CHAPTER 4

GENETIC ALGORITHM FOR SELECTING ENGINEERING CONTROLS

This chapter discusses a GA approach to determine a set of feasible engineering controls for optimal noise reduction. The selection of engineering controls for workplace noise reduction can mathematically be formulated as a *zero-one* nonlinear programming problem which is *NP* hard. Two mathematical models representing the engineering noise control problem (ENCP) are shown in Chapter 3. This chapter mainly discusses the solution procedure to the safety-based ENCP represented by model E2. Given a noise control budget and a set of worker locations, the problem objective is to find a combination of feasible engineering controls to minimize the maximum daily noise load. A GA is developed to find the optimal (or *near-optimal*) noise control solution for this problem. Suitable GA parameters and operations are determined from the computational experiments. Then, the effectiveness of GA by comparing GA solutions to those from the optimization approach is discussed. The last section demonstrates how GA is applied to solve ENCP.

4.1 Problem Description

Generally, an industrial workplace has several primary noise sources (manufacturing machines) and secondary noise sources (air compressors, industrial fans, industrial pumps, and cooling towers). The noise generated from these sources is transmitted to workers who are present in that workplace. In most countries, safety law requires that workers do not receive a daily noise exposure beyond the permissible level. For example, the permissible noise exposure level in the United States is set at 90 dB*A* for an 8-hour workday (OSHA, 1983). If there is any worker whose daily noise exposure exceeds this limit, an effective noise control program needs to be implemented. An engineering approach for noise control is recommended as the first line of defense owing to its high effectiveness.

Typical engineering controls include reducing noise levels at the noise sources (or controlling at the source) and blocking the noise transmission path (or controlling along the path). To reduce the noise at any noise source, there usually are several techniques that can be applied. An important result of this noise control is that noise levels at all worker locations are reduced, but the attenuation levels differ depending on the distances between the noise source and individual worker locations. Similarly, there are several noise control techniques for controlling the noise along its path (e.g., putting up physical barriers or curtains), and with varying noise attenuation capabilities. It is noted that only noise levels at the worker locations in which the direct paths between the noise source and the locations are blocked will be reduced. Comparing between controlling at the source and along the transmission path, the former is more effective than the latter, but it is also more expensive to implement. With a given noise control budget, safety practitioners must decide on the combination of engineering noise control(s) to yield the maximum noise attenuation. More specifically, if controlling at the source is being considered, it is necessary to determine which noise source(s) is/are to be controlled and with which noise control technique(s). In case of blocking along the path, the type of barrier/curtain and its location will depend on the worker locations where noise exposures are to be reduced.

The safety-based ENCP as in model E2 is a variant of the binary knapsack problem. Given the limited budget, a set of engineering controls are to be selected for implementation to achieve the maximum noise attenuation without exceeding the budget.

According to objective function and constraints (Eqs. and Inequalities (3.19) - (3.24)) in Chapter 3, safety-based ENCP is the minimax optimization problem with nonlinear constraints. Also, since it is the *zero-one* nonlinear programming problem, it cannot be solved to optimality when the problem size is large. To yield the optimal or near-optimal solution, a genetic algorithm to solve safety-based ENCP is developed.

The objective of the safety-based ENCP is to minimize the maximum daily noise load at any worker location, $l_{\max}$.

The safety-based ENCP requires three sets of constraint: (1) a budget constraint; (2) a noise load constraint; and (3) a binary variable constraint.

A genetic algorithm approach is employed to optimally select feasible engineering controls for the maximum noise attenuation without exceeding the noise control budget. The detailed discussion on GAs can be found in Holland (1975), Michalewicz (1996), and Gen and Cheng (1997 and 2000). The following sections explain the GA specifically developed for safety-based ENCP. Topics covered include: (1) GA procedure, (2) chromosome coding and initial population, (3) crossover, (4) mutation, (5) fitness and evaluation function definitions, (6) repairing procedures (7) selection techniques, and (8) termination rules.

4.2 GA Procedure

The GA procedure is illustrated in Fig. 4.1. Parameters required for the proposed GA include crossover probability *Pc*, mutation probability *Pm*, population size *Popsize*, and maximum generation *Max_gen*. Firstly, set an initial generation as *gen* = 0. If a repair procedure is required, a repair rate must also be specified. Next, binary string v_k (k = 1, 2,…, *Popsize*) is created. Each string (chromosome) represents a feasible solution for the safety-based ENCP. Essential GA operations including crossover, mutation, and selection are part of the evolution process. According to the survival-of-the-fittest rule, an evaluation function (to determine a fitness value) must be evaluated prior to the selection. The best chromosome is registered after the selection process. Then, update the *gen* value (*gen* = *gen* +1). Repeat the GA procedure until *gen* = *Max_gen*. In addition, if the repairing procedure is employed, it will be executed after the crossover and mutation operations.

