Chapter 4

PERFORMANCE EVALUATION OF GENETIC ADAPTIVE PLANS

4.1 Introduction

In the preceding chapters we have introduced a class of genetic adaptive algorithms for study, and we have focused our attention in particular on the behavioral characteristics of the basic family of plans R1 on test function F1. The emphasis has been on understanding how these adaptive models operate in finite time and space. In this chapter we apply the insight gained by these studies to the problem of improving the performance of genetic adaptive plans on E.

4.2 The Performance of R1 on E

In the last chapter we studied the effects of changing the various parameters of R1 on its performance on test function F1. In this section we will extend these observations to the performance of R1 on E. As noted earlier, optimizing the performance of R1 over its parameter space is prohibited by the cost of simulation analysis on existing facilities. As before, however, we extend our insight by analyzing a few well-chosen members of the family of plans defined by R1. Recall that in choosing a particular member in R1, four parameters must be specified: the population size N, the mutation rate P_m, the crossover rate P_c, and the generation gap G. Based on the results of the previous chapter, the following members were chosen for analysis on E:

	N	P_m	P_c	G
R1(	50,	.001,	1.0,	1.0)
R1(	50,	.001,	.8,	1.0)
R1(	50,	.001,	.6,	1.0)
R1(	50,	.001,	.8,	.8)
R1(	50,	.01 ,	.8,	1.0)
R1(	50,	.01 ,	.6,	1.0)
R1(	100,	.001,	.8,	1.0)
R1(	100,	.001,	.6,	1.0)

Recall from chapter 1 that, for each f_e in the environment E, local robustness was defined by

$$x_e(T) = \frac{1}{T} \sum_{i=1}^{T} f_e(t)$$

for on-line performance and

$$x_e^*(T) = \frac{1}{T} \sum_{i=1}^{T} f_e^*(t)$$

for off-line performance with the associated <u>global</u> measures of robustness defined by

$$X_E(T) = \frac{1}{|E|} \sum_{E} x_e(T)$$

and

$$X_E^*(T) = \frac{1}{|E|} \sum_{E} x_e^*(T)$$

respectively. Based on previous experience and with an eye for practical applications, T=6000 was chosen as a reasonable bound on the interval of observation. Tables 4.1a and 4.1b summarize the performance measures obtained from this evaluation and there were very few surprises.

The first three members analyzed differed only in their crossover rates of 1.0, .8, and .6 respectively. As we observed before on test function F1, reducing the crossover rate improves both off-line and on-line performance. The fourth member analyzed illustrates that on E as well as F1, reducing the generation gap is not as effective as reducing the crossover rate. The fifth and sixth members analyzed confirm on E the observation regarding the tradeoff between off-line and on-line performance presented by changing the mutation rate. The only mild surprise came with the evaluation of the last two members supporting a population of size 100. Contrary to our earlier observations, increasing the population size degraded both off-line and on-line performance indices. Upon reflection, however, the reason for this change seems clear. Recall that our earlier observations over 10,000 trials suggested that increasing the population size improved long-term performance at the expense of short-term performance. By shortening the evaluation period to 6000 trials, we have put more emphasis on the short-term behavior and

T=6000	N =50 P_m=.001 P_o=1.0 G = 1.0	N = 50 P_m=.001 P_c=.8 G = 1.0	N = 50 P_m=.001 P_o=.6 G = 1.0	N = 50 P_m=.001 P_c=.8 G =.8	N = 50 P_m=.01 P_c=.8 G =1.0	N =50 P_m=.01 P_c=.6 G =1.0	N =100 P_m=.001 P_c=.8 G =1.0	N =100 P_m=.001 P_o=.8 G =1.0
$x^*_{F1}(T)$	.22	.199	.146	.185	.136	.169	.214	.157
$x^*_{F2}(T)$	.34	.23	.31	.36	.199	.147	.288	.248
$x^*_{F3}(T)$	-26.2	-26.5	-27.2	-26.4	-26.6	-26.4	-26.1	-26.8
$x^*_{F4}(T)$	35.9	34.67	34.5	37.5	34.1	34.2	36.6	37.8
$x^*_{F5}(T)$	4.13	3.75	2.56	3.58	3.31	2.46	4.92	4.75
$X^*_E(T)$	2.88	2.47	2.06	2.64	2.23	2.11	3.18	3.23

Table 4.1a: Off-line performance of R1 on E

T=6000	N =50 P_m=.001 P_c=1.0 G =1.0	N =50 P_m=.001 P_c=.8 G =1.0	N =50 P_m=.001 P_c=.6 G =1.0	N =50 P_m=.001 P_c=.8 G =.8	N =50 P_m=.01 P_c=.8 G =1.0	N =50 P_m=.01 P_c=.6 G =1.0	N =100 P_m=.001 P_c=.8 G =1.0	N =100 P_m=.001 P_c=.8 G =1.0
$x_{F1}(T)$	4.6	4.27	3.65	4.02	7.35	7.6	6.94	5.69
$x_{F2}(T)$	78.4	76.8	76.7	78.4	161.6	160.5	134.8	114.37
$x_{F3}(T)$	-22.1	-23.17	-23.3	-22.1	-18.4	-19.01	-21.4	-21.3
$x_{F4}(T)$	97.3	93.2	89.9	96.5	127.8	128.4	102.7	98.14
$x_{F5}(T)$	42.2	34.1	36.5	41.2	84.1	92.2	45.4	38.1
$X_E(T)$	40.08	37.04	36.69	39.60	72.49	73.94	53.61	47.01

Table 4.1b: On-line performance of R1 on E

hence should expect to see a performance degradation.

These results confirm our earlier observations of the behavior of R4, and they suggest that the off-line performance of R1(50,.01,.6,1.0) and the on-line performance by R1(50,.001,.6,1.0) are about the best that can be expected from these simple genetic plans.

4.3 Elitist Model R2

Earlier observations of the behavior of R1 suggested that generating N new individuals for each new population A(t+1) was in fact too high a sampling rate. High-performance individuals were lost before the genetic operators were able to produce improvements. An improvement in performance was obtained by reducing the crossover rate and/or the generation gap which, in turn, reduced the number of new individuals produced for A(t+1). Moreover, we observed that reducing the crossover rate produced better performance improvements than a corresponding reduction in the generation gap. This, we felt, was due to the fact that, because of the selection processes, high-performance individuals were more likely to survive into the next generation via a reduction in the crossover rate than with a reduction in the generation gap. In this section we consider the implications of giving high-performance individuals special treatment by modifying the basic plan R1 to include the following elitist policy:

Let $a^*(t)$ be the best individual generated up to time t. If, after generating A(t+1) in the usual fashion, $a^*(t)$ is not in A(t+1), then include $a^*(t)$ to A(t+1) as the $(N+1)^{th}$ member.

Such a policy guarantees that the best individual generated will not be lost from one generation to the next as a consequence of sampling effects or the application of genetic operators. From the hyperplane analysis point of view, this policy will bias the distribution of trials in favor of those hyperplane partition elements which have produced the best-performing individual. This suggests that the effect of such a policy on performance may be to improve local search at the expense of global search.

In order to evaluate the effects of such a policy, two members of this family were evaluated on E:

R2(50,.001,.8,1.0)
and R2(50,.001,.6,1.0)

Figures 4.1 - 4.3 compare the behavior of these plans with their R1 counterparts on test function F1. Figure 4.1 illustrates that the allele loss rate is slightly better with R2. This is probably due to the fact that appending the best individual to A(t+1) prevents one or two alleles from being counted as lost. Figures 4.2 and 4.3 illustrate that R2 produces both off-line and on-line curves for F1 which are significantly better than those produced by R1.

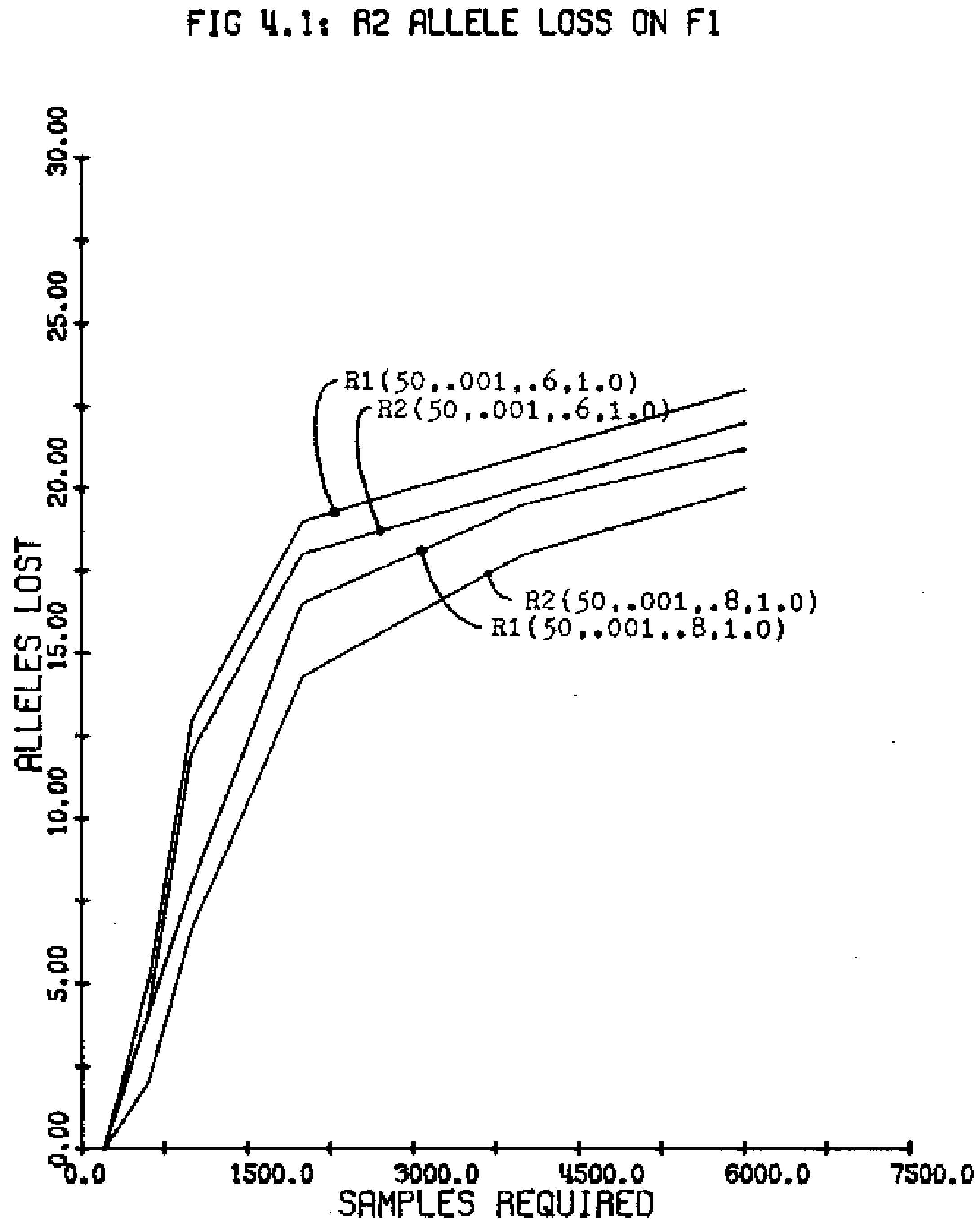


Figure 4.1: Allele loss for R2 on F1.

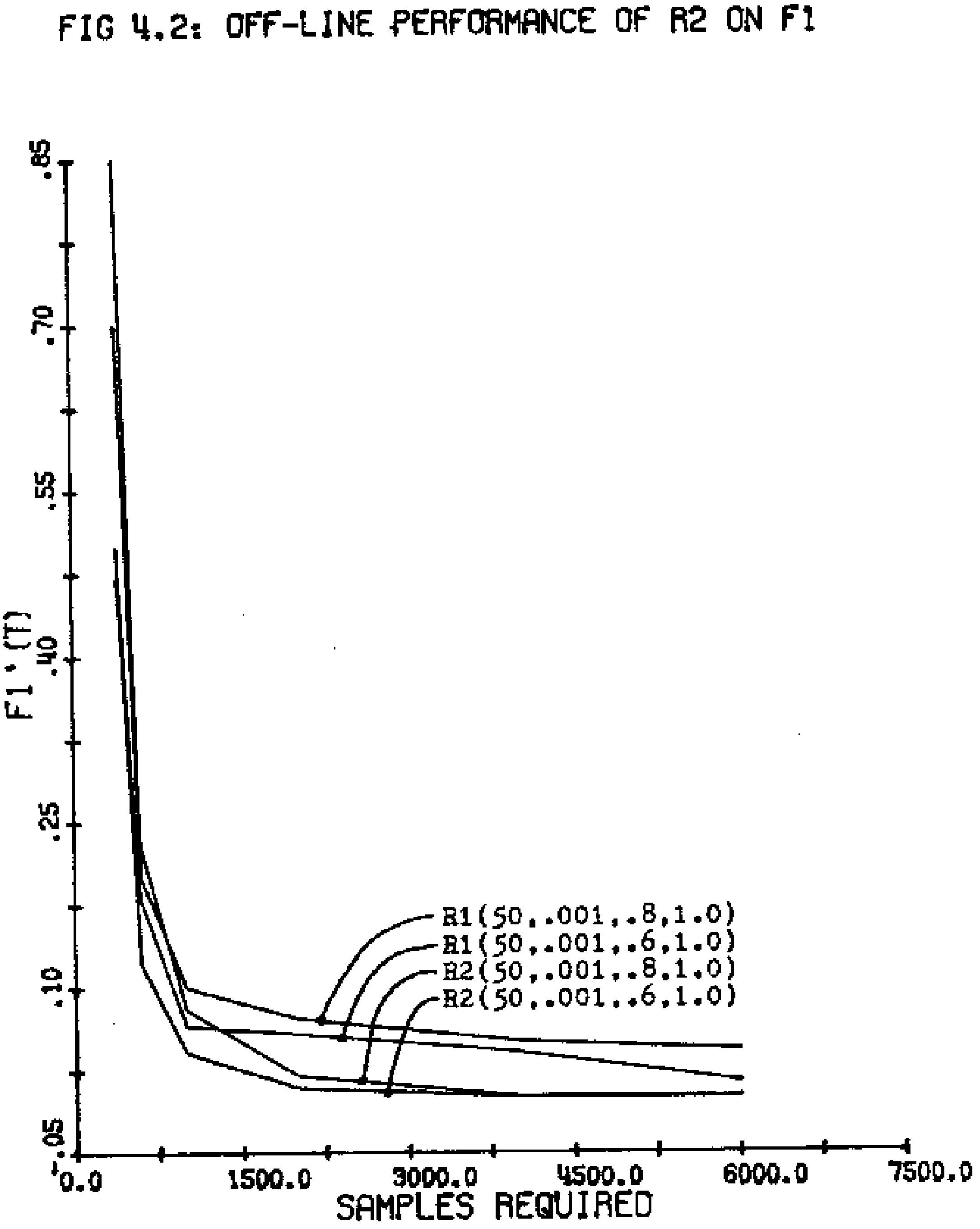


Figure 4.2: Off-line performance curves for R2 on F1.

FIG. 4.3: ON-LINE PERFORMANCE OF R2 ON F1

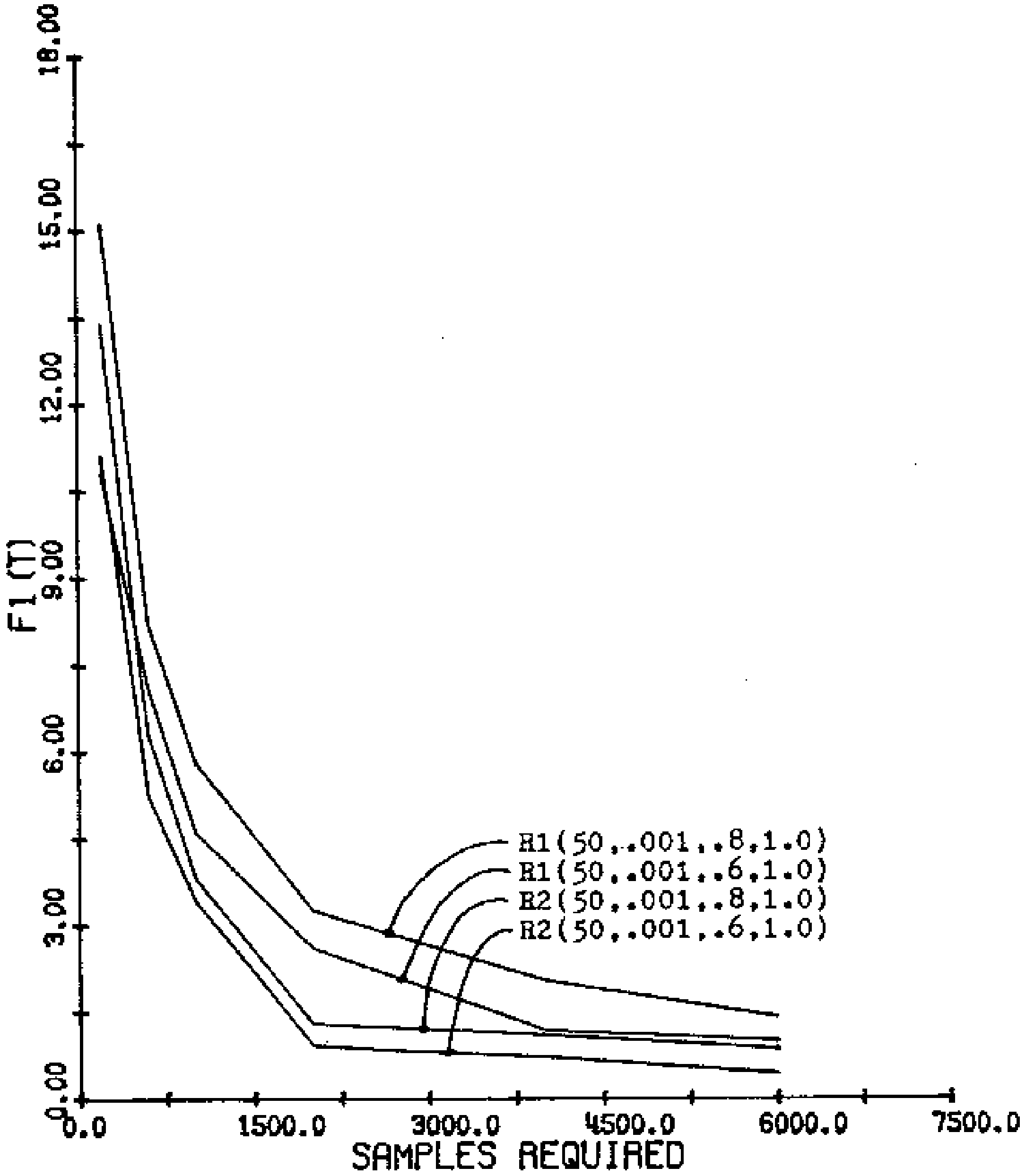


Figure 4.3: On-line performance curves for R2 on F1.

