inter•noise 96
25th Anniversary Congress – Liverpool

INCE: 38.2

A GENETIC ALGORITHM FOR THE SIMULTANEOUS OPTIMIZATION OF THE SENSOR AND ACTUATOR POSITIONS FOR AN ACTIVE NOISE AND/OR VIBRATION CONTROL SYSTEM

D A Manolas, T Gialamas & D T Tsahalis

Laboratory of Fluid Mechanics & Energy, Chemical Engineering Department, University of Patras, PO Box 1400, 26500 Patras, Greece

1. INTRODUCTION

In the last decade, the advancement of digital signal processing technology has revived an old idea of noise control. This idea, named Active Noise Control (ANC), relies in i) the measurement of the initial noise (primary field) and ii) the introduction of a proper artificially-created "anti-noise" field that, when superimposed to the primary field, results in a combined noise field of lower intensity than the primary noise field. The Active Noise Control System (ANCS) consists of a) a set of sensors, b) a control unit and c) a set of actuators.

During the development of an ANCS for an application case, all these three components should be designed optimally. The optimality of (b) is limited by the existing digital signal processing technology (hardware and software). The optimal selection of (a) and (c) for an ANC application is based, not always successfully, on traditional optimization techniques. In this paper, a new approach for the optimal simultaneous selection of a set of sensors and actuators for an ANCS, based on Genetic Algorithms, is presented. The results point out the efficiency of the Genetic Algorithm-based optimization technique for this type of problems.

2. DESCRIPTION OF PROBLEM AT HAND

For the application of an ANCS in a demonstrator (in this case an aircraft), the available locations for the positioning of the sensors and actuators are identified. Due to technoeconomical constraints many times it is necessary to use a number of sensors and a number of actuators smaller than the available sensor and actuator locations inside the aircraft, respectively. Therefore, the definition of the present optimization problem is the following: given the total number of available locations for sensors ($\mathbf{M_T}$) and actuators ($\mathrm{N_T}$) for the ANCS, the optimal locations for the positioning of **M** sensors and **N** actuators should be selected simultaneously, where $\mathbf{M}<\mathbf{M_T}$, $\mathbf{N}<\mathbf{N_T}$ and $\mathbf{M}>\mathbf{N}$.

As optimal configuration (optimal locations of **M** sensors and **N** actuators) is defined as the one that produces the highest average noise reduction over the $\mathbf{M_T}$ sensor locations in the control space e.g., in the present application the inside of the demonstrator aircraft. The average noise reduction of any given configuration of sensors and actuators in the aircraft can be calculated a-priori (before the ANCS installation) using a simulation model. This model, given the **M** sensor and N actuator positions, it determines the optimum excitation amplitude and phase of each actuator and calculates the reduced sound field and the average noise reduction over the M_T sensor locations. This model is based on the Least Squares method and its detailed description is not among the purposes of this work, since it may be regarded as a black-box simulation model for the optimization Genetic Algorithm described later. The reader may refer to [2] for a description of the basics of such a model.

The reason behind the exclusion of an exhaustive search of all the sensor-actuator configurations and selection of the optimal, is their vast number. Assuming that **M** locations should be used out of M_T, the number of possible sensor configurations becomes $[M_T!/\{(M_T-M)!\cdot M!\}]$. In the same way the number of possible actuator configurations is $[N_T!/\{(N_T-N)!\cdot N!\}]$, and therefore, the number sensor-actuator configurations is $[M_T!/\{(M_T-M)!\cdot M!\}]\cdot[N_T!/\{(N_T-N)!\cdot N!\}]$. For the case of the demonstrator aircraft and the selected control unit M_T=80 and N_T=52, while M=48 and N=26. Therefore, the number of possible configurations was: $[80!/32!\cdot 48!]\cdot[52!/26!\cdot 26!] \approx 1.087\cdot 10^{37}$. Assuming that for the calculation of the average noise reduction to be achieved by each of the configurations, the simulation model needs 0.001 seconds (an assumption far better than reality), then the evaluation of all possible configurations needs $1.087\cdot 10^{34}$ seconds, or $3.45\cdot 10^{26}$ years! It is obvious that intelligent optimization techniques should be implemented in this case. In the next section, the application of the optimization method of Genetic Algorithms (GAs) to this problem is presented.

