Эволюционные алгоритмы с адаптивным вызовом операторов для задач составления расписаний на перестановках с ресурсными ограничениями

М.Ю. Сахно

Омский филиал института математики
им. С.Л. Соболева СО РАН
Исследование выполнено за счет гранта Российского научного фонда № 22-71-10015, https://rscf.ru/project/22-71-10015.

Report Structure

- Genetic Algorithm with Generational Scheme
- Crossover and Mutation Operators
- Problem Statements
- Results of the Genetic Algorithm with Generational Scheme
- Genetic Algorithm with Adaptation
- Optimized Crossover
- Conclusions and Further Research

Genetic Algorithm with Generational Scheme

1: Construct the initial population $P^0 = \{\pi_j^0\}$ of k permutations. Save n_e individuals with the best objective values as elites of P^0. Put $t = 0$.

2: Until termination condition is met, perform

2.1 for $i \leftarrow 1$ to $(k - n_e)/2$

2.1.1 Select two parent permutations π^1 and π^2 using operator $Sel(P^t)$.

2.1.2 Construct $(\pi^{1\prime}, \pi^{2\prime}) = Cross(\pi^1, \pi^2)$.

2.1.3 Apply the mutation operator to constructed permutations: $Mut(\pi^{1\prime})$ and $Mut(\pi^{2\prime})$ and save the result as individuals $\pi_{2i-1}^{t+1}, \pi_{2i}^{t+1}$ for population P^{t+1}.

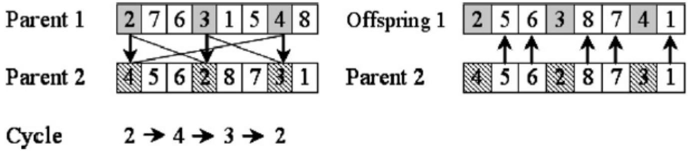
2.2 Copy elites of P^t to P^{t+1} and identify elites of P^{t+1}.

2.3 Put $t = t + 1$.

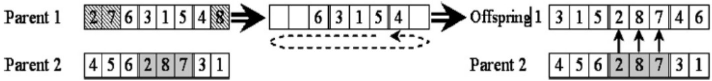
3: Return the best found individual.

Crossover Operators

Cycle Crossover (CX)

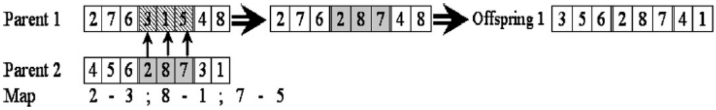


Order Crossover (OX)

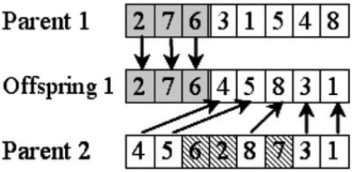


Crossover Operators

Partially Mapped Crossover (PMX)

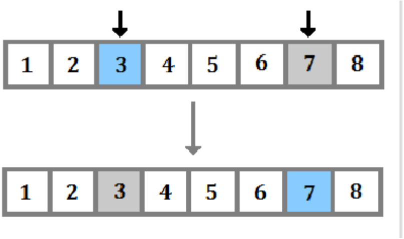


One Point Crossover (1PX)

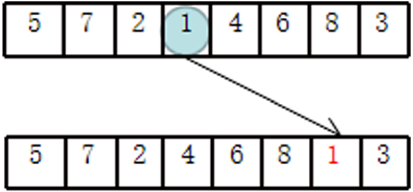


Mutation Operators

Exchange (swap) mutation



Shift (insert) mutation



Speed Scaling Scheduling

Processors and Jobs

m is the number of speed-scalable processors

$\mathcal{J} = \{1, \ldots, n\}$ is the set of jobs:
V_j is the processing volume (work) of job j
$size_j$ is the number of processors required by job j
$W_j := \frac{V_j}{size_j}$ is the work on one processor

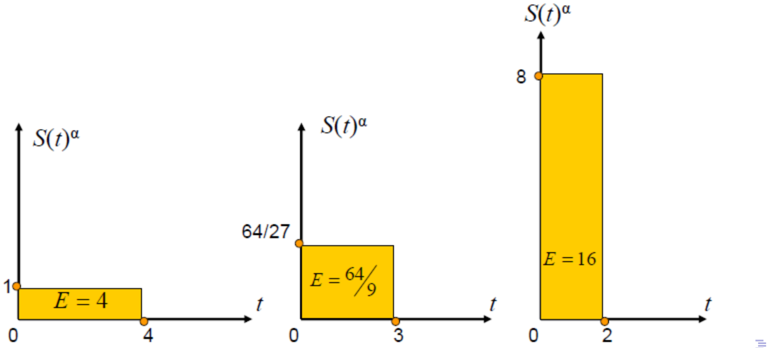
Parameters

Preemption and migration are characterized for the systems with single image of the memory.
Non-preemptive instances arise in systems with distributed memory.