4.3 GA Operations

4.3.1 Chromosome Coding and Initial Population

Binary encoding is employed in the proposed GA to create chromosomes because the decision variables of the safety-based ENCP are zero or one. The length of chromosome is equal to the number of engineering controls that are feasible for the problem being considered, $s+\sum_{t=1}^{q} b_t$. An initial population is randomly generated. Note that the number of chromosomes in the (initial and subsequent) populations is constant and is denoted by *Popsize*.

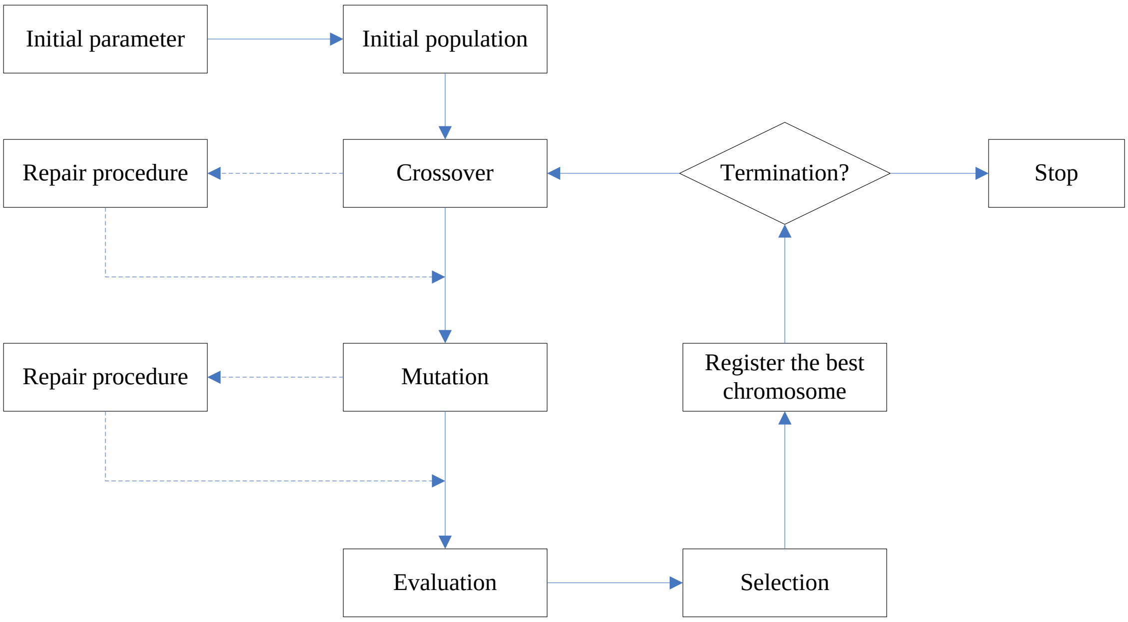


Fig. 4.1 The genetic algorithm procedure

4.3.2 <u>Crossover</u>

Crossover is a GA operation which attempts to generate two new chromosomes that may be stronger than their parents. Two parent chromosomes are randomly selected from the current population for mating. Two new chromosomes, called offspring, will be created by swapping some parts of the parent chromosomes. Crossover probability *Pc* indicates the number of chromosome pairs that will be involved in the crossover operation. For our GA procedure, two crossover techniques are considered: (1) single-point crossover, and (2) two-point crossover.

- *Single-point Crossover*

Single-point crossover is a simple technique that combines two parent chromosomes to generate two offspring. To achieve this, a random cut-point is chosen and two new offspring are generated by swapping the left-hand-side segments after the cut-point of the selected parents. Fig. 4.2 (a) illustrates the single-point crossover technique.

- *Two-point Crossover*

This crossover technique combines two parent chromosomes by choosing two random cut-points. Unlike the single-point crossover, the middle segments between both cut-points of the two parents are swapped to create two offspring as illustrated in Fig. 4.2 (b).

0 1 1|0 1 1 0 0 1 0
1 0 1|0 1 0 0 0 0 1 Parents

1 0 1|0 1 1 0 0 1 0
0 1 1|0 1 0 0 0 0 1 Offspring

(a)

0 1 1|**0 1 1**|0 0 1 0
1 0 1|**0 1 0**|0 0 0 1 Parents

0 1 1|**0 1 0**|0 0 1 0
1 0 1|**0 1 1**|0 0 0 1 Offspring

(b)

Fig. 4.2 (a) Single-point crossover and (b) Two-point crossover

Given

$cross\_no$ = number of selected chromosomes involved in the crossover
= $round(Pc \times popsize)$

$cross\_pair$ = number of pairs of chromosomes involved in the crossover

then

$$cross\_pair = \begin{cases} cross\_no/2, & \text{if } cross\_no \text{ is even} \\ (cross\_no/2+1)/2, & \text{otherwise} \end{cases} \qquad (4.1)$$

4.3.3 Mutation

Mutation is a GA operation which makes random alterations to various chromosomes. Random mutation changes a small number of bits in chromosomes depending on mutation probability Pm which indicates the number of mutated bits. A single-point mutation, which is used in this research, alters a value "1" to "0," and vice versa. Letting mut_no and $chro\_l$ denote number of mutated bits and length of chromosome, respectively, then mut_no can be computed as follows.

$$mut\_no = Pm \times chro\_l \times popsize. \qquad (4.2)$$

The mutation operation is illustrated in Fig 4.3.