Finally, the associated performance indices for both off-line and on-line performance are tabulated below in comparison with their R1 counterparts:

T=6000	R1(.8)	R2(.8)	R1(.6)	R2(.6)
$x^*_{F1}(T)$	.199	.178	.146	.093
$x^*_{F2}(T)$	.230	.076	.310	.182
$x^*_{F3}(T)$	-26.5	-26.3	-27.2	-27.1
$x^*_{F4}(T)$	34.67	29.61	34.5	26.76
$x^*_{F5}(T)$	3.75	5.86	2.56	3.95
$X^*_E(T)$	2.47	1.88	2.06	.778

T=6000	R1(.8)	R2(.8)	R1(.6)	R2(.6)
$x_{F1}(T)$	4.27	2.88	3.65	2.71
$x_{F2}(T)$	76.8	51.73	76.7	50.46
$x_{F3}(T)$	-23.17	-22.9	-23.3	-24.02
$x_{F4}(T)$	93.2	65.8	89.9	59.92
$x_{F5}(T)$	34.1	36.2	36.2	37.49
$X_E(T)$	37.04	26.74	36.69	25.31

These results confirm our intuition about the behavior of elitist plan R2. Because of its more conservative sampling policy, on-line performance is consistently improved. Off-line performance is improved as well, but notice that the improvements come on the unimodal surfaces, particularly F4, while the performance degraded significantly on F5.

4.4 Expected Value Model R3

The problem of premature allele loss and the subsequent convergence to a non-optimal plateau which was analyzed closely in the previous chapter has still not been resolved. As we have seen, changing the various parameters of the genetic models affects both the allele loss rate and performance curves on test function F1, but gives no satisfactory solution to either. In this section we explore the possibility of resolving these problems by modifying the sampling techniques used in R1 and R2.

We begin by focusing our attention again on the two competing hyperplanes associated with a particular gene position. As we have seen, test function F1 has the characteristic that it has the same average value on both partition elements, so that neither hyperplane *theoretically* has any selective advantage over the other. However, the next generation A(t+1) is produced by taking a finite number of samples from A(t) using a selection distribution computed from *sample* means. This opens the door for stochastic side effects from two sources: the error involved in the sample means (and hence the selection probabilities), and the error involved in only taking a finite sample from A(t) using the selection distribution. The error in the sample means is, of course, a function of the population size and the variance of F1 on the associated hyperplanes, and can be resolved, as we have seen, by increasing the population size at the expense of initial performance. The Markov process simulations in the section on genetic drift modelled the second source of error which, we have seen, cannot be ignored. In R1 and R2 this sampling process is used to produce the offspring of individuals in A(t). As a consequence, the actual number of offspring produced by an individual can differ markedly from the expected number of offspring. As we saw in chapter 2, the offspring determine the number of trials allocated to a particular hyperplane in the next generation. Hence, the sampling side-effects can lead to considerable disparities between the expected and actual number of trials allocated to competing pairs of hyperplanes. This suggests that we consider redefining the sampling process used in R1 and R2 in such a way as to force the actual number of offspring to more closely approximate the expected number.

Genetic adaptive plan R3 attempts to accomplish this in the following way. The expected number of offspring, $u(a_{it})/\overline{u(t)}$, is computed and associated with each individual a_{it} in A(t) before selection begins. Each time an individual is selected as a partner for crossover, its associated offspring count is decremented by .5. Each time an individual is selected to produce an offspring without applying crossover, its associated offspring count is decremented by 1. When the offspring count falls below zero, an individual is no longer available for selection.

This modification to the selection process forces the actual number of offspring to always be less than $u(a_{1t})/\overline{u(t)} + 1$ and generally less than $u(a_{1t})/\overline{u(t)} + .5$, resulting in a leveling effect on the sampling error. If the high rate of allele loss exhibited by R1 and R2 on F1 is due in part to this sampling error, R3 should exhibit a reduced rate of allele loss and a corresponding improvement in performance.

In order to evaluate these hypotheses, the following three members from the family of plans defined by R3 were chosen for evaluation on E:

R3(50,.001,1.0,1.0)
R3(50,.001, .8,1.0)
R3(50,.001, .6,1.0)

Figures 4.4 - 4.6 compare the behavior of R3(50,.001,.6,1.0) on F1 with its corresponding R1 and R2 counterparts. Figure 4.4 illustrates that, as we had hoped, the allele loss rate is considerably reduced with the modified sampling technique. Figures 4.5 and 4.6 compare the performance curves of R1, R2, and R3 for test function F1. Notice that R3 performed significantly better than R1 on F1, but not quite as well as R2.

Finally, the associated performance indices for both off-line and on-line performance of R3 on E are tabulated below in comparison with R1 and R2:

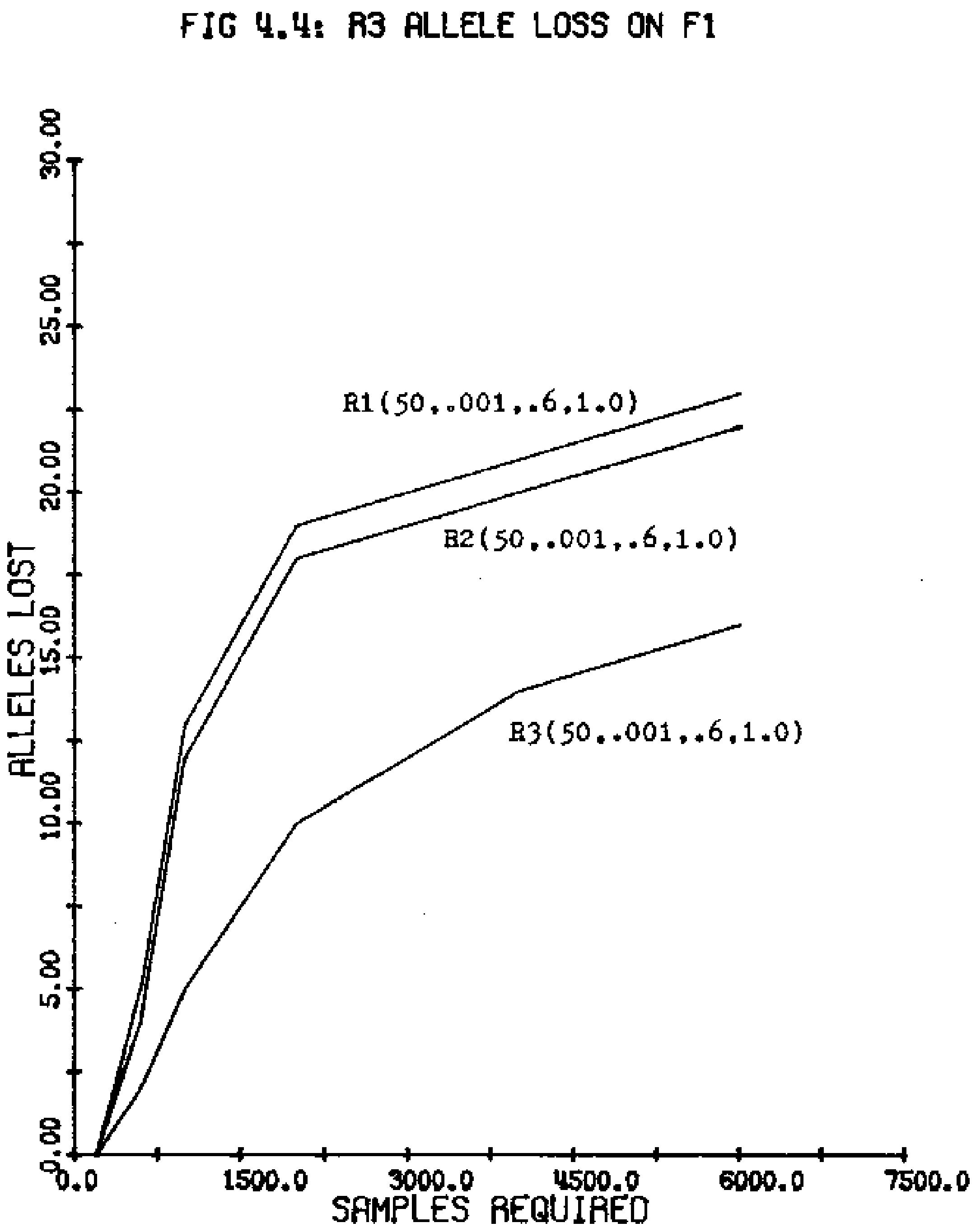


Figure 4.4: Allele loss for R3 on F1.

FIG 4.5: OFF-LINE PERFORMANCE OF R3 ON F1

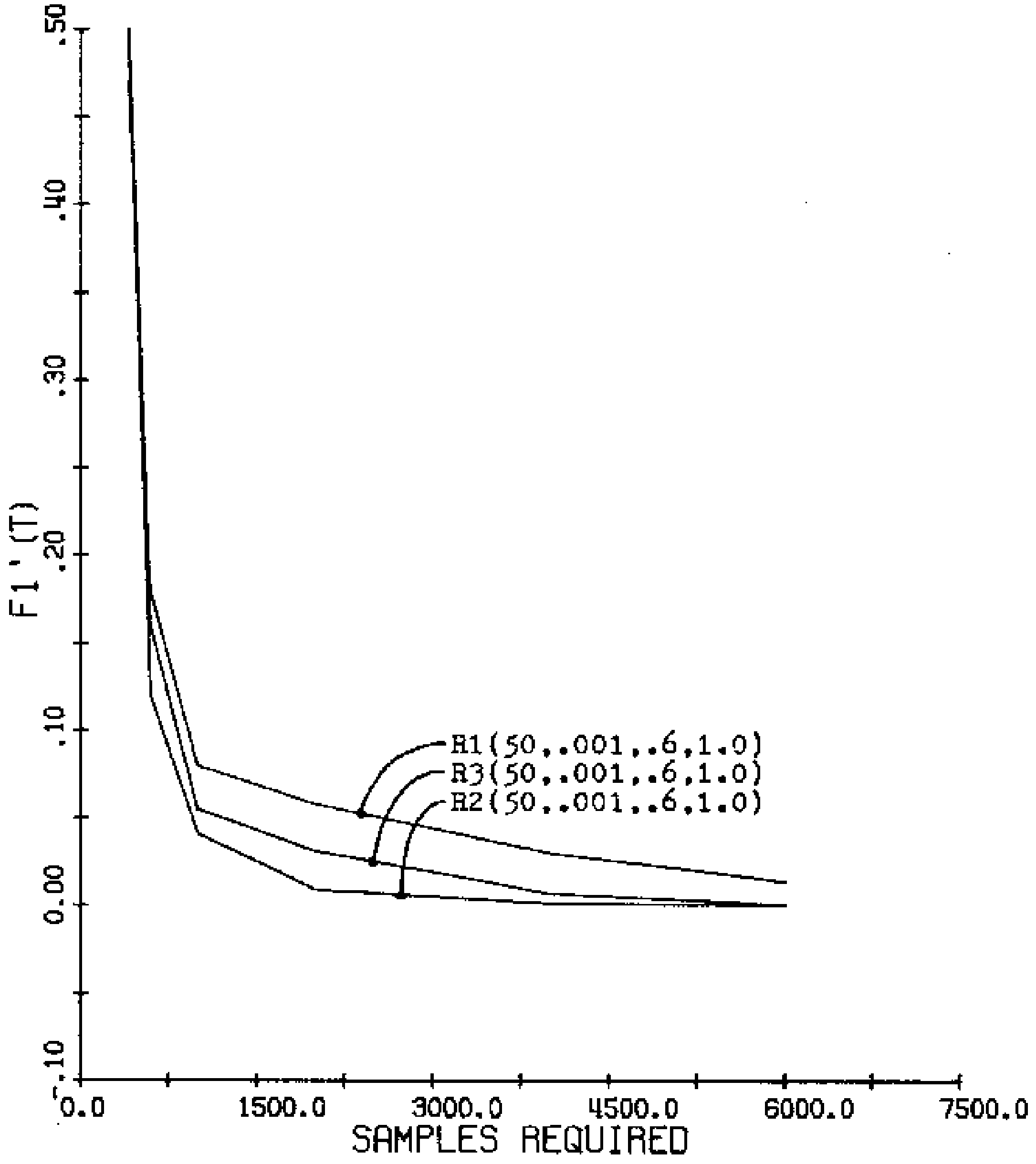


Figure 4.5: Off-line performance curve for R3 on F1.

FIG. 4.6: ON-LINE PERFORMANCE OF R3 ON F1

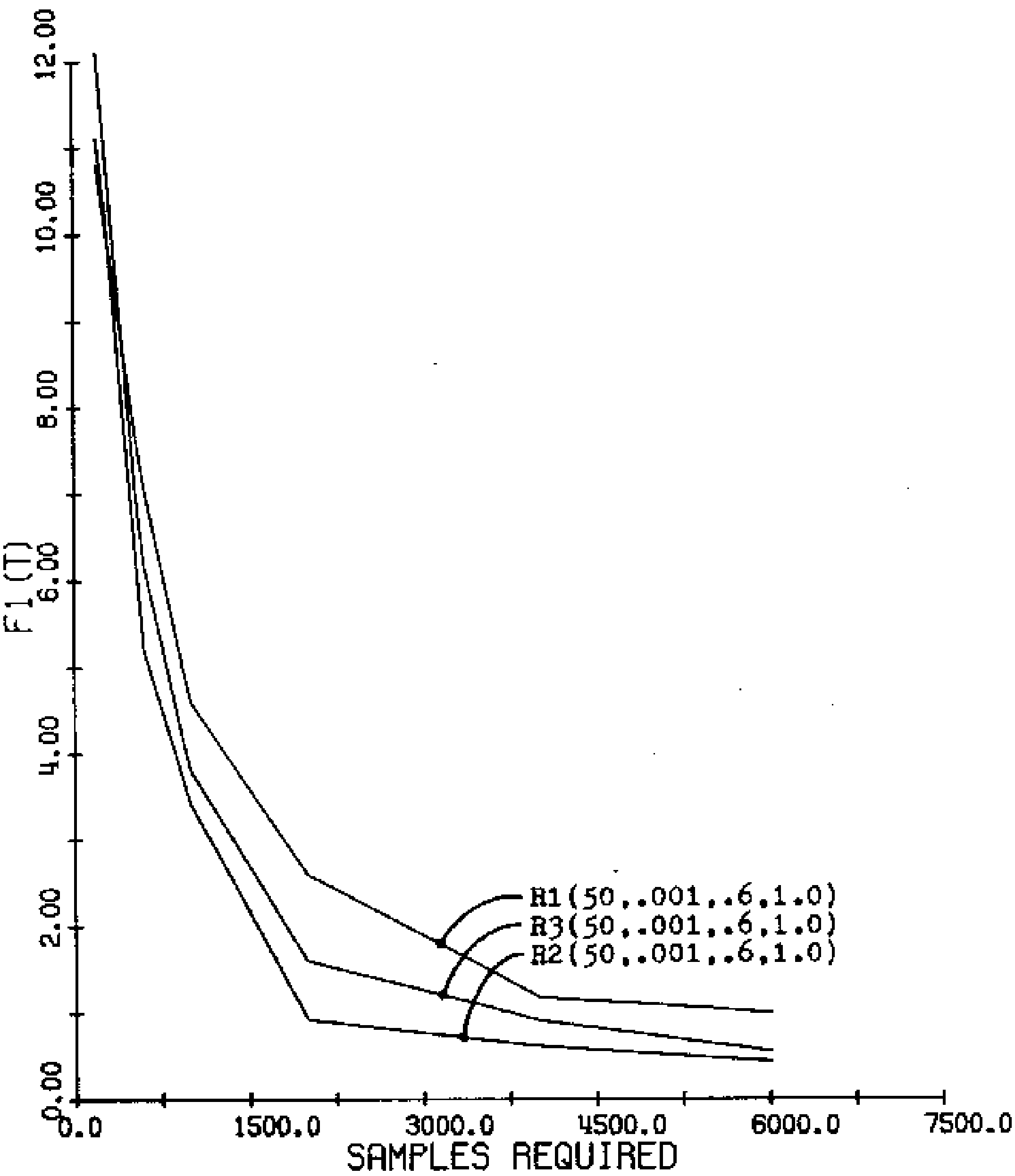


Figure 4.6: On-line performance curve for R3 on F1.

T=6000	R3(1.0)	R3(.8)	R3(.6)	R2(.6)	R1(.6)
$x^*_{F1}(T)$	.410	.130	.166	.093	.146
$x^*_{F2}(T)$	.428	.213	.364	.182	.310
$x^*_{F3}(T)$	-27.3	-27.1	-27.2	-27.1	-27.2
$x^*_{F4}(T)$	21.17	20.22	21.07	26.76	34.5
$x^*_{F5}(T)$	4.31	2.86	3.20	3.95	2.56
$X^*_E(T)$	.196	-.735	-.483	.778	2.06

T=6000	R3(1.0)	R3(.8)	R3(.6)	R2(.6)	R1(.6)
$x_{F1}(T)$	3.53	2.41	3.42	2.71	3.65
$x_{F2}(T)$	65.65	46.64	55.15	50.46	76.7
$x_{F3}(T)$	-24.6	-24.87	-24.94	-24.02	-23.3
$x_{F4}(T)$	48.28	49.08	44.18	59.92	89.9
$x_{F5}(T)$	36.46	32.29	37.31	37.49	36.2
$X_E(T)$	25.46	21.68	22.65	25.31	36.69

These results yield two interesting observations. First, note that R3(.8) performed slightly better on E than R3(.6), suggesting that, by reducing the sampling error, a higher sampling rate can be supported. Secondly, note that, although R2 outperformed R3 on F1, R3 showed an overall improvement in robustness on E. Both of these observations confirm our intuition about the behavior of the elitist and expected value models.