Homogeneous Model in Speed-scaling

If a processor runs at speed s then the energy consumption is s^{α} units of energy per time unit, where $\alpha > 1$ is a constant (practical studies show that $\alpha \leq 3$).

It is supposed that a continuous spectrum of processor speeds is available.

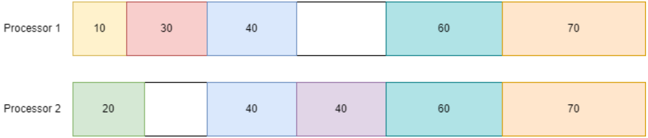


Problem 1

$m = 2$, E is the energy budget.

The aim is to find a feasible schedule with the minimum total completion time so that the energy consumption is not greater than a given energy budget.

Solution



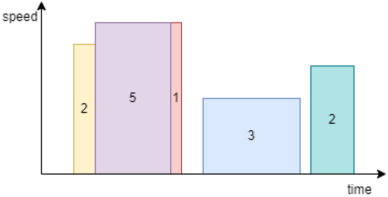
Lower Bound

Problem 2

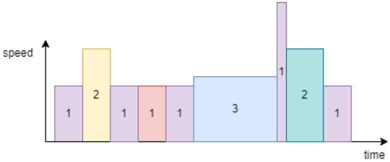
$m = 1$, the jobs have release dates and deadlines.

The objective is to find a feasible schedule that minimizes the total energy consumption.

Solution



Lower Bound



Results for Problem 1

30 instances, $n = 50$

Parameter values of genetic algorithm

Parameter name	Parameter value
k	200
n_e	2
P_{IPRand}	0.2
$Selection$	Ranking
P_{Cross}	0.8
$Crossover$	1PX
P_{Mut}	0.2
$Mutation$	Shift (insert)

Relative deviation of objective function found by the GA from the lower bound

avg: 2.03%
min: 0.83%
max: 3.83%

Results for Problem 1

30 instances, $n = 50$

Parameter values of genetic algorithm found by IRACE

Parameter name	Parameter value
k	244
n_e	146
P_{IPRand}	0.43
$Selection$	Ranking
P_{Cross}	0.7
$Crossover$	1PX
P_{Mut}	0.63
$Mutation$	Exchange (swap)

Relative deviation of objective function found by the GA from the lower bound

avg: 1.99%
min: 0.82%
max: 3.86%

Results for Problem 2

30 instances, $n = 50$

Parameter values of genetic algorithm

Parameter name	Parameter value
k	200
n_e	2
P_{IPRand}	0.2
$Selection$	Ranking
P_{Cross}	0.8
$Crossover$	CX
P_{Mut}	0.2
$Mutation$	Shift (insert)

Relative deviation of objective function found by the GA from the lower bound

avg: 0.00%
min: 0.00%
max: 0.01%

Results for Problem 2

30 instances, $n = 50$

Parameter values of genetic algorithm found by IRACE

Parameter name	Parameter value
k	170
n_e	64
P_{IPRand}	0.56
$Selection$	Tourney
$TourneySize$	7
P_{Cross}	0.79
$Crossover$	CX
P_{Mut}	0.48
$Mutation$	Shift (insert)

Relative deviation of objective function found by the GA from the lower bound

avg: 0.00%
min: 0.00%
max: 0.04%

Time comparison for Problem 2

	GA	GA_{irace}
avg	75.2	25.7
min	2	1
max	342	333

Table: Time in seconds

Genetic Algorithm with Adaptation

1: Construct the initial population $P^0 = \{\pi_j^0\}$ of k permutations. Save n_e individuals with the best objective values as elites of P^0. Put $t = 0$.

2: Until termination condition is met, perform

2.1 for $i \leftarrow 1$ to $(k - n_e)/2$

- 2.1.1 Select two parent permutations π^1 and π^2 using operator $Sel(P^t)$.
- 2.1.2 Choose crossover operator and construct $(\pi^{1\prime}, \pi^{2\prime}) = Cross(\pi^1, \pi^2)$.
- 2.1.3 Update the weight of the chosen crossover.
- 2.1.4 Apply the mutation operator to constructed permutations: $Mut(\pi^{1\prime})$ and $Mut(\pi^{2\prime})$ and save the result as individuals $\pi_{2i-1}^{t+1}, \pi_{2i}^{t+1}$ for population P^{t+1}.

2.2 Copy elites of P^t to P^{t+1} and identify elites of P^{t+1}.

2.3 Put $t = t + 1$.

3: Return the best found individual.

