Journal of Combinatorial Optimization 1, 1–22 (1997)


# Quelling Cabin Noise in Turboprop Aircraft via Active Control

REX K. KINCAID* AND KEITH E. LABA†
*Department of Mathematics, The College of William and Mary, Williamsburg, Virginia*

SHARON L. PADULA
*NASA Langley Research Center, Hampton, Virginia*

*Received ; Revised*

**Abstract.** Cabin noise in turboprop aircraft causes passenger discomfort, airframe fatigue, and employee scheduling constraints due to OSHA standards for exposure to high levels of noise. The noise levels in the cabins of turboprop aircraft are typically 10 to 30 decibels louder than commercial jet noise levels. However, unlike jet noise the turboprop noise spectrum is dominated by a few low frequency tones. Active structural acoustic control is a method in which the control inputs (used to reduce interior noise) are applied directly to a vibrating structural acoustic system. The control concept modeled in this work is the application of in-plane force inputs to piezoceramic patches bonded to the wall of a vibrating cylinder. The goal is to determine the force inputs and locations for the piezoceramic actuators so that (1) the interior noise is effectively damped; (2) the level of vibration of the cylinder shell is not increased; and (3) the power requirements needed to drive the actuators are not excessive. Computational experiments for data taken from a computer generated model and from a laboratory test article at NASA Langley Research Center are provided.

**Keywords:**

## 1. Introduction

Most aircarriers use turboprop aircraft to transport passengers between local or nearby airports and their hubs. Turboprop aircraft are preferred for short hops due to their superior fuel efficiency at low speeds. However, the benefits of decreased fuel cost and increased travel range come at the price of increased levels of cabin noise and vibration. Turboprop noise levels are typically 10 to 30 decibels louder than commercial jet noise levels (Aerospace Eng., 1994).

Turboprop noise is fundamentally different from jet noise. The former is caused by periodic sources such as rotating engine parts, rotating props and by vortices shed from the propellers. Turboprop noise spectrum is dominated by a few low frequency tones. On the other hand, jet noise is caused by random sources such as the mixing of high speed, high temperature gases. Jet noise spectrum is dominated by high frequency broad band noise. Jet

NAG-1-1783
*The author was supported in this research by an ASEE fellowship and NASA grant #301291.
†The author completed this work as part of his undergraduate honors thesis.

noise can be controlled by putting sound absorbing material in the walls of the fuselage. This passive treatment is not effective for propeller noise (DeMeis, 1995).

Interior noise levels are of concern to commercial airlines for three reasons. First, the airlines must provide a comfortable working environment for their employees. OSHA mandates ear protection or restricted working hours for employees who are exposed to high levels of noise. Second, the airlines must attract customers. Uncomfortable noise levels may encourage the public to seek alternate travel options. Third, the airlines must consider safety issues. High noise and vibration levels cause fatigue in the airframe as well as fatigue in the passengers and crew.

Optimization has a role in reducing interior noise and vibration due to turboprops. One idea is to optimize the shape or the material properties of the fuselage so that characteristic frequencies of the interior space do not match the tones in the propeller noise spectrum (Engelstad, 1995). Another idea is to optimize the sound produced by an array of loud speakers so that the noise due to the propeller is canceled by the "anti-noise" produced by the speakers (DeMeis, 1995). A third idea is optimize the inputs to inertial actuators attached to the airframe so that the vibration of the fuselage is reduced (Aerospace Eng., 1994). These last two active control methods attempt to counteract the primary noise (or vibration) source by introducing a secondary source which is of the same frequency and amplitude as the primary, but is 180 degrees out of phase. In theory, the result is a complete silencing of the noise source.

Most of the optimization research applied to aircraft noise or vibration control employs traditional optimal control or nonlinear programming techniques. The present research is somewhat unique in its use of discrete optimization techniques. Unlike past research which seeks the optimum inputs for a fixed array of actuators, we select the best array of actuators and sensors from a large set of possible locations.

An early example of one-dimensional active noise control taken from Leug (1936) will serve to illustrate the salient features of active noise control systems. Consider a single frequency noise source, $S_1$ traveling down a duct as illustrated in figure 1. The sound is

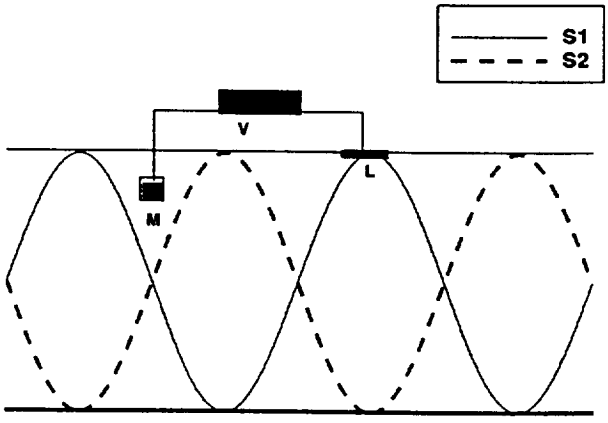



*Figure 1.* Overview of active noise control systems.

detected with a microphone, $M$, and is passed through a controller, $V$, to a loudspeaker, $L$. The controller adjusts the sound emitted by the loudspeaker so that the wave produced, $S_2$, completely cancels out $S_1$. The controller finds the appropriate sound for the loudspeaker by means of a transfer function. A transfer function establishes the relationship between measurements at $M$ and the best input (phase and amplitude) at $L$. For more details on the history of active noise control the interested reader is referred to Fuller and von Flotow (1995).

A variety of active structural acoustic controls (ASAC) systems have been proposed for aircraft (cf. Silcox et al., 1990). We consider an acoustic control model in which the control inputs, which are used to reduce interior noise, are applied directly to a vibrating structural acoustic system (e.g., an aircraft fuselage). The feasibility of this approach has been demonstrated by Cabell et al. (1993) using measured data from the aft cabin of a Douglas DC-9 fuselage. We study a simpler problem. We model the aircraft fuselage with a cylinder and the turboprop propellers with a single monopole noise source located near the exterior of the cylinder. The monopole simulates the blade passage frequency of a turboprop propeller. To measure the performance of the actuators, $p$ microphones (sensors) are mounted inside the cylinder. Then given a set of $q$ potential locations for the actuators we seek a subset of size $k \ll q$ as the locations of actuators for the active controls. The cylinder and monopole noise model is studied both as a computer generated mathematical model and as a laboratory test article. It is hoped that the lessons learned in this simplified model will provide fertile ground for understanding the noise control problem in turboprop aircraft.

The control inputs modeled and tested with a laboratory test article are in-plane forces created by piezoceramic patches bonded to the wall of a vibrating cylinder (cf. Silcox et al., 1992 for an evaluation of piezoceramic actuators). The cylinder is excited by an exterior noise source—an acoustic monopole—located near the outside of the cylinder wall. The goal is to determine the force inputs and sites for the piezoceramic actuators so that (1) the interior noise is effectively damped; (2) the level of vibration of the cylinder shell is not increased; and (3) the power requirements needed to drive the actuators are not excessive. We study external monopole frequencies representative of the blade passage frequency of a turboprop propeller. We also study cases in which the cavity mode couples with cylinder vibration modes. Cabell and Lester (1993) noted that in some cases the actuators reduced interior noise but increased cylinder shell vibration levels. We want to know if optimized actuator sites also have this undesirable feature. Previous work (Lester and Silcox, 1991; Cabell and Lester, 1993) has focused almost exclusively on meeting objective (1) and solving a complex least-squares problem to arrive at an optimal force vector for a given set of actuator sites.

How to determine the best set of actuator sites to meet objectives (1)–(3) is the main contribution of our research effort (cf. Kincaid, 1995). The selection of a best set of actuator sites from a set of potential sites is done via a simple tabu search. Our experiments confirm that tabu search is able to uncover better solutions than those selected based upon engineering judgement alone. In addition, the high quality solutions found by tabu search, when minimizing interior noise, do not further excite the cylinder shell. Thus, we are able to meet objective (2) without imposing an additional constraint or forming a multi-objective

performance measure. An additional observation is that in many cases the amplitude and phase values for several chosen actuator sites were nearly identical. This natural grouping means that fewer control channels are needed and the resulting control system is simpler. Currently no power requirements have been set, so objective (3) cannot be addressed. A set of experiments with a laboratory test article (a cylinder) have been performed (Palumbo et al., 1996). For these experiments the transfer matrices were generated experimentally. The predicted performance of the best actuator sites found for the computer generated mathematical model by tabu search are shown to correlate well with the actual measured performance (cf. Section 6).

## 2. Problem description

The dominant frequencies of the propeller generated noise are typically on the order of 100 Hz to 400 Hz. Moreover, propeller generated noise occurs at several frequencies and it is clear that a good set of actuators for noise control at one frequency is not necessarily good for noise control at other frequencies. Nevertheless, we decouple the noise control problem and consider only one frequency at a time. Eventually solutions that offer performance tradeoffs for a variety of frequencies must be addressed, but we do not do so here.

Figure 2 shows the cylinder for the computer generated mathematical model test results. In the mathematical model a steady-state response is assumed and both the cylinder shell response (vibrations) and interior acoustic response (noise) are expressed as a finite series

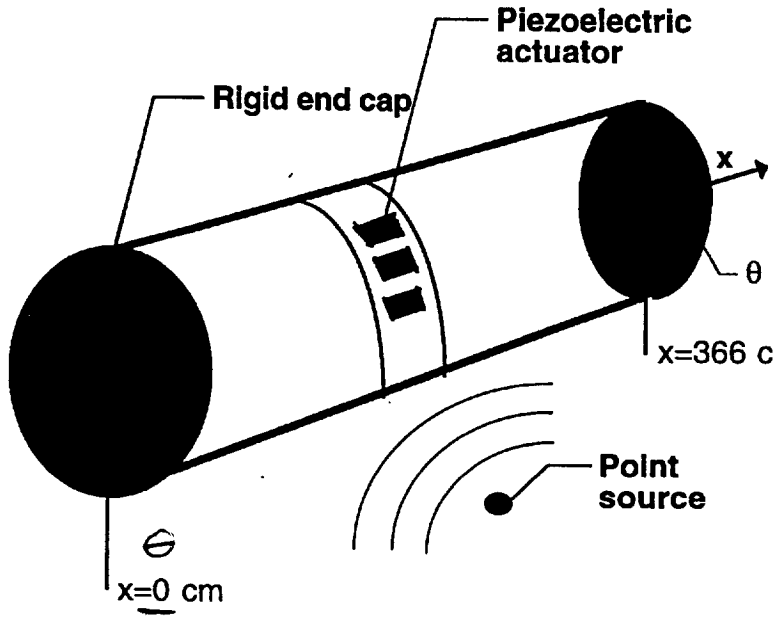



*Figure 2.* Cylinder model for ASAC location problem.

of modes $(m, n)$, with the interior response represented as

$$p(x, r, \theta) = \sum_{n=0} \sum_{m=0} J_n(\alpha_m r) \cos\left(\frac{m\pi x}{L}\right) \left[P^c_{m,n} \cos(n\theta) + P^s_{m,n} \sin(n\theta)\right]$$

where $L$ is the length of the cylinder and $(x, r, \theta)$ gives the cylindrical coordinates. The functions $J_n(\,)$ involve inverse Fourier transforms with integrands in terms of complex Bessel functions. The modal coefficients $P^c_{m,n}$ and $P^s_{m,n}$ can be written as combinations of modal transfer functions available in the literature and cylinder shell displacement coefficients. Please see Silcox et al. (1990) for details.

In practice the interior noise cannot be measured everywhere and we approximate $p(x, r, \theta)$ by an array of sensor microphones. The best location of these sensors is an optimization problem we plan to address in a subsequent manuscript. Some details about the sensor location problem are given in Section 6 and in Palumbo et al. (1996). For the remainder of this work, we will refer to the interior noise generated by the monopole noise source as a $p$ dimensional vector $\vec{p}$ where each component represents a sensor (microphone). It is this noise $\vec{p}$ that we seek to control by introducing $k$ actuators (chosen from $q$ possible actuator sites) at various locations along the interior surface of the cylinder shell.

To model objective (1) a $p$ by $k$ complex transfer matrix $H$ is determined, either experimentally or via a mathematical model. Each entry of $H$ provides the contribution of each of the potential $k$ actuators sites to interior noise reduction at each of the fixed $p$ error microphone locations, whenever a unit amplitude input is applied to actuator site $k$. Objective (2) requires more information. A pair of complex three dimensional matrices of shell displacement modal coefficients, $Ws_{jmn}$ and $Wc_{jmn}$, (one to account for the coupling with the sine functions and the other for the cosine functions) must be determined, either experimentally or with a mathematical model. Each entry of the matrices gives the displacement coefficient produced by a unit amplitude input at actuator site $j$ for mode $m, n$. These two matrices are then used to compute the vibration levels of the cylinder shell at all modes for any combination of the $j$ actuator sites for a given force input. Lastly, since power is directly related to the magnitude of the force inputs required, objective (3) can be measured by considering either the Euclidean or max norm of the actuator force input vector.

Recall that a complex transfer matrix $H$ can be generated (either experimentally or with a mathematical model) for the complete set of $q$ actuator sites. A column $j$ of $H$ yields the contribution of a unit amplitude input at actuator site $j$ in controlling the noise at each of the $p$ sensors. Hence, given a particular set of $k$ columns (actuators) the submatrix of $H$ induced by these columns, $H^k$, is the associated transfer matrix. Once a particular set of $k$ actuator sites have been selected then a force (amplitude and phase) must be selected for each actuator so that the resultant force cancels out $\vec{p}$ as nearly as possible. In the acoustic literature the force vector $\vec{f}$ is chosen as the solution of the complex least squares problem

$$\|H^k \vec{f} + \vec{p}\|_2. \tag{1}$$

We note that although the usual convention is to have a subtraction operator in (1), the acoustic literature uniformly displays the plus operator. Thus, we adhere to acoustic literature convention. The solution to (1) is found by solving

$$H^{k*} H^k \vec{f} = -H^{k*} \vec{p} \tag{2}$$

for $\vec{f}$ ($*$ denotes the complex conjugate transpose). This means that each time a different set of $k$ columns of $H$ is selected a new complex least squares problem must be solved.

We address the question of selecting an initial set of $k$ columns of $H$ by referring the reader again to the example in figure 1. If there exists a column of $H$ which is collinear with $\vec{p}$ then, as in figure 1, we would need only pick this single actuator site, driven at the appropriate amplitude, and total silencing would be achieved. Typically this situation does not exist. However, a likely set of columns for a starting solution are the $k$ columns of $H$ most nearly collinear with $\vec{p}$. To find these columns, we determine the angle between each actuator $j$ site and $\vec{p}$. This is done by calculating the dot product of each column of the $H$ matrix, $\vec{h}_j$, with the vector $p$ and dividing that product by the product of the norms of each. That is, the expression;

$$\frac{\vec{h}_j \cdot \vec{p}}{\|\vec{h}_j\| \|\vec{p}\|}$$

would calculate the cosine of the angle between the two vectors. Since the actuators which would be most effective, theoretically, in silencing the outside noise source would be those either in phase or 180 degrees out of phase, we seek cosine values close to either 1 or $-1$. Thus, the $k$ initial actuators are chosen which have the absolute value of the cosine closest to one.

The performance of this set of $k$ columns (or any set of $k$ columns for that matter) is measured on a decibel scale given by the expression below.

$$10 \cdot \log_{10} \frac{\|H^k \vec{f} + \vec{p}\|_2}{\|\vec{p}\|_2} \tag{3}$$

The decibel value computed in (3) compares the interior noise norm with actuator controls in place with the norm of the uncontrolled interior noise. Thus, a negative decibel (dB) level signifies a decrease in interior cylinder noise due to the control effects of the actuators. An analogous decibel expression computes the vibration level on the cylinder shell.

## 3. Plain vanilla tabu search

In our initial efforts to generate solutions to the ASAC location problem we used a tabu search methodology that invoked only the most basic features of tabu search. We describe the search in this section and call it plain vanilla tabu search (PVTS) to clearly distinguish it from the more general methodology that tabu search encompasses. Additional refinements to PVTS are discussed in Sections 5 and 7.