4.5 Elitist Expected Value Model R4

At this point in the analysis of genetic adaptive plans, it is difficult to resist combining the two previous models to produce an expected value model with an elitist policy. The motivation here is to increase our confidence in the observations and inferences made so far about the behavior of genetic plans, rather than providing new insights. If our analysis of the preceding sections is correct, we should expect that adding an elitist policy to R3 should improve its performance on the unimodal surfaces at the expense of multimodal performance.

In order to evaluate this hypothesis, two members of R4 were chosen for analysis on E:

R4(50,.001,.8,1.0)
R4(50,.001,.6,1.0)

Figures 4.7 - 4.9 compare the behavior of R4(.6) on test function F1 with its R2 and R3 counterparts. Figure 4.7 illustrates again that adding an elitist policy reduces slightly the allele loss rate on F1. Figures 4.8 and 4.9 illustrate the improved off-line and on-line performance curves generated by R4.

Finally, the associated performance indices for both off-line and on-line performance of R4 on E are tabulated below in comparison with R3 and R2:

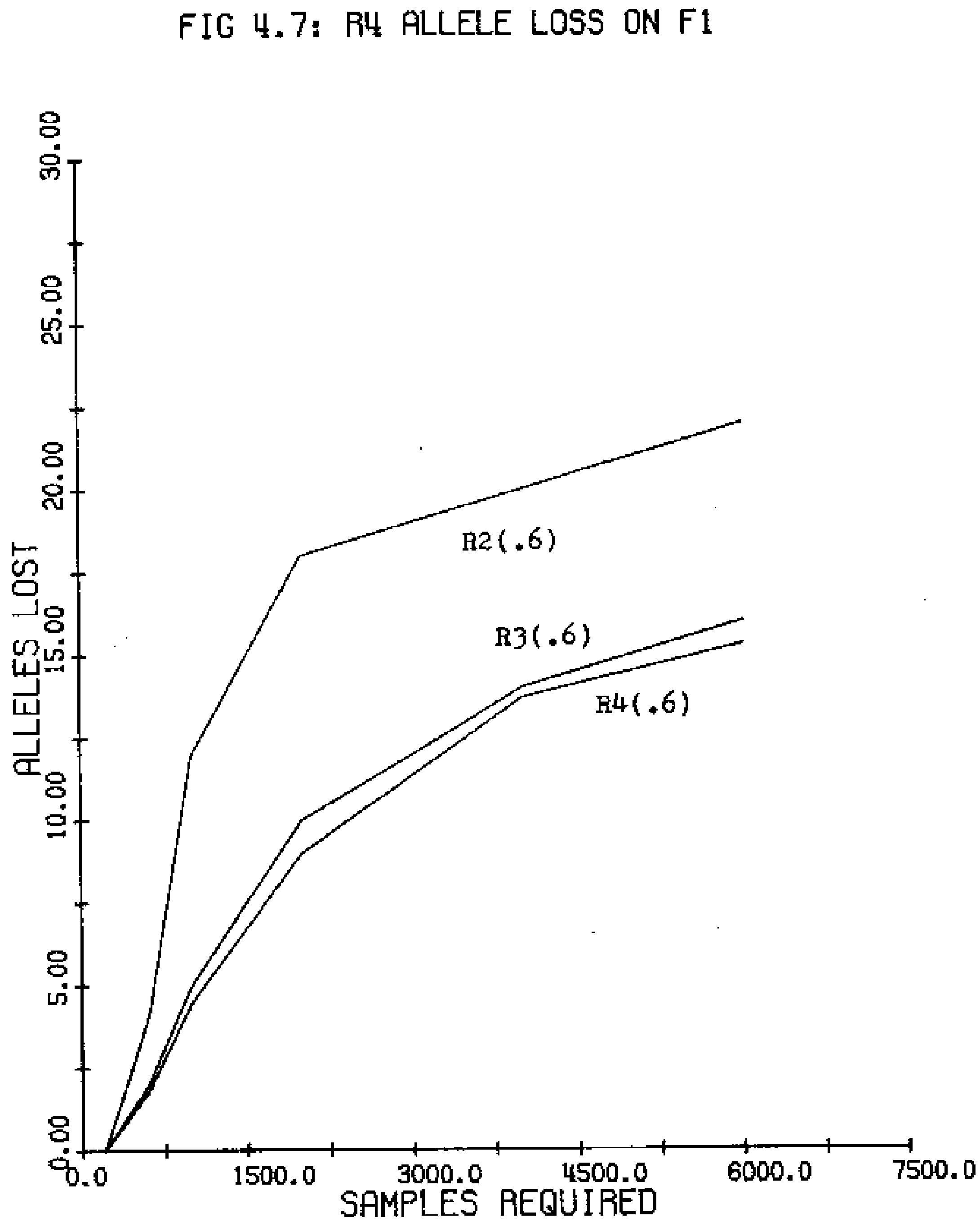


Figure 4.7: Allele loss for R4 on F1.

FIG 4.8: OFF-LINE PERFORMANCE OF R4 ON F1

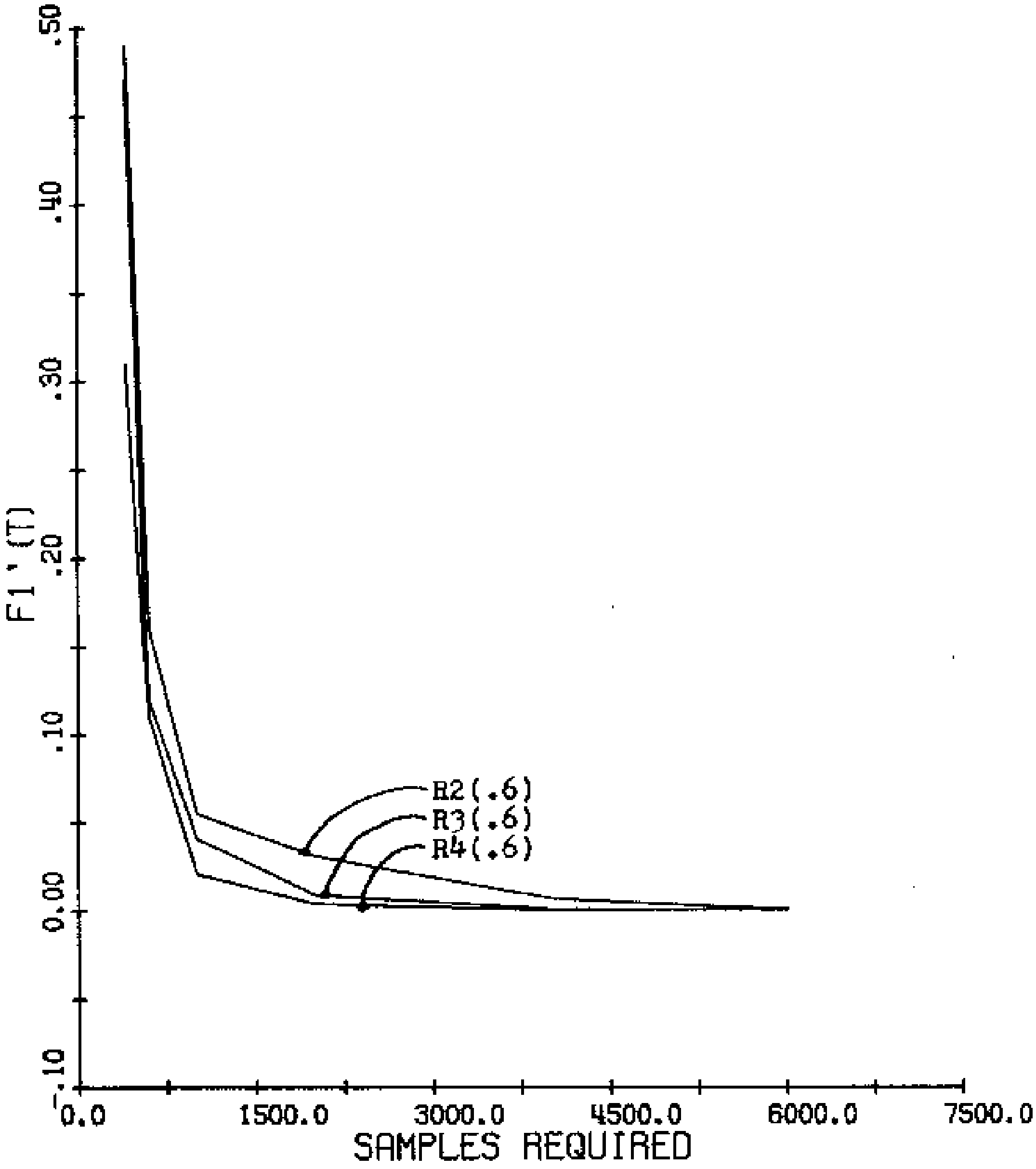


Figure 4.8: Off-line performance curve for R4 on F1.

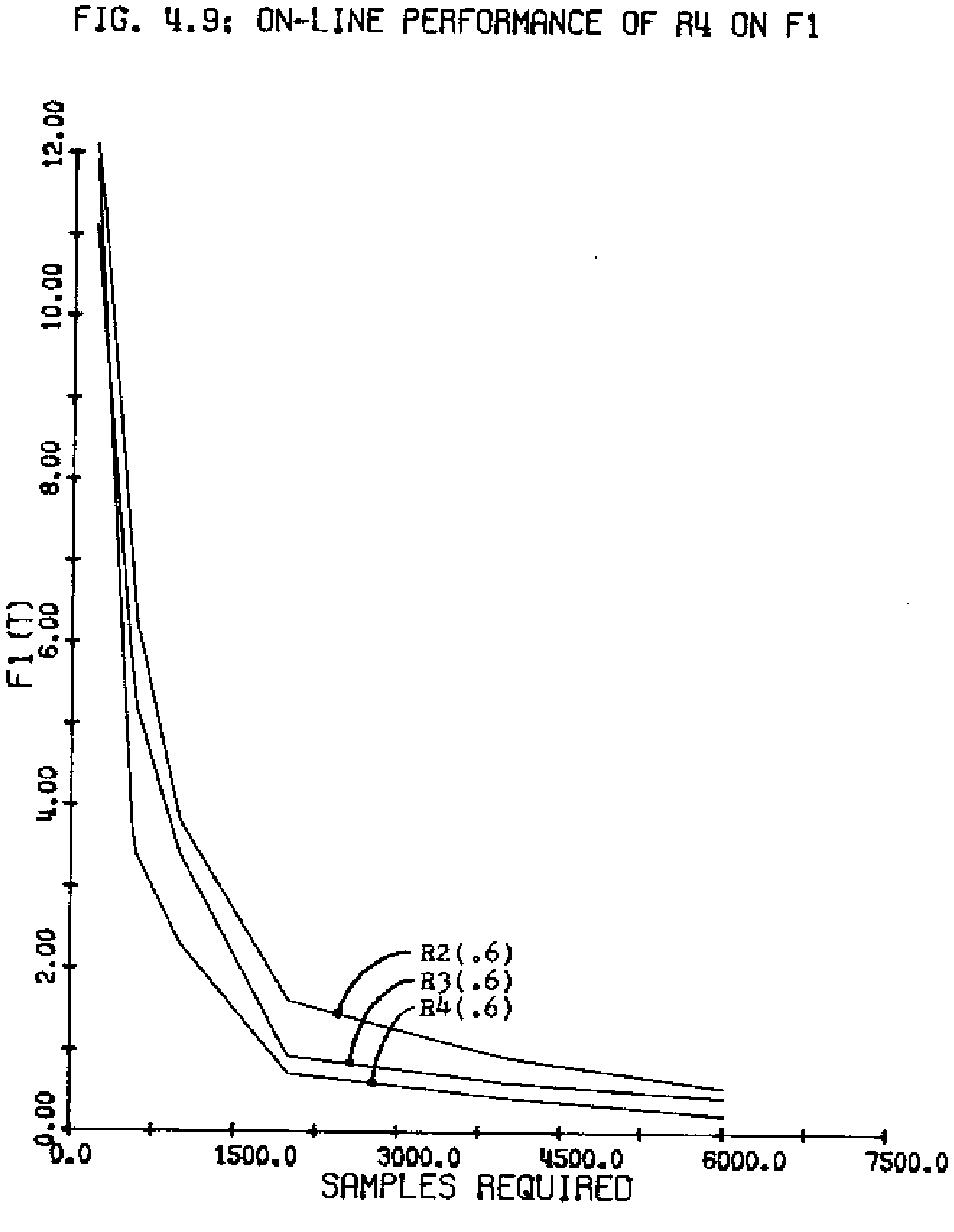


Figure 4.9: On-line performance curve for R4 on F1.

T=6000	R4(.8)	R4(.6)	R3(.8)	R3(.6)	R2(.6)
$x^*_{F1}(T)$	.097	.113	.130	.166	.093
$x^*_{F2}(T)$	.201	.221	.213	.364	.182
$x^*_{F3}(T)$	-27.5	-28.2	-27.1	-27.2	-27.1
$x^*_{F4}(T)$	18.21	17.62	20.22	21.07	26.76
$x^*_{F5}(T)$	2.98	3.34	2.86	3.20	3.95
$X_E(T)$	-1.20	-1.38	-.735	-.483	.778

T=6000	R4(.8)	R4(.6)	R3(.8)	R3(.6)	R2(.6)
$x_{F1}(T)$	2.41	2.32	2.41	3.42	2.71
$x_{F2}(T)$	35.46	34.76	46.64	55.15	50.46
$x_{F3}(T)$	-25.35	-26.49	-24.87	-24.94	-24.02
$x_{F4}(T)$	42.53	40.73	49.08	44.18	59.92
$x_{F5}(T)$	33.59	34.34	32.29	35.31	37.49
$X_E(T)$	17.73	17.12	21.68	22.65	25.31

These results suggest that our intuition about the effects of an elitist policy on the behavior of these genetic plans is correct. The performance on the unimodal surfaces (F1-F4) is considerably improved at the expense of performance on the difficult multimodal surface F5. Notice however that performance on F5 was only slightly affected (an observation which will be explored more fully later) and, as a consequence, R4 generated the best overall performance we have seen on E.

It is worth considering at this point whether the performance generated by R4(50,.001,.6,1.0) is about the best R4 can do on E. We have been evaluating these particular parameter settings to provide straightforward comparisons with the preceding models. However, the changes we have made to the genetic algorithm may also affect the choice of the parameter settings. To answer this question, the performance of the following members of R4 was evaluated on E:

```
R4( 50,.001,.8,1.0)
R4( 50,.001,.6,1.0)
R4( 50,.001,.4,1.0)
R4( 50,.01 ,.6,1.0)
R4( 50,.05 ,.6,1.0)
R4( 50,.001,.6, .8)
R4( 50,.001,.6, .6)
R4(100,.001,.6,1.0)
```

Tables 4.2a and 4.2b compare the off-line and on-line performance indices for each of the parameter settings. The first three members evaluated differ only in their crossover rates and they suggest that a crossover rate of .6 is still a reasonable choice. Notice, however, that because of the overall improvement in performance, R4 is considerably less sensitive than R1 to changes in the crossover rate. The fourth and fifth members illustrate the effects of an increased mutation rate. Here the results differed from before. Increasing the mutation rate degraded <u>both</u> the off-line and on-line performance of R1. This observation will be explored in more detail later. The sixth and seventh members illus-

T=6000	N =50 P_m=.001 P_c=.8 G =1.0	N =50 P_m=.001 P_c=.6 G =1.0	N =50 P_m=.01 P_c=.4 G =1.0	N =50 P_m=.01 P_c=.6 G =1.0	N =50 P_m=.05 P_c=.6 G =1.0	N =50 P_m=.001 P_c=.6 G =.8	N =50 P_m=.001 P_c=.6 G =.6	N =100 P_m=.001 P_c=.6 G =1.0
$x_{F1}^*(T)$	.097	.113	.134	.068	.128	.213	.112	.103
$x_{F2}^*(T)$	.201	.221	.205	.259	.240	.193	.410	.148
$x_{F3}^*(T)$	-27.5	-28.2	-28.5	-27.6	-26.1	-27.9	-28.2	-27.5
$x_{F4}^*(T)$	18.21	17.62	16.24	21.81	35.98	21.66	17.1	28.5
$x_{F5}^*(T)$	2.98	3.34	8.53	3.98	3.51	2.81	11.5	5.39
$X_E^*(T)$	-1.20	-1.38	-.678	-.297	2.75	-.605	.184	1.33

Table 4.2a: Off-line performance of R4 on E

T=6000	N =50 P_m=.001 P_c=.8 G =1.0	N =50 P_m=.001 P_c=.6 G =1.0	N =50 P_m=.01 P_c=.4 G =1.0	N =50 P_m=.01 P_c=.6 G =1.0	N =50 P_m=.05 P_c=.6 G =1.0	N =50 P_m=.001 P_c=.6 G =.8	N =50 P_m=.001 P_c=.6 G =.6	N =100 P_m=.001 P_c=.6 G =1.0
$x_{F1}(T)$	2.41	2.32	2.30	5.88	15.83	2.35	2.42	4.63
$x_{F2}(T)$	35.46	34.76	40.02	105.0	221.72	42.35	41.44	62.9
$x_{F3}(T)$	-25.35	-26.49	-27.01	-21.5	-14.04	-26.1	-26.2	-24.05
$x_{F4}(T)$	42.53	40.73	39.03	96.38	182.0	47.14	49.2	68.38
$x_{F5}(T)$	33.59	34.34	35.06	91.46	207.1	36.76	41.28	43.82
$X_E(T)$	17.73	17.12	17.88	55.45	122.5	20.50	21.63	31.14

Table 4.2b: On-line performance of R4 on E

trate again the negative effect on performance generated by reducing the generation gap. And the last member evaluated illustrates again that increasing the population size degrades performance over the interval of observation.

