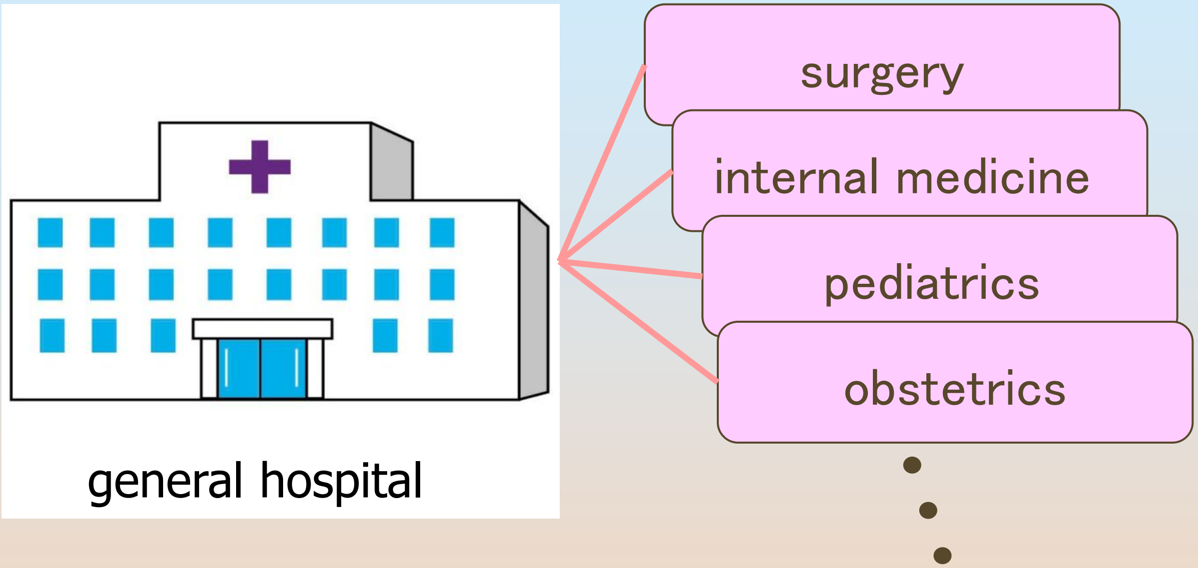
Effective Mutation Operator for Nurse Scheduling by Cooperative GA and Its Parallel Processing

Makoto Ohki

Department of Information & Electronics
Graduate School of Engineering
Tottori University

1. Introduction

1. Introduction



- General hospital consists of several sections.
- About 15–30 nurses belong to each section.
- A section director or a manager arranges a shift schedule of the nurses every month.

1. Introduction

many requirements

- requirements on the hope holiday.
- duty load in equality.
- the number of the night shift in equality.
- affinity between the nurses in the night shift.
- etc. ・・・・

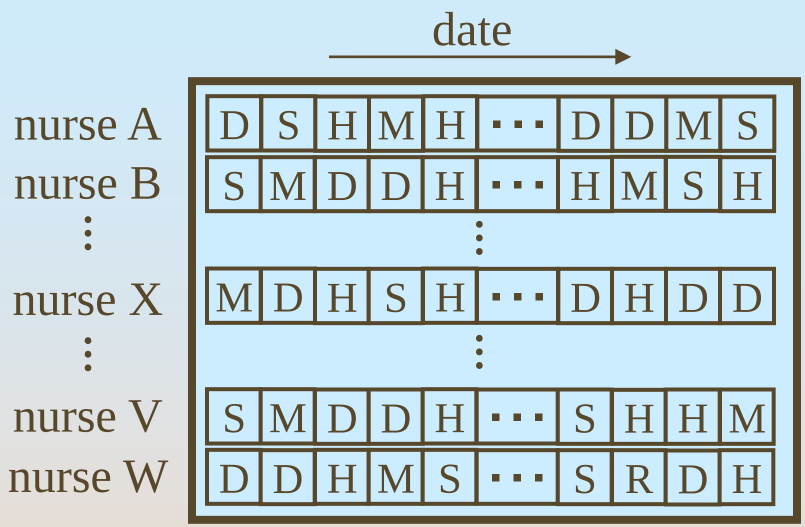
⇒ Veteran director requires one or two weeks.

⇓

Automatic Nurse Scheduling

- The nurse scheduling is very complex task, because the director consider many requirements for the scheduling.
- In our investigation, even veteran director needs one or two weeks for the nurse scheduling.
- This means a great loss of work force and time.
- Therefore, computer software for the nurse scheduling is strongly required recently.

1. Introduction



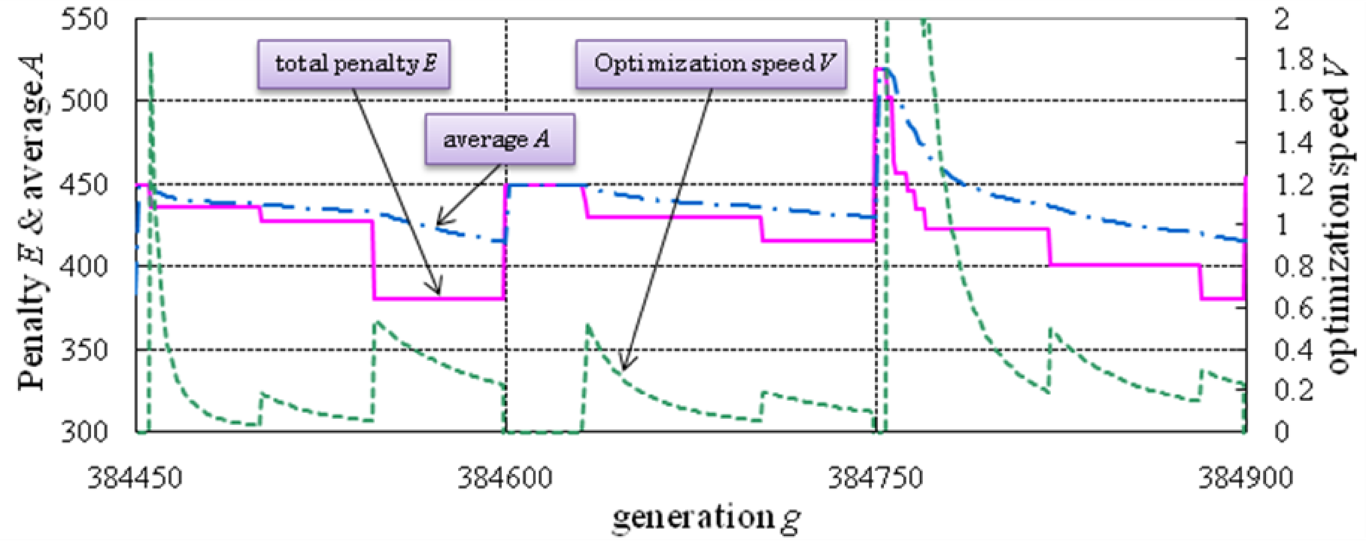
- We discuss the cooperative GA (CGA) to generate & optimize the nurse schedule.
- The conventional CGA searches solutions **only by using crossover** operator, because it is considered as the only one operator keeping **consistency** of the population.

- A **mutation** changing small parts of the population is very important for the CGA.

1. Introduction

- We have proposed an **effective mutation operator** for the CGA. (Detail of this operator is described in subsection 2.4)

Conventional Mutation Operator



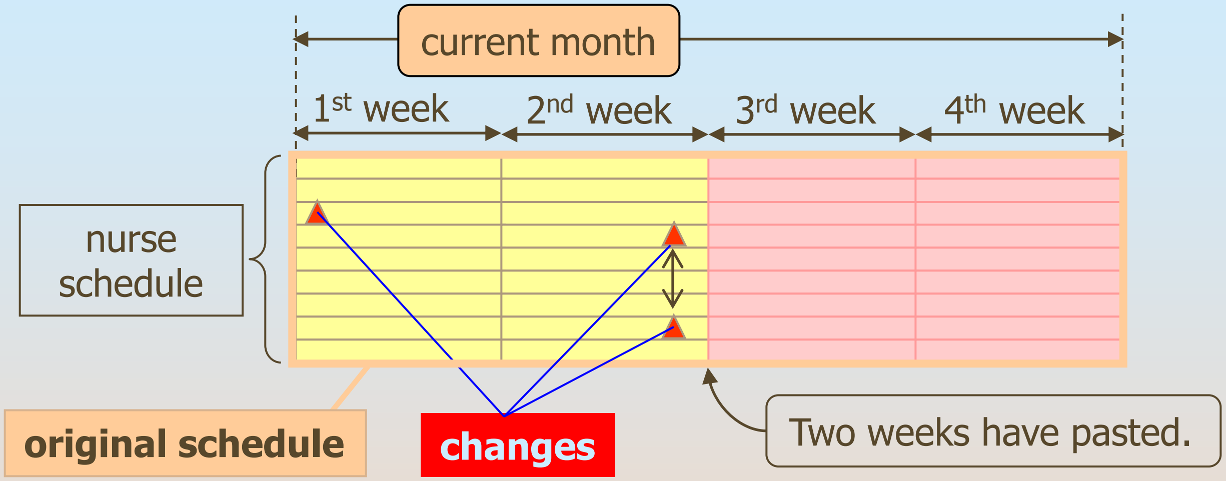
It is activated depending on the optimization speed.

- However, there several parameters to define this mutation operator.
- This means that this operator is difficult to apply.

3. Parallel Processing of the CGA

- Actually there are the cases that nurses whom a rest has been assigned to are forced to attendance by means of emergency.
- There are the cases that a nurse whom duty has been assigned to takes a rest due to diseases.
- These change of shift schedule are summarized as following five cases.
  – unplanned absence of a certain nurse,
  – unplanned attendance of a certain nurse,
  – new assignment of a new face,
  – resignation of a certain nurse, and
  – shift replacement between certain two nurses.

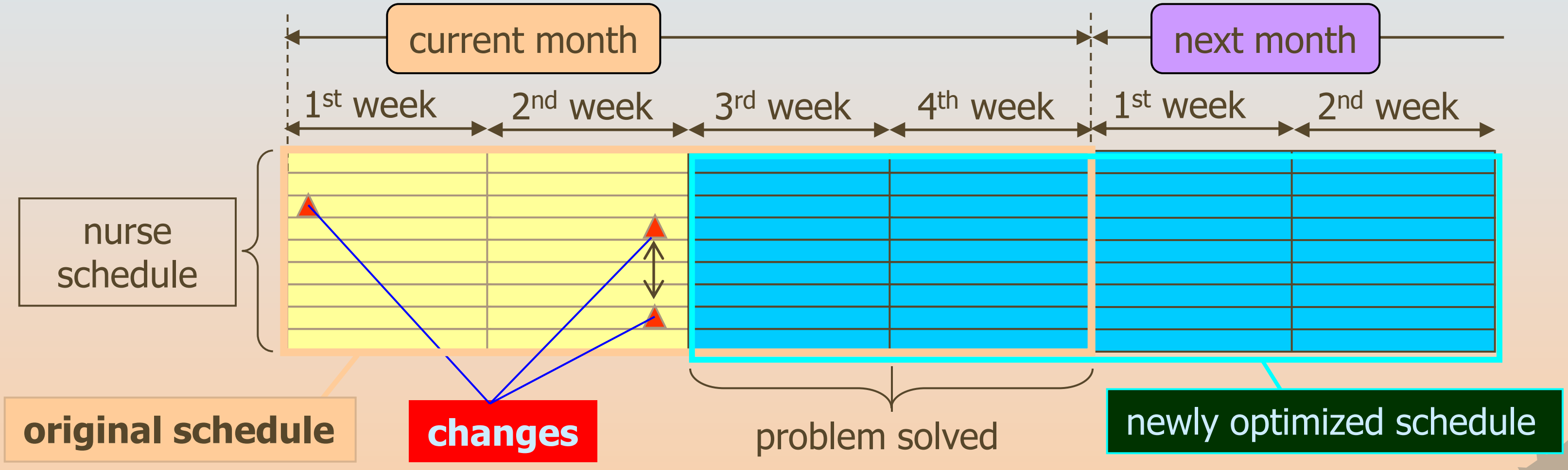
1. Introduction



- By means of the **changes**, several **inconvenience** occurs, for example, imbalance of the number of the holidays/attendance.
- Such an **inconvenience** causes the fall of the nursing level of the whole nurse organization.
- The changed schedule must be re-optimized to **break off** the **inconvenience** as much as possible.