The definitions and notations that follow are taken from Glover (1989, 1990) and Kincaid and Berger (1993). Let $\Sigma$ denote the set of all feasible states. $\Sigma$ for the ASAC location problem is the set all subsets of size $k$ actuators selected from the set of size $q$ of potential actuators. Thus for ASAC $\Sigma$ is of size $q!/(k!(q-k)!)$. We call $\Sigma$ the *state space* and let $S$ denote an element of $\Sigma$. To differentiate between states we define a criterion function $c : \Sigma \to \mathbf{R}^+$ and to proceed from one state to another state we define a move set $\Delta$. For the

ASAC location problem $c(S)$ is the interior noise reduction (calculated in decibels) for state $S$. $\Delta$ may take on a variety of forms for the ASAC location problem but all involve designating the set of $k$ columns of the complex transfer matrix $H$ currently in the solution as *on* and the $q-k$ of $H$ not in the solution as *off*. One example of $\Delta$ is to consider all pair-wise interchanges of the *on* column indices with the *off* column indices. A move $\delta \in \Delta$ is a function from $\Sigma \rightarrow \Sigma$. The outcome of applying all moves $\delta \in \Delta$ to a state $S \in \Sigma$ is the set of states reachable from $S$. This is typically called the *neighborhood* of $S$. The *value* of a move is the difference between the criterion function values after and before the move, $c(\delta(S)) - c(S)$.

PVTS begins with an initial state $S_0$, chosen either at random or constructed by selecting the $k$ columns of $H$ most nearly collinear with $\vec{p}$. Next PVTS generates a sequence of moves $\delta_0, \delta_1, \ldots$ which determines a sequence of states through which the search proceeds. The mechanism by which a move is selected is crucial. PVTS, at a state $S_t$, selects the greatest available one-move improvement. That is, a next move $\delta_t$ is defined to be

$$c(\delta_t(S_t)) = \max_{\delta \in \Delta - \tau + \alpha} c(\delta(S_t)). \tag{4}$$

As will be described below $\tau$ restricts the move set and $\alpha$ extends the move set. Without the addition of $\tau$ and $\alpha$ (4) would describe the move selection criterion in a greedy local improvement search and the search would conclude with a local optima. In effect, PVTS selects the largest uphill move, if one is available, or the least downhill move. Hence, there is no natural stopping criteria and the length of the sequence of generated moves must be specified.

PVTS attempts to avoid entrapment in local optima by keeping a list of previously selected moves and deleting them from the move set $\Delta$ for a state $S$ in the hope of avoiding a return to a previously observed state. A list of move *attributes* of length *tabusize*, called a *tabu list* and designated by $\tau$ is constructed and updated during each iteration of the search. An example of a move for the ASAC problem is to swap a pair of column indices $(i, j)$ where $i \in$ *on* and $j \in$ *off*. The attributes of this move are the indices $i$ and $j$ and the designations *drop* for $i$ and *add* for $j$.

An *admissible move* in PVTS is a move $\delta \in \Delta$ that is either not on the tabu list $\tau$ or one that meets an aspiration level criterion. A *best* admissible move is one that yields the greatest improvement or the least degradation in the criterion function value. The best admissible move is appended to the tabu list after the examination of all moves $\delta \in \Delta$. Once the tabu list becomes full, which occurs after tabusize iterations, the "oldest" move is removed. Moreover, it is possible for a move in $\tau$ to be selected provided that it meets one (or more) *aspiration level criterion*. The purpose of an aspiration level criterion is to choose "good" moves by allowing the tabu status of a move to be overridden if the move satisfies a particular condition. We label the set of tabu moves that meet an aspiration criterion $\alpha$. The goal is to do this in a manner that retains the search's ability to avoid returning to a previously generated state (*cycling*). If $\delta \in \tau \subseteq \Delta(S)$ and $c(\delta(S)) > c_{\text{best}}$, then $\delta$ is admissible. The tabu status is overridden, because the move $\delta$ meets the aspiration level criterion. Cycling is avoided here since if $\delta(S)$ had appeared previously, then $c(\delta(S)) > c_{\text{best}}$ would not be possible. This particular aspiration criterion (and the only one we use) is easy to implement but treats all states with the same criterion function value as if they were the same state.

## 4. Performance of PVTS for the mathematical model

A computer generated mathematical model of a cylinder with either one or two external noise sources was developed at NASA Langley Research Center. This model allows the user to specify dimensional characteristics of the cylinder and its distance from the monopole noise sources. Given a particular frequency for the noise source(s) the model generates all data necessary to compute interior cylinder noise levels, cylinder shell vibration levels, and the forces (power) needed to drive the actuators. In particular the model generates the 2-dimensional complex transfer matrix $H$ and the 3-dimensional complex shell displacement modal coefficient matrices $Ws_{jmn}$ and $Wc_{jmn}$. The user must specify the number of sensors and actuators as well as their location on the cylinder. Each sensor and actuator is centered at coordinates $(x, \theta, r = a)$. The acoustic monopole is located at $(x = L/2, \theta = 0, r = 1.2a)$ where $L$ is the length of the cylinder and $a$ is the cylinder radius. We tested 102 potential actuator sites which are distributed about the cylinder in 6 rings of 17 actuator sites (see figure 3). The sensors are placed in 6 rings of 18 for a total of 108 and are distributed along the top half of the cylinder (from 0 to 180 degrees). The sensors and actuators are not co-located.

In Table 1 we report on the performance of actuator locations determined by PVTS with respect to interior cylinder noise, cylinder shell vibration, and power usage for a monopole

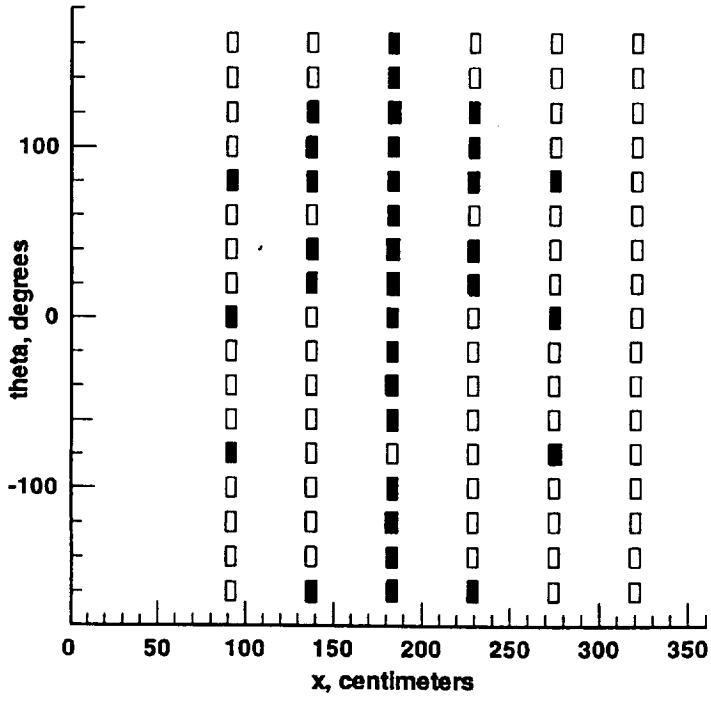



*Figure 3.* 34 actuator solution at 200 Hz.

*Table 1.* PVTS with three objectives at 200 Hz.

| Num. | Interior dB | Shell dB | Power |
|---|---|---|---|
| 4 | −14.2464 | +2.1293 | 77.69 |
| 8 | −19.5128 | −0.7041 | 75.71 |
| 14 | −23.7465 | −5.4969 | 94.15 |
| 16 | −25.1868 | −7.3179 | 104.80 |
| 20 | −26.3284 | −7.1582 | 121.16 |
| 34 | −32.1625 | −8.1127 | 179.17 |

noise source with a frequency of 200 Hz. The choice of $\Delta$ for PVTS in this section is the set of all pairwise swaps between actuator sites that are *on* and those that are *off*. We examine the entire move set and select the best admissible move $\delta$. Then, for the next tabusize iterations, the inverse of $\delta$ is considered a tabu move. The transfer matrices and shell displacement modal coefficient matrices generated by the model correspond to the 200 Hz and 275 Hz frequencies of a single monopole noise source. The 200 Hz case excites only one dominant interior cavity mode of the cylinder while the 275 Hz case excites several cavity modes of similar importance. Thus, the 200 Hz case is considered easier to control than the 275 Hz case. The construction of the initial feasible solution (state $S_0$) is done by selecting the $k$ columns of $H$ most nearly collinear with $\vec{p}$ as is described in Section 2. Table 1 contains the results for several choices of $k$, the number of actuators to be located, for the 200 Hz noise source.

PVTS seeks only to minimize the interior noise objective. It is surprising that these solutions also yield a significant decrease in the shell vibration levels for $k > 8$ (as measured by the modeled data at the 108 sensors). The expectation was that as more and more actuators were placed on the cylinder shell that the forces needed to drive these actuators to damp interior noise would increase the vibration levels on the shell. Consequently, all of the experiments reported in the literature are for fewer than 16 actuators. The success of the 34 actuator case at 200 Hz led us to try computational experiments with larger numbers of actuators.

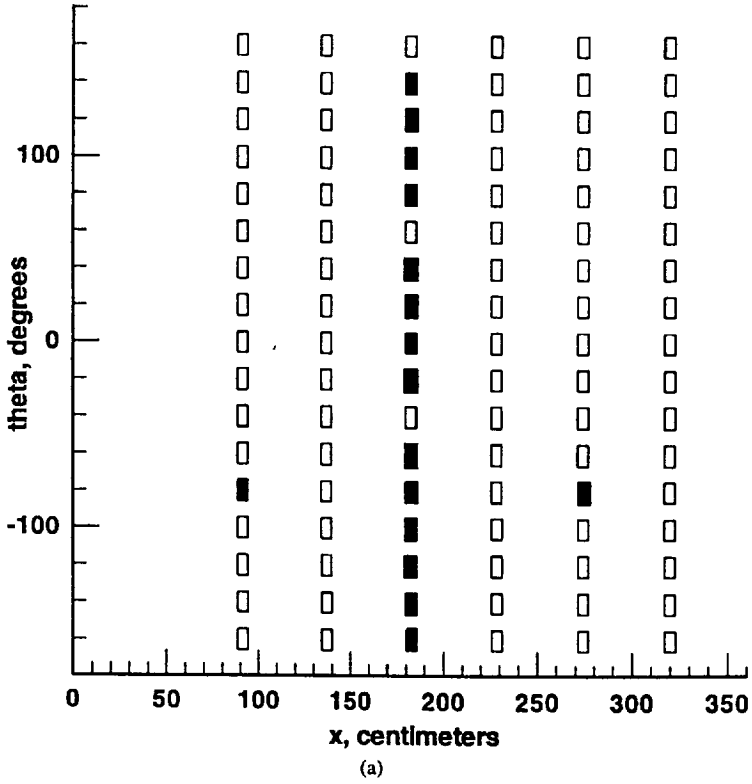
Unfortunately we were unable to run complete experiments for larger actuator sets due to excessive computing time requirements. However, we do have limited results for 68, 98 and 101 actuators. We ran two iterations of PVTS for 68 actuators resulting in a −39.9969 dB interior noise reduction, a −7.3405 dB shell vibration reduction, and an amplitude norm of 308.98. For 98 and 101 actuators we were only able to execute the construction of the initial solution. With 98 actuators the initial solution had a −47.8411 dB interior noise reduction, a +0.71 dB shell vibration increase, and an amplitude norm of 2127.62. With 101 actuators the initial solution had a −46.0802 dB interior noise reduction, a +17.17 dB shell vibration increase, and an amplitude norm of 7943.19. These computational experiments with larger actuator sets show that we should expect continued effective performance with up to at least 68 actuators. It is also clear that problems with the shell vibration levels are starting to occur near the 100 actuator mark.

Currently there are no set requirements for power levels. The numbers we report under the power column in Table 1 are the Euclidean norm of the amplitudes of the forces required

to drive the actuators. This number provides a mechanism for comparing the magnitudes of potential power requirements between solutions. It is known that power will eventually need to be considered when flight tests are done. In fact, in one preliminary flight test (Cabell et al., 1993) the power requirements of the solution generated by the mathematical model were too large to be used in the flight test.

An additional practical consideration is the number of independent channels to be driven by onboard controllers. We have assumed that there is one channel available for each actuator. However, in practice there may be a limited number of channels and solutions for which groups of actuators can be driven by the same channel are desirable. Figure 3 displays the solution to the 34 actuator case. The $x$-axis runs the length of the cylinder (in centimeters) and the $y$-axis runs the circumference of the cylinder in degrees from $-180$ to $+180$ with 0 degrees directly in line with the monopole noise source. There is an obvious symmetry to the solution about the center of the cylinder (180 cm) and the symmetric pairs are driven by the same forces (amplitude and phase). Hence, rather than 34 separate channels only 25 channels are required for this solution.

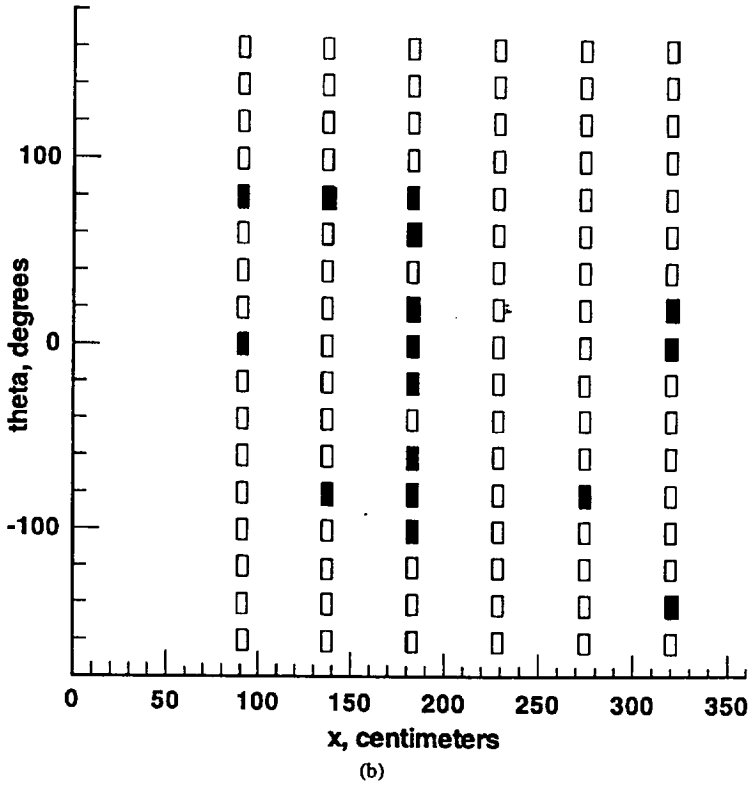
The shaded rectangles in figure 4(a) show the set of 16 actuators chosen by PVTS (whose performance is described in Table 1) for 200 Hz. As in the 34 case in figure 3 the solution




(a)

*Figure 4.* (a) 16 actuator solution at 200 Hz, (b) 16 actuator solution at 275 Hz.

(*Continued on next page.*)




(b)

*Figure 4.* (*Continued*).

is symmetric but only two pairs of actuators can be driven together by the same controller. Figure 4(b) shows the PVTS solution for controlling interior noise due to a monopole with a frequency of 275 Hz. The pattern is highly irregular due to the existence of several cavity (interior of the cylinder) modes of similar importance which must be controlled. How to determine sets of actuators that work well simultaneously for several frequencies is a subject of future research efforts.

## 5. Experiments with move sets

In this section we examine the effects of different move sets $\Delta$ and different move attributes on computation time and solution quality of PVTS when applied to the ASAC location problem. The transfer matrix for all experiments in this section is for a 200 Hz frequency of a monopole noise source and was generated by the mathematical model. The 200 Hz case excites only one dominant interior cavity mode of the cylinder and, consequently, is considered an easier case to control. Computation time is considered with respect to both the size of $\Delta$ and the number of iterations required to uncover the best solution. All versions

of PVTS tested include the construction of an initial feasible solution (state $S_0$) as described in Section 2.