Adaptive Technique

1: Choose a crossover. The probability of choosing each operator is proportional to its weight.
2: Apply chosen crossover to the parent genotypes.
3: Update the weight of the chosen crossover:

$$\phi_a = \begin{cases} w_1, \text{if the new solution is a new global best,} \\ w_2, \text{if the new solution is better than the current one,} \\ w_3, \text{if the new solution is better than one of the parents or both.} \end{cases}$$

$$\rho_a = \lambda \rho_a + (1 - \lambda)\phi_a.$$

Results of Genetic Algorithm with Adaptation

30 instances, $n = 50$

Relative deviation of objective function found by the GA with Adaptation from the lower bound for Problem 1

Crossover operators: 1PX

avg: 2.06%
min: 0.83%
max: 3.88%

Relative deviation of objective function found by the GA with Adaptation from the lower bound for Problem 2

Crossover operators: CX, PMX, 1PX

avg: 0.00%
min: 0.00%
max: 0.01%

Adaptation: Mutation + Crossover for Problem 1

avg: 2.13%
min: 0.82%
max: 3.84%

Optimal Recombination Problem (ORP)

Given two parent solutions π^1 and π^2. It is required to find a permutation π' such that:

(I) $\pi'_i = \pi^1_i$ or $\pi'_i = \pi^2_i$ for all $i = 1, \ldots, n$;

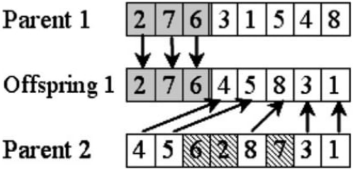
(II) π' has the minimum value of objective function $E(\pi')$ among all permutations that satisfy condition (I).

Optimal recombination may be considered as a best-improving move in a large neighbourhood defined by two parent solutions.

The ORP is NP-hard, but "almost all" instances are polynomially solvable.

Optimized Crossover

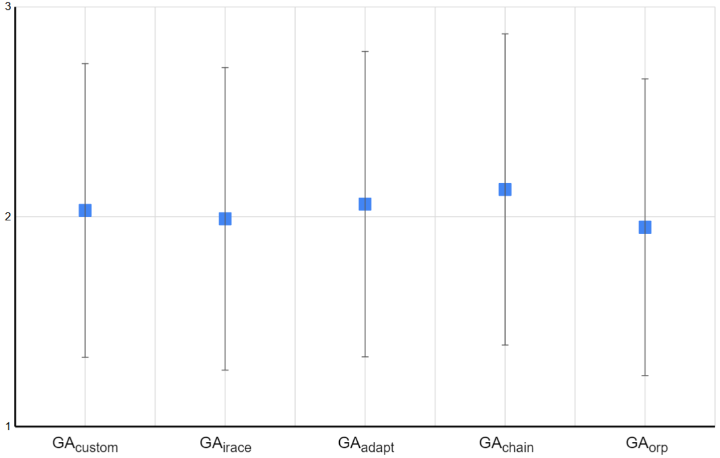
One Point Crossover (1PX)



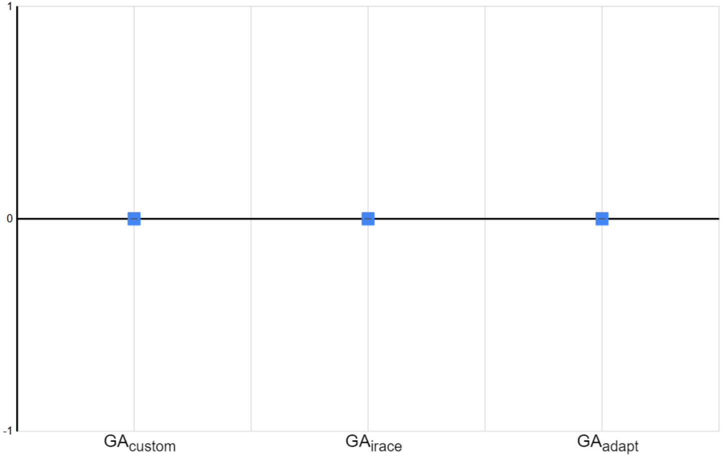
Results of Genetic Algorithm with Optimized Crossover for Problem 1

avg: 1.95%
min: 0.78%
max: 3.57%

Conclusions. Results of Genetic Algorithm Problem 1



Conclusions. Results of Genetic Algorithm Problem 2



Conclusions and Further Research

We recommend

- Apply IRACE for numeric parameters.
- Apply adaptation for operators.

Further Plans

- Trying Poisson Mutation
- Trying IRACE for Genetic Algorithm with Adaptation

Thank you for your attention!