3. THE APPLICATION OF GENETIC ALGORITHMS FOR THE OPTIMAL SENSOR AND ACTUATOR POSITIONING

The popularity of Genetic Algorithms (GAs) has increased exponentially the last decade. This popularity is due to their following characteristics: a) they need only an objective function value for each point in the optimization space, b) they provide usually more than one equivalently good solution to the problem they deal with, and c) due to their simple format, the incorporation of constraints is very easy. For an overview of GAs the reader is referred to Goldberg [1].

In order for the GA to be applied to this optimization problem the parameter space was coded. The purpose of this coding was to transform the independent variables of this problem into a string (array of integers) that the GA would be able to use for the optimization. Since the problem at hand was a combinatorial one, the following coding was selected: the available sensor locations were numbered with integer codes from 1 to 80 and the actuator locations were numbered from 1 to 52. Any configuration of **M** sensors and N actuators was represented by a (M+N)-element string, consisting of two sub-strings; one containing the integers that corresponded to the **M** sensor locations selected for sensor positioning, and one containing the integers that corresponded to the N actuator locations selected for actuator positioning.

Based on his coding the initial population of strings was created. Each of the strings comprising this population was created in random, as follows: Using a random number generator, **M** different random integers from 1 to 80 were generated to take the positions of the sensor sub-string. Then, N different random integers were generated for the actuator sub-string. In this way a complete string was created. This procedure was repeated as many times as necessary in order to create the strings of the initial population.

The fitness of each string was calculated using the ANCS simulation model discussed earlier. The average noise reduction that was calculated for each string, served as its objective function value. In case that the string contained the same sensor, or actuator, location more than once it corresponded to a meaningless ANCS, since only one sensor, or actuator, may be positioned in each location. In this case the fitness value was set to be 0.1 (penalty). Since the fitness function values were roughly in the area from 1 to 7, it is obvious that the penalization fitness value was very low.

The operators that were applied to the initial population, were the following:

- **Reproduction**. The classic "roulette" reproduction operator was used. For a description of this operator the reader is referred to Goldberg [1].
- **Crossover**. The crossover operator used was similar to the classic one, with the following variations (Fig. 1). It was not performed directly to the two strings selected for the crossover (parents), but between two temporal strings. The two temporal strings were formed by removing, temporarily, the sensors and actuators that appeared simultaneously in both parent strings. After the crossover between the temporal strings was performed, two temporal offsprings were created. Then the sensors and actuators common to the parent strings, that had been removed temporarily, are placed back in their places giving rise to the two new strings (offspings). Among the two parents and the two offsprings, the two best strings were kept for the next population. This crossover operator has been applied successfully to similar optimization problems by the LFME team [3], [4].
- **Mutation**. The classic mutation operator was used. For a description of this operator please refer to Goldberg [1].

From the application of these operators to the population, a new population was created. The generation of new populations, from the old ones, continued in a similar manner, until a maximum predefined number of generations was reached.

4. RESULTS AND CONCLUSIONS

Several runs of the developed GA were performed with the following population sizes: 200, 400, 600, 800 and 1000 strings. The number of generations performed for each run was set to 1000.

It is known that the GA starts searching from many points in the optimization space, and continuously converges to the global optimum. In each generation there is a string that is the best of the population. The fitness of this best string versus the time it was achieved is given in Fig. 2, for each GA run. The maximum fitness achieved was 6.46 dBs for the case of 600 strings. Further rise of the population size leads to no better solution thus the optimal population size is about 600 strings. Moreover, it is evident that almost all the GA runs converged to a fitness value between 6.3dBs and 6.5dBs, showing in this way that the global optimum should be near that value.