1. Introduction

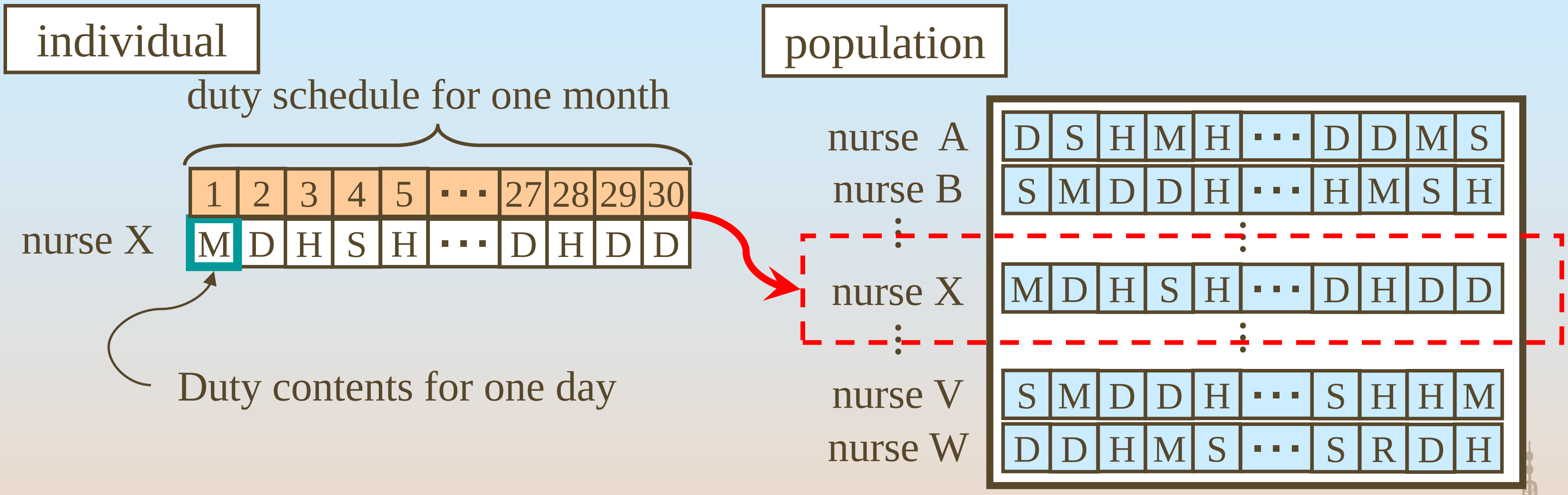
- On the other hand, there is the demand that the shift schedule already shown does not want to change if possible.
- We define a penalty function to calculate the difference on the remaining weeks of the current month.



- In this research, we treat the optimization of the coming four weeks (one month) including the remaining weeks of the current month and the several weeks of the next month.

2. Nurse Scheduling

2. Nurse Scheduling



- An individual consists of the sequence of the duty symbols.
- The individual shows the one-month schedule of the nurse X.
- Gathering all the individuals, the population is constructed.
- In the CGA, there are not two or more individuals giving the same nurse's schedule.
- The CGA optimizes the population by using crossover and mutation operators.

2. Nurse Scheduling

- For the nurse scheduling, the manager considers many requirements.
- The following requirements are satisfied by means of the coding and the genetic operations.
  - meeting, training and requested holiday must be accepted.
  - the number of nurses at each shift interval must be secured.
- The following requirements are evaluated by penalty functions.
  - duty load depending on the duty pattern of consecutive 3 days. (F_1)
  - 4 or more night shifts should not be assigned. (F_2)
  - prohibited duty patterns. (F_3)
  - fairness of the holidays and the night shifts assignment. (F_4, F_5)
  - more than or equal to 6 consecutive duty days. (F_6)
  - nursing levels must be kept at each shift interval. (F_7, F_8, F_9)
  - unfavorable combinations in the night shift. (F_{10})
  - two or more new faces should not assigned to the midnight shift. (F_{11})
  - one or more expert or more skilled nurses must be assigned on day time. (F_{12})

2. Nurse Scheduling

- Finally, we summarize those penalties into one total penalty function.

$$E = \sum_{k=1}^{4} H_k + h_5 F_{13}$$

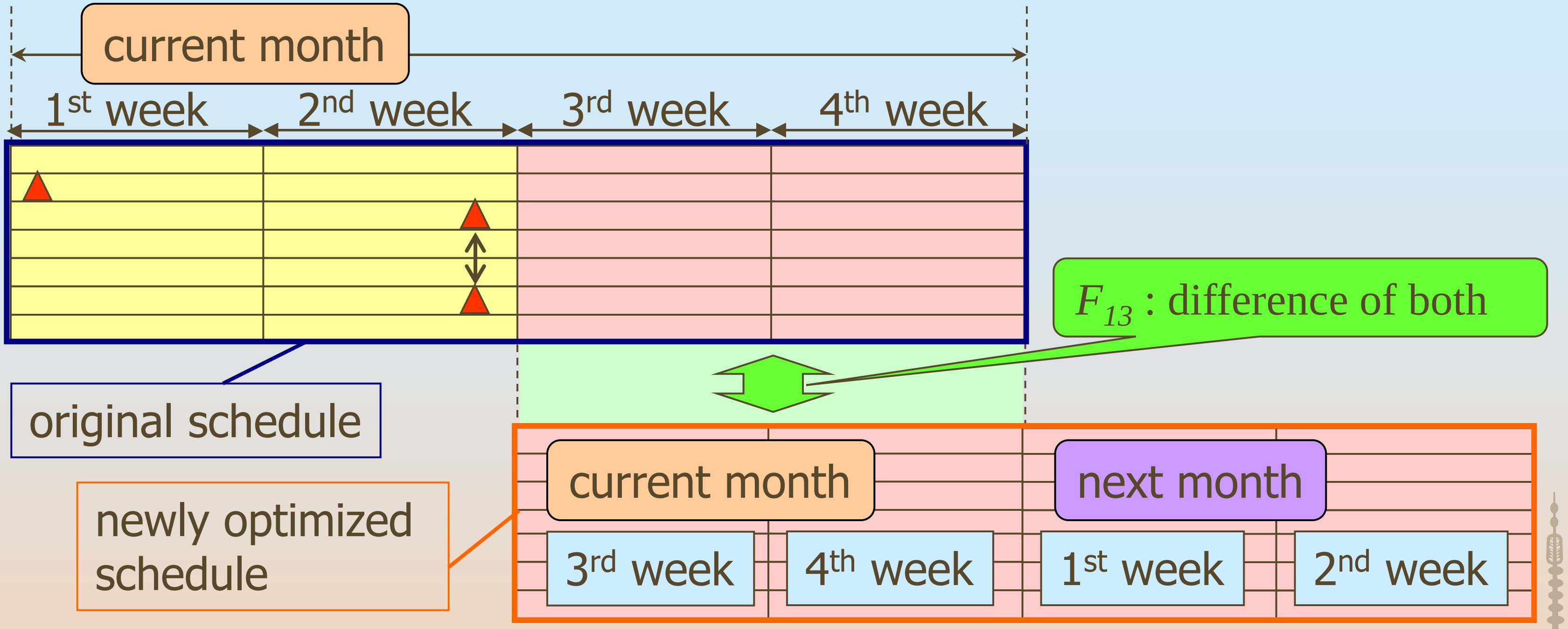
$$H_1 = \sum_{i=1}^{M} \left(h_{11} F_{1i} + h_{12} F_{2i} + h_{13} F_{3i} \right)$$

$$H_2 = \sum_{i=1}^{M} \left(h_{21} F_{4i} + h_{22} F_{5i} + h_{23} F_{6i} \right)$$

$$H_3 = \sum_{j=1}^{D} \left(h_{31} F_{7j} + h_{32} F_{8j} + h_{33} F_{9j} \right)$$

$$H_4 = \sum_{j=1}^{D} \left(h_{41} F_{10j} + h_{42} F_{11j} + h_{43} F_{12j} \right)$$

2. Nurse Scheduling



- Suppose that two weeks passed.
- There are several changes in the passed two weeks.
- Now, we want to cancel disproportion by these changes in the coming four weeks.
- We define a penalty function (F_{13}) which denotes the difference between the original and the newly optimized schedules.

2. Nurse Scheduling

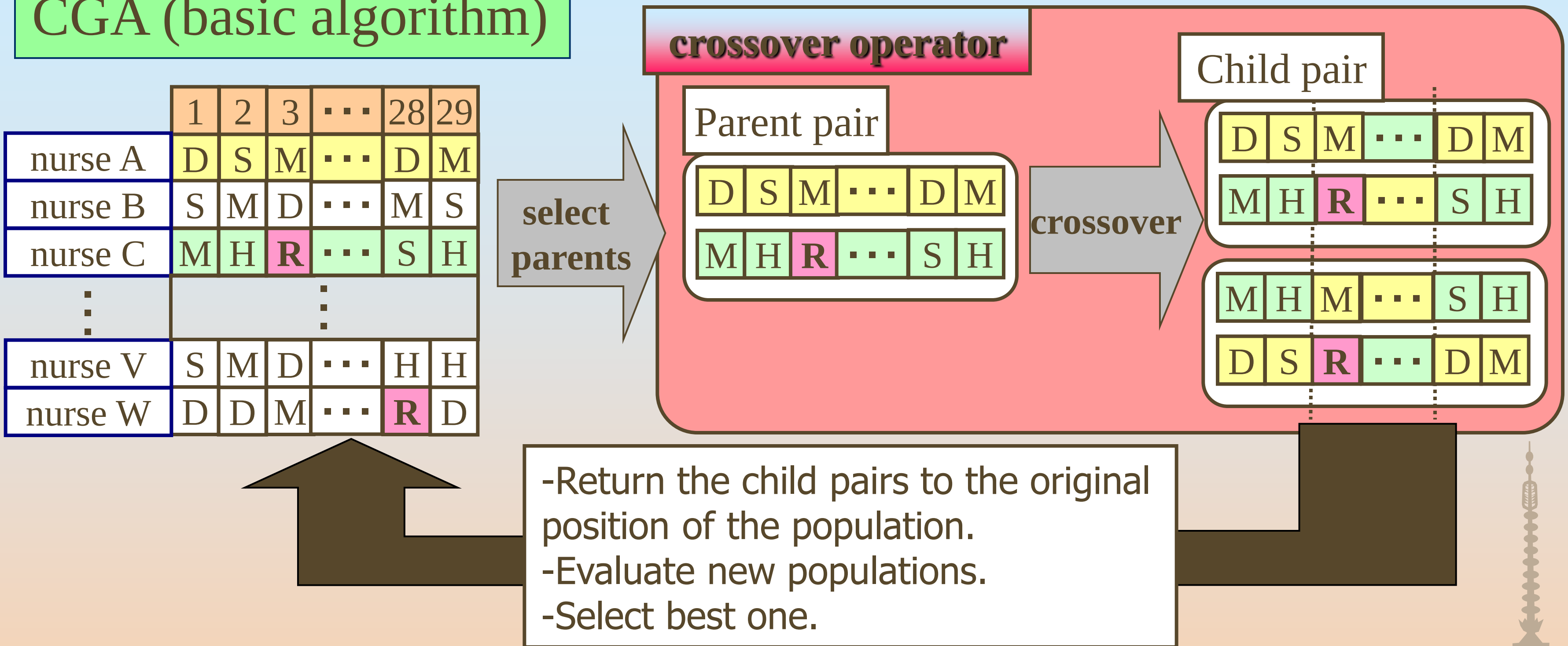
CGA (initialization)

- First, the CGA initialize the population.
- The requested holiday (R), the meeting (m) and the training (T) are treated as the fixed duty, which CGA does not move them.
- CGA put them onto the population initially.
- We suppose that the number of nurses in the day time, the semi-night and the midnight shift are defined as 6, 3 and 3 respectively in the application here.
- CGA randomly assigns the duty symbols satisfying the specific numbers.

