T. (1999). "Recovery experiments for adiabatic shear localization: A novel experimental technique." *Journal of Applied Mechanics, ASME*, 66 (1): 10-20.
- Choi, I. K., and Seo, J. M. (2002). "Estimation of historical earthquake intensities and intensity-PGA relationship for wooden house damages." *Nuclear Engineering and Design*, 212: 165-182.
- Chou, J., and Ghaboussi, J. (2001). "Genetic algorithm in structural damage detection." *Computers & Structures*, 79: 1335-1353.
- Clough, R.W., and Penzien, J. (1993). *Dynamics of structures*, New York: McGraw-Hill Inc.
- Coello, C. A. C. (2000). "Constraint-handling using an evolutionary multiobjective optimization technique." *Civil Engineering and Environmental Systems*, 17(4): 319–346.
- Darwin, C. (1859). *On the Origin of Species by Means of Natural Selection or the Preservation of Favored Races in the Struggle for Life*, Murray, London, UK.
- Dasgupta, D., and Michalewicz, Z. (1997). "Evolutionary algorithms: an overview." in *Evolutionary Algorithms in Engineering Applications*, Dasgupta, D., and Michalewicz, Z. (editors), New York: Springer, 3-28.
- Davies, R. M. (1948). "A critical study of Hopkinson pressure bar." *Philosophical Transactions of the Royal Society of London Series A*, 240: 375-457.
- Davis, L. (1991). *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York.
- De Jong, K. (2001). *Evolutionary Computation: A Unified Approach*, MIT Press.

References

- Deshpande, V. S., and Fleck, N. A. (2000). "High strain rate compressive behaviour of aluminum alloy foams." *International Journal of Impact Engineering*, 24: 277-298.
- Douglas, J. (2003). "Earthquake ground motion estimation using strong-motion records: a review of equations for the estimation of peak ground acceleration and response spectral ordinates." *Earth-Science Reviews*, 61 (1-2): 43-104.
- Duffy, J., Campbell, J. D., and Hawley, R. H. (1971). "On the use of a torsional split Hopkinson bar to study rate effects in 1100-0 aluminum." *Journal of Applied Mechanics, ASME*, 38: 83-91.
- Fairbairn, E. M. R., Silvano, M. M., Filho, R. D. T., Alves, J. L. D., and Ebecken, N. F. F. (2004). "Optimization of mass concrete construction using genetic algorithm." *Computers & Structures*, 82: 281-299.
- Feng, X. T., and Yang, C. X. (2004). "Coupling recognition of the structure and parameters of non-linear constitutive material models using hybrid evolutionary algorithms." *International Journal for Numerical Methods in Engineering*, 59(9): 1227-1250.
- Fogel, D. B. (1992). *Evolving Artificial Intelligence*, Ph.D Thesis, University of California, San Diego, CA.
- Fogel, D. B. (1995). *Evolutionary Computation: Towards a New Philosophy of Machine Intelligence*, Piscataway, New Jersey: IEEE Press.
- Fogel, D. B. (1997). "The advantages of evolutionary computation." in *Bio-Computing and Emergent Computation*, Lundh, D., Olsson, B., and Narayanan, A., (editors), Singapore, World Scientific Press, 1-11.
- Fogel, L. J. (1962). "Autonomous automata." *Industrial Research*, 4: 14-19.
- Fogel, L. J. (1997). "A retrospective view and outlook on evolutionary algorithms." in *Computational Intelligence: Theory and Applications, 5th Fuzzy Days*, Reusch, B. (editor), Berlin, Springer-Verlag, 337-342.
- Fogel, L. J., Owens, A. J., and Walsh, M. J. (1966). *Artificial Intelligence through Simulated Evolution*, John Wiley & Sons Inc., New York.

