Title	Structured Approach for Local Clustering Organization
Author(s)	今野, 陽子
Citation	北海道大学. 博士(情報科学) 甲第11745号
Issue Date	2015-03-25
DOI	10.14943/doctoral.k11745
Doc URL	http://hdl.handle.net/2115/58698
Type	theses (doctoral)
File Information	Yohko_Konno.pdf

Structured Approach for

Local Clustering Organization

Graduate School of
Information Science and Technology
Hokkaido University, Japan

Yohko Konno

Submitted in partial fulfillment of the requirements for
the Degree of Doctor of Information Science and Technology,
Hokkaido University,
March 2015

Contents

List of Figures

List of Tables

Preface

In the last years, advancement demands of decision-making are getting so important on large-scaling of information technology. Combination optimization problems are one of important problems appearing in different situation ranging from the industrial network in the physical society to application of science and technology. And those are discrete system problems that the computational complexity of solution increases with factorial order for a problem scale. For combination optimization problems, an approximate algorithm is adopted as a solution when we need getting a highly quality solution in real time when an issue of practical society in particular, in consideration of enough quality, a design possible to update and analysis is required. Then, as a guidance of a general technique to design of the quality, a technique of a diversification and an intensification of exploration are important [Kubo 09, Nonobe 08]. Those are to change the structure of current solution within no-search area, and also to search the neighborhood that is similar structure of current solution.

This dissertation focuses on an applied scheduling problem in a combinatorial optimization problem in the supply-chains, and it focuses to “Local clustering organization” (LCO) that is applicable to them. LCO developed as one of heuristics, an “Iterated local search” (ILS) for large-scale TSP [Frukawa 05-1, 05-2]. According to this algorithm for TSP, several application studies based on LCO have been proposed, i.e. “Multiple traveling salesman problem” (m-TSP), “Vehicle rooting problem” (VRP), “Job-shop scheduling problem” (JSP) [Frukawa 06, Konno 09], and “Order picking problem” (OPP) [Iwasaki 13].

LCO is reasonable for implementation to general scheduling optimization problems, it has a simple framework developed based on the principle of “Self-Organization

Map” (SOM) [Kohonen 89, 01, 07] for TSP. We focused on the high speed performance of LCO. LCO succeeded to the characteristic of global optimization that neuronal dynamics of SOM brought. As for the search of a local neighborhood based on Riccatti-type learning law, SOM had the function of an intensive search. Furthermore, the expansion to LCO became able to apply to larger-scale problems than SOM, by succession of high-speed performance of SOM. However, there were some works that has not been yet researched. LCO's present behavior is not clear what kind of operation is effective at what kind of situation. LCO has the defect of operation that was easy to be caught in a local minima. LCO has a possibility of performance to make up by control of the intensive exploration that computes on a large search-pace effectively for node-based operation. The goal of this dissertation constructs a design methodology to make a performance enhancement of the local clustering which is LCO search operator. The discussion is divided into two phases. (1) A case of extension of variation of the operation is “Extended local clustering”. (2) A case of intensive search for a search neighborhood of a solution by the extension of an operation is “Structured local clustering”. Each mission is approached as follows. (1) For an expansion local site clustering, it approaches by introduction of an operation based on insertion method. (2) For a structured local clustering, it approaches by extracting a structural partial solution element depending on a characteristic of the mechanism of improvement. In (2), as a frame catching solution structurally, it considers solution expression to be comprised of the subset (node-set) of the solution element (simple node). The operation of the search enable with maintaining a set for the node-set. They are applied to TSP and Job-shop scheduling problem (JSP) and show the results those experimented on benchmark problems. This dissertation is organized about the approach and results of inspection experiments for issues of LCO as follows.

Chapter 1 describes an explanation of the original LCO algorithm developed based on the principle of SOM for TSP., and the issues in the conventional study and a related study. LCO is one of meta-heuristics, but the systematic approach was not discussed as one of meta-heuristics, how to plan the diversification and intensification of search. Therefore, in addition of the explanation of algorithm, for upgrade of the relation of LCO and meta-heuristics, it is given about the various problems that we must investigate meta-heuristic technique.

Chapter 2 gives a theory and method for design of operation of LCO for application to combinatorial optimization problems. This chapter produces the frame of total local clustering design through the principle of LCO operation, and analysis of the original operation, then explains structured local clustering. First, we define the framework of patterns of local clustering of LCO operator based on SOM's learning law. The positioning of original LCO's operator is analyzed using this framework. Then the extended operation is designed by using the definition. It brings to the LCO operator the combination with the variation of operation. As the second, the structured local clustering is explained for bringing an intensive exploration with reduction of the search space by corresponding structure of problems.

Chapter 3 shows the suggestion of structured local clustering in TSP and the results of inspection experiments. TSP is a problem to minimize movement cost of all cities, and the encoding of LCO is sequence of the city number. In structured local clustering operator, LCO extracts the partial tour that is immobilized locally on a search process. Then it considers the set of the city simple substance to connect between partial tours is as a bridge, and the solution is divided to those parts. In addition, it introduces the insertion method and applies the extended operation for the recombination. This operates local clustering in a solution structure to be mixed of node and node-set, and concentrates to the search in bridges.

Chapter 4 shows the suggestion of structured local cluster ring in JSP and the results of inspection experiment. JSP is a problem to assign plural jobs with process precedent relations to a resource by the most suitable order, and the encoding of LCO is sequence of the multiplex model of the job number. In structured local clustering, it makes clear that the exploration search for the job sequence of each resource in interpretation from the analysis of the behavior. Then the solution structure is extended in a phenotype of the resource division and makes a centralization of search.

Finally, the conclusion is stated in Chapter 5.

Chapter 1

1. LCO based on SOM Principle

1.1. Introduction

Heuristic is applied to combinatorial optimization problems for the aim of getting a high quality solution in real time. The guiding principle of design of meta-heuristics that heuristics are colligated organically is the strategy of the search neighborhood, i.e. the technique of "diversification" and "intensification" of search is required [Kubo 09, Nonobe 08]. The diversification is an operation to change the structure of an existing solution and to search in the non-search area. The intensification is an operation to search for the solutions with similar structure of an existing solution. This dissertation discusses about the design method of operation of "Local clustering organization" (LCO) [Frukawa 05-1, 05-2]. For this study, as the first, this chapter provides explanation of the LCO algorithm developed based on the principle of "Self-organization maps" (SOM) [Kohonen 89, 01, 07] for TSP. LCO is one of meta-heuristics, and studies as hybrid heuristics with other single-solution-based and population-based meta-heuristics. However, the systematic approach is not discussed, how to plan the diversification and intensification of search. Therefore, in addition of the explanation of algorithm, for upgrade of the relation of LCO and meta-heuristics, it is given about the various problems that we must investigate meta-heuristic technique.

This chapter is organized as follows. Section 1.2 introduces the background of the study of LCO, and the relations of LCO and the meta-heuristics, and positioning. Section 1.3 defined TSP based on the development of LCO. In Section 1.4, SOM principle for TSP is explained. Section 1.5, the LCO algorithm based on

SOM is explained. In Section 1.6, the local clustering of the LCO operator is explained. The problem of LCO on meta-heuristics is discussed in Section 1.7. Finally the conclusion is stated in Section 1.8.

1.2. **Background**

LCO was developed as an "Approximate algorithm" to solve large-scale TSP with high-speed. The findings led the accuracy nearly 1.1% in large-scale TSP in several seconds. The principle of LCO based on SOM for TSP, and SOM is one of "Artificial neural network" (ANN) [Furukawa 05-1, 05-2]. It is the algorithm which is used the mechanism of improvement of solutions giving the action of local clustering and self-organization. When SOM applies to TSP, city coordinates are defined input vectors as the reference. But LCO refers the distance between 2 cities. In the general TSP, the cost as a traveling or a scheduling is not proportional to the distance defined by city coordinates, that is, the cost between 2 cities is not defined by the city coordinates. LCO is able to apply to general TSPs using their costs, no matter how to costs given.

SOM minimizes tour globally by repetition of the local attraction of vector signals, by the global potential-function is chosen stochastically. This learning causes a local extraction stochastically and clustering, by referencing costs between 2 cities to determine the sequence of cities. Thus global optimization is attained. Such a phenomenon is observed frequently in the natural network. This mechanism applied to LCO operator for the optimization which leads by continual sorting to minimize cost locally.

LCO and meta-heuristics:

LCO that has such an algorithm is one of meta-heuristics. The LCO algorithm was inspired from SOM, and it is one of "Variable neighborhood search" (VNS) [Boussaïd 13, Mladenovic 97] in the single-solution based search method. In this VNS, "Lin-Kernighan algorithm" is included, that is famous for TSP solver [Lin 73, Helsgaun 00, 09, Yoshihara 09]. LCO is possible to use by combination with some meta-heuristics that is single-solution based method and population-based method for hybrid methods. “Local search” (LS), “Iterated local search” (ILS), “Tabu search” (TS), “Simulated annealing” (SA), “Guided local search” (GLS), “Multi level method” (ML), “Greedy randomized adaptive search procedure” (GRAS), and so on, are included in the single-solution based method. Population based methods has “Evolutionary computation” (EC) and “Swarm intelligence” (SI), and so on. For example, “Genetic algorithm” (GA), “Evolutionary programming”, “Scatter search” (SS), “Ant colony optimization” (ACO) and “Particle swarm optimization” (PSO). Those meta-heuristics are designed based on the strategies of the neighborhood is brought by the solutions search. Then for the design of those meta-heuristics, a technique of a diversification of search and an intensification of search is required.

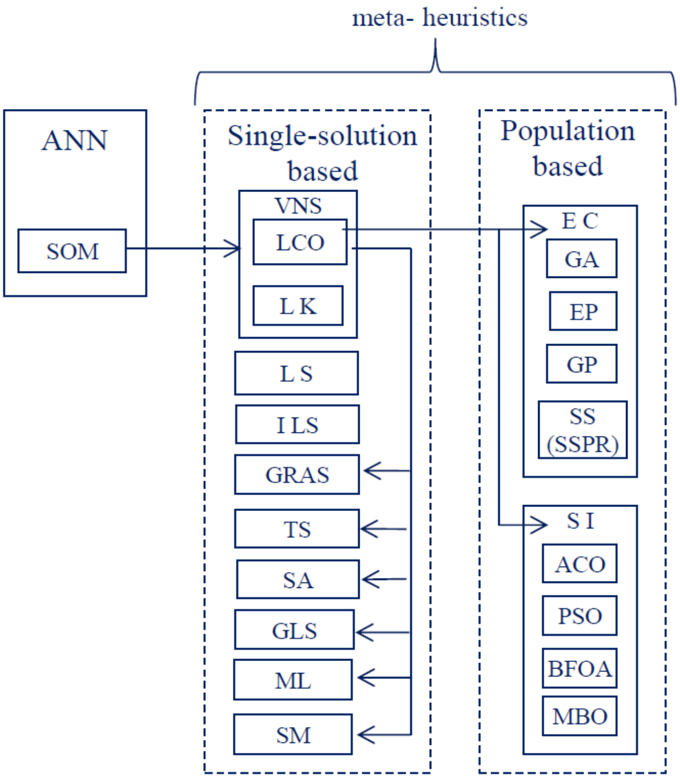


Figure 1. 1 LCO and meta-heuristics.

Diversification and intensification:

An examination of design of diversification and intensification for LCO is this dissertation's main theme. When the search space of all solutions *P* in one combinatorial optimization problem is as *F*, then the search area ***F***' of LCO is limited. There is a globally optimal solution is *P** is the outside of ***F***' mostly. Then the diversification is an operation to change the structure of the solution and to search in non-search area. The intensification is an operation to search for solutions with similar structure. In original LCO, it is unknown whether operation of the local cluster ring is enough, and it is not examined about intensification. This study suggests the design method of the operation that the aim is a diversification and centralization of LCO and shows the efficacy by using this design method.

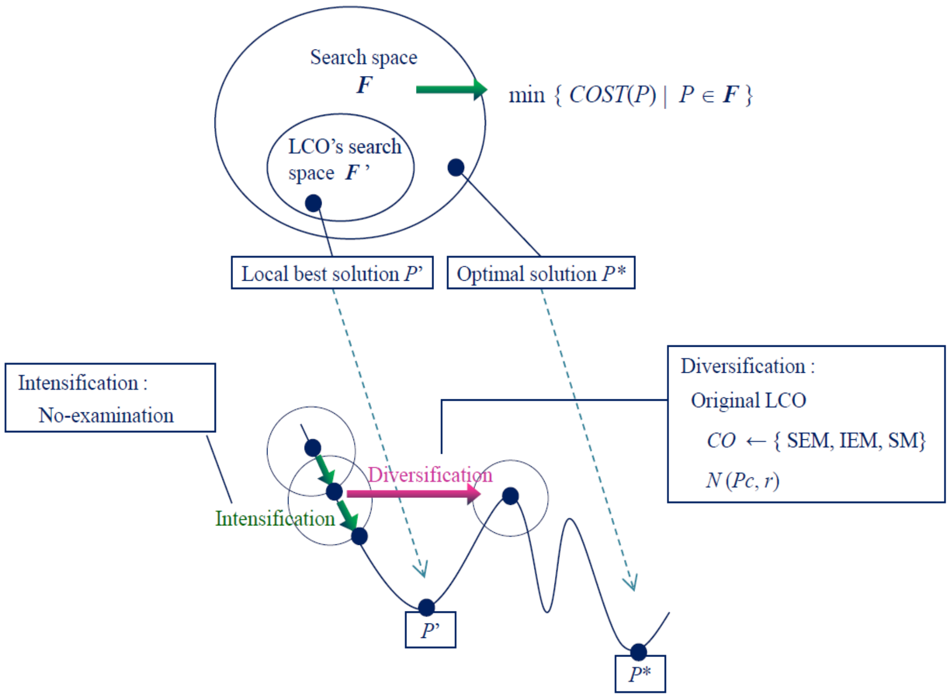


Figure 1. 2 Diversification and intensification of search.

LCO as Variable neighborhood search:

LCO is able to be considered one of VNS. VNS is a meta-heuristic proposed by is systematic change of neighborhood within a local search. Its strategy consists in the exploration of dynamically changing neighborhoods for a given solution. At the initialization step, a set of neighborhood structure *s* has to be defined. These neighborhoods can be arbitrarily chosen, but often a sequence { N_1, N_2,.... Nc_{max} } of neighborhoods with increasing cardinality is defined. In principle they could be included one in the other ($N_1 \in N_2 \ldots \in Nc_{nmax}$). However, such a sequence may produce an inefficient search, because a large number of solutions can be revisited.

VNS

Select a set of neighborhood structures *Nc*, c=1,2,…c_{max}

Choose, at random, an initial solution *s* in the search space

$c \leftarrow 1$

while $c < c_{max}$ do

    Select the best solution *s*' in the c^{th} neighborhood *Nc* (*s*) of *s*

    if *s*' is better than *s* then

        s $\leftarrow$ s'

        $c \leftarrow 1$

    else

        $c \leftarrow c+1$

    end

end

return the best solution met

Figure 1. 3 Algorithm of variable neighborhood search. [Mladenovic 97]

In the algorithm of VNS, an initial solution is generated first. Then VNS's main flow starts. This cycle consists of 3 steps, i.e. shaking, local search and move. In the shaking step, a solution *s*' is randomly selected in the *c*th neighborhood of the current solution *s*. Then, *s*' is used as the initial solution of a local search procedure, to generate the solution *s*''. The local search can use any neighborhood structure and is not restricted to the set $\{Nc, c=1, ..., c_{max}\}$. At the end of the local search process, if *s*'' is better than *s*, then s'' replaces *s* and the cycle starts again with $c = 1$. Otherwise, the algorithm moves to the next neighborhood $c + 1$ and a new shaking phase starts using this neighborhood. This algorithm is efficient if the neighborhoods used are complementary, i.e. if a local optimum for a neighborhood *Ni* is not a local optimum for a neighborhood *Nj*. VNS is based on the variable neighborhood descent (VND), which is a deterministic version of VNS.

1.3. Traveling Salesman Problem (TSP)

TSP is one of famous combinatorial optimization problem as NP-hard. It is a problem for finding Hamilton circuit with minimum total cost when it is given a complete undirected graph $G = (V, E, w)$ with the set of vertices *V*, the set of edges *E*, and set of costs of each edge *w*. Where *n* is the number of nodes, i.e. cities of TSP, it is hard to find a exact solution by becoming larger to the number *n* of vertex, because the total number of tour is as follows.

$$f_{comb}(n) = \frac{(n-1)!}{2} \tag{1.1}$$

Subject to n : number of city

There are symmetric TSP (STSP) and asymmetric TSP (ATSP) in TSP. ATSP is able to define in double STSP. This study targets STSP. By the following explanation, STSP is called TSP.

1.4. Self-Organization Map (SOM)

This section explains about SOM algorithm to solve TSP using Riccatti-type learning law. SOM was developed as a neural network. It minimizes the tour globally by self organization that neuron's topology network learns reference-vector signals of input city coordinates. As for the learning law, SOM is based on Riccatti equation, that Hebb's raw was given the modifying of “active learning” and “active forgetting”. The learning equation is given as follows:

$$\frac{d\mathbf{w}_j}{dt} = P\left(\mathbf{x} - Q\mathbf{w}_j\right) \tag{1.2}$$

where $\mathbf{x}$= {$\mathbf{x}_i$; i=1,2,…, n} is an input reference-vector given by cities coordinates, $\mathbf{w}$= {$\mathbf{w}_j$; j=1,2,…, m} is a neuron’s synapse-vector for learning input reference-vector, P is a plasticity-control function for activation of neurons, and Q is a forgetting-rate function for reversibility. An update of synapse-vector $\mathbf{w}$ in the learning process is as follows:

$$\mathbf{w}_j(t+1) = \mathbf{w}_j(t) + fc(t)\,(\mathbf{x} - Q\mathbf{w}_j(t)) \tag{1.3}$$

where $fc(t)$ (< 1) is a learning parameter to determine quantity of movement. Then, Nc is a subset of neighborhood neurons including a neuron c of $j \in Nc$. The neuron c of $\mathbf{w}_j$ which minimizes the Euclidean distance with the input vector $\mathbf{x}_i$ is selected as follows:

$$c = \arg\ \min_j \ \left\{|\ \mathbf{x}_i - \mathbf{w}_j\ |\right\} \tag{1.4}$$

The SOM algorithm is as follows:

(step1) Set the 2 dimensional city coordinate by reference-vector $\mathbf{x}$= {$\mathbf{x}_i$; i=1,2,…, n} as an input vector.

(step2) Set the 2 dimensional coordinate of neuron in synapse-vector $\mathbf{w}_j(t)$, where $m > n$. A set of neurons is arranged along a small circle, and all neurons are connected to make a circled topology. Set the iteration number t=0.

(step3) Set t=t+1. Choose one input reference-vector $\mathbf{x}_i$ randomly as a learning signal.

(step4) Select a neuron c that the vector $\mathbf{w}_c(t)$ is the closest to input reference-vector $\mathbf{x}_i$ chosen by (step3), based on Equation (1.4).

(step5) Update synapse-vectors of a set of neighborhood neurons Nc around neuron c to the reference-vector, so that it is made to come near the reference-vector, based on Equation (1.3).

(step6) If a set of input reference-vectors $\{\mathbf{x}_i\}$ is equal to a set of synapse-vectors $\{\mathbf{w}_c(\mathrm{t})\}$, i.e. each synapse-vector catches one input reference-vector, the procedure ends. Otherwise, return to (step 3).