### 5.1. *Variations in move sets for PVTS*

At any iteration of PVTS, we designate $k$ of the $q$ total potential actuator sites as *on* and $q - k$ of the total actuator sites as *off*. The actuator sites designated as *on* are the columns of $H$ used in solving the complex least squares problem for $f$. A natural choice for $\Delta$ is the set of all pair-wise swaps between actuator sites that are *on* and actuator sites that are *off*. In this case, the size of $\Delta$ for any state $S$ is $k(q - k)$. In our implementation of PVTS we examine the entire move set and select the best admissible move. Hence, we call this the *best swap* move set. The move attributes are the column indices that are swapped and these are added to a 2-dimensional tabu list $\tau$. A move is considered tabu if it is the inverse of any best swap move placed in $\tau$. Section 5.2 examines other ways in which these move attributes can be used to define $\tau$.

An alternative to this prescription for $\Delta$ is to address each of the sets—*on* and *off*—separately. That is, we examine two distinct move sets per iteration of PVTS. For example, we may first consider the set of actuator sites that are *off* and select the best admissible actuator site in *off* to add to the *on* set. (The best admissible move is the move $\delta \in (\Delta - \tau + \alpha)$ that increases the objective value the most or degrades it the least.) The size of $\Delta$ in this case is $q - k$. The resulting set *on* is now of size $k + 1$ and we next seek to find the *on* actuator site, among the original $k$ sites, that degrades the objective function value the least. The size of $\Delta$ for this second phase is $k$. We call this the *best add/best drop* move set and it is of size $q$. A final alternative is to use the same move sets just described, but apply them in the reverse order. We call this the *best drop/best add* move set and it is also of size $q$.

In both of these variations two separate tabu lists are maintained—one for best add and one for best drop. In the case of best add, the member of *off* that is added is not allowed to be selected for deletion for tabusize iterations. Similarly for best drop, the member of *on* that is deleted may not be added for tabusize iterations. Moreover, the value of tabusize for the added actuators tabu list can be shorter than the tabu list for the dropped actuators because, in our experiments, there are fewer actuators to drop (8 to 34) than actuators to add (94 to 68). However, since the search results proved insensitive to the value of tabusize we kept the two lists the same size.

A series of experiments were performed to gauge the performance of these three move sets. Clearly the best add/best drop and best drop/best add examine significantly fewer moves per iteration. The question is whether the quality of the solutions generated is significantly degraded by this decrease. The experiments varied in the number of actuators to be placed, namely, 8, 14, 16, 20, and 34. The maximum number of iterations allowed was 50. Longer runs would have been of interest for the 34 actuator problmes, but a single run of 50 interations took roughly 12 hours on our Sun Sparc network. For 8, 14, and 16 actuators we found that the value of tabusize was not important. In fact, only one move set experiment with tabu search yielded results that were better than simply performing a greedy local search (see Table 8). In the 20 and 34 actuator experiments the tabusize does play a role. However, extensive computational testing was not done due to long execution

*Table 2.* Best swap move set.

| Num. | Best obj. | Iter. | Tabusize |
|---|---|---|---|
| 8 | −19.5129 | 4 | 5 |
| 14 | −23.7466 | 14 | 8 |
| 16 | −24.4407 | 17 | 8 |
| 20 | −26.3280 | 22 | 10 |
| 34 | −32.1625 | 34 | 10 |

*Table 3.* Best add/best drop move set.

| Num. | Best obj. | Iter. | Tabusize |
|---|---|---|---|
| 8 | −19.5128 | 4 | 10/10 |
| 14 | −23.7464 | 11 | 10/10 |
| 16 | −24.3955 | 14 | 10/10 |
| 20 | −26.1152 | 46 | 10/10 |
| 34 | −31.2295 | 25 | 10/10 |

times. The values chosen performed extremely well given the 50 iteration limit, but were not particularly sensitive. That is, if a tabusize of 10 is specified an increase of up to 15 would not change the results given. We tried to use the smallest values of tabusize that yielded good solutions (since runtime increases with tabusize) and keep these values relatively constant across the experiments. The best swap move set yielded the following results.

The columns of Table 2 (as well as Tables 3–8) record the number of actuators placed; the best observed decibel reduction and the iteration at which it was first achieved; and the size of the tabu list respectively. The above results for interior noise dB reduction are not identical to those in Section 4. In Section 4, the number of iterations was not always 50. In Table 1 the superior results for 16 and 20 actuators for the best swap move set were found by running 70 iterations of PVTS. Next we compare the performance of the best swap move set with the best add/best drop move set and with the best drop/best add set. The computational results for the best add/best drop are given in Table 3.

Comparing the data from the two structures, it can be seen that PVTS with the best add/best drop move set never finds a solution better than that of PVTS with the best swap move set. The greatest difference in solution quality lies in the 34 actuator case, thus implying that as the number of actuators increases, the solutions found by the best add/best drop procedure may not be as good as the best swap procedure. The most obvious discrepancy lies in the computation time of the two searches. In the best add/best drop procedure, since only $q$ exchanges are required at each iteration versus $k(q-k)$ exchanges for the best swap procedure, the computation time is greatly diminished. In the cases where fewer actuators are placed, this difference is not significant, but in the 20 and 34 cases, the best add/best drop procedure steps through the search about 10 times faster per iteration than the best swap procedure.

*Table 4.* Best drop/best add move set.

| Num. | Best obj. | Iter. | Tabusize |
|---|---|---|---|
| 8 | −19.0984 | 5 | 10/10 |
| 14 | −23.7464 | 11 | 10/10 |
| 16 | −24.3955 | 12 | 10/10 |
| 20 | −26.3282 | 18 | 10/10 |
| 34 | −31.004 | 29 | 10/10 |

The best drop/best add procedure produces similar results to that of the best add/best drop structure owing to the fact that the only difference between the two methods is the order in which the add and drop are performed. The computational data for the best drop/best add method is presented in Table 4.

As in the best add/best drop procedure, the objective values are never better than that of the best swap procedure, and in some of the cases the search achieves the same solution. Once again, the computation time is lower by about a factor of 10 over the best swap procedure. The comparison between best drop/best add and best add/best drop is less clear. For cases 14 and 16 the two result in the same solution at nearly the same iteration. Best add/best drop wins for cases 8 and 34, but loses in case 20.

### 5.2. *Variations in tabu list for best swap PVTS*

Another way in which PVTS can vary in its approach lies in the method in which exchanges between *on* and *off* are designated tabu. In this section we examine three variations of the assignment of move attributes to a tabu list for PVTS with the best swap move set. In all cases the initial feasible solution (state $S_0$) is constructed as described in Section 2. As we described earlier a move is considered tabu if the inverse of the move is in $\tau$. We call this the *inverse* tabu restriction. We have recorded this method's performance in Table 2.

Our first variation on the best swap/inverse method of Table 2 is to prohibit moves containing either (but not necessarily both) of the dropped or added actuators for tabusize iterations. For example, using the best swap/inverse method, if actuator 5 were turned off and actuator 7 were turned on, then the exchange of 7 and 6 at the following iteration would be permitted, and the exchange of 6 and 5 could be made at the iteration following that, returning us to the solution attained only 3 iterations previously. By using the method outlined above for placing moves on the tabu list, the moves following the 7 and 5 exchange would be considered tabu since no moves containing either 7 or 5 would be permitted. Consequently, the search would be encouraged to move into other areas of the solution space and perhaps toward solutions better than those located with the best swap/inverse method. Of course the tabu status of any of these moves may be overcome by meeting the aspiration criterion. Due to the increased restriction placed upon the search, we will designate this the *strict* tabu restriction. Note that a maximum tabusize of 5 is given. This is roughly equivalent to a tabusize of 10 for the less restrictive versions of the search. The experimental results obtained by PVTS with the best swap/strict method are recorded in Table 5.

*Table 5.* PVTS best swap/strict results.

| Num. | Best obj. | Iter. | Tabusize |
|---|---|---|---|
| 8 | −19.5129 | 4 | 2 |
| 14 | −23.8586 | 24 | 5 |
| 16 | −25.1868 | 39 | 5 |
| 20 | −26.3280 | 19 | 5 |
| 34 | −30.7684 | 49 | 5 |

The solution quality of the PVTS with best swap/strict dominates all previous methods examined for cases 8, 14, 16, and 20. In fact, it is the only version of PVTS that found the best known solutions for the 14 and 16 actuator cases in fewer than 50 iterations. But, in the 34 case the best solution is nearly the worst found (only the greedy search is worse). This demonstrates the potential for the strict tabu restriction to uncover better solutions in cases with not too many local minima or without the presence of wide basins. We suspect that for more than 16 actuators the number of local optima continues to increase and that basins may exist. A basin (cf. Battiti and Tecchiolli, 1992) is a collection of small valleys within a larger valley. PVTS is not designed to escape from wide basins. That is, a tabu list is effective in escaping from isolated local optima but may not be sufficient to escape from a basin containing several local optima. The strict tabu restriction forces the search to explore a greater area of the solution space yet, in doing so, it may not examine the finer areas of the space. The computation time for the PVTS with best swap/strict is about the same as that of the PVTS with best swap/inverse.

Two variations on the placement of move attributes for best swap on the tabu list are examined next. Instead of placing both move attributes (add and drop) on the tabu list as in best swap/inverse and best swap/strict we place a single attribute on the tabu list (either add or drop). When an add move attribute is placed on the tabu list then the associated actuator cannot be dropped for tabusize iterations, while in the case of the drop move attribute the dropped actuator is not allowed to be added tabusize iterations. We call these variations *best swap/add* and *best swap/drop* respectively. The implementation of these tabu restrictions is like that of the strict tabu restriction. For example, if actuator site 7 is tabu in best swap/add then no move may contain 7 as a drop move attribute. The computational results for the add tabu structure are given in Table 6.

*Table 6.* PVTS best swap/add results.

| Num. | Best obj. | Iter. | Tabusize |
|---|---|---|---|
| 8 | −19.5129 | 4 | 5 |
| 14 | −23.7465 | 11 | 5 |
| 16 | −24.4407 | 16 | 5 |
| 20 | −26.3284 | 21 | 8 |
| 34 | −32.1075 | 42 | 10 |

*Table 7.* PVTS best swap/drop results.

| Num. | Best obj. | Iter. | Tabusize |
|---|---|---|---|
| 8 | −18.8480 | 39 | 5 |
| 14 | −21.3229 | 9 | 5 |
| 16 | −22.1631 | 22 | 5 |
| 20 | −24.1409 | 35 | 8 |
| 34 | −29.2625 | 37 | 10 |

The results for the best swap/add tabu method are very similar to that of the best swap/inverse method. The solution quality is nearly the same and the iteration at which the best observed solution is uncovered are comparable except for the 34 actuator case (42 for add and 34 for inverse). The solution times were nearly the same as well.

The best swap/drop method prohibits the adding of the dropped actuators for tabusize iterations. Like the best swap/add procedure, the best swap/drop procedure works like strict tabu, except that it prohibits any move involving the most recently dropped actuators. The experimental data for the best swap/drop method are shown in Table 7.

The performance of the best swap/drop method is poor. The solution quality is dominated by many of the other methods we have examined and the computation time is roughly the same as best swap/inverse. We did not experiment further with this method although it can be argued that tabusize should be larger for this method than for best swap/add since there are a larger number potential add move attributes from which to select. However, based on experience with this data set, the performance of PVTS has not been sensitive to the parameter tabusize.

Lastly we report on the performance of a fourth variation of PVTS with a best swap move set in which the *tabusize* is zero. Variation four then is equivalent to a greedy local search. A greedy local search stops at a local optimum and has no mechanism for finding better local optima other than restarting the search in another area of the solution space. The search proceeds until it reaches an iteration in which there are no improving moves in the neighborhood of the current solution. When this occurs, the search stops and the best solution is the one found at the most recent iteration. One of the purposes of this experiment is to gauge the hardness of the ASAC location problem for these 200 Hz data sets. The results for the greedy local search (PVTS with a best swap move set and $tabusize = 0$) are given in Table 8.

*Table 8.* Greedy search results (no tabu list).

| Num. | Best obj. | Iter. | Tabusize |
|---|---|---|---|
| 8 | −19.5129 | 4 | 0 |
| 14 | −23.7465 | 11 | 0 |
| 16 | −24.3956 | 12 | 0 |
| 20 | −23.3469 | 6 | 0 |
| 34 | −28.5558 | 13 | 0 |

The greedy search produces solutions as high in quality as PVTS with best swap (Table 2) for the smaller order cases, 8, 14, and 16, but for the 20 and 34 cases, it produces solutions of significantly lower quality. Our suspicion is that the number of local optima increases with $k$. To test this hypothesis, we executed 20 replications of the greedy local search (PVTS best swap with $tabusize = 0$) for 8, 14, and 16 actuators, each starting from a randomly generated solution. For 8 actuators the average decibel value was $-18.76$ and the average number of iterations was 8.1. The $-19.5129$ solution was the final solution in 12 of the 20 replications and there were only 7 local optima. This supports the conjecture that the terrain of the objective function space contains few local minimums for the 8 actuator case. For 14 actuators the average decibel value was $-22.39$ and the average number of iterations was 12.9. The $-23.7465$ solution was discovered in only 3 of the 20 replications and there were 18 local optima. This supports the notion that as the number of actuators is increased the terrain of the objective function space is bumpier and that there are more local optima. In addition, we note that the aspiration criterion was never invoked in any of the computational experiments of Section 5 (even for the 20 and 34 actuator cases). We believe this supports the notion that even when the objective function terrain becomes bumpier it is still relatively uncomplicated.

### 5.3. *Summary of numerical experiments*

We summarize the results of Tables 2 through 8 in Table 9 with respect to solution quality. Recall that add/drop and drop/add are an order of magnitude faster than any of the best swap variations. This is an important point since the computation time (on a Sun Sparc Classic) for 34 actuators is roughly 12 hours for 50 iterations of any of the best swap variations. With regard to solution quality best swap/strict is the clear winner for 20 actuators or less. More work needs to be done for the 34 actuator case. We note that, although we evaluate the performance of our heuristic techniques by declaring one technique superior to another when its solutions are better by as little as .1 dB, in practice laboratory tests are not able to distinguish between solutions this finely. In Section 7 we discuss some ongoing efforts in developing tabu search codes that can escape from basins.

*Table 9.* Comparison of best objective values (* indicates best value).

| | 8 | 14 | 16 | 20 | 34 |
|---|---|---|---|---|---|
| BS/Inverse | −19.513* | −23.746 | −24.441 | −26.328* | −32.162* |
| Add/Drop | −19.513* | −23.746 | −24.396 | −26.115 | −31.229 |
| Drop/Add | −19.098 | −23.746 | −24.396 | −26.328* | −31.000 |
| Greedy | −19.513* | −23.746 | −24.396 | −23.347 | −28.556 |
| BS/Strict | −19.513* | −23.859* | −25.187* | −26.328* | −30.768 |
| BS/Add | −19.513* | −23.746 | −24.441 | −26.328* | −32.107 |
| BS/Drop | −18.848 | −21.323 | −22.163 | −24.141 | −29.262 |

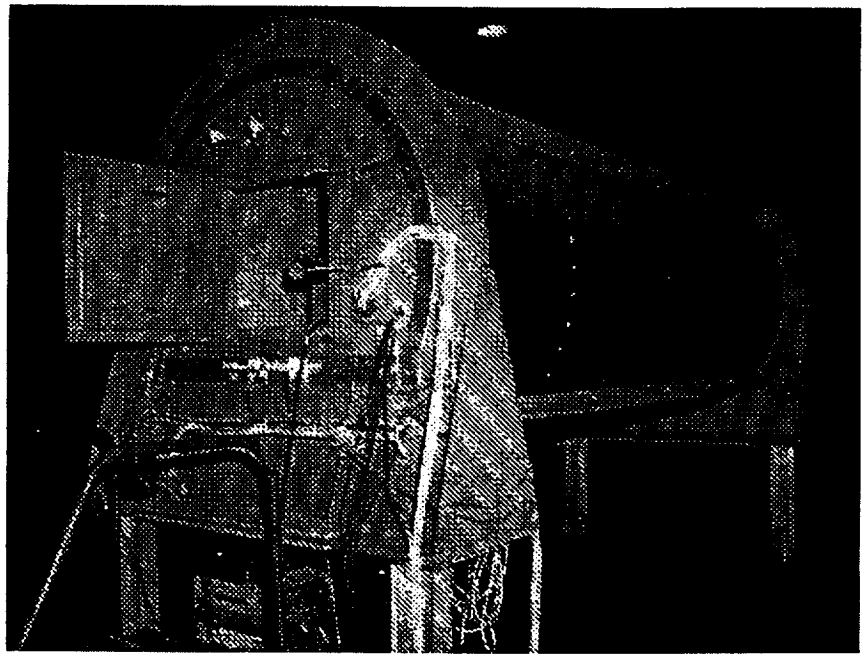


*Figure 5.* Laboratory cylinder for ASAC test.