From the results, the following conclusions can be drawn:

1. GAs offer a promising, efficient and easy-to-apply optimization strategy for the problem of optimally positioning simultaneously sensors and actuators in an ANCS.
2. Even though it is not evident from the graphs, the GA developed provides more than one string -sensor and actuator configurations- with similar near-optimal average noise

reduction. This is very important since it enables the engineer-designer to choose the configuration that fits better to the possible engineering constraints on the positioning of the actuators.

a. mating

	First Sub-string (M Sensor locations)										Second Sub-string (N Actuator locations)					
Parent String 1	1	4	22	24	32	56	59	63	72	79	8	13	26	31	42	51
Parent String 2	4	8	42	63	64	66	68	70	73	80	15	26	27	31	35	52

(with gray the similar sensor, or actuator, positions are denoted)

b. creation of temporal strings, crossing site selection and cutting

Temporal String 1	1	22	24	32	56	59	72	79	8	13	42	51
Temporal String 2	8	42	64	66	68	70	73	80	15	27	35	52

c. creation of the temporal offsprings

Temporal Offspring 1	1	22	24	32	56	59	73	80	15	27	35	52
Temporal Offspring 2	8	42	64	66	68	70	72	79	8	13	42	51

d. creation of the offsprings

	First Sub-string (M Sensor locations)										Second Sub-string (N Actuator locations)					
Offspring 1	1	4	22	24	32	56	59	63	73	80	15	27	26	31	35	52
Offspring 2	4	8	42	63	64	66	68	70	72	79	8	26	13	31	42	51

Fig. 1. The Crossover Operator

5. ACKNOWLEDGMENTS

The work presented in this paper was supported by the European Communities under contract AER2-CT94-0062 of the BRITE/EURAM programme.

6. REFERENCES

1. Goldberg D. E., "Genetic Algorithms in Search, Optimization, and Machine Learning", (Addison-Wesley Publishing Company Inc., 1989)
2. Nelson P. A., Elliott S.J., "Active Control of Sound" (Academic Press, London, 1993).
3. Katsikas S.K., Tsahalis D.T., Manolas D.A., Xanthakis S., "A Genetic Algorithm for Active Noise Control Actuator Positioning", Mechanical Systems and Signal Processing, November (1995)
4. Tsahalis D.T., Katsikas S.K., Manolas D.A., "A Genetic Algorithm for Optimal Positioning of Actuators in Active Noise Control: Results from the ASANCA Project", Proceedings of Inter-Noise 93, Leuven, Belgium, pp. 83-88 (1993).

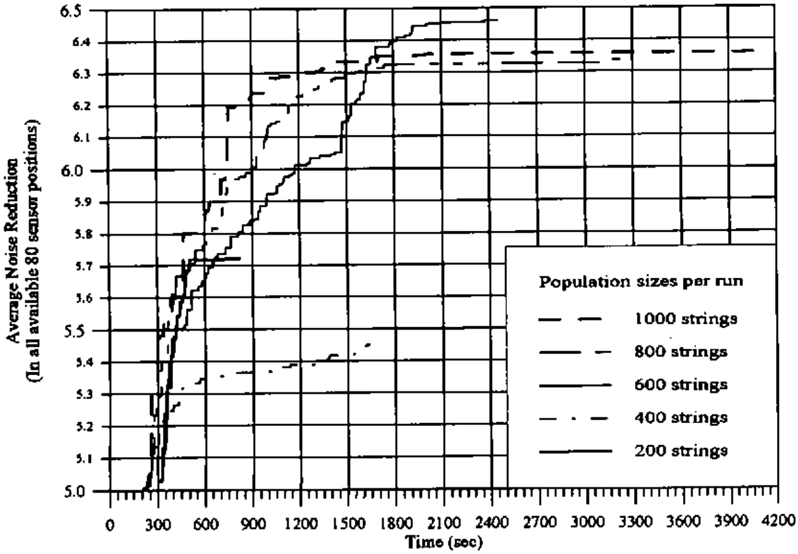

Fig. 2. GA runs for populations size of 200, 400, 600, 800 and 1000 strings