References

- Follansbee, P. S., and Frantz, C. (1983). "Wave propagation in the split Hopkinson pressure bar." *Journal of Engineering Materials Technology*, 105: 61-66.
- Fraser, A. P. (1994). *Genetic Programming in C++*, University of Salford, Cybernetics Research Institute, Technical Report 040.
- Friedberg, R. (1958). "A learning machine, Part I." *IBM Journal of Research and Development*, 2(1): 2-13.
- Friedberg, R., Dunham, B., and North, J. (1959). "A learning machine, Part II." *IBM Journal of Research and Development*, 3(3): 282-287.
- Friswell, M. I., Penny, J. E. T., and Garvey, S. D. (1998). "A combined genetic and eigensensitivity algorithm for the location of damage in structures." *Computers & Structures*, 69: 547-556.
- Gary, G. and Zhao, H. (2000). "Dynamic testing of fibre polymer matrix composite plates under in-plane compression." *Composites: Part A: Applied Science and Manufacturing*, 31: 835-840.
- Gary, G., Rota, L., and Zhao, H. (1995). "Testing viscous soft materials at medium and high strain rates." in Kawata, K., and Shioiri, J. (editors) *IUTAM Symposium on Constitutive Relation in High/Very High Strain Rates*, 25-32.
- Gen, M., and Cheng, R. (1997). *Genetic Algorithms and Engineering Design*, John Wiley & Sons, Inc., New York.
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, Massachusetts.
- Gomez, J. T., Shukla, A., and Sharma, A. (2001). "Static and dynamic behavior of concrete and granite in tension with damage." *Theoretical and Applied Fracture Mechanics*, 36: 37-49.
- Gray, G. J., Li, Y., Murray-Smith, D. J., and Sharman, K. C. (1996b). "Structural system identification using genetic programming and a block diagram oriented simulation tool." *Electronics Letters*, 32 (15):1422-1424.
- Gray, G. J., Murray-Smith, D. J., Li, Y., and Sharman, K. C. (1996a). "Nonlinear model structure identification using genetic programming." Late Breaking

References

-
-
- Papers at *Genetic Programming 1996 Conference*, Stanford University July 28-31, 1996: 32-37.
- Gray, G. J., Murray-Smith, D. J., Li, Y., Sharman, K. C., and Weinbrenner, T. (1998). "Nonlinear model structure identification using genetic programming." *Control Engineering Practice*, 6: 1341–1352.
- Grefenstette, J. (1991). "Lamarckian learning in multi-agent environment." In *Proceedings of 4th International Conference On Genetic Algorithms*, Belew, R., and Booker, L. (editors), Morgan Kaufmann Publishers, San Mateo, CA, 301-310.
- Grefenstette, J. (1994). *Genetic Algorithms for Machine Learning*, Kluwer Academic Publishers, Norwell, MA.
- Gritz, L., and Hahn, J. K. (1997). "Genetic programming evolution of controllers for 3-D character animation." In *Genetic Programming 1997: Proceedings of the 2nd Annual Conference*, Koza, J. R., Deb, K., Dorigo, M., Fogel, D. B., Garzon, M., Iba, H., and Riolo, R. L. (editors), Morgan Kaufmann Publishers, San Francisco, CA, 139-146.
- Groetsch, C. W. (1993). *Inverse Problems in the Mathematical Sciences*, Braunschweig: Vieweg, Germany.
- Hao, H., and Xia, Y. (2002). "Vibration-based damage detection of structures by genetic algorithm." *Journal of Computing in Civil Engineering, ASCE*, 16(3): 222-229.
- Harding, J., Wood, E. D., and Campbell, J. D. (1960). "Tensile testing of materials at impact rates of strain." *Journal of Mechanical Engineering and Science*, 2: 88-96.
- Haupt, L. R., and Haupt, S. E. (1998). *Practical Genetic Algorithms*, John Wiley & Sons, Inc., New York.
- Himmelblau, D. H. (1972). *Applied Nonlinear Programming*, McGraw-Hill, New York.