0 1 **1** 0 1 1 0 **0** 1 **0**	Parent
0 1 **0** 0 1 1 0 **1** 1 **1**	Offspring

Fig. 4.3 Mutation operation

4.3.4 Fitness and Evaluation Function Definitions

An evaluation function is used to evaluate the fitness of chromosomes in each generation. The chromosomes having high evaluation values will potentially be selected for the next generation. To obtain the evaluation function, a fitness function and a penalty coefficient have to be defined. Details of these topics can be found in Michalewicz *et al* (1996) and Gen and Cheng (1997 and 2000).

- *Fitness Function*

The fitness function is problem specific. For the safety-based ECNP, a fitness value is defined as the maximum daily noise load l_{max}. When comparing between two chromosomes, since the problem objective is to minimize l_{max}, a stronger chromosome is the chromosome that has a lower l_{max} than the other one. The fitness function $f_k(v_k)$ can be written as

$$f_k(v_k) = l_{max} \qquad (4.3)$$

- *Penalty Function*

Since the safety-based ENCP has an upper bounded constraint which is the engineering control budget EB, a penalty term is added to the fitness function so that the chromosome that falls in infeasible space will have a lesser chance to be selected for the next generation. A penalty coefficient p_k, where k = 1,…, *Popsize*, is proportional to the amount of extra budget that can be determined from the following function.

$$p_k = \begin{cases} 0, & \text{if budget constraint is satisfied} \\ \theta\left(\sum_{t=1}^{q}\sum_{u=1}^{b_t}\left(cs_{tu} \times ys_{tu}\right) + \sum_{v=1}^{s}\left(cb_v \times yb_v\right)\right) - EB, & \text{otherwise} \end{cases} \tag{4.4}$$

where θ is a large positive value.

- *Evaluation Function*

From Eqs. (4.3) and (4.4), the evaluation function $eval(v_k)$, where k = 1,…, *Popsize,* can be expressed as

$$eval(v_k) = \frac{1}{f_k(v_k) + p_k} \qquad k = 1, 2, \ldots, Popsize. \tag{4.5}$$

4.3.5 Repairing Procedures

After performing the crossover and mutation operations, new chromosomes may be infeasible since the total cost exceeds the noise control budget. They have to be repaired before they can be considered for the next generation. Further discussion on the chromosome repairing issue can be found in Michalewicz *et al* (1996). The number of infeasible offspring to be repaired must not be greater than the value computed from [repair rate × *Popsize*]. Here, we consider two repair procedures, each of which can be employed to repair any infeasible chromosomes.

- *Random Repair Procedure*

This technique randomly changes bits that have a value “1” to “0.” This random change is repeated until the budget constraint is satisfied.

- *Ordered Repair Procedure*

The amount of noise generated from a noise source reaching a worker location depends on how far the location is from the noise source. Let us define a noise impact of the noise source as a sum of intensities (in W) of noise from that noise source measured at all worker locations. Thus, the noise impact of noise source t, T_t, can be computed from

$$T_t = \left[\sum_{j=1}^{n} \frac{10^{\left(\frac{L_t - 120}{10}\right)}}{d_{jt}^2}\right] \tag{4.6}$$

There are 21 steps required to complete the ordered repair procedure.

Given
bn_i = value of bit number i
i^* = selected bit number
j^* = selected worker location number
rank_no = ranking number
sum_bit = sum of bit values of bit number ranging between L and U
t^* = selected noise source number

Step 1: Rank T_t in ascending order.
Step 2: Determine *rank_no* of all noise sources from the order of T_t obtained in Step 1.

Step 3: Set $e = 1$.
Step 4: Select t^* having *rank_no* = e.
Step 5: Set $L = \sum_{t=1}^{t^*-1} b_t + 1$ and $U = \sum_{t=1}^{t^*} b_t$.
Step 6: Calculate *sum_bit* = $\sum_{i=L}^{U} bn_i$. If *sum_bit* is greater than or equal to 1, then go to Step 7. Otherwise, go to Step 10.
Step 7: Randomly select i^* where $L \le i^* \le U$.
Step 8: If bn_{i^*} = "1," then change it to "0" and go to Step 9. Otherwise, return to Step 7.
Step 9: If $\left[\sum_{t=1}^{q}\sum_{u=1}^{b_t}(cs_{tu} \times ys_{tu}) + \sum_{v=1}^{s}(cb_v \times yb_v)\right] > EB$, then go to Step 10. Otherwise, stop repairing.
Step 10: If $e < q$, then set $e = e + 1$ and go to Step 4. Otherwise, go to Step 11.
Step 11: Calculate $\bar{L}_j$ when no engineering control is implemented and rank $\bar{L}_j$ in ascending order.
Step 12: Determine *rank_no* of all worker locations from the order of $\bar{L}_j$ in Step 11.
Step 13: Set $e = 1$.
Step 14: Set $L = \sum_{t=1}^{q} b_t + 1$ and $U = \sum_{t=1}^{q} b_t + s$.
Step 15: Select j^* having *rank_no* = e.
Step 16: Calculate *sum_bit* = $\sum_{i=L}^{U} bn_i$. If *sum_bit* is greater than or equal to 1, then go to Step 17. Otherwise, stop repairing.
Step 17: Set $k = 1$.
Step 18: If $NRb_{j^*,v=k} > 0$ and bn_i (where $i = L + k - 1$) = "1," then let bn_i = "0." Otherwise, go to Step 19.
Step 19: If $\left[\sum_{t=1}^{q}\sum_{u=1}^{b_t}(cs_{tu} \times ys_{tu}) + \sum_{v=1}^{s}(cb_v \times yb_v)\right] > EB$**,** then go to Step 20. Otherwise, stop repairing.
Step 20: If $k < s$, then set $k = k + 1$ and return to Step 18. Otherwise, go to Step 21.
Step 21: If $e < n$, then set $e = e + 1$ and return to Step 15. Otherwise, stop repairing.