## 6. Laboratory test validation

As we mentioned earlier, a set of experiments with a laboratory test article (see figure 5) have been performed and reported on in Palumbo et al. (1996). For these experiments the transfer matrices were generated experimentally. The predicted performance of the best actuator sites found for the mathematical model by tabu search are shown to correlate well with the actual measured performance.

The cylinder in figure 5 is 3.6 m long and 1.68 m in diameter. The outer shell is a 9 layer filament wound graphite epoxy composite. Total skin thickness is 1.7 mm. In an effort to make the cylinder as similar as possible to a commercial fuselage the cylinder is stiffened with composite stringers and ring frames. To complete the fuselage effect, 3 inner trim panel sections are attached. Each trim panel has a honeycomb core sandwiched between two graphite epoxy laminate sheets. The interested reader is referred to Lyle and Silcox (1995) for a detailed description of the laboratory test article. The monopole noise source was a 100 watt electrodynamic loudspeaker. An array of 8 piezoceramic patches were installed on one of the trim panels. A set of 6 microphones were swept along a boom throughout the interior of the cylinder (near the shell) and data was taken at 462 locations. The 462 transfer functions at these locations were used to construct transfer matrices (462 by 8) for three frequencies—210 Hz, 230 Hz and 275 Hz. The 210 Hz and 275 Hz were expected to be similar to the 200 Hz and 275 Hz cases studied by the computer generated

mathematical model and reported on in Section 4. The 230 Hz case has weak acoustic modes but strong structural modes.

The laboratory test was done with only eight potential actuator sites of which four were to be selected. This was somewhat disappointing since our mathematical model that simulated the laboratory experiment showed that many more actuators should be used. One of the reasons so few actuator sites were tested was the lack of confidence in the optimization techniques and the modeling assumptions necessary to provide a tractable model. This is not uncommon and Padula and Kincaid (1995) report on several other modeling experiences at NASA Langley. Hence, one of the main purposes of the laboratory experiments for us was to validate our optimization approach and the modeling assumptions. (An example of a modeling assumption is the least squares approach for determining a solution $\vec{f}$. Here it is assumed that $H\vec{f} = \vec{p}$ is an appropriate description of the interactions between the control forces and the interior noise field.)

A second optimization problem arose during the laboratory test. The initial transfer matrices were constructed by moving a boom through the cylinder and taking measurements at 462 pre-determined locations. However, the laboratory controller was only able to receive and process noise measurements from 8 microphones. The question that arose was at which of the 462 locations should 8 permanent sensors be located. This sensor selection problem is the subject of Kincaid and Laba (1997) and will not be discussed here. Once the 'best' set of 8 microphones was determined we solved the actuator selection problem with PVTS. We verified that the PVTS solution was optimal by testing all 8 choose 4 combinations of actuators. Laboratory results for this case are reported in Table 10.

Table 10 catalogs the performance of PVTS versus standard acoustical techniques in the selection of the best set of 8 sensors. The acoustical techniques (cf. Palumbo et al., 1996) involve engineers examining visual and numerical displays of the modal decomposition of the interior pressure field $\vec{p}$ and the individual actuator responses. Dominant modes in the interior pressure field are matched with dominant modes in the actuator responses. The columns marked PVTS compare the predicted noise reduction versus the actual noise reduction measured in the laboratory. The column marked modal analysis show noise reduction levels achieved in laboratory tests run prior to PVTS selection of actuator and sensor locations.

The good news in Table 10 is that the measured values for PVTS are nearly 1 dB better than the modal decomposition analysis solutions. Hence, even for this relatively easy case (only 4 actuators) the optimization approach proved superior. A second positive feature is that the predicted performance, although always at least a 1 dB overestimate, is still a reasonable

*Table 10.* Experimental results with actuator/sensor location.

| | PVTS | | Modal analysis |
|---|---|---|---|
| Freq. (Hz) | Predicted | Measured | measured |
| 210 | −5.3 dB | −4.4 dB | −2.1 dB |
| 230 | −3.8 dB | −2.5 dB | −1.2 dB |
| 275 | −5.7 dB | −3.9 dB | −2.7 dB |

predictor of the measured performance. This provides evidence that the model assumptions made in the optimization approach are not too far off. A third positive result of these experiments is that the acoustic engineers now believe that the optimization approach is an important contributor in finding good solutions the ASAC problems. Further experiments with over 100 potential actuator sites are in the planning stages.

## 7. Discussion and conclusions

We have successfully modeled the ASAC actuator location problem as a discrete optimization problem. In addition we have shown that PVTS provides a means for determining good solutions for $k < 34$. The laboratory experiments with 4 actuators support both the validity of the modeling assumptions as well as the superiority of solutions found by PVTS over solutions determined by engineering judgement alone. It remains to be shown that the solutions found by PVTS for $k \gg 4$ will simultaneously reduce interior noise and shell vibrations as predicted by the data generated by the mathematical model. Based upon the success of the laboratory experiments with $q = 8$ and $k = 4$ actuators, additional experiments with roughly $q = 100$ actuators are planned.

There are several avenues of research that we are currently pursuing. First it is important to solve the complex least squares problem (Eq. (1) of Section 2) as quickly as possible. Each time we examine a move $\delta \in \Delta$ we solve the least squares problem by computing the inverse of $H^{k*}H^k$ in Eq. (2). The total number of least squares problems solved in PVTS per iteration is $k(q-k)$. However, since only one column of $H^k$ changes with $\delta$ it should be possible to compute the solution to (2) more efficiently. We have attempted to do this by computing the $QR$ factorization of $H^k$ (hereafter we will suppress the superscript $k$ on $H$) and update $Q$ and $R$ for each move.

$Q$ is a unitary $p \times p$ matrix while $R$ is a $p \times k$ complex matrix. $Q$ can be partitioned into $Q_1$, the first $k$ columns, and $Q_2$, the remaining $p-k$ columns. $R$ can also be partitioned into $R_1$, the first $k$ rows of $R$, a $k$ by $k$ right triangular complex matrix, and $R_2$, the remaining $p-k$ rows of zeros. Since the $Q_2$ portion of $Q$ will be multiplied by the zero rows of $R_2$, these two portions may be eliminated. Thus the resulting factorization of $H$ is $Q_1 * R_1$. To solve the equation $H^*H\vec{f} = -H^*\vec{p}$, the factorization can be substituted in (2) for $H$ and $H^*$, resulting in the equation

$$R_1^* R_1 \vec{f} = -R_1^* Q_1^* \vec{p}$$

If $R_1^*$ is invertible the equation can be rewritten as

$$R_1 \vec{f} = -Q_1^* \vec{p}$$

which, since $R_1$ is a right triangular matrix can efficiently be solved by back substitution. Golub and VanLoan (1989) describe a procedure for appending a column (or deleting a column) of $H$ and efficiently updating (downdating) $Q$ and $R$ whenever $H$ is a real matrix. We extended this procedure to the complex case. Unfortunately, our computational experiments with these routines have not resulted in any significant time savings. Clearly,

there should be time savings available by not having to reinvert $H$ (or resolve for $Q$ and $R$) for each move that is examined.

A second avenue of current research is to include more advanced tabu search capabilities. PVTS works well for 20 or fewer actuators but seems to have problems starting at 34 actuators. In addition to frequency-based and recency-based diversification schemes, we are examining the reactive tabu search scheme of Battiti and Tecchiolli (1994). We are particularly interested in the basin escape mechanism. Battiti and Tecchiolli (1994) suggest a random generation of moves which, when applied to the current solution, would direct the search in a random direction with hopefully a high probability of finding a new solution outside of the large basin. One drawback to this method is that there is no guarantee of moving the search outside of the basin, and, even if this is achieved, it is possible that the search may restart close to the basin and descend back into it a number of iterations later. A second method of restarting the search is frequency-based (cf. Laba, 1996), which relies upon the frequency with which certain moves have been performed. We plan to test the frequency-based basin escape mechanism on larger instances of the ASAC location problem.

A third problem of interest is the ASAC sensor location problem. We have implemented relatively straightforward adaptations of the codes for the ASAC actuator location problems to the sensor location problem. However, a systematic study of the interactions between the two solution procedures is warranted. It appears that the correct sequence is to first solve the $k$ (out of $q$) actuator location problem with the full complement of potential sensor locations (462 in our example) and then find the set of sensors that most closely approximates the full model for the best set of $k$ actuators.

There are a host of other interesting issues as well. What are good ways to model the selection of actuators that must work for multiple frequencies? Should we pick certain actuators for certain frequencies; or, are there particular actuators that will work well for multiple frequencies? How should the grouping problem for minimizing the number of controller channels be addressed? How many actuators is too many? Is there a better model (than the linear regression model) for finding the control forces $\vec{f}$? There is much to be done, both from a modeling standpoint and from a heuristic development standpoint.

## References

——, "Noise and vibration control: Three technologies are combined to provide significant reduction of aircraft cabin noise and vibration levels," *Aerospace Engineering*, vol. 14, no. 9, pp. 10–12, 1994.

R. Battiti and G. Tecchiolli, "The reactive tabu search," *ORSA Journal on Computing*, vol. 6, pp. 126–140, 1994.

R.H. Cabell and H.C. Lester, "Investigation of Force Distributions for Interior Noise Control Using a Neural Network," *Proceedings of the Second Conference on Recent Advances in Active Control of Sound and Vibration*, Blacksburg, VA, April 1993, pp. 55–69.

R.H. Cabell, H.C. Lester, G.P. Mathur, and B.N. Tran, "Optimization of Actuator Arrays for Aircraft Interior Noise Control," *Proceedings of the 15th AIAA Aero-Acoustics Conference*, Long Beach, CA, AIAA-93-4447, 1993.

R. DeMeis, "Quieting cabin noise," *Aerospace America*, pp. 20–21, Feb. 1995.

S.P. Engelstad, K.A. Cunefare, S. Crane, and E.A. Powell, "Optimization strategies for minimum interior noise and weight using FEM/BEM," *Internoise*, vol. 2, pp. 1205–1208, 1995.

C.R. Fuller and A.H. von Flotow, "Active control of sound and vibration," *IEEE Control Systems*, pp. 9–19, Dec. 1995.

F. Glover, "Tabu search, Part I," *ORSA J. on Computing*, vol. 1, pp. 190–206, 1989.

F. Glover, "Tabu search: A tutorial," *Interfaces*, vol. 20, pp. 74–94, 1990.

G. Golub and C. VanLoan, *Matrix Computations*, Johns Hopkins University Press: Baltimore, MD, 1989.

R.K. Kincaid, "Actuator Placement for Active Sound and Vibration Control of Cylinders," NASA ASEE Contractor Report 198210, 1995, p. 85.

R.K. Kincaid and R.T. Berger, "The damper placement problem on space truss structures," *Location Science*, vol. 1, pp. 219–234, 1993.

R.K. Kincaid and K.E. Laba, "Reactive Tabu Search and Sensor Selection in Active Structural Acoustic Control Problems," Working Paper, Department of Mathematics, College of William and Mary, 1997.

K.E. Laba, "The Application of a Tabu Search on the Problem of Active Structural Acoustic Control," Undergraduate Honors Thesis, Department of Mathematics, The College of William and Mary, 1996.

H.C. Lester and R.J. Silcox, "Active Control of Interior Noise in a Large Scale Cylinder Using Piezoelectric Actuators," *Proceedings of International Specialists Meeting: Rotorcraft Acoustics and Rotor Fluid Dynamics*, Philadelphia, PA, Oct. 1991.

P. Leug, "Process of Silencing Sound Oscillations," U.S. Patent No. 2,043,416, 1936.

K.H. Lyle and R.J. Silcox, "A Study of Active Trim Panels for Noise Reduction in an Aircraft Fuselage," Paper no. 951179 in SAE General, Corporate and Regional Aviation Meeting and Exposition, May 1995.

S.L. Padula and R.K. Kincaid, "Aerospace Applications of Integer and Combinatorial Optimization," NASA Technical Memorandum 110210, 1995.

R.J. Silcox, C.R. Fuller, and H.C. Lester, "Mechanisms of active control in cylindrical fuselage structures," *AIAA Journal*, vol. 28, pp. 1397–1404, Aug. 1990.

R.J. Silcox, S. Lefebvre, V.L. Metcalf, T.B. Beyer, and C.R. Fuller, "Evaluation of Piezoelectric Actuators for Control of Aircraft Interior Noise," AIAA paper no. 92-02-091, Presented at *DGLR/AIAA 14th Aeroacoustics Conference*, Aachen, Germany, May 11–14, 1992.

# Reactive Tabu Search and Sensor Selection in Active Structural Acoustic Control Problems

Rex K. Kincaid † and Keith E. Laba ‡
Department of Mathematics
The College of William and Mary
Williamsburg, Virginia 23187-9795

**Abstract.**

A Reactive Tabu Search (RTS) is examined. In addition to a dynamic tabu tenure RTS also detects when the search has entered an unproductive area of the search and restarts RTS based on distinctive features of the unproductive area. We explore the effectiveness of RTS over a more traditional static tabu search for a two variable unconstrained discrete optimization model with 9 nearly identical minima and 513 other local minima. One of the key features of this problem is that the two dimensional domain allows us to provide graphical descriptions of the performance of RTS. We then apply RTS to a sensor selection problem in active structural acoustic control. The objective in this problem is to select a set of 8 sensors out of 462 potential sensor locations so that the noise measured at the 8 chosen sensors is as close as possible to the noise measured at all 462. Computational experiments for data taken from a laboratory test article at NASA Langley Research Center are provided.

NAG1-1783

† The author was supported in this research by NASA grant #~~501291~~.

‡ The author completed this work while a LARSS student at NASA-LaRC.

## Introduction

Tabu search and, more generally, adaptive memory programming continues to have startling success in efficiently generating high-quality solutions to difficult practical optimization problems. Glover (1996) provides forty-two vignettes each of which describes a different application of tabu search by researchers and practitioners. The applications include standard problems such as the p-median problem, job shop scheduling and the quadratic assignment problem, as well as unusual applications including polymer chemistry, forest management, and the control of flexible space structures.

Here we study a recent variant of tabu search called Reactive Tabu Search (RTS) which was developed by Battiti and Tecchiolli (1992, 1994a). RTS has since been applied in a variety of problem settings including neural networks (Battiti et al. 1994b and Battiti and Tecchiolli 1995a), 0/1 knapsack (Battiti and Tecchiolli 1994a) and the quadratic assignment problem (1994a,1994c). Battiti and Tecchiolli (1995b) have compared the performance of RTS against alternative heuristics. Two of the features of RTS in which we are particularly interested are the dynamic aspect of the length of the tabu list (or tabu tenure) and the mechanisms for detecting and escaping from basins. Our version of RTS differs from the one described by Battiti and Tecchiolli in many technical details (e.g. we don't use hashing to manage our memory structures) but the spirit of our approach is the same as theirs and so we use the name RTS to link our work with theirs.

