# Chapter 3

# STOCHASTIC EFFECTS IN FINITE GENETIC MODELS

## 3.1 Introduction

In this chapter we will explore the characteristics of finite genetic adaptive systems, that is, genetic plans which have limited memory and time to adapt to the problem at hand. As one might expect, the behavior of such systems can vary considerably from the norm predicted by mathematical analysis involving expected values, the law of large numbers, and limit theorems. The motivation for analyzing finite models, of course, is that they correspond to the observed behavior in any practical application of genetic adaptive systems. We will pursue this analysis by considering in more detail the characteristics of plan R1 introduced in the preceding chapter.

## 3.2 The Problem of Premature Convergence

We begin by analyzing the behavior of plan R1 on test function F1 (see appendix A). Here the problem consists of finding the minimum point on the three dimensional parabolic surface given by

$$F1(X) = \sum_{1}^{3} x_1^2, \ x_1 \leq 5.12, \ \Delta x_1 = .01$$

As illustrated in appendix C, plan R1 generates an exponential decrease in both $f(t)$ and $f^*(t)$ over the interval $1 \leq t \leq 10{,}000$. However, since R1 is a stochastic process, these curves represent the performance of R1 averaged over the number of independent runs. Table 3.1 depicts the behavior of R1 for a particular run on test function F1. Notice that there is little or no improvement in $f(t)$ and $f^*(t)$ from $t=3000$ on, even though $f^*(t)$ is still greater than the minimum of zero at the origin. This behavior is typical of plan R1. After an initial reduction in $f(t)$ and $f^*(t)$, a threshold seems to be crossed after which little or no improvement is generated. If we look more closely at the population $A(t)$ maintained by R1, the reason for this lack of improvement becomes clear: each individual in A(3000) is very nearly alike. Recall that plan R1 uses a binary genetic representation for points in the solution space A. That is, each gene position can take on only the values 0 or 1, and for this problem, $|A| = (10^3)^3 = 10^9$ requiring $\ell = 30$ gene positions. If we consider $A(t)$ a reservoir of gene values (alleles), the "lost" column in table 3.1 illustrates that in A(3000) 22 of the 30 gene positions have no instances of one of the two possible alleles. That is, plan R1 has converged to a particular allele in all but 8 positions and hence reduced the search space for crossover to $2^8 = 256$ points. Moreover, if we say that plan R1 has <u>effectively converged</u> to a particular allele whenever an allele is

| trials t | generations | f(t) | $f^*(t)$ | lost | converged |
|---|---|---|---|---|---|
| 50 | 1 | 25.62 | 2.016 | 0 | 0 |
| 250 | 5 | 11.20 | 0.409 | 1 | 1 |
| 450 | 10 | 8.11 | 0.187 | 2 | 2 |
| 1000 | 25 | 5.48 | .151 | 6 | 11 |
| 2000 | 60 | 2.19 | .109 | 14 | 18 |
| 3000 | 100 | 1.42 | .018 | 22 | 25 |
| 6000 | 335 | 1.05 | .018 | 24 | 26 |
| 10000 | 560 | .82 | .018 | 25 | 27 |

Table 3.1: The data associated with a single run of R1 on F1.

found in more than 95% of the population, then the "converged" column in table 3.1 illustrates that by t = 3000, R1 has effectively converged in 25 of the 30 positions.

This reduction in the search space A is precisely the behavior discussed in chapter 2. Unfortunately, however, in this case the optimum for F1 is not contained in the reduced subspace. Nor is it very likely that plan R1 will find the optimum for $t > 3000$. To see this recall that crossover can generate a point in A for trial only if all the alleles for that point are present in the population. Hence, crossover effectively searches only the reduced subspace of 256 points. Moreover, because of the similarity of individuals in A(3000), the results of many crossovers will be to produce an offspring identical with one of the parents, providing no new points for trial. Comparing the "trials" column and the "generation" column in table 3.1 illustrates this reduced effectiveness of R1 as alleles are lost from the population. Initially, nearly 50 new trials are generated per generation by crossover and mutation. However, from generation 60 on ($t > 3000$), there are fewer than 15 new trials per generation.

Restating these observations in terms of the hyperplane analysis of chapter 2 yields further insight into the problem. For 25 of the 30 first-order hyperplane partitions of A, plan R1 has chosen to allocate almost all of the trials from t = 3000 on to one of the two competing partition elements. However, if we consider the symmetry of test function F1 on A, it should be clear that every first-order partition of A presents plan R1 with a 2-armed bandit problem in which the machines have equal payoffs. In other words, no particular allele has an advantage over its competitor; yet in 25 of 30 positions, one allele seems to have almost completely dominated, effecting a dramatic reduction in the search space.

Immediately one thinks of increasing the mutation rate as a simple direct way of maintaining variability in the population. But we must be careful at this point of applying a cure to a symptom rather than the problem. Certainly increasing mutation will increase the variation in the population maintained by R1 on F1. But recall that on F1 no particular allele has an advantage over its competitor. For the other functions in E there clearly are alleles which yield much higher performance than their competitors. Increasing mutation in these cases will retard the dominance of the better performing alleles and slow the adaptive response. What we attempt to understand in this chapter is why R1 has such a high rate of allele loss on F1. The hope is that understanding this problem will provide insight into improved performance on E.

## 3.3 Genetic Drift

The phenomenon of genetic drift is a well-studied problem in population genetics. Since it is an artifact of the application of random selection processes to finite populations, it has considerable bearing on the finite genetic models under study in this chapter. Genetic drift can be illustrated by the following simple stochastic model. Suppose we have a population A(t) of N individuals and we generate A(t+1) by making N uniformly random selections from A(t) with replacement and apply no genetic operators. Again we focus our attention on the alleles of a particular gene and observe the number of instances of these alleles in the population. If we assume a binary genetic representation and a uniformly random initial population A(0), then the expected number of 0-alleles $R_i(t)$ for gene i is N/2. However, as t increases, the variance of $R_i(t)$ also increases to the extent that wide deviations from the norm are quite likely.

To see this more clearly, we can represent the above model as a Markov process in which the states are simply the N+1 possible values of $R_i(t)$. The transition probability $P_{jk}$ is simply the probability of k successes in N Bernoulli trials with a probability of success on each trial of j/N. That is,

$$P_{jk} = \binom{N}{k}\left(\frac{j}{N}\right)^k\left(1-\frac{j}{N}\right)^{N-k}$$

The initial state probabilities $P_k$ are simply

$$P_k = P_{\frac{N}{2},k} = \binom{N}{k}\left(\frac{1}{2}\right)^k\left(\frac{1}{2}\right)^{N-k} = \binom{N}{k}\left(\frac{1}{2}\right)^N$$

With no genetic operators defined, it should be clear that states $R_1(t) = 0$ and $R_1(t) = N$ are absorbing states. Since the other N-1 states are all transient, the probability of being in either of the absorbing states increases over time and in fact approaches 1. Of even more interest is the expected number of generations to <u>first entry</u> into a particular state. We are interested in those states in which one of the alleles under observation has managed to dominate a certain percentage of the population. To illustrate the effects of genetic drift we will focus our attention on 4 states: 70, 80, 90, and 100% dominance.

The expected number of generations $f_{jk}$ to first entry into state k from state j is given by:

$$f_{jk} = \sum_{n=0}^{\infty} n * f_{jk}^{n}$$

where $f_{jk}^{n}$ is the probability of first entry to k from j in exactly n steps. Unfortunately, the computation of these expected values is difficult since the terms $f_{jk}^{n}$ are computed recursively as

$$P_{jk}^{n} = \sum_{i=1}^{n} f_{jk}^{i} * P_{kk}^{n-i}$$

in terms of the extended transition probabilities $P_{jk}^{n}$ which are themselves computed by raising the transition matrix

$$P = \left[P_{ij}\right]$$

to the n$^{th}$ power.

However, for our purposes, we can estimate the expected values by simulation, the results of which are illustrated in figure 3.1. As might be expected, the number of generations to reach a particular state of dominance is a linear function of population size. The slopes associated with the (70, 80,90, and 100%) states are roughly 1/5, 2/5, 4/5, and 8/5 allowing for a predicted rate of dominance. Figure 3.2 illustrates more clearly the role of population size in increasing the expected number of generations to first entry into one of the four states. Moreover, it illustrates that the effects of genetic drift cannot be ignored even in a population of size 100 if the number of generations exceeds 50.

To reduce these stochastic effects over the interval of adaptation, we can of course increase the population size sufficiently to minimize genetic drift, but we do so at the expense of maintaining a larger population and in general a slower adaptive response. A second alternative which immediately comes to mind is to add a mutation operator which would counteract the allele loss

FIG 3.1: GENETIC DRIFT VARYING POPULATION SIZE

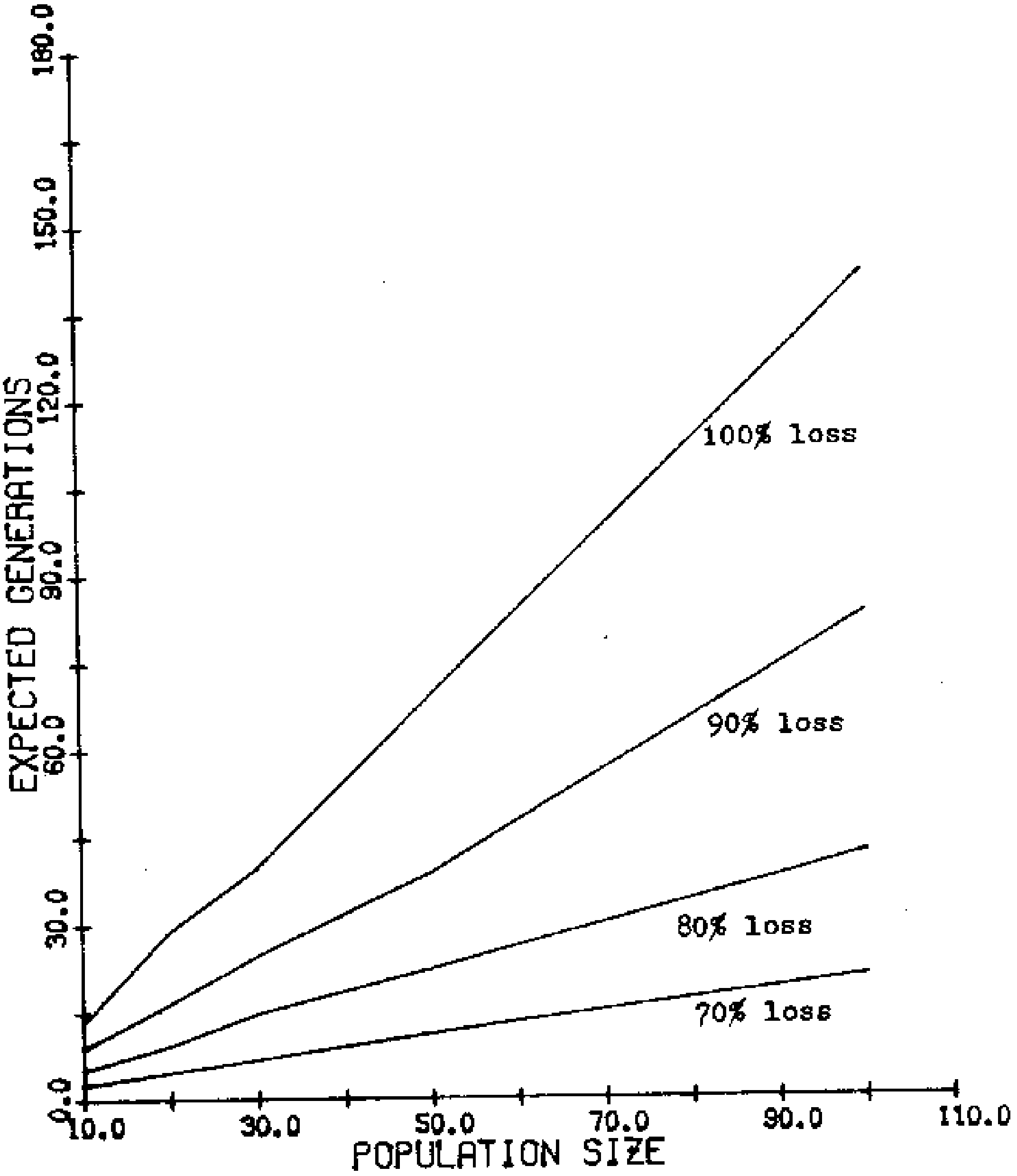


Figure 3.1: The rate of allele loss due to genetic drift as a function of population size.