References

- Holland, J. (1975). *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor.
- Holland, J., and Reitman, J. (1978). "Cognitive systems based on adaptive algorithms." In *Pattern-Directed Inference Systems*, Waterman, D. A., and Hayes-Roth, F. (editors), Academic Press, 313-329.
- Hoffmeister, F. and Bäck, T. (1990). "Genetic algorithms and evolution strategies: similarities and differences." In *Parallel Problem Solving from Nature, 1st Workshop PPSN I*, PP: 455-469. Springer-Verlag.
- Hooke, R. and Jeeves, T. (1961). "Direct search solution of numerical and statistical problem." *Journal of ACM*, 8(2):212-219.
- Hopkinson, B. (1914). "A method of measuring the pressure in the deformation of high explosives by the impact of bullets." *Philosophical Transactions of the Royal Society of London Series A*, 213: 437-452.
- Howard, L. M., and D'Angelo, D. J. (1995). "The GA-P: a genetic algorithm and genetic programming hybrid." *IEEE Expert Evolutionary Programming*, 10(3): 11-15.
- Huang, C. H. (2001). "An inverse non-linear force vibration problem of estimating the external forces in a damped system with time-independent system." *Journal of Sound and Vibration*, 242(5): 749-765.
- Janikow, C., and Michalewicz, Z. (1991). "An experimental comparison of binary and floating point representations in genetic algorithms." In *Proceeding of 4th International Conference on Genetic Algorithms*, Morgan Kaufmann, Los Altos, Calif., 31-36.
- John, O. H. (1998). *LS-DYNA Theoretical Manual*.
- Kammer, D. C. (1998). "Input force reconstruction using a time domain technique." *Journal of Vibration and Acoustics, ASME*, 120: 868-874.
- Kazarlis, S., Papadakis, S. E., Theocharis, J. B., and Petridis, V. (2001). "Microgenetic algorithms as generalized hill-climbing operators for GA

References

- optimization.” *IEEE Transactions on Evolutionary Computation*, 5(3): 204-217.
- Kenndy, S. (1993). “Five ways to a smarter genetic algorithm.” *AI Expert*, 35-38.
- Khu, S. T., Liong, S. Y., Babovic, V., Madsen, H., and Muttill, N. (2001). “Genetic programming and its application in real-time runoff forecasting.” *Journal of the American Water Resource Association*, 37(2), 439–451.
- Kirkpatrick, S., Gelatt Jr., C.D., and Vecchi, M.P. (1982). *Optimization by Simulated Annealing*, IBM Research Report RC 9355.
- Kojima, F., Kubota, N., and Hashimoto, S. (2001). “Identification of crack profiles using genetic programming and fuzzy inference.” *Journal of Materials Processing Technology*, 108: 263-267.
- Kolsky, H. (1949a). “An investigation of the mechanical properties of materials at very high rates of loading.” *Proceedings of the Royal Society of London Series B*, 62: 676-700.
- Kolsky, H. (1949b). “An investigation of the mechanical studies in plastic wave propagation.” *Journal of Mechanics and Physics of Solids*, 10: 195-223.
- Kolsky, H. (1963). *Stress Waves in Solids*, Oxford: Clarendon Press.
- Koza, J. R. (1992). *Genetic Programming: On the Programming of Computers by Natural Selection*, MIT Press, Cambridge, MA, USA.
- Koza, J. R. (1994). *Genetic Programming II: Automatic Discovery of Reusable Programs*, MIT Press, Cambridge, MA, USA.
- Koza, J. R., Bennett, F. H., Keane, M. A., and Andre, D. (1997). “Evolution of a time-optimal fly-to controller circuit using genetic programming”. In *Genetic Programming 1997: Proceedings of the 2nd Annual Conference*, Koza, J. R., Deb, K., Dorigo, M., Fogel, D. B., Garzon, M., Iba, H., and Riolo, R. L. (editors), Morgan Kaufmann Publishers, San Francisco, CA, 139-146.
- Koza, J. R., David Andre, Bennett, F. H., and Keane, M. (1998). *Genetic Programming III*, MIT Press, Cambridge, MA, USA.