In section 1 of this manuscript we describe and test a version of RTS for determining minima of a highly multi-modal function. Following Battiti and Tecchiolli (1994a) the increase in the size of the tabu list increases quickly as repeated solutions (cycles) are encountered and decreases more slowly when repeated solutions are not found. RTS detects unproductive regions of the search space and restarts the search based upon distinctive features of the unproductive region. An example of an unproductive region is a region of similar valued local minimum forming a bumpy plateau. We call this a basin. If the basin is large enough it is unlikely that increasing the length of the tabu list (or tabu tenure) would be sufficient to escape and the search should be restarted. An additional feature of the function in section 1 is that the domain is $\mathcal{R}^2$ allowing us to illustrate graphically our test results. We compare the performance of RTS against a traditional static tabu search in order to illustrate the effectiveness of the RTS features. In section 2 we apply RTS to a sensor selection problem in active noise control. The data is taken from laboratory tests conducted at NASA-Langley Research Center.

## 1. Reactive tabu search with basin escape for Levy No. 3

The intent of this section is to illustrate the salient features of RTS. We do this by examining the terrain of a function $f : [-10, 10] \times [-10, 10] \rightarrow \mathcal{R}$. Since we can plot the domain and range of $f$ together in $\mathcal{R}^3$ we are able to demonstrate key features of the search mechanism graphically.

1.1 Description of discretized test function

The function we chose is a frequent test problem for global (continuous) optimization routines (see Hansen 1992, Jansson and Knuppel 1994, and Levy et al. 1981). The function is

$$f(x,y) = \sum_{i=1}^{5} i * cos[(i-1)x + i] * \sum_{j=1}^{5} j * cos[(j+1)y + j]; x, y \in [-10, 10]$$

and is referred to as Levy No. 3 in the literature †. The resolution is a grid of 50 points along each axis (every 0.4 units) for a total of 2500 plotted points. We make use of this resolution in section 1.2 in conjunction with the structure of our tabu list.

For our computational experiments we require a discrete optimization problem. To discretize the domain of $f(x,y)$ for $x, y \in [-10, 10]$ we impose a regular grid of 800 points along each axis (every 0.025 units) for a total of 640,000 points. (The 2500 point domain was too small to pose as an interesting optimization problem, but the 50 by 50 grid is used later in another aspect of the search.) It is known that the continuous version of Levy No. 3 has 9 global minima and 760 local minima. The nearest points to the 9 global optima on the 800 by 800 grid are given in Table 1.1. There are 522 local minima out of the 640,000 grid points.

| x | y | f(x,y) |
|---|---|---|
| 4.975 | -1.425 | -176.540 |
| -1.300 | -1.425 | -176.505 |
| -7.600 | -1.425 | -176.458 |
| 4.975 | 4.850 | -176.397 |
| -1.300 | 4.850 | -176.363 |
| -1.300 | -7.700 | -176.354 |
| -7.600 | 4.850 | -176.316 |
| -7.600 | -7.700 | -176.307 |

Table 1.1 9 points on grid closest to 9 global optima

† Figure 1.1 is plotted using the Maple command `plot3d(sum(i*cos((i-1)*x+i),i=1..5) * sum(j*cos((j+1) *y+j),j=1..5), x=-10..10,y=-10..10,grid=[50,50]);` which plots $f(x,y)$ over the interval $[-10, 10]$ for both $x$ and $y$.

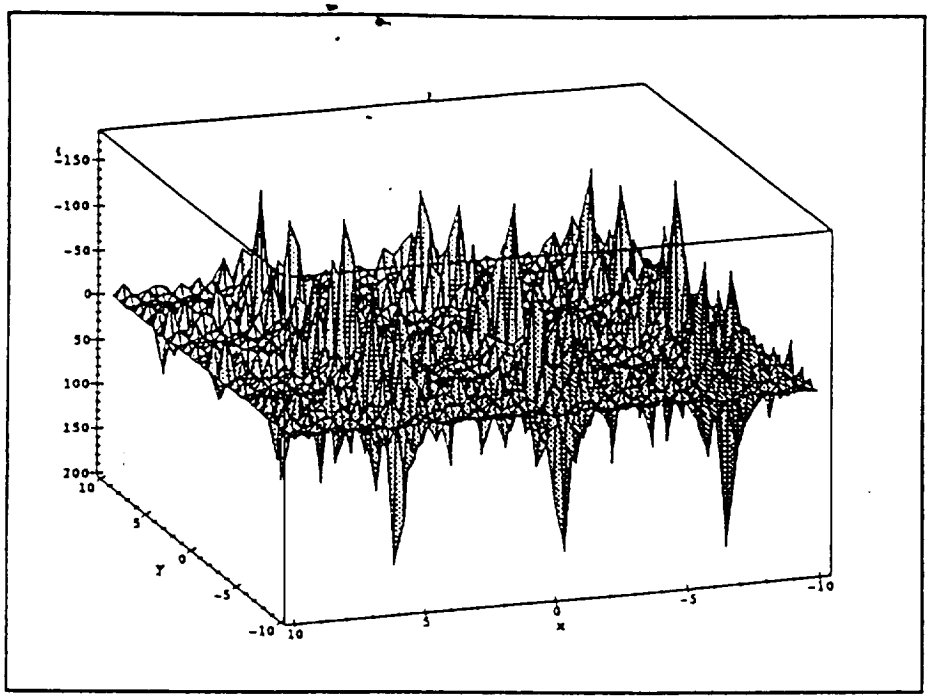



Figure 1.1 Plot of $f(x, y)$ evaluated over 50 by 50 grid

1.2 Basic features of tabu search procedures

We begin by describing a neighborhood structure for the 800 by 800 grid version of Levy No. 3. Given an interior point $(x, y)$ of the grid the neighbors of this point are all of the compass points–North, Northeast, East, Southeast, South, Southwest, West, and Northwest–generated by adding or subtracting the grid size (0.025) from $(x, y)$. For example, the grid point $(x, y+0.025)$ is the North neighbor of $(x, y)$. At each iteration of tabu search the eight neighbors of the current point are evaluated and the best neighbor among the non-tabu and tabu neighbors, which satisfy the aspiration criteria, is selected (where best means lowest $f(x, y)$ value). The aspiration criteria checks to see if the function value at a tabu neighbor is better than the best observed function value. If so, the tabu neighbor is said to satisfy the aspiration criteria. Once a best neighbor is selected it becomes the new current point and the $(x, y)$ coordinates of the best neighbor are placed on the tabu list.

The tabu list is a collection of $(x, y)$ coordinates of previously selected best neighbors. This differs from the usual tabu search convention of placing attributes on the tabu list. Here, since the solutions have only two components–x and y–we can maintain a list of the complete solution rather than attributes of the solution. Experiments utilizing the directional attributes (North, South, etc) proved uninteresting. We believe this is due to the limited number of attribute values. Below we describe the tabu list structure utilized by both our reactive tabu search and our static tabu search.

We store each of the points on the tabu list so that we can keep track of points that are visited more than once. The idea is that if the search continues to return to the same point several times then it is likely that the search is cycling and one of the functions of a tabu list is to avoid cycling. Here, instead of maintaining the frequency with which each point (640,000 of them) is uncovered by the search, we maintain the frequency distribution of collections of points. In particular, we place a coarser 50 by 50 grid (see Figure 1.1a) over the finer 800 by 800 grid. Each point of the coarse grid is mapped to a 16 by 16 set of grid points (256 point in total) on the fine grid. We maintain the frequency data for each of the 50 by 50 grid points (2500 total points). We call each of these coarse grid points a *zone*. Moreover, we make further use of these zones in conjunction with the tabu list. A pass through either the reactive or static versions of tabu search may terminate for a variety of reasons (to be described later) and restarted at a new point. Upon completion of a single pass the frequency data on the 2500 zones is used to update a matrix describing which zones are to remain tabu in subsequent passes. In particular if the frequency of a zone on a pass is greater than 16 (the number of grid points on a horizontal or vertical line through a zone) then that zone will be permanently labeled tabu. So, in addition to the tabu list the search also checks to see if each neighbor is a member of a tabu zone.

One of the situations in which a restart for both the reactive and static versions of tabu search is needed is when we determine that the search is near an attractor from which neither the static tabu list size nor the dynamically changing tabu list size will let the search escape. In other words, the search is stalled in a deep valley or a collection of deep valleys. Here again, our searches rely on zones rather than on individual points to attempt to escape the attractor. If the frequency of any zone is greater than 512 then we restart the search at a new (randomly generated) grid point. The choice of 512 follows from the observation that if each point in a zone is visited (on average) twice then the search is likely to be either cycling or near an attractor.

Both of our tabu search procedures rely upon a randomly generated point to begin the search. A uniform $[-10.0, 10.0]$ distribution is sampled for an initial $x$ coordinate and an initial $y$ coordinate. In a real application, as is described in Section 2, problem specific information is exploited to generate a high quality initial solution. Starting points are also needed for restarting the search when the search is near an attractor or, in the reactive version, when a basin is detected. In both of these cases information about the search terrain is exploited to generate solutions for restarting the search. When an attractor is detected starting points from non-tabu zones are generated randomly among the non-tabu zones on the 800 by 800 grid. For a basin the reactive search generates a new starting point based upon particular features of the basin.

When an attractor is detected both the reactive and static versions of tabu search generate points for a restart in the same way. We call the sequence of moves generated until a restart is needed a *pass* of the search. Upon completion of a single pass we compute the frequency counts of solutions uncovered in each zone.

If the frequency of a zone is greater than 16 for that pass then that zone is labeled tabu in subsequent passes. 16 is the minimum number of moves required to cross a zone. In addition, we do not allow a point for a restart to be a point in any of the eight adjacent neighboring zones of any tabu zone. The idea is to keep the random starting point away from zones that have been previously searched. The $x$ and $y$ coordinates of a potential starting point are each generated from a uniform $[-10, 10]$ distribution. If the point $(x, y)$ is a member of a tabu zone or any of the eight neighboring zones then the point is discarded. This process is repeated until an appropriate starting point is found. Although more sophisticated techniques could have been employed–by sampling only from zones that are not tabu nor adjacent to tabu zones–our computational experiments showed than no more than ten tries were required to find a new starting point even after 50 passes of the search.

1.3 Reactive versus Static Tabu Search

In section 1.4 we report on computational experiments that compare and contrast a reactive tabu search (RTS) with a traditional static tabu search. In this section we describe the structural differences between RTS and a static tabu search. RTS and the static scheme share the same neighborhood structure, tabuzones, and aspiration criteria. However, the static search has a fixed tabu list size of 1500 and no capability to detect or escape from basins. The tabu list size of 1500 was determined by experimentation. Indeed the performance of the static tabu search is sensitive to this parameter's value. In our experiment 1500 produced the best results. Thus, the static tabu search does not have the ability to alter the length of the tabu list, nor does it make any attempt to discover basins and escape from them. The only mechanism for terminating a pass of the static search is the detection of cycling associated with stalling in a deep valley or a collection of valleys (frequency-based diversification). As described in section 1.2 if the frequency of a zone is greater than 512 then we restart the search at a new randomly generated grid point. To avoid extremely long passes we set a maximum of 5000 iterations per pass. Finally, the static tabu search generates its random starting points just as in RTS.

Clearly more sophisticated versions of a static tabu search exist. We made no attempt to include recency-based diversification or intensification schemes. However, these additional features could also be included in RTS and were not. The main intent here is to highlight the distinctive features of RTS. The name RTS was coined by Battiti and Tecchiolli (1994) who first described the procedure. The basic features include a tabu list whose length is dynamic and a mechanism for detecting and escaping from a basin of local minima. We initialize the length of the tabu list, *tabusize* to 1000. If the frequency of a zone is larger than 256, indicating that each of the 256 points in the zone are visited once on average, then we conclude that we are likely to cycle and tabusize is increased. That is, $tabusize_{new} = 1.2 * tabusize_{old}$. Conversely if tabusize has not increased for tabusize iterations then we decrease tabusize. That is, $tabusize_{new} = 1/1.2 * tabusize_{old}$.

There are two ways for a single pass of RTS to terminate–if a basin is found or

if the frequency of a zone is too large. A basin is a collection of local minimums of similar magnitude from which it is difficult for a tabu search armed only with a tabu list and an aspiration criteria to escape. The reason that basins are so difficult to escape from tabu search is that, as the search moves through the basin, it is placing the most recently performed moves on the tabu list. Basins are so wide, however, that as the search makes its way further into the basin, the moves performed as the search first entered the basin will no longer be tabu, thus allowing the move to be made again. If enough of the moves leave the tabu list, eventually, the search may perform the exchanges again, resulting in a return to the beginning of the basin. If the tabu list is increased in length as in RTS, the moves at the beginning of the basin may not be allowed, thus preventing a return to the beginning of the basin. Since the basins may be quite wide and require many moves for the search to make its way through it, however, the length of the tabu list may become so large that every possible move becomes tabu, thus preventing the search from going any further. Even if this extreme case is not realized there is a point at which the length of the tabu list may inhibit the ability of the search to effectively explore and escape the basin.

For Levy No. 3, we say that the search has entered a basin if the difference between the best value of $f(x, y)$ and the worst value of $f(x, y)$ is less than 30.0 for more than 1024 iterations. 1024 was selected with the idea that we would traverse at least four zones and 30.0 was selected by apriori knowledge of the range of $f(x, y)$. We knew that the range of values for $f(x, y)$ was roughly 350 and we wanted to avoid the flat regions of the solution space. For the problem in Section 2 we determine this value by observing the range of values generated during trial runs of the static tabu search. When a basin is detected the search restarts at a point generated from two features of the basin—the coordinates of the best point in the basin and the *distance* between the best and worst points in the basin. We call this latter value the diameter of the basin. By distance we mean the maximum of the difference between the x and y coordinates of the two points. Then we generate all eight neighbors of the best point at a distance of 6 times the diameter of the basin. We used a value of 6 so that we would be certain to evaluate points relatively far outside the basin. Among these eight points we select the non-tabu point with lowest function value. If all eight points are tabu we generate a new point randomly among the non-tabu zones.

## 1.4 Computational experiments

One of the goals of this section is to provide visual examples of the features of RTS. To do this we compare computational results from RTS with our static tabu search. Figure 1.2 summarizes the performance of the static tabu search on Levy No. 3. Figure 1.2 plots two data items, the $f(x, y)$ values on the z-axis and the $(x, y)$ values (or the projection of $f(x, y)$ onto the $xy$-plane. Each numbered region denotes a single pass of the search started at a randomly generated grid point. The checkerboard pattern in $xy$-plane shows that the search moves through a series of adjacent valleys (local minima). The unexamined white spaces of the

checkerboard most likely correspond to hills (local maximum) which were avoided by the search. Clearly the fixed memory size of 1500 and the aspiration criteria are able to force the search out of small local minima and in most cases avoid cycling. Five of the seven passes terminate due to the maximum iteration limit. Only passes one and two detect the presence of cycling associated with stalling in a deep valley before 5000 iterations are complete. Notice that three of the nine best solutions are uncovered by the search and that to get to these three solutions relatively deep valleys were explored and escaped. Besides these nine valleys with minima on the order of -176 there are many valleys with minima on the order of -145 and -116 in the landscape (see Figure 1.1). Although not illustrated here, we observed that both the static search and RTS have great difficulty escaping from the -145 and -116 valleys using only a tabu list and aspiration criteria. The lowest point in pass four in Figure 1.2 has an objective value on the order of -116. The total number of iterations (neighborhood searches) generated by the seven passes was 31,658.

Figures 1.3 summarize the performance of RTS on Levy No. 3. As in Figure 1.2 both the $f(x, y)$ values on the z-axis and the associated $(x, y)$ values are recorded. As before, each numbered region denotes a single pass of the search. Pass one is nearly the same as the one generated by the static tabu search. The difference is that in Figure 1.2 the search terminated due to a repeated solution (iteration 1725), but in Figure 1.3 RTS terminates due to the detection of a basin (iteration 2943). The variable length of the tabu list accounts for RTS not terminating at the same iteration (1725) as the static tabu search scheme. Clearly the valley for pass 1 in Figure 1.3 is not a basin (a collection of local minima). Instead of a basin the search detects circling around a single valley at nearly a constant height. Hence, since the change in the objective value is less than 30.0 (points at nearly the same depth of the valley) for more than 1024 iterations the basin escape mechanism of RTS is activated.

