A guided exploration method of genetic algorithm for Flexible Job Shop Problem

J. Autuori, F. Hnaien, and F. Yalaoui

ICD-LOSI, Université de Technologie de Troyes, 12 rue Marie Curie,
CS42060, 10004 Troyes Cedex, France
{julien.autuori,faicel.hnaien,farouk.yalaoui}@utt.fr

1 Introduction

The metaheuristic exploration is characterized by the convergence to improve the solutions and the diversification to avoid the local minimum. In this paper, a method to guid the solution space exploration of a metaheuritic is proposed. A *Mapping* Method (MaM) convert the solution space in a one dimensional space. The solutions are converted to binary and classified according to the Hamming distance with the reference solution. This space is divided in several zones. The zones containing the best solutions and the unexplored zones are identified. In this zones, the Local Search (LS) and creation of solutions are associated to improve the convergence and the diversification. To experiment the MaM, a Genetic Algorithm (GA) is implemented in order to solve the Flexible Job Shop Problem (FJSP) with the objective to minimize the makespan. The solution space explorations of the GA, the GA with LS (GA-LS) and GA with MaM (GA-Map) are compared and the experimental results prove the performance amelioration due to MaM.

2 Adaptation of *Mapping* Method

The MaM converts a multidimensional solutions in one dimensional space, where each solution is represented by only one position. The steps of MaM are as follows:

1. A binary conversion is applied for each solution Y, $Y \longrightarrow X^b$.
2. Comparison of X^b with X^{ref}, $X = d_H(x_i^b, x_i^{ref})$, $\forall i \in \{1, ..., n\}$.
3. $X = \{x_1, ..., x_n\}$ is the binary representation of solution Y on the *mapping* with the size n.
4. Decimal conversion of X using the *mapping* function $f(X)$ (see equations (1) and (2)):

$$\begin{aligned} D = \{0,1\}^n & \longrightarrow I \in \mathbb{N} \\ f(X = \{x_1, ..., x_n\}) & \longrightarrow a \end{aligned} \tag{1}$$

$$f(X) = \sum_{i=0}^{d_H(X)-1} C_n^i + \sum_{i=1}^{n} \left[C_{i-1}^{e(X,i)-1} \cdot (1 - x_i) \right] \tag{2}$$

with X^{ref} the reference binary solution, $d_H(X)$ is the Hamming distance with the reference solution, a is the position of the solution Y on the one dimensional map and $e(X,i) = \sum_{j=1}^{i} x_j$.

The map is divided in zones according to MaM. The solutions in a zone has the same Hamming distance. The zone exploration quality are classified according to number of explored solutions, unexplored zone (uz) or explored zone (ez). In addition, an ez with one or more best solutions is noted bz.

The FJSP, usually defined like in Pezzella et al. (2007) [3], can be divided into two sub-problems. The first one is the assignment of each operation to one machine, this machine belongs to a set machine qualified of operation. The second one is to schedule the operations assigned to machine, as in classical job shop problem. Garey et al. (1976) [2] have shown that the FJSP is NP-hard with the objective to minimize the *makespan*.

In order to solve the FJSP, Pezzella et al. (2007) [3] propose a GA. At each iteration, a roulette wheel select the parents for the reproduction. The Precedence Preserving Order-based (POX) and the assignment crossover operators are used respectively for the sequence and assignment parts. The Precedence Preserving Shift (PPS) and assignment mutation operators are adopted. The new individuals are evaluated after the crossover and the mutation. The best individuals of the parents and offspring are selected for the next population.

In this paper we keep the same structure of GA but with hybridizations. A Local Search Phase (LSP) is applied periodically. The LS chosen is the *Simple Hill Climbing* (HC). For the GA-LS, the best solutions are selected for the HC. For the GA-Map1, the HC is applied on the solutions in the bz. The GA-Map2 performs the HC on the solutions in the bz and created in the uz.

The experimentations are make on 5 instances of FJSP from generator used by Brandimarte et al. (1993) [1], noted (mk11, mk12, mk13, mk14 and mk15). The instance size of FJSP is defined by the number of machine m, the maximum number of operations on a job h and the number of job n $(m/h/n)$.