D: day shift, S: semi-night shift, M: midnight shift
R: requested holiday, H: holiday
m: meeting, T: training

2. Nurse Scheduling

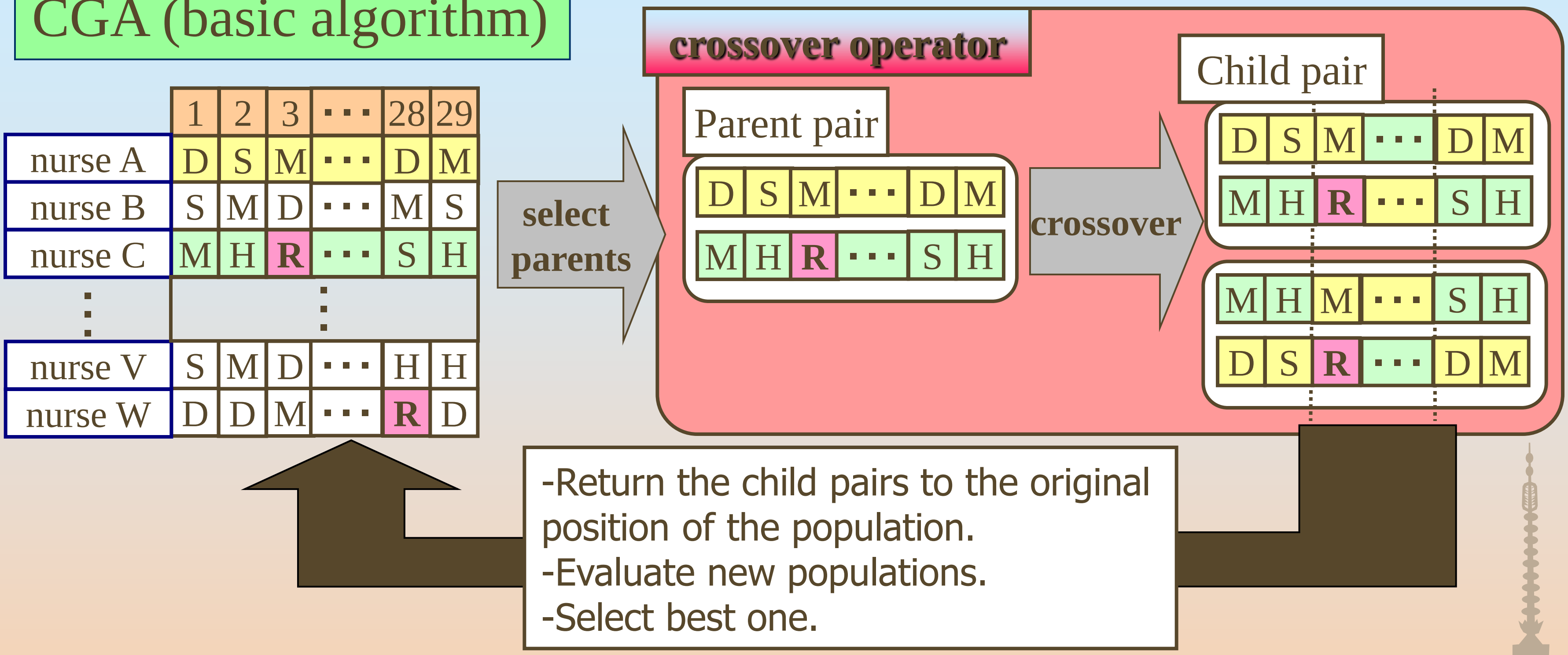
CGA (basic algorithm)



- CGA searches good solution by basically using the crossover operator.

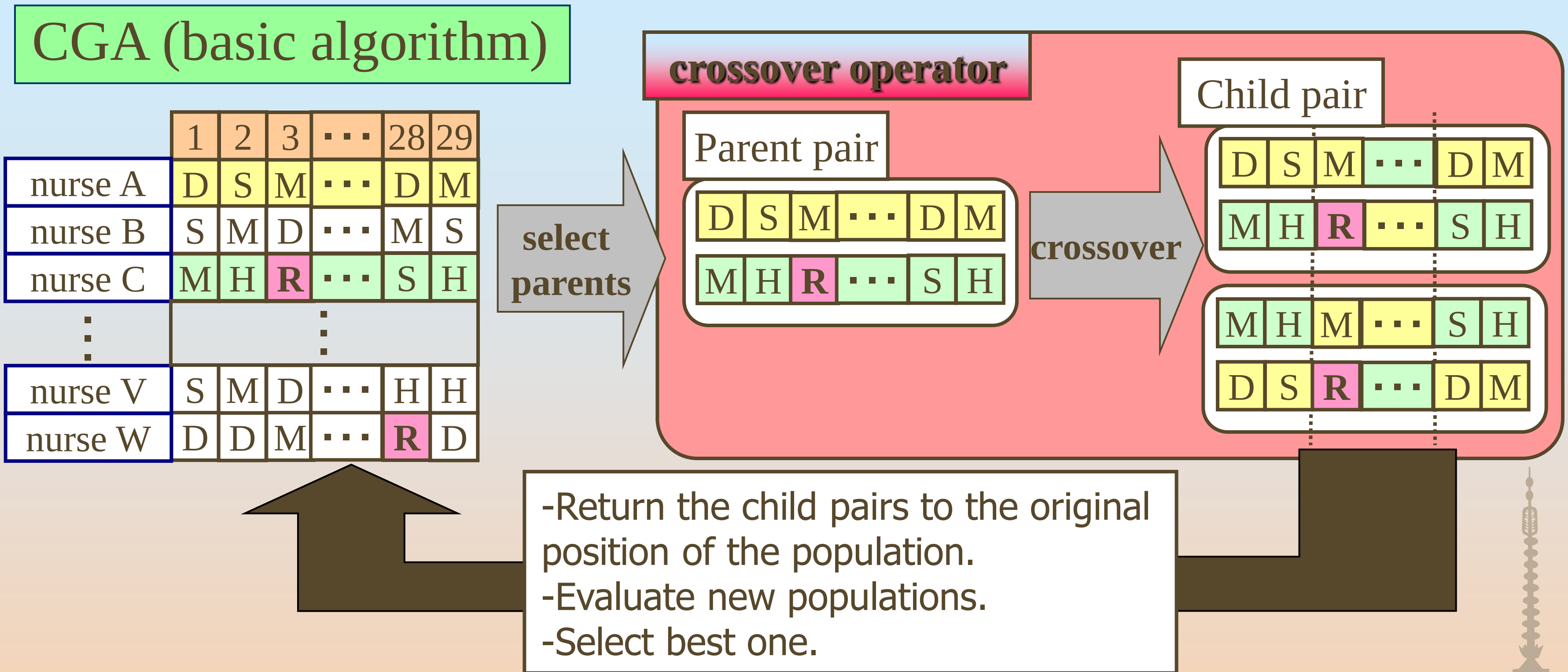
2. Nurse Scheduling

CGA (basic algorithm)



- The crossover operator selects two individuals, where one is selected by roulette selection manner and another is randomly selected.
- By the two-points crossover, two child pairs are regenerated.

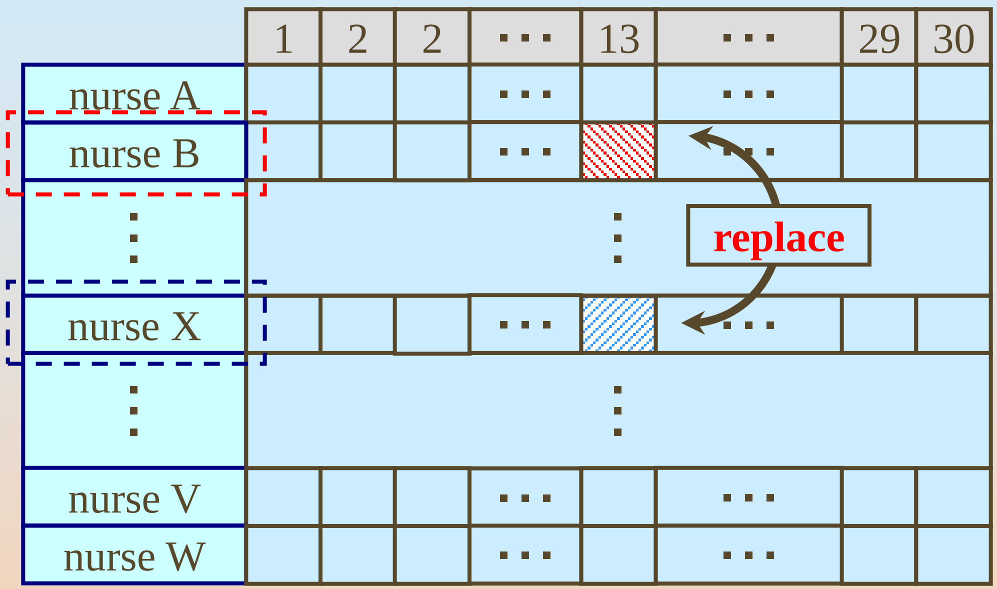
2. Nurse Scheduling



- Setting these new individuals back to the original positions of their parent, the population is evaluated by the penalty, E.
- This procedure is applied to 100 parent pairs in 1 generation.
- Therefore, 200 new populations are (locally) searched around the original population.

2. Nurse Scheduling

CGA (mutation operator)



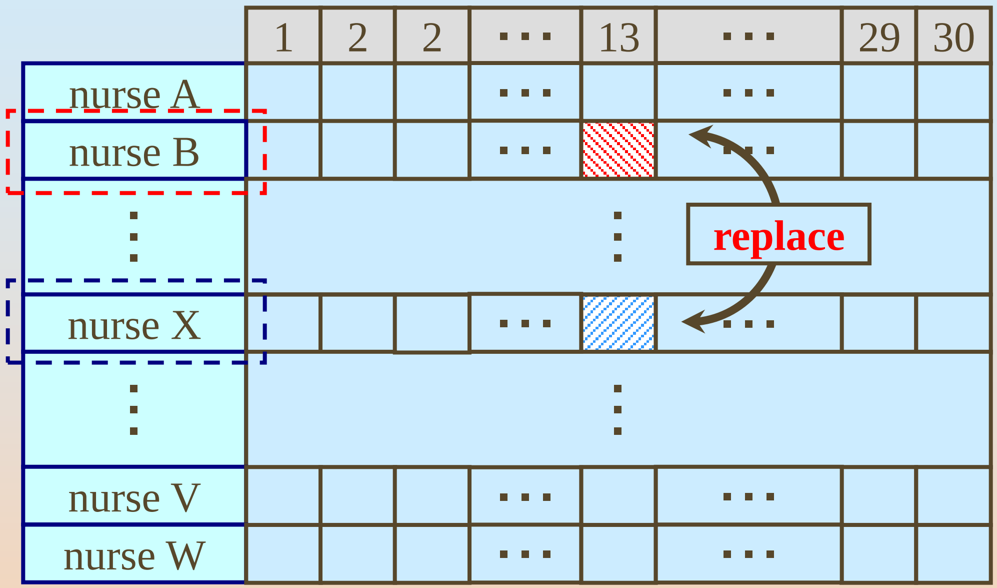
When the mutation is activated, the following procedures are executed.

① Randomly select one of duty dates.

② Randomly select two nurses. If one of them or both two are fixed duty, return to ①.

2. Nurse Scheduling

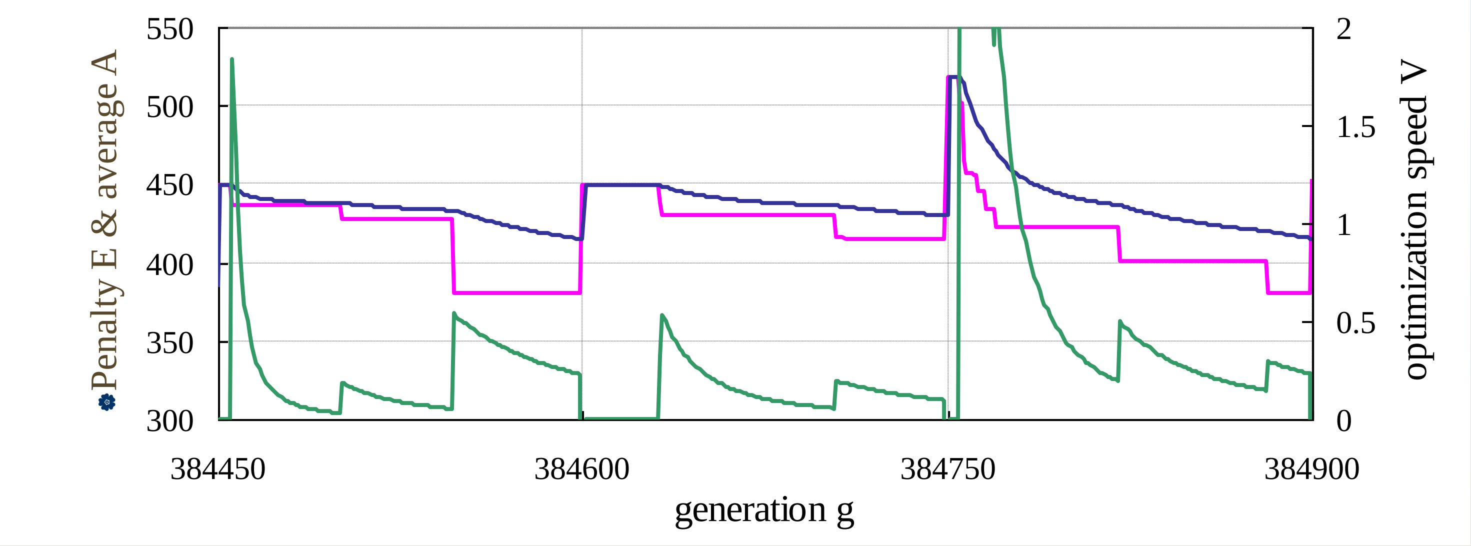
CGA (mutation operator)



③ Replace these two duty contents.

④ The new schedule provided in this way is generally worsened, but a global search is enabled by receiving this forcibly.