As the iteration number t increases, the number of a set of neighborhood neuron set Nc decreases. A number of neighborhood neuron set Nc is as $r(t)$, then $r(t)$ is determined as follows:

$$r(t) = \max\left(\frac{m}{2} - \frac{t}{\beta},\quad 1\right) \tag{1.5}$$

where β is a reducing factor. The learning parameter $fc(t)$ to determine quantity of movement decreases as follows.

$$fc(t) = \frac{1}{\log(t+2)} \tag{1.6}$$

1.5. Local Clustering Organization (LCO)

In the behavior of learning based on Equation (1.3), SOM attracts the close neurons' synapse-vector signals, holds that neurons with similar synapses are gathered together in a local area. As a result of its self-organization, neurons are clustered into groups. SOM for TSP thus minimizes tour globally by repetition of the local attraction of vector signals, by the global potential-function is chosen. However, SOM has a lack that is not applicable when a city is not given by coordinate. LCO is given an operator clustering by costs, instead of directly calculating Equation (1.3).

Now *Cost*(c, i) is as the cost function between city c and i. The numbers of city is equal as numbers of neuron, i.e. n=m. Let consider *Cost*(c, i) is as the cost function between city c and i. The number of city is equal to numbers of neurons, i.e. n=m. Since the learning is advancing, a neuron is being attracted to the reference-vector, *Cost*(c, i) is the function of synapse-vector, so it can replace *Cost*($\mathbf{w}_c$, $\mathbf{w}_i$). Also the cost between c and k is *Cost*($\mathbf{w}_c$, $\mathbf{w}_k$). Now, the city sequenced is as c, i, k by self-organization. Then following is derived by clustering.

$$|\,\mathbf{x}_i - \mathbf{w}_c\,| - |\,\mathbf{x}_k - \mathbf{w}_c\,| < 0 \tag{1.7}$$

Let B($\mathbf{u}$, $\mathbf{v}$) continuous function of vector is introduced. B($\mathbf{u}$, $\mathbf{v}$) is considered as a complemented curve of a cost between 2 cities, that is $B(\mathbf{w}_i, \mathbf{w}_j)$=*Cost*($\mathbf{w}_i$, $\mathbf{w}_j$). Taylor expansion is applied to B($\mathbf{u}$, $\mathbf{v}$) about $\mathbf{v}$.

$$B(\mathbf{u},\mathbf{v} + \delta\mathbf{v}) \cong B(\mathbf{u},\mathbf{v}) + \left(\frac{\partial B(\mathbf{u},\mathbf{v})}{\delta\mathbf{v}}\right)^T \delta\mathbf{v} \tag{1.8}$$

$$B(\mathbf{w}_c\,,\mathbf{w}_i) \cong B(\mathbf{w}_c\,,\mathbf{w}_c) + g^T(\mathbf{w}_i\,,\mathbf{w}_c) \tag{1.9}$$

$$B(\mathbf{w}_c\,,\mathbf{w}_k) \cong B(\mathbf{w}_c\,,\mathbf{w}_c) + g^T(\mathbf{w}_k\,,\mathbf{w}_c) \tag{1.10}$$

where $g = \left(\frac{\partial B(\mathbf{w}_C,\mathbf{w}_C)}{\delta \mathbf{v}} \right)$. The absolute value of Equation (1.9) and (1.10) is removed, then triangle inequality and complemented equation applied.

$$Cost(\mathbf{w}_c, \mathbf{w}_i) < |\ Cost(\mathbf{w}_c, \mathbf{w}_c)\ | + |\ g\ |\ |\ (\mathbf{w}_i - \mathbf{w}_c)\ | \tag{1.11}$$

$$Cost(\mathbf{w}_c, \mathbf{w}_k) < |\ Cost(\mathbf{w}_c, \mathbf{w}_c)\ | + |\ g\ |\ |\ (\mathbf{w}_k - \mathbf{w}_c)\ | \tag{1.12}$$

The following is derived when to deduct both members of an inequality.

$$Cost(\mathbf{w}_c, \mathbf{w}_i) - Cost(\mathbf{w}_c, \mathbf{w}_k) < |\ g\ |\ (|\ \mathbf{w}_i - \mathbf{w}_c\ | - |\ \mathbf{w}_k - \mathbf{w}_c\ |) \tag{1.13}$$

The following relation is derived based on Equation (1.7).

$$Cost(\mathbf{w}_c, \mathbf{w}_i) - Cost(\mathbf{w}_c, \mathbf{w}_k) < 0 \tag{1.14}$$

$$Cost(\mathbf{w}_c, \mathbf{w}_i) < Cost(\mathbf{w}_c, \mathbf{w}_k) \tag{1.15}$$

That is, after self-organization in SOM, the clustering based on cost sorting is caused between 2 cities, to produce the same effect as SOM. The synapse-vector $\mathbf{w}_c$, $\mathbf{w}_i$ and $\mathbf{w}_k$ correspond to city's coordinates. The learning approximately realizes the following relations in the local area after self-organization.

$$|\mathbf{x}_c - \mathbf{w}_i| < |\mathbf{x}_c - \mathbf{w}_k| \Leftrightarrow Cost\ (P_c, P_i) < Cost(P_c, P_k) \tag{1.16}$$

Thus the new algorithm of LCO is as follows. A set of neighborhood is selected by the learning low for clustering to exchange the city sequence.

(step1) Neurons are considered to be cities. Give the cost between cities. As an initial tour, a round-trip for a set of city nodes P= {Pi; i=1,2,…, n} is generated randomly. Set t=0 as an iteration number.

(step2) Set t=t+1. Choose one city Pc randomly. Determine a set of neighborhood $N(Pc, r)$. The r is the number of nodes in the neighborhood adjusted as t increases.

(step3) Exchange the node sequence in a neighborhood to satisfy the fitness of the tour cost. This operation is defined "local clustering".

(step4) If the cost evaluation of total tour includes improvement, accept the new sequence, and update total tour.

(step5) If t=T (T: termination), the procedure is end. Otherwise, repeat from (step2) to (step5).

```
LCO
----------------------------------------------------------------------------
Search point  P0 ∈ F
P0 = { p1, p2, …, pn}        : Encoding and Choose an initial solution in the search space
E P0 = COST (P0)             : Evaluation function
t ←0
Pt ←P0
Et ←E P0
while  the stopping criterion is not satisfied   do
    Nt | Nt ⊂ Pt                    : Select a neighborhood
    COt ← { SEM, IEM, SM}           : Select a local clustering operation
    while Nt criterion is not satisfied do
        Pt' ←COt (Pt)       : Local clustering operation
        E_Pt' = COST(Pt')   : Evaluation function
        if  Et'< Et  then                        R (COt)  : Iteration of local
            Et ←Et'                                         clustering operation
            Pt ←Pt'                              Move strategy:   first improvement
        end
    end
    t ← t+1
end
return   the best solution
```

Figure 1. 4 Algorithm of LCO.

In the LCO algorithm, the solution structure that the elements express the order of solution is encoded first. There are 2 parameters, i.e. the determination of the neighborhood for the LCO's search operation and the selection of the operation's variation stochastically. This operation is a local clustering method. Then the selected operation is repeated in the neighborhood. According to a move strategy for a solution, a new solution is accepted. LCO uses a first improvement as the strategy.

1.6. **Local Clustering Algorithm**

In the LCO algorithm, nodes are sequenced by repetition of local clustering operation referencing costs. This section explains a method of operation of the exchange of the node sequence concretely. Now, a set of nodes with city number for a round-trip i.e. $P = \{Pi\ ;\ i=1,2,\ldots,n\}$, where n is the number of nodes. Then a neighborhood is as $N(Pc, r)$, that the center node is Pc and the radius is r.

Simple exchange *m*ethod (SEM):

SEM replaces a center of node with the right or left node in a clustering neighborhood. And repeat exchanging the position of 2 nodes. When the evaluation is improved, the sequence is accepted.

(step1) One of nodes Pc is chosen from a set P of nodes randomly, and a neighborhood $N(Pc, r)$ is decided.

(step2) Set an index of left side in a neighborhood $N(Pc, r)$ to L, set a right side of a neighborhood $N(Pc, r)$ to R, thus set an index L=c-r, and R=c+r. Set $CL=CR=c$.

(step3) Set $dL=CL-1$, and $dR=CR+1$.

(step4) Exchange P_{CL} for P_{dL}.

Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq$

$F(P_{next})$, accept the new sequence.

(step5) Exchange P_{CR} for P_{dR}.

Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step6) Set dL=dL-1, and set dR=dR+1. Then repeat from (step4) to (step6) until dL=L and dR=R. When CL=L and CR=R, the procedure is end.

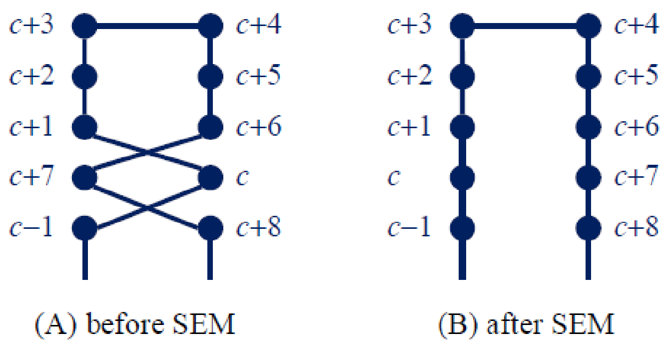


Figure 1. 5 Effect of SEM (applied toPc and Pc_{+7}).

Inverse exchange method (IEM):

IEM repeats inverse-sorting of the tour between pick-upped 2 nodes in a clustering neighborhood. If current sequence is as {c, c+1, c+2, c+3, }, the inverted sequence is {c+3, c+2, c+1, c}. When the evaluation is improved, the sequence is accepted.

(step1) One of nodes Pc is chosen from a set P of nodes randomly, and a neighborhood $N(Pc, r)$ is decided.

(step2) Set an index of left side in a neighborhood $N(Pc, r)$ to L, set a right side of a neighborhood $N(Pc, r)$ to R, thus set an index L=c-r, and R=c+r.

(step3) Invert between P_C and P_{dL}.

Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step4) Invert between P_C for P_{dR}.

Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step5) Set $dL=dL$-1, and set $dR=dR$+1. Then repeat from (step3) to (step5) until $dL=L$ and $dR=R$.

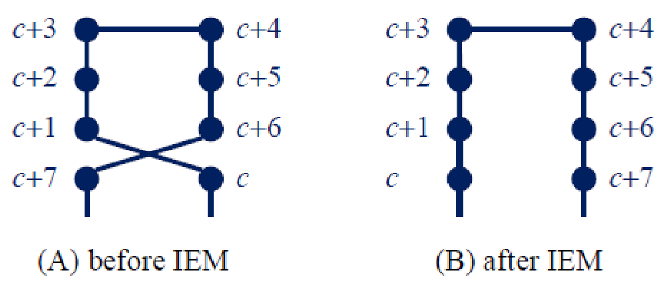


Figure 1. 6 Effect of IEM (applied toPc and Pc_{+7}).

Smoothing method (SM):

In the local area, SM repeats inverse-sorting from the left side to right side and repeat inverse the position of 2 nodes. The starting point is shifted to right sequentially in the clustering neighborhood, so it is calculated in the total hits. When the cost is improved, the tour sequence is accepted. When the evaluation is improved, the sequence is accepted.

(step1) One of nodes Pc is chosen from a set P of nodes randomly, and a neighborhood $N(Pc, r)$ is decided.

(step2) Set an index of left side in a neighborhood $N(Pc, r)$ to L, set a right side of a neighborhood $N(Pc, r)$ to R, thus set an index L=c-r, and R=c+r. Set CL=L.

(step3) Set $dL=CL+1$.

(step4) Invert between P_{CL} and P_{dL}.

Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step5) Set $dL=dL+1$. Then repeat (step4) until $dL= R$.

(step6) Set $CL=CL+1$. Repeat from (step3) to (step6) until $CL= R-1$. When $CL= R$, the procedure is end.

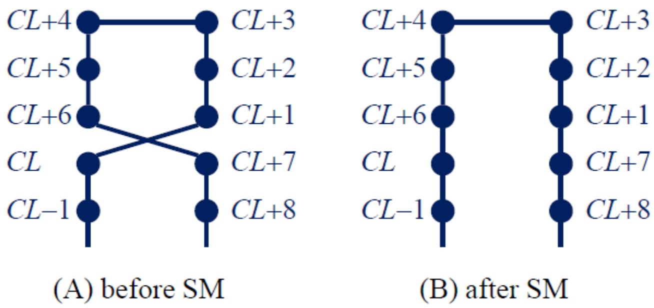


Figure 1. 7 Effect of SM (applied toP_{CL} and P_{CL+7}).

Mixed clustering:

LCO operates these 3 kinds of method randomly. In original LCO for TSP, the probability of select is SEM:IEM:SM=2:2:1 on the experiments. In one clustering of each method, the numbers of scheduling order as follows.

f_{cls} (SEM) =2r

f_{cls} (IEM) = 2r

f_{cls} (SM) = 2r(2r-1)

1.7. Issue

LCO is one of VNS included in meta-heuristics, and has 2 kinds of parameter. The principle of the LCO algorithm was developed based on SOM. Here are the main points of the original LCO algorithm are shaped.

(A) LCO is ILS exploring first improvement of the solutions, with the random positioning for the clustering neighborhood.
(B) The encoding of LCO is a set of permutation arranging elements given a node number, and is considered ring form.
(C) The evaluation of improvement is given by referencing costs given between each 2 cities.
(D) LCO operates the city sequence by exchanging of city position in a subset selected from a tour as a neighborhood, related to a clustering of neurons.
(E) The set of operation neighborhood is related to numbers of city and decreases by increase in iteration number, related to the learning law of SOM.

The advantage to study LCO in Heuristics is as follows.

Advantage to study LCO in Approximate algorithm:
(A) LCO succeeded to the characteristic of global optimization that neuro -dynamics of SOM brought.
(B) As for the search of a local neighborhood based on Riccatti-type learning law, SOM had the function of an intensive search.
(C) Furthermore, the expansion to LCO became able to apply to larger-scale problems than SOM, by succession of high-speed performance of SOM.

Then this section gives what is issue. Table1.1 is an comparison of LCO and GA applied “Memetic algorithm” (MA or Genetic local search) in examples of small-scale TSP. This MA is used LCO for an improvement of a sequence except the crossover part after crossover. In the results, MA is better than LCO, thus it assumes that LCO operator is possible to improve. Then it is considered that a set of the set of nodes of crossover part is fixed, and LCO searches the other nodes. Meta-heuristics needs such an effective variation of operation and an intensification of search. The details of the issues are as follows.

Table 1. 1 Comparison of LCO with GA.

Methods	LCO		GA+LCO (Memetic algorithm)	
	Accuracy [%]	Time [sec]	Accuracy [%]	Time [sec]
ulysses16	0.005	0.010	0.000	18.594
ch130	0.096	0.009	0.004	14.291
double circle (city no.=100)	0.160	0.027	0.000	1.992

The values are percentage above the best/ lower bound. Times were measured on a 2300MHz AMD Opteron Processor-6134 PC. (trial=10, GA's Crossover:GSX2)

What LCO operates on problem:

(A) The variation of the operation is limited to 2-opt type and an exchange of 2 nodes. These doesn't examined to enough as a search operation.

(B) The operation is limited to the applications to the range of a neighborhood shifted from SOM.

(C) The policy of the selection rate of each operation is indistinct.

What neighborhood setting brings:

(A) It is not clear about the necessity of adjustment for a random positioning of operation neighborhood.

(B) It is not clear about the relation of the parameter adjustment in a range of neighborhood and the search process, though depend on numbers of city and an iteration.

(C) It is not investigated whether improvement may be limited by a convergence of neighborhood.

(D) Relations with expansion and contraction of a neighborhood by addition and deletion of neurons in SOM algorithm are indistinct.

About general faculty and strategy for heuristics:

(A) LCO is first improvement, and there is not a function of best improvement around a search neighborhood.

(B) There is not a function of a stack of exploration process.

(C) There is not functions of restart and circular search of a starting point.

(D) There is not an exploitation of structural parts.

(E) A policy of the intensification of search is indistinct. The search is not given priority to width and is given priority to depth and seems to search. However, it's not considered to move ahead by excluding a range without possibility of improvement either, too.

1.8. Conclusion

This chapter introduced LCO based on SOM principle for TSP. In addition, the themes for extension of LCO were stated.

(A) LCO is one of single-solution based meta-heuristics, included in VNS (variable neighborhood search). The design of meta-heuristics is required a diversification and an intensification of search.

(B) In the SOM (Self-organization map) algorithm, synapse-vector learns reference-vector (city coordinates). The clustering by this learning causes a sorting by cost. This mechanism applied to LCO (Local clustering organization).

(C) LCO refers the cost between 2 cities, so LCO is able to apply to scheduling problems without city coordinates.

(D) LCO algorithm is operated by local clustering with a clustering neighborhood. The clustering neighborhood is determined by Riccatti-type learning law.

(E) LCO has 3 kinds of local clustering operation. SEM (Simple exchange method) exchanges 2 cities, and IEM (Inverse exchange method) exchanges between 2 cities inverse. SM (Smoothing method) operate m-step 2-opt that calculated an exchanging 2 nodes in the total hits within a neighborhood.

(F) The LCO algorithm has 2 kinds of parameter, i.e. the neighborhood and the clustering methods. The issues of LCO from the standpoint on categories of operation, the neighborhood and the general faculty for meta-heuristics were examined. Then, the diversification and intensification of LCO don’t discuss yet.

Chapter 2

2. Design of Local Clustering

2.1. Introduction

This chapter gives a theory and method for design of operation of LCO for application to combinatorial optimization problems. For a design of meta-heuristic, “diversification” and “intensification” of search are the general policies to be considered. The diversification is an extension of operation variation. The intensification is a search limited to an effective area for improvement in large search spaces. First, we define the framework of patterns of local clustering of LCO operator based on SOM's learning law that is original LCO.

The positioning of original LCO's operator is analyzed using this framework. Then this framework is used for the design of extended operation. It brings to the LCO operator the combination with the variation of operation. As the second, it is necessary to bring an intensive exploration with reduction of the search space by corresponding structure of problems. It is defined as structured local clustering. The goal of this chapter produces the frame of total local clustering design through the principle of LCO operation, and analysis of the original operation, then explains structured local clustering.

This chapter is organized as follows. Section 2.2 explains general combination optimization problems and here studied problem domain. Section 2.3 raise a subject and expands the approach of this study. In Section 2.4, the local clustering is leaded based on the principle. Section 2.5 is the validation of LCO operator by comparison with λ-opt method. Section 2.6 explains the methodology of architecture design necessary to apply structured local clustering to domain. Section 2.7 shows positioning and the purpose of this doctoral dissertation. Finally, the conclusion is stated in Section 2.8.

2.2. Combinatorial Optimization Problems

Combinatorial optimization is a mathematical programming problem that a variable should be decided by combination, i.e. a solution or subset of universal set. When it selects a combination that satisfies the requisite condition, it is a problem that the best solution is selected from the combinations. In other words, when one subset *P* in one set *F* satisfies the condition, it demand *P* minimizing function *COST*(*P*). Combinatorial optimization is a modeling of real social problems, and to solve for finding the best solution. Optimization problems appear in many situations in our daily life. Such problems appear whenever a process has to be arranged in such a way that resources, money or time are saved. [Hartmann 05, Pardalos 10]

This dissertation discuss combinatorial optimization problems, those are in practice much harder to solve, e.g. TSP and JSP.

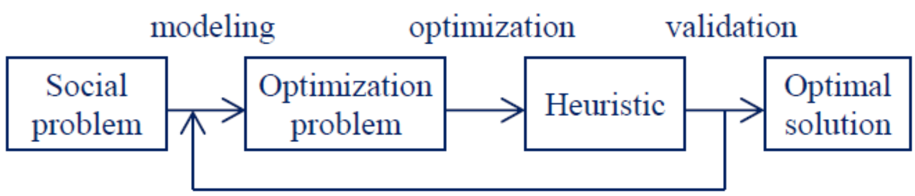


Figure 2. 1 Combinatorial optimization approach.