As a result, the starting point of the search for pass two is generated from the characteristics of the single valley (best point and diameter) as if it were a basin. Starting solutions for passes 3–6 are generated in the same manner. That is, passes 2,–3, 4, and 5 all terminate due to the detection of a basin. In each of these cases RTS detects what we initially had in mind for a basin—a collection of local minima (valleys). Each of the passes (2–5) contains a pair of small valleys which fulfill the basin criteria. Pass 6 terminates with a repeated zone and, consequently, pass 7 begins with a randomly generated starting point. Pass 7 finds a point with function value $-176$ at iteration 1145 and then spends the next 3200 iterations escaping from this valley and uncovering several small nearby valleys. These valleys trigger the basin escape mechanism and pass 7 concludes. As in the static search results three of the seven passes find $-176$ solutions. Two of these $-176$ solutions (passes 1 and 7) were uncovered by the static method as well. Due to the addition of the basin escape mechanism, which directs the search away from potentially unproductive areas of the search space, only two of the passes exhibit the checkerboard patterns observed in Figure 1.2. The total number of iterations for the seven passes is 17,299, almost half of the number required for the static search.

Figure 1.4 zooms in on pass 6 from the 2-dimensional diagram of the search given in Figure 1.3. The search starts at the point (7.3, -2.0) constructed from the best point in the basin detected in pass 5 and proceeds straight downhill to a local minimum with value -19.7. The search then spends roughly 600 iterations escaping from the local minimum. A total of five valleys are traversed with the third and fourth valleys having the best local minimum of the seven, $-141$ and $-176$ respectively. As can be seen from Figure 1.6 the search does not stop in the fifth valley but cycles back into the previously traversed second valley. Although not readily seen from the figure the search continues this cycle by repeating valley three as well before terminating by meeting the repeated solution (zone) condition. In all five valleys the search goes straight to the bottom (a greedy local search) and then spends several hundred iterations climbing back out again. The patterns exhibited shows the search backing out of the local minimum along contours of the function. This pattern is nearly identical for all passes of RTS.

Lastly we examine the length of the tabu list–tabusize–in RTS. Figure 1.5 catalogs the value of tabusize over the search history (iterations) for pass 6. The values of tabusize for pass 6 are plotted with triangles and dashes. As a reference point the fixed memory size of the static search is plotted as a horizontal line segment. As can be seen from Figure 1.4 pass six visits 5 valleys. The first valley traversed (iterations 1-600) is the block numbered with a one in Figure 1.4. The search next proceeds in a Northwest direction and traverses its second (iterations 601-1250) and third (iterations 1251-2429) valleys. The fourth valley is traversed on iterations 2430-3496, while the fifth valley is traversed by RTS on iterations 3497-4377. The best solution is found in valley four at iteration 2450 with a value of -176.540. As was described in Figure 1.4 RTS dives almost straight to the minimum point of each valley and then works its way back out again. What is the impact of allowing tabusize to change? The value of tabusize increases from 833 at iteration 2363 to 2489 at iteration 2369. The growth of tabusize in iterations 2363-2369 comes just as the search is trying to escape from valley three. Valley three is a deep local minima. Its lowest point has a value on the order of -141. Tabusize increases again at iteration 3087 while in valley four. Presumably this assists the search in escaping from the -176.540 local minima at iteration 3496. After iteration 4377 the search falls back into valley 2 from which it escapes for a second time on iteration 5055. Next the search falls into valley three again from which it is unable to escape prior to the search meeting its repeated solution condition. Thus, pass 6 stops at iteration 5917 with a tabusize of 6193. These latter increases of tabusize were not large enough to drive it away from the previously discovered valleys. Hence, the need for additional mechanisms to aid the search in avoiding long cycles is supported.

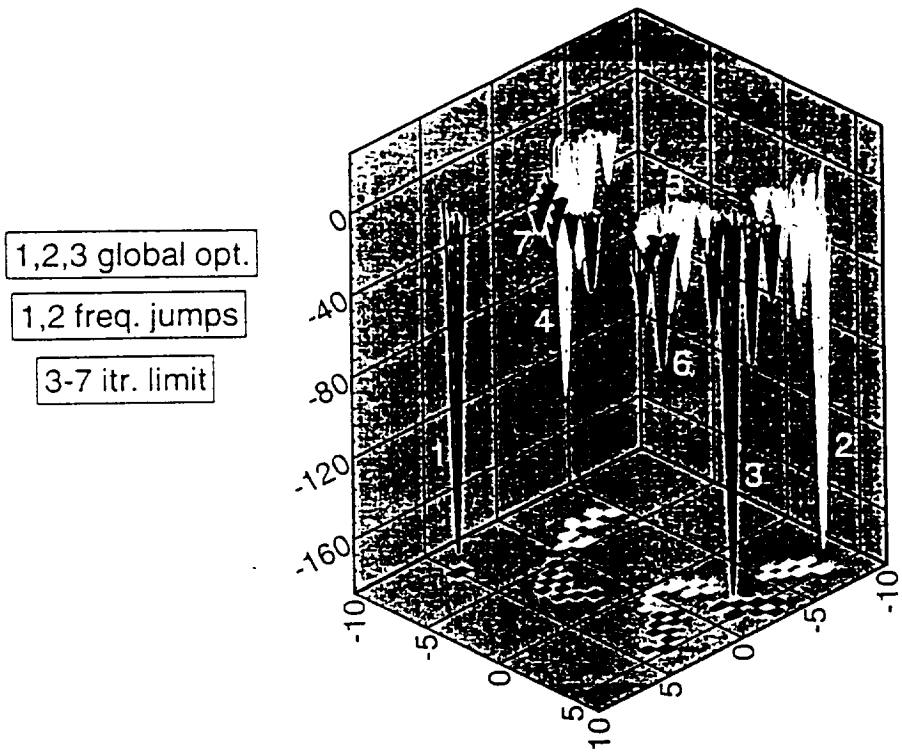



Figure 1.2 Static tabu search Levy No. 3–31,568 Iterations

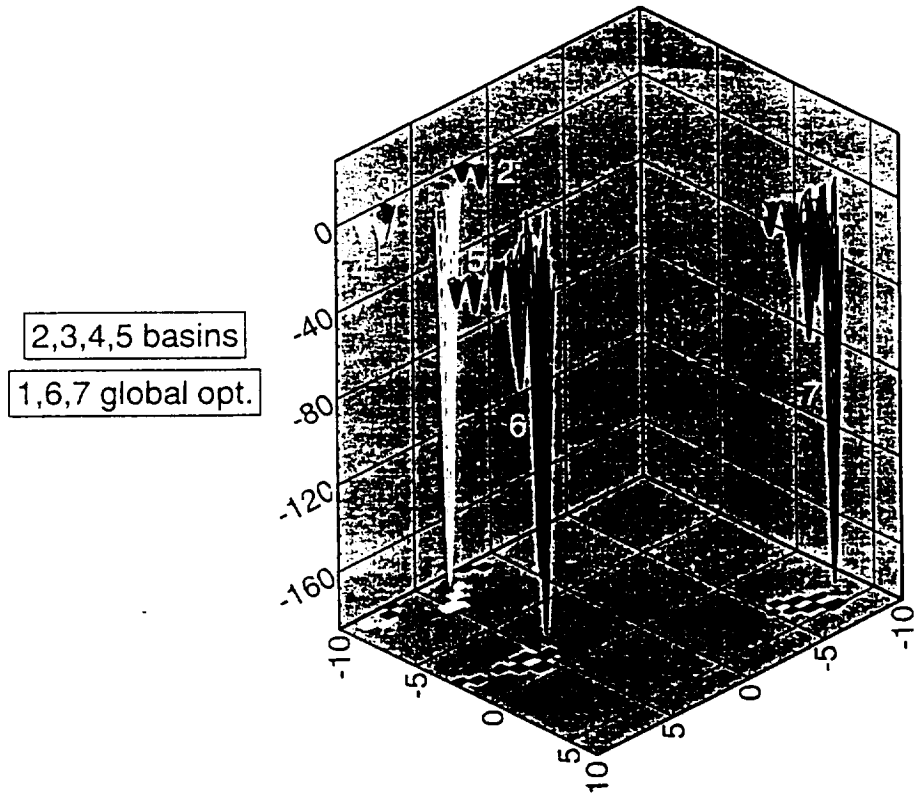



Figure 1.3 RTS for Levy No. 3–17,299 Iterations

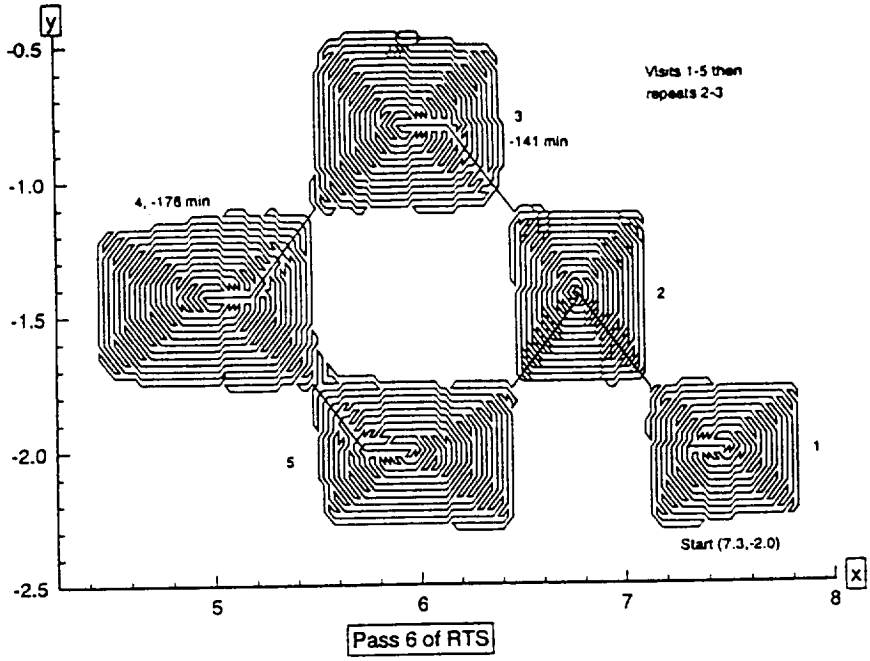



Figure 1.4 Summary of pass 6 of RTS in 2-dimensions

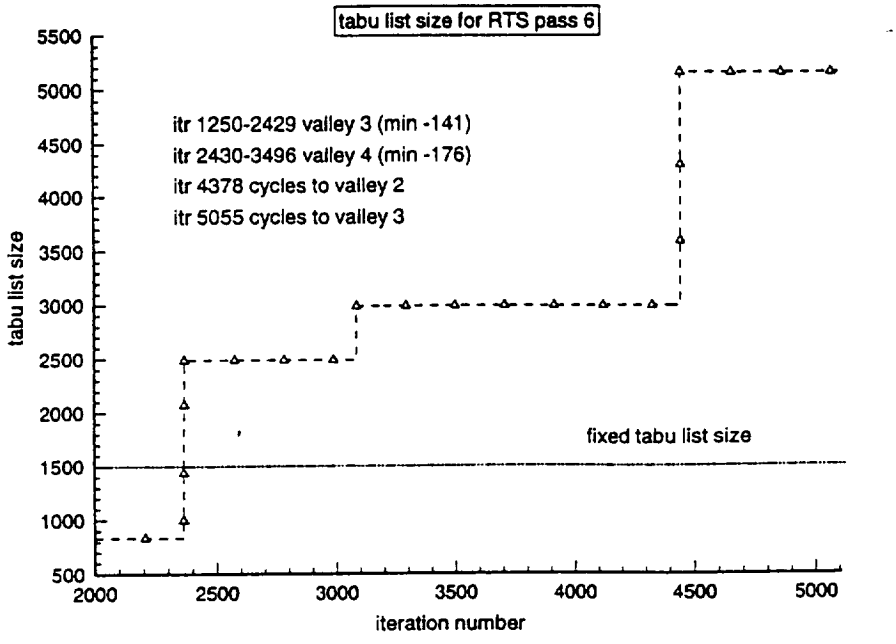



Figure 1.5 Size history of tabu list in RTS for pass 6

## 2. Sensor Selection for Active Noise Control.

Most aircarriers use turboprop aircraft to transport passengers between local or nearby airports and their hubs. Turboprop aircraft are preferred for short hops due to their superior fuel efficiency at low speeds. However, the decreased cost to the aircarrier comes at the cost of increased levels of cabin noise for the customer. Two broad classes of control methods exist to address the reduction of cabin noise—passive and active control. Passive control consists of introducing vibration absorption methods, such as the installation of liners in the shells of aircraft which attempt to diffuse the vibration and reduce the effects of noise. Passive methods, unfortunately, have proven ineffective in reducing cabin noise for turboprop aircraft (DeMeis, 1995) particularly in the low frequency range associated with the blade passage frequency of propellers. Active control methods attempt to counteract propeller noise by introducing a secondary (or multiple secondary) sound source which is of the same frequency and amplitude as the propeller noise, but is 180 degrees out of phase. Thus, in theory, the result is a complete silencing of the propeller noise source.

An early example of one-dimensional active noise control taken from Leug (1936) will serve to illustrate the salient features of active noise control systems. Consider a single frequency noise source, $S_1$ traveling down a duct as illustrated in Figure 2.1. The sound is detected with a microphone, $M$, and is passed through a controller, $V$, to a loudspeaker, $L$. The controller adjusts the sound emitted by the loudspeaker so that the wave produced, $S_2$, completely cancels out $S_1$. The controller finds the appropriate sound for the loudspeaker by means of a transfer function. A transfer function establishes the relationship between measurements at $M$ and the best input (phase and amplitude) at $L$. For more details on the history of active noise control the interested reader is referred to Fuller and von Flotow (1995).

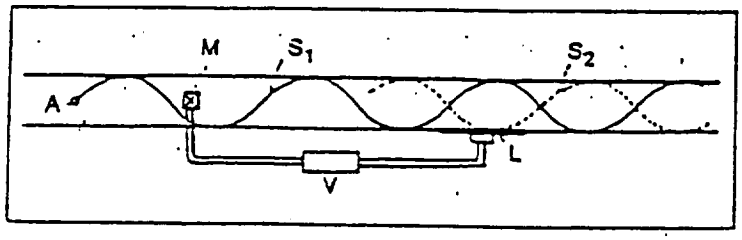



Figure 2.1. Simple Active Noise Control System

A variety of active structural acoustic controls systems have been proposed for aircraft (cf. Silcox et al. 1990). We consider an acoustic control model in which the control inputs, which are used to reduce interior noise, are applied directly to

a vibrating structural acoustic system (e.g. an aircraft fuselage). The feasibility of this approach has been demonstrated by Cabell et al. (1993) using measured data from the aft cabin of a Douglas DC-9 fuselage. We study a simpler problem. We model the aircraft fuselage with a cylinder and the turboprop propellers with a single monopole noise source located near the exterior of the cylinder. The monopole simulates the blade passage frequency of a turboprop propeller. To measure the performance of the actuators $p$ microphones (sensors) are mounted inside the cylinder. In this set of experiments the locations of the actuators are known and we wish to determine the best location for the $p$ sensors given $m >> p$ potential sensor locations. NASA engineers believe that the data collected and analyzed from the cylinder model will provide them with enough information to move to the next stage of development–an in-flight test of the the active control mechanism.