4.3.6 <u>Selection Techniques</u>

For the selection procedure, two basic topics are discussed: (1) sampling space, and (2) sampling mechanism. Various methods for selecting chromosomes are later examined in the computational experiment.

- *Sampling Space*

Two types of sampling space are investigated in the proposed GA. They are: (1) regular sampling space, and (2) enlarged sampling space.

  - *Regular Sampling Space*

The size of regular sampling space is always equal to *Popsize*. This is because newly generated offspring will replace their parents after their birth. Originally, this procedure is called *generational replacement*.

  - *Enlarged Sampling Space*

Both parents and offspring have been retained in the sampling space, called enlarged sampling space. Let *cross_pair* be the number of pairs which two strings are selected to do crossover; therefore, the size of sampling space is equal to *Popsize* + (*cross_pair* × 2) + *mut_no*. In this method, parents and offspring have their chances to be selected for the new generation depending upon their fitness values.

- *Sampling Mechanism*

The sampling mechanism involves how to select chromosomes from the sampling space for the new generation. Two sampling mechanism techniques are considered.

  - *Roulette Wheel Selection with Elitist Selection*

The roulette wheel selection technique is an elitist approach in which the best chromosome has a highest probability to be selected for the new generation. The basic roulette wheel is a stochastic sampling with replacement. The higher the evaluation function value a chromosome has, the greater potential it will be selected as a member of the new generation. The new generation has the same population size as the previous one. With the elitist selection, the best chromosome is firstly selected for inclusion in the new generation.

  - *Ranking Selection*

The evaluation function values of all chromosomes in the sampling space are firstly calculated. Then, they are sorted and listed in descending order (i.e., from the best to the worst). The number of chromosomes to be selected for inclusion in the new generation is *Popsize*. This approach prohibits duplicate chromosomes from passing onto the new generation.

4.3.7 Termination Rules

Since the GA is an iterative approach, the GA procedure is terminated when the number of iterations has reached the maximum generation denoted by *Max_gen*.

4.4 Analysis of GA Parameters

When applying the GA, it is known that the quality of the solution and the effectiveness of the GA are likely to be influenced by the parameter settings. A computational experiment is conducted to investigate effects of the crossover probability *Pc*, mutation probability *Pm*, population size *Popsize*, and maximum generation *Max_gen* on l_{max}.

The experiment is designed as a full-factorial experiment with four factors (i.e., *Pc*, *Pm*, *Popsize*, and *Max_gen*) and three replicates. A dependent variable in this experiment is the maximum daily noise load l_{max}. The number of levels (treatments) and the settings of each factor are shown in Table 4.1. There are 360 runs in the experiment. Two problem sizes (determined by the numbers of noise sources and worker locations) are investigated: (1) 8 noise sources (q = 8) and 8 worker locations (n = 8), and (2) 20 noise sources (q = 20) and 20 worker locations (n = 20). The results of the analysis of variance (ANOVA) of the 8 × 8 and 20 × 20 problem sizes are shown in Tables 4.2 and 4.3, respectively.

Table 4.1 Factors and levels of the full-factorial experiment

Factors	Number of Levels	Settings
Pc	5	0.1, 0.2, 0.3, 0.4, 0.5
Pm	6	0.05, 0.10, 0.15, 0.20, 0.25, 0.30
Popsize	2	50, 100
Max_gen	2	100, 10000