2. Nurse Scheduling



- Average value A of the penalty value for N_g generations after mutation,

$$A(g) = \frac{1}{N_g} \sum_{i=0}^{N_g - 1} E(g - i). \qquad (19)$$

- Optimization Speed V (Time difference of the average, A)

$$V(g) = A(g-1) - A(g) \qquad (20)$$

- Mutation is executed when optimization speed becomes less than a threshold ε.

$$V(g) < \varepsilon . \qquad (21)$$

- Optimization may not advance for several generations after the mutation.
- Then, the mutation is prohibited for an interval G_G right after the mutation.

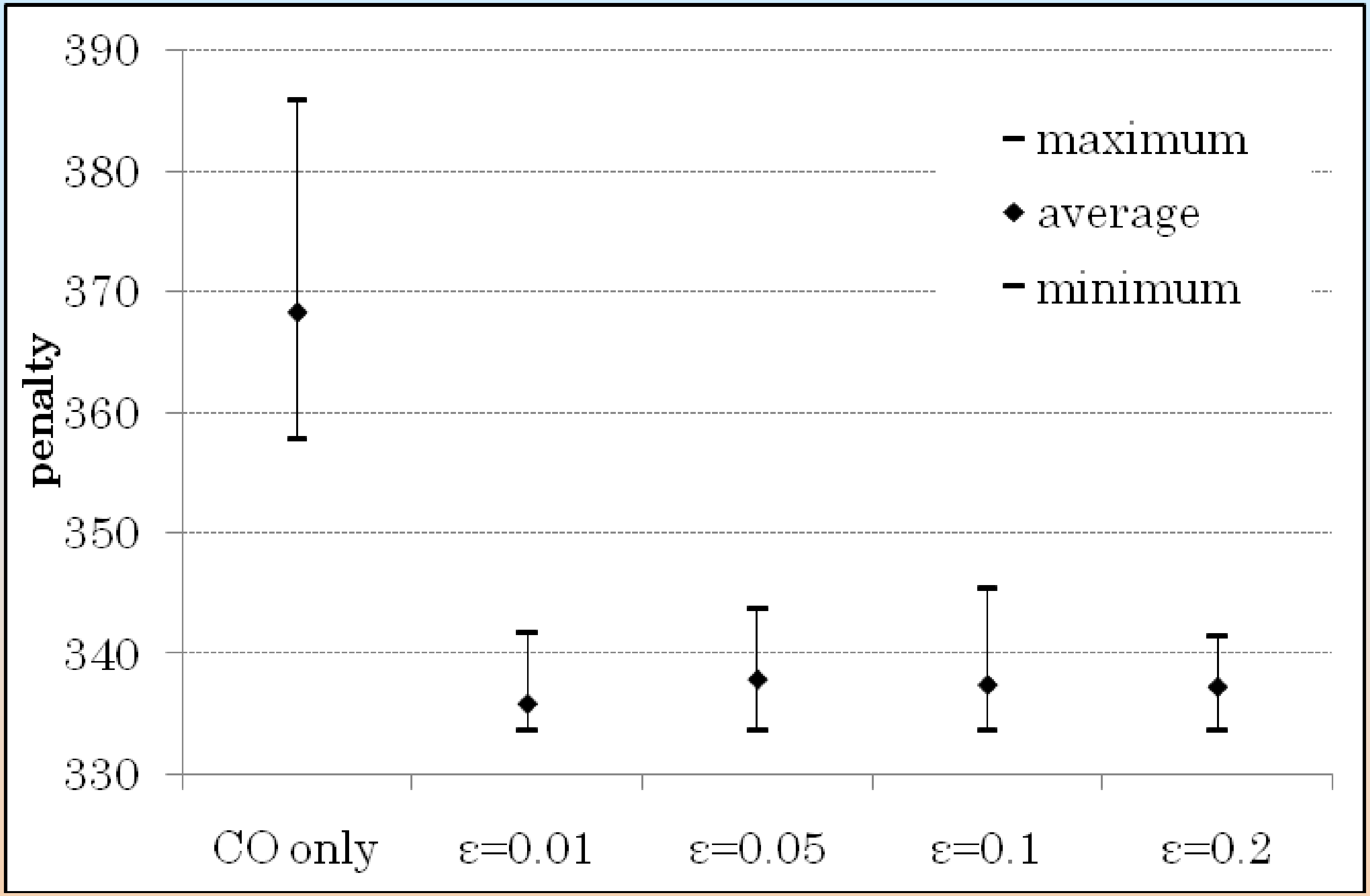
Guard Interval

3. Practical Experiment

3. Practical Experiment

- We have tried experiment of the nurse scheduling.
- The number of nurses : 23
- Two weeks have passed.
- Two replacement, one emergency attendance, one unplanned absence in the past two weeks.

- 1,000,000 generations are performed.
- ten time of the optimization are executed.
- The guard interval G_{G} is defined as 50.
- The speed threshold ε is tried from 0.01 to 0.2.



- The mutation operator provides better schedules than the case using only the crossover operator.

4. Improvement

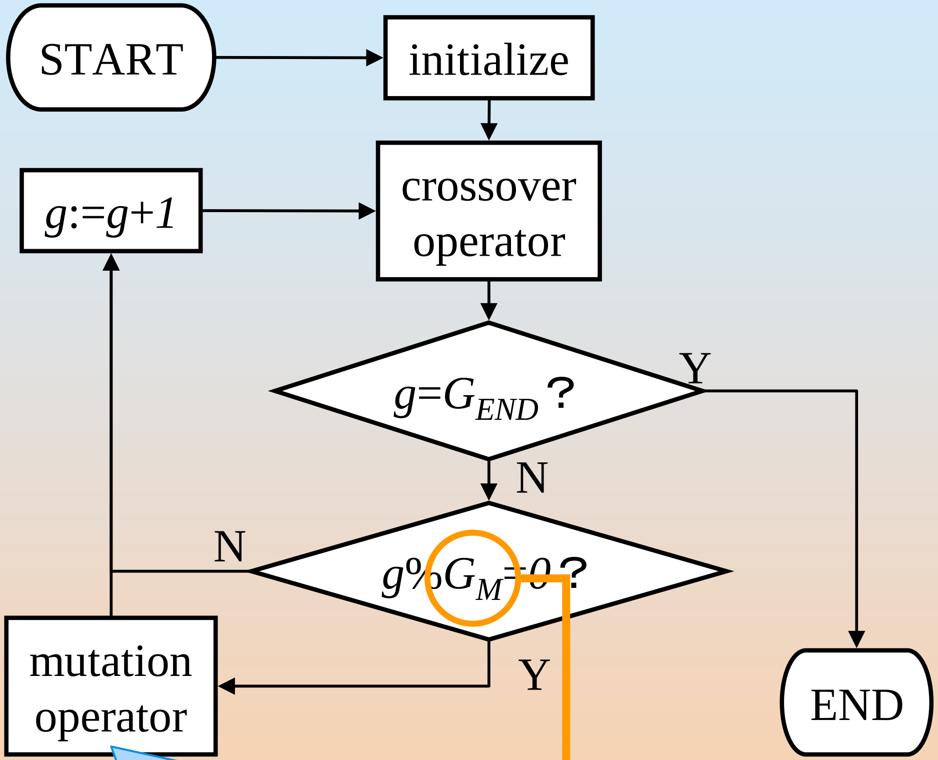
4. Improvement 4.1 Periodic Mutation Operator

- Problems of the conventional mutation operator
  - There are several parameters to define the mutation operator, G_G and ε.
  - A good result is not provided unless these parameters are defined adequately.
  - Actually, when the guard interval is defined as 50, the mutation with the threshold except the range from 0.01 to 0.2 does not provide good result.

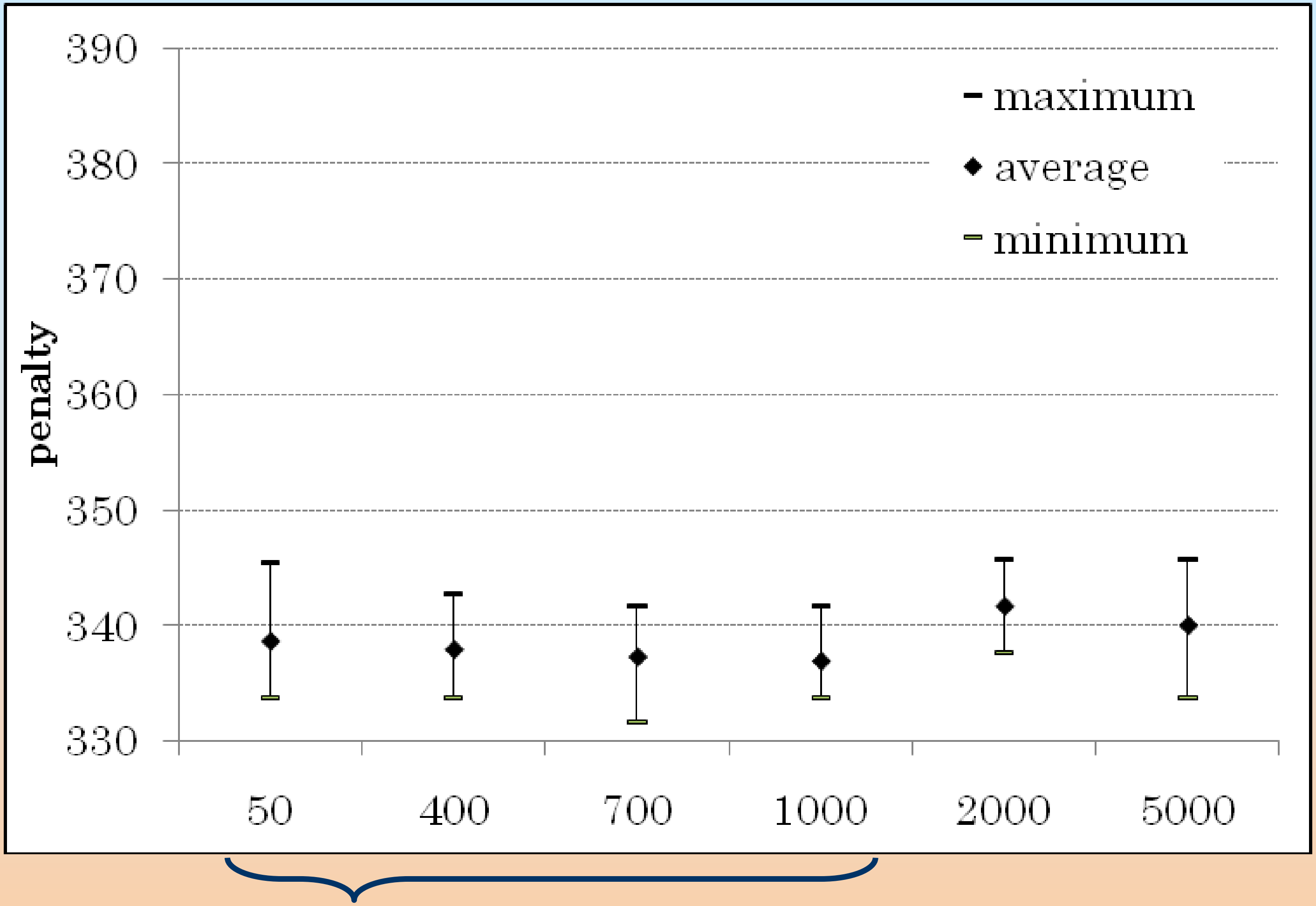
- Periodic Mutation Operator
  - Very simple algorithm
  - Only one parameter to define the mutation operator