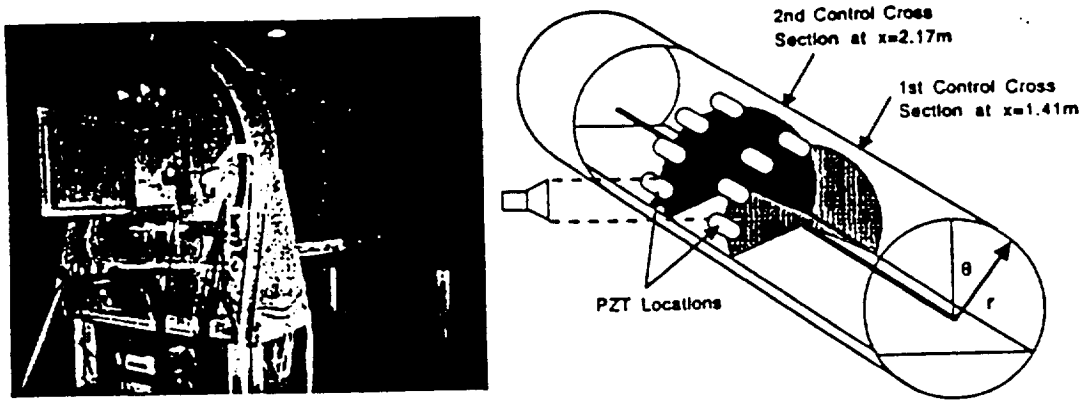
2.1 Description of Laboratory Model.

The dominant frequencies of the propeller generated noise are typically on the order of 100 Hz. to 400 Hz.. Moreover, propeller generated noise has several dominant frequencies and it is clear that a good set of actuators for noise control at one frequency is not necessarily good for noise control at other frequencies. Nevertheless, we decouple the noise control problem and consider only one frequency at a time. Eventually solutions that offer performance tradeoffs for a variety of frequencies must be addressed, but we do not do so here. Figure 2.2 shows the cylinder for the laboratory experiments performed at NASA-Langley Research Center. The cylinder is 3.6m long and 1.68m in diameter. (cf. Lyle and Silcox 1996 as well as Palumbo et al. 1996 for details). The monopole noise source is a 100 watt electrodynamic loudspeaker. An array of 8 piezoceramic actuators are installed on the interior of the cylinder.

In a mathematical model of the cylinder a steady-state response is assumed and both the cylinder shell response (vibrations) and interior acoustic response (noise) are expressed as a finite series of modes $(m, n)$, with the interior response represented as

$$p(x, r, \theta) = \sum_{n=0} \sum_{m=0} J_n(\alpha_m r) cos(\frac{m\pi x}{L})[P^c_{m,n} cos(n\theta) + P^s_{m,n} sin(n\theta)]$$

where $L$ is the length of the cylinder and $(x, r, \theta)$ denotes the location of the monopole noise source. The functions $J_n()$ involve inverse Fourier transforms with integrands in terms of complex Bessel functions. The modal coefficients $P^c_{m,n}$ and $P^s_{m,n}$ can be written as combinations of modal transfer functions available in the literature and cylinder shell displacement coefficients. Please see Silcox et al. (1990) for details.




- Outer shell 9 layers of filament wound graphite epoxy, 1.7mm

- Eight PZT actuators bonded to inner surface of trim panels

Figure 2.2 Cylinder model for active structural acoustic control

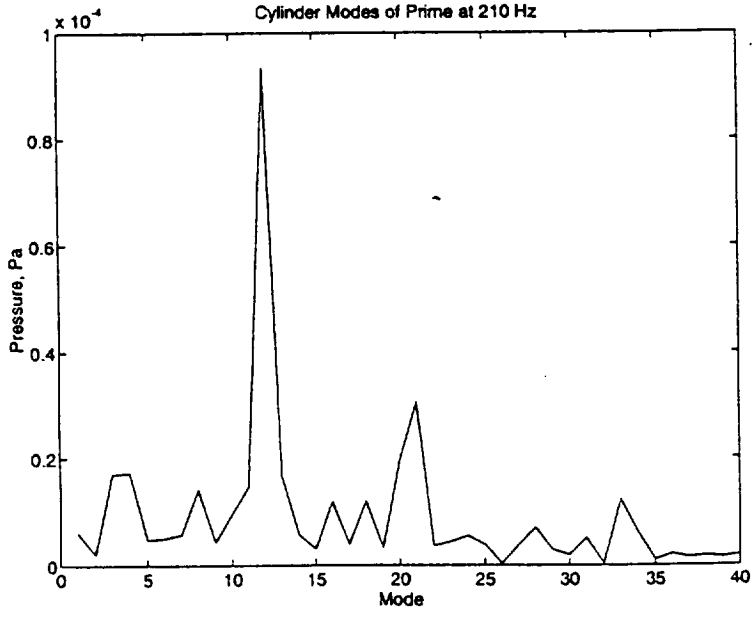



Figure 2.3. Acoustic modal response at 210 Hz.

In practice the interior noise cannot be measured everywhere and we approximate $p(x, r, \theta)$ by an array of sensor microphones. In the laboratory experiment a set of six microphones were swept on a boom throughout the interior of the cylinder (near the shell) and data was collected at 462 discrete points (see Figure 2.4). The collected data is used to construct transfer functions at each point. These transfer functions are then used to link the effects of each actuator location on the shell to the noise levels at frequencies of interest for each of the 462 data acquisition points. In our laboratory model there were only 8 actuators. The cost of the actuators and the difficulty in taking measurements limited the experiment to 8 actuators. An unlimited number of actuator locations could have been tested if the engineers had found a way to temporarily afix the actuator to the cylinder shell, take the measurements needed for the transfer functions, move the actuator to a new location, and repeat the process. The transfer functions generated at each of the 462 sensor data points and 8 actuators was used to construct 462 by 8 transfer matrices $H$ for three frequencies–210 Hz., 230 Hz. and 275 Hz. Respectively, these three frequencies were chosen by the acoustic engineers as representative of the three types of frequencies likely to be encountered in any model—strong acoustic modes with strong structural modes; weak acoustic modes with strong structural modes; and strong acoustic modes with weak structural modes. Figure 2.3 is a plot of the acoustic modes for the 210 Hz. case. Notice the strong peak at mode number 12.

The laboratory experiments was scaled down so that only 4 of the 8 actuator locations were selected. This was done by considering all 8 choose 4 combinations and evaluating each combinations noise reduction performance over the 462 data acquisition points. For larger problems the selection of the best set of actuator locations is a non-trivial discrete optimization (see Kincaid et al. 1996). The controller that adjusts the forces at the 4 actuators cannot process sensor inputs from 462 sensors. The controller board in the experiment was able to process inputs for 8 sensors. The optimization process for which RTS was implemented addressed this question. Which of the 462 data acquisition points are the *best* locations to place 8 sensors (microphones)? The performance measure for a given set of sensor locations is given by the transfer matrix $H$ for a given frequency. Each entry of $H$ provides the contribution of a unit amplitude input at each of the $k = 4$ actuators sites to interior noise reduction at each of the $m = 462$ data acquisition points for a given frequency. Thus, each row $i$ of $H$ records the contribution of a unit amplitude input at all 4 actuator sites in reducing the noise at sensor location $i$. That is, $H$ has one row for each of the 462 data acquisition points and one column for each of the 4 actuators. Given a particular subset $I$ of rows (or row indices) of $H$, the submatrix of $H$ induced by these rows, $H^I$, is the associated transfer matrix needed for computing the forces to drive the 4 actuators. Once a particular set $I$ of sensor sites have been selected a force (amplitude and phase) must be selected for each actuator so that the resultant force cancels out $\vec{p}$ as nearly as possible. In the acoustic literature the force vector $\vec{f}$ is chosen as the solution of the complex least squares problem

$$||H^I \vec{f} + \vec{p}^I||_2. \tag{1}$$

The solution to (1) is found by solving

$$H^{I*}H^{I}\vec{f} = -H^{I*}\vec{p}^{I} \tag{2}$$

for $\vec{f}$ ($*$ denotes the complex conjugate transpose). The range of (1) is $\mathcal{R}$, the real numbers. This value is the performance measure associated with each set of 8 rows of $H$ for a given frequency. Thus, our goal is to find a set $I$ of size 8 whose solution to (1) is as small as possible. Moreover, since the number of combinations is extremely large (462 choose 8), RTS is a candidate method for generating good solutions.

An initial set of sensor locations $I$ is selected via a constructive technique which takes into account the physical characteristics of the laboratory experiment. The 462 data acquisition points are spread evenly across 7 cross sections of the cylinder, with 66 data points in each cross section (see Figure 2.4). A starting solution is generated by selecting the 8 data acquisition points that have the largest primary noise reading subject to the restriction that at least one sensor location must be chosen from each of the 7 cross sections.

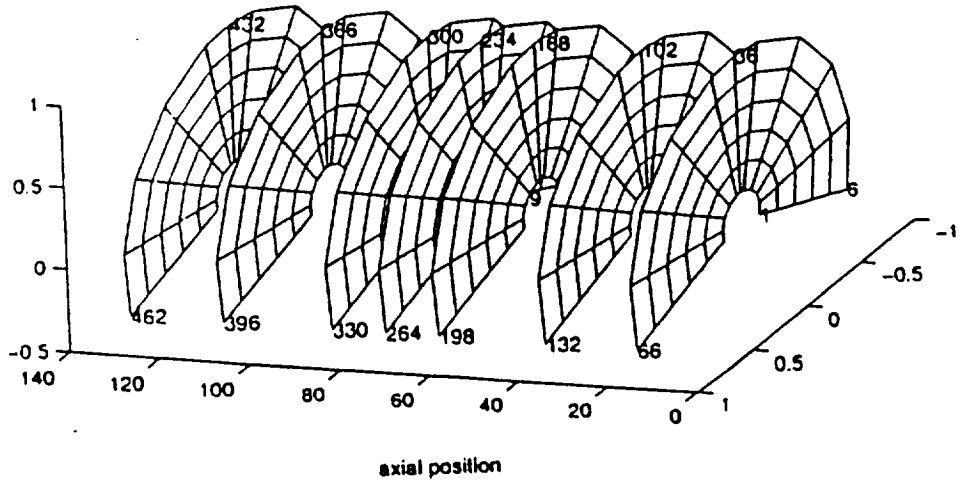



Figure 2.4. Data acquisition points: potential sensor locations

The performance of this initial set of rows $I$ (or any set of rows $I$) is measured on a decibel scale given by the expression below.

$$10 \cdot log_{10} \frac{||H^{I}\vec{f} + \vec{p}^{I}||_2}{||\vec{p}^{I}||_2} \tag{3}$$

The decibel value computed in (3) compares the interior noise norm with actuator controls in place with the norm of the uncontrolled interior noise. Thus, a negative

decibel (dB) level signifies a decrease in interior cylinder noise due to the control effects of the actuators.

There is one additional assumption that should be made clear. We have assumed that a linear control law appropriately models the active noise control of the cylinder. The main reason for this assumption is that the laboratory controller implements a linear control system to adjust the amplitudes and phases of the actuators. Our optimization model selects a best set of sensors by solving a complex least squares problem for the actuator forces and then evaluates the sensors performance using (3). In the laboratory experiment, the microphones are attached to the specified sensor locations and a controller selects the forces to drive the actuators using a time averaged gradient descent scheme which minimizes the mean of the squared error sensor signals. Hence, what we really want our optimization model to determine are sets of sensors that perform well with the linear controller in the lab. As a result we found that solutions with low dB (minimizing (3)) and small variances associated with the least squares solutions worked best.

2.2 Variance Penalty Term.

To quantify the responsiveness of a low dB solution to the variances computed for each actuator site in the complex least squares problem a penalty term was added to the performance measure. There are several likely candidates for the penalty term including a weighted sum of the variances, the maximum absolute variance, and the maximum relative variance. Each of these penalty terms was successful in steering the search away from low dB solutions with high variance components. We report on experiments that add a penalty term for any solution with a variance component larger than 5. An actuator whose force amplitude has a high variance may contain zero in a statistical confidence interval for the amplitudes true value, meaning that it is possible for the actuator to provide no contribution to noise cancellation. For the laboratory experiments it is important to select a microphone and actuator combination so that all the actuators have significant forces and are fairly reliable in their performance. If not, the laboratory controller has great difficulty in selecting actuator forces that are stable and the predicted performance of the actuator set (in terms of dB levels) is far from the actual performance in the lab.

When adding a penalty term for the variance to the dB performance measure it is natural to ask if their is any correlation between variance and dB levels. In partial answer to this question we tested 10,000 random samples of 8 sensor locations for a fixed choice of 4 actuator sites. There are about $10^{16}$ possible solutions (462 choose 8). Two histograms are presented below in Figure 2.5. The horizontal axis in the histogram on the left contains ranges of decibel reduction achieved when each of these solutions was generated. The horizontal axis for the graph on the right contains ranges of the maximum variance for the same solution samples. For both graphs, the vertical axis displays the number of solutions which fell into the ranges on the horizontal axis.

The left histogram shows that there are relatively few high quality noise reduction solutions and the right histogram shows that there are also relatively few

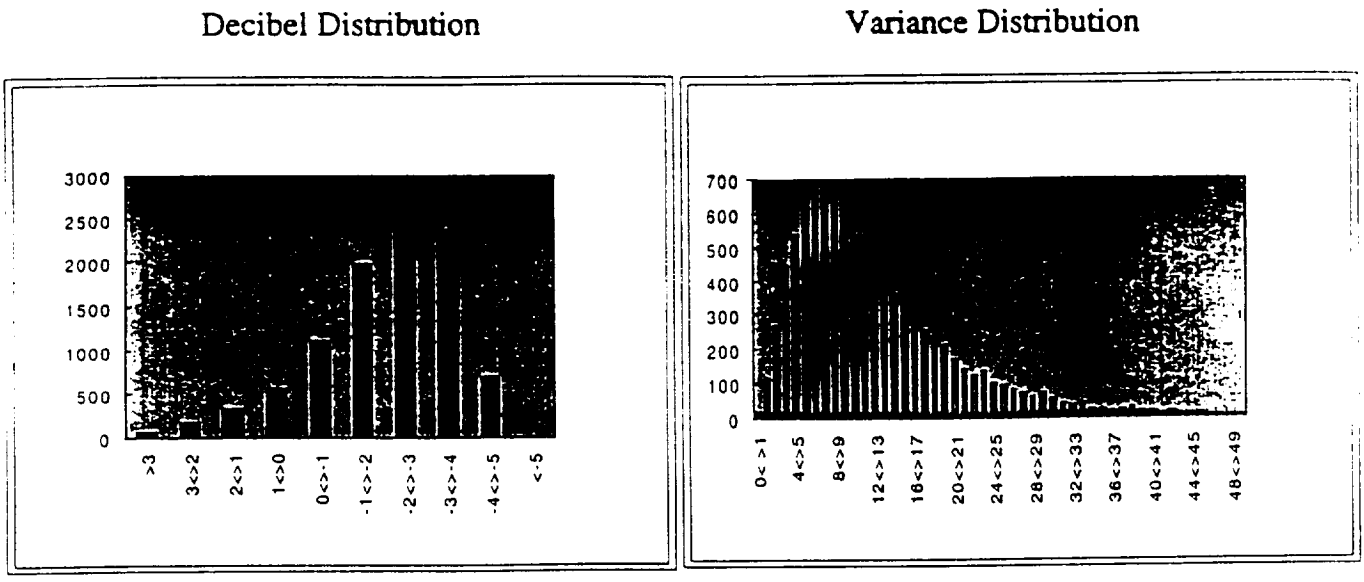



Figure 2.5. Histograms of Noise Reduction and Variance

solutions whose maximum variance is less than five. We also examined the histogram of the maximum variance for all solutions with a noise reduction level of -3 dB or less. We did this to determine whether there was any relationship between high quality solutions and low variance. The distribution of the variance for the high quality dB solutions was nearly identical to the histogram on the right in Figure 2.5. Hence, it appears that there is no connection between high quality solutions and low variance. We expect this low correlation to make the search for solutions that are good for both measures more difficult than simply minimizing the dB measure alone.

2.3 Computational Experiments.

The reactive features of the RTS procedure for the sensor selection problem are identical to the ones for Levy No. 3 in Section 1. However, the underlying tabu search features are different. Unlike the Levy No. 3 problem for which each solution had 8 neighbors, each solution to the sensor selection problem has $(462-8)*8 = 3532$ neighbors. Also, unlike Levy No. 3 a solution to the sensor problem is characterized by a vector of length 462 not of length 2. Hence, we make use of traditional attribute based tabu search rather than keep the actual solution. The stored attributes are which row (sensor) number is added to the solution and which row (sensor) number is deleted from the solution. We initialize the length of the tabu list at 2. Now, instead of recording each solution as a vector of length 462 we store the product of the sum of the row numbers that are in the current solution with the solution's objective function value. This number uniquely defines each solution.