Table 4.2 ANOVA table for the 8 × 8 problem size

Source of Variation	Degrees of Freedom	Sum of Squares	Mean Square	F_0	*P-value*
Pc	4	0.0000012	0.0000003	3.45	0.009
Pm	5	0.0000006	0.0000001	1.45	0.205
Popsize	1	0.0000023	0.0000023	27.35	0.000
Max_gen	1	0.0000047	0.0000047	54.84	0.000
Pc × *Pm*	20	0.0000024	0.0000001	1.42	0.112
Pc × *Popsize*	4	0.0000002	0.0000001	0.63	0.640
Pc × *Max_gen*	4	0.0000012	0.0000003	3.45	0.009
Pm × *Popsize*	5	0.0000004	0.0000001	0.95	0.449
Pm × *Max_gen*	5	0.0000006	0.0000001	1.45	0.205
Popsize × *Max_gen*	1	0.0000023	0.0000023	27.35	0.000
Error	309	0.0000262	0.0000001		
Total	359	0.0000421			

Table 4.3 ANOVA table for the 20 × 20 problem size

Source of Variation	Degrees of Freedom	Sum of Squares	Mean Square	F_0	*P-value*
Pc	4	0.0007400	0.0001850	5.82	0.000
Pm	5	0.0003666	0.0000733	2.31	0.045
Popsize	1	0.0006142	0.0006142	19.31	0.000
Max_gen	1	0.0466907	0.0466907	1468.05	0.000
Pc × *Pm*	20	0.0005300	0.0000265	0.83	0.673
Pc × *Popsize*	4	0.0003666	0.0000917	2.88	0.023
Pc × *Max_gen*	4	0.0001372	0.0000343	1.08	0.367
Pm × *Popsize*	5	0.0000562	0.0000112	0.35	0.880
Pm × *Max_gen*	5	0.0000178	0.0000036	0.11	0.990
Popsize × *Max_gen*	1	0.0000899	0.0000899	2.83	0.094
Error	309	0.0098276	0.0000318		
Total	359	0.0594368			

From Tables 4.2 and 4.3, it is found that *Pc*, *Popsize* and *Max_gen* have significant effects on $l_{\max}$ in both problem sizes. *Pm* only has a significant effect on $l_{\max}$ in the 20 × 20 problem. Based on the results from the statistical analysis, we set *Pc* = 0.5, *Pm* = 0.05, and *Popsize* = 50. *Max_gen*, however, will vary with the problem size.

4.5 Analysis of GA Operations

In this section, the effects of sampling space, selection method, crossover and mutation techniques, and repair procedure are investigated in another computational experiment. As shown in Table 4.4, six treatments (P1, P2, …, P6) with different combinations of sampling space, selection method, crossover and mutation techniques, and repair procedure are described. For each treatment, nine problem sizes as indicated by the numbers of noise sources and worker locations (4 × 4, 6 × 6, 8 × 8, 10 × 10, 15 × 15, 20 × 20, 30 × 30, 40 × 40, and 50 × 50) are examined. For each problem size, five sub-problems (S-1 to S-5) are tested. All sub-problems are randomly generated using L_t, b_t, and s ranging between 85 - 105 dBA, 0 - 3 methods, and 2 - 16 methods, respectively.

Table 4.4 GA operations for the six treatments

GA Operation	Treatment					
	P1	P2	P3	P4	P5	P6
Sampling Space	Regular	Enlarged	Enlarged	Enlarged	Enlarged	Enlarged
Selection Method	Roulette Wheel	Roulette Wheel	Roulette Wheel	Ranking	Roulette Wheel	Roulette Wheel
Crossover	Single-point Crossover					
Mutation	Single-point Mutation					
Repair Procedure	None	None	None	None	Random[a]	Ordered[a]

[a]Repair rate = 0.20

Table 4.5 shows the maximum generations for the nine problem sizes in the experiment. The experiment is repeated with 10 replicates for each sub-problem. Therefore, the experiment consists of 450 experimental runs (9 problem sizes × 5 sub-problems × 10 replicates). The GA procedure is implemented in VBA (Microsoft Excel) and is run on Pentium IV, 2.80 GHz, and 512 MB RAM personal computers.

Table 4.5 Maximum number of generations for the nine problem sizes

Problem size	4 × 4	6 × 6	8 × 8	10 × 10	15 × 15	20 × 20	30 × 30	40 × 40	50 × 50
Max_gen	2,000	2,000	2,000	4,000	4,000	7,000	8,000	8,000	8,000

An average $l_{\max}$ from the 10 replicates is used as a quantitative measure to represent the GA solution and to compare among different solutions. A plot of the average $l_{\max}$ versus the number of generations of the 20×20 problem size (sub-problem: S-2) is illustrated in Fig. 4. It is seen that the average $l_{\max}$ converges quickly to the best solution within the first two hundred generations, after which it levels off. Fig. 4.5(a) and Fig. 4.5(b) show changes in the average $l_{\max}$ and average CPU time, respectively, with respect to the problem size for all six treatments and for the optimization approach. An optimization software tool called LINGO is used to solve the safety-based ENCP to optimality. Its computation time limit is set at 50,000 seconds. From Fig. 5(a), it is seen that combinations P2, P3, P5, and P6 are superior to the others due to their lower average $l_{\max}$'s. For the two largest problem sizes (40 × 40 and 50 × 50), the average $l_{\max}$ from combination P6 is found to be the lowest.