4. Improvement 4.1 Periodic Mutation Operator

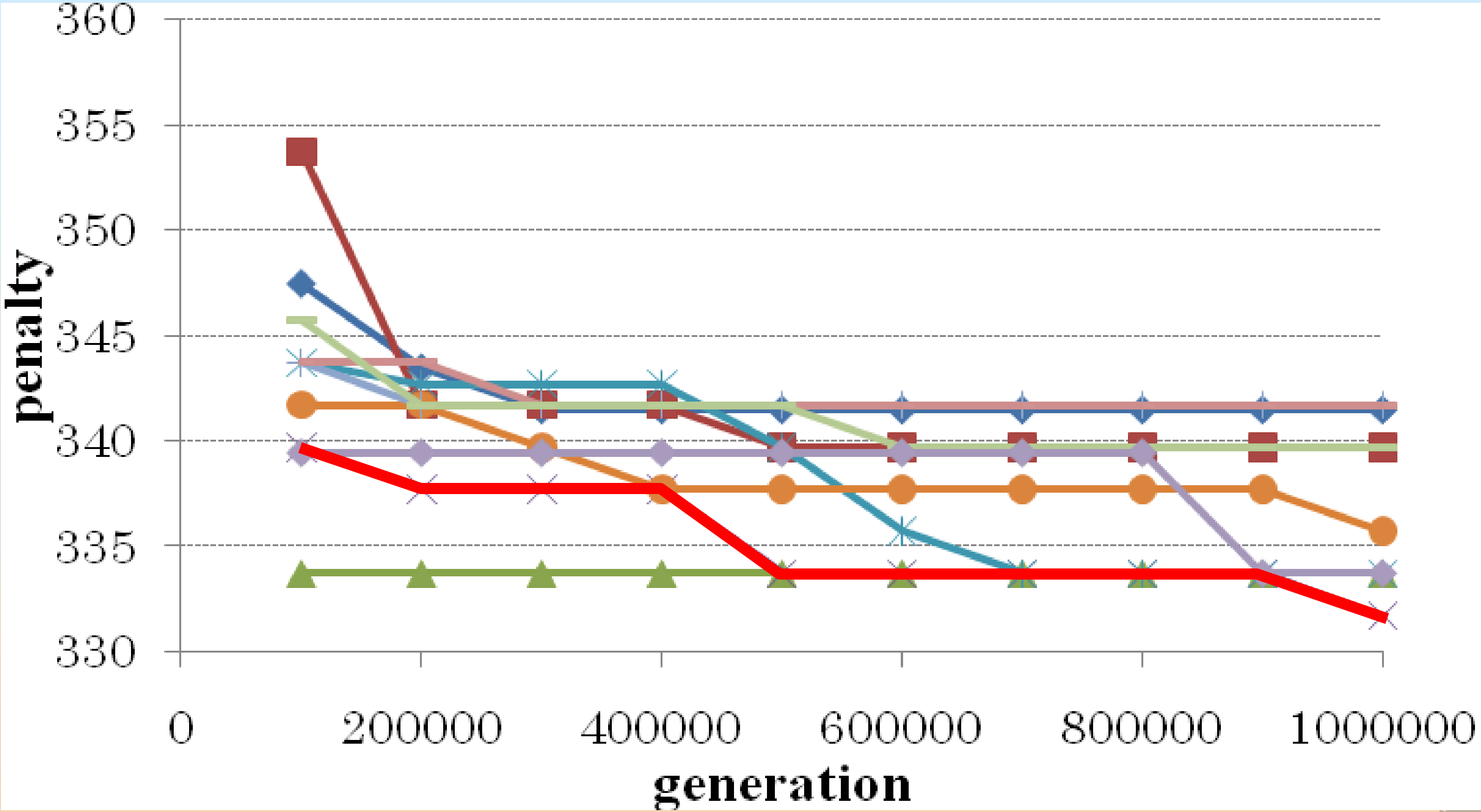
Algorithm of the periodic mutation operator



❁ This mutation is activated periodically with the mutation period G_M.



- The mutation period is effective on wide range from 50 to 1000.
- This means that we do not need to mention the period too much.



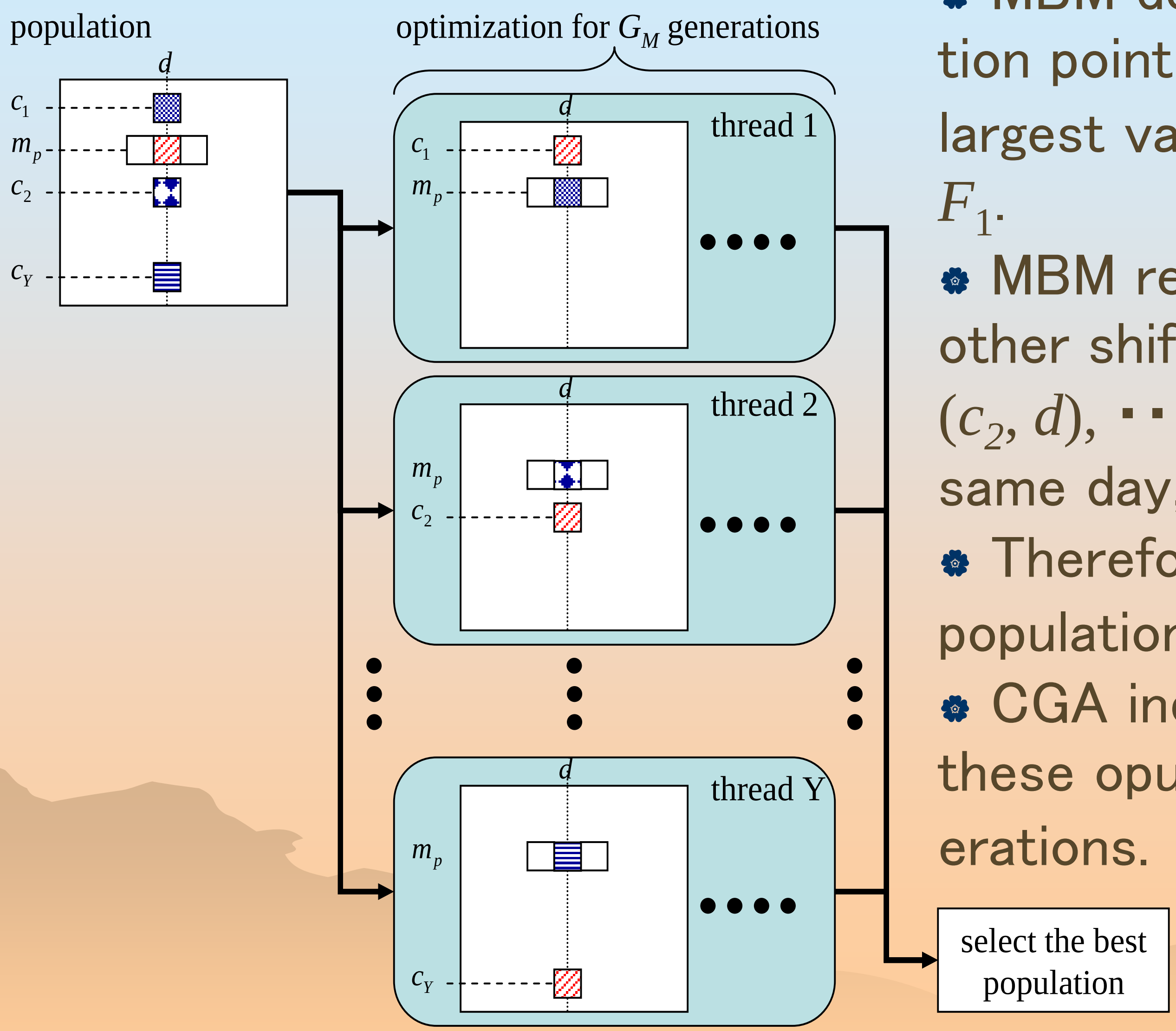
- This figure shows optimization progresses of ten trials when the periodic mutation operator with the period, 700.
- This case has provided the best result in our experience.

4. Improvement 4.2 Multi-Branched Mutation

- In our investigation, 200,000 generations are enough to simply optimize a schedule for four weeks.
- The nurse scheduling including the changes in the past weeks is becomes very difficult.
- Then we have investigated about optimization progress in detail.

- The optimization sometimes stagnates even when the periodic mutqation operator is applied.
- Only the F_1 has a big value in many cases whereas a penalty except the F_1 becomes approximately zero.

Multi-Branched Mutation (MBM) Operator



- MBM determines a mutation point, (m_p, d), giving the largest value of the penalty F_1.
- MBM replaces the point to other shift candidates, (c_1, d), (c_2, d), ･･･,(c_X, d) on the same day, d.
- Therefore, X new mutated populations are regenerated.
- CGA individually optimizes these opulations for G_M generations.

Decision of the shift candidates, $(c_1, d), (c_2, d), \cdots, (c_X, d)$

MBM1

- The mutation position, (m_p, d), is tried to replaced with the all feasible candidates $(c_1, d), (c_2, d), \cdots, (c_X, d)$.
- All the populations are forcibly accepted.

MBM2

- The mutation position, (m_p, d), is tried to replaced with the N_{MBM} feasible candidates, (c, d), giving the larger Z.

$$Z(c,d) = p_{cd} + f_{2cd} + f_{3cd} + f_{6cd} \quad (j = 1,2,\ldots,D) \qquad (22)$$

MBM3

- The mutation position, (m_p, d), is tried to replaced with the N_{MBM} feasible candidates, (c, d), giving the larger p_{cd}.

4. Improvement 4.3 Parallel Processing

Problem

- By executing our program, one optimization for 1,000,000 generations takes more than 100 minutes.
- As shown in the results of the periodic mutation, 10 time of optimization are necessary to provide a good schedule.
- Enormous computational cost is required.

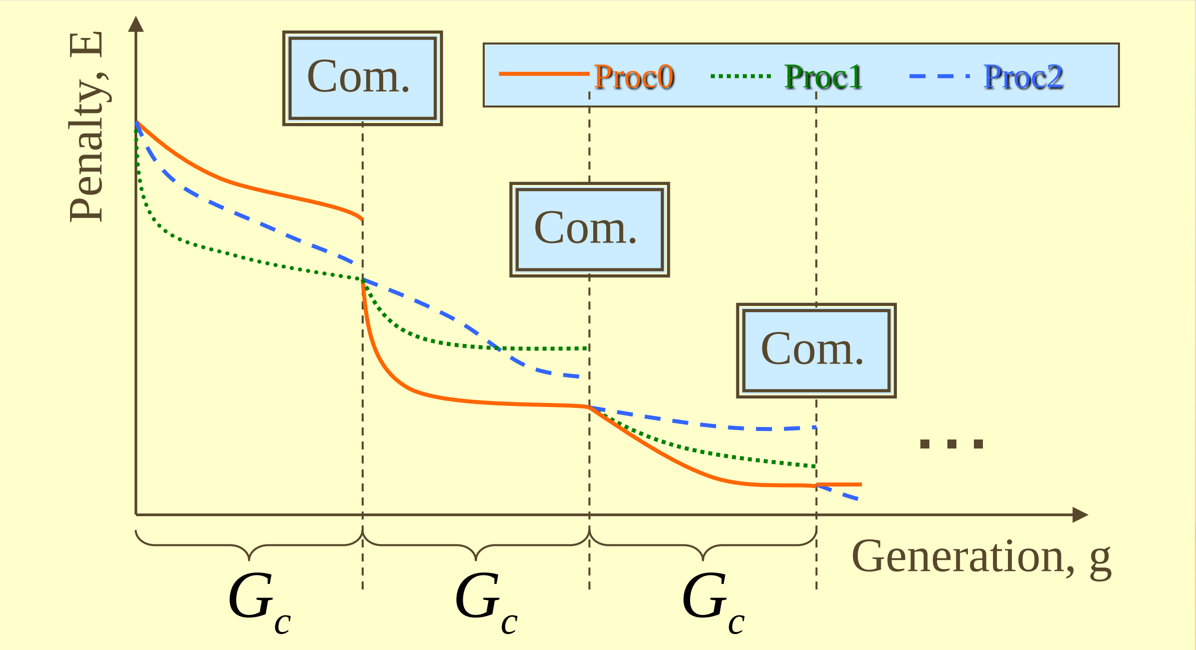
Background

- In most hospitals of our country, ten or more PCs are connect-ed each other via LAN in each section.
- In the night time, almost PCs are sleeping or under shutdown.