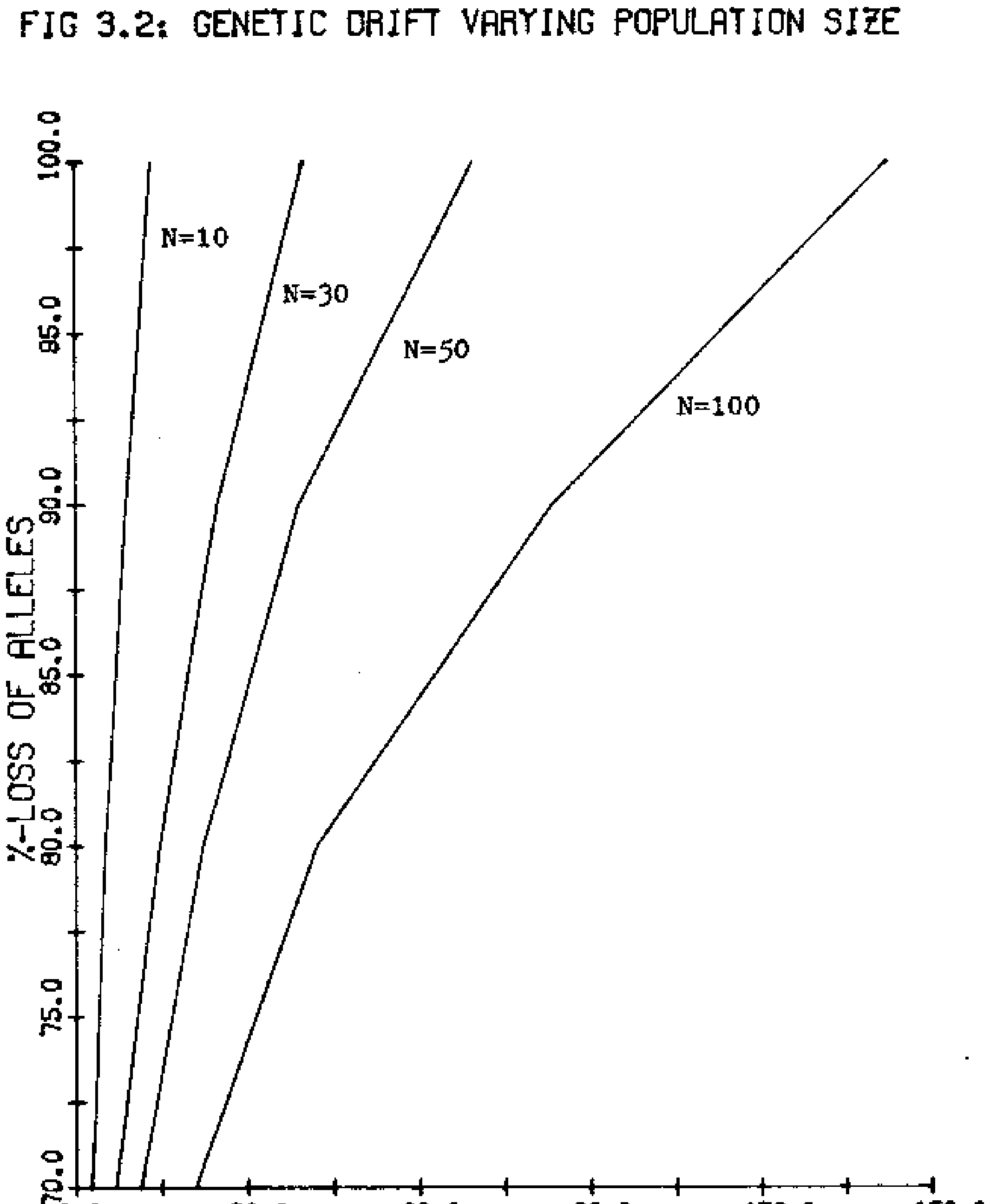


Figure 3.2: The rate of allele loss due to genetic drift as a function of population size.

due to genetic drift and allow smaller population sizes. To evaluate this alternative, a mutation operator can be easily added to the Markov process discussed above. While the addition of mutation complicates the expected value computations even more, it is intuitively clear that the states 0 and N are no longer absorbing states. Figures 3.3 and 3.4 illustrate the effects of several mutation rates on the simulated Markov process with populations of size 50 and 100. As might be expected, one mutation per generation is sufficient to increase the expected first-entry times to the 100%-loss state beyond the bounds of a practical adaptive interval. However, the effects on the first entry times to the other states are much less pronounced. The expected first entry to a 90%-loss state with a population of size 50 is still less than 60 generations.

## 3.4 The Effects of Population Size on R1

The analysis of the preceding sections has yielded considerable insight into the behavior of plan R1. As we have seen, the loss of alleles from A(t) corresponds to a dramatic decrease in the space being searched by R1. If this reduced space does not contain the optimum, we have seen that R1 will very likely remain on a non-optimal plateau with mutation providing only a low-probability chance of escape. Since this is the case, it is critical that alleles are lost only if their com-

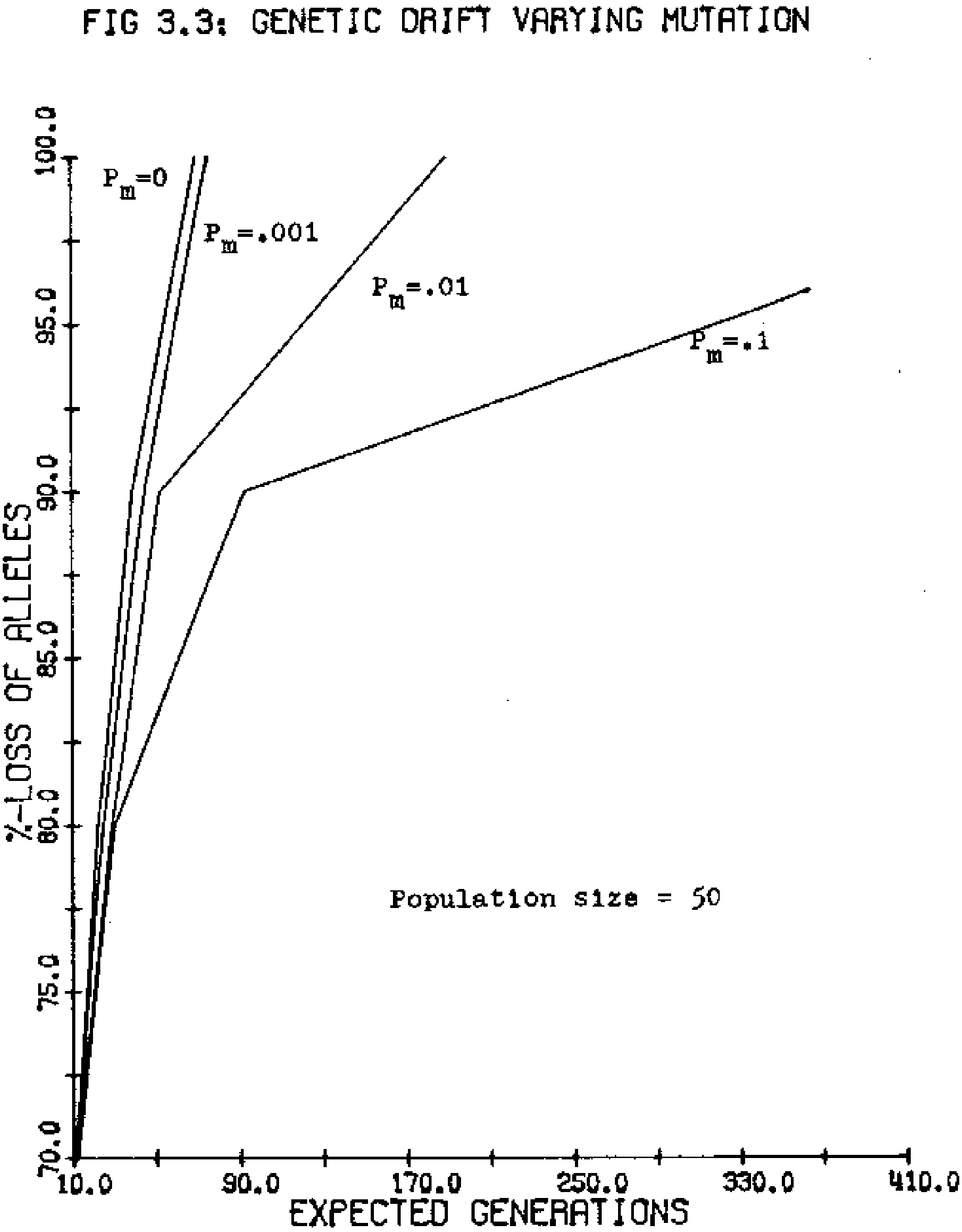


Figure 3.3: The rate of allele loss due to genetic drift as a function of the mutation rate.

FIG 3.4: GENETIC DRIFT VARYING MUTATION

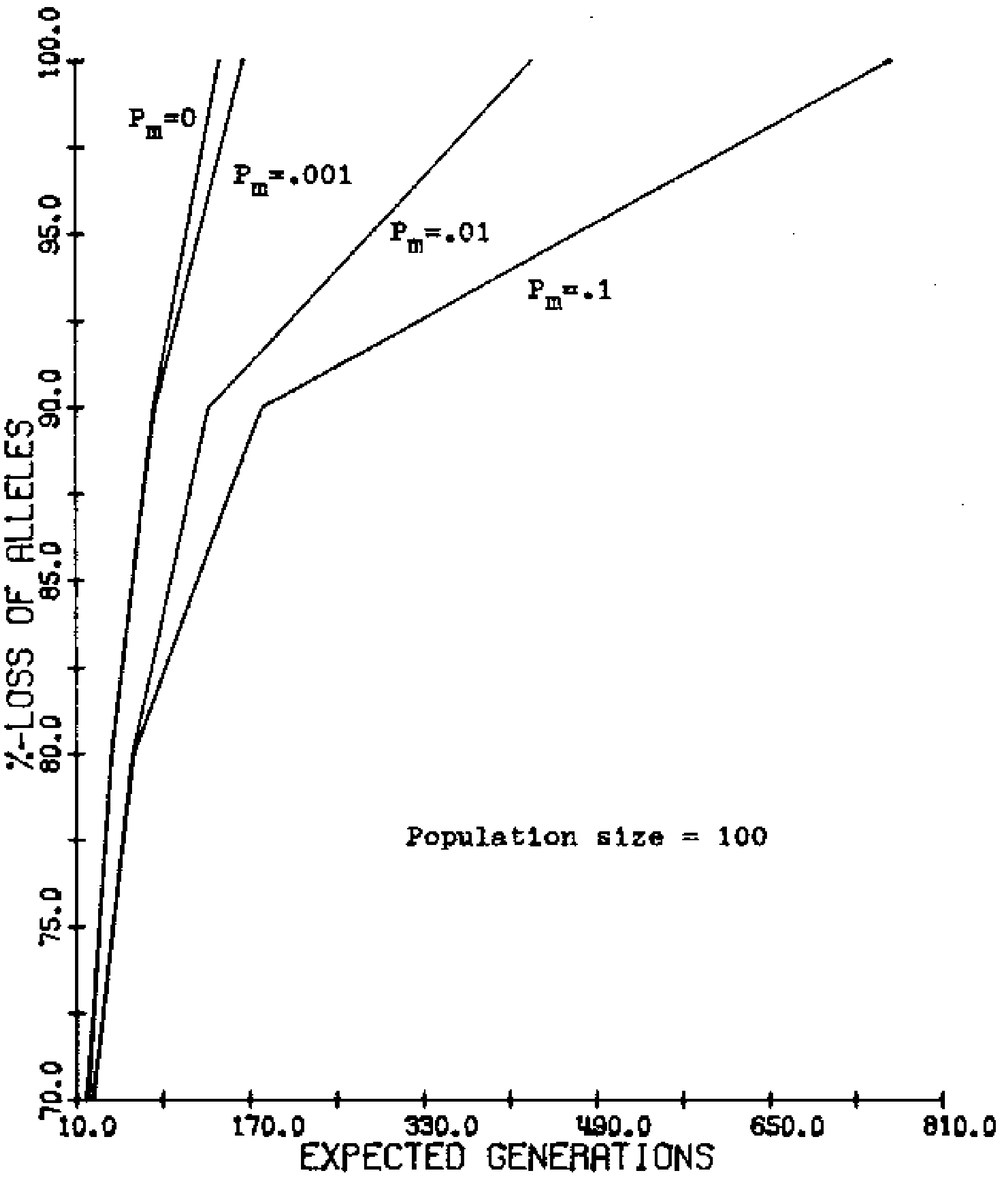


Figure 3.4: The rate of allele loss due to genetic drift as a function of the mutation rate.

petitors are in fact better. However, on test function F1, alleles are lost even when there is no selection differential and R1 converges to a non-optimal plateau. The preceding section suggests that this may be due in part to stochastic effects and suggests two approaches for alleviating the problem: changing the population size and the mutation rate of plan R1. In this section we explore the effects of population size on the behavior of R1.

Recall from appendix C that plan R1 maintained a population of 50 individuals and a mutation rate of .001. Note further from table 3.1 that 100 generations had elapsed by the time R1 generated A(3000). Referring back to figure 3.3, we see that for a population size of 50 and a mutation rate of .001, the expected number of generations for the simulated Markov process to enter the 100%-loss state was approximately 75. Hence, the allele loss observed in A(3000) could be due entirely to genetic drift. If this is the case, increasing the population size should reduce considerably the rate of allele loss on test function F1. Whether or not this will also improve the performance of R1 on F1 is not quite so obvious. Clearly, premature convergence is to be avoided. However, increasing the population size may also have the effect of slowing down the rate of convergence beyond acceptable bounds.

In order to evaluate these hypotheses, the behavior of R1 on F1 was also observed with population sizes of 100 and 200, leaving the mutation rate unchanged at .001. Figure 3.5 contrasts the average rate of allele loss for the various population sizes. As expected, increasing the population size reduces the allele loss considerably over the interval of observation. The effect here is heightened by the fact that the time scale is in terms of the number of trials rather than the number of generations. That is, the allele loss was reduced in part because fewer generations (and hence fewer stochastic effects) were involved in generating the same number of sample points.