In summary, then, these results suggest that the performance generated by R4(50,.001,.6,1.0) is about the best R4 can do on E. It is important, however, to emphasize the extent of the improvements in performance we have achieved by moving to type R4 genetic plans. Recall from the evaluation in appendix C that, if on-line performance is desired, one simply cannot afford to use random search. Even the simplest genetic plan generates significantly better on-line performance. However, when measuring off-line performance, we saw that random search gave R1 considerably stiffer competition and produced in several cases better performing individuals over the interval of observation. To illustrate the improved performance of R4, figures 4.10 - 4.14 compare the off-line performance curves of random search on each of the test functions in E with the best curves generated by R1 and R4 as well as the (unattainable) optimal off-line curve given by:

$$f^*(t) = MIN(f) \text{ , } t=1,\ldots,T$$

On test functions F1 and F2 plan R4 located the minimum without difficulty within the interval of observation. On the larger search spaces associated with F3 and F4, the minimum was not found within 6000 trials. But notice

FIG. 4.10: OFF-LINE PERFORMANCE ON F1

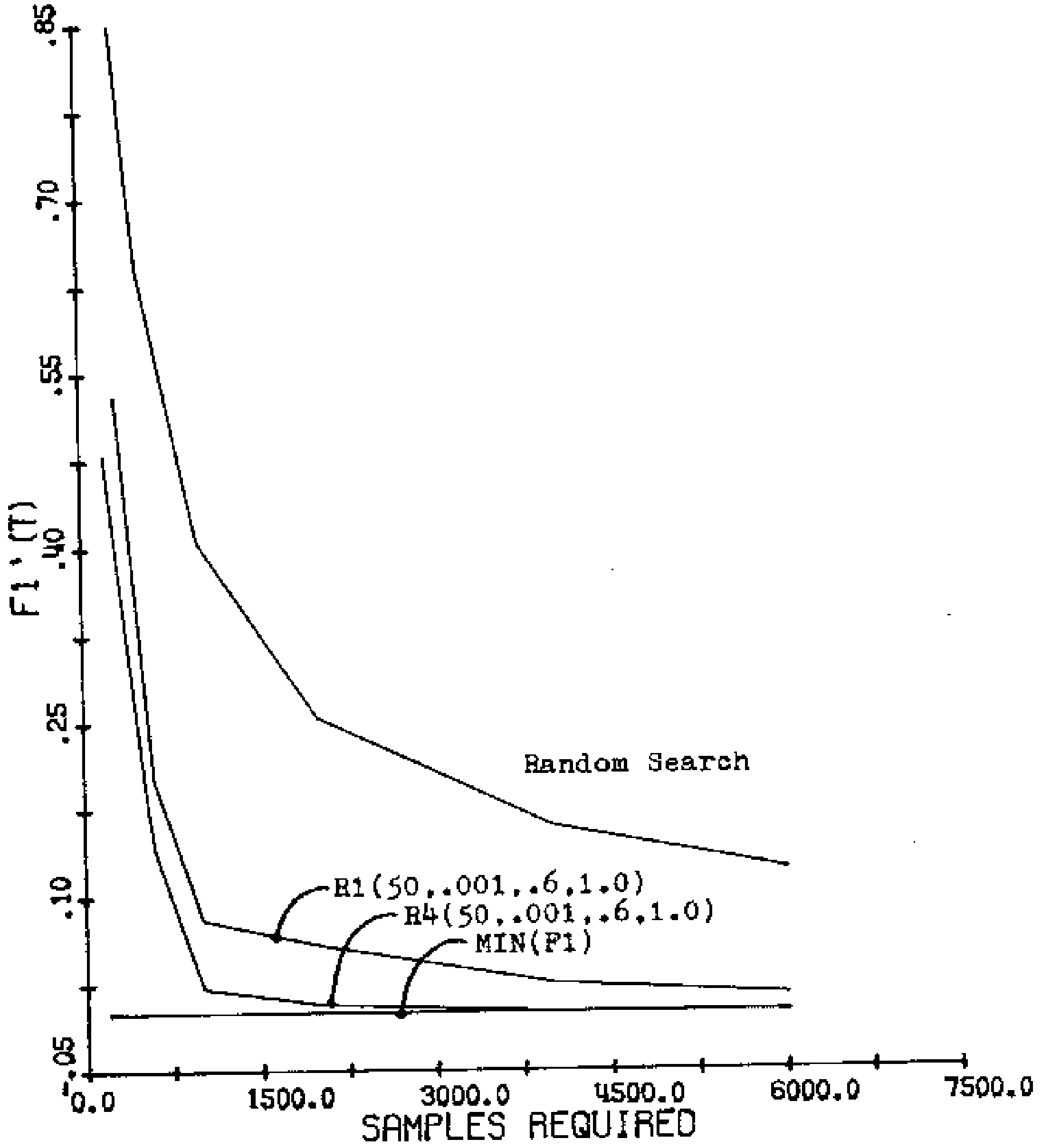


Figure 4.10: A comparison of off-line performance curves generated on F1.

FIG. 4.11: OFF-LINE PERFORMANCE ON F2

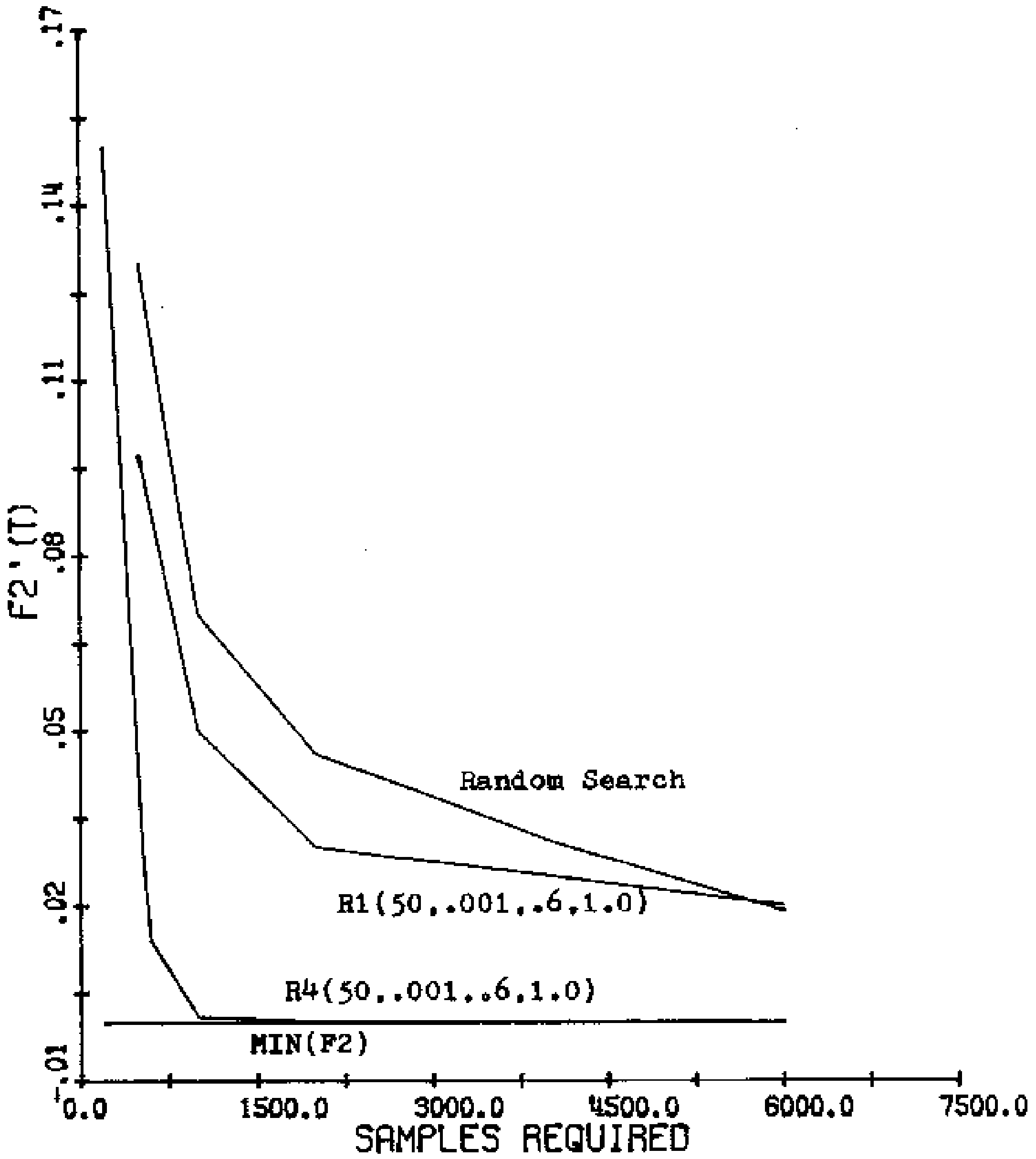


Figure 4.11: A comparison of off-line performance curves generated on F2.

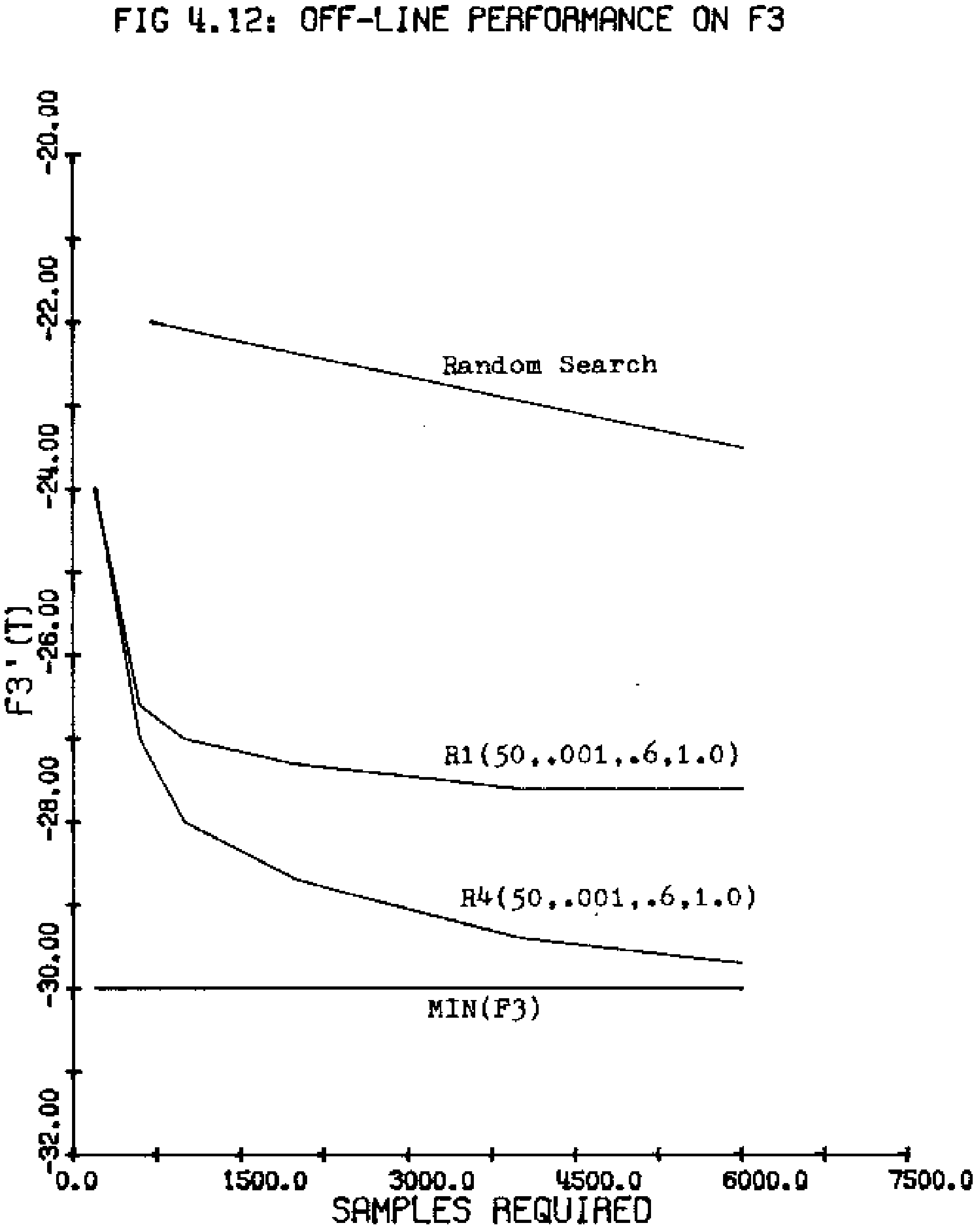


Figure 4.12: A comparison of off-line performance curves generated on F3.

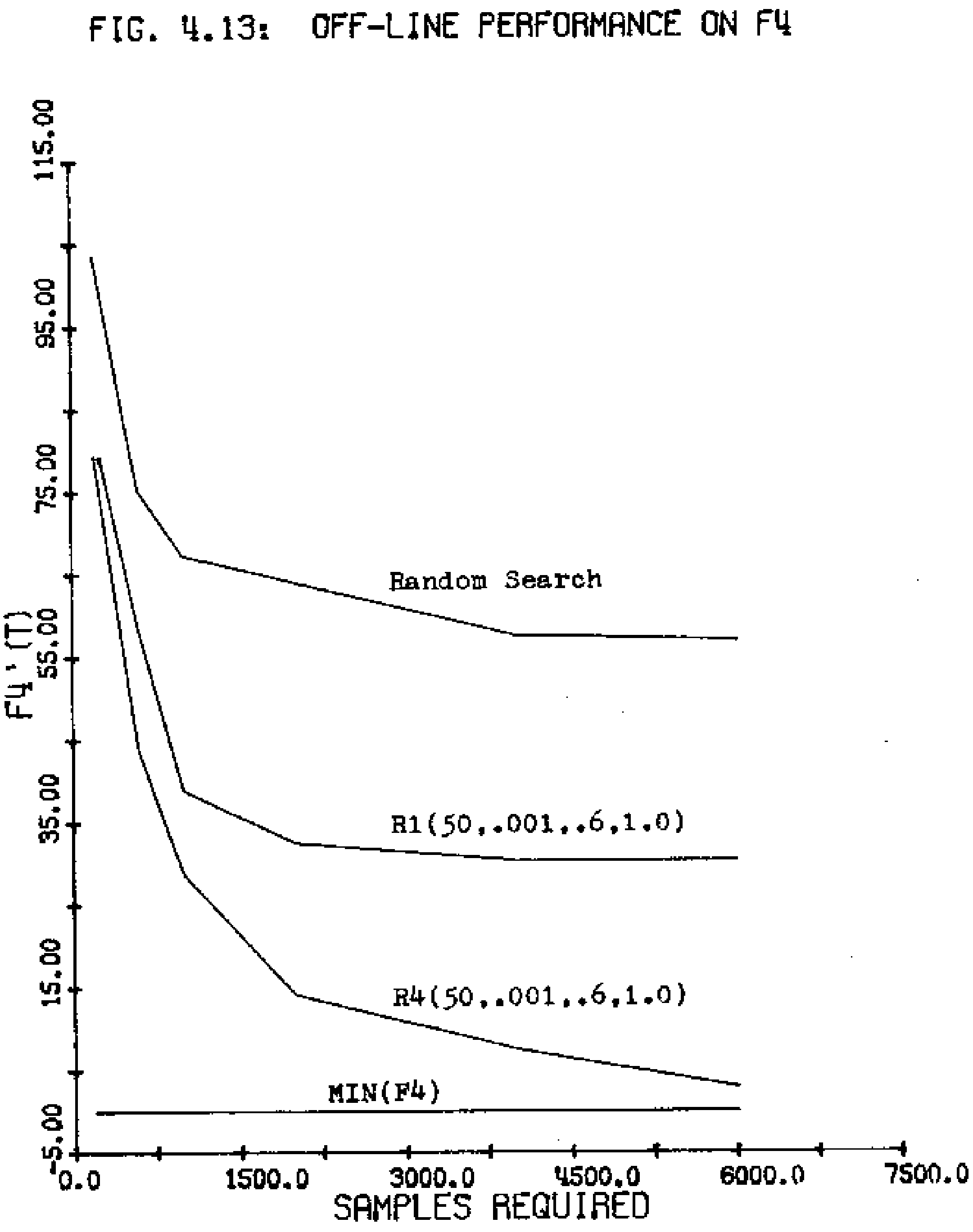


Figure 4.13: A comparison of off-line performance curves generated on F4.