.

Here, it considers the definition of combinatorial optimization is given as follows.

When a set ***F*** which is not empty, a mapping *COST*: $F \rightarrow R(Z)$ from *F* to a real number are given, it demand *x* to give the following.

$$\min \{ COST(P) \mid P \in F \} \tag{2.1}$$

where *Z* is a set of integers, *R* is a set of real numbers. *F* is a set of feasible solutions, an element $P \in F$ is a feasible solution.

A solution *P* minimizing an objective function *COST*(*P*) is globally optimal solution, and the set of *F* is as *P**.

A set *P** of globally optimal solution is defined as follows.

$$F^* = \{ P \in F \mid COST(P) \leq COST(y),\ \forall y \in F \} \tag{2.2}$$

This dissertation discusses about TSP and JSP. TSP is one of most important problem that social application widely, base on various situation. JSP is examined as with the limitation condition next. Because benchmark problem is provided in TSP and JSP, general performances are able to be inspected.

2.3. **Subject and Approach**

LCO developed based on SOM has a similar potential prime mover, and it is possible to apply to larger-scale problems than SOM by succession of high-speed performance of SOM. However, the operation is not analyzed whether variation of operation is enough. This section explains the analysis of LCO operation and the method of design operator. The LCO operator has 3 kinds of clustering method as original based on SOM's operation. First, the patterns of operation is analysis, i.e. it is defined the local clustering. Based on this definition, the diversification and intensification is discussed. About the diversification, the positioning of original local clustering is validated. In addition, those are comparison of λ-opt method for validation. From the analysis, the extended local clustering is considered.

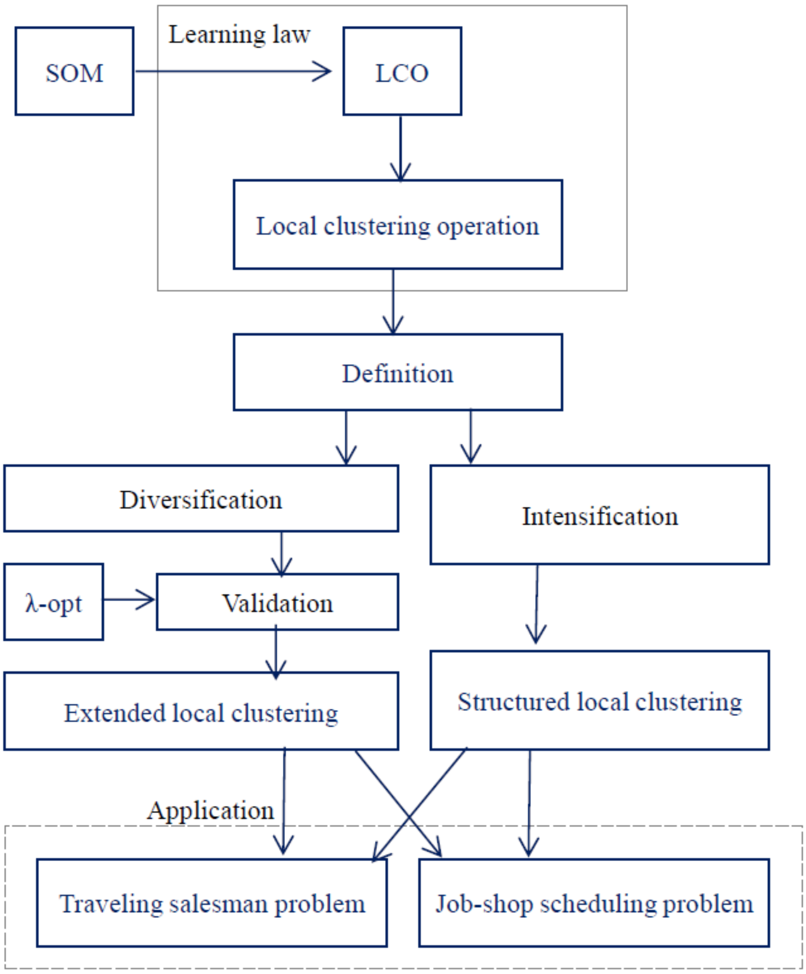


Figure 2. 2 Outline for design of structured local clustering.

In the search space that is brought by the extension of operation, it is considered to limit the search space to more improved area along a problem structure. Now, this study gives the definition of structured local clustering and the design method. Then in each chapter following this chapter, by using a numerical experiment of TSP and JSP, it is analyzed the mechanism related to the improvement to solve, and is examined about the design of structured local clustering.

What is included in structure:

The structure to be shown here is the characteristic that each framework of various combination optimization problems has. It is different in each problem, and an examination is necessary. For examples, those are an encoding, a weight, a measure or elements for combine, an evaluation function, etc. The more one is the characteristic that analysis of a mechanism to be related to an improvement in a search process reveals. Furthermore, it is excluded realizing the individual characteristic of the problem, but the algorithm will give a response for the dynamic change in the search process.

2.4. Define of Local Clustering

The local clustering of LCO was developed based on SOM's learning for solving TSP. In the SOM algorithm, the city's order is led by a neuron synapse-vector learns a city coordinate. The exchange of the city sequence in SOM correspond the exchange by referencing the cost of between 2 cities. That relation is applied to LCO operation. This algorithm lead LCO from SOM’s learning law is explained in chapter 1. As for an applied operation, a local clustering effect the triangle inequality of TSP in a neighborhood. When new local clustering is designed, it examines learning equation (1.16) of relations of LCO and SOM. It is necessary for the clustering method of LCO to let this satisfy. In the original LCO, SEM node exchanging operation was developed. Then IEM and SM were applied, LCO improved performance for large-scale TSP. These exchanging operation of λ-opt types are possible to extend for parts or node-set based operation, not only node-bases.

$$\mathbf{w}i\,(t+1) = \mathbf{w}i\,(t) + hc\,(t)\,(\,\mathbf{x}c\,(t) - \mathbf{w}i\,(t)\,) \tag{2.3}$$

subject to
- $\mathbf{w}i$: vector of neuron i
- $\mathbf{x}c$: vector of city c
- $hc(t)$: learning factor
- t : iteration

$$|\mathbf{x}_c - \mathbf{w}_i| < |\mathbf{x}_c - \mathbf{w}_k| \Leftrightarrow Cost\,(P_c, P_i) < Cost(P_c, P_k) \tag{2.4}$$

Here, a city is as node, when improvement is caused by the operation between 2 nodes, it examines what kind of pattern is considered. In addition the pattern of operation is also examined in case of the node-set, that is a block combination.

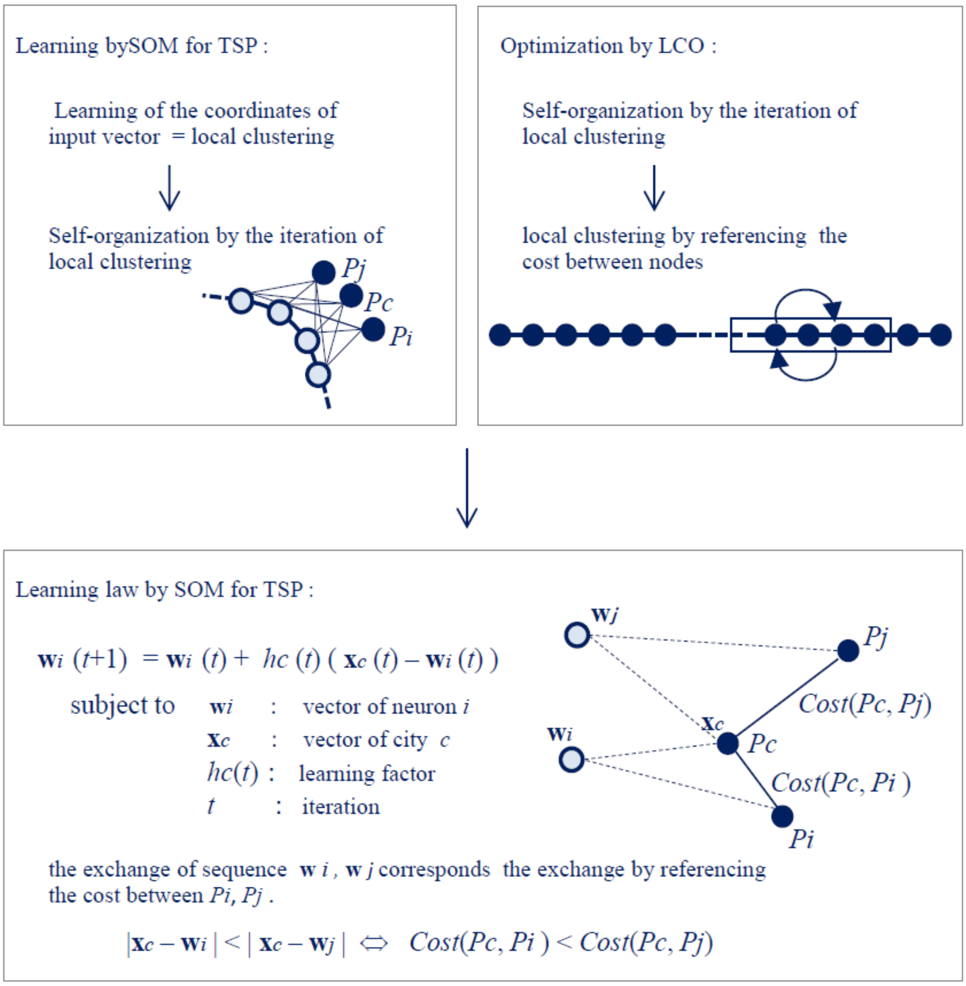


Figure 2. 3 Relation of local clustering of LCO and SOM.

Definition Node based local clustering:

Now node-based operation is discussed. Let P is one of search point:

$$P = (P1, P2, \ldots P\alpha, \ldots.Pc, \ldots,Pj, \ldots, Pi, \ldots, P\beta \ldots Pn) \tag{2.5}$$

where n is a number of nodes that is solution elements. When the subset of P that is a neighborhood between α and β is

$$N\alpha,\beta = (P\alpha, \ldots, Pc, \ldots, Pj, \ldots, Pi, \ldots, P\beta) \tag{2.6}$$

Then the evaluation function is

$$\begin{aligned} COST(P) &= COST(P\alpha, \ldots. Pc, \ldots,Pj, \ldots, Pi, \ldots, P\beta) \\ &= COST(N\alpha,\beta, Pc, Pj, Pi) \end{aligned} \tag{2.7}$$

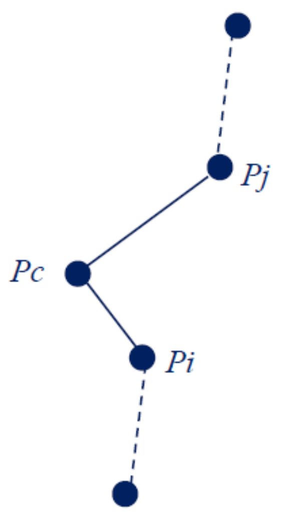


Figure 2. 4 Connection of nodes.

When the costs is compared by exchanging *i* and *j*, and a new search point *P'* to produce by exchange is

$$COST\ (N\alpha,\beta\ , Pc, Pi, Pj) < COST\ (N\alpha,\beta\ , Pc, Pj, Pi\ )$$
$$\text{subject to} \quad \alpha \le c, i, \quad j \le \beta$$
$$P' = \quad (P1, P2, \ldots.Pc, \ldots,Pi, \ldots, Pj, \ldots Pn\ ) \tag{2.8}$$

Then the clustering operation *CO* is defined by a neighborhood $N\alpha,\beta$ and the operated nodes.

$$CO\ (N\alpha,\beta\ , Pc, Pi, Pj\ )$$
$$Bool \leftarrow COST\ (N\alpha,\beta\ , Pc, Pi, Pj\ ) < COST\ (N\alpha,\beta\ , Pc, Pj, Pi\ ) \tag{2.9}$$

Definition node-set based local clustering:

The local clustering operation is possible to apply to the case of node-set. It suggests a pattern of the operation including node-sets, and shows that the design variation of the operation is extended. In the case of node-set operation, it is necessary that the direction that a node-set is connected and the connection method between 2 node-sets are considered. In the direction, to forward and to reverse a node-set is considered. In addition, as the connection method between 2 node-sets are considered, i.e. terminal that is connected the side of 2 node-sets, minimum and heuristic.

Let *P* is one of search point:

$$P = \quad (P1, P2, \ldots\ P\alpha \quad , \ldots.Pc, \ldots,Pj, \ldots, Pi, \ldots, P\beta \ldots Pn\ ) \tag{2.10}$$

where *n* is a number of nodes. Here it considers that *P* is comprised of subset *B* (Block) of *div* units.

$$P = \quad = \quad (B1, B2, \ldots\ B\alpha\ , \ldots Bc, \ldots,Bj, \ldots, Bi, \ldots\ B\beta \ldots\ Bdiv\ ) \tag{2.11}$$

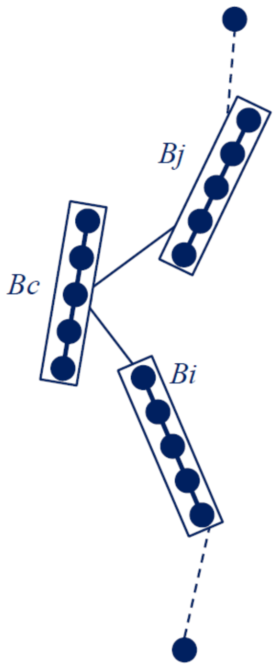


Figure 2. 5 Connection of node-set.

When the subset of P that is a neighborhood between α and β is

$$N\alpha b,\beta b \quad = \quad (B\alpha \;,\, \ldots\, Bc,\, \ldots,\, Bj,\, \ldots,\, Bi\,,\, \ldots,\, B\beta\,) \tag{2.12}$$

When the costs is compared by exchanging i and j, and a new search point P' to produce by exchange is

$$\begin{aligned}
&COST\,(N\alpha b,\beta b,\, Pc,\, Pi,\, Pj) < COST\,(N\alpha b,\beta b\,,\, Pc,\, Pj,\, Pi\,)\\
&\quad subject\ to \qquad \alpha \le c,\, i, \quad j \le \beta\\
&P' = \quad (B1,\, B2,\, \ldots.Bc,\, \ldots,Bi\,,\, \ldots,\, Bj\,,\, \ldots Bdiv\,)
\end{aligned} \tag{2.13}$$

Then the clustering operation *CO* is

$$
\begin{aligned}
&CO\,(N\alpha b,\beta b,\ \ Bc\,,\ \ Bi\pm,\ \ Bj\pm\ \ ,L\,)\\
&Bool \leftarrow COST\,(N\alpha b,\beta b,\ Bc,\ \ Bi\pm,\ \ Bj\pm\ \ ,L\,)\\
&\quad <\ \ COST\,(N\alpha b,\beta b\,,\ Bc,\ \ Bj\pm,\ \ Bi\pm\ \ ,L\,)
\end{aligned}
\tag{2.14}
$$

subject to $\pm$: Direction of node-set (+: forward , –: reverse)

L : Connection method

(1: terminal, 2: minimum, 3: heuristic)

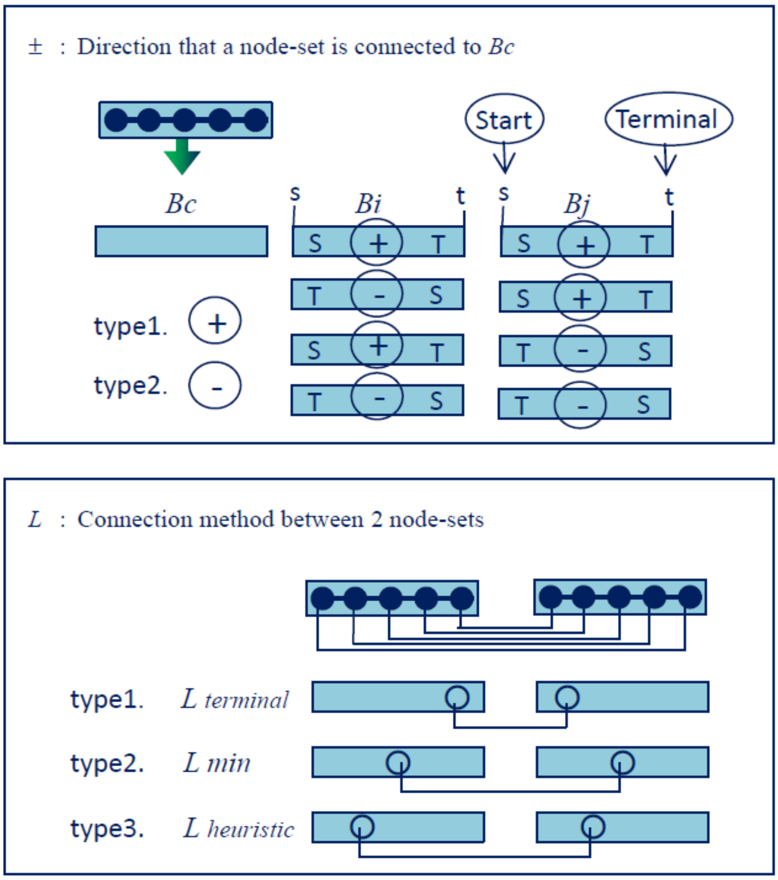


Figure 2. 6 Parameters of node-set operation, direction node-set and connection method.

Variation of local clustering operation by node and node-set base:

The definition of LCO's local clustering operation *CO* is presented by a neighborhood and nodes according to Equation (2.9) and (2.14). The variation of local clustering is expressed on the matrix table. The list is shared by the case of *c* is a node and the case of *c* is a node set.

Node based: $CO\ (N\alpha,\beta\ , Pc, Pi, Pj\ )$

Node-set based: $CO\ (N\alpha b,\beta b,\ \ Bc\ ,\ \ Bi\pm,\ \ Bj\pm\ \ , L\ )$

subject to $\pm$: direction of node-set

L : connection method of node-set

Case of c is a node : $CO\ (N, P_c,\ A_i,\ A_j\ , L\ )$

		A_j		
	Pc	Pj	Bj +	Bj –
A_i	Pi			
	Bi +			
	Bi –			

Case of c is a node-set : $CO\ (N, B_c,\ A_i,\ A_j\ , L\ )$

		A_j		
	Bc	Pj	Bj +	Bj –
A_i	Pi			
	Bi +			
	Bi –			

Figure 2. 7 Variation of local clustering operation**.**

Analysis of positioning of local clustering:

The original LCO has 3 kinds of local clustering, SEM, IEM and SM. The positioning of local clustering on the matrix is as follows. In analyzing method of the local clustering using this matrix, it is possible to analyze the behavior by visualizing the recombination of nodes.

(step 1) A behavior of the operation of exchange and recombination of nodes or node-sets is shown using a chart of nodes sequence.

(step 2) Nodes that the linked edges are cut before and after operation are given order.

(step 3) The nodes' order is compared before and after the operation, and the nodes are given c, i, j.

(step 4) The operation is positioned on the matrix of local clustering.