So we see that the problem of premature loss of alleles can be effectively removed by increasing the population size maintained by R1. However, it remains to be seen what effect this has on the performance of R1. Recall from chapter 1 that two local measures of adaptive performance were defined for functions in E:

$$\text{off-line} \quad x_e^*(s) = \frac{1}{T}\sum_{t=1}^{T} f_e^*(t)$$

$$\text{and on-line} \quad x_e(s) = \frac{1}{T}\sum_{t=1}^{T} f_e(t)$$

where $f_e(t)$ is the performance rating given to the sample solution generated by the adaptive plans for evaluation

FIG 3.5: A1 ALLELE LOSS VARYING POPULATION SIZE

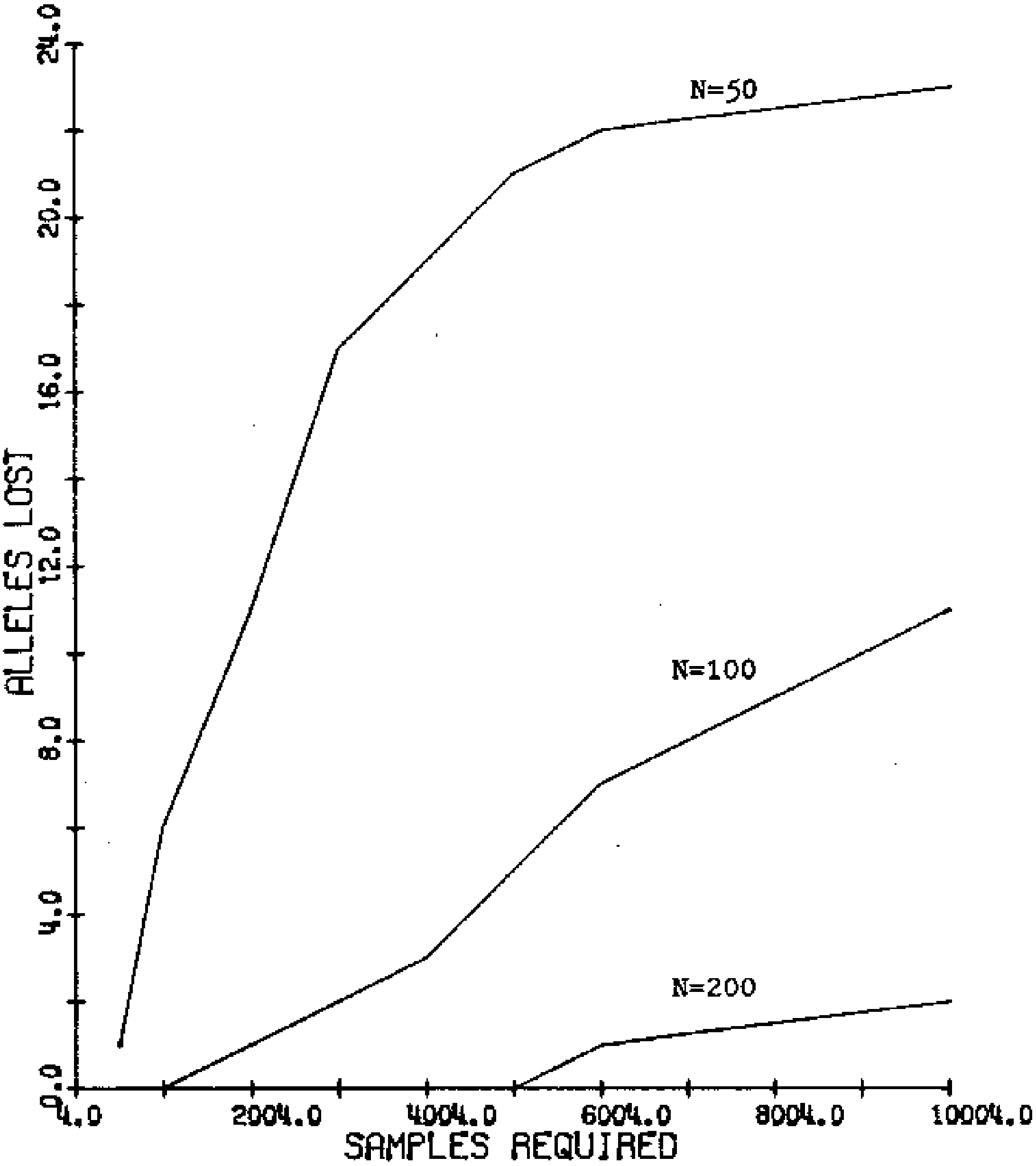


Figure 3.5: The effects of population size on allele loss for R1 on test function F1.

at time t, and where $f_e^*(t)$ is defined by:

$$f_e^*(t) = \min\left\{f_e(1), f_e(2), \ldots, f_e(t)\right\}$$

Figure 3.6 illustrates the effects of population size on $F1^*(t)$. The tradeoff here is clear. Initially R1(50) outperforms the larger populations, but converges prematurely to a non-optimal plateau. R1(100) and R1(200) respond more slowly but yield better long-term performance.

Figure 3.7 illustrates the effects of population size on F1(t). Here the interval required for the tradeoff to become apparent is considerably longer with R1(50) outperforming the others over the first 25,000 trials.

At this point a few words of explanation about the notation being developed in chapter 3 is in order. As we shall see, genetic plan R1 is really a family of plans defined by such parameters as the population size, the mutation rate, and so on. Specific members of this family will be designated by notation of the form R1(X,Y,Z) specifying the actual parameter values. For purposes of clarity, two notational conveniences will be used. First, parameters which have not yet been introduced into the discussion will be suppressed. So, for example, in the preceding paragraph we refer to R1(X) even though by the end of the chapter four parameters will have been defined. Secondly, in a particular context where it is clear that only one

FIG 3.6: R1 OFF-LINE VARYING POPULATION SIZE

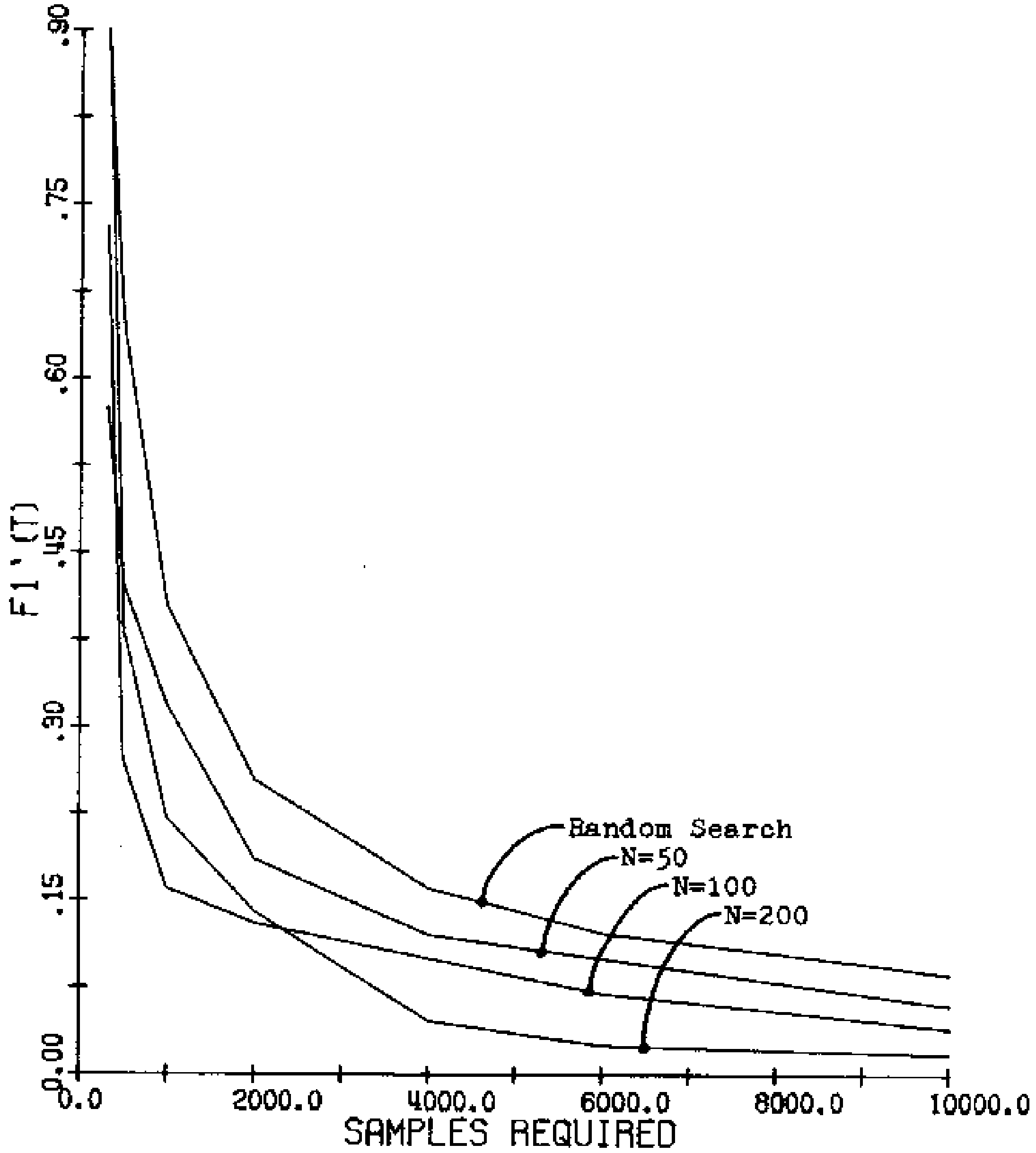


Figure 3.6: The effects of population size on off-line performance of R1 on test function F1.

FIG 3.7: R1 ON-LINE VARYING POPULATION SIZE

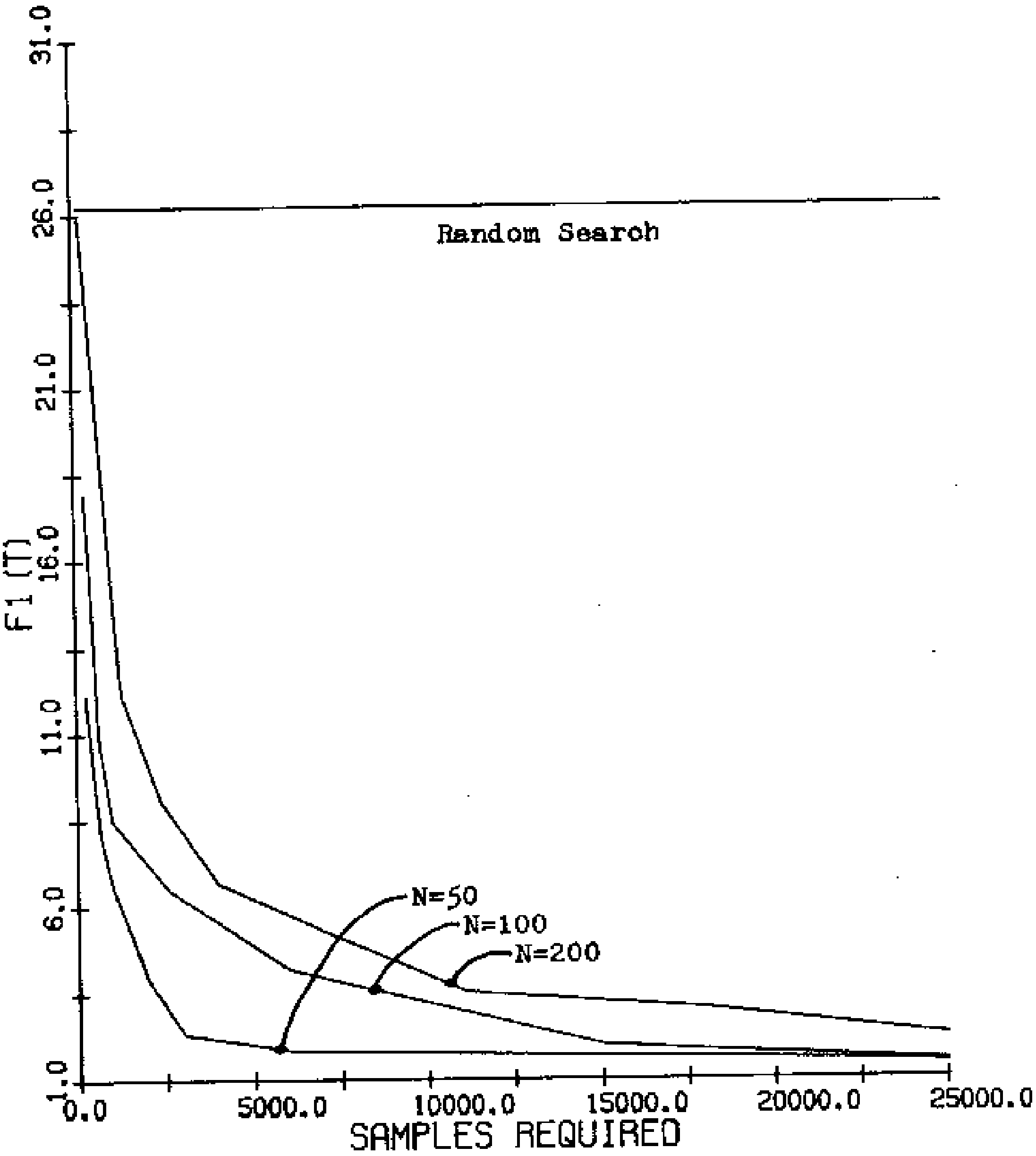


Figure 3.7: The effects of population size on on-line performance of R1 on test function F1.

parameter is under study, the values of the other parameters will be suppressed. So, for example, we may refer to R1(Z) in situations in which X and Y are clearly fixed.

### 3.5 The Effects of Mutation Rate on R1

In this section we explore the second alternative approach to the problem of premature allele loss on F1, namely, changing the mutation rate for R1. Recall from appendix C that R1 maintained a population of 50 individuals and a mutation rate of .001. Referring back to figure 3.3 we note that a considerable reduction in allele loss was achieved in the simulated Markov process with a population size of 50 by increasing the mutation rate. This suggests that the allele loss in R1 might also be reduced by increasing the mutation rate. How an increase in the mutation rate will affect the performance of R1 is not so obvious. Clearly, reducing the premature allele loss will increase the potential for improving long-term performance as we saw in the previous section. However, recall that in chapter 2 we were able to neglect the effects of mutation (at .001) on the near-optimal sampling rate of R1. As we increase the rate of mutation, we increase its effects on sampling which, in turn, may negatively affect the performance of R1.