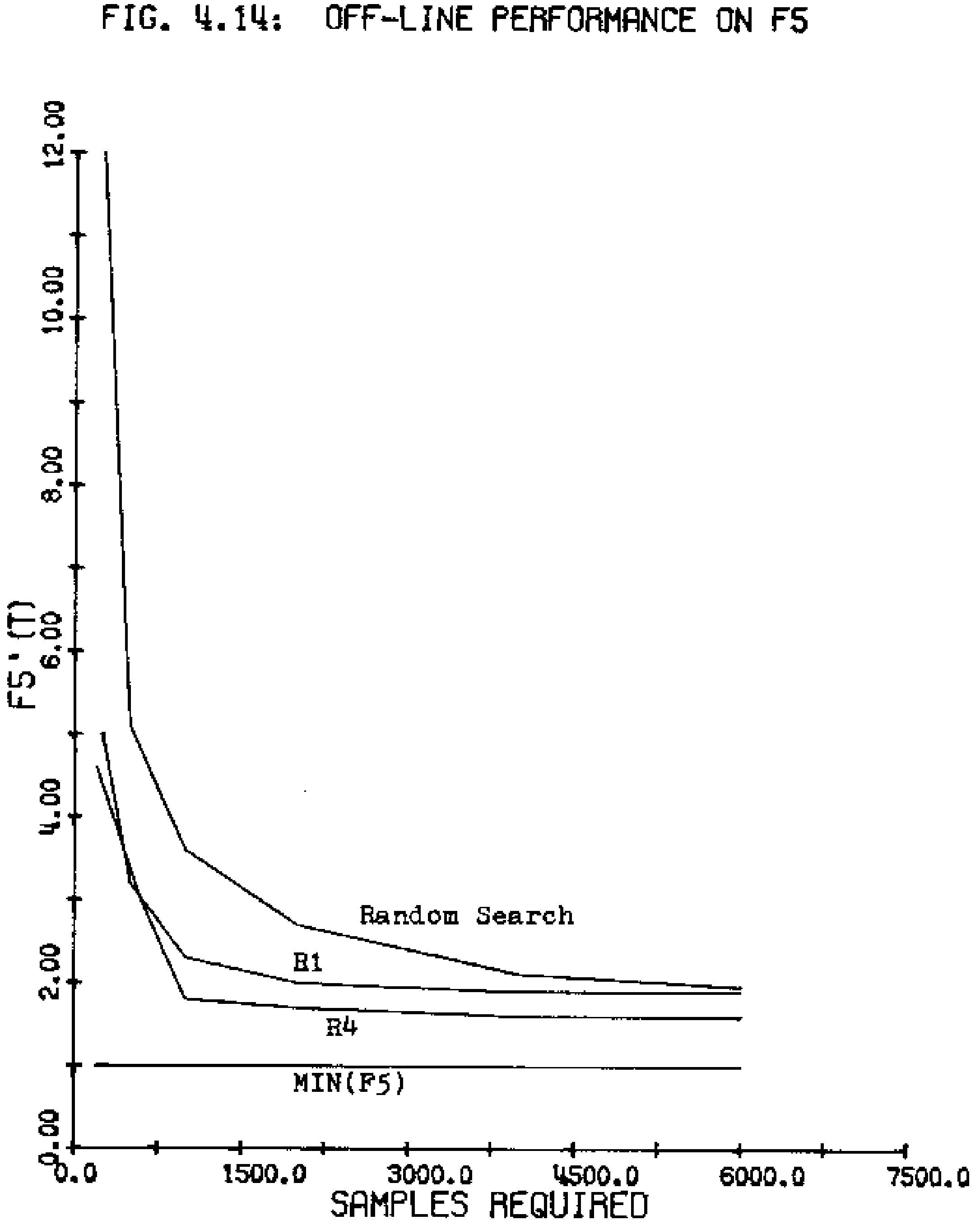


Figure 4.14: A comparison of off-line performance curves generated on F5.

that the problem of pre-mature convergence to a non-optimal plateau is no longer evident. Consistent progress is made over the entire interval of observation. Only on the difficult multimodal surface defined by F5 do we still see convergence to a non-optimal plateau. It is this problem which will be addressed in the next sections.

4.6 Improving the Performance of R4 on F5

As noted in the previous section, considerable progress has been made in improving the performance of finite genetic models on E. By modifying the sampling technique used in R1 so that the actual number of offspring more closely approximate the expected number, the allele loss rate due to stochastic side-effects has been reduced considerably with a corresponding improvement in performance. In addition, by adding an elitist policy to R3, significant improvement on the unimodal surfaces was observed. Figure 4.15 summarizes the remaining fly in the ointment: the performance of genetic plans on F5. By going to plan R3, the premature convergence generated by R1 was replaced by an off-line performance curve which made slow but steady progress over the interval of observation. Adding the elitist policy to R3 improved initial off-line performance, but once again we observe convergence to a non-optimal plateau. This also suggests an explanation to the

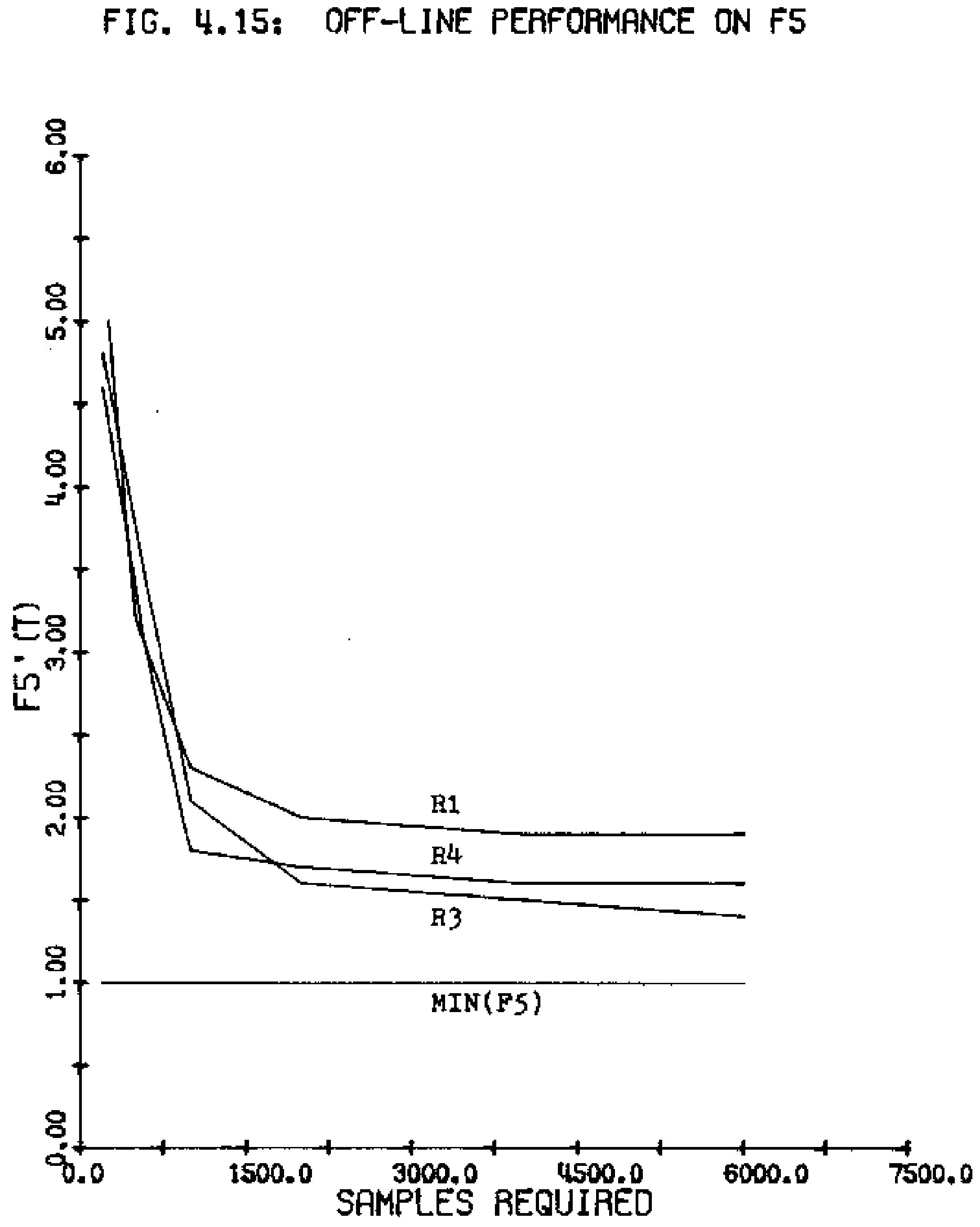
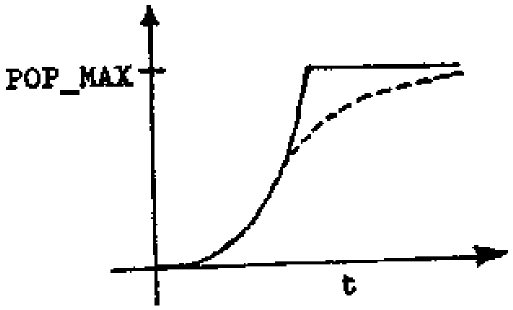


Figure 4.15: Off-line performance curves for genetic plans on F5.

observation noted in the previous section that adding an elitist policy to R3 did not degrade the performance indices for R4 on F5 as much as had been expected. As figure 4.15 illustrates, the average values (performance indices) of the off-line curves generated by R3 and R4 are very nearly the same. Over a longer time interval, the negative effects of the premature convergence with R4 would have been more clearly seen. In this section we address the problem of improving the global search properties of R4 without, hopefully, having to give up the improved local performance.

We begin by considering this problem in terms of the hyperplane analysis introduced in chapter 2. Recall that genetic plans have the property that the best of competing hyperplanes were allocated an exponentially increasing number of trials relative to their competitors. This property was shown to be a consequence of the fact that the number of instances of a particular hyperplane in A(t) changed over time in proportion to the hyperplane's performance relative to its competitors. If a particular hyperplane outperforms its competitors for a relatively small number of generations, we saw that the number of instances of that hyperplane in A(t) increased exponentially to a point of complete dominance. When such a hyperplane is in fact the best, this is precisely what we want to happen with the corresponding allele loss effecting a reduction in the search space.

However, we are working with finite genetic plans which maintain reasonably small populations which evaluate hyperplane performance via sample means based on a relatively small number of samples. As a consequence, it is not difficult to imagine that some of the premature allele loss observed may be the result of non-optimal hyperplanes appearing to be the best for a sufficiently long time to effect a reduction in the search space. More to the point, it is easy to imagine that on the difficult surface defined by F5 a hyperplane associated with a relatively good local optimum could quickly dominate A(t) and cause the observed premature convergence. These observations suggest that the premature allele loss rate and the performance of genetic plans on multimodal functions could be improved by making it more difficult for hyperplanes to dominate A(t). It should be clear, however, that overall performance on E may be seriously degraded unless a solution is chosen carefully, since it is precisely this exponential increase in trials which generates the kind of performance exhibited by R4 on the unimodal surfaces. What we seek is a solution which permits exponential exploitation of the observed best without allowing them to readily dominate a finite population. This suggests that, rather than allow exponential growth until total dominance occurs, genetic plans should admit "controlled" growth in the form of an "S" curve as illustrated below:



Such an approach permits initial exponential exploitation of hyperplanes for rapid performance improvements, while at the same time making it considerably more difficult for a hyperplane to completely dominate A(t).

The difficulty, of course, is in finding a reasonable implementation for this conceptually simple solution. Consider for a moment the alternatives within the R4 framework. Increasing the population size serves both to improve the sample means and increase the time required for a hyperplane to dominate A(t). As we have seen, however, this results in a significant degradation in performance over the interval of observation. Increasing the generation gap reduces the rate at which decisions are made and hence the rate of dominance. However, within the genetic context, a value larger than 1.0 makes no sense. Increasing the crossover rate has the opposite effect from that desired. As we saw in chapter 2, crossover becomes increasingly less likely to interfere with hyperplane growth as it begins to dominate A(t). Increasing the mutation rate seems to be the only possible solution within the R4 framework. As the number of instances of a particular hyperplane begin to dominate A(t), the number of their offspring expected to undergo mutation increases and effects a reduction in the hyperplane's growth rate.

In order to explore the aspects of such a solution, the performance of R4 was evaluated on F5 with mutation rates of .001, .005, .01, and .05, respectively. Figure 4.16 illustrates clearly the effect that increasing the mutation rate has on the off-line performance of R4 on F5. As the mutation rate increases, the shape of the off-line performance curve changes to reflect less dramatic initial performance and more uniform progress over the entire interval of observation. Note that R4(.01) very nearly converges to the minimum within 6000 trials. However, its initial performance is less impressive than R4(.001). This suggests an explanation to the observation noted in the previous section that, unlike R1, the performance of R4 on E was actually degrading slightly by increasing the mutation rate from .001 to .01. With R1, increasing the mutation rate served to reduce its high rate of allele loss and improve performance. However, with R4's reduced allele loss rate and improved local performance, increasing the mutation rate generated an improvement in longer-term performance at the expense of initial performance. Had we evaluated R4 over a longer time interval, the long-term improvements would have been more clearly visible.

Finally, figure 4.17 illustrates clearly what we have

FIG. 4.16: OFF-LINE PERFORMANCE OF R4 ON F5

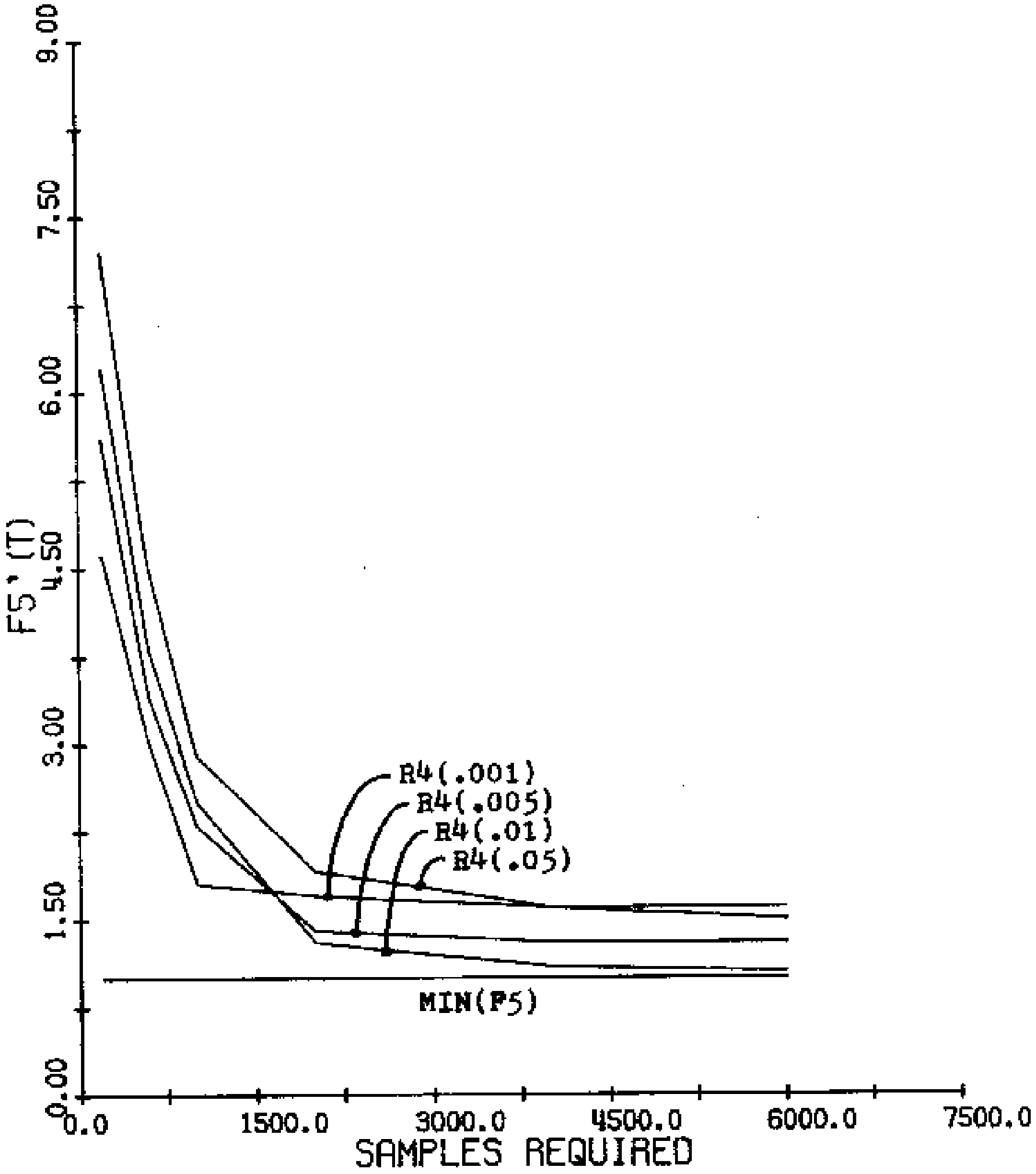


Figure 4.16: Off-line performance for R4 on F5 as a function of mutation rate.

FIG. 4.17: ON-LINE PERFORMANCE OF R4 ON F5

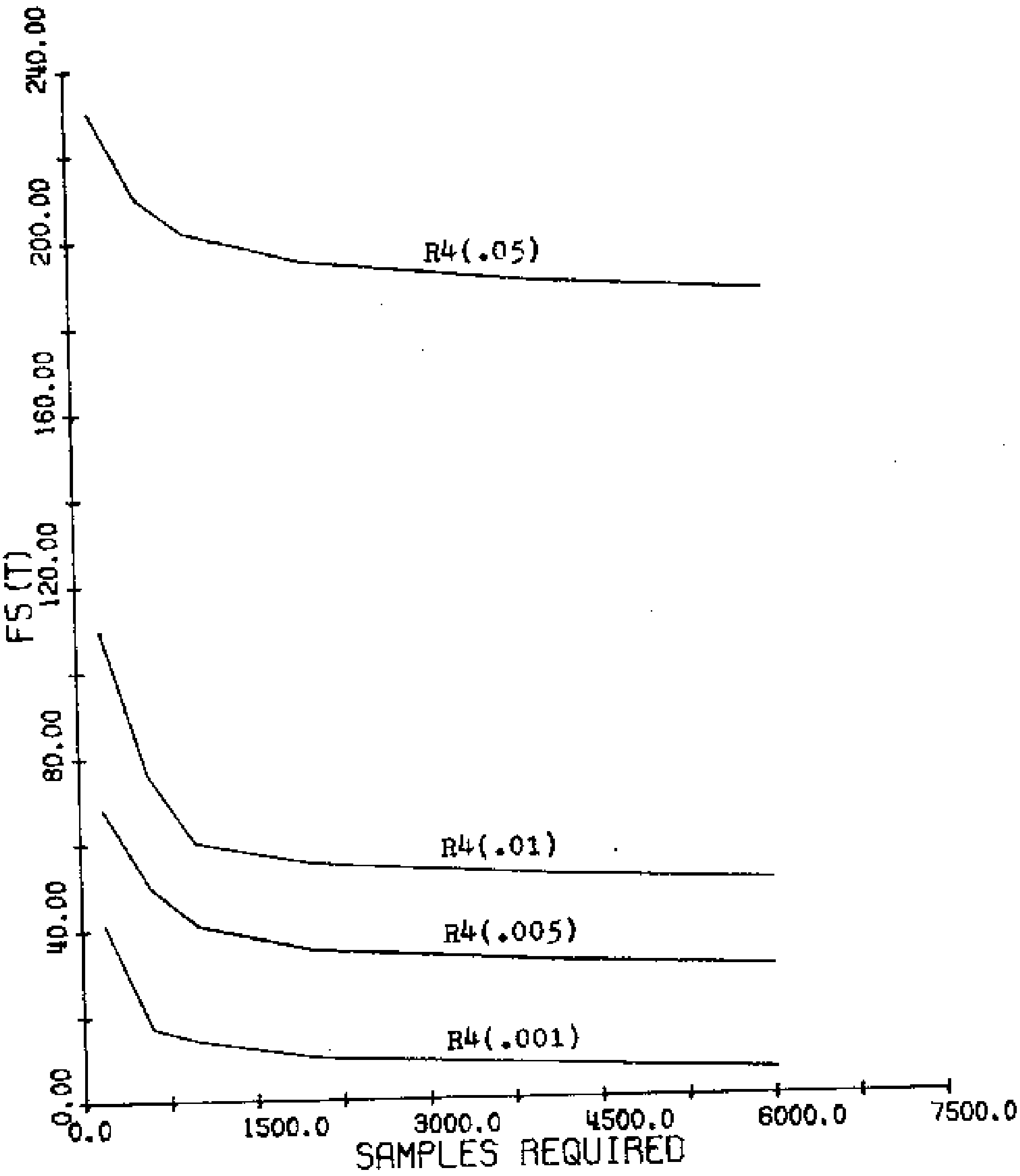


Figure 4.17: On-line performance for R4 on F5 as a function of mutation rate.

seen before. If on-line performance is required, any increase in the mutation rate seriously degrades performance.