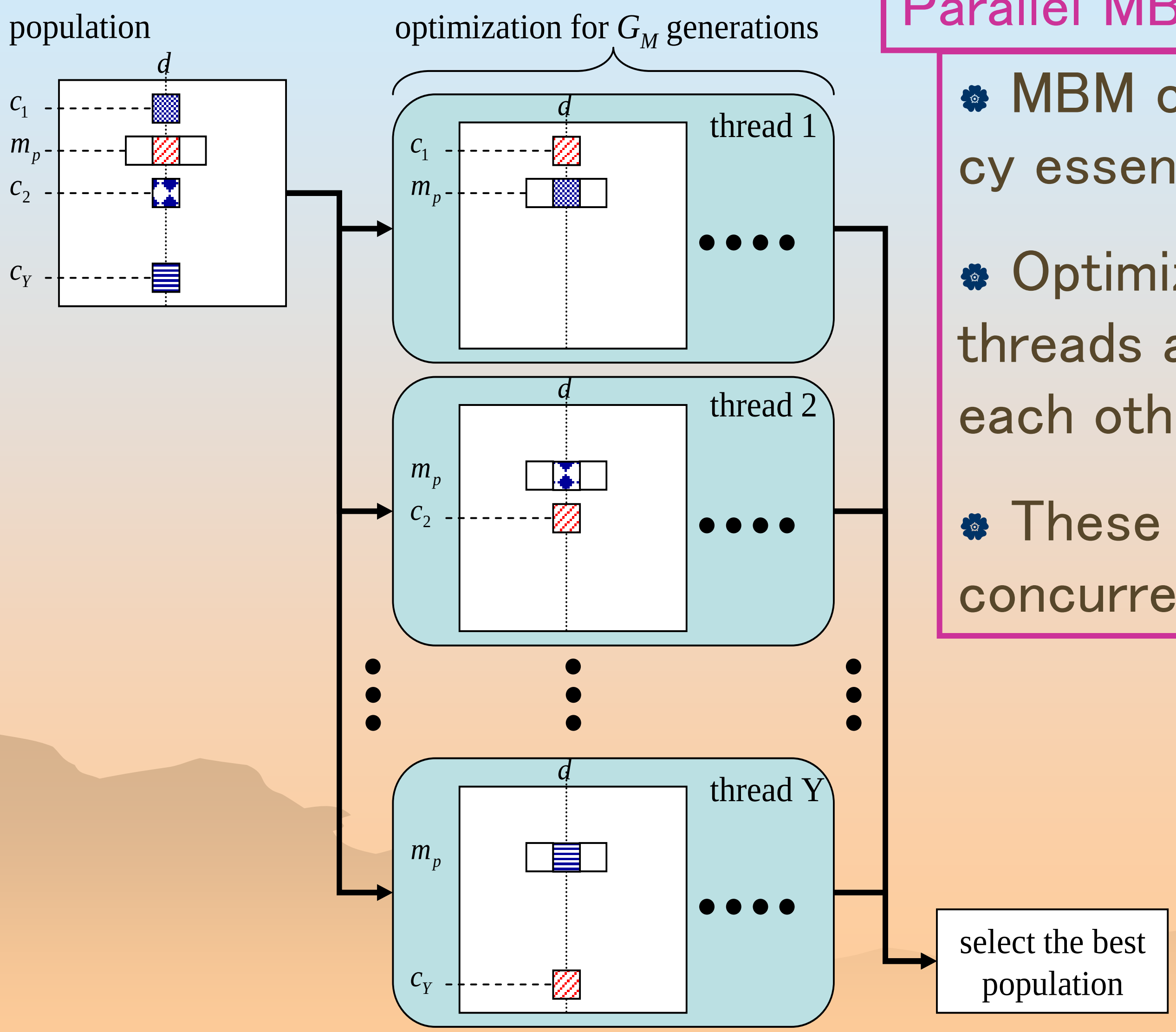
4. Improvement 4.3 Parallel Processing

Previous Work

- We have proposed simple parallel processing technique for the nurse scheduling by using CGA.
- Several threads are generated on the several PCs. And these threads communicate every G_C generation period each other and share the best schedule.



4. Improvement 4.3 Parallel Processing



Parallel MBM

- MBM contains concurrency essentially.
- Optimization on these threads are independent each other.
- These thread can be concurrently executed.

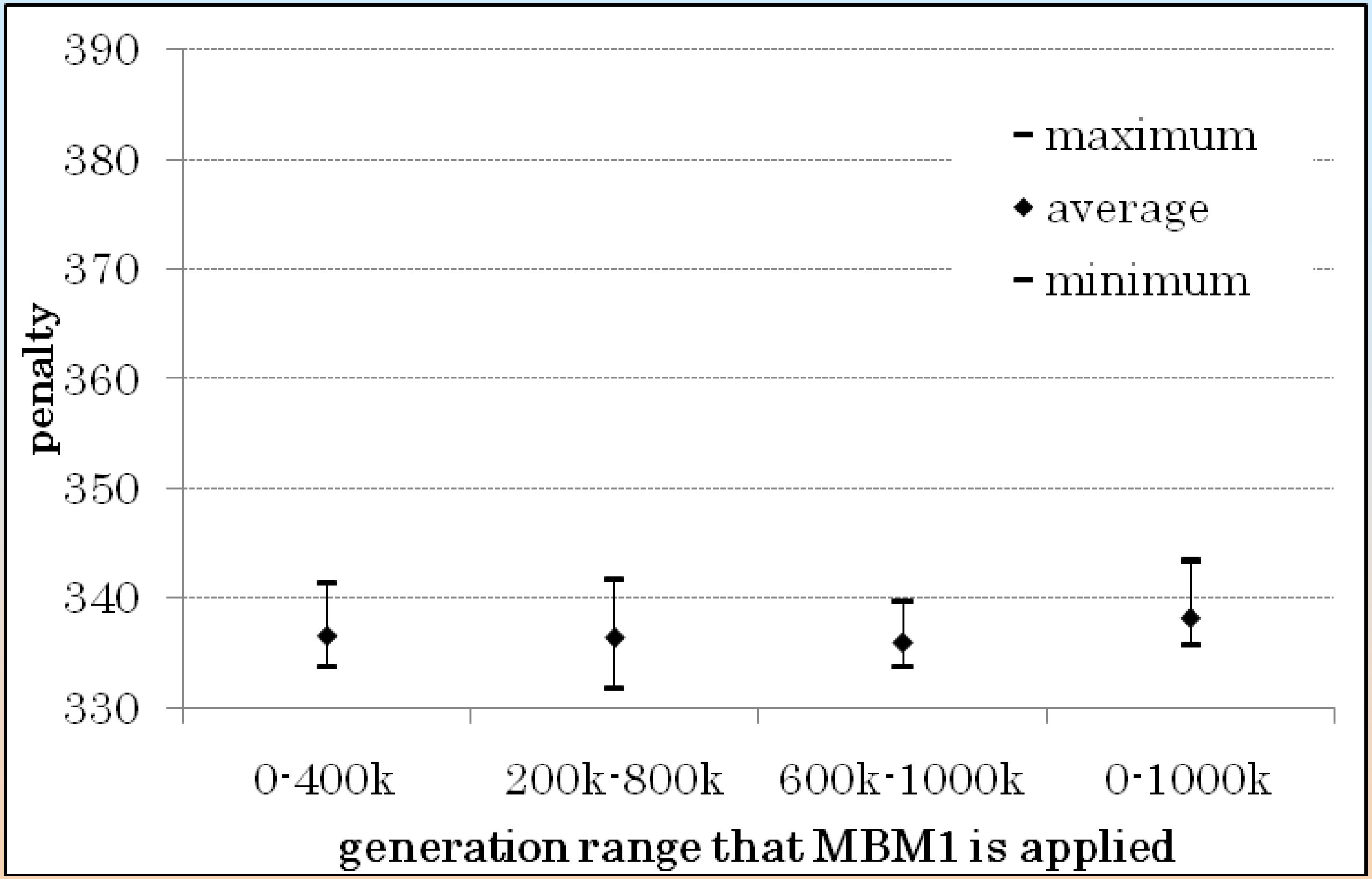


Fig.11. Optimization results by using MBM1 with several generation ranges that MBM1 is applied.

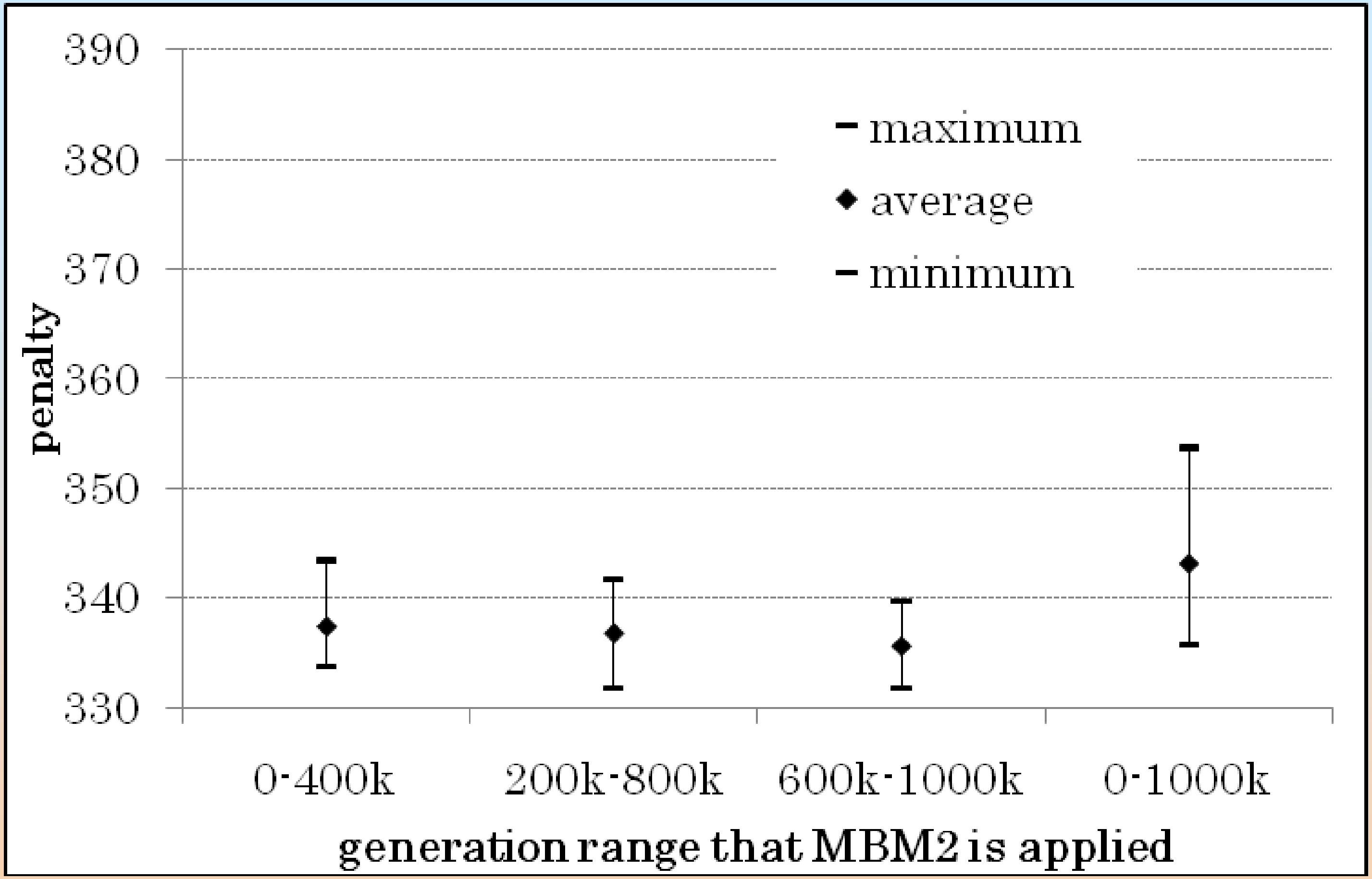


Fig.11. Optimization results by using MBM2 with several generation ranges that MBM2 is applied.