In terms of computation time, the average CPU time increases with the problem size in all six treatments (see Fig. 4.5(b)). Furthermore, the increases are found to be progressive when the problem size is 15 × 15 or larger. When LINGO is utilized, the optimal solution could be obtained only when the problem size is small (not larger than 15 × 15). Among the six treatments, combination P1 requires the least amount of CPU time to yield the best solution, while combinations P2, P3, P4, and P6 require relatively equal computation times.

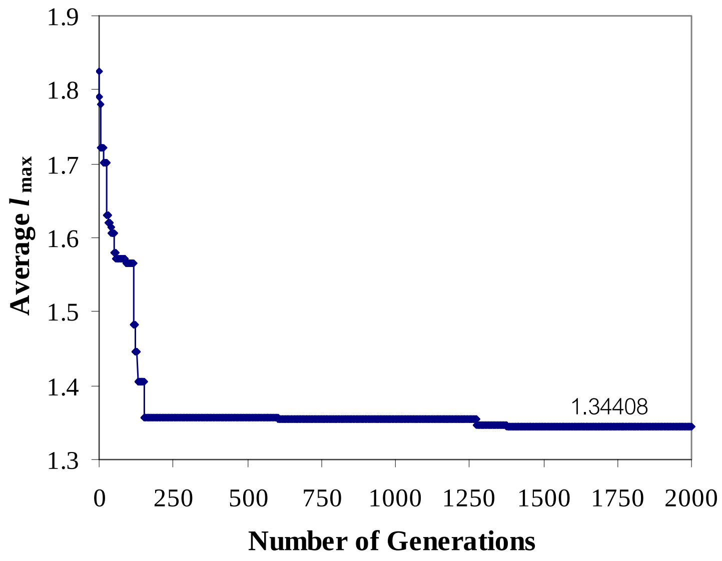
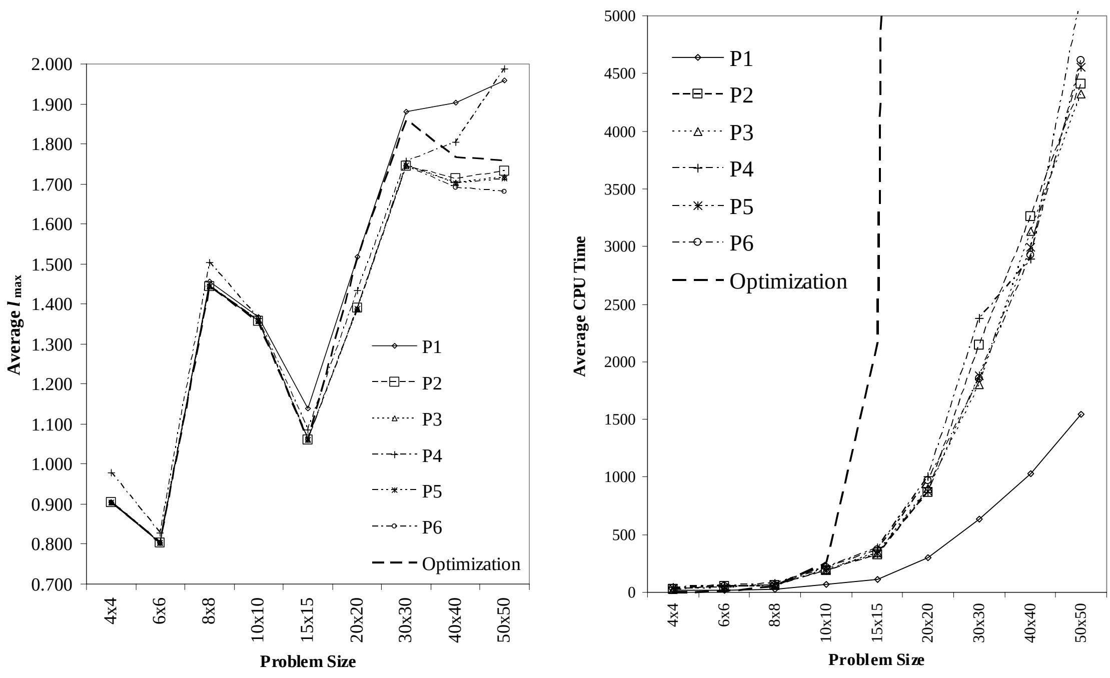


Fig. 4.4 Plot of number of generations vs. l_{max} for the 20 × 20 problem size (sub-problem: S-2)



(a) (b)

Fig. 4.5 Plots of (a) problem size vs. average l_{max}, and (b) problem size vs. average CPU time

Since our emphasis is on the quality of the solution (as measured by how low the average l_{max} is), the GA operations employed in combination P6 are chosen as those to be used in the GA approach to the safety-based ENCP. Specifically, enlarged sampling space, roulette wheel selection, single-point crossover, single-point mutation, and ordered repair procedure are employed, with P_c = 0.5, P_m = 0.5, and *Popsize* = 50.