4.7 Crowding Factor Model R5

Holland has suggested that the kind of controlled growth we seek in finite genetic models occurs in nature as a consequence of crowding. That is, as more and more like individuals dominate an environmental niche, the competition for limited resources increases rapidly resulting in lower life expectancies and birth rate. In this section we consider the effects of including such a feature in genetic adaptive plans as an alternate to increasing the mutation rate in order to improve performance on multimodal surfaces.

If we think of the genetic plans in terms of the overlapping generation models, connections between the natural and artificial systems are more intuitive. In particular, consider a model in which only a few offspring are produced each generation (e.g. G=.1). Plans of this sort produce A(t+1) as follows:

- produce G*N offspring using selection and the genetic operators
- using a uniform distribution on A(t), insert the G*N offspring into A(t) by selecting G*N individuals to "die".

Stated in this form, the concept of life expectancy is more clearly defined for these artificial systems. And, as we noted in chapter 3, the expected number of offspring of an individual is directly related to the number of generations it survives. What we seek is a method for reducing the life expectancy of individuals which are instances of a hyperplane rapidly dominating A(t).

One interesting approach to this problem is as follows. When selecting an individual in A(t) to die, pick several candidates initially and choose that one which is most similar to the new individual being inserted into the population. For the genetic models under study here, similarity is defined in terms of the number of matching binary alleles. Intuitively, this approach has the right characteristics. Until a hyperplane begins to dominate, the modified replacement policy has little effect, allowing initial exponential growth. However, as a hyperplane begins to dominate A(t), instances of that hyperplane become increasingly more likely to be replaced by other instances, resulting in reduction in the hyperplane growth rate.

This approach will clearly increase the amount of processing required to produce the next generation. Additional information (which, incidentally, has a derivative-like flavor) is being computed at each time step to control the allocation of trials. In this section, however, we will ignore the processing time tradeoffs and concentrate on the effects of this approach on the performance of genetic plans on E.

In order to gain further insight into this approach to controlled growth, a fifth parameter was defined for the genetic plans under study: a crowding factor parameter CF which specifies the number of individuals initially selected as candidates to be replaced by a particular offspring. CF=1 is equivalent to no crowding factor, and as CF increases, the more likely it becomes that similar individuals replace one another. As an initial study of the effects of the crowding factor, the behavior of the following four members of this new class of genetic plans was analyzed: R5(50,.001,.6,.1,CF) where CF = 1, 2, 3, and 4. Figure 4.18 illustrates the allele loss rates of each of the plans on test function F1. Recall that, because of the symmetry of F1, theoretically there should be no allele loss. As one can see, increasing the crowding factor results in a dramatic decrease in the allele loss rate. Figures 4.19 and 4.20 give the off-line and on-line performance curves generated by these plans on test function F5. Recall that these studies were motivated by the observation that the off-line performance curves of R4 on F5 suggested that premature convergence was still a problem on multimodal surfaces. Figure 4.19 illustrates that the crowding factor has in fact the right effect on F5 with R5(2) very nearly converging to the global minimum within the interval of observation. Figure 4.20 illustrates that, like mutation, increasing the crowding factor adversely affects on-line performance. This, of course, is due to the constraints placed on the number of samples allocated to the observed best.

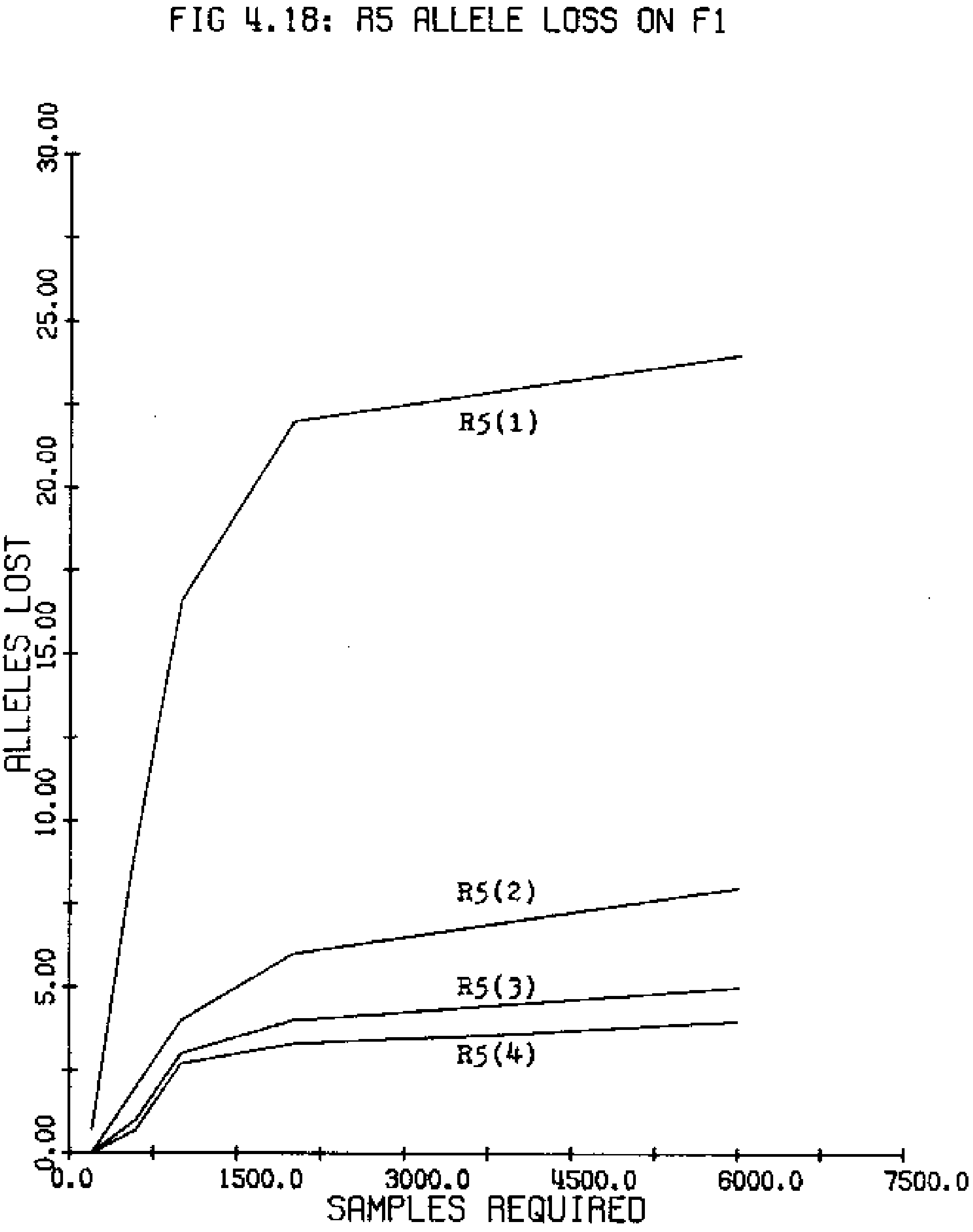


Figure 4.18: Allele loss for R5 on F1.

FIG. 4.19: OFF-LINE PERFORMANCE OF R5 ON F5

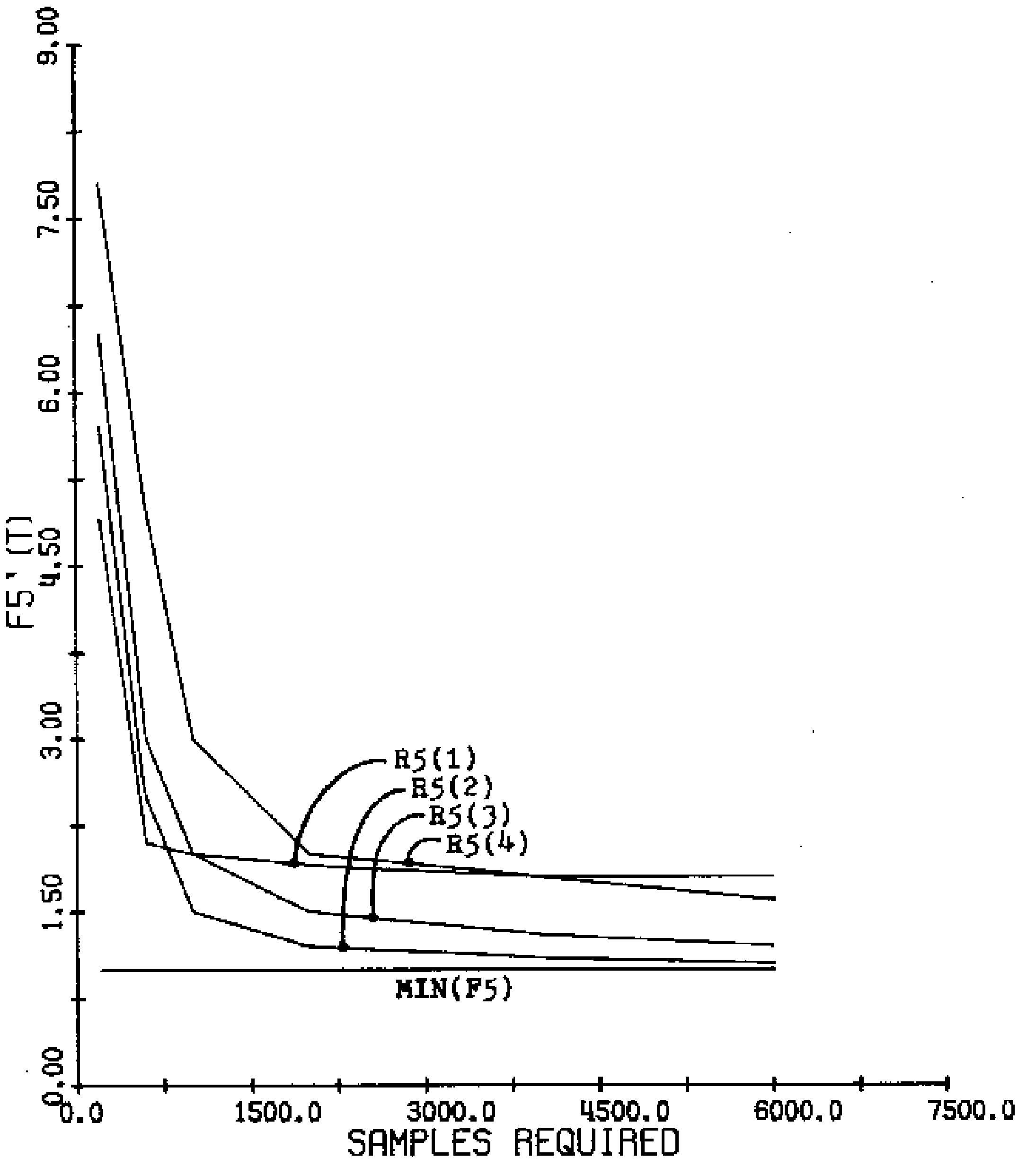


Figure 4.19: Off-line performance for R5 on F5 as a function of the crowding factor.

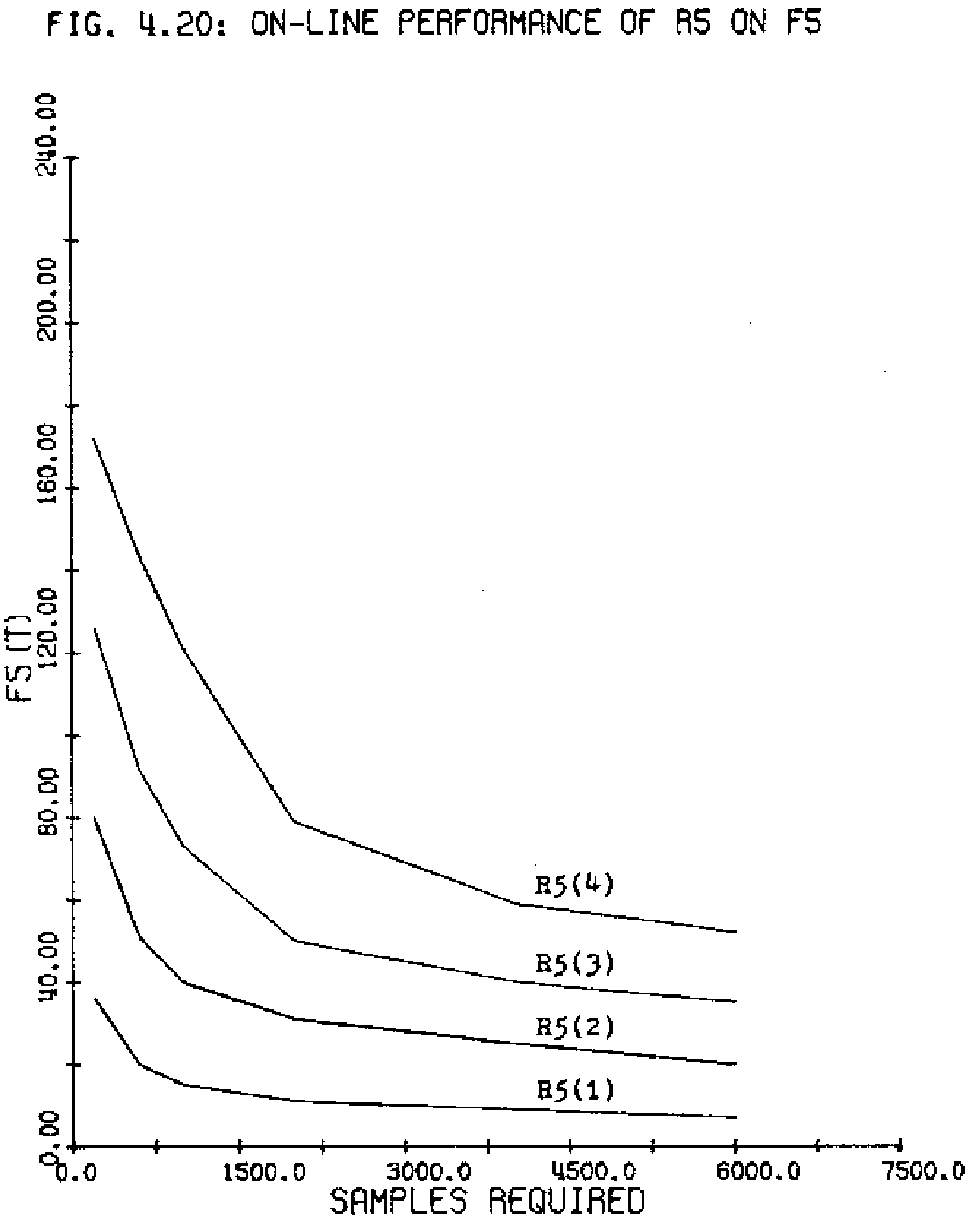


Figure 4.20: On-line performance for R5 on F5 as a function of the crowding factor.

Any reduction slows the overall improvement rate. Notice, however, that these negative effects are not nearly so severe with increases in the crowding factor as they are with increases in mutation. The reason seems obvious. Mutation controls the growth rate by randomly changing allele values, an approach which becomes more likely to produce performance degradation as adaptation progresses. On the other hand, the crowding factor provides the same kind of controlled growth rate by reducing the number of offspring produced by instances of dominating hyperplanes, rather than modifying offspring allele values.