The population size is $p = 100$. The crossover rate is $pc = 0.45$ and the mutation rate is $pm = 0.02$ for each operators. The GA iteration number is $iter = 2000$ and the others algorithms work during the same time of the GA. The period LSP is $T = 100$.

The Table 1 contains the convergence of GA proposed by Pezzella et al. (2007) [3], GA-LS, GA-Map1 and GA-Map2. The instance sizes and the algorithm names are in the first and second columns. The third column contains the intermediary results with a period of 100 iterations.

Table 1. Convergence of the algorithms for one simulation

Instance	Algorithm	Convergence
mk11	GA	926 853 830 806 803 803 803 803 803 803 803 803 803 803 803 803 803 803 803 803
(5/8/30)	GA-LS	846 816 806 801 801 796 796 796 796 796 796 796 796 796 796 796 796 796 796
	GA-Map1	846 821 821 810 804 804 801 801 801 798 798 798 798 798 798 798
	GA-Map2	846 821 800 800 796 796 796 792 790 790 790 790 790 790 790
mk12	GA	631 589 578 576 576 576 576 576 576 576 567 565 563 563 563 563 563 563 563 563
(10/10/30)	GA-LS	598 576 576 568 563 563 561 554 554 554 554 554 554 554 554 554 554 554
	GA-Map1	587 562 561 554 554 554 554 550 546 541 541 541 541 541 541 541 541
	GA-Map2	587 576 563 554 554 554 554 547 543 542 541 539 537 536 534 523
mk13	GA	495 441 439 439 439 439 429 427 426 421 421 421 421 417 417 417 417 409 405 405
(10/10/30)	GA-LS	441 439 427 427 427 420 414 409 409 406 405 405 405 404 404 404 403
	GA-Map1	435 434 422 422 418 415 408 408 408 408 408 405 403 403 403 403
	GA-Map2	435 433 427 421 416 416 416 416 413 404 404 404 400 400 400
mk14	GA	734 709 699 677 666 662 659 659 659 659 659 659 658 650 650 650 650 650 650 650
(15/12/30)	GA-LS	709 700 700 698 698 698 698 676 671 664 662 662 662 662 662 662 662 662 654
	GA-Map1	709 698 673 665 665 663 663 663 663 657 653 649 647 646 646 646 646 644
	GA-Map2	709 706 706 704 694 683 677 676 672 640 640 640 640 640 640 640 638
mk15	GA	560 527 503 498 495 488 485 472 470 470 469 460 458 456 456 456 454 454 451 451
(15/12/30)	GA-LS	521 507 494 491 489 489 481 473 471 468 468 463 453 453 451 446 445 445 445
	GA-Map1	521 507 485 480 473 472 469 466 463 462 462 462 460 454 449 444 441 441
	GA-Map2	521 498 486 481 480 472 472 462 452 451 449 446 446 446 442 439 437

The convergence of three hybridized algorithms (GA-LS, GA-Map1 and GA-Map2) are better than GA with same stopped condition time. However, GA-(Map1,Map2) have a better convergence than GA-LS. In GA-Map1, the HC performed on solution in the bz accelerates the convergence. In the GA-Map2, the increase of the diversification by the exploration of the uz improves the results.

3 Conclusion and perspectives

The MaM gives the interesting performances with the GA to solve FJSP with the makespan minimization. The convergence and diversification are improved by the HC and the creation solutions in the selected zones. In perspective, in order to confirmation the results, adaptation for other problem, method and objective can be tested.

References

1. Brandimarte P.: Routing and scheduling in flexible job shop by tabu search, 41, 157-183 (1993)
2. Garey M.R., Johnson D.S., Sethi R.: The complexity of flowshop and job shop scheduling, Mathematics of Operations Research, 1, 117-129 (1976)
3. Pezella F., Morganti G., Ciaschetti G.: A genetic algorithm for the Flexible Job-shop Scheduling Problem, Computer & Operation Research, 35, 3202-3212 (2008)