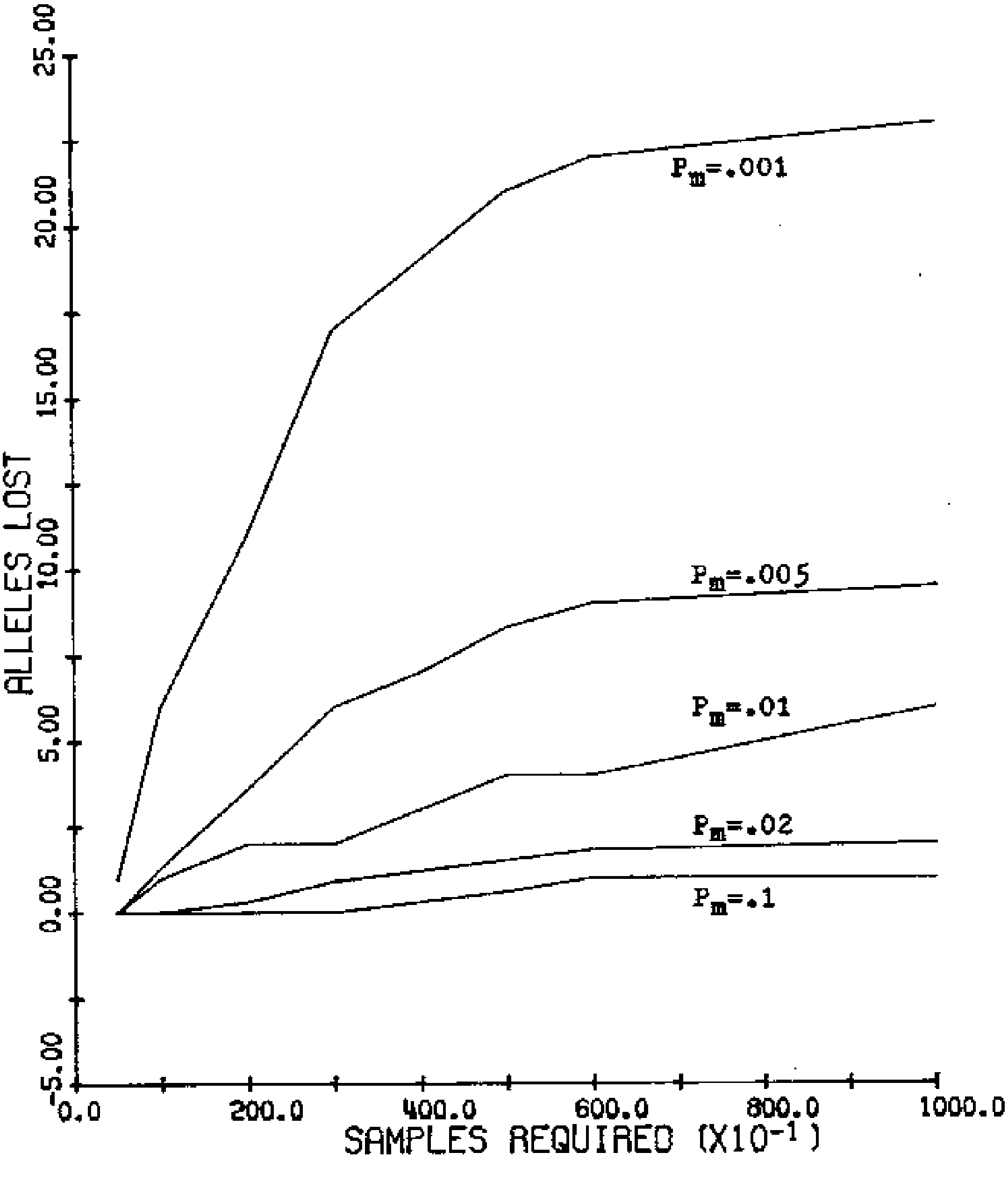
In order to evaluate these hypotheses, the behavior of R1 on F1 was observed with mutation rates of .005, .01,

.02, and .1, leaving the population size unchanged at 50. Figure 3.8 contrasts the average rate of allele loss for the various mutation rates. As expected, increasing the mutation rate reduces considerably the allele loss over the interval of observation. Clearly, the problem of premature allele loss can be solved by raising the mutation rate. However, its effect on the performance of R1 must also be considered.

Figure 3.9 illustrates the effects of increasing the mutation rate on the off-line performance of R1 on F1. Increasing the mutation rate has the effect of improving initial performance. As noted earlier, a mutation rate of the same order of magnitude as 1/POP_SIZE seems to be about the best setting. Increasing the rate more definitely degrades off-line performance. These observations tend to confirm our intuition about R1. With too low a mutation rate, the performance of R1 is degraded by the premature loss of alleles. With too high a mutation rate, the performance is degraded by the sub-optimal allocation of trials to competing hyperplanes.

Figure 3.10 illustrates the effects of increasing the mutation rate on the on-line performance of R1 on F1. Here the effects of mutation are clear. When every trial counts in the performance rating, any increase in the application of a random search operator like mutation has a degrading effect on the performance of R1.

FIG 3.8: R1 ALLELE LOSS VARYING MUTATION



3.8: The effects of mutation rate on allele loss for R1 on test function F1.

FIG 3.9: R1 OFF-LINE VARYING MUTATION

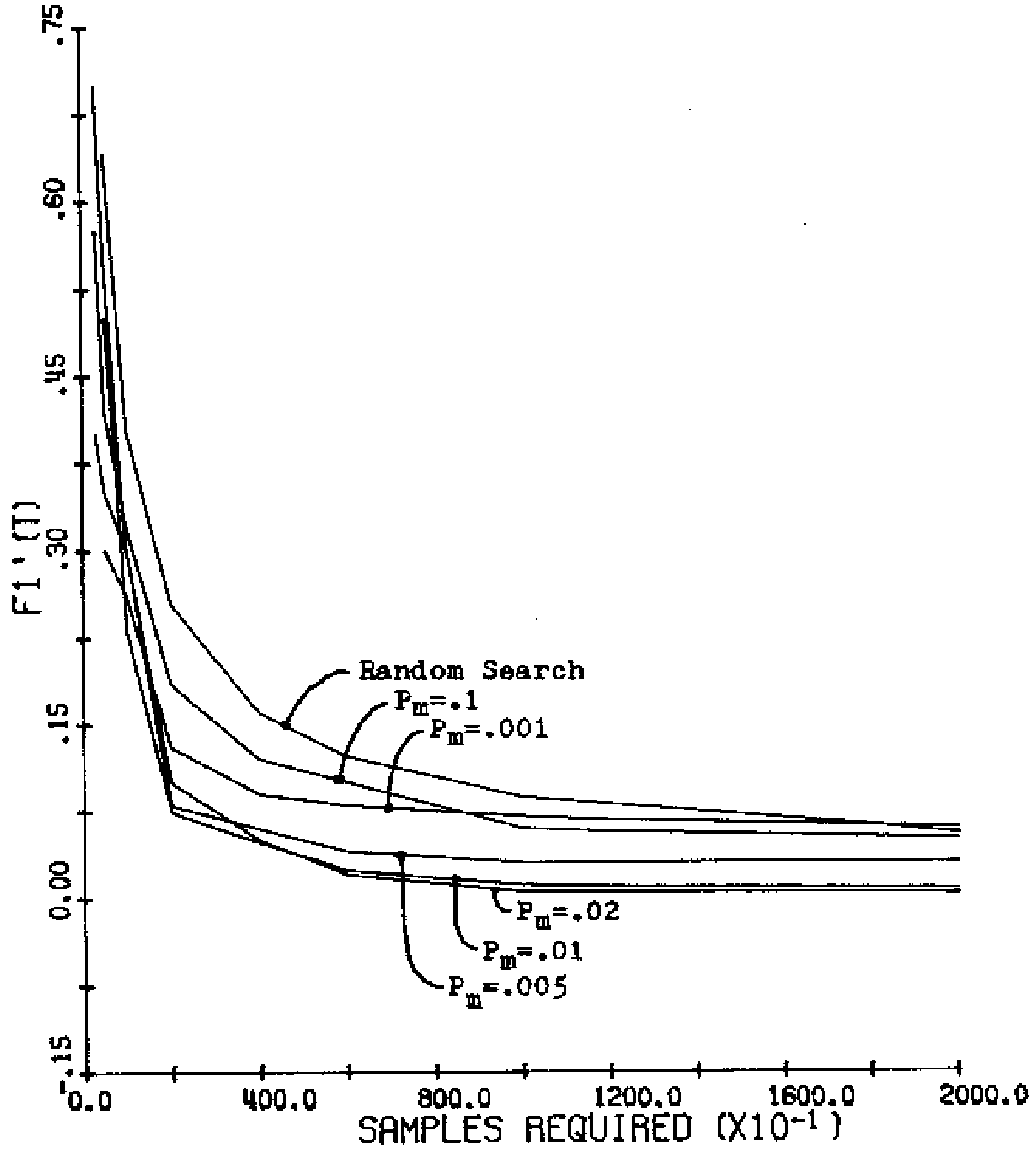


Figure 3.9: The effects of mutation rate on off-line performance of R1 on test function F1.

FIG 3.10: R1 ON-LINE VARYING MUTATION

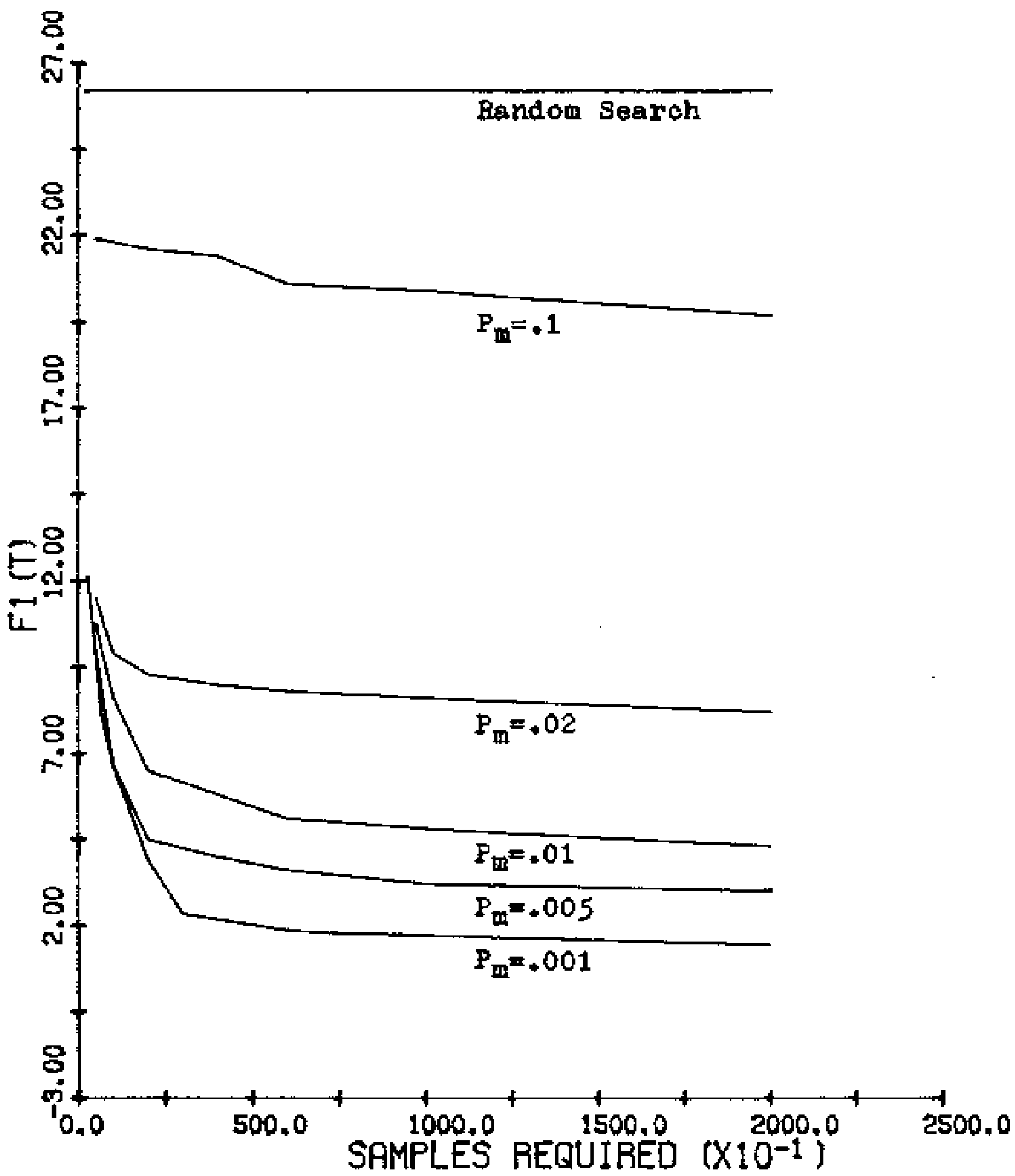


Figure 3.10: The effects of mutation rate on on-line performance of R1 on test function F1.

## 3.6 The Effects of Crossover Rate on R1

As described in appendix C, plan R1 produces an individual for the next generation A(t+1) by selecting two parents, applying crossover to produce an offspring, and then applying mutation to each gene position with probability $P_m$. In this section we explore the effects of reducing the number of individuals in A(t+1) produced by crossover. This variation is easily accomplished within the framework of R1 as follows:

```
Do I=1 to POP_SIZE:
    - select an individual $a_{it}$ from A(t) using the
        selection probabilities.
    - with probability $P_c$ apply crossover to $a_{it}$ by
        selecting a mate from A(t) using the selection
        probabilities and choosing a crossover point.
    - apply mutation at each gene position with
        probability $P_m$.
```

Since crossover is the principle search operator in R1, the effect of lowering the crossover rate is to reduce the number of new trials per generation. This reduction should in turn heighten the stochastic effects noted in the previous sections and increase the rate of allele loss generated by R1 on F1. As a consequence, we would expect the performance of R1 to be adversely affected, since fewer trials will have been allocated before the allele loss has reduced A(t) to a nearly uniform population.

In order to evaluate these hypotheses, the behavior of R1 on test function F1 was observed with crossover rates of $P_c$ = .8, .6, and .4, leaving the population size and mutation rate unchanged at N = 50 and $P_m$ = .001.

Figure 3.11 compares the rate of allele loss for R1 on F1 as a function of the crossover rate. As expected, the rate of allele loss increases as the crossover rate decreases. Figures 3.12 and 3.13 compare the off-line and on-line performance curves for R1 on F1 as a function of the crossover rate. Here the results were unexpected. In spite of the fact that the rate of allele loss is increased, lowering the crossover rate initially improved performance. Only when the crossover rate was lowered to .4 was any negative effect on performance observed.