We began this analysis of the crowding factor by arbitrarily choosing an overlapping generation model for which G=.1. Since we saw in previous studies that increasing the generation gap led to improved performance, it is of interest to explore that possibility here. In this situation we do not have quite the freedom in choosing G that we had before since as G increases beyond .5, the concept of the crowding factor becomes less meaningful, and makes little sense at all if nearly all the population is being replaced. As a consequence, the effects of crowding were analyzed for two other values of G, .2 and .4. Figures 4.21 and 4.22 give the off-line performance curves for each of these settings and illustrate two points of interest. The first observation is that, for the same crowding factor, increasing the

FIG. 4.21: OFF-LINE PERFORMANCE OF R5 ON F5

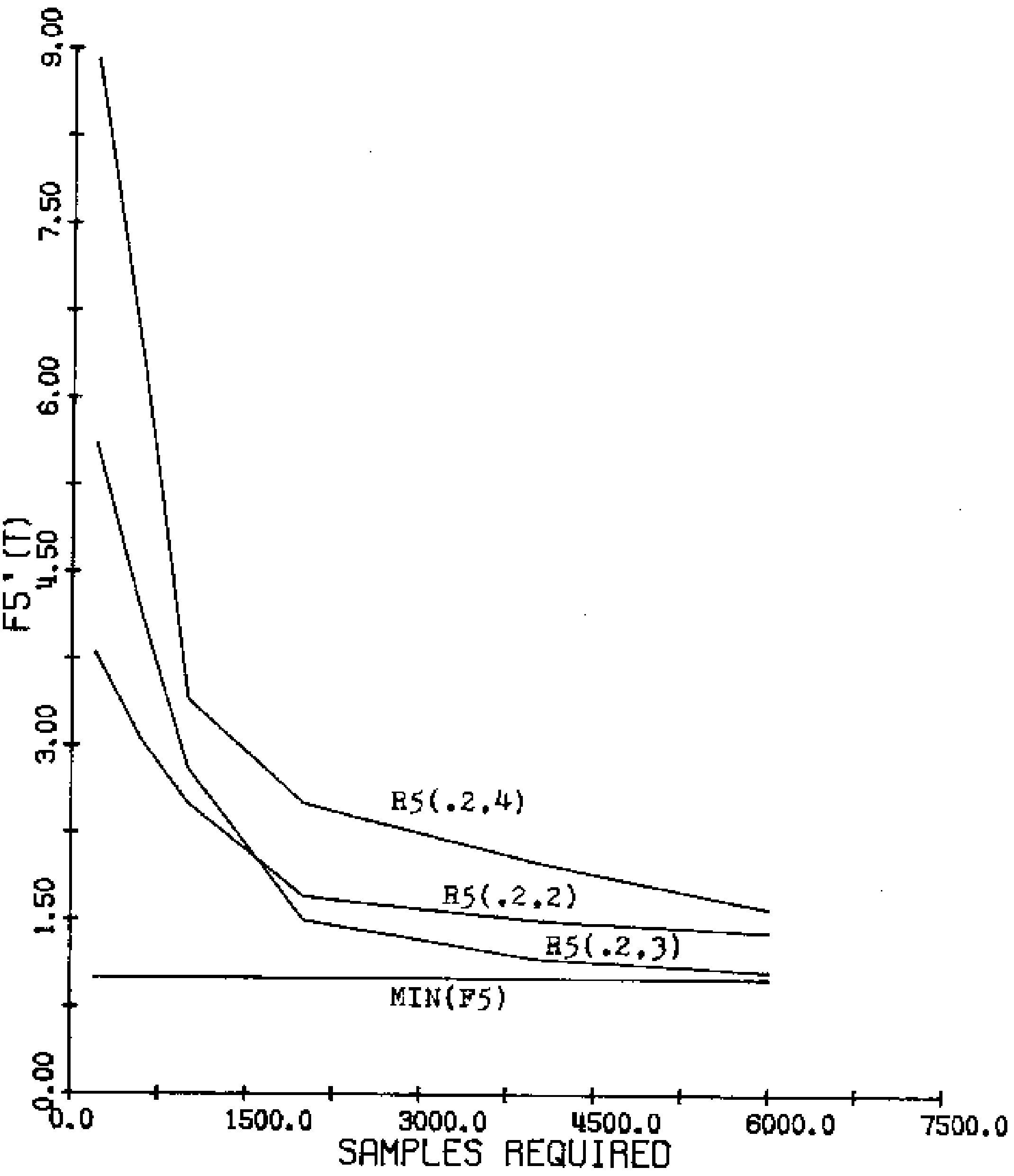


Figure 4.21: Off-line performance for R5 on F5 as a function of the crowding factor.

FIG. 4.22: OFF-LINE PERFORMANCE OF R5 ON F5

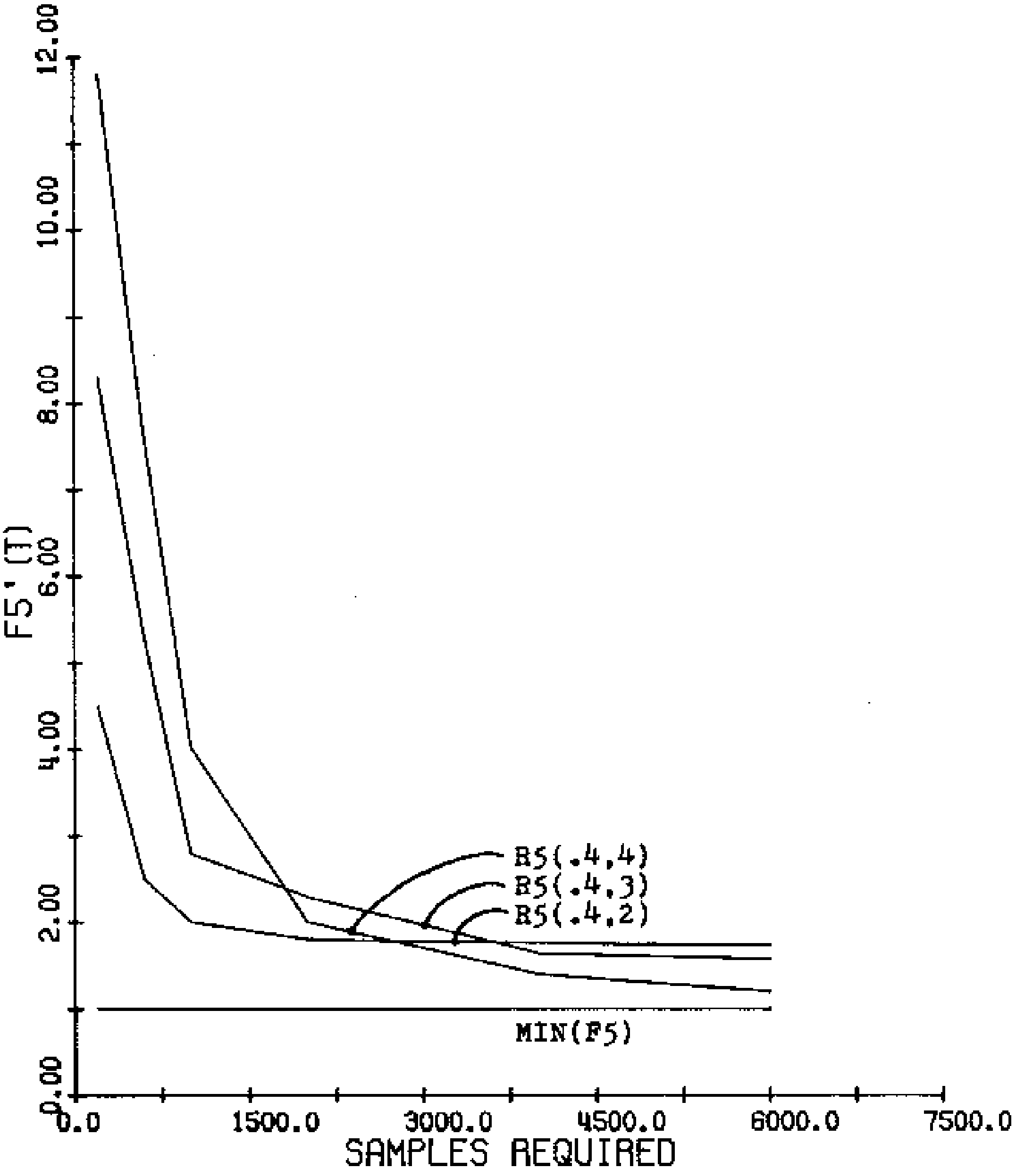


Figure 4.22: Off-line performance for R5 on F5 as a function of the crowding factor.

generation gap actually degraded the off-line performance curves of R5 on F5. In each case an increase in the crowding factor was required to preserve the same kind of performance curve. These observations suggest an interesting interaction between crowding and generation gaps: the larger the generation gap, the less effective crowding becomes. One possible explanation for this interaction is as follows. As the generation gap increases, the lifespan (in terms of generations) of an individual is reduced with a compensating increase in the number of offspring per generation. The crowding factor operates by reducing the lifespan of individuals and, hence, with shorter average lifespans its effectiveness is reduced.

Finally, it remains to be seen what effect crowding has on the overall performance of genetic plans on E. To analyze these effects, the following members of R5 were evaluated on E:

R5(50,.001,.6,.1,2)
R5(50,.001,.6,.1,3)
R5(50,.001,.6,.1,4)
R5(50,.001,.6,.2,2)
R5(50,.001,.6,.2,3)
R5(50,.001,.6,.2,4)
R5(50,.001,.6,.4,3)
R5(50,.001,.6,.4,4)

Tables 4.3a and 4.3b give the corresponding off-line and on-line performance indices computed over 6000 trials. As in the previous section these results indicate quite clearly the tradeoffs in performance we face. Including a crowding factor in genetic plans improves significantly

T=6000	R5(.1,2)	R5(.1,3)	R5(.1,4)	R5(.2,2)	R5(.2,3)	R5(.2,4)	R5(.4,3)	R5(.4,4)
$x^*_{F1}(T)$	.186	.185	.093	.092	.103	.161	.170	.129
$x^*_{F2}(T)$	.284	.111	.103	.226	.125	.097	.476	.411
$x^*_{F3}(T)$	-26.8	-26.38	-25.9	-27.5	-26.73	-27.08	-27.18	-26.5
$x^*_{F4}(T)$	21.79	28.55	28.85	19.36	25.13	23.85	22.73	23.36
$x^*_{F5}(T)$	3.07	4.48	5.07	5.28	4.79	4.37	3.88	3.21
$X^*_E(T)$	-.299	1.39	1.64	-.508	.684	.279	.015	.122

Table 4.3a: Off-line performance of R5 on E

T=6000	R5(.1,2)	R5(.1,3)	R5(.1,4)	R5(.2,2)	R5(.2,3)	R5(.2,4)	R5(.4,3)	R5(.4,4)
$x_{F1}(T)$	6.49	7.47	7.02	5.03	6.94	6.79	6.42	6.90
$x_{F2}(T)$	130.05	145.85	152.24	100.84	134.26	128.11	112.85	127.7
$x_{F3}(T)$	-20.95	-18.64	-17.7	-21.46	-20.4	-18.8	-21.1	-19.0
$x_{F4}(T)$	76.2	95.85	101.87	71.37	87.12	91.83	74.99	87.11
$x_{F5}(T)$	45.3	58.5	65.14	46.7	56.39	66.28	67.12	67.89
$X_E(T)$	47.36	57.81	61.71	40.49	52.86	54.84	48.06	53.12

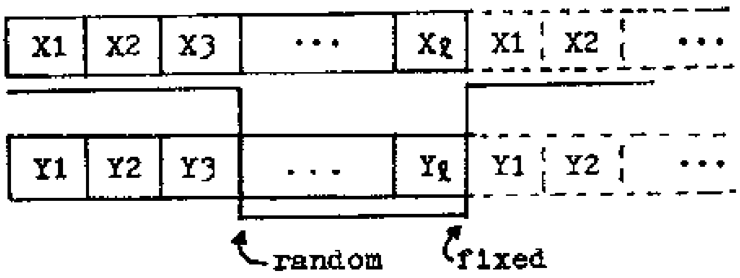
Table 4.3b: On-line performance of R5 on E

their performance on multimodal surfaces at the expense of rapid convergence on the unimodal surfaces. On the other hand, the distinction between these tradeoffs becomes less evident as the interval of observation increases and more emphasis is placed on convergence.

4.8 Generalized Crossover Model R6

Recall from the hyperplane analysis introduced in chapter 2 that genetic plans generate near-optimal allocation of trials to competing hyperplanes whose definition length (the shortest gene segment containing all the fixed positions) was short relative to the chromosome length ℓ. This was due to the fact that crossover disrupted the allocation of trials according to performance with a probability directly proportional to the definition length of a hyperplane. This means that the performance of the genetic plans under study may in fact be representation dependent. That is, one binary representation of the space to be searched may be less affected by crossover than another representation because high-performance groups of alleles are physically closer together. One solution to this problem is to allow the representation itself undergo adaptation be introducing a genetic inversion operator which physically permutes genes on a chromosome without loss of functional position. Studies by Franz (1972) suggest that changes effected by inversion are difficult to detect except perhaps in long-term behavior. Since we are concerned with practical applications, we explore an alternate approach in this section: the possibility of modifying the crossover operator itself to reduce representation dependencies.

Recall again how crossover has been defined to this point. After selecting two individuals, a crossover point is selected uniformly from the ℓ-1 positions between the ℓ genes. The offspring consists of the first segment of the first parent up to the crossover point and the remaining segment of the second parent. If we think of a chromosome as a circle with the first gene immediately following the last then it becomes immediately clear that there are in fact 2 crossover points: one fixed at position zero and the other randomly selected.



An immediate generalization to the present crossover operator is to allow both crossover points to be randomly selected. Further generalization can be made by allowing an arbitrary number of crossover points. But notice that the actual number of crossover points is always an even number since (from the circular viewpoint) you always end up back where you started.

In order to understand what effects these changes have on the allocation of trials to competing hyperplanes, we generalize the discussions of chapter 2. We need to compute the probability that an offspring after crossover lies in a different hyperplane partition element than its parent. If we let $x_1, x_2, \cdots, x_k$ be the k positions defining a hyperplane, then the offspring will certainly lie in the same hyperplane if there were an even number of crossover points between each consecutive pair of fixed points (x_i, x_j). Hence, if we think of the hyperplane's k fixed points as dividing a chromosome into k segments (in the circular viewpoint), the probability of staying in the same hyperplane is at least as great as the probability that each segment contains an even number of crossover points. Restating this as the probability of the loss of an offspring to another hyperplane, we have:

$$\Pr\left[\text{crossover loss}\right] \leq 1 - P_{nk}$$

where P_{nk} is the probability that each of the k segments received an even number (including zero) of the n=2m crossover points.

In order to get a feeling for how these probabilities change by increasing the number of crossover points, consider the effects on second-order hyperplanes. They divide the chromosome up into two segments of length

$l_1=x_2-x_1$ and $l_2=l-(x_2-x_1)$ with the probability of a randomly selected crossover point falling in one of them given by l_1/l. If we assume the convention that whenever an odd number of crossover points are randomly selected, the final even crossover point is defined to occur at position zero, then we have:

$$P_{n2} = \sum_{i=0}^{m} \binom{n}{2i} \left(\frac{l_1}{l}\right)^{2i} \left(\frac{l_2}{l}\right)^{n-2i}$$

where n is the number of randomly selected crossover points with m=n/2 (integer division).

Figure 4.23 illustrates how the loss probability $1-P_{n2}$ changes both as a function of the definition length l_1 and the number of randomly selected crossover points n for second order hyperplanes with a chromosome length of $l=30$. Notice that there are two distinct families of curves: one for n even and one for n odd. When an odd number of crossover points are randomly selected, the probability of loss for widely spaced fixed points remains high since it remains likely that all the crossover points will fall in the long segment defined by the two fixed points. On the other hand, randomly selecting an even number of crossover points immediately drops the loss probabilities to .5 or less with $l/2$ spacings becoming the most likely victims.

How these generalizations of the crossover operator

FIG 4.23: PROBABILITY OF LOSS DUE TO CROSSOVER

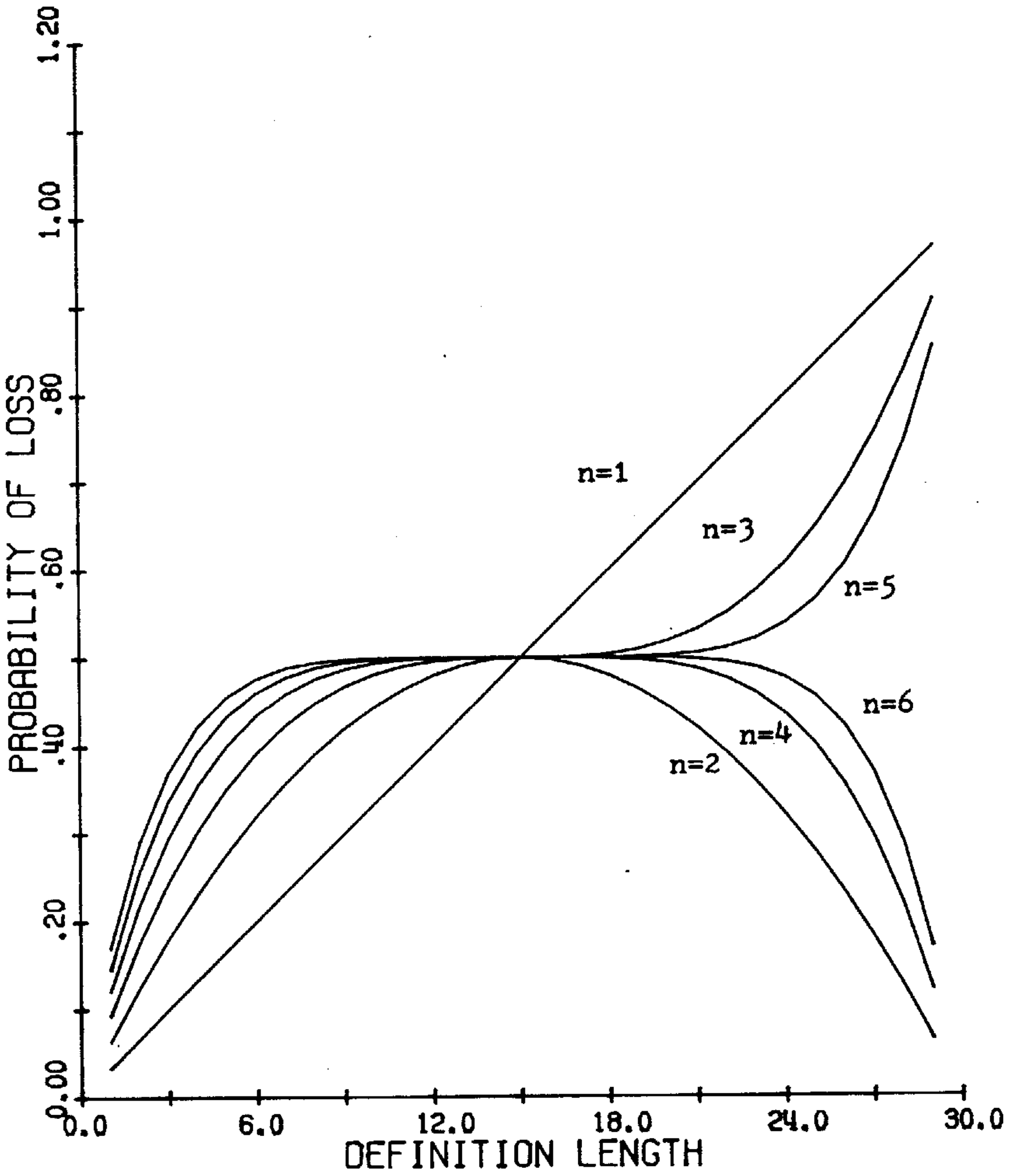


Figure 4.23: Loss probability curves for second order hyperplanes on chromosomes of length 30 as a function of the number of crossover points.

will affect the performance of genetic plans is not clear. We may find that they reduce representation dependencies at the cost of lower overall performance on E. On the other hand one can imagine that perhaps a modest increase in the number of randomly selected crossover points may generate performance improvements while larger numbers of crossover points may be too disruptive to the allocation of trials to higher order hyperplanes.

