1 I

The jo
est cor
comple
is one
bers.

mach[redacted]
mach[redacted]
mach[redacted]
mach[redacted]
mach[redacted]
mach[redacted]

[redacted]

2 Job Sh[redacted]

The job shop[redacted]
N jobs to be [redacted]
sumes the fol[redacted]

- A mach[redacted] time.
- The pr[redacted] called a[redacted]
- An ope[redacted]
- A job c[redacted]
- An ope[redacted] *machin*[redacted] for ope[redacted]
- An op[redacted] called *j*[redacted] set of j[redacted] represe[redacted]
- A feas[redacted] called a[redacted]

The JSP is t[redacted] the total ela[redacted]

The flow sh[redacted] JSP, where [redacted] all jobs.

on the same machine. For N jobs there exist N(N-1)/2 job pairs. If there exist N jobs and M machines, total MN(N-1)/2 bits are required to represent a solution. For example, 90 bits for N=M=6, 450 bits for N=M=10, and 950 bits for N=20, M=5.

Figure 1 shows symbolic and binary representations of a schedule for the 6 × 6 (N=6 jobs, M=6 machines) problem [Muth and Thompson, 1963]. In the binary matrix, the rows represent the 15 job pairs and the columns represent the bit vectors. For the matrix B, B(1,1)=1 means job 1 is to be executed prior to job 2 on machine 1, and B(1,3)=0 means job 1 is to be executed after job 2 on machine 3. In the symbolic matrix, the i-th row represents the sequence on machine i. For example, the first row indicates that on machine 1 operations are to be executed in the sequence job4 → job3 →...

Note that the above schedule is an optimal schedule. Figure 1 shows that bits have a tendency to continue within a machine. This results confirm the heuristic that a schedule tends to keep the processing sequence for each job pair. We can make use of this good schedule heuristic in genotype initialization and crossover. Note also that, in a binary representation, each bit has its own meaning, as seen in conventional representation.

4 Evaluation

As stated above, a symbolic representation can be represented in a binary form. The inverse, however, does not hold. This means that the space of binary representation properly includes the space of symbolic representation. In fact, for the 6 × 6 problem, the number of elements of the former amounts to $2^{90} \simeq 10^{27}$, while that of the latter amounts to $(6!)^6 \simeq 10^{17}$. Hereafter, a binary representation is interchangeably called a genotype. Any evaluation of a genotype should take its wider space into consideration.

The idea behind our evaluation process is as follows. In general, a genotype g, produced either initially or by conventional crossover, is a binary representation, i.e., represents not necessarily a legal schedule. The evaluation function first finds a legal genotype g' as similar to g as possible, and then evaluates g' to determine the fitness of g. The fitness of g is evaluated as the total elapsed time of the corresponding schedule.

The procedure that creates a legal genotype g' from an illegal genotype g is called the *harmonization algorithm*. The Hamming distance is used to assess the similarity of g' to g. The harmonization algorithm goes through two phases: local harmonization and global harmonization. The former creates a symbolic representation from g, removing local inconsistency within each machine. The symbolic representation may contain global inconsistencies between machines. By removing all global inconsistencies, the latter creates a schedule from the symbolic representation. The legal genotype g' represents the schedule.

							sum
job1:	*	0	0	1	1	0	2
job2:	1	*	0	0	1	1	3
job3:	1	1	*	1	1	0	4
job4:	0	1	0	*	0	0	1
job5:	0	0	0	1	*	1	2
job6:	1	0	1	1	0	*	3

(a) Original priority

							sum
job1:	*	0	0	1	1	0	2
job2:	1	*	0	0	1	1	3
job3:	1	1	*	1	1	$\boxed{1}$	5
job4:	0	1	0	*	0	0	1
job5:	0	0	0	1	*	1	2
job6:	1	0	$\boxed{0}$	1	0	*	2

(b) After selecting job

							sum
job1:	*	0	0	1	1	0	2
job2:	1	*	0	$\boxed{1}$	1	1	4
job3:	1	1	*	1	1	1	5
job4:	0	$\boxed{0}$	0	*	0	0	0
job5:	0	0	0	1	*	1	2
job6:	1	0	0	1	0	*	2

(c) After selecting job

							sum
job1:	*	0	0	1	1	$\boxed{1}$	3
job2:	1	*	0	1	1	1	4
job3:	1	1	*	1	1	1	5
job4:	0	0	0	*	0	0	0
job5:	0	0	0	1	*	1	2
job6:	$\boxed{0}$	0	0	1	0	*	1