CO (*N*, P_c, A_i, A_j, *L*)

		A_j		
	Pc	*Pj*	*Bj* +	*Bj* –
A_i	*Pi*	SEM		
	Bi +			
	Bi –	IEM,SM		

CO (*N*, B_c, A_i, A_j, *L*)

		A_j		
	Bj	*Pj*	*Bj* +	*Bj* –
A_i	*Pi*			
	Bi +			
	Bi –			

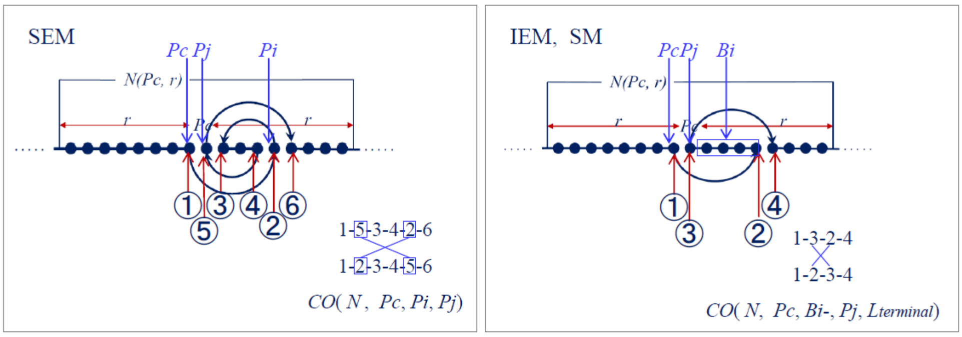


Figure 2. 8 Positioning of original local clustering.

2.5. **Validation**

This validation uses that the operation of original LCO is correspondence to λ-opt. The behavior on one search point by one operation of the local clustering is possible to compare with the operation of λ-opt. The operation of SEM and SM brings an exchange of 2 nodes, i.e. it is cut 4 edges of both side of nodes, and it is called the "swap" operation, included 4-opt based operation. Likewise, the inverse operation of IEM is related with 2-opt. It is confirmed that 3-opt lacks here. Generally, 2-opt and 3-opt are used together to make the performance for meta-heuristics hybrid method [Nguyen 07]. Thus, LCO comprised of SEM, IEM and SM is easy to catch onto the local solution. Therefore, in this dissertation, it introduces IM (Insert method) to original LCO operation. IM covers 3-opt.

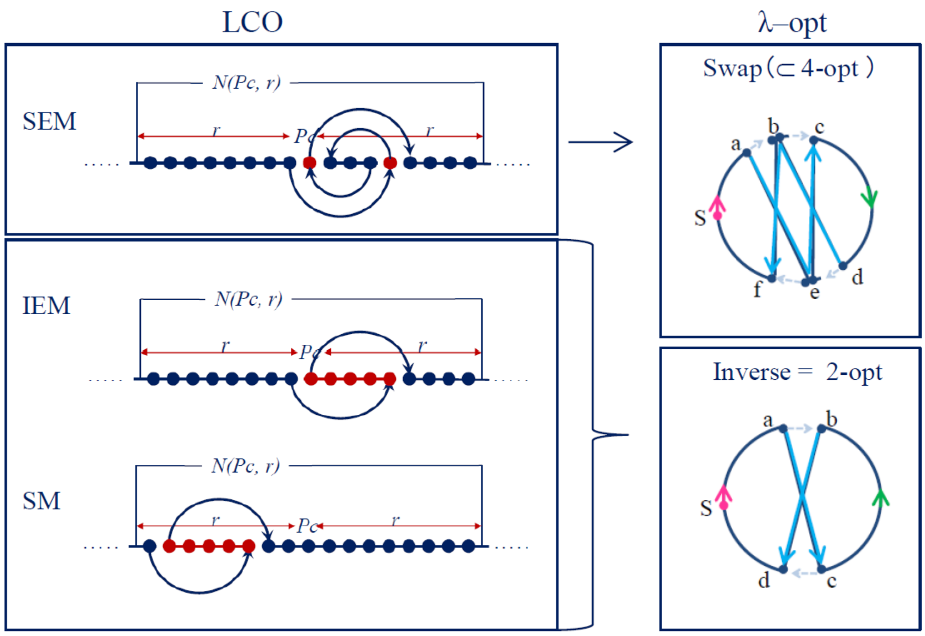


Figure 2. 9 Validation of original LCO by comparison with λ-opt.

Extended local clustering:

What kinds of operation lacks, it is possible to judge from the positioning of the matrix of the local clustering. In addition, this matrix gives a design strategy of local clustering which consists of a combination of nodes and node-sets. The original operation has 2-opt type and swap type. To improve the performance of original LCO, at first priority, it is necessary to apply the operation of the 3-opt type to LCO. Also, when LCO is used as hybrid method with other meta-heuristics, the design technique of LCO operator is necessary. The IM operation as the 3-opt type brings moving a subset optimized locally. An application and the procedure of the IM for TSP and JSP are explained in chapter 3 and chapter 4.

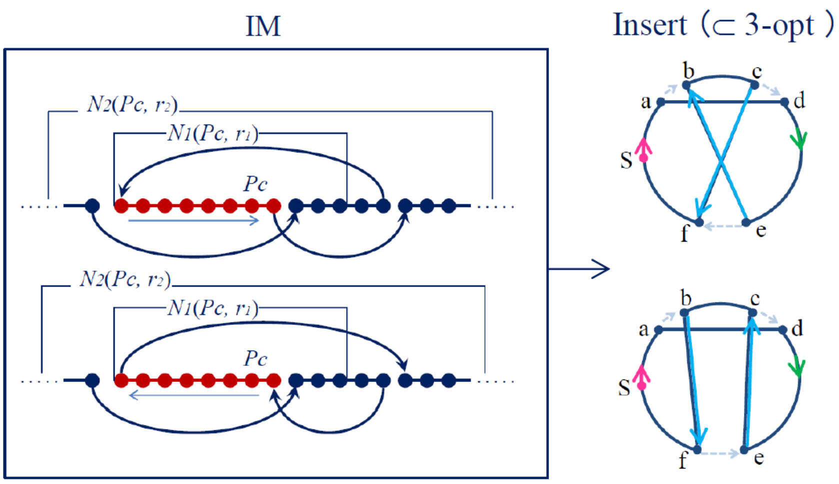
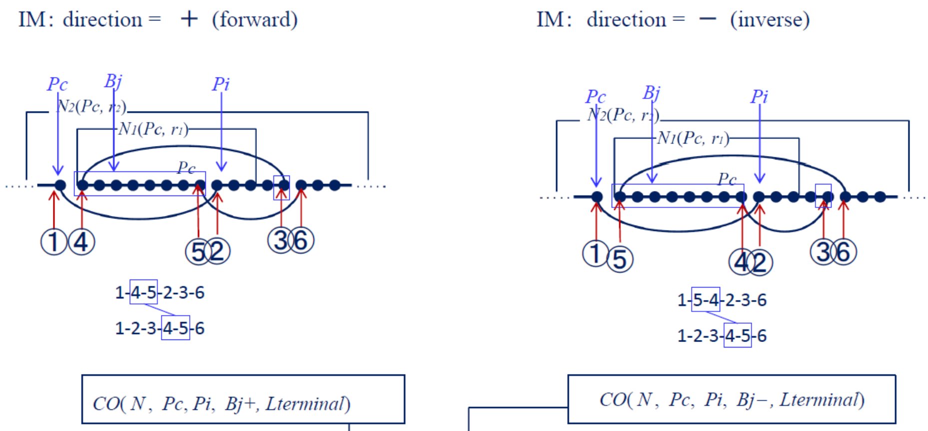


Figure 2. 10 Introduction of extended local clustering.



		Aj		
	Pc	*Pj*	*Bj* +	*Bj* −
Ai	*Pi*	SEM (Swap-opt)	IM (Insert⊂3-opt)	IM (Insert⊂3-opt)
	Bi +			
	Bi −	IEM,SM (2-opt)		

		Aj		
	Bc	*Pj*	*Bj* +	*Bj* −
Ai	*Pi*			
	Bi +			
	Bi −			

Figure 2. 11 Positioning of extended local clustering.

2.6. Design of Structured Local Clustering

The structure in this study means to get structurally the variation of the local clustering, and to get structurally the problem structure. Then the solution structure is considered abstractly to get the problem structure by parting into node-sets and nodes. The majority of combinatorial optimization problem have a structure. This section examines “structured local clustering” used the structure that a solution is constructed by combining parts. (This idea is similar parts optimization method, or the building block of GA.) A solution of target problem is supposes that is possible to divide into some parts.

A “part” of solution x ($\in F$) is elements of the solutions. (It is not empty subset)

A partition of solution x *is* defined satisfy as subset $\{p_1, p_2, \ldots, p_m\}$, $p_1 \cup p_2 \cup \ldots \cup p_m = x$, and $p_1 \cap p_2 = 0$ $(i \neq j)$. A component $p_1, p_2, \ldots, p_m$ of division is a candidate of a part. Furthermore, a problem is simplified by fixing parts which is likely to be included in the best solution one after another. The requirement of a part premises as follows.

Requirement of part:

(A) An objective function is determined for a set of some parts.

(B) A small optimization problem is defined.

(C) For an optimization problem, heuristic exists.

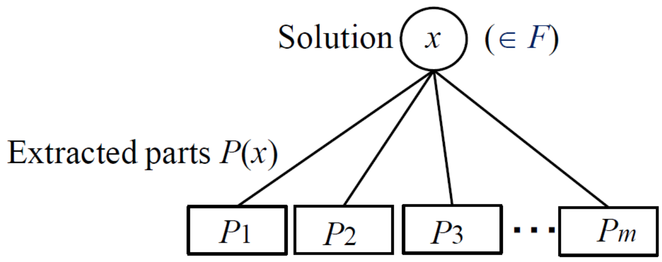


Figure 2. 12 Extract Parts.

Framework that parts are applied to LCO:

(step1) A structural requirement for search is determined.

(step2) A method to divide x is determined.

(step3) An operation method of the structured local clustering is determined.

(step4) A method to improve parts $P(x)$, and a judging method of validity of divided set $P(x)$ is determined.

(step5) A set $P(x)$ is that x is divided into m unit.

(step6) A neighborhood is selected in divided set $P(x)$.

(step7) The neighborhood is applied to the local clustering operation, and a solution x is improved.

(step9) Parts are judged, and are improved.

(step10) While a termination is not satisfied, it is repeated from (step 6) to (step9).

One solution x is divided to a set of part $P(x)$ according to reference one structure. In the parts, one part that operates as a set of nodes is treated “node-set”. Then, one parts block that operates as a set of nodes connect between 2 node-sets is “bridge”.

Structured local clustering:

A partial-set optimized locally is considered as a node-set. As for the bridge, it is considered both the length and the sequence are still fluctuating. We can design these node-sets and bridges for our strategy of diversification and intensification of search. The original LCO has 2 types of operator, i.e. 2-opt type and swap type. The extended operator can be applied to structured local clustering by using node-sets, bridges and the matrix of local clustering operation. These operator designed systematically defined as structured local clustering. A designer needs to examine how to decide a node-set by the structure of the problem first.

For example, TSP is a problem to find a sequence minimizing cost while a round trip of each spot. In TSP, sub-tours optimized locally in a search process appear dynamically. Those are able to consider as node-sets. As a different viewpoint of an intensification, the original LCO repeats swap and “first improvement” of move strategy without a method to make confirmation whether searched or not in its search neighborhood. When it searches a neighborhood of one search point until the evaluation is improved with acceptance of bad solution, the method of move strategy is best improvement. Such intensification of search is given a search of depth priority. To avoid a loop of the search without possibility of an improvement, a fluctuation and a stack are studied, like for example, the methods of “Simulated annealing” (SA) and “Tabu-search” (TA).

This study approaches to give an indicator of the intensification for local search. The strategy in this study gives the subsets of candidate of a best solution optimized locally on a node based permutation of same solution, as a node-set. The solution structure that fixed a subset fills the valley of some local minima, and concentrates a search on a bridge linked node-sets. Or, even if a node-set is used forward or reversed, it compresses a node-based combination.

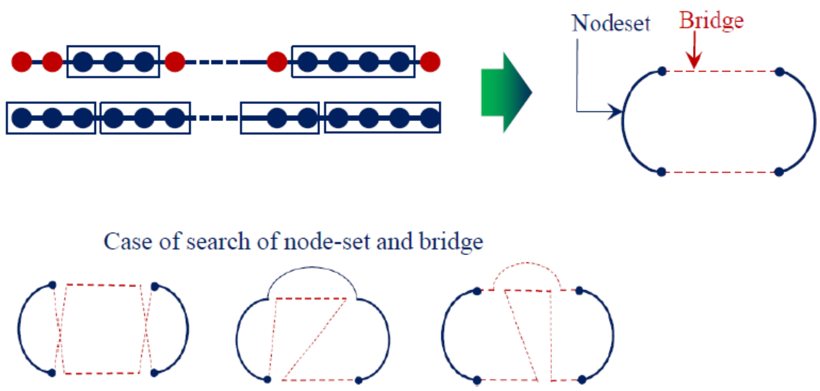


Figure 2. 13 Solution structure of structured local clustering.

Design of structured local clustering:

When it wants to make an intensification of search is in the search place to be able to be improvement, we consider where it can immobilize. And it apply to a structure of the solution consisting of nodes and node-sets. This problem structure abstracts it as node-sets and bridges. Node-sets are immobilized, and operation is applicable for node-sets and the between node-sets (bridge). Structured local clustering is as follows.

(A) For a intensification of the search, it examines to divide the structure of the solution into node-sets and bridges.

(B) The operation consisting of node-sets and bridges examines support with the matrix of the local clustering. The operation pattern on the matrix of the local clustering examines for each problem.

Move strategy:

A move strategy is how to accept or reject when cost turned worse. The original LCO is the sequence is accepted when the evaluation is improved. As another way, when it consider about another way of optimization in a local area, there is a strategy to move to the best solution in searches. Then, it accepts deterioration of solution. However, the aim is to find the anticipated better solution.

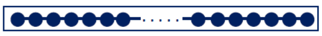
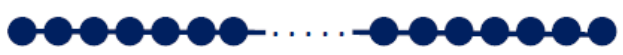
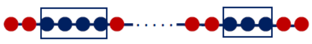
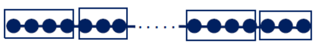
Solution type		How to use subset	Move strategy How to accept when cost turned worse.
Optimal			
Node		Non- use (node-exchange)	$C_{previous} > C_{next}$ or $C_{previous} \leq C_{next}$
Node and Node-set		Forward or Inverse	- (ditto)
Node-set		Forward or Inverse	- (ditto)

Figure 2. 14 Solution types and how to use node-set.

2.7. **Objective**

In TSP and JSP, the basic performance of structured local clustering is tested, and this dissertation shows an efficacy of the performance in comparison with the original LCO. The contents in Chapter 3 and Chapter 4 are as follows.

Multiplexed TSP:

It is the test of the minimization of connection of node-set, in the case that a solution is consisted node-sets only. It searches recombination pattern between node-sets. LCO is divided aptly into node-sets, but the combination is easy to cause a local minima, it searches recombination pattern between node-sets.

TSP (general TSP instances) :

First, it is tested the effect of extension added by using the matrix. It is the test of the intensification to bridges, in the case that a solution is consisted node-sets and bridges. A problem is divided in parts of node-sets and bridges. Here, the node-set is extracted as candidate sets of best solution from initial solutions. Then the solution is operated an encoding given by fixed node-set and the other is extracted as bridges including in parts of one node unit.

JSP (general JSP instances):

For a scheduling of job process on the same machine, it limits the neighborhood range of search. It makes a intensive search for the inside of each neighborhood. To divide encoding into a subset, it changes the solution structure to divide by each machine.

2.8. **Conclusion**

This section discussed the design method for structured local clustering with the diversification and intensification, as LCO operator to apply LCO to combinatorial optimization problems.

(A) LCO operator was defined based on SOM's learning law. This operator pattern is shown on matrix of local clustering. In local clustering, node-set based operation is effective as well as node-based operation. Then it also satisfy the learning law of original LCO.

(B) The positioning operation of original LCO was shown by using the matrix of local clustering. The operation variation is possible to analyze by this matrix. Based on this matrix, IM (insert method) as new operation was introduced for the extended local clustering. The extended local clustering bring the diversification of search.

(C) For an intensification of search, the structured local clustering was introduced. The structured local clustering is designed by extraction of node-sets, and examined the operation structurally using matrix. This aim is that operator brings the solution of non-search by the combination with the variation of operation, and brings an intensive exploration with reduction of the search space. A node-set is the characteristic that analysis of a mechanism to be related to an improvement in a search process reveals.

(D) In this dissertation, about TSP and JSP, the basic performance is tested. The applications introduced on extraction of node-set and the insertion method are discussed.

Chapter 3

3. Local Clustering Operation for Traveling Salesman Problem (TSP)

3.1. Introduction

This chapter discusses about the design of local clustering for "Traveling salesman problem" (TSP). The original LCO operator is extended as the "extended local clustering". Then the operation is positioned newly on the matrix of local clustering. For the "structured local clustering" applied to TSP, it elucidates the mechanism of the exploration. Then the extraction of node-sets is shown as one method of design how to improve the performance of extending neighborhood by the "extended local clustering". The extended local clustering brings the diversification of search, and the intensification of search is considered by the structured local clustering. The basic performance of extended LCO is tested using TSP instances.

This chapter is organized as follows. Section 3.2 raise an issue, and gives the concept of applying the structured local clustering. Section 3.3 analyzes the local clustering for LCO algorithm in TSP. Then Section 3.4 examines the extended local clustering. In Section 3.5, the structured local clustering in JSP is proposed. In Section 3.6, the new algorithm is experimented by "usa13509" as one of famous benchmark problems. Section 3.7 discusses the experimental results. In Section 3.8, another type of structured local clustering applied to multiplexed-TSP is introduced. Section 3.9 discusses about extension in this chapter and the related researches. Finally, the conclusion is stated in Section 3.10.

3.2. **Concept**

The theme in this chapter is an analysis of the mechanism related to the improvement to solve TSP, and the design of structured local clustering. LCO was developed based on a principle of SOM for TSP. LCO optimizes globally by repetition of the local optimization stochastically. LCO uses the cost between 2 cities instead of learning of the city coordinate in SOM. This conversion enables the application to general scheduling problems which is not proportional to the distance defined by city coordinates. However, LCO has the defect of operation that was easy to be caught in a local minima.

TSP is a problem to find a sequence minimizing cost while a round trip of each spot. It causes the difficult to bring the quality and dynamic division that a search domain is multimodal. LCO operates based on exchanging of 2 city nodes and 2-opt on a neighborhood. The combination of swap types is effective to improve the cross of tour every neighborhood within high speed. Then, while the learning equation gradually narrows a range, it is accompanied by local convergence that the operation is limited to neighborhood. LCO is an iterated local search that the search point is positioned randomly. This effect is to bring possibility to reach the diversified solution from the start point. Then, LCO repeats swap and first improvement of move strategy without a method to make confirmation whether searched or not in its search neighborhood. To avoid such condition, a fluctuation and a stack are studied.

This chapter approaches to give an indicator of the intensification for local search. The strategy in this study gives the node-sets of candidate of a best solution optimized locally on a node based permutation of same solution, as a “subtour”. The solution structure that fixed a subset fills the valley of some local minima, and concentrates a search on a bridge linked sub-tours. Even if a subtour is used forward or reversed, it compresses a node-based combination.

$$f_{cmb}(m) = \frac{(m-1)!}{2} \times 2^m \tag{1}$$

Subject to $\quad m\;$: number of subtour

This method is applied to LCO. It is structured local clustering for TSP, and the performance is evaluated.