References

-
-
- Krishnamoorthy, C. S., Venkatesh, P. P., and Sudarshan, R. (2002). "Object-oriented framework for genetic algorithms with application to space truss optimization." *Journal of Computing in Civil Engineering, ASCE*, 16(1): 66-75.
- Laarhoven, P.J.M.V. and Aarts, E.H.L. (1987). *Simulated Annealing: Theory and Application*, Kluwer Academic Publishers.
- Lataillade, J. L., Bacon, C., Collombet, F., and Delaet, M. (1994). "The benefit of Hopkinosn bar of composite and ceramic materials." *Wave propagation and Emerging Technologies*, Kinra, W. K., Clifton, R. J., and Johnson, G. C. (editors.), ASME, 188: 85-93.
- Law, S. S., and Fang, Y. L. (2001). "Moving force identification: Optimal state estimation approach." *Journal of Sound and Vibration*, 239(2): 233-254.
- Law, S. S., Chan, T. H. T., and Zeng, Q. H. (1997) "Moving force identification: a time domain method." *Journal of Sound and Vibration*, 201(1): 1-22.
- Lawrence, D. (1987). *Genetic Algorithms and Simulated Annealing*, Morgan Kaufmann Publishers, London.
- Lee, O. S. (2003). "Dynamic deformation behavior of rubber under high strain rate compressive loading." *International Journal of Modern Physics B*, 17: 1415-1420.
- Lee, O. S., and Kim, M. S. (2003). "Dynamic material property characterization by using split Hopkinson pressure bar (SHPB) technique." *Nuclear Engineering Design*, 226: 119-125.
- Lee, S. C., and Han, S. W. (2002). "Neural-network-based models for generating artificial earthquakes and response spectra." *Computers & Structures*, 80: 1627-1638.
- Lemaitre, J. (1992). *A Course on Damage Mechanics*, Springer-Verlag.
- Leung, Y. W., and Wang, Y. P. (2001). "An orthogonal genetic algorithm with quantication for global numerical optimization." *IEEE Transactions on Evolutionary Computation*, 5(1): 41-53.

References

- Lewis, J. L., and Goldsmith, W. (1973). "A biaxial split Hopkinson bar for simultaneous torsion and compression." *Review of Scientific Instruments*, 44: 811-813.
- Li, J., and Kwan, R. S. K. (2003). "A fuzzy genetic algorithm for driver scheduling." *European Journal of Operational Research*, 147(2): 334-344.
- Li, Q. M., and Meng, H. (2003). "About the dynamic strength enhancement of concrete-like materials in a split Hopkinson Pressure bar test." *International Journal of Solids and Structures*, 40: 343-360.
- Li, Q. S., Zhang, Y. H., Wu, J. R., and Lin, J. H. (2004). "Seismic random vibration analysis of tall buildings." *Engineering Structures*, 26: 1767-1778.
- Li, X. B., Lok, T. S., Zhao, J., and Zhao, P. J. (2000). "Oscillation elimination in the Hopkinson bar apparatus and resultant complete dynamic stress-strain curves for rocks." *International Journal of Rock Mechanics & Mining Sciences*, 37: 1055-1060.
- Liang, Y. C., Feng, D. P., and Cooper, J. E. (2001). "Identification of restoring forces in non-linear vibration systems using fuzzy adaptive neural networks." *Journal of Sound and Vibration*, 242(1): 47-58.
- Lifshitz, J. M., and Leber, H. (1994). "Data processing in the split Hopkinson pressure bar tests." *International Journal of Impact Engineering*, 15(6): 723-733.
- Lingras, P., and Davies, C. (2001). "Application of rough genetic algorithms." *Computational Intelligence*, 17(3): 435-445.
- Lis, J., and Eiben, A. E. (1997). "A multi-sexual genetic algorithm for multiobjective optimization." In *Proceedings of 4th IEEE Conference on Evolutionary Computation*, 59-64.
- Liu, G. R., and Han, X. (2003). *Computational Inverse Techniques in Non-destructive Evaluation*, CRC Press.