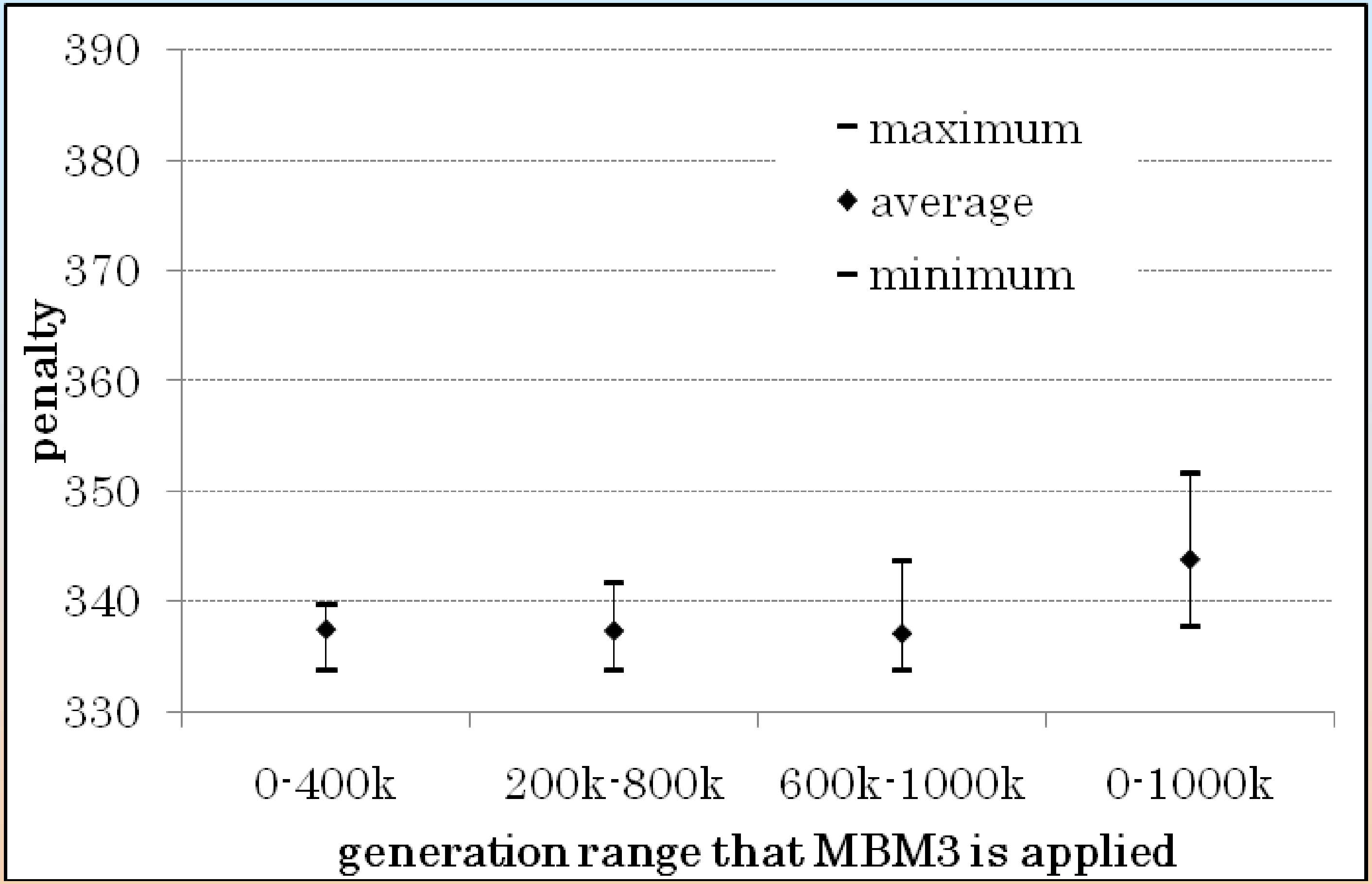
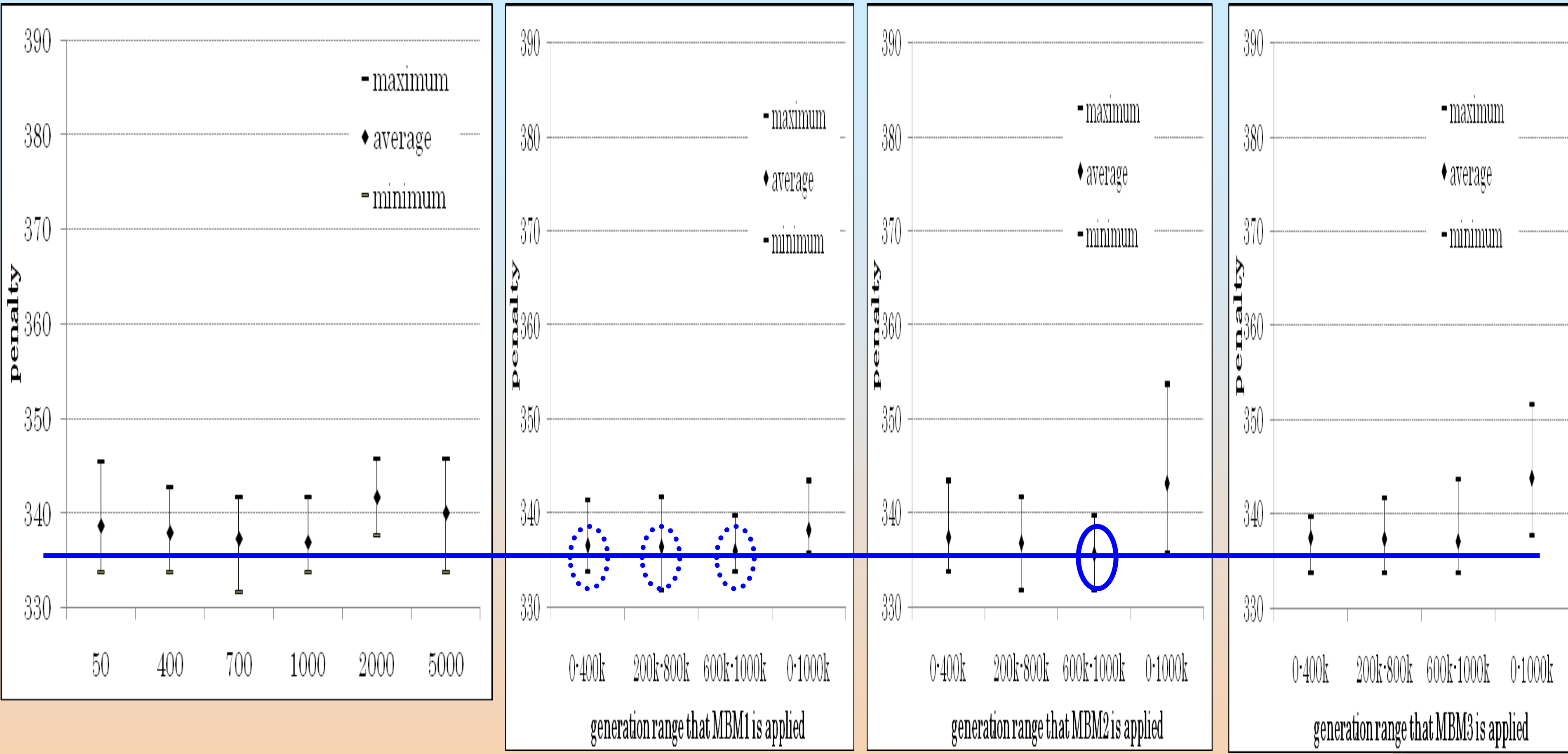


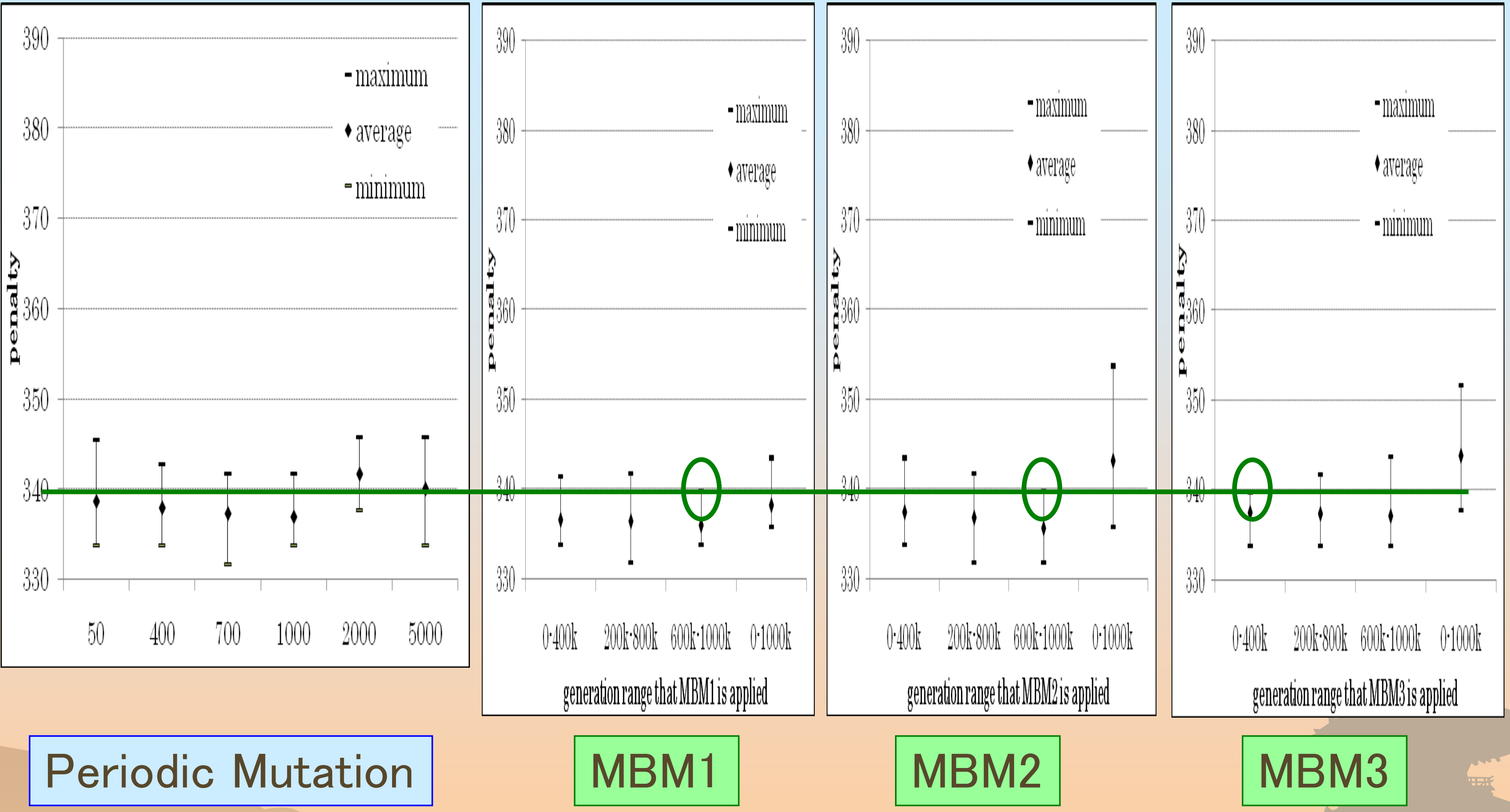
Fig.11. Optimization results by using MBM3 with several generation ranges that MBM3 is applied.

The red positions denote the best results in our experience.
And the dotted circles denote the sub-optimal schedule.

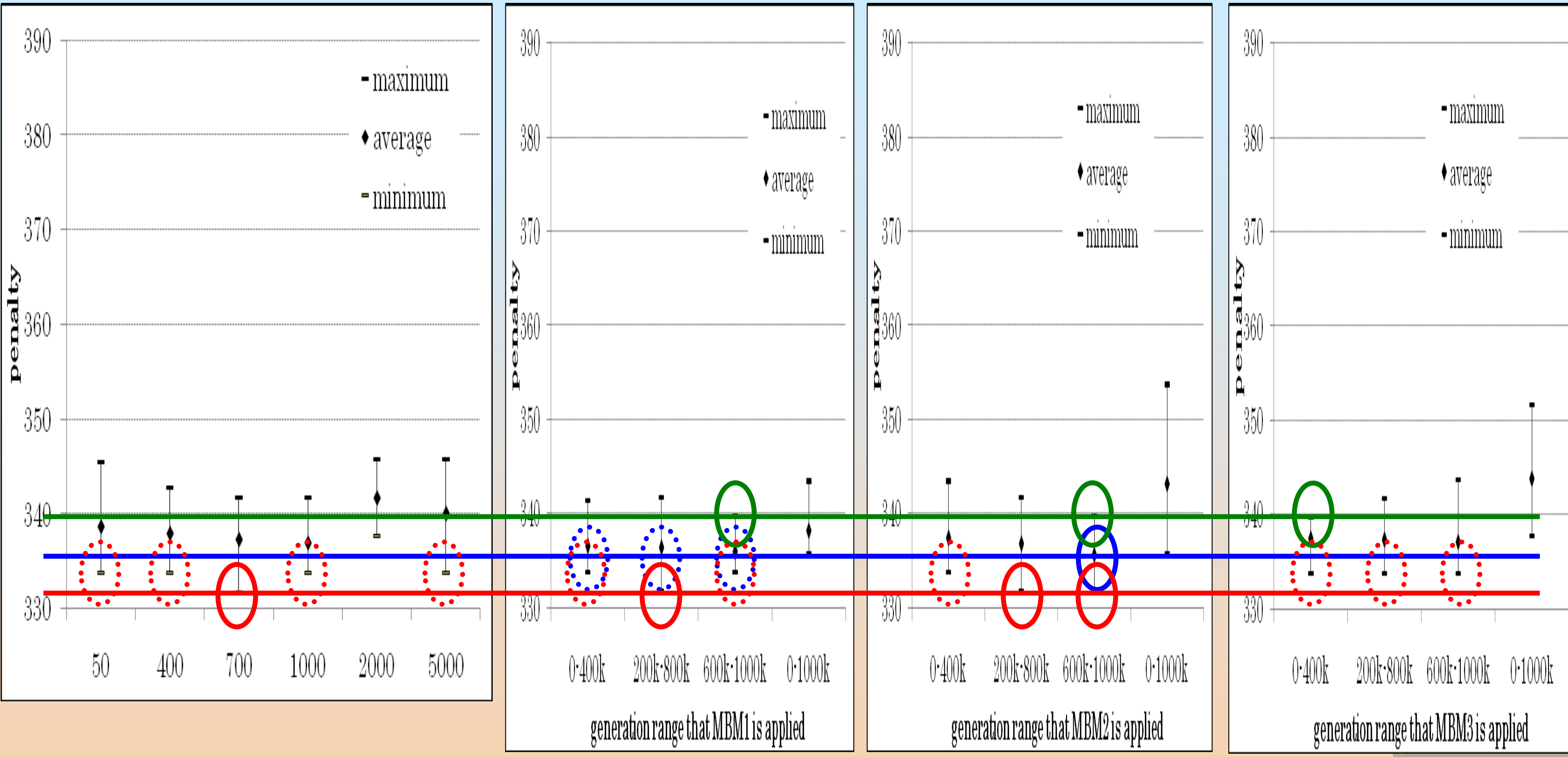


Periodic Mutation | MBM1 | MBM2 | MBM3

The blue position denotes the best average.
And the dotted circles denote averages near the best one.