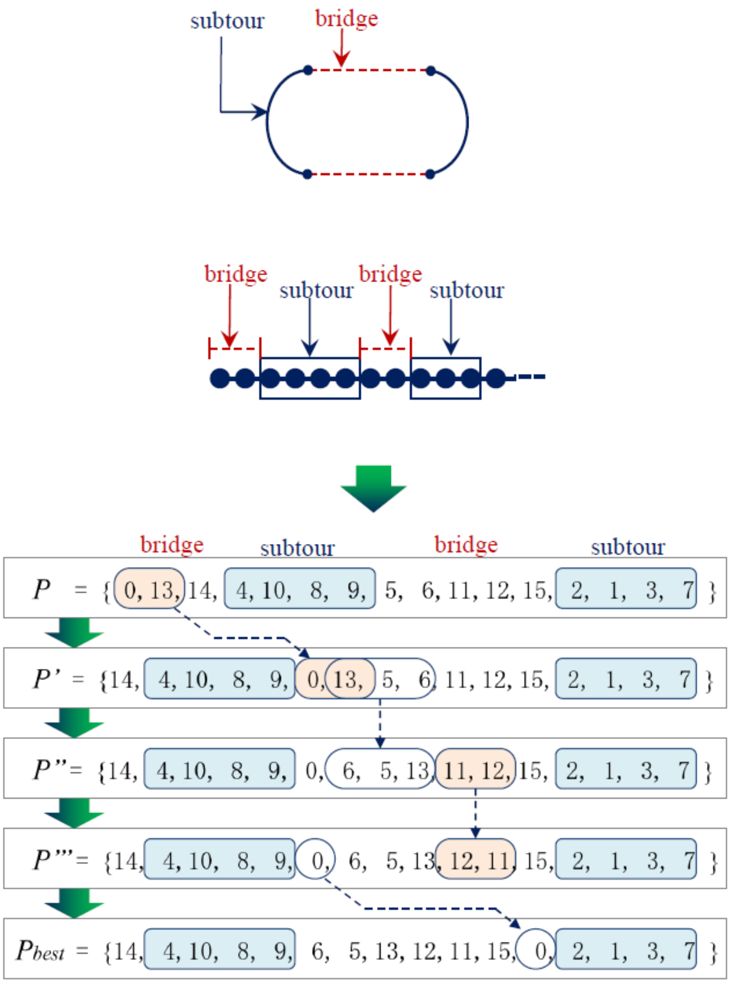


Figure 3. 1 Concept of structured local clustering for TSP.

3.3. **Local Clustering**

The original LCO operator is using 3 kinds of the clustering operations stochastic randomly, i.e. “Inverse Exchange method” (IEM), “Simple Exchange Method” (SEM) and “Smoothing Method” (SM). About the general flow of procedure is shown in Chapter 1. The ratio of the selection of each method is IEM: SEM: SM=2:2:1 in the original LCO for TSP. The original LCO has 2 kinds of parameters. One of them is the stochastic selection of clustering operations and another one is the stochastic selection of the positioning for the clustering neighborhood.

Inverse Exchange Method (IEM):

IEM repeats inverse-sorting of the tour between pick-upped 2 nodes in the clustering neighborhood. When the cost is improved, the tour sequence is accepted.

Simple Exchange Method (SEM):

SEM replaces the center node with the left or right node in the clustering neighborhood, and repeat exchanging the position of 2 nodes. When the cost is improved, the tour sequence is accepted.

Smoothing Method (SM):

In the local area, SM repeats inverse-sorting from the left side to right side and repeat inverse the position of 2 nodes. The starting point is shifted to right sequentially in the clustering neighborhood, so it is calculated in the total hits. When the cost is improved, the tour sequence is accepted.

Analysis:

A search space of LCO is constructed in a set of feasible tours. The behavior on the operation is explained in comparison with the λ-opt that is used in LKH method as high-performance heuristic for TSP [Helsgaun 00]. The λ-opt operator watches the set of edges and each edge's connection in one total tour. The improvement of the tour cost is played by the recombination of *k*-number of edges. This recombination behavior has a role for explaining the behavior that local operation of LCO gives in a search of one total-tour.

Inverse Exchange Method (IEM):

An inverse-operation in IEM causes the recombination of 2 edges of the ends of target partial tour on the structure of a tour. It makes cancellation of the crossing tour by reverse between 2 nodes. This recombination is equivalent to the operation of 2-opt.

Simple Exchange Method (SEM):

A position-replacing in SEM causes the recombination of 4 edges of both sides of 2 target nodes on the structure of a tour. This recombination is the operation of 4-opt provided that 2 target edges are considered 2 nodes. When IEM is applied 2 times, the double crossing may improve like SEM operation. But SEM is effective when one time of IEM is accompanied by the cost up.

Smoothing Method (SM):

An inverse-operation in SM causes the recombination of 2 edges of the ends of target partial tour on the structure of a tour. SM makes cancellation of the crossing tour by reverse between 2 nodes. This recombination is equivalent to the operation of 2-opt. This recombination of one operation is equivalent to the operation of IEM. But SM repeats by the starting point is shifted from the left side to right side in a clustering neighborhood, then it is calculated in the total hits.

Positioning of IM by matrix of local clustering:

SEM: *CO*(*N* , *Pc*, *Pi*, *Pj*)

IEM, IM: *CO*(*N*, *Pc*, *Bi*−, *Pj*, *Lterminal*)

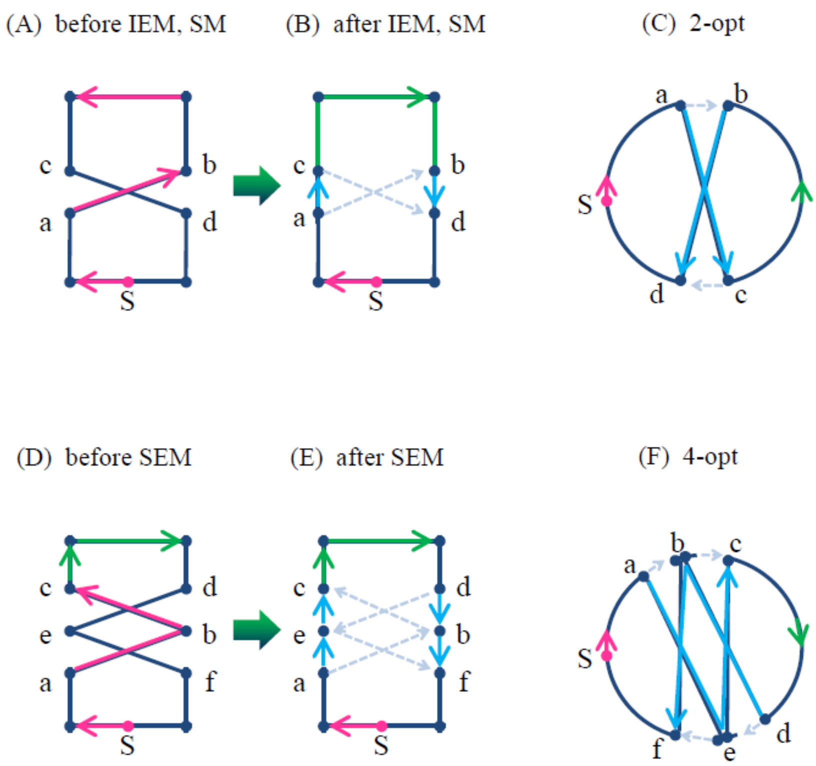


Figure 3. 2 Comparison of local clustering with λ-opt algorithm in 1 tour.

3.4. Extended Local Clustering

The local clustering of the original LCO operates based on 2-opt and swap of node sequence in a selected neighborhood. To expect more improvement of the solution, it is possible that the wide local area is selected. However it spends the time on the repetition of the search. This section examines an operation that move subset of nodes to outside of local area in the total tour. “Insert Method” (IM) operator cut the edge between 2 nodes of a certain length in then local area, then it is inserted in the place in prospect improvement in the total-tour. Then, it considers λ-opt operator, The LCO operator is not enough in comparison with λ-opt.

3-opt is required as first priority. IM is equivalent to 3-opt in comparison with λ-opt, and expands the variation of the solution by the search pattern. Now, the main variables for explanation of LCO applied to TSP are defined.

n : city number

n_{max} : numbers of city nodes

$P = \{Pc : c = 1, 2, \ldots\ldots, n_{max}\}$: set of solution nodes

$N(Pc, r) = \{ Pc\text{-}r, \ldots, Pc\text{-}d, \ldots, Pc, \ldots, Pc+d, , \ldots, Pc+r\}$

: neighborhood set of one element (Pc)

d, r : number of elements from Pc in a neighborhood

$S(Pc, j) = \{ Pc + j;\quad j = 0, 1, \ldots\ldots, j_{max}\}$: subset in tour: subtour,

j : subtour's city number

j_{max} : number of subtour nodes

Insert method (IM):

(step1) One of nodes Pc is chosen from a set P of nodes randomly, and a neighborhood $N(Pc, r)$ is decided.

(step2) Set an index of left side in a neighborhood $N(Pc, r)$ to L, set a right side of a neighborhood $N(Pc, r)$ to R, thus set an index L=c-r, and R=c+r. Set j=0 the length of subset, and dL=L+1 as the index of insertion. Then $S(P_L, j)$ is set from the node of L.

(step3) Cut edges both side of subtour $S(P_L, j)$. Remove $S(P_L, j)$ from the current position, and insert it in the other position dL in the tour.

(step4) Evaluate a scheduling of before insertion $F(P_{prev})$ and after insertion $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step5) Invert and insert subtour $S(P_L, j)$. to dL in the tour.

(step6) Evaluate a scheduling of before insertion $F(P_{prev})$ and after insertion $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step7) Set dL=dL+1. Then repeat (step3) and (step7) until dL=L-2. The insertion position is shifted and tried sequentially.

(step8) Set j= j+1. Repeat from (step3) and (step8) until j=$2r$.

Positioning of IM by matrix of local clustering:

CO(*N* , *Pc*, *Pi*, *Bj+*, *Lterminal*)

CO(*N* , *Pc*, *Pi*, *Bj–*, *Lterminal*)

		A_j		
	Pc	*Pj*	*Bj* +	*Bj* –
A_i	*Pi*	SEM (Swap⊂4-opt)	IM (Insert⊂3-opt)	IM (Insert⊂3-opt)
	Bi +			
	Bi –	IEM,SM (2-opt)		

Figure 3. 3 Positioning of IM.

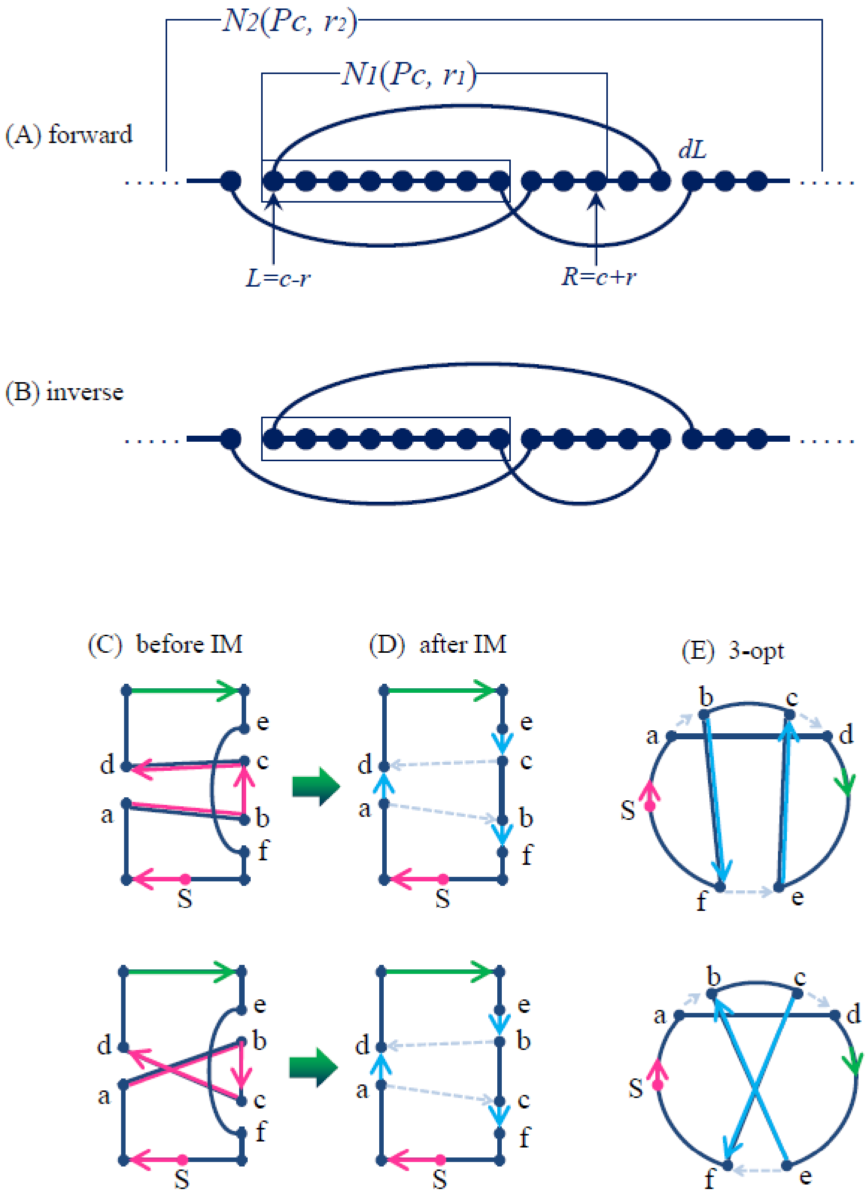


Figure 3. 4 Extended local clustering, IM.

3.5. Structured Local Clustering

Figure 3.7 is a graph of 2 converged solutions of ch130 that is one of the instances in TSP-LIB including local minima are superposed [TSPLIB]. When the TSP within 130 nodes forms 1 tour, each node connects 2 nodes to both sides. In this figure, the node that a connection node is 2 nodes and 3 nodes is seen. It is shown that there are nodes which have high influence for the local minima. Now, the set of edges that the connection nodes are fixed is extracted as a subtour for the candidate part of best tour by comparing multiple tours. In addition, the set of the nodes that connection nodes are unstable is extracted as a bridge. The total tour consists of some bridges and subtours.

"Edge based Inverse Exchange Method" (eIEM) and "Edge based Insert Method" (eIM) are introduced for "Subtour extract Extended LCO" (SELCO) operator. The edge based clustering operates with holding linked-node of subtour. That makes positioning in the insertion and recombination more effectively than node-based operation. The edge based operator compresses the search cost than the node-based operator.

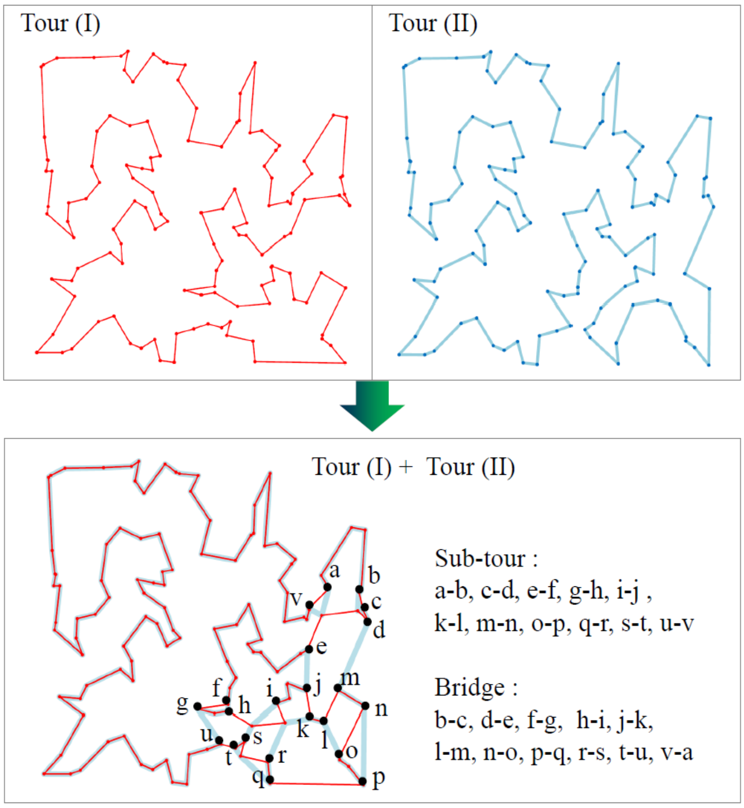


Figure 3. 5 An example of extract subtours and bridges from 2 tours in ch130 by SLCO.

Edge based Inverse Exchange Method (eIEM):

The "eIEM" repeats inverse-sorting of the tour between pick-upped 2 nodes in a clustering neighborhood. When the cost is improved by the exchange, eIEM accepts the tour sequence. Then, in operation of the position exchange, subtours included in the total tour are considered to be blocks and the nodes must not be taken to pieces.

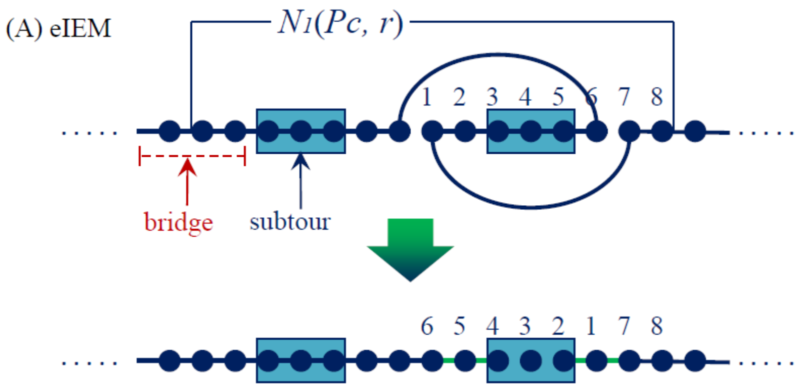


Figure 3. 6 Structured local clustering, eIEM.

Edge based Insert Method (eIM):

The "eIM" selects subset between pick-upped 2 nodes in a clustering neighborhood. Then, it removes this tour from the current position, and insert it in the other position of the tour. The insertion position is shifted and tried sequentially. It repeats insertion-sorting by changing the length of the tour. When the cost is improved by the insertion, accept the tour sequence. Then, in operation of the position exchange, subtours included in the total tour are considered to be blocks and the nodes must not be taken to pieces.

Positioning of eIEM and eIM by matrix of local clustering:

eIEM

CO(N, Pc, Bi–, Pj, Lterminal)

CO(N, Bc, Bi–, Pj, Lterminal)

eIM

CO(N , Pc, Pi, Bj+, Lterminal)

CO(N , Pc, Pi, Bj–, Lterminal)

$CO(N, \quad Bc, Pi, Bj+, Lterminal)$

$CO(N, \quad Bc, Pi, Bj-, Lterminal)$

		Aj		
	Pc	Pi	Bj +	Bj –
Ai	Pi		eIM	eIM
	Bi +			
	Bi –	eIEM		

		Aj		
	Bc	Pi	Bj +	Bj –
Ai	Pi		eIM	eIM
	Bi +			
	Bi –	eIEM		

Figure 3. 7 Positioning of eIEM and eIM.

How to extract subtour:

For extract ssubtour, multiple initial-tours are prepared. One tour have best cost is selected. The overlapping edges are extracted from multiple initial-tours for subtours included in this best total-tour. One subtour has more than 2 nodes which linked. 2 kinds of parameters are prepared for this procedure. One of them is the population size G of initial tours and another one is the ratio g that extracts partial tour overlapping in the population.