References

- Liu, J. J., Ma, C. K., Kung, I. C., and Lin, D. C. (2000). "Input force estimation of a cantilever plate by using a system identification technique." *Computer Methods in Applied Mechanics and Engineering*, 190: 1309-1322.
- López, A. M., López, H., Ojea, G., and González, V. M. (1999). "System inductive modeling using genetic programming with a genetic algorithm for parameter adjustment." In *Proceedings ETFA'99 7th IEEE International Conference on Emerging Technologies and Factory Automation*, 2: 949-952.
- Lundberg, B., and Henchoz, A. (1977). "Analysis of elastic waves from two-point strain measurement." *Experimental Mechanics*, 17: 213-218.
- Lundberg, B., Carlsson, J., and Sundin, K. G. (1990). "Analysis of elastic waves in non-uniform rods from two-point strain measurement." *Journal of Sound and Vibration*, 137: 483-493.
- Ma, C. K., Chang, J. M., and Lin, D. C. (2003). "Input forces estimation of beam structures by an inverse method." *Journal of Sound and Vibration*, 259(2): 387-407.
- Ma, C. K., Tuan, P. C., Lin, D. C., and Liu, C. S. (1998). "A study of an inverse method for the estimation of impulsive loads." *International Journal of Systems Science*, 29(6): 663-672.
- Ma, G. W., Wang, C., and Soh, C. K. (2004). "Identification of compression behaviour of aluminium foam using genetic algorithm." *WCCM VI in conjunction with APCOM'04*, Sept. 5-10, 2004, Beijing, China.
- Magyar, G., Johnsson, M., and Nevalainen, O. (2000). "An adaptive hybrid genetic algorithm for the three-matching problem." *IEEE Transactions on Evolutionary Computation*, 4(2): 135-146.
- Maulik, U., and Bandyopadhyay, S. (2003). "Fuzzy partitioning using a real-coded variable-length genetic algorithm for pixel classification." *IEEE Transactions on Geoscience and Remote Sensing*, 41(5):1075-1081.
- Maynard-Smith, J. (1994). *Evolutionary Genetics*, Oxford University Press, Oxford.

References

-
-
- Mazars, J. (1986). "A description of micro and macroscale damage of concrete structures." *Engineering Fracture Mechanics*, 25: 729-737.
- McKay, B., Willis, M., and Barton, G. (1997). "Steady-state modelling of chemical process systems using genetic programming." *Computer & Chemical Engineering*, 21(9): 981-996.
- Melhem, H. G., and Nagaraja, S. (1996). "Machine learning and its application to civil engineering." *Civil Engineering Systems*, 13(4): 259-279.
- Meng, H. (2002). *Numerical Split Hopkinson Pressure Bar (NSHPB) Test and Its Applications in the Assessment and Improvement of SHPB Test Results*, Ph.D Thesis, Nanyang Technological University, Singapore.
- Meng, H., and Li, Q. M. (2003). "An SHPB set-up with reduced time-shift and pressure bar length." *International Journal of Impact Engineering*, 28: 677-696.
- Meyers, M. A. (1994). *Dynamic Behavior of Materials*, John Wiley & Sons Inc., New York.
- Michalewicz, Z. (1996), *Genetic Algorithms + Data Structures = Evolution Programs*, 3rd. Springer-Verlag, New York.
- Mitchell, M., Holland, J., and Forrest, S. (1994). "When will a genetic algorithm outperform a hill-climbing?" In *Advances in Neural Information Processing Systems 6*, Cowen, J. D., Tesauro, G., and Alspector, J. (editors.), San Mateo, CA: Morgan Kaufmann.
- Niimi, A., and Tazaki, E. (2000). "Genetic programming combined with association rule algorithm for decision tree construction." In *Proceedings of International Conference on Knowledge-Based Intelligent Engineering Systems & Allied Technologies*, 746-749.
- Nordin, P., and Banzhaf, W. (1995). "Complexity compression and evolution." In *Genetic Algorithms: Proceedings of the 6th International Conference (ICGA95)*, Eshelman, L. (editor), Pittsburgh, PA, USA: Morgan Kaufmann, 310-317.