When a solution is repeated 3 times we conclude that it is likely RTS is stuck in a cycle and the length of the tabu list is increased. That is, $tabusize_{new} = 1.5 * tabusize_{old}$. Conversely if $tabusize$ has not been increased for tabusize iterations then we decrease $tabusize$. That is, $tabusize_{new} = (1/1.5) * tabusize_{old}$. Lastly, a basin is detected in the difference between the best and worst objective function values is less than .008 for more than 10 iterations (each iteration searches the entire neighborhood). As in section 1 the number .008 associated with detecting a basin was found by trial runs of the search to get a rough idea of the terrain. Once RTS recognizes a basin, it escapes the area by generating a new starting solution. The new starting solution is generated by the same constructive procedure as the initial solution with the addition of penalties for sensor locations (rows of the matrix $H$) which have been examined often, thereby forcing the search to select relatively infrequently examined solutions.

Unlike the example in Section 1 we cannot visualize the performance of many of the RTS features. However, it easy to show the effect of dynamical changing the length of the tabu list as is done in Figure 2.6. The horizontal axes represent the iteration number, and the vertical axes are the dB values of the best solution at that particular iteration. The left graph is a static tabu search with a fixed length tabu list ($tabusize = 2$) and the right graph is RTS. Both begin with a tabu list length of 2, but while the static tabu search cycles after 12 or so iterations, and continues to do so for the duration of the run (50 iterations), RTS recognizes that it is cycling and lengthens the tabu list thereby allowing it to search additional areas of the solution space. In doing so, RTS ultimately finds a better solution than the static tabu search. Of course it is possible to adjust the length of the tabu list in the static tabu search and avoid the cycle. The point is that RTS does this without any user intervention.

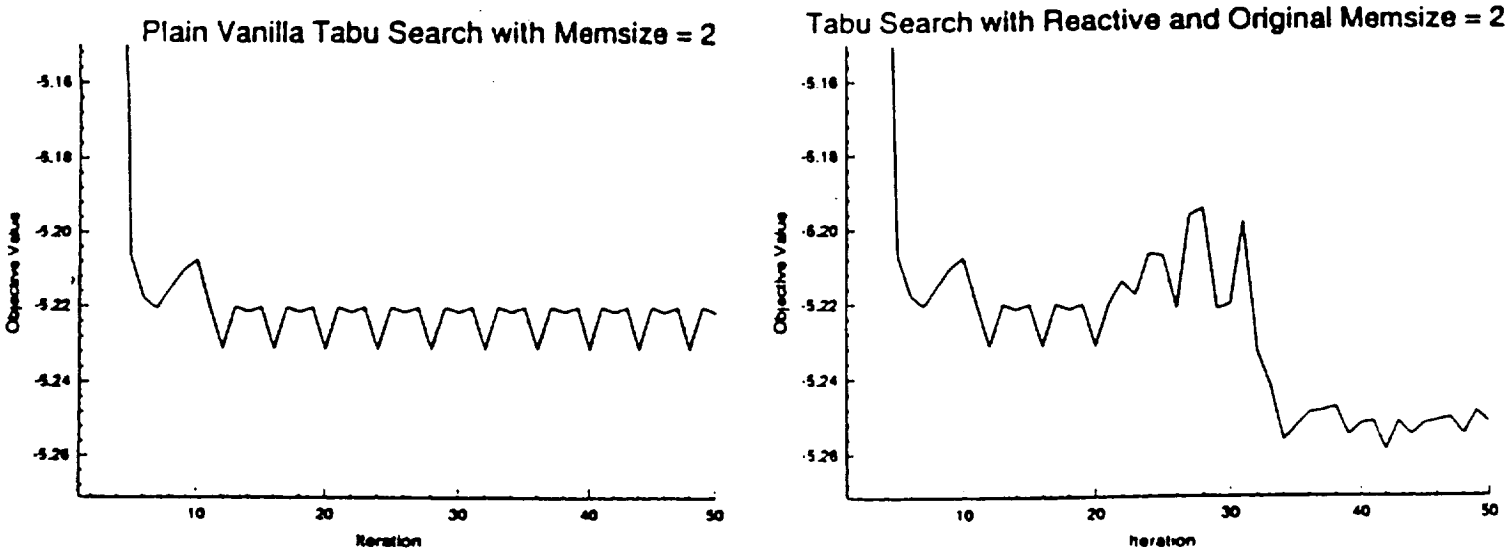



Figure 2.6 Fixed memory tabu search vs. reactive memory size in RTS.

The adaptive memory length aspect of RTS enables RTS to escape many of the deep valleys associated with the dB and variance penalty performance measure. The dynamic length of the tabu list in RTS does not solve all the problems which arise from a difficult terrain in the solution space, however. As we saw for Levy No. 3, sometimes there exist large collections of local optima within the solution space which are so large that it may be impossible for the search to escape using basic tabu search mechanisms and a dynamic tabu list length. As we saw with Levy No. 3 in section 1, a basin is a wide relatively unproductive collection of local minima and so a basin escape mechanism is implemented. Figure 2.7 records the results of RTS for the 210 Hz. case. Here, three basins were detected and each restart led to successively better solutions (the variance penalty value is not included in the y-axis values) with the origin corresponding to the lowest obtainable dB level of $-5.27$. That is, with 4 actuators and 462 sensors the dB level is $-5.27$. The x-axis denotes the iteration number on each pass. Although the results of Figure 2.7 illustrate the usefulness of the basin escape mechanism for the sensor location problem, we note that the level of improvement in the dB levels from one restart to the next is insignificant with respect to the accuracy of the model. That is, our experience with the laboratory controller (Table 2.1) demonstrates that we cannot predict that a $-5.27$ dB solution will be better than a $-5.23$ dB solution. In an attempt to make the problem more difficult we tested the model with all 8 actuators in place and the selection of 16 sensor locations. The results for the 210 Hz. case were nearly identical to those in Figure 2.7.

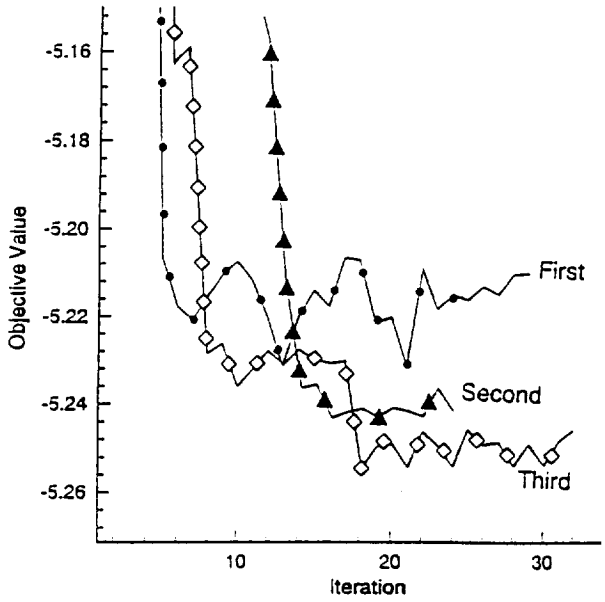



Figure 2.7. Three passes of RTS for 210 Hz (4 actuators/8 sensors)

The applicability of tabu search to the sensor selection problem was validated by a laboratory experiment. The question to be resolved was how well did the tabu search generated solutions compare to other standard engineering techniques for

selecting sensor locations. The competitor was an acoustical technique (cf. Palumbo et al. 1996) that involves examining visual and numerical displays of the modal decomposition of the interior pressure field $\vec{p}$ and the individual actuator responses. Dominant modes in the interior pressure field are matched with dominant modes in the actuator responses. Table 2.1 catalogs the performance of tabu search versus a standard acoustical technique in the selection of the best set of 8 sensors. The columns marked TS compare the predicted noise reduction of the solution found by tabu search versus the actual noise reduction measured in the laboratory for that solution. Recall that in the laboratory model microphones are attached to the selected sensor locations and then a linear controller is used to determine the forces applied to the 4 actuators. The column marked modal analysis show noise reduction levels achieved in laboratory tests run for sensors selected based on a modal analysis of noise field.

| **Freq.** | **TS** | **TS** | **Modal Analysis** |
|---|---|---|---|
| Hz. | Predicted | Measured | Measured |
| 210 | -5.3 dB | -4.4 dB | -2.1 dB |
| 230 | -3.8 dB | -2.5 dB | -1.2 dB |
| 275 | -5.7 dB | -3.9 dB | -2.7 dB |

Table 2.1. Experimental Results with Actuator/Sensor Location

There are several observations to be made from Table 2.1. First, we see that the measured performance in the lab for the tabu search sensor location solution dominates the modal analysis solutions. The significance of this result for us is that it convinced the acoustic engineers that optimization modeling for this problem can yield much better results than engineering analysis alone. The tabu search solutions are roughly 1.2 better than the modal analysis solutions. (Remember that the dB scale is logarithmic.) As a result of this dominance, optimization modeling has become an integrable part of the active noise control problem at NASA-Langley Research Center. Secondly, as expected, the 210 Hz. and 275 Hz. problems are easier to control due to the strong acoustical modes. The 230 Hz. case is difficult to control due to the weak acoustical mode. Thirdly, the forces selected by the linear controller resulted in less noise reduction than the optimization model predicted. There are two reasons for this. The controller uses a gradient search procedure to select the actuator forces. It does not solve the complex least squares problem as we did within tabu search. However, the second reason is just as important. The robustness of the data needed to construct the transfer matrices, $H$, for each frequency is questionable. Data taken on different days typically result in slightly different transfer matrices. Hence, since the data collection and the laboratory optimization tests were run on different days the transfer functions that the laboratory controller is minimizing are not necessarily the same as the ones in

the tabu search model.

## 3. Discussion

The computational experiments on a discretized version Levy No. 3 in section 1 illustrated the potential benefits of RTS over a static tabu search. The value of the fixed memory size in the static tabu search was a sensitive parameter whose value of 1500 was arrived upon after many experiments. In the dynamic growth and decay of this parameter for RTS one still must select the rates of growth and decay. We did no experimentation with these parameters. We used values suggested by Battiti and Tecchiolli (1994a) and made only minor adjustments. This suggest that the rates of growth and decay are not sensitive parameters. We also show the usefulness of the basin escape mechanism. In particular, three $-176$ solutions were uncovered by RTS in 17,299 iterations as were found by the static tabu search in 31,658 iterations–a significant savings. Here a parameter for the associated basin depth must be determined. We did not perform any tests to determine the sensitivity of the search to this value. However, the known structure of the terrain would lead one to believe a relatively wide range of values would work.

For the noise control model in Section 2, we examined only a few significant discrete frequencies of the noise emitted by a propeller. Each of these important frequencies was examined individually, resulting in different optimal or near optimal combinations of microphones and actuators. Current research efforts (cf. Palumbo and Padula 1997) show how to combine solutions from each of the important frequencies and/or how to generate solutions that are valid for several frequencies. An important consequence of the work presented in Section 2 is that it convinced acoustic engineers that an optimization model was a useful approach for selecting sensor locations. In fact, experiments are planned for the summer of 1997 to allow the placement of a larger set of actuators chosen from a potential set of about 100 actuators. This increase in the number of potential actuators has been shown (Kincaid et al. 1996) to complicate the solution space terrain and necessitate the addition of more advance tabu search mechanisms (RTS, diversification, intensification, candidate list strategies etc.) to uncover superior solutions. For example, if 16 actuators are to be selected from the 100 sites then approximately 32 sensors will be chosen from the 462 data acquisition points. The choice of the numbers 16 and 32 is driven by hardware controller requirements and is not a decision variable for the optimization model at this time.

The RTS experiments in Section 2 used the same parameters as in Section 1. The dynamic growth and decay the length of the tabu list for RTS used exactly the same rates of growth and decay as in Section 1. We did no experimentation with these parameters. Determining the value of the parameter associated with the detection of a basin was based on observations of the range of values on trial runs. Hence, as in Section 1 some knowledge of the terrain is needed to select this parameter's value. We also found that the gain in the quality of the solutions found by RTS were not significant. However, this may change when the larger laboratory experiment is run (select 16 actuators from 100 actuator sites and 32 sensors from

462 sensor sites) or when data taken from planned aircraft flight tests are used. It is not known if the increase in the size of $H$ from 8 by 4 to 32 by 16 will significantly complicate the search terrain for low dB and low variance solutions, but we suspect that it will.

## References


Battiti, R. and G.Tecchiolli (1992). "Parallel based search for combinatorial optimization: Genetic algorithms and tabu search," *Microprocessor and Microsystems*, 16:351-367.

Battiti, R. and G. Tecchiolli (1994a) "The Reactive Tabu Search," *ORSA Journal on Computing*, 6, 126-140.

Battiti, R. P. Lee, A. Sartori, and G. Tecchiolli (1994b). "Combinatorial optimization for neural nets: RTS algorithm and silicon," Technical Report no. 9406-04, IRST, Trento, IT.

Battiti, R. and G. Tecchiolli (1994c). "Simulated annealing and tabu search in the long run: a comparison on QAP tasks," *Computer and Mathematics with Applications*, 28(6):1-8.

Battiti, R. and G. Tecchiolli (1995a). "Training neural nets with the reactive tabu search," *IEEE Transactions on Neural Networks*, 6(5):1185-1200

Battiti, R. and G. Tecchiolli (1995b). "Local Search with Memory: Benchmarking RTS," *Operations Research Spektrum*, 17(2/3):67-86.

Cabell, R.H. and H.C. Lester (1993) "Investigation of Force Distributions for Interior Noise Control Using a Neural Network," *Proceedings of the Second Conference on Recent Advances in Active Control of Sound and Vibration*, Blacksburg, VA, April, 55-69.

Fuller, C.R and A.H. von Flotow (1995) "Active Control of Sound and Vibration," *IEEE Control Systems*, December, 9-19.

Glover, F. (1996) "Tabu Search and Adaptive Memory Programming–Advances, Applications, and Challenges," to appear in Interfaces in Computer Science and Operations Research, Barr, Helgason and Kennington (eds.) Kluwer Academic Publishers.

Hansen, E. *Global Optimization Using Interval Analysis*, Marcel Dekker, New York (1992).

Jansson, C. and O. Knuppel, "Numerical Results for a Self-Validating Global Optimization Method," working paper Technische Universitat Hamburg-Harburg, Technische Informatik III, D-21071 Hamburg, Germany (1994).

Kincaid, R.K., K.E. Laba, and S.L. Padula (1996) "Actuator Location Problems in Active Structural Acoustic Control," submitted for publication.

Leug, P., (1936) "Process of Silencing Sound Oscillations," U.S. Patent No. 2,043,416.

Levy, A. V., A. Montalvo, S. Gomez, and A. Calderon, *Topics in Global Optimization*, Lecture Notes in Mathematics No. 909, Springer-Verlag, New York (1981).

Lyle, K.H. and R.J. Silcox, (1995) "A Study of Active Trim Panels for Noise Re-

duction in an Aircraft Fuselage," paper no. 951179 in SAE General, Corporate and Regional Aviation Meeting and Exposition, May.

Palumbo, D., S.L. Padula, K.H. Lyle, J.H. Cline, and R.H. Cabell (1996) "Performance of Optimized Actuator and Sensor Arrays in an Active Noise Control System," presented at 2nd AIAA/CEAS Conference, AIAA 96-1724.

Palumbo, D. and S.L. Padula (1997) "Optimizing an Actuator Array for the Control of Multi-Frequency Noise in Aircraft Interiors," presented at 3rd AIAA/CEAS Aeroacoustics Conference, May (AIAA 97-1615).

Silcox, R.J., C.R. Fuller, and H.C. Lester, (1990) "Mechanisms of Active Control in Cylindrical Fuselage Structures," *AIAA Journal*, Vol. 28, August, 1397-1404.