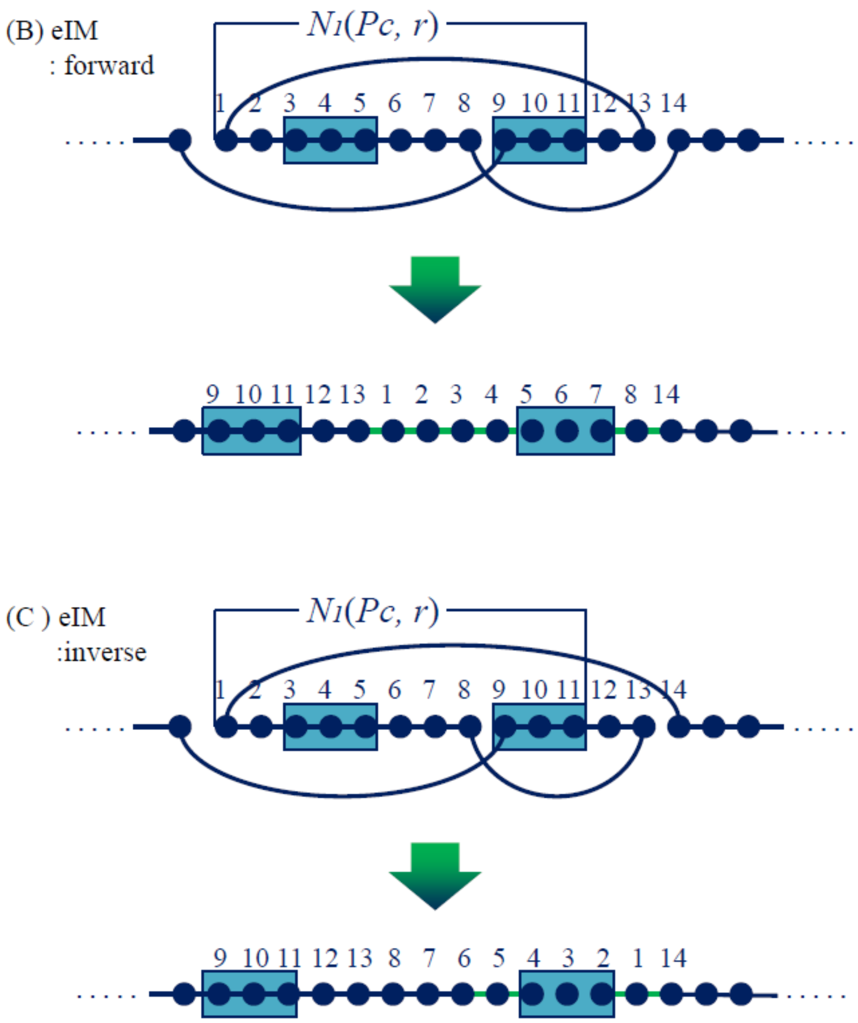


Figure 3. 8 Structured local clustering, eIM.

3.6. Experimentation

The objective of this experimentation is a performance test of comparing the original LCO with the extended local clustering (ELCO) and the structured local clustering (SLCO). The scale of problem of verification is the problem of on one instance “usa13509” of TSPs within 3000 steps as execution termination. The instance is known optimal cost in TSP-LIB [TSPLIB, NTSP].

The neighborhood was set the upper limit to 2/3 of total nodes determined randomly. LCO used the mixed clustering of 4 kinds of original methods randomly with the probability was IEM:SEM:SM=2:2:1 for the stochastic selection of the clustering operations in accordance with the original LCO. As an application of extended local clustering, ELCO (extended LCO) used 2 kinds of operation added IM randomly with the probability was IEM:IM=1:1. In addition, as ELCO-SEM used IEM:SEM:IM=1:1:1. As an application of structured local clustering, SLCO (structured LCO) used 2 kinds of mixed clustering with the probability of eIEM: eIM=1:1, which are subtour extraction based operation. For extraction of subtour, it used G=10 initial tours and, g=0.5, that is extracted partial tour included in 5 initial tours which had high evaluation in 10 initial tours. An initial tour and the repeat factor of operation were fixed uniquely. Only in SLCO, the multiple initial tours were created in every run.

This experimentation set an initial tour first for comparing the convergence. The initial tour is obtained fast by using IEM of LCO operator of an equivalent to 2-opt. Those are once reached to an initial solution stage, and the performance of improvement from the next stage is verified. Greedy algorithm, 2-opt or those combination are used for the construction of the initial-tour of large-scale TSP [Nguyen 07, Nagata 07].

Table 3. 1 Obtained initial tours of usa13509 TSP by IEM.

.

Method	Benchmark (opt.)	Best (%)	Worst (%)	Average (%)	Time_avr. (min)
IEM	usa13509 (19,982,859)	0.133	0.145	0.140	3.380
LCO	-	0.152	0.154	0.153	2.315

The values are percentage above the best/ lower bound. Times were measured on a 2300MHz AMD Opteron Processor-6134 PC. (Non-improvement counter for the termination=1000, trial=10)

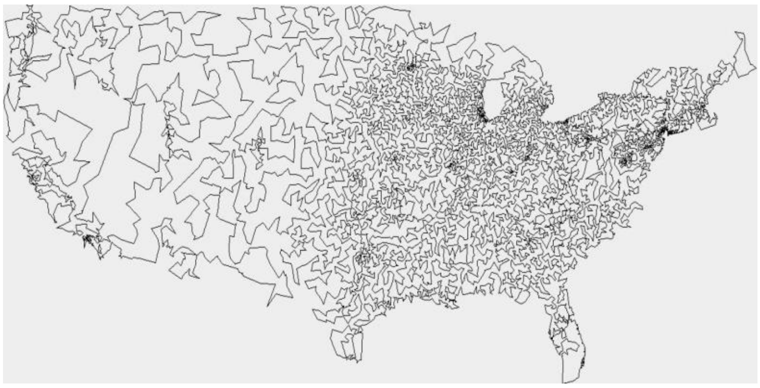


Figure 3. 9 The obtained initial tour of usa13509 TSP by IEM. .

3.7. **Results**

About the process of improvement cost, it is focused to the graph of Figure 3.11. It is the result of the search process comparison between LCO, ELCO, ELCO-SEM and SLCO. Almost none improvement was seen in LCO after the initial stage, and SLCO curved most. About the average obtained tour cost by each method and the standard error of best and worst, Figure 3.12 shows that the operation introduced IM was improved the convergence of LCO. SLCO had best solution and best average, so that improved a local minima of ELCO. However, the upper solution was a little improvement. The accuracy is shown in Table 3.2. The accuracy when each method had terminated is LCO:ELCO:SLCO = 1.00:0.61:0.59. The accuracy also that SLCO improved the performance of the average and best-tour than ELCO. Then the worst was not so different. (SLCO→ELCO→ELCO-SEM→LCO: The best was SLCO, and the ELCO followed.)

About the processing time of improvement cost, it is focused to the graph of Figure 3.13 that is the improvement process in time as y-axis. LCO had high-speed than other operator. The comparison of accuracy when LCO had terminated is

LCO:ELCO:SLCO = 1.00:0.73:0.68. (LCO→ELCO-SEM→SELCO→ELCO: The best was LCO, and the ELCO-SEM followed.) In the same operation running count, LCO was little improvement, and the convergence of the solution had already carried at the initial solution. However, LCO has the operability that is high speed. SLCO starts a little late for getting multi initial-tour. That elapsed time was 12 - 20 minutes.

About the contribution rate in the improvement, it is focused to the graph of Figure 3.14 and Figure 3.15. LCO used IEM and SM for operation, and SEM was not used. In the case of operation for IM plus SEM for IM-SEM, SEM was used a little. However, ELCO improved the tour more than ELCO-SEM. According to the processing time of Figure 3.13, the ELCO and ELCO-SEM spent the time by IM. ELCO-SEM put the time shorter than ELCO by selecting of SEM. SLCO became quicker in the time for improvement than ELCO.

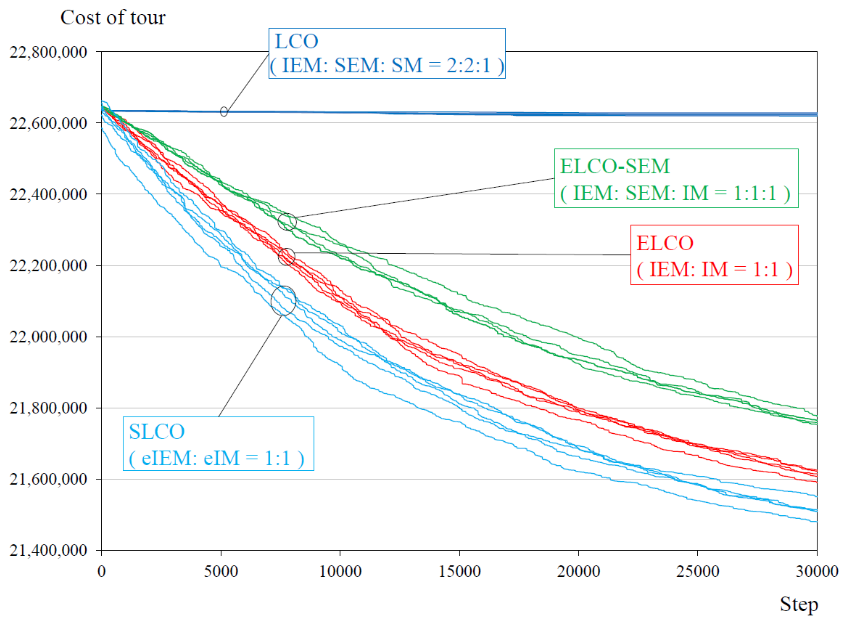


Figure 3.10 Comparison of the search process by LCO, ELCO, ELCO-SEM, and SLCO in usa13509. (x-axis: cost, y-axis: max 30000 steps as termination, 5 trials)

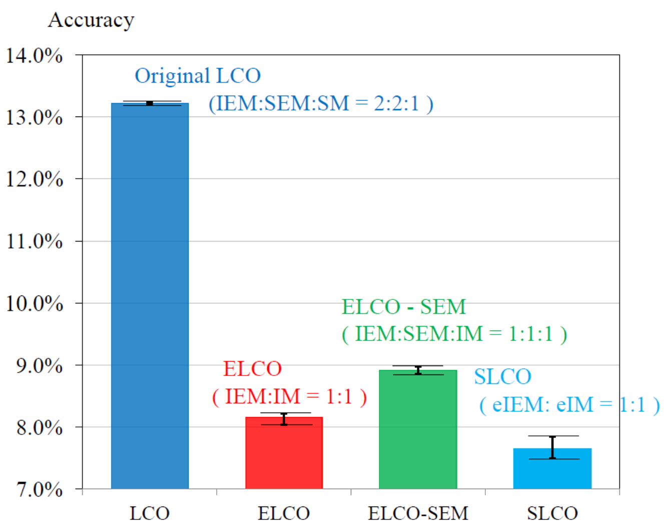


Figure 3.11 Average and the error of obtained cost by LCO, ELCO, ELCO-SEM, and SLCO in usa13509.

Table 3. 2 Accuracy of obtained cost by LCO, ELCO, ELCO-SEM, and SLCO in usa13509

Method	Best (%)	Worst (%)	Accuracy (%)
LCO	0.1320	0.1324	0.1322
ELCO	0.0804	0.0822	0.0815
ELCO-SEM	0.0886	0.0897	0.0891
SLCO	0.0766	0.0785	* 0.0750

* : significant $p < 0.1$

The values are percentage above the best/ lower bound. Times were measured on a 2300MHz AMD Opteron Processor-6134 PC. (Iteration=30000, trial=5)

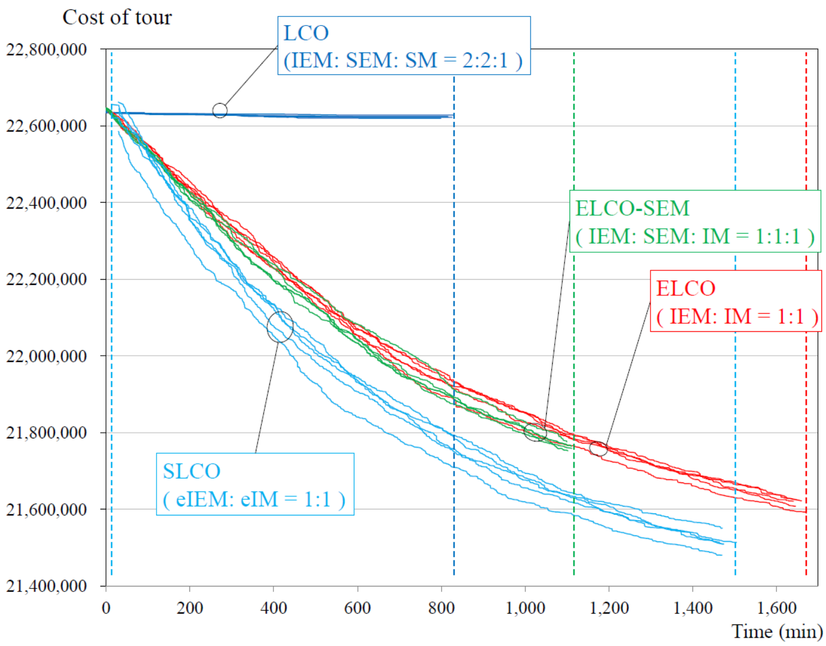


Figure 3.12 Comparison of the search process by LCO, ELCO, ELCO-SEM, and SLCO in usa13509. (x-axis: cost, y-axis: minute in 30000 steps, 5 trials)

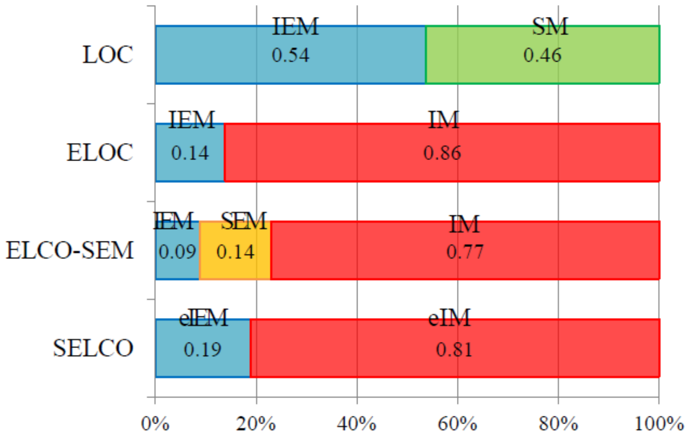


Figure 3.13 Ratio of contribution on number of improvement by LCO, ELCO, ELCO-SEM, and SLCO in usa13509.

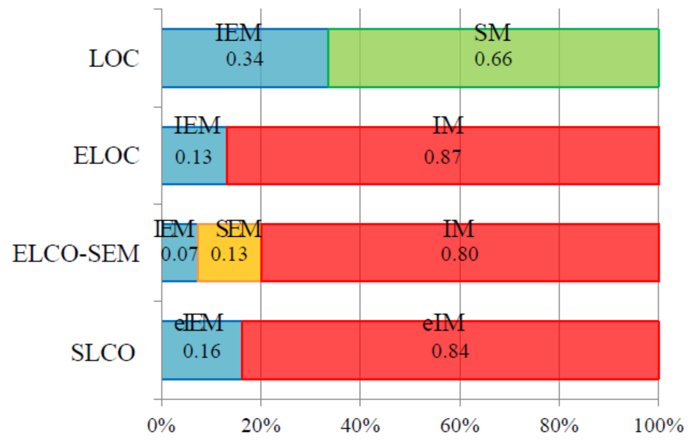


Figure 3.14 Ratio of contribution on cost of tour by LCO, ELCO, ELCO-SEM, and SLCO in usa13509.

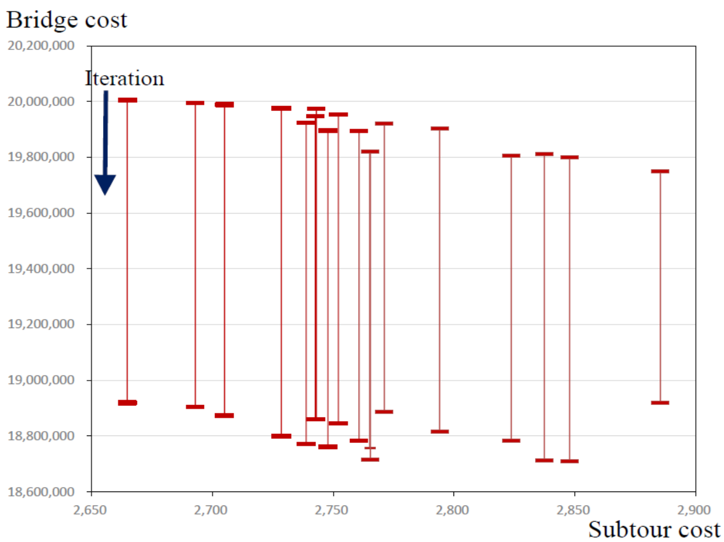


Figure 3.15 Cost to decrease by improvement of SLCO's bridges in usa13509.

Table 3. 3 Initial parameter for extract node-sets and bridges

Initial parameter g/G	Accuracy (%)	Node-set			Bridge		
		No.of node-sets	No.of nodes	Cost	No.of bridges	No.of nodes	Cost
2/10	0.0860	3,662	9,219	6,433,329	3,662	4,290	16,195,704
5/10	0.0750	2,474	5,561	2,764,829	2,474	7,948	19,899,712
8/10	0.0745	1,854	4,009	1,717,932	1,854	9,500	20,938,072

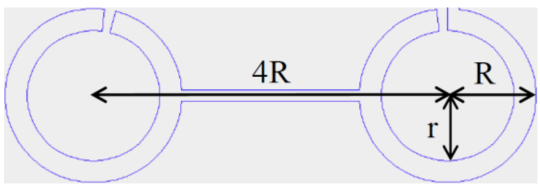
Table 3. 3 shows an initial parameter for extract node-sets and bridges. The node-sets extract a overlapping part from multiple initial tours. Then the best tour is partitioned into node-sets of the subtours and bridges. The number of initial solutions *G*, and better solutions *g* are used these node-sets, and these are the parameters for the structured local clustering. (ratio =*g*/*G*) Table 3.3 shows a comparison of the setting of initial parameters. In usa13509 instance, where it employ many initial solutions, the extraction node-sets decreases, then accuracy is improved. This effect depends on the problem, so an investigation in each TSP is necessary.

3.8. **Multiplexed TSP**

The other type of structured local clustering is examined in the instances which multiplex TSP of a continuous shape with the multi-optimal-solutions. LCO to treat in this section searches for a combination of best connection between the subtours without the bridge. Now, the connected type of double-circles TSP that is easy to divide was used as the instance of multiplexed TSP. When all of subtours which divided into some sub-problems are brought to optimal tours, the best solution is obtained by optimally combine operation. An advantage of operator applied to TSP is given by the making efficiency and accuracy of optimization and combination of divided sub-problems with the quality of division of the problem. Those are able to explain also as follows, (1) approach to divide (2) approach to combine (3) conquest method of divided sub-problems. Then, the single of double-circles TSP was optimized by ELCO.

Setting of multiplexed TSP instance:

There are C-type and O-type in the shape of optimal solution in the double-circle TSP, this experiment use C-type [Yamamura 9]. Single double-circle TSP is arranged into a lattice shape, and the length between 2 centers of neighbor circles=4*R*. The division unit=100 nodes is set the number of nodes of 1 double-circle.



Number of node in 1 double-circle	100
Ratio of the radius	R : r = 4:3
Shape of optimal solution	C-type
Length between 2 double-circles	4R (each center)

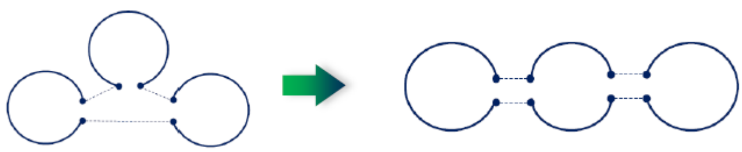


Figure 3. 16 Connected double-circle used in the experiment.