References

- Ory, H., Glaser, D., and Holzdeppe, D. (1986). "Quality of modal analysis and reconstruction of forcing functions based on measured output data." Proceedings of *Fourth IMAC*, Los Angeles, California, USA, 850-857.
- Paola, M. D., and Zingales, M. (2000). "Digital simulation of multivariate earthquake ground motions." *Earthquake Engineering and Structural Dynamics*, 29: 1011-1027.
- Porto, V. W. (1997), "Evolutionary programming", *Handbook of Evolutionary Computation*, Back, T., Fogel, D. B., and Michalewics, Z. (Editors), Oxford University Press, New York, Chapter B1.4.
- Rechenberg, I. (1964). "Cybernetic solution path of an experimental problem." English translation of lecture given at the *Annual Conference of the WGLR*.
- Rechenberg, I. (1965). *Cybernetic Solution Path of an Experimental Problem*, Royal Aircraft Establishment Library Transactions 1122.
- Rechenberg, I. (1973). *Evolution strategies: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*, Fromman-Holzboog, Stuttgart.
- Rechenberg, I. (1994). "Evolution strategy." In *Computational Intelligence-Imitating Life*, Zurada, J., Mark, R., and Robinson, C. (editors), IEEE Press, 147-159.
- Renders, J., and Bersini, H. (1994). "Hybridizing genetic algorithms with hill-climbing methods for global optimization: two possible ways." In *Proceedings of 1st IEEE Conference on Evolutionary Computation*, 312-317.
- Roffoei, F. R., Mobarake, A., and Ahmadi, G. (2001). "Generation of artificial earthquake records with a nonstationary Kanai-Tajimi model." *Engineering Structures*, 23: 827-837.
- Ross, B. J., Fueten, F., and Yashkir, D. Y. (2001). "Automatic mineral identification using genetic programming." *Machine Vision and Applications*, 13(2): 61-69.

References

- Ross, C. A., Tedesco, J. W., and Kuennen, S. T. (1995). "Effects of strain rate on concrete strength." *ACI Materials Journal*, 92(1): 37-47.
- Ross, C. A., Thompson, P. Y., and Tedesco, J. W. (1990). "Split-Hopkinson pressure bar tests on concrete and mortar in tension and compression." *ACI Materials Journal*, 86(5): 475-481.
- Salhi, A., Glaser, H., and Roure, D. D. (1998). "Parallel implementation of a genetic programming based tool for symbolic regression." *Information Processing Letters*, 66: 299-307.
- Samuel, A. L. (1959). "Some studies in machine learning using the game of checkers." *IBM Journal of Research and Development*, 3(3): 210-229.
- Sanchez, L. (2000). "Interval-valued GA-P algorithm." *IEEE Transactions on Evolutionary Computation*, 4(1): 64-72.
- Sarma, K. C., and Adeli, H. (2000). "Fuzzy genetic algorithm for optimization of steel structures." *Journal of Structural Engineering, ASCE*, 126(5): 596-604.
- Schwefel, H. P. (1981). *Numerical Optimization of Computer Models*, John Willey & Sons, Inc., New York.
- Schwefel, H. P. (1995). *Evolution and Optimum Seeking*, John Willey & Sons, Inc., New York.
- Schwefel, H. P. (2000). "Advantages (and disadvantages) of evolutionary computation over other approaches." in *Evolutionary Computation 1, Basic Algorithms and Operators*, Back, T., Fogel, D. B., and Michalewicz, M., (editors), Philadelphia, Institute of Physics Publishing, 20-22.
- Sean, L. (2000). "Two fast tree-creation algorithms for genetic programming." *IEEE Transactions on Evolutionary Computation*, 4(3): 274-283.
- Sean, L., and Lee, S. (1997). "A comparison of crossover and mutation in genetic programming." In *Genetic Programming 1997: Proceedings of the 2nd Annual Conference*, Koza, J. R., Deb, K., Dorigo, M., Fogel, D. B., Garzon, M., Iba, H., and Riolo, R. L. (editors), Morgan Kaufmann Publishers, San Francisco, CA, 240-248.