The green positions denote the smallest maximum value.



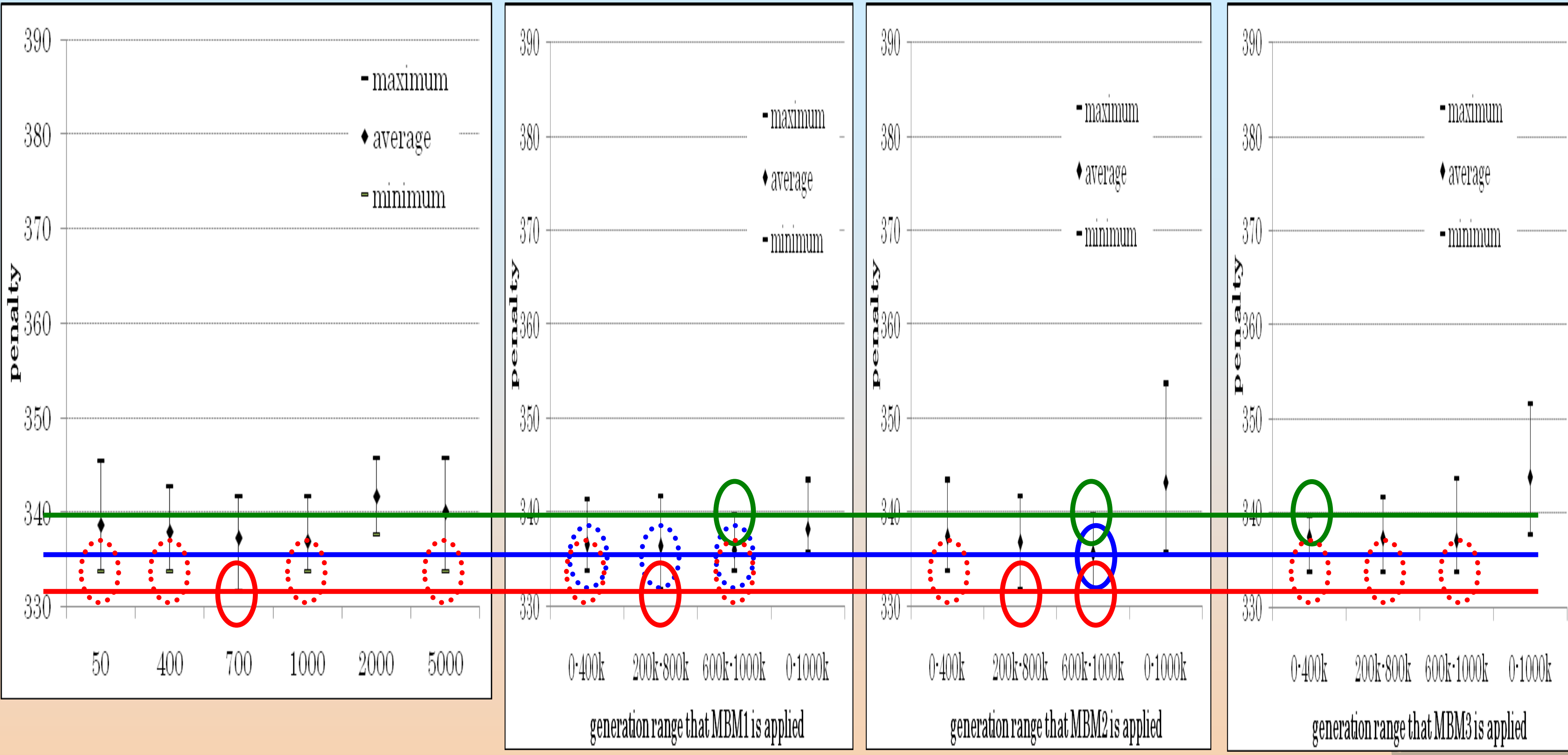
Periodic Mutation

MBM1

MBM2

MBM3

In any cases, applying **MBM** to whole generations yields unfavorable results.



Periodic Mutation

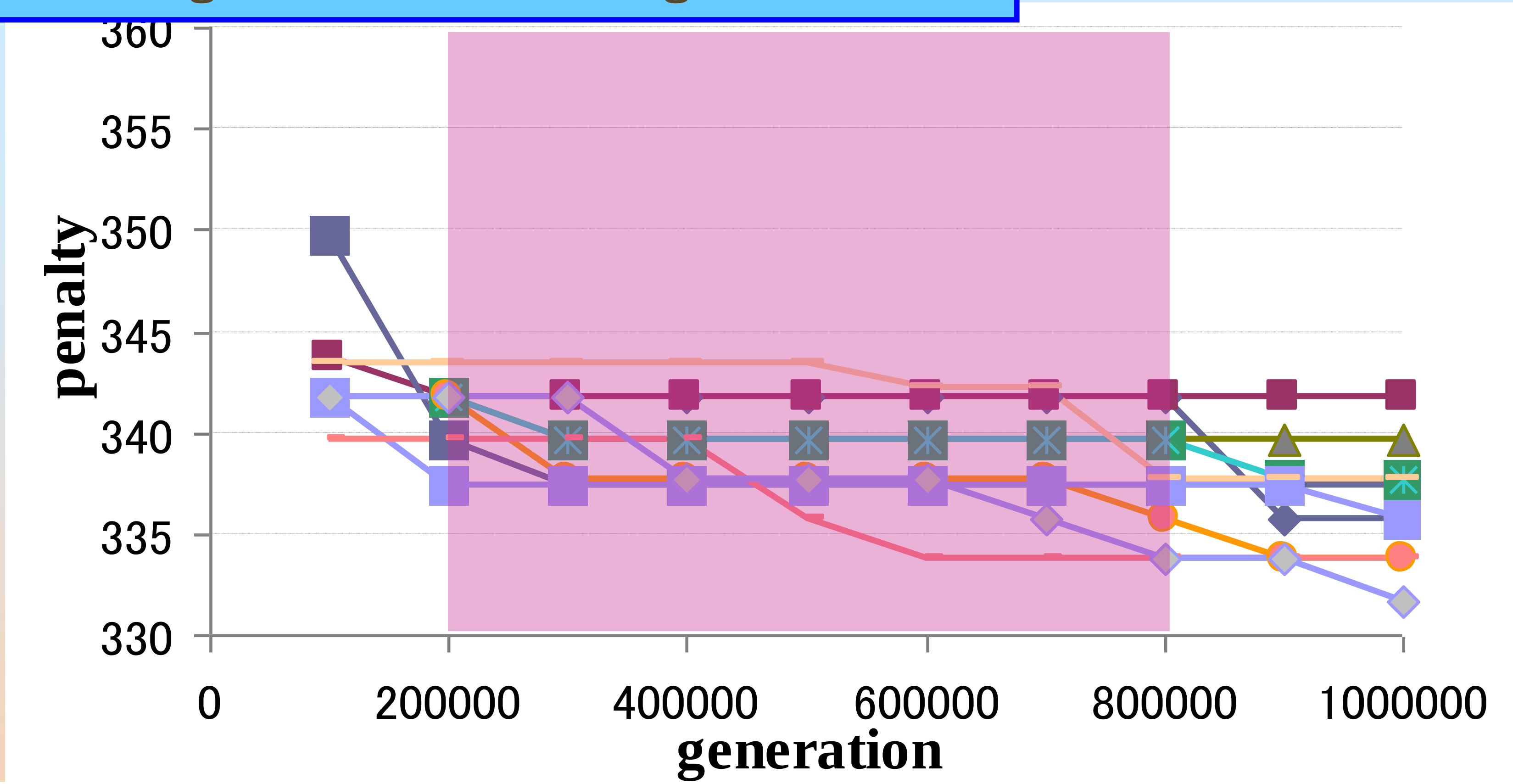
MBM1

MBM2

MBM3

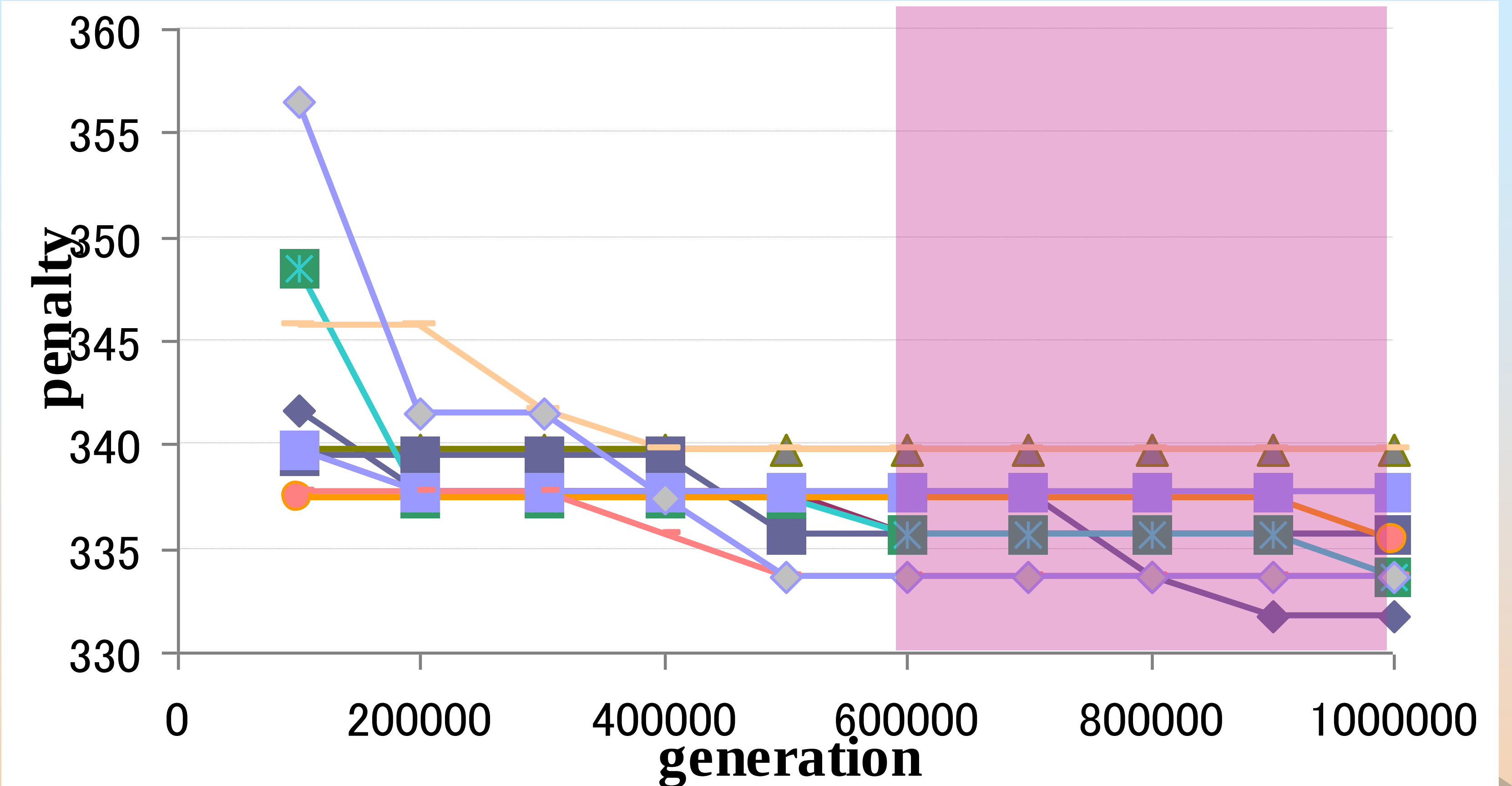
When **MBM2** is applied from middle stage of the optimization to the end stage, effective results are provided.

Shortening of the number of generations



Optimization results of ten trials when **MBM1** is applied from 200k to 800k generations.

- In the generations that applied MBM, penalties decrease remarkably. Besides, 600k generations are enough to provide almost best schedule.



Optimization results of ten trials when MBM2 is applied from 600k to 1000k generations.

- In the generations that applied MBM, the penalties decrease remarkably.

5. Conclusion

5. Conclusion

- In this research, we have treaded the nurse scheduling by using **CGA**.
- We have defined the penalty function to deal when a shift schedule has been changed in the middle of a month.
- The nurse scheduling including such changes become difficult, and the enormous generations has been necessary to provide a good schedule.
- To provide the difficulty, we have proposed the mutation operator depending on the optimization speed.
- The mutation operator has two parameters to be carefully defined.

5. Conclusion

- Then, we have proposed the simple **periodic mutation** operator.

- And we have proposed **MBM** operator to improve the value of penalty, F_1, where a penalty except the F_1 becomes approxi-mately zero in many cases.

- Furthermore, the **parallel processing** of the MBM is imple-mented based on the natural concurrency of the MBM.

Thank you very much
for your kind attention!