Table 3. 4 Obtained solutions of 100 cities double-circle TSP using LCO, ELCO, and GA

Method	Best	Worst (%)	Average (%)	Arrival at opt.	Time (sec)
LCO	opt.	0.352	0.160	2/10	0.027
ELCO	opt.	-	-	10/10	0.069
GA (GSX2)	opt.	-	-	10/10	1.992

The values are percentage above the best/ lower bound. Times were measured on a 2300MHz AMD Opteron Processor-6134 PC. (Non-improvement counter for the termination=10, trial=10)

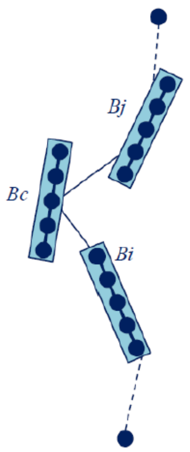
Positioning of connection operation of each node-set by matrix of local clustering:

CO (*N*, *Bc*, *Bi* +, *Bj*+, *Lmin*)

CO (*N*, *Bc*, *Bi* –, *Bj*+, *Lmin*)

CO (*N*, *Bc*, *Bi* +, *Bj*– , *Lmin*)

CO (*N*, *Bc*, *Bi* –, *Bj*– , *Lmin*)



		Aj		
	Bc	*Pi*	*Bj* +	*Bj* -
Ai	*Pi*			
	Bi +		xIM	xIM
	Bi -		xIM	xIM

Figure 3. 17 Positioning of eIEM and eIM.

Design of the algorithm:

(step1) Total tour initialization: Initialize the total tour.

(step2) Division and subtour optimization: Divide the total-tour to d sub-tours sequential by unit=u nodes. Thus, total number of nodes n_{max} is given $d \times u$) Also for this division method, the application algorithm by Euclidean distance may be considered.

(step3) Subtour conquest: Improve each subtour by ELCO.

(step4) Subtour couplingt: Make a order to connect subtours. Pick up nodes from each subtour one by one, and optimize a degeneration tour that connected all subtours by ELCO. By a degeneration tour, calculate the coupling. Arrange to connect a subtour having high coupling. As for this method, Karp's partitioning algorithm is referred [Karp 72, 77, Arora 98].

(step5) Subtour combination: Construct totaltour by using insertion combination that one subtour is inserted to another subtour by the search of the position for minimizing cost, then it is calculated in the total hits. The forward direction and inversion are considered. Then, the next subtour connects to the tour that was already combined.

(step6) Total-tour improvement: Improve to total-tour by using ELCO with the parameter of local area size less than $2u$ randomly.

Table 3.4 shows the result in 1double-circles by LCO, ELCO, and GA (crossover: GSX2). ELCO covered the accuracy of LCO. Table 3.5 is the elapsed time and the accuracy by SLCO in the number of connection is 25-200. The result was optimized, and the time without initialization is less than 1 minute. In Figure 3.18, the circles have nested structure of neighbor circles having the high coupling.

Table 3. 5 Parameters and operations used for experiment.

LCO	IEM:SEM:SM=1:1:1
ELCO	IEM:IM=1:1
GA	operation and parameter
population size	20
max generation	1000
mutation rate	0.1
elite size rate	0.1
crossover	GSX2
mutation	4 double bridge (non sequential 4-opt neighborhood)
total tour improvement method (for hybrid)	LCO (IEM:SEM:SM=1:1:1)

Table 3. 6 Elapsed time and accuracy by LCO for connected double-circles TSP

Number of connection	**Number of node**	**Arrival at opt.**	**Search time without initialization(min)**
25	2,500	5/5	0.008
50	5,000	5/5	0.026
100	10,000	5/5	0.076
150	15,000	5/5	0.416
200	20,000	5/5	0.665

Times were measured on a 2300MHz AMD Opteron Processor-6134 PC. (Non-improvement counter for the termination=10, trial=5)

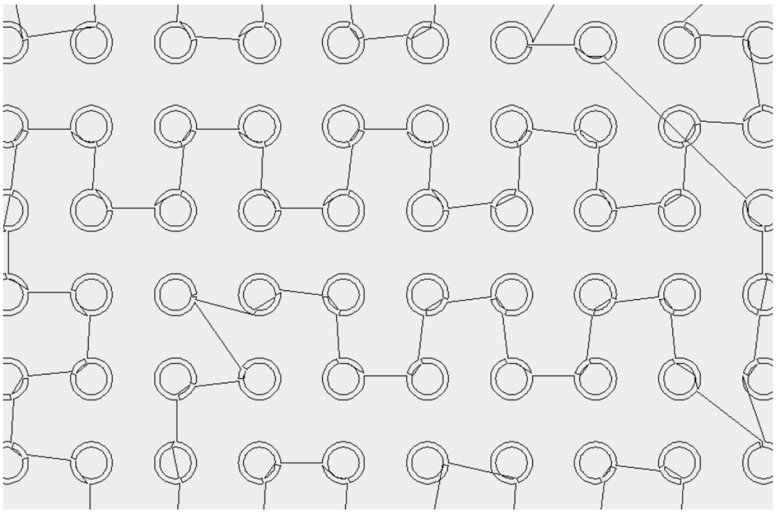


Figure 3. 18 Example of local minima which failed in combination.

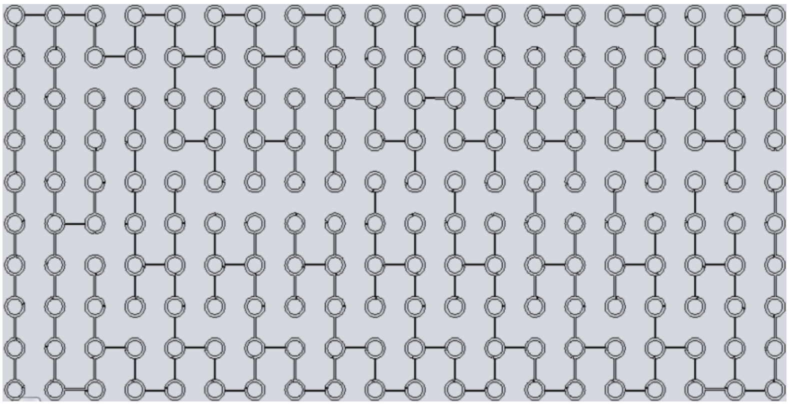


Figure 3.19 Obtained solution by ELCO and D&C applied to the 200-connected double-circles TSP.

3.9. **Discussion**

Search neighborhood:

TSP is a problem to find a sequence minimizing cost while a round trip of each spot. It causes the difficult to bring the quality and dynamic division that a search domain is multimodal. Now, in Figure 3.16, one local minima of spot A is able to be escaped to the search of spot B. The line is as a node-based search of LCO. For example, the move strategies of LCO are considered giving fluctuation, or giving its taboos.

The clustering operation of LCO operates node sequence in the chosen local area randomly. Now, this set of chosen local area is defined as a clustering neighborhood $N(Pc, r)$, where r is the number of nodes are including local area. The $N(Pc, r)$ is one part of 1 tour. The neighborhood is instead of using synopsis vectors in self-organization. In the original LCO, this clustering algorithm consists of sorting. Three types of sorting methods were proposed assuming that partial tour in $N(Pc, r)$ is connected in the sequence. In this extending, the behavior of 3 clustering operations is analyzed.

In analyzing of the original LCO, the clustering operations are explained as compares with λ-opt method. The λ-opt is used in LKH heuristic that is known as one of high-performance heuristics for TSP. The λ-opt is one of most basic search operator in ILS to solve TSP. The λ-opt operator watches the set of edges and each edge's connection in one total tour. The improvement of the tour cost is played by the recombination of chosen k edges. This analysis reveals that the 3 types of clustering operation of the original LCO based on 2-opt and exchanging nodes. Then this chapter introduces a new method of 3-opt type that is one of important operations to LCO.

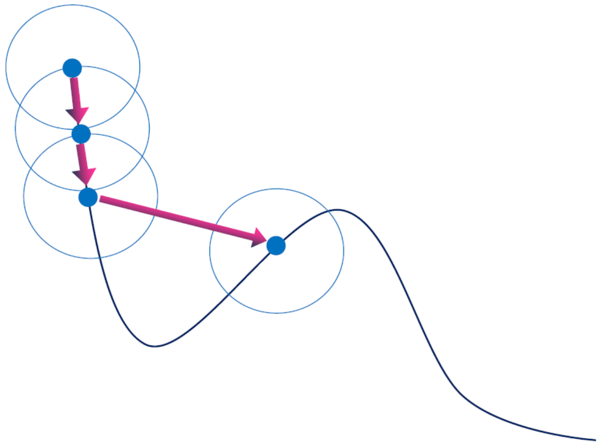


Figure 3.20 Search neighborhood.

Intensification:

Furthermore, as another one of the extending to improve, it examines an intensive search on the search space. For an intensive search, it is necessary to evade a loop and the stagnation of the search. It suggests to memorize where LCO should search in the total route in repetition of searched neighbor states, in large-scale TSP. In this study, 1 tour is divided into subtours and bridges. Then each subtour keeps the node sequence on a search process and concentrates the search on sort of bridges. It is based on the hypothesis that LCO can find the solution which is better than the current local minima, when LCO optimizes bridges that an existing solution has subtours and binds connect between subtours.

For the extract of subtours, it exploits the search process of multiple initial solutions. The overlapping edges of multiple initial-tours are extracted as subtours those are a set of edges of partial solution candidate. The subtour keeps a linked node and makes positioning in the insertion and recombination more effectively than node-based operation. The extract of subtour explains based on "Memetic algorithm" (MA) [Nguyen 02, 07, Schleuter 97]. Hybrid methods of GA and ILS are known as MAs and "Genetic local search" algorithms which are effective to solve TSP. The operation of crossover of GA exploits overlapping partial tour of the population-based solutions as individual to leave offspring of next generation. Then, the ILS is applied to a tour while keeping an inherited partial tour. Its operation is related to subtour, though it does not correspond. "Edge assembly crossover" (EAX) of GA is known as one of highest performance crossovers [Nagata 07]. It generates the multiple tours based on combination of partial tours of 2 individuals, and search for the solutions of the neighborhood range of the individuals.

Table 3. 7 The approach for extension of LCO

Method	**Added operator**	**Added strategy**
Original LCO (LCO)	Swap neighborhood 2-opt neighborhood	Randomness of the selection of local area and operation methods
Extended LCO (ELCO)	Insertion neighborhood 3-opt neighborhood	Extension of neighborhood
Sructured LCO (SLCO)	Node-set based operation with subtour extract	Parts and construction by exploitation of process

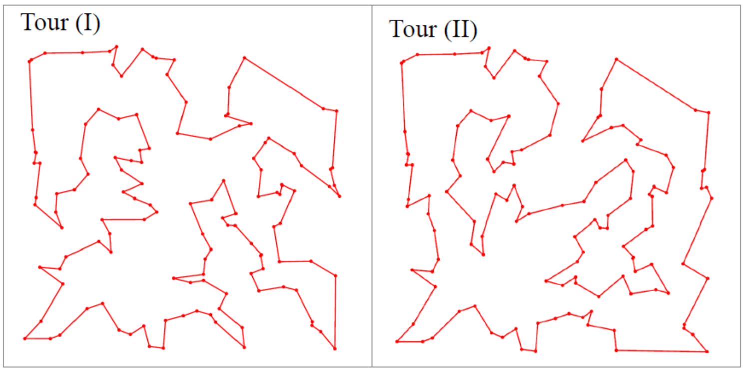


Figure 3.21 Different shapes of obtained solution by proposed method ELCO applied to ch130.

3.10. **Conclusion**

This chapter elucidated that the analysis of the mechanism related to the improvement to solve TSP, and the design of extended local clustering and structured local clustering for LCO. This chapter elucidated as follows.

(A) LCO was developed based on the principle of SOM. The clustering method was based on swap in the neighborhood, and that was easy to be caught in a local minima. In analysis of the operation of LCO by comparison with λ-opt method, each clustering operation of LCO was assigned to the 2-opt and 4-opt based operator, then 3-opt type was lacking. Then, IM (Insert method) as the extended local clustering clustering operation of 3-opt type was introduced to ELCO (Extended LCO).

(B) For making a concentration of search, it was proposed that the method 1 tour is divided into subtours and bridges. Then each subtour kept the nodes sequence on a search process as partial solution candidate, and concentrated the search on sort of bridges. This concentration of the search was introduced to SLCO (Structured LCO). SLCO operator was extended to edge-based operation, that compress the search cost than the node-based operator.

(C) The performances were analyzed focusing on “usa13509” instance. The accuracy when each method had terminated is LCO:ELCO:SLCO = 1.00:0.61:0.59. The comparison of accuracy when LCO had terminated is LCO:ELCO:SLCO=1.00:0.73:0.68. Almost none improvement was seen in LCO after the initial stage and SELCO improved most. In the processing speed time, LCO was fastest, but so little improvement. SLCO improved the speed from ELCO, it had an effect of the concentration of search.

(D) In analysis the contribution rate of each operation method in the improvement, IM was contributed most in the improved cost and times. It was used over 80 percent for IM, but IM spent the search time in the speed. In future work, the improvement of the concentration of search is examined.

(E) Structured local clustering for subtour connect without bridge, connected type of double-circles TSP that is easy to divide was used as the instance of multiplexed TSP. The subtours were optimized by ELCO, and calculated combination order, then those were combined by the search of insertion and inversion. This result led the optimal solution in possibility of 100%.

Chapter 4

4. Local Clustering Operation for Job-shop Scheduling Problem (JSP)

4.1. Introduction

This chapter introduces about the design of structured local clustering applied to "Job-shop scheduling problem" (JSP). It elucidates the mechanism of the exploration in JSP, and shows one method of design how to improve the performance by extending LCO. This chapter is organized as follows. Section 4.2 raise an issue, and gives the concept of applying the structured local clustering. Section 4.3 defines JSP. Section 4.4 explains the LCO algorithm for JSP. Section 4.5 introduces local clustering to JSP. Then Section 4.6 examines the extended local clustering. In Section 4.7, the structured local clustering in JSP is proposed. In Section 4.8, the new algorithm is experimented by "10 Though Problems" [JSP 98, 00]. The scale of JSP is the problem has 15 ~ 30 jobs. Also the problems having 50 ~ 100 jobs with over 500 elements produced at random for this experimentation. The convergence is tested by values obtained in the short term and long term. Section 4.9 discusses the experimental results. Finally, the conclusion is stated in Section 4.10.

4.2. Concept

The theme in this chapter is an analysis of the mechanism related to the improvement to solve JSP, and the design of structured local clustering. To solving

TSP, the encoding of LCO solution is a permutation that is arranged city elements. The clustering algorithm for TSP is a sorting by referring the cost between 2 cities. Then, to extract subtours as candidate part of best tour is bring the intensification of exploration.

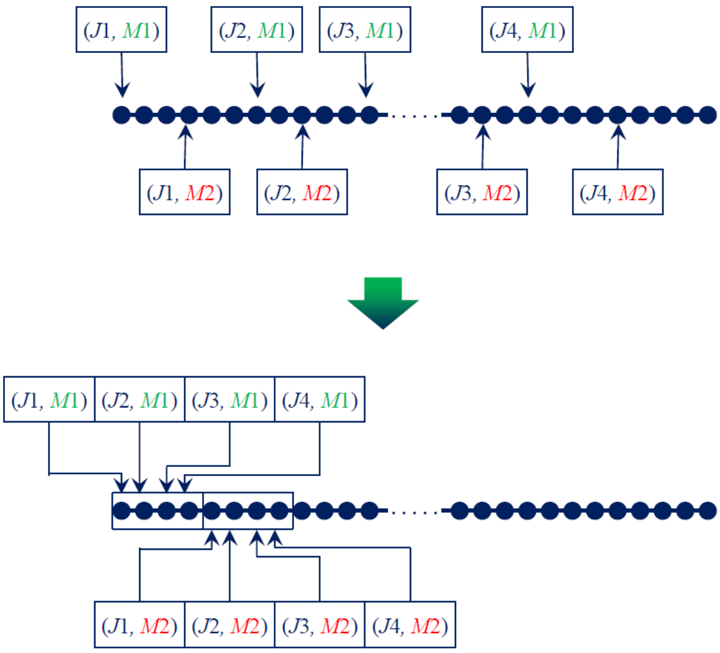


Figure 4. 1 Concept of structured local clustering for JSP.

JSP is an optimization problem which ideal jobs are assigned to resources at particular times. A resource is considered as one machine. In the case of JSP, the encoding of solution is a set of elements meaning a job and machine number. The machine process is arranged by the same job number. The clustering algorithm for JSP is a sorting similar TSP. The evaluation function of JSP calculates the schedule with a Gantt-chart of machines assigned job-processes. In other words, it shows that the scheduling of jobs to assign in the same machine is related to the total cost.

So, it examines the sorting method that is concentrated on scheduling of a job process in a same machine shown by a Gantt-chart.

Now, it supposes that can extract jobs meaning same machine from a permutation of multiplexed job number. For an approach of this function, a solution structure divided by each machine is proposed. In addition, LCO operates local clustering in each machine process. This method is applied to LCO. It is structured local clustering for JSP, and the performance is evaluated.

4.3. **Job-Shop Scheduling Problem (JSP)**

JSP is one of the most important optimization problems for jobs are assigned to resources at particular times. It is known as NP-hard. We are given j_{max} jobs {$J1$, $J2$, … , Jj_{max}}, which need to be scheduled on m_{max} identical machines {$M1$, $M2$, … , Mm_{max}}. We try to minimize the "makespan". The makespan is the total length of the schedule, that is, when all the jobs have finished processing. Each job is given a order of machines with required time. Each machine is not able to process more than one job at a time [Sait 01, Kuroda 02, Ou 08].

4.4. **LCO for JSP**

To apply LCO to JSP, the encoding of solution, and the method that evaluate the encoding, i.e. "dispatching rule" are necessary. Then LCO operates the solution elements by application of its clustering and the evaluation. Now, the main variables for explanation of LCO applied to JSP are defined.

$J = \{Jj \quad : j = 1, 2, \ldots\ldots, j_{max} \quad \}$: set of Jobs

$M = \{Mm : m = 1, 2, \ldots\ldots, m_{max}\}$: set of machines

j : job number

m : machine number

j_{max} : number of all jobs

m_{ma} : number of all machines

J_{jm} : process of job (J_j) assigned machine (M_m)

$S_m(j)$: total process from start to job (j) assigned machine (M_m)

$P = \{Pc : c = 1, 2, \ldots\ldots, j_{max} \times m_{max}\}$: set of solution nodes

$N(Pc,r) = \{ Pc\text{-}r, \ldots, Pc\text{-}d, \ldots, Pc, \ldots, Pc+d, , \ldots, Pc+r\}$

: neighborhood set of one element (Pc)

d, r : number of elements from Pc in a neighborhood

4.4.1. **Encoding**

The element of encoding of solution for JSP is a job number as an ordering factor. Now, we consider that convert the element to “node” operated by local clustering. It expresses by a set of multiplexed job numbers meaning different machine number. That is a machine process is arranged by the same job number. Depending on preference relations of machines in a problem definition, a machine number is assigned to a job by an appearance order of jobs. The method to interpret a set of nodes, pick up a node sequentially from the left. A job is assigned the applicable machine number and processing time, while make a Gantt chart. The objective function of scheduling is a minimization of a "Maximum completion time" when all nodes are assigned, i.e. a minimization of the “makespan”.

4.4.2. **Dispatching Rule**

The encoding and an evaluation of scheduling are shown as follows by using the instance given 2-job 3-machine.