References

-
- Sen, M. K., and Stoffa, P. L. (1996). "Bayesian inference, Gibbs' sampler, and uncertainty estimation in geophysical inversion." *Geophysical Prospecting*, 44(2): 313–350.
- Shan, R. S., Jiang, Y. J., and Li, B. Q. (2000). "Obtaining dynamic complete stress-strain curve for rock using the Split Hopkinson Pressure Bar technique." *International Journal of Rock Mechanics & Mining Sciences*, 37: 983-992.
- Shao, S. W., and Yoshisada, M. (1996). "Approach to failure mode analysis of large structures." In *Proceedings of 7th Specialty Conf. on Probabilistic Mechanics and Structural Reliability*, Frangopol D. M. and Grigoriu M. D., (editors) ASCE, New York, 704–707.
- Sharman, K. C., Alcazar, A. I. E., and Li, Y. (1995). "Evolving signal processing algorithms by genetic programming." *Proceedings of the First IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications*, 473-480.
- Sheta, A. F., and Mahmoud, A. (2001). "Forecasting using genetic programming." In *Proceedings of the 33rd Southeastern Symposium on System Theory*, 343-347.
- Soh, C. K., and Yang, J. P. (1996). "Fuzzy controlled genetic algorithm search for shape optimization." *Journal of Computing in Civil Engineering, ASCE*, 10(2): 143-150.
- Soh, C. K., and Yang, Y. W. (2000). "Genetic programming based approach for structural optimization." *Journal of Computing in Civil Engineering, ASCE*, 14(1): 31-37.
- Soule, T., and Foster, J. A. (1998). "Effects of code growth and parsimony pressure on population in genetic programming." *Evolutionary Computation*, 6(4): 293-309.
- Spencer, G. F. (1994). "Automatic generation of programs for crawling and walking." In *Advances in Genetic Programming*, Kinnear, Jr. K. E. (editor), MIT Press, Cambridge, MA, 335-353.

References

- Srikanth, R., George, R., Warsi, N., Prabhu, D., Petry, F. E., and Buckles, B. P. (1995). "A variable-length genetic algorithm for clustering and classification." *Pattern Recognition Letters*, 16(8): 789-800.
- Starkey, J. M., and Merrill, G. L. (1989). "On the ill-conditioned nature of indirect force-measurement techniques." *Journal of Model Analysis*, 4(3): 103-108.
- Stelzner, A. D., Kammer D. C., and Milenkovic P. (2001). "A time domain method for estimating forces applied to an unrestrained structure." *Journal of Vibration and Acoustics, ASME*, 123: 524-532.
- Stevens, K. K. (1987). "Force identification problems: An overview." *Proceedings of SEM Spring Conference on Experimental Mechanics*, Houston, Texas, USA, 14-19 June, 838-844.
- Svante, W. (2002). *A Method for Dispersive Split Hopkinson Pressure Bar Analysis Applied to High Strain Rate Testing of Spruce Wood*, Ph.D Thesis, Uppsala University, Sweden.
- Tackett, W. A. (1993). "Genetic programming for feature discovery and image discrimination." In *Proceedings of the 5th International Conference on Genetic Algorithms, ICGA-93*, Forrest, S. (editor), University of Illinois at Urbana-Champaign: Morgan Kaufmann, 303-309.
- Takuya, I., Hitoshi, I., and Satoshi, S. (1998). "Depth-dependent crossover for genetic programming." In *Proceedings of IEEE International Conference on Evolutionary Computation*, IEEE Press, 775-780.
- Tao, J. S., Liu, G. R., and Lam, K. Y. (2001). "Excitation force identification of an engine with velocity data at mounting points." *Journal of Sound and Vibration*, 242(2): 321-331.
- Thrainsson, H., and Kiremidjian, A. S. (2002). "Simulation of digital earthquake accelerograms using the inverse discrete Fourier transform." *Earthquake Engineering and Structural Dynamics*, 31: 2023-2048.
- Tull, M. P., Sluss, J. J., and Havlicek, J. P. (2002). "Product demand forecasting using genetic programming." in Badiru, A. B., and Cheung, J. Y. (editors),