(d) Final priority

Figure 2 : Local Harmonization

The [redacted]
each [redacted]
desc [redacted]
algo [redacted]
use [redacted]
the [redacted]
be e [redacted]
pair [redacted]
get [redacted]
2 (a [redacted]
dire [redacted]
elem [redacted]
exec [redacted]
gori [redacted]
high [redacted]
Wh [redacted]
one [redacted]
erat [redacted]
cons [redacted]
repe [redacted]
the [redacted]
pea [redacted]
the [redacted]
It s [redacted]
job [redacted]
who [redacted]
gen [redacted]

It i [redacted]
niza [redacted]
whi [redacted]
a ge [redacted]
a le [redacted]

The [redacted]
ded [redacted]
des [redacted]
ing [redacted]

Th [redacted]
bol [redacted]
ma [redacted]
job [redacted]
sto [redacted]
can [redacted]

Th [redacted]
tha [redacted]
tio [redacted]
the [redacted]
it r [redacted]
rith [redacted]
tio [redacted]

Table 1 : Main Results f

Papers	Algorithm	6 × 6 p
Balas1969	BAB	55
McMahon1975	BAB	55
Barker1985	BAB	55
Carlier1989	BAB	55
Nakano1991	GA	55

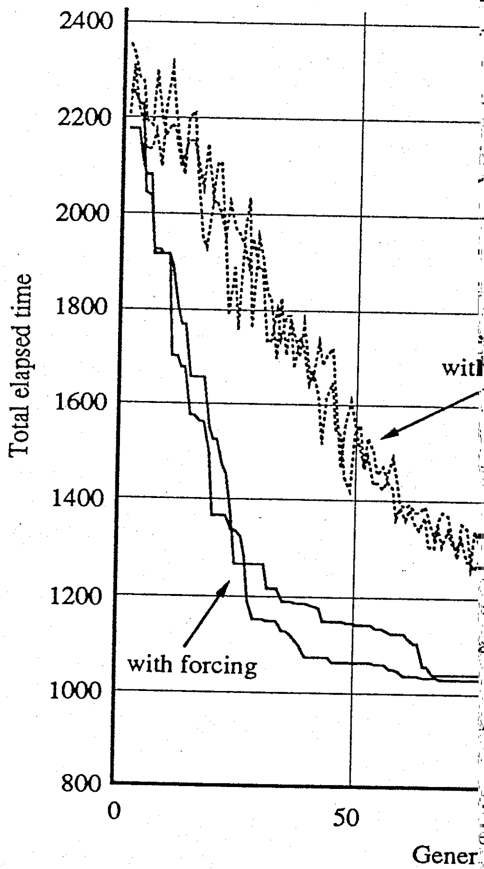

Figure 3 : Convergence of G

6 Experiments

This section shows the results of experiments conducted using three well-known JSP benchmarks [Muth and Thompson, 1963]:

- 6 × 6 (6 jobs, 6 machines) problem
- 10 × 10 (10 jobs, 10 machines) problem
- 20 × 5 (20 jobs, 5 machines) problem

Our best results (total elapsed time) are shown in Table 1 together with the historical progress in branch and bound methods.

7 Co [redacted]

This pa [redacted] effective [redacted] the job [redacted] introduc [redacted] survival. [redacted] makes it [redacted] Even th [redacted] tional c [redacted] tion met [redacted] finding [redacted] replacem [redacted] gal one, [redacted] lution q [redacted] JSP ben [redacted] ated by [redacted] those ob [redacted]

Future [redacted] complex [redacted] sented i [redacted] represen [redacted] also wor [redacted]

Ackno [redacted]

The aut [redacted] for discu [redacted] Inooka [redacted] shop pr [redacted] helpful [redacted]

Refere [redacted]

[Balas, 1 [redacted] disjun [redacted] algori [redacted]

[Barker [redacted] G.B. [redacted] shop. [redacted]

[Carlier [redacted] J Car [redacted] solvin [redacted] 35(2): [redacted]

[Clevela [redacted] and S [redacted] to sch [redacted] *Int. C* [redacted] *Appli* [redacted] 1989. [redacted]

[Davis, 1 [redacted] with [redacted] *Conf.* [redacted] *plicat* [redacted] 1985. [redacted]