IT3708 BIO-Inspired AI

Project 4 Report - Job Shop Scheduling Problem

Stefan Sobczyszyn Borg

```
18  2  9 14  4 19  8  5  7 16 15 11  0 17 12  3 10  6  1 13
14 11 18 17  4  5  1 16 13 19 15  7  9  0  8  3  2  6 12 10
 1 11  9  3 18  4 14 10  7 13  0  2  5 17 19 12 16 15  8  6
13  1  0 11  6 10 14 19  4 16  8  9 17 12 18  2  5  3 15  7
 2 17 13  4 12 14  5  0 18  9 10  1 19 15  6 11  7 16  8  3
```

Figure 1: Preference list representation of a solution for problem 3 from the test data. Each row represents the machine and each column represents a job. This means that for the first machine, if there is a deadlock between job 18 and job 2, job 18 will have preference. This representation yielded a makespan of 1232.

In the following algorithm, S is the schedule being constructed. The set A is used to hold the set of schedulable operations, where an operations o is said to be schedulable if it has not been scheduled yet.

Algorithm 1. Hybrid Giffler and Thompson

1. Set $S = \{\ \}$;
2. Let $A = \{o_{ji} \mid 1 \le j \le N\}$;

while $A \neq \emptyset$ **do**

3. $\forall o_i \in A$ let $st(o_i)$ be the lowest starting time of i, if scheduled now;
4. Let $o_k \in A$ such that $st(o_k) + du(o_k) \le st(o) + du(o)$, $\forall o \in A$; where $du(o)$ is the processing time for operation o. (if two or more operations are tied, pick the leftmost operation in the chromosome);
5. Set M^* is the machine that is to process o_k;
6. Let $B = \{ o \in A \mid$ it is to process on machine M^* and $st(o) < st(o_k) + du(o_k)\}$;
7. Let $o_t \in B$ such that $st(o_t) \le st(o)$, $\forall o \in B$;
8. Select $o^* \in B$ such that o^* is the leftmost operation in the chromosome and add o^* to S with starting time $st(o^*)$;
9. Let $A = A \backslash \{o^*\} \cup \{SUC(o^*)\}$; where $SUC(o)$ is the next operation to o in its job if any exists;

end while

Figure 2: Pseudocode for the Giffler Thompson algorithm for building active schedules from a preference list representation

The solution representation is the same for all the algorithms. Individuals are represented with a preference-list for each machine, and uses the Giffler Thompson algorithm to build an active schedule from the representation. The GT algorithm keeps track of the next operation for each job (given by the input specification). It then selects the machine, m, with the potential earliest finish time given the possible operations. The next step is to identify all operations which have a starting time before the finish time on m and are performed on m. If there are several operations satisfying the mentioned conditions, there is a deadlock. Deadlocks are resolved by the preference list representation. The preference list contains an identifier for each job. The operations (and their corresponding job) in deadlock are compared to their place in m's preference list, and the operation from the first occurring job is added to the schedule. The makespan of the generated schedule is decided by the ordering of jobs in the preference list.

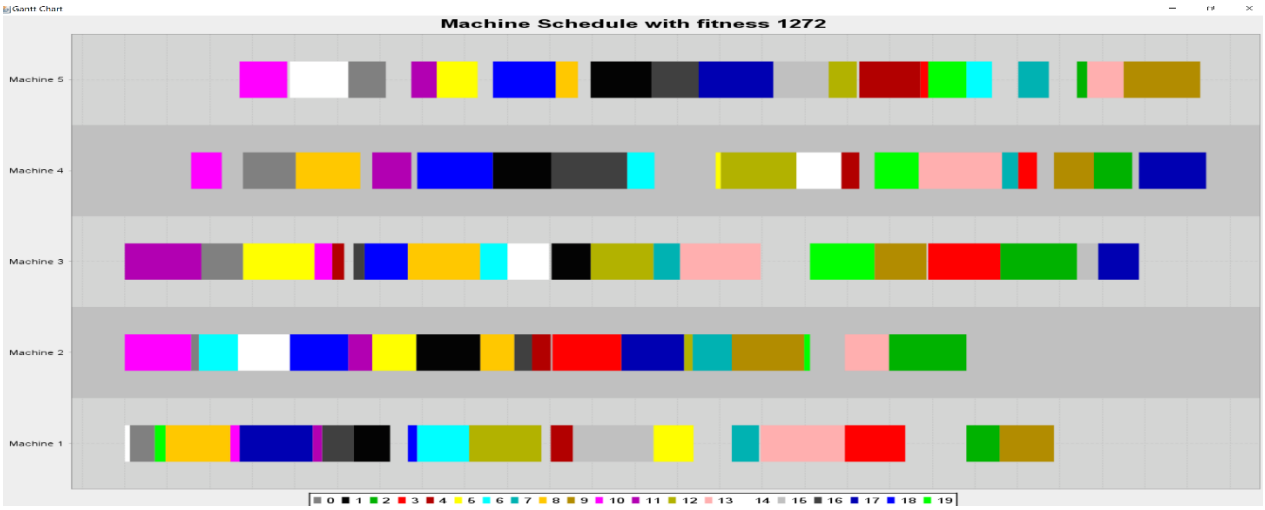

Figure 3: Schedule found for problem 3 using bee's algorithm (BA). Makespan found: 1272