In an attempt to understand this phenomenon, consider for a moment the effects of the two genetic operators: crossover and mutation. Until the allele loss in A(t) is extensive, applying crossover to two individuals generally produces an offspring quite distinct from either parent. On the other hand, applying mutation to an individual at the rate of .001 changes on the average $\ell$*(.001) alleles. In the case of test function F1, the number of genes per individual is $\ell$=30, so that crossover affects on the average .03 gene positions. So we see that with only these two genetic operators, lowering the crossover rate in R1 has the effect of increasing the likelihood that members of A(t) will produce an offspring nearly identical to themselves, if not identical. Since parents are

FIG 3.11: R1 ALLELE LOSS VARYING CROSSOVER

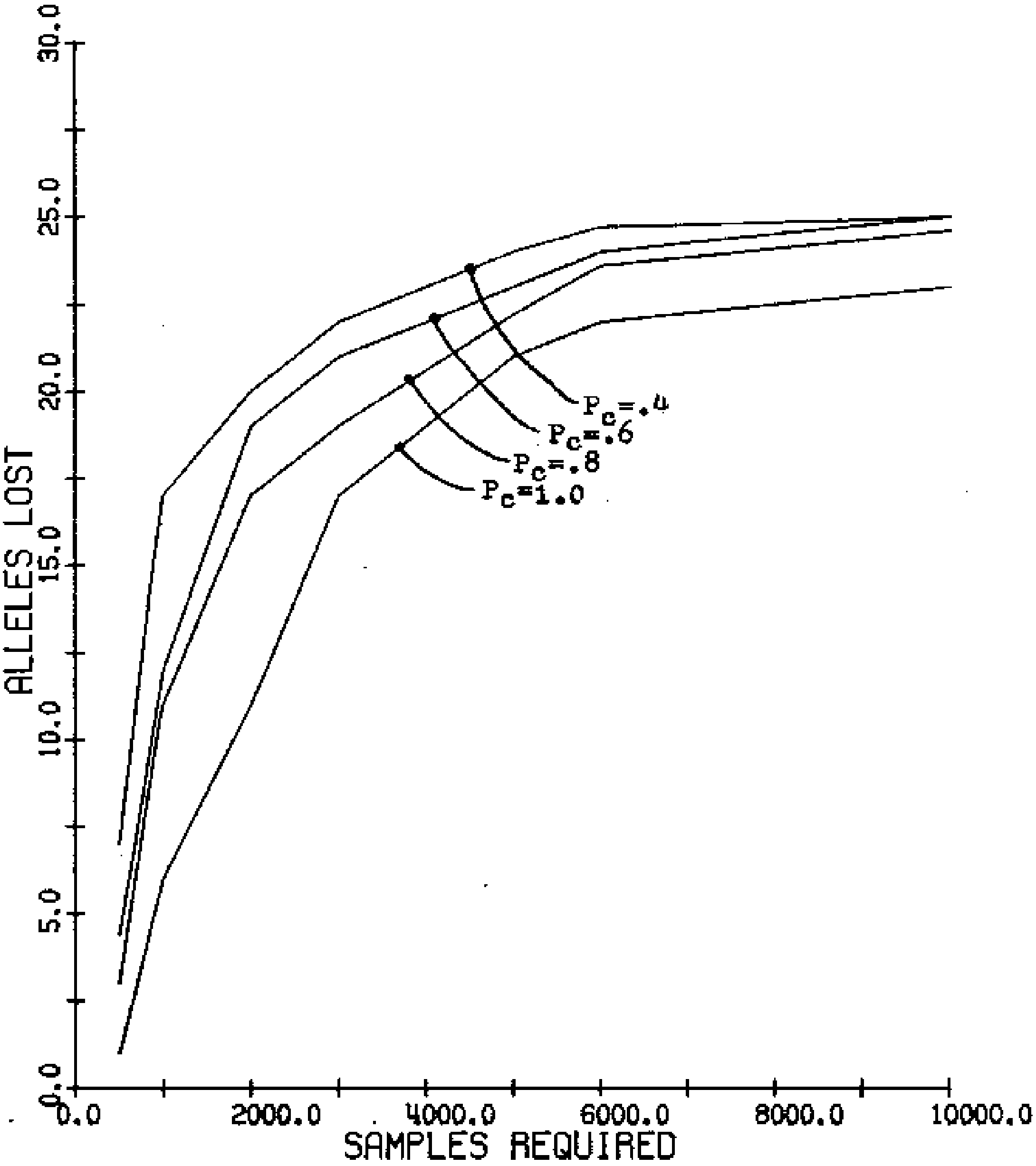


Figure 3.11: The effects of crossover rate on allele loss for R1 on test function F1.

FIG 3.12: R1 OFF-LINE VARYING CROSSOVER

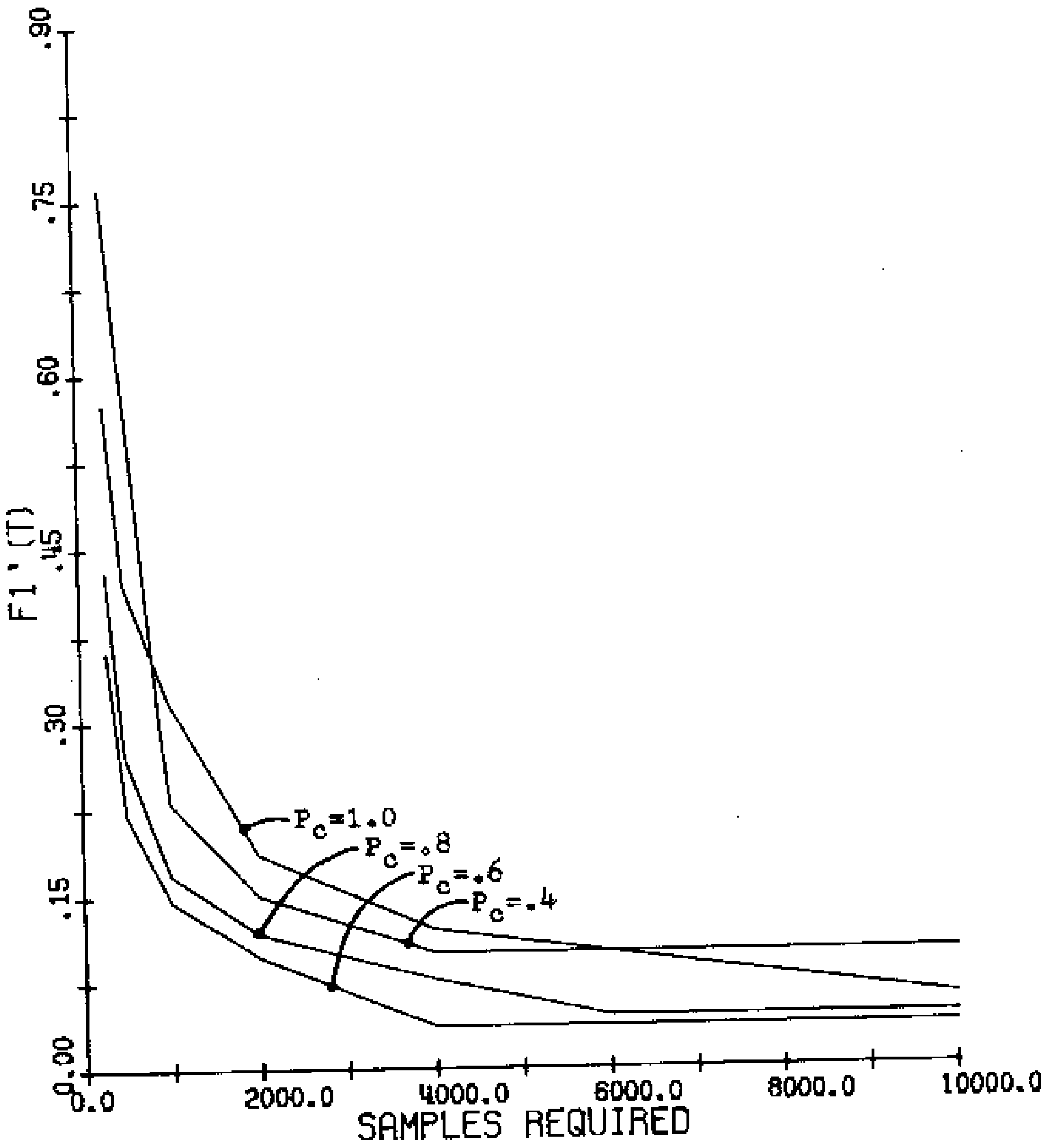


Figure 3.12: The effects of crossover rate on off-line performance of R1 on test function F1.

FIG 3.13: R1 ON-LINE VARYING CROSSOVER

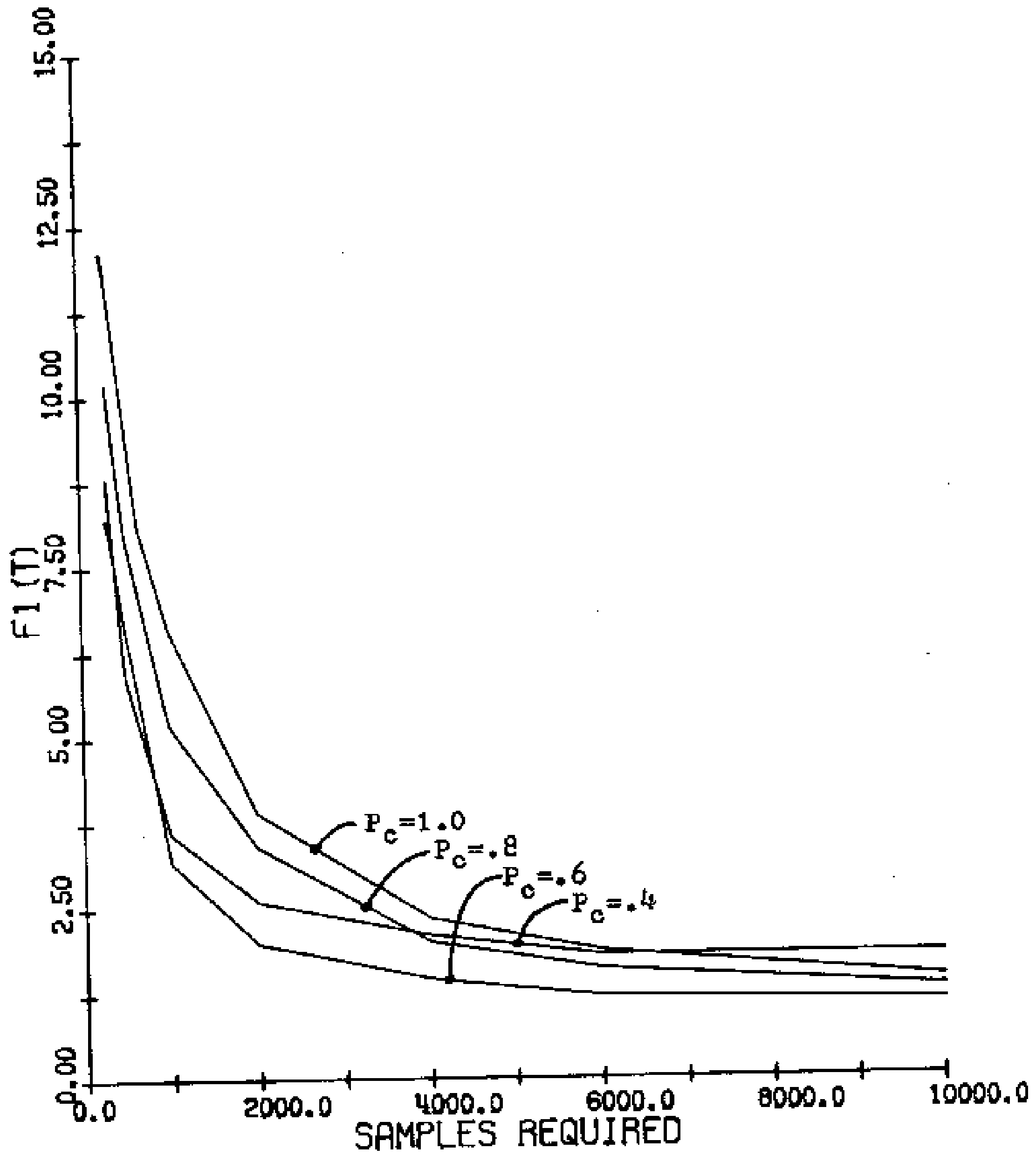


Figure 3.13: The effects of crossover rate on on-line performance of R1 on test function F1.

selected on the basis of performance, the result is to increase the probability of high-performance individuals surviving into the next generation. Here again we encounter the delicate tradeoff between further exploration and preserving the status quo. Applying crossover at the rate of 1.0 seems to be too high a sampling rate for R1(50,.001). High-performance individuals are discarded faster than crossover can produce improvements, terminating with the usual premature convergence due to allele loss. On the other had, a crossover rate of .4 seems to be too low a sampling rate for R1(50,.001). Too little exploration combined with the increased rate of allele loss causes rapid convergenge to a non-optimal plateau.

## 3.7 The Effects of Generation Gap on R1

Recall that plan R1 is designed to produce the next generation A(t+1) by replacing all N individuals from A(t). A genetic model of this type is described as having non-overlapping generations; that is, parents do not exist simultaneously with their offspring. It is not immediately clear whether non-overlapping generations are good or bad in an artificial genetic adaptive model. From an implementation point of view, the distinction poses the classic tradeoff between storage and cpu time. Non-overlapping models require storage for two populations: A(t) and A(t+1). If generations overlap, less storage is required, but more generations are required (recomputing selection probabilities) to produce the same number of trials. In this section we ignore the time-space tradeoff and explore the effect of overlapping generations on the performance of R1.

Overlapping generations can be incorporated into R1 by adding a new parameter called the generation gap G which specifies the fraction of A(t+1) to be generated via the genetic operators. Obviously, G must lie in the range $0 \leq G \leq 1$ with $G = 1$ the default value used in the previous simulations. If $G < 1$, the remaining positions in A(t+1) are filled by selecting individuals from A(t) without replacement using a uniform distribution. As before, we inquire as to the expected number of offspring produced by an individual $a_{it}$ in A(t). If we assume that the selection probabilities do not change much over the life-time of an individual, then on any particular generation the expected number of offspring from $a_{it}$ is given by:

$$(N*G) * p(a_{it})$$

where N is the population size and $p(a_{it})$ is the probability of selecting $a_{it}$. The number of generations $a_{it}$ is expected to survive is simply the waiting time to extinction. Each generation $a_{it}$ has a probability G of disappearing; hence, the waiting time is $\frac{1}{G}$ and the total number of offspring produced by $a_{it}$ is given by:

$$(\frac{N*G}{G})*p(a_{it}) = N*p(a_{it})$$

which is the same as the non-overlapping model.

On the basis of our experiences with the crossover rate, we would expect that reducing the generation gap should increase the rate of allele loss since fewer trials are made per generation. Its effect on performance is not quite so obvious. Clearly, the reduced sampling rate should improve the performance of R1(50,.001) on F1 as it did in the case of crossover. However, note that the individuals which are likely to survive into the next generation are selected at random, rather than on the basis of performance. This should reduce the extent of the improvement observed when the crossover rate was reduced.