Next, we perform a statistical analysis to study differences in the average l_{max} between solutions from the GA (i.e., combination P6) and the optimization approach (i.e., LINGO). The results (% deviation) are shown in Table 4.6. When the problem sizes are small (e.g., 4 × 4, 6 × 6, and 8 × 8), the GA is able to yield the optimal solutions in all sub-problems. For the next two larger problem sizes (10 × 10 and 15 × 15), the GA is effective in about 50% of the sub-problems solved. Nevertheless, at its worst performance, the solution from the GA is still only 0.29% greater than the optimal solution.

For the problem sizes greater than 15 × 15, LINGO is able to solve four (out of 20) problems to optimality. In some problems, LINGO can find only feasible solutions within 50,000 seconds. There are six problems (with the problem sizes 40 × 40 and 50 × 50) for which LINGO cannot obtain feasible solutions within 50,000 seconds. When the GA is used, a maximum % deviation from the optimal solutions is found to be 2.14% (at the 40 × 40 problem). In those problems for which LINGO can find feasible solutions, the solutions from the GA are superior to those from LINGO.

Thus, it is evident that the GA approach is an effective means for solving the safety-based ENCP. The GA solution is optimal when the problem size is small. For larger problems for which the optimization approach fails to find the optimal solutions, the GA can yield the solutions with small deviations from the best solutions obtained by the optimization software tool used. Additionally, the computation time when using the GA is also short, making the GA a very practical means for solving the safety-based ENCP.

Table 4.6 % deviation of average l_{max} $[(l_{max}(\text{P6}) - l_{max}(\text{LINGO}))/l_{max}(\text{LINGO})]$

Problem Size	Sub-problem				
	S-1	S-2	S-3	S-4	S-5
4 × 4	0.00	0.00	0.00	0.00	0.00
6 × 6	0.00	0.00	0.00	0.00	0.00
8 × 8	0.00	0.00	0.00	0.00	0.00
10 × 10	0.00	0.05	0.00	0.00	0.08
15 × 15	0.04	0.00	0.29	0.17	0.00
20 × 20	0.70	-4.31[a]	-15.53[a]	-3.07[a]	-13.17[a]
30 × 30	-0.52[a]	-0.87[a]	-17.87[a]	-9.20[a]	-1.50[a]
40 × 40	2.14	*	0.00	0.03	*
50 × 50	-18.20[a]	*	*	*	*

*LINGO cannot find any feasible solution within 50,000 seconds.
[a]LINGO can find only the feasible solution.

4.6 Numerical Example and Result

Let us consider an industrial facility with eight machines (q = 8) and eight worker locations (n = 8). Location coordinates of the eight machines and their noise levels (measured at 1-m distance) are shown in Table 4.7. At present, there are eight workers being assigned to eight different worker locations, and each worker must be at the same worker location for 8 hours. Location coordinates of the eight worker locations are also shown in Table 4.7. Ambient noise level in this facility is assumed to be 70 dB*A*. When no engineering noise control is implemented, 8-hour TWAs at the eight worker locations are as shown in Table 4.8. The maximum daily noise load l_{max} is found to be 2.2038 (at worker location WL5).

Table 4.7 Location coordinates and noise levels of the eight machines and location coordinates of the eight worker locations

Machine	Location Coordinate (m)		Noise Level (dB*A*)	Worker Location	Location Coordinate (m)	
	x-coordinate	*y*-coordinate			*x*-coordinate	*y*-coordinate
M1	3	2	90	WL1	3	3
M2	6	2	89	WL2	6	3
M3	9	2	89	WL3	9	3
M4	12	2	91	WL4	12	3
M5	3	6	95	WL5	3	5
M6	6	6	94	WL6	6	5
M7	9	6	93	WL7	9	5
M8	12	6	93	WL8	12	5

Noise control data for this example is as shown below.

- Noise control budget EB = 20,000 baht.
- There are two methods for blocking the noise transmission path. When applied, noise reduction occurs at worker locations WL5 and WL6. The amount of noise reduction is 7 dB*A* at each location. The barrier cost is 3,800 baht, and is the same for both methods.
- There are two methods for controlling noise at the machines. Noise reduction data (in dB*A*) at the eight machines when each method is utilized (wherever applicable) and noise control costs (in baht) are as follows:

$$[NRs_{tu}] = \begin{bmatrix} 10 & - \\ 8 & - \\ 8 & - \\ 10 & - \\ 8 & 12 \\ 8 & 12 \\ 8 & 12 \\ 8 & 12 \end{bmatrix}, \quad [cs_{tu}] = \begin{bmatrix} 5{,}000 & - \\ 3{,}500 & - \\ 3{,}500 & - \\ 5{,}000 & - \\ 4{,}500 & 5{,}500 \\ 4{,}500 & 5{,}500 \\ 4{,}500 & 5{,}500 \\ 4{,}500 & 5{,}500 \end{bmatrix}$$

The GA is applied to find feasible engineering controls that will minimize the maximum daily noise load such that the total noise control cost does not exceed 20,000 baht. The GA parameters are Pc = 0.5, Pm = 0.05, $Popsize$ = 50, and Max_gen = 2,000 generations. Enlarged sampling space, roulette wheel selection with elitist selection, single-point crossover, single-point mutation, and ordered repair procedure are selected as GA operations.