To explore these possibilities, plan R5 was modified to accept a sixth parameter, CP, which specifies the number of crossover points to be randomly selected. Up to this point we have been implicitly using a value of CP=1. If CP is odd, the final crossover point is assumed to occur at position zero.

As an initial attempt to understand the implications of generalized crossover, five members of R6 were evaluated on test function F1:

R6(50,.001,.6,1.0,1,1)
R6(50,.001,.6,1.0,1,2)
R6(50,.001,.6,1.0,1,3)
R6(50,.001,.6,1.0,1,4)
R6(50,.001,.6,1.0,1,8)

Figure 4.24 depicts the allele loss generated by these members of R6 on test function F1, and illustrates that the allele loss rate actually increases as CP does. This is a surprising observation for which an explanation is not immediately clear. It may very well be the case that the previous disruption of the allocation of trials to the longer hyperplanes may have counteracted some

FIG 4.24: R6 ALLELE LOSS ON F1

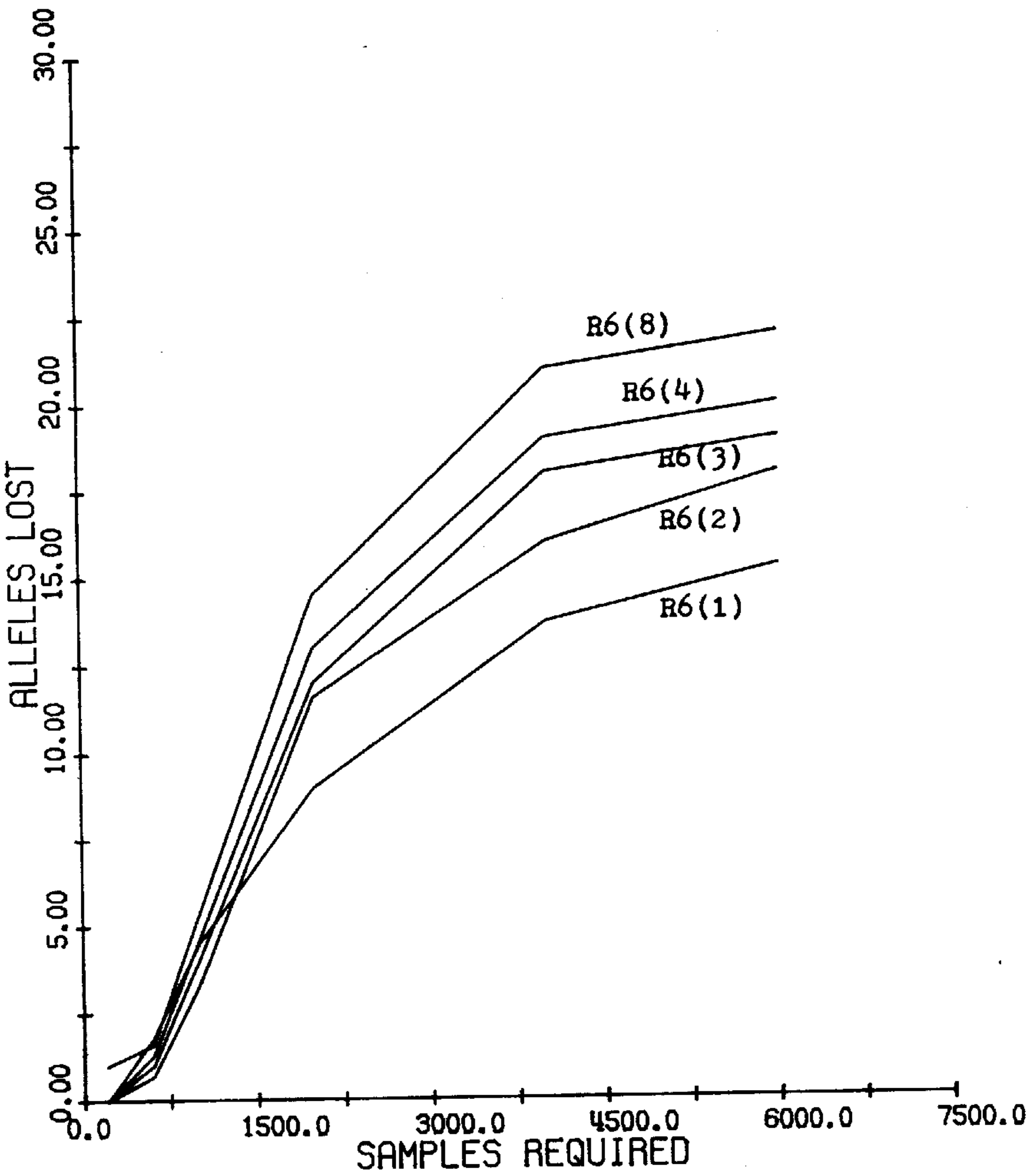


Figure 4.24: Allele loss for R6 on F1 as a function of the number of crossover points.

of the stochastic side-effects of genetic drift and, hence, the removal of some of this disruption opened the door for increased allele loss.

Figures 4.25 and 4.26 give the off-line and on-line performance curves generated by these members of R6 on F1. In general, increasing the number of crossover points seems to degrade slightly the off-line performance of R6 with R6(8) exhibiting the old problem of premature convergence. This is probably due to the previously noted increased allele loss rate. It is also interesting to note that initial on-line performance also degrades somewhat as CP increases, suggesting that increasing the number of crossover points leads to a less conservative sampling policy in the initial stages of adaptation.

Finally, to evaluate the effects of generalized crossover on the performance of R6 on E, the behavior of the following members was analyzed on E:

R6(50,.001,.6,1.0,1,1)
R6(50,.001,.6,1.0,1,2)
R6(50,.001,.6,1.0,1,3)
R6(50,.001,.6,1.0,1,4)
R6(50,.001,.6,1.0,1,8)
R6(50,.01, .6,1.0,1,2)
R6(50,.001,.6, .1,2,2)

Tables 4.4a and 4.4b summarize the off-line and on-line performance indices for these evaluations over 6000 trials. The first five members analyzed differed only in the number of crossover points selected. The best overall off-line performance was achieved with CP=2, chiefly on the basis of its performance on F4. Note that

FIG. 4.25: OFF-LINE PERFORMANCE OF R6 ON F1

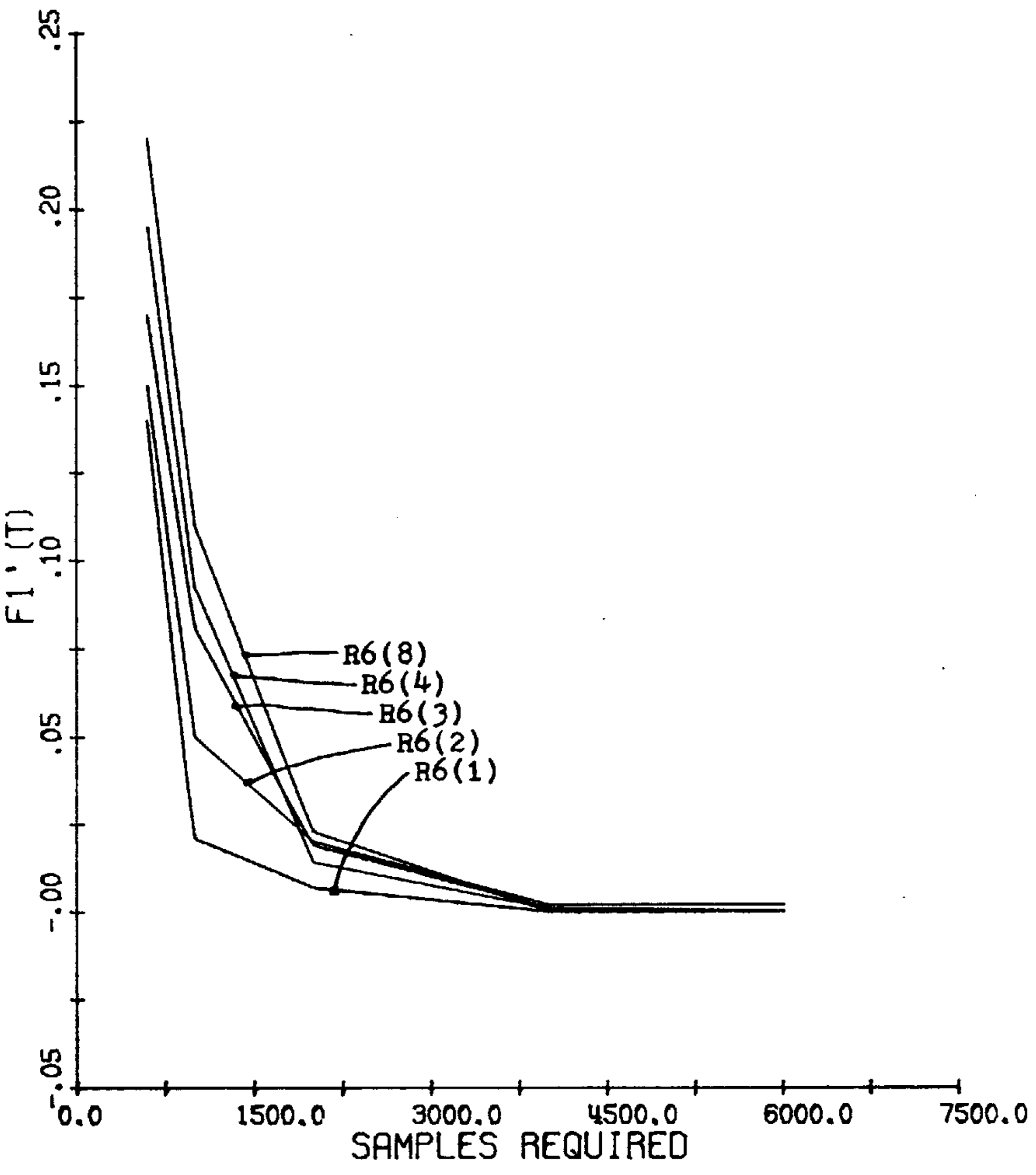


Figure 4.25: Off-line performance curves for R6 on F1 as a function of the number of crossover points.

FIG. 4.26: ON-LINE PERFORMANCE OF R6 ON F1;

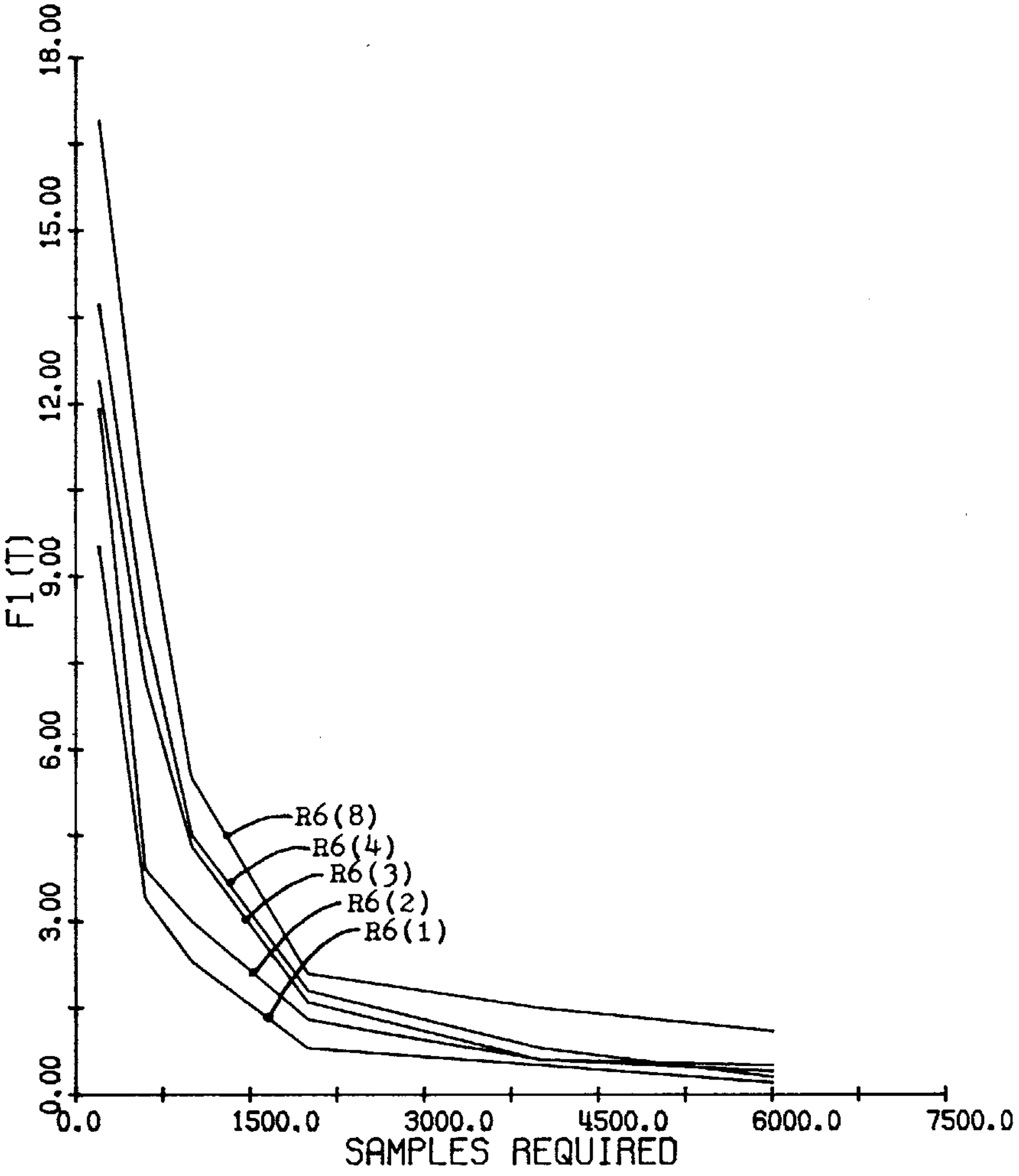


Figure 4.26: On-line performance curves for R6 on F1 as a function of the number of crossover points.

T=6000	CP=1	CP=2	CP=3	CP=4	CP=8	P_c=.01 CP=2	G =.1 CF=2 CP=2
$x^*_{F1}(T)$	.113	.117	.122	.127	.138	.129	.116
$x^*_{F2}(T)$	.221	.197	.237	.244	.289	.133	.215
$x^*_{F3}(T)$	-28.2	-27.99	-27.86	-27.53	-26.91	-28.3	-27.1
$x^*_{F4}(T)$	17.62	14.01	18.19	21.06	25.23	19.59	23.9
$x^*_{F5}(T)$	3.34	4.74	5.27	5.61	5.85	3.95	3.87
$X^*_E(T)$	-1.38	-1.79	-.808	-.098	.919	-.899	.200

Table 4.4a: Off-line performance of R6 on E

T=6000	CP=1	CP=2	CP=3	CP=4	CP=8	P_c=.01 CP=2	G =.1 CF=2 CP=2
$x_{F1}(T)$	2.32	2.67	2.88	3.23	3.51	6.47	5.07
$x_{F2}(T)$	34.76	36.99	38.97	41.25	46.85	112.6	96.7
$x_{F3}(T)$	-26.49	-26.17	-25.83	-25.14	-24.61	-22.16	-22.5
$x_{F4}(T)$	40.73	37.54	42.64	47.97	49.8	95.9	83.8
$x_{F5}(T)$	34.34	36.91	38.27	42.38	45.63	82.65	74.5
$X_E(T)$	17.12	17.59	19.39	21.94	24.24	55.09	47.51

Table 4.4b: On-line performance of R6 on E

performance of F5 degrades significantly as CP increases, indicating again how important a low allele loss rate is. As we saw on F1, on-line performance degrades significantly as CP increases.

The last two members were chosen as likely candidates to improve the off-line performance of R6 on E. Time and resources prohibited a more detailed analysis of the six parameters defining R6. However, since we had earlier noted the increased allele loss rate of R6 on F1, the two most likely candidates for improvement were increased mutation rates and the crowding factor. One instance of each was chosen; neither improved off-line performance over 6000 trials and, as we have seen before, both degrade on-line performance.

4.9 Summary

We began this chapter by analyzing the performance of the basic family R1 of genetic plans on E. While this family outperforms random search on E, we noted that there was considerable room for improvement. To this basic plan we added an elitist policy which biased the allocation of trials slightly toward the hyperplanes which produced the currently best individual. This resulted in improved performance on E, particularly on the unimodal surfaces. In fact, performance on F5 was degraded suggesting that an elitist policy improves the local search properties of genetic plans. As an alternative, expected value model R3 was introduced which attempted to improve performance by minimizing the difference between the actual and expected number of offspring produced by individuals in A(t). This produced a significant increase in performance on E. R3 was then modified to include the previously mentioned elitist policy. Again the performance on E was improved to the extent that on test functions F1-F4, there were no signs of premature convergence over the interval of observation. Rather, R4 generated steady progress toward the minimum with convergence within 6000 trials on F1 and F2. F5, however, remained a difficult challenge. To that end, several members of the R4 family were analyzed on F5 to see whether a change in parameters would improve the off-line performance curve. Increasing the mutation rate to .01 seemed about the most effective change for F5 but resulted in an overall decrease in off-line performance on E. As an alternative approach to improving the global search properties of R4, a crowding factor model was introduced which attempted to slow the growth rate of hyperplanes beginning to dominate the finite population A(t). Like increasing the mutation rate, the crowding factor improved off-line performance on F5 at the expense of on-line performance, but the tradeoff was less pronounced. Finally, generalized crossover model R6 was introduced in an attempt to alleviate possible representation problems caused by the disrupting effect crossover has on long-definition hyperplanes. By allowing several crossover points to occur when generating an offspring, the disruptiveness on long-definition hyperplanes can be considerably reduced. Although time did not permit a complete analysis of the effects, no significant improvement in performance on E was noted. In fact overall performance was seen to degrade as the number of crossover points increased.