References

-
-
- Fuzzy Engineering Expert Systems with Neural Network Applications, John Wiley & Sons, New York, 274-281.
- Walsh, P., and Ryan, C. (1996). "PARAGEN: A novel technique for the auto-parallelization of sequential programs using genetic programming." In Genetic Programming 1996: Proceedings of the 1st Annual Conference, Koza, J. R., Goldberg, D. E., Fogel, D. B., and Riolo, R. L. (editors), MIT Press, Cambridge, MA, 406-409.
- Wang, C., Ma, G. W., Zhao, J., and Soh C. K. (2004). "Identification of hard rock dynamic property by using genetic algorithm." International Journal of Rock Mechanics and Mining Sciences, 41(3):453.
- Wang, C., Yang, Y. W., and Soh, C. K. (2005). "Genetic search for ground motion of dynamic systems." Journal of Sound and Vibration, (submitted).
- Wigham, P. A., and Recknagel, F. (2001). "An inductive approach to ecological time series modelling by evolutionary computation." Ecological Modelling, 146(1-3): 275-287.
- Williams, D. (1948). Dynamic Loads in Aeroplanes under Given Impulsive Loads with Particular Reference to Landing and Gust Loads on a Large Flying Boat, Aeronautical Research Council, Technical Report No. 2221.
- Wilson, S. W. (1995). "Classifier Fitness Based on Accuracy." Evolutionary Computation, 3(2): 149-175.
- Wolpert, D. H., and Macready, W. G. (1995). No Free Lunch Theorems for Search, The Santa Fe Institute SFI-TR-95-02-010.
- Yang, J. P., and Soh, C. K. (1997). "Structural optimization by genetic algorithms with tournament selection." Journal of Computing in Civil Engineering, ASCE, 11(3): 195-200.
- Yang, Y. W. (2001). Genetic Programming for Structural Optimization and Design, PHD thesis, Nanyang Technological University, Singapore.

References

- Yang, Y. W., and Soh, C. K. (2000). "Fuzzy logic integrated genetic programming for structural optimization and design." *Journal of Computing in Civil Engineering, ASCE*, 14(4): 249-254.
- Yang, Y. W., and Soh, C. K. (2002). "Automated optimum design of structure using genetic programming." *Computers & Structures*, 80: 1537-1546.
- Yang, Y. W., Wang, C., and Soh, C. K. (2005). "Force identification of dynamic systems using genetic programming." *International Journal for Numerical Methods in Engineering*, 63(9):1288-1312.
- Yang, Y. W., Xu, J. F. and Soh, C. K. (2005). "An evolutionary programming algorithm for continuous global optimization" *European Journal of Operational Research*, (under print).
- Yao, X. (1996). *Evolutionary Computation: Theory and Applications*, World Scientific Publishing, Singapore.
- Yen, J., and Lee, B. (1997). "A simplex genetic algorithm hybrid." In *Proceeding of the 1977 IEEE International Conference on Evolutionary Computation*, 175-180.
- Yoon, J. H., and Shoemaker, C. A. (2001). "Improved real-coded GA for groundwater bioremediation." *Journal of Computing in Civil Engineering, ASCE*, 15(3): 224-231.
- Yu, Y., and Jin, K. (1990). "The dynamic properties of ores at high strain rates." *Explosion Shock Waves*, 10(3): 266-271.
- Zerva, A., and Beck, J. L. (2003). "Identification of parametric ground motion random fields from spatially recorded seismic data." *Earthquake Engineering and Structural Dynamics*, 32: 771-791
- Zhang, Q. F., and Leung, Y. W. (1999). "An orthogonal genetic algorithm for multimedia multicast routing." *IEEE Transactions on Evolutionary Computation*, 3(1): 53-62.
- Zhao, H. (1998). "A study of specimen thickness effects in the impact tests on polymers by numeric simulations." *Polymer*, 39(5): 1103-1106.

References

- Zhao, H. (2003). "Material behaviour characterization using SHPB techniques, tests and simulations." *Computers & Structures*, 81: 1301-1310.
- Zhao, H., and Gray, G. (1996). "On the use of SHPB techniques to determine the dynamic behavior of materials in the range of small strains." *International Journal of Solids and Structures*, 33(23): 3365-3375.
- Zhao, H., and Gary, G. (1997). "A new method for the separation of waves. Application to the SHPB technique for an unlimited duration of measurement." *Journal of Mechanics and Physics of Solids*, 45(7): 1185-1202.
- Zhao, H., Gary, G., and Klepaczko, J. R. (1997). "On the use of a viscoelastic split Hopkinson pressure bar." *International Journal of Impact Engineering*, 19: 319-330.