A noise control solution recommended by the GA requires the following engineering controls:

- Reducing noise at machine M5 using engineering control method 1
- Reducing noise at machine M6 using engineering control method 1
- Reducing noise at machine M7 using engineering control method 2
- Reducing noise at machine M8 using engineering control method 2

The total noise control cost is 20,000 baht. As a result, the *reduced* daily noise loads at the eight worker locations are 1.1173, 1.0570, 1.0570, 1.2311, 0.8351, 0.8011, 0.5987, and 0.5586, respectively. Note that the maximum daily noise load l_{max} = 1.2311 (at worker location WL4) is the minimum among those feasible solutions found by the GA. Since there are several daily noise loads that exceed 1, noise hazard has not yet been eliminated. For ease of comparison, updated 8-hour TWAs at the eight worker locations after implementing the recommended engineering controls are also shown in Table 4.8.

To eliminate noise hazard, the noise control budget *EB* has to be increased. Using a trial-and-error approach, it is found when *EB* is set at 28,000 baht, the 8-hour TWAs at all worker locations do not exceed 90 dB*A* (see Table 4.8). The *new* recommended engineering controls are as follows:

- Reducing noise at machine M1 using engineering control method 1
- Reducing noise at machine M4 using engineering control method 1
- Reducing noise at machine M5 using engineering control method 1
- Reducing noise at machine M6 using engineering control method 1
- Reducing noise at machine M7 using engineering control method 1
- Reducing noise at machine M8 using engineering control method 1

Table 4.8 8-hour TWAs at the eight worker locations

Worker Location	8-hour TWA (dB*A*)		
	Before Implementing Engineering Controls	After Implementing Engineering Controls (*EB* = 20,000 baht)	After Implementing Engineering Controls (*EB* = 28,000 baht)
WL1	92.3[a]	90.8[a]	84.8
WL2	92.1[a]	90.4[a]	90.0
WL3	92.0[a]	90.4[a]	89.9
WL4	92.6[a]	91.5[a]	84.8
WL5	95.7[a]	88.7	88.0
WL6	95.2[a]	88.4	88.1
WL7	94.4[a]	86.3	87.4
WL8	94.0[a]	85.8	86.5

[a]Exceeding the daily permissible level.

When comparing between the average l_{max}'s obtained from the GA and LINGO (only in small-sized problems for which LINGO can find the optimal solutions), it is seen that the GA is exceptionally effective since it is able to yield the average l_{max}'s that are identical to those obtained from LINGO. When the problem size is large (e.g., 10 × 10 and 15 × 15), the average l_{max} obtained from the GA is slightly greater than that from LINGO (the % deviation is found to be small). When the problem size is very large, LINGO will have difficulty finding the optimal solution within the given time limit of 50,000 seconds. Depending on the problem size, LINGO may or may not be able to find the best feasible solution within the time limit. The GA, on the other hand, is able to yield the feasible solution in relatively short time irrespective of the problem size. These findings confirm the effectiveness of the GA in solving the ENCP.

From the given numerical example, when the noise control budget is set at 20,000 baht, the recommended engineering controls cannot completely eliminate noise hazard since daily noise loads at some worker locations are still greater than the permissible level. By increasing the budget to 28,000 baht, a new noise control solution that is effective can now be obtained. In most real situations, the noise control budget is limited and fixed. As such, other noise control approaches should be considered. For instance, job rotation can be implemented to rotate workers among worker locations so as to reduce their noise hazard exposures. The use of hearing protection devices (HPDs) can be additionally enforced to reduce the amounts of perceived noise at selected worker locations. It should be noted that job rotation and the use of HPDs are not as effective as engineering noise controls, but they usually are less expensive. In practice, a combination of noise control approaches should be implemented to keep the total noise control cost from exceeding the budget and to achieve safety daily noise exposures in all workers.

GA developed for solving the safety-based ENCP (model E2) can also be modified to solve the cost-based ENCP (model E1) by revising the fitness and penalty function. The GA parameters and GA procedure for the safety-based ENCP can also be adapted to determine the solution for the cost-based ENCP.

For the cost-based ECNP, a fitness value is defined as the maximum total cost of engineering controls, *EC*. The fitness function $f_k(v_k)$ can be written as

$$f_k(v_k) = EC \tag{4.6}$$

Since the cost-based ENCP has an upper bounded constraint which is the *permissible daily noise load*, l_p, a penalty term is added to the fitness function. The penalty coefficient p_k, where k = 1,…, *Popsize*, is proportional to the amount of noise exceeding the permissible daily noise load that can be determined from the following function.

$$p_k = \begin{cases} 0, & \text{if noise constraint is satisfied} \\ \theta(l_{\max} - 1), & \text{otherwise} \end{cases} \tag{4.7}$$

where θ is a large positive value.