In order to evaluate these hypotheses, the behavior of R1 was observed on F1 with generation gaps of .8, .6, and .4 leaving the population size, mutation rate, and crossover rate unchanged at N = 50, $P_m$ = .001 and $P_c$ = 1.0.

Figure 3.14 compares the rate of allele loss for R1 on F1 as a function of the generation gap. As expected, the rate of allele loss increases as the generation gap decreases. Figures 3.15 and 3.16 compare the off-line and on-line performance curves for R1 on F1 as a function of the generation gap. As expected, lowering the generation gap provides an initial improvement in performance

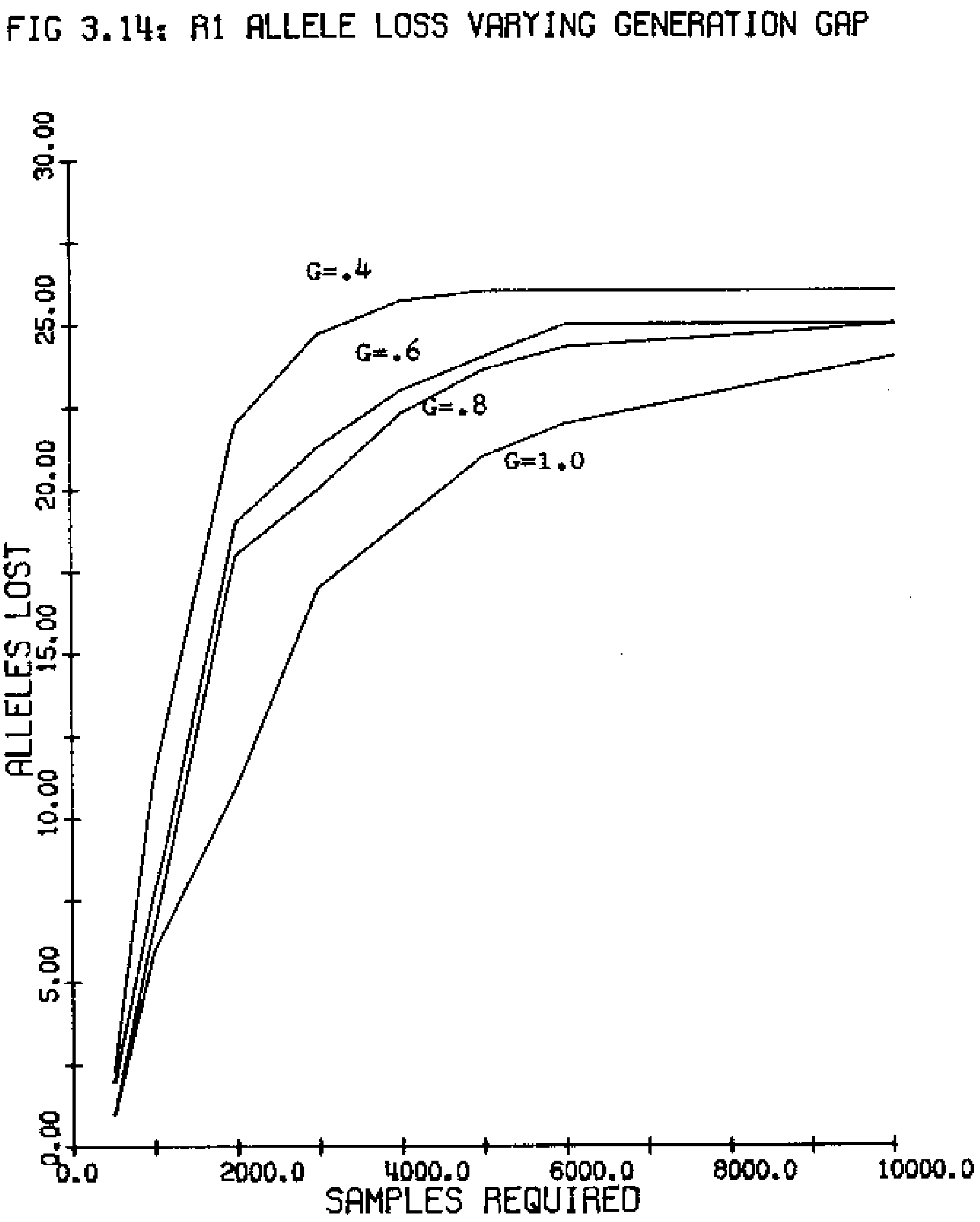


Figure 3.14: The effects of generation gap on allele loss of R1 on test function F1.

FIG 3.15: R1 OFF-LINE VARYING GENERATION GAP

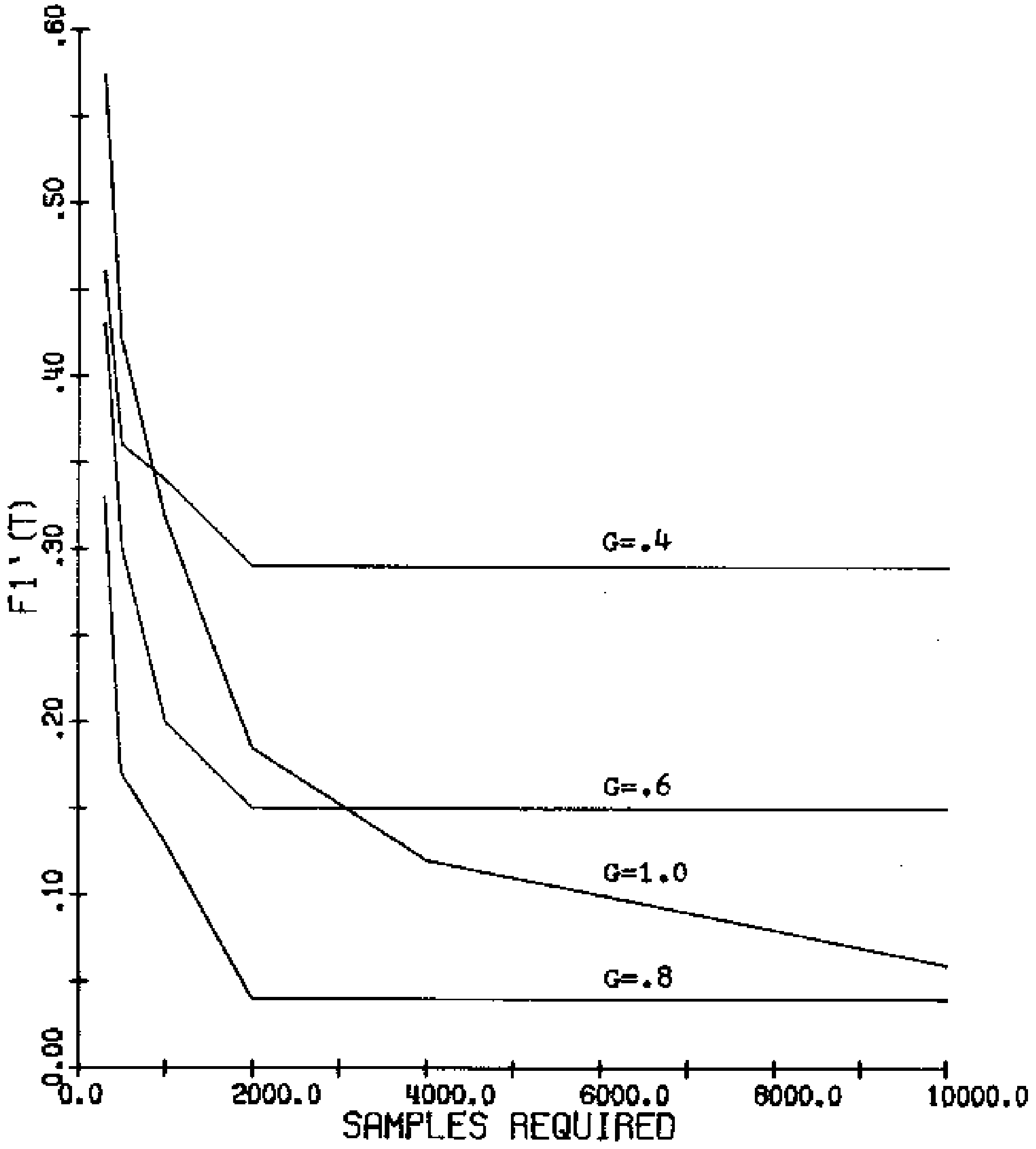


Figure 3.15: The effects of generation gap on off-line performance of R1 on test function F1.

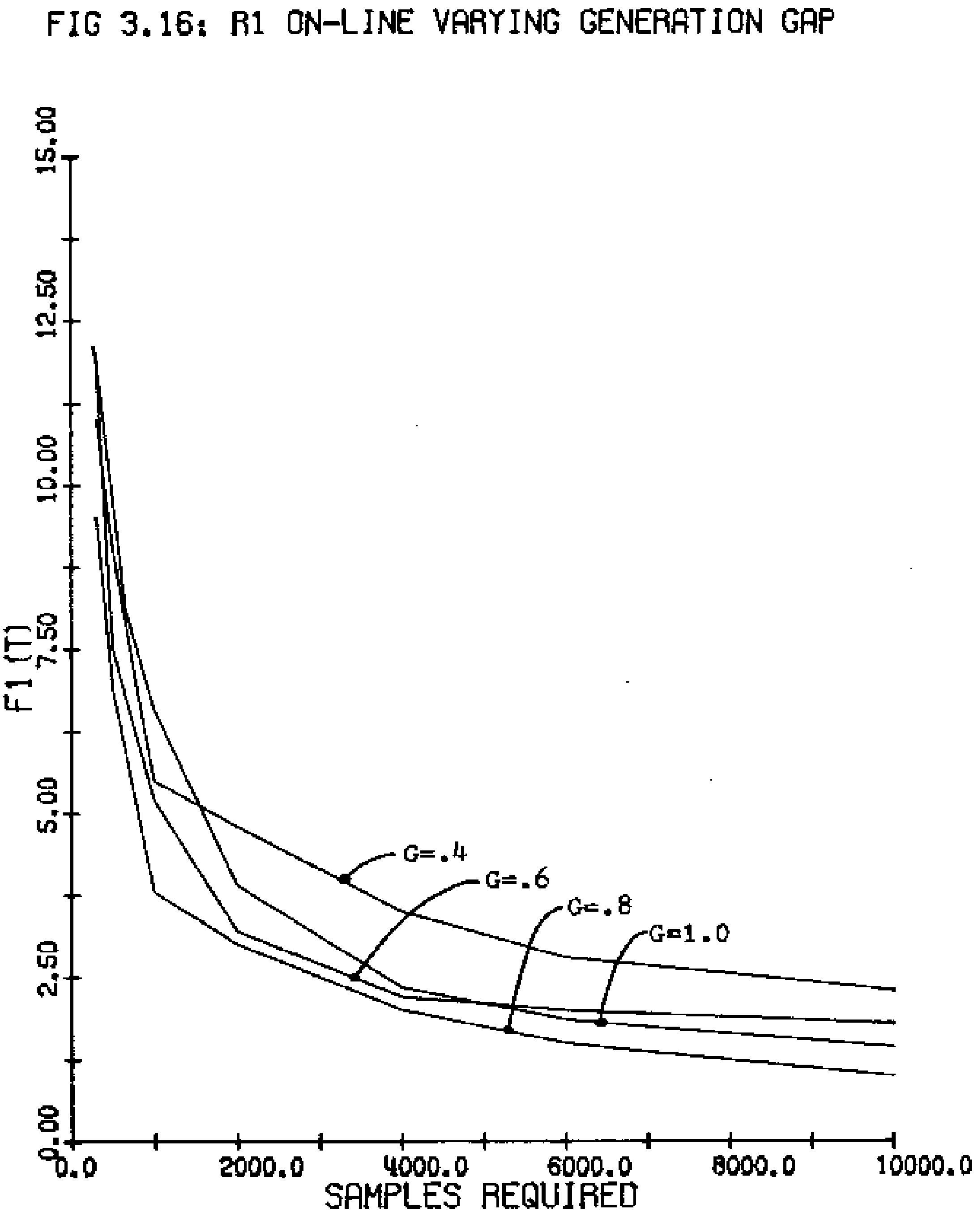


Figure 3.16: The effects of generation gap on on-line performance of R1 on test function F1.

but also produces an earlier convergence to a non-optimal plateau, the improvement being considerably less dramatic than that generated by a corresponding reduction in the crossover rate.

## 3.8 Improving the Performance of R1 on F1

In the preceding sections we have isolated several parameters in the definition of plan R1 and have explored the effects of independently changing these parameters on the behavior of R1 on test function F1. The motivation for these studies was to gain further insight into how R1 operates and, in particular, to analyze the problem of premature convergence to a non-optimal plateau. As we have seen, no one of the parameters studied both satisfactorily resolves the problem of premature convergence and substantially improves the performance of R1 on F1. In this section we explore the possibility of resolving these problems by changing various combinations of parameter settings for R1.

In this chapter R1 has evolved into a family of genetic plans, a member of which is selected by specifying the values of four parameters: the population size N, the mutation rate $P_m$, the crossover rate $P_c$, and the generation gap G. Ideally, we would like to apply optimization techniques to the space of algorithms defined by these parameters and optimize with respect to premature allele loss, off-line, and on-line performance.

In reality, however, this approach is prohibited by the cost involved in analyzing the behavior of a single member of this family. Because each plan is a stochastic process, at least 5 (and often more) simulations are required to produce analysis measurements within reasonable standard error limits. In terms of present university rates, this can mean a cost of as much as $50 to evaluate a single plan on F1 alone. We will avoid this problem by applying the insight gained from the previous sections to the selection of a few well-chosen combinations of parameters to confirm and extend our understanding of the basic genetic plan R1.

We begin by noting that of the four parameters analyzed, reducing the crossover rate produced the single best improvement in the performance of R1 on F1, even though the allele loss rate actually increased in the process. This, we felt, was due to the reduced sampling rate effected by reducing the number of new individuals produced by crossover. As we observed, reducing the generation gap also lowered the sampling rate, but the improvement in performance is not as substantial as the corresponding reduction in crossover because of the difference in the kind of individual most likely to survive into the next generation. If these observations are correct, we would expect that sampling rates produced by a combination of reduced crossover rates and generation gaps should not be as effective in improving the performance of R1 on F1 as the equivalent sampling rate produced by crossover alone.