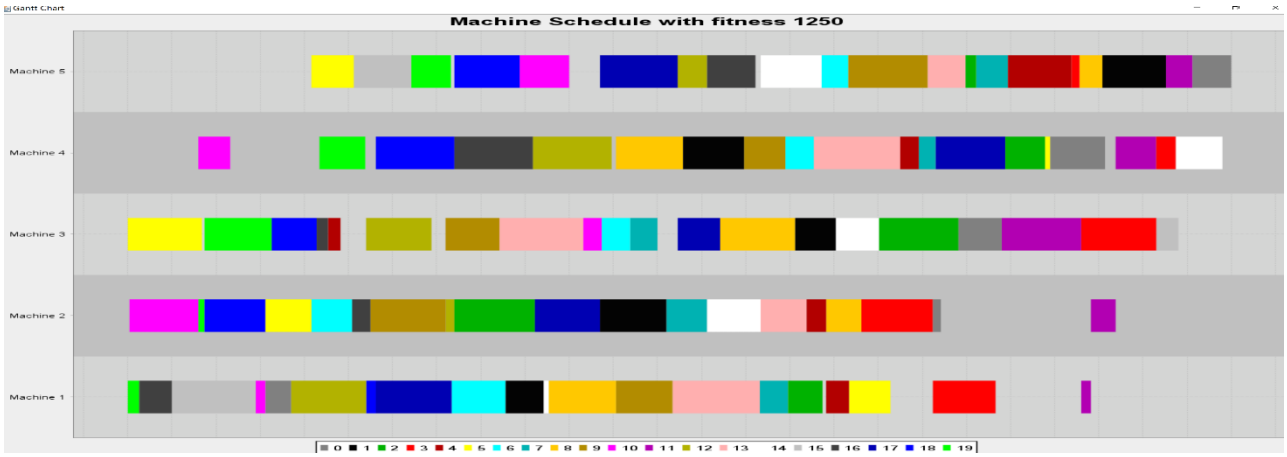

Figure 4: Schedule found for problem 3 using particle swarm optimization algorithm (PSO). Makespan found: 1250

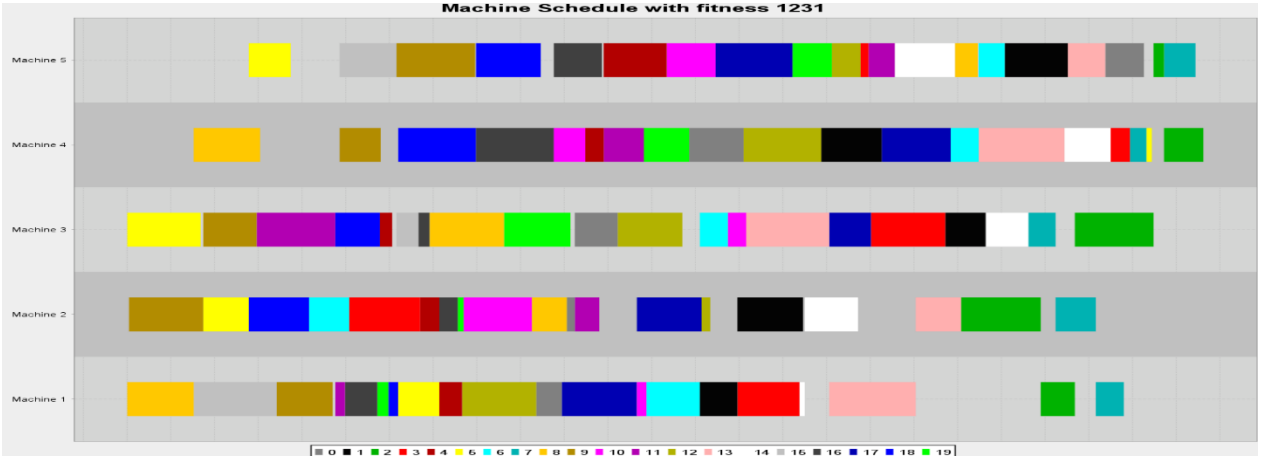

Figure 5: Schedule found for problem 3 using ant colony optimization algorithm (ACO). Makespan found: 1231