In order to evaluate this hypothesis, the behavior of R1 on F1 was observed for 4 different combinations of crossover rates and generation gaps ($P_c$=.8,G=1.0), ($P_c$=.8,G=.8), ($P_c$=.6,G=1.0), and ($P_c$=.6,G=.8), holding the population size and mutation rate fixed at N=50 and $P_m$=.001. The performance curves generated by these combinations on test function F1 are illustrated in Figures 3.17 and 3.18, and they confirm our intuition about the behavior of plan R1. R1(.8,.8) performed better on F1 than R1(.8,1.0), but not as well as R1(.6,1.0) which has an equivalent sampling rate. As we saw previously, a combined sampling rate of less than .6 (in this case R1(.6,.8)) adversely affects the performance of R1 on F1. These observations suggest that reasonable settings for the crossover rate and generation gap of R1 are approximately $P_c$=.6 and G=1.0.

Alternatively, we saw that increasing the mutation rate improved considerably the allele loss rate, but the effects on performance were mixed. The best on-line performance was generated by a mutation rate of approximately $P_m$=1/N while any increase in mutation adversely affected on-line performance. This, we felt, was due to the fact that mutation is in fact an effective method for combatting premature allele loss and, hence, improving off-line performance. But because it accomplishes

FIG 3.17: R1(50,.001,X,Y) OFF-LINE PERFORMANCE

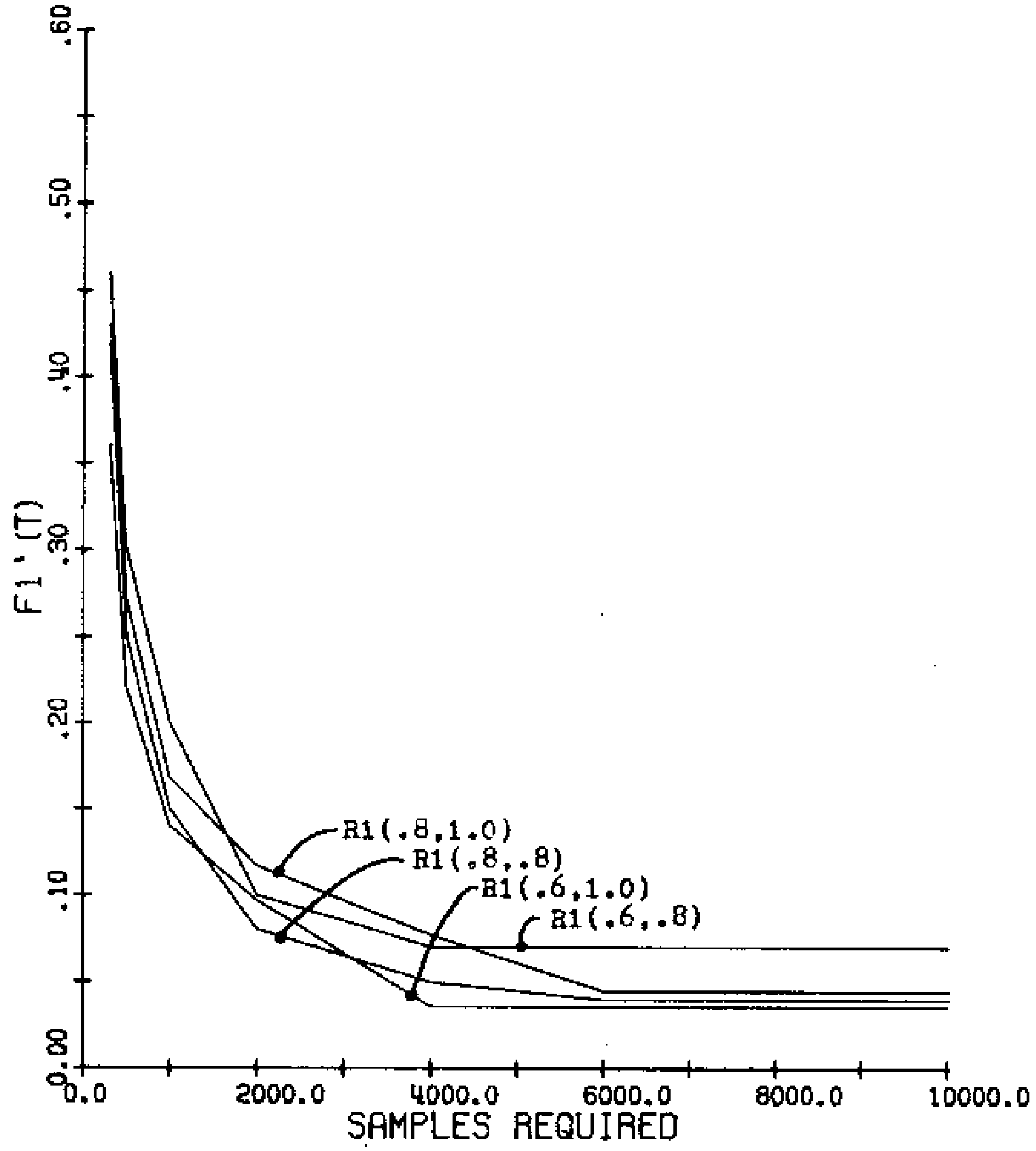


Figure 3.17: Off-line performance of R1 on F1 as a function of crossover rate and generation gap.

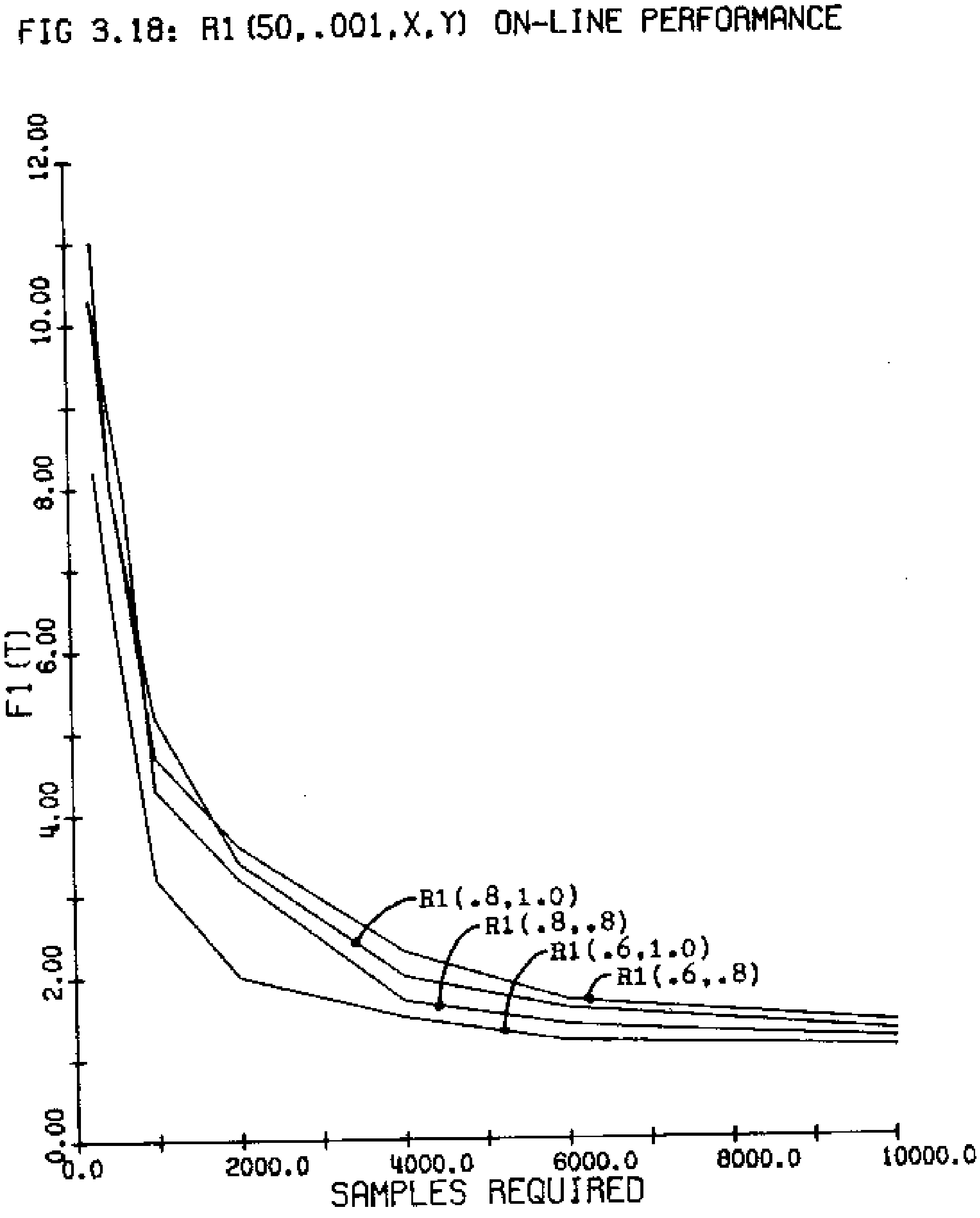


Figure 3.18: On-line performance of R1 on F1 as a function of crossover rate and generation gap.

this in its random sampling style, the price is paid in its adverse effect on on-line performance. If these observations are correct, we should expect to see the same kind of behavior changes produced by varying the mutation of R1(50,x,.6,1.0) as we saw with R1(50,x,1.0,1.0), but perhaps less dramatic changes since a crossover rate of .6 has already improved the performance curves.

To evaluate these hypotheses, the behavior of R1 on F1 was observed with mutation rates of $P_m$=.001, .01, and .1, leaving the population size, the crossover rate, and the generation gap fixed at N=50, $P_c$=.6, and G=1.0. Figures 3.19 and 3.20 compare the performance curves generated by the various mutation rates. These observations confirm our intuition about the effects of mutation on the performance of R1 and emphasize again the tradeoff between on-line and off-line performance.

Finally, we observed that increasing the population size reduced the rate of premature allele loss, but its effects on the performance of R1 were mixed. Larger populations responded more slowly but generated better long-term off-line performance, while increasing the population size adversely affected on-line performance over the interval of observation. This, we felt, was due to the fact that increasing the population size reduces considerably the allele loss and hence improves long-term performance, but at the cost of taking more samples before a decision (a generation) is made con-

FIG 3.19: R1 (50,X,.6,1.0) OFF-LINE PERFORMANCE

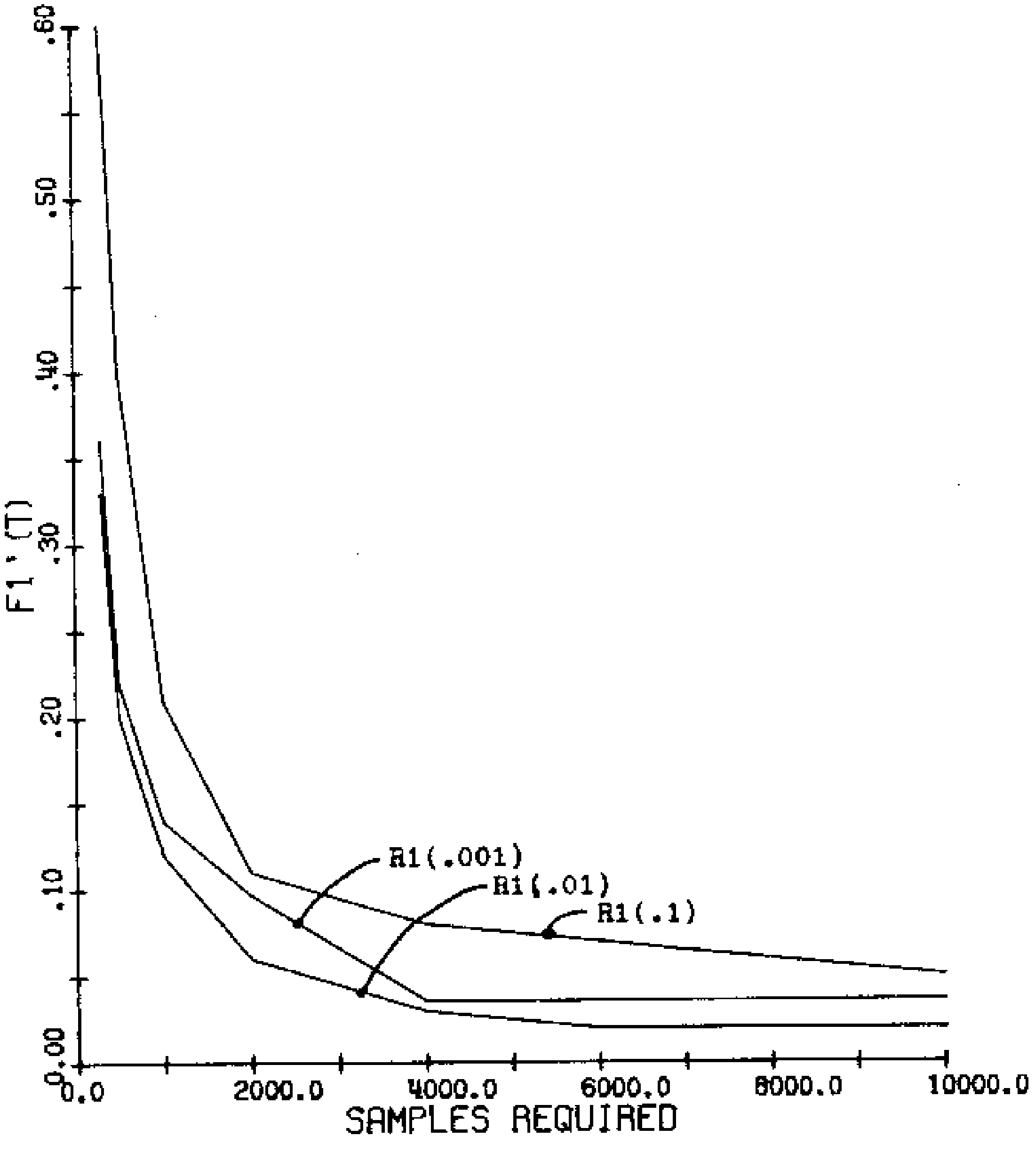


Figure 3.19: Off-line performance of R1 on F1 as a function of mutation rate.

FIG 3.20: R1(50,X,.6,1.0) ON-LINE PERFORMANCE

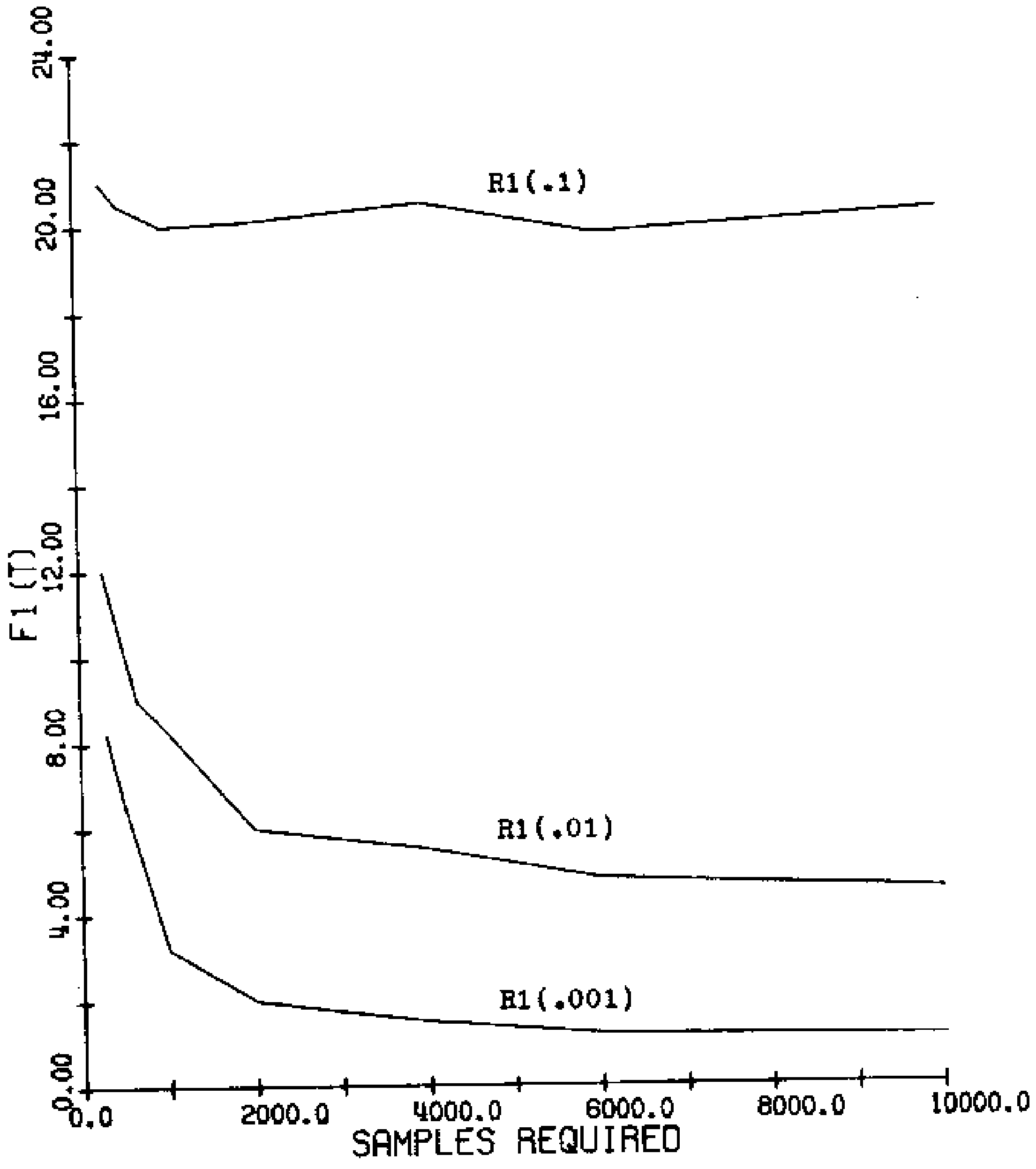


Figure 3.20: On-line performance of R1 on F1 as a function of mutation rate.

cerning the re-distribution of trials. If these observations are correct, we should expect to see the same kind of changes in the behavior produced by increasing the population size of R1(x,.001,.6,1.0) as we saw with R1(x,.001,1.0,1.0), but perhaps less dramatic changes since a crossover rate of .6 has already improved the performance curves.

To evaluate these hypotheses, the behavior of R1 on F1 was analyzed for population sizes of N=50, 100, and 200, leaving the mutation rate, the crossover rate, and the generation gap unchanged at $P_m$=.001, $P_c$=.6, and G=1.0. Figures 3.21 and 3.22 compare the performance curves produced by the various population sizes. These observations confirm our intuition about the effects of population size and emphasize again the tradeoff between on-line and off-line performance.

These observations also suggest that no particular combination of the four parameter settings is going to dramatically improve the performance of R1 on F1, and that perhaps the off-line performance generated by R1(50,.01,.6,1.0) and the on-line performance generated by R1(50,.001,.6,1.0) are about the best that can be expected from the basic genetic plan R1.

## 3.9 Summary

We began this chapter by noting that, although plan R1 outperforms random search on test function F1, it

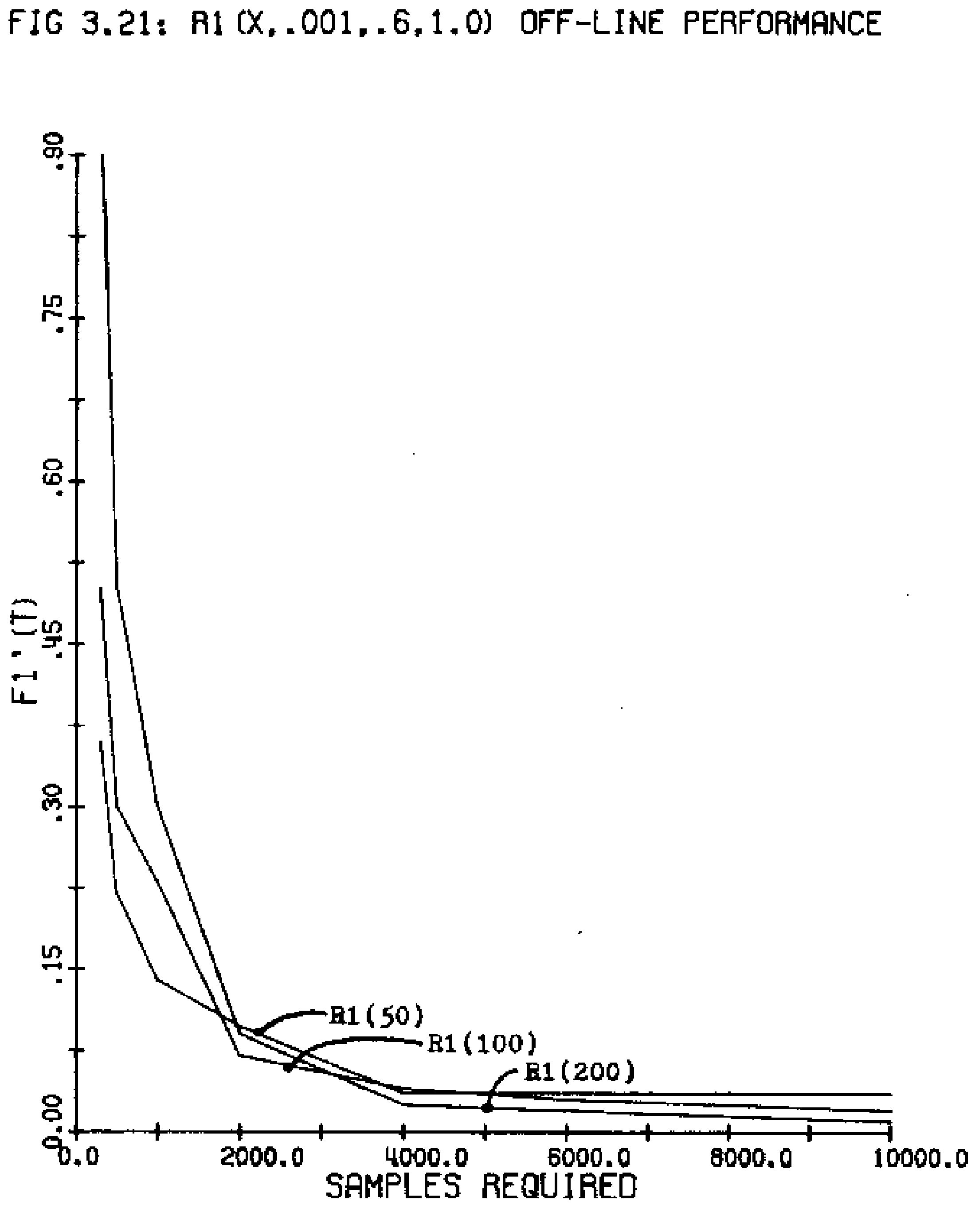


Figure 3.21: Off-line performance of R1 on F1 as a function of population size.

FIG 3.22: R1(X,.001,.6,1.0) ON-LINE PERFORMANCE

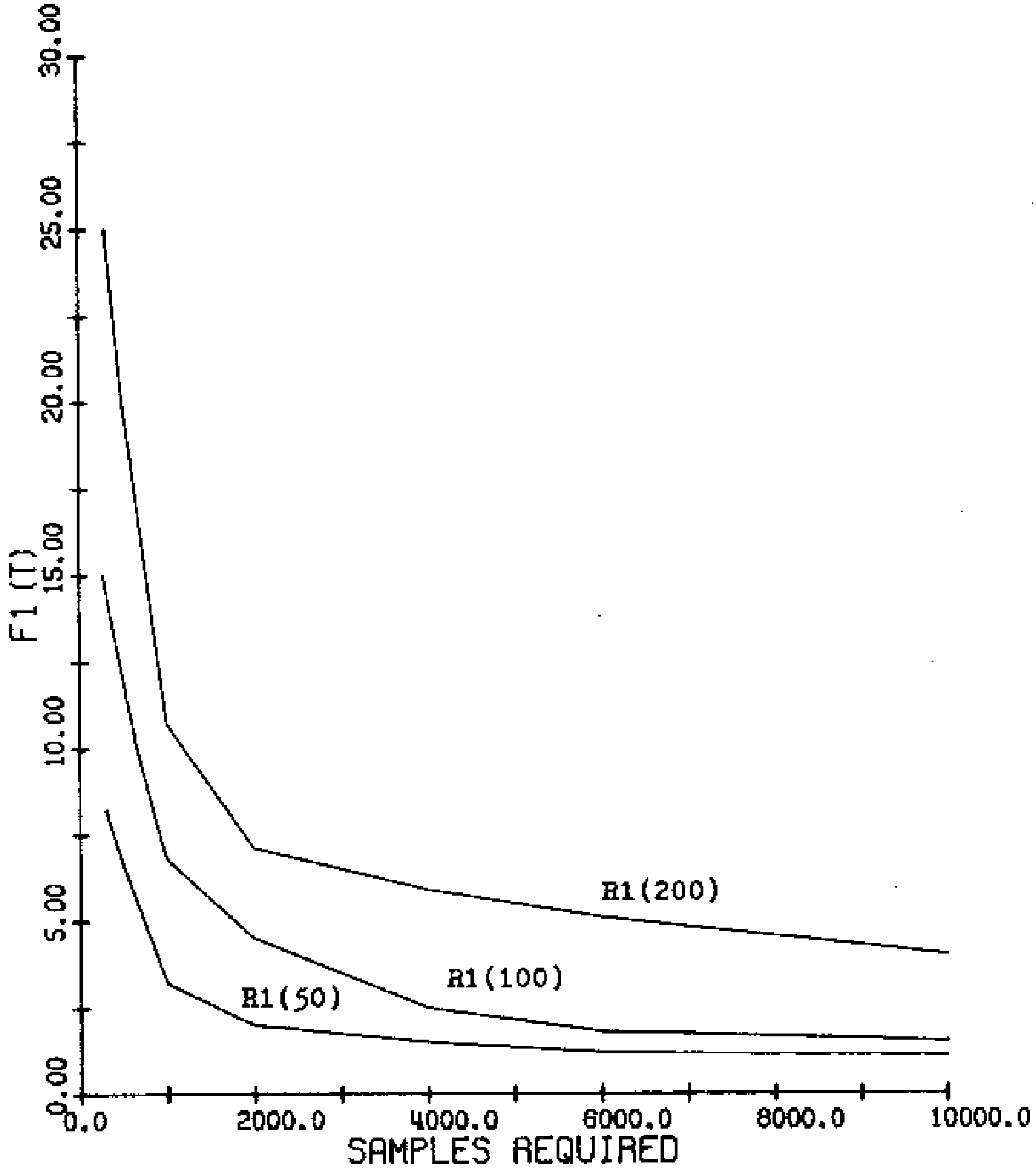


Figure 3.22: On-line performance of R1 on F1 as a function of population size.

suffers from the problem of premature convergence to a non-optimal performance plateau caused by a loss of alleles in A(t), even though on F1 no allele has any selective advantage over its competitor. We saw via Markov process simulation that such allele loss rates can in fact be caused by the stochastic side-effects of generating new populations from old ones using only a finite number of random samples. In order to understand and, perhaps, alleviate the problem, the effects of changing various parameters of genetic plan R1 were analyzed. As we observed, increasing the population size maintained by R1 reduces considerably the rate of allele loss, but also poses a tradeoff in performance. Larger populations respond more slowly, but yield better long-term performance. Alternatively, the allele loss can be counteracted by increasing the mutation rate. However, the effects on performance are mixed. A mutation rate of about 1/POP_SIZE seems to generate the best off-line performance for R1. But any increase in the mutation rate adversely affects on-line performance. Reducing the crossover rate did nothing to alleviate the premature convergence problem; rather, it increased the rate of allele loss. Surprisingly, however, it did effect an improvement in the initial performance of R1, suggesting that generating A(t+1) by replacing every individual in A(t) was, perhaps, too high a sampling rate. Reducing the generation gap of R1 was also observed to increase the rate of allele loss rather than alleviate it. As with crossover, even with the increased rate of allele loss, an improvement in initial performance was observed. Finally, several combinations of parameter values were analyzed in an attempt to improve the performance of R1 on F1. As we observed, no particular settings significantly improved performance suggesting that this is about the best we can expect from R1 on F1.