(step1) A set P of nodes are given by job number. A set P is generated randomly.

$$P = \{1, 2, 2, 1, 1, 2\} \tag{1}$$

(step2) Then one of nodes pick up from a left side in a set. A machine number is assigned by an appearance order of jobs, according to preference relations of machines in a problem definition.

$$\{ J_{11}, J_{22}, J_{23}, J_{12}, J_{13}, J_{21}\} \tag{2}$$

(step3) Table 4.1 is a definition of JSP, and Figure 4.2 is the Gantt chart. A process of Table 4.1 as the problem definition in a set P is as follows. In making the Gantt-chart of Figure 4.2, the start time of each job considers the previous end time of same job number, and the previous end time of same machine number.

$$S_1(2) = \{J_{11}, J_{21}\} \tag{3}$$
$$S_2(2) = \{J_{22}, J_{12}\}$$
$$S_2(2) = \{J_{22}, J_{12}\}$$

Table 4. 1 Example of Process of 2Job 3Machine

Process	1 [Machine, Time]	2 [M, T]	3 [M, T]
J1	[M1, T30]	[M2, T50]	[M3, T10]
J2	[M2, T40]	[M3, T40]	[M1, T20]

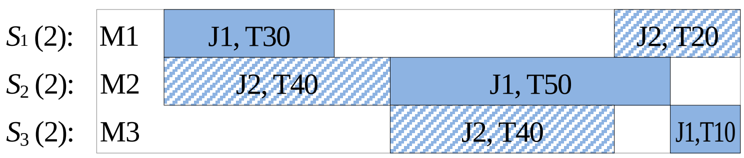


Figure 4. 2 Gantt chart given by P for example of JSP .

4.4.3. **Algorithm and Objective Function**

(step1) A set P of nodes is given by arranged randomly from r=1 to $j_{max}\times m_{max}$. This is as an initial solution $P0$.

(step2) A set P is considered the nodes form one ring which connects between r=1 and $j_{max}\times m_{max}$.

(step3) One of nodes Pc is chosen randomly. And a neighborhood $N(Pc, r)$ is decided. Then, the upper bound of $2r$ is set $2/3 \times (j_{max}\times m_{max})$ [Furukawa 06].

(step4) LCO plays local clustering in a neighborhood $N(Pc, r)$. If the scheduling is improved after clustering, the solution is accepted. An objective is a minimization of the makespan.

$$F(P) = \min (F_1(P)) \quad (4)$$

Subject to $F_1(P)$: maximum completion time

$$F(Pt) \geq F(Pt\text{-}k)$$

Subject to $F(Pt)$: evaluation in step t

$F(Pt\text{-}k)$: best evaluation from start(step 0) until step $t\text{-}k$

(step5) The termination is time not to be able to improve constant steps, or when step reaches a number designated. Otherwise, (step 3) and (step 4) are repeated.

4.5. **Local Clustering**

LCO has 4 kinds of local clustering operation, i.e. "Simple exchange method", "Inverse exchange method", "Smoothing method" and "Symmetrical exchange method". These are selected stochastically in each step.

4.5.1. **Local Clustering Algorithm**

Simple exchange method (SEM):

SEM replaces a center of node with the right or left node in a clustering neighborhood. And repeat exchanging the position of 2 nodes. When the evaluation is an equivalent or improved, the sequence is accepted.

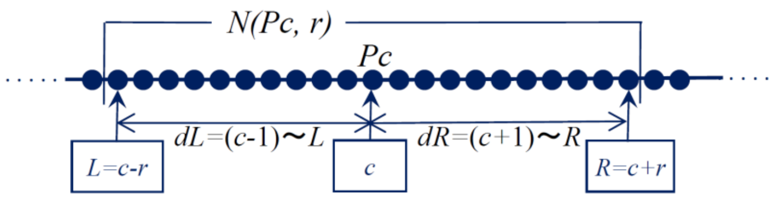


Figure 4. 3 Clustering neighborhood for JSP.

(step1) One of nodes Pc is chosen from a set P of nodes randomly, and a neighborhood $N(Pc, r)$ is decided.

(step2) Set an index of left side in a neighborhood $N(Pc, r)$ to L, set a right side of a neighborhood $N(Pc, r)$ to R, thus set an index L=c-r, and R=c+r. Set $CL=CR=c$.

(step3) Set $dL=CL$-1, and $dR=CR$+1.

(step4) Exchange P_{CL} for P_{dL}.

Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step5) Exchange P_{CR} for P_{dR}.

Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step6) Set dL=dL-1, and set dR=dR+1. Then repeat from (step4) to (step6) until $dL=L$ and $dR=R$.

(step7) Set CL=CL-1, and CR=CR+1. Repeat from (step4) to (step7) until CL=L+1 and CR=R-1. When CL=L and CR=R, the procedure is end.

Inverse exchange method (IEM):

IEM repeats inverse-sorting of the tour between pick-upped 2 nodes in a clustering neighborhood. If current sequence is as {c, c+1, c+2, c+3, }, the inverted sequence is {c+3, c+2, c+1, c}. When the evaluation is an equivalent or improved, the sequence is accepted.

(step1) One of nodes Pc is chosen from a set P of nodes randomly, and a neighborhood $N(Pc, r)$ is decided.

(step2) Set an index of left side in a neighborhood $N(Pc, r)$ to L, set a right side of a neighborhood $N(Pc, r)$ to R, thus set an index L=c-r, and R=c+r.

(step3) Invert between P_C and P_{dL}.

Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step4) Invert between P_C for P_{dR}.

Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step5) Set dL=dL-1, and set dR=dR+1. Then repeat from (step3) to (step5) until $dL=L$ and $dR=R$.

Smoothing method (SM):

In the local area, SM replaces the nodes from the left side to right side. And repeat exchanging the position of 2 nodes. The starting point is shifted to right sequentially in a neighborhood, so it is calculated in the total hits. When the evaluation is an equivalent or improved, the sequence is accepted. (SM operator for

JSP is different from it for TSP.)

(step1) One of nodes Pc is chosen from a set P of nodes randomly, and a neighborhood $N(Pc, r)$ is decided.

(step2) Set an index of left side in a neighborhood $N(Pc, r)$ to L, set a right side of a neighborhood $N(Pc, r)$ to R, thus set an index L=c-r, and R=c+r. Set CL=L.

(step3) Set dL=CL +1.

(step4) Exchange P_{CL} for P_{dL}.

Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step5) Set dL=dL+1. Then repeat (step4) until dL= R.

(step6) Set CL=CL+1. Repeat from (step3) to (step6) until CL= R-1. When CL= R, the procedure is end.

Symmetrical exchange method (SYM):

SYM exchanges the position of symmetrical 2 nodes of right and left of the center of a neighborhood. In addition, in a left neighborhood, SM exchanges the position of 2 nodes. When the evaluation is an equivalent or improved, the sequence is accepted.

(step1) One of nodes Pc is chosen from a set P of nodes randomly, and a neighborhood $N(Pc, r)$ is decided.

(step2) Set an index of left side in a neighborhood $N(Pc, r)$ to L, set a right side of a neighborhood $N(Pc, r)$ to R, thus set an index L=c-r, and R=c+r. Set CL= c.

(step3) Set dL=CL-1, and dR=CR+1.

(step4) Exchange P_{dL}.for P_{dR}.

Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step5) Exchange P_{CL} for P_{dL}.

Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step6) Set dL=dL-1, dR=dR+1, and CL=CL-1. Then repeat from (step4) to (step6) until dL=L and dR=R. When CL<L and CR>R, the procedure is end.

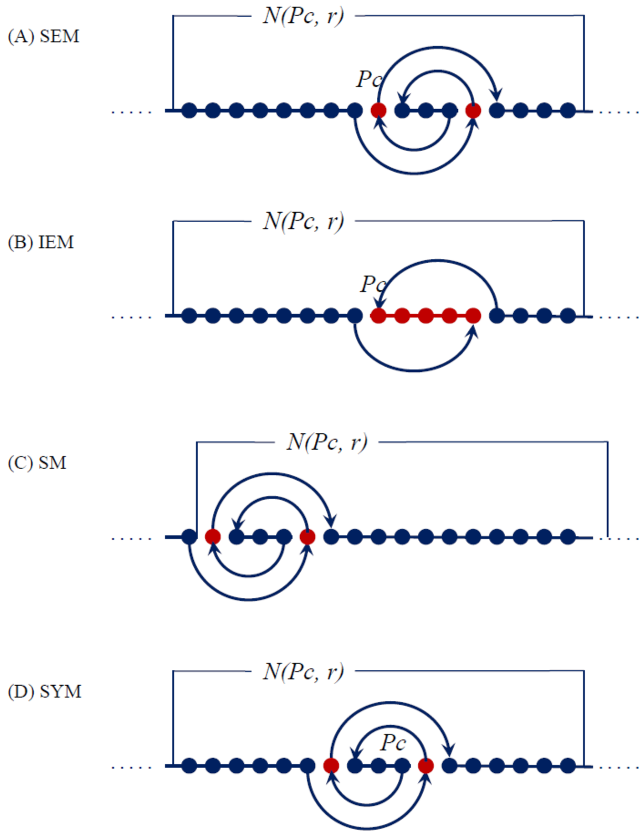


Figure 4. 4 Each local clustering, SEM, IEM, SM, and SYM.

Mixed clustering:

LCO operates these 4 kinds of method randomly. In one clustering of each method, the numbers of scheduling order as follows.

f_{cls} (SEM) = r(r-1)

f_{cls} (IEM) = 2r

f_{cls} (SM) = 2r(2r-1)

f_{cls} (SYM) = 2r

Positioning of SEM, IEM, SM, and SYM by matrix of local clustering:

SEM, SM, SYM: *CO(N , Pc, Pi, Pj)*

IEM: *CO(N, Pc, Bi−, Pj, Lterminal*)

		A_j		
	Pc	*Pj*	*Bj* +	*Bj* −
A_i	*Pi*	SEM, SM, SYM (Swap⊂4-opt)		
	Bi +			
	Bi −	IEM (2-opt)		

Figure 4. 5 Positioning of SEM, IEM, SM, and SYM.

4.5.2. **Move Strategy**

A move strategy is how to accept a solution on. When an evaluation of scheduling by exchanging nodes is equal before exchanging, the solution is accepted, as well as the case that an evaluation is improved. A solution converges with process of a time, and a solution decreases possibility of the movement. And the solution of JSP is not decided uniquely on even an evaluation of the same scheduling. The range of the search-space is brought more globally by acceptance of exchange when an evaluation does not change. About not converging in one pattern, it is explained in structured local clustering of Section 4.7.

4.6. **Extended Local Clustering**

The local clustering of the original LCO operates by exchange order of job numbers in a selected neighborhood. This section examines an operation refers a job number as an indicator. Now, it pays an attention to one job of all jobs. “Insert method” (IM) pulls out the node of one job and inserts it in other place. That is, it coordinates the schedule of one job in the current overall schedule. In addition, the good order of jobs is held excluding its job's position.

Insert method (IM):

(step1) One of nodes *Pc* is chosen from a set *P* of nodes randomly, and a neighborhood *N*(*Pc*, *r*) is decided. One start job *j* and end job *jc* selected.

(step2) Set an index of left side in a neighborhood *N*(*Pc*, *r*) to *L*, thus set *L*=c-*r*. Insert one job *j* as *Pin* from *L*=*c*-*r*. (This repeats until the right side *R*=*c*+*r* later.)

(step3) Search one node having a same job number as *j i*n all nodes. This node is as *Pout*. Search *Pout* from *Pin*+1. Then delete *Pout*. Thus a set *R* has nodes of $j_{\max} \times m_{\max}$. (This repeats until *Pin*-1 around all nodes *P* later.)

(step4) Evaluate a scheduling of before insertion $F(P_{prev})$ and after insertion $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step5) Set *Pout*= *Pout*+1. Then repeat (step3) and (step4) until *Pout*=*Pin*-1 around all nodes.

(step6) Set *Pin*= *Pin*+1. Repeat from (step2) to (step4) until *L*=*R*.

(step7) Set $j = j+1$. Repeat from (step2) and (step4) until $j = jc$.

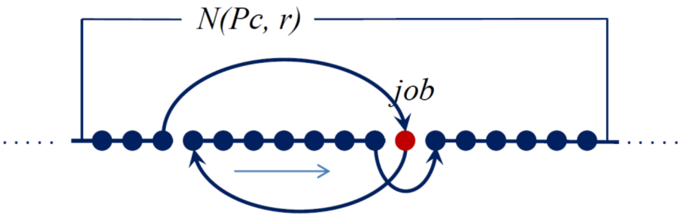
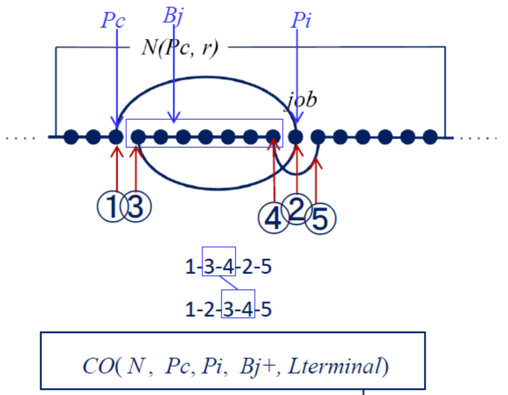


Figure 4. 6 Extended local clustering, IM.

Positioning of IM by matrix of local clustering:

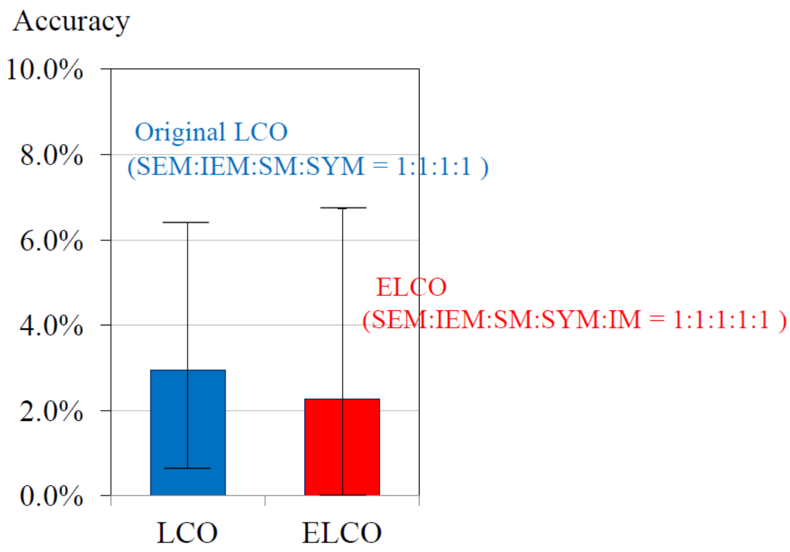
CO(Nfix , Pc, Pi, Bj)



		Aj		
	Pc	*Pj*	*Bj* +	*Bj* −
Ai	*Pi*	SEM, SM, SYM (Swap⊂4-opt)	IM (Insert⊂3-opt)	
	Bi +			
	Bi −	IEM (2-opt)		

Figure 4. 7 Positioning of IM.

Figure 4.6 is the result of bench test applied IM. The instance is “la24” one of “10 though problems”, and it has 150 nodes= 15 jobs×10 machines. As ELCO, the extended local clustering introduced IM was compared with the original local clustering that is mixed clustering of 4 kinds of method (LCO). The experimentation was tried 30 times. The termination set it as a case without 400 times of improvement, as a convergence. The result got the optimal solution with probability of 2%, it was not brought in LCO.



Times were measured on a CPU of Intel Pentium(R) M processor 1.80GHz. (Non-improvement counter for the termination=400, trial=30)

Figure 4. 8 Comparison of 2 methods of LOC by la24 (15Job*10Machine).

4.7. **Structured Local Clustering**

Analysis:

This section explained a mechanism of optimization and a structured local clustering. The objective is to design the operation that is possible to adopt a process to bring the best solution with high probability. The behavior that LCO gets the optimal solution was analyzed using “10 though problems”. Figure 4.7 is 2 cases that reached the optimal solution in la24. Though the solutions have different expressions in these 2 cases, those scheduling results are the same. In other words, the Gantt chart is the same constitution. Thus, the convergence of the solution is not one expression pattern. In addition, LCO exchanges the job on a same machine process, i.e. those are jobs in the process of M8 on Example 1, and M5 on Example 2. The details of analysis are shown as follows.

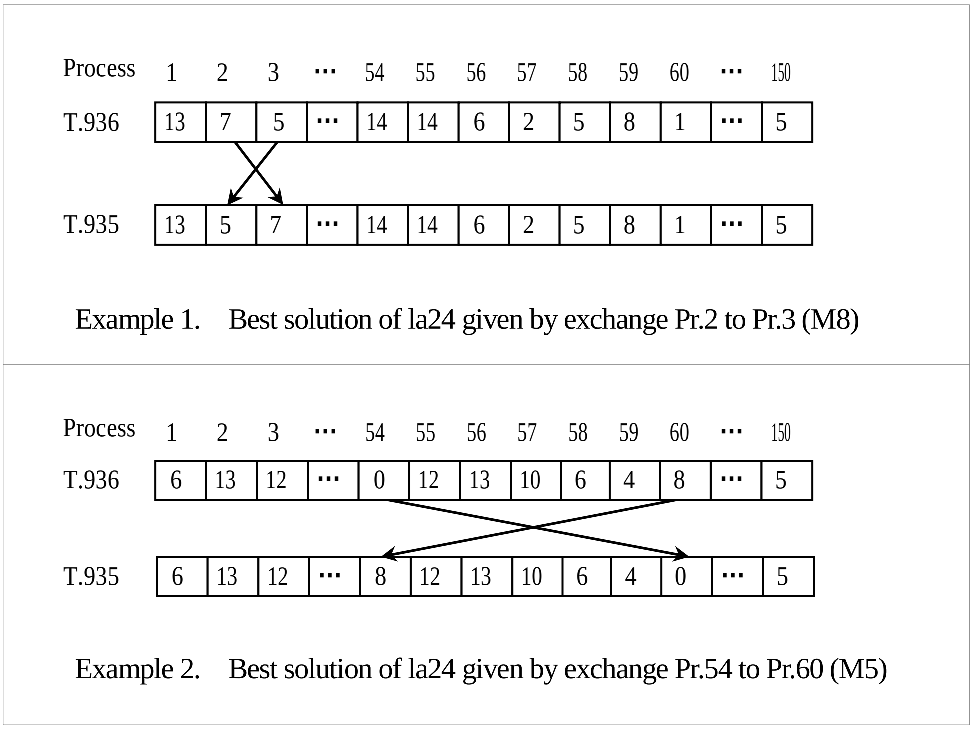


Figure 4. 9 Case of arrival to optimal solution in la24.

(A) The timing when a schedule improves was seen generally when node sequence changes in a same machine process and an overall schedule is improved. About this timing, it corresponded to the cases of 100% of improvements in neighborhood taking the optimal solution.

(B) One schedule has plural pattern of the node sequence. An evaluation of scheduling with a dispatching rule is related to it. When a job is assigned to a machine process from a top of sequence, if the job is set oneness sequence to each machine, a same Gantt chart is constructed.

(C) One solution of convenient sequence is prepared. This solution conform to the sequence that has connected jobs from the first jobs to last jobs assigned to each machine of optimal solution. It reaches to the optimal by several exchanges. Thus, the sequence in consideration of the job process built on each machine may contribute to optimization.

Encoding of solution:

Now, to constitute a set P of solution elements of LCO by every machine is defined as machine division. Then a solution structure of the machine division is introduced as a new encoding of set Q of local clustering. In addition, an evaluation method of scheduling is necessary. A solution conform to the sequence that has connected jobs from a first jobs to last jobs assigned to each machine to apply LCO dispatching rule.

$$Q = \left(\begin{array}{ll} Q_1(j_{max}) & = \{Q_{11}, Q_{12}, \cdots, Q_{1j_{max}}\} \\ Q_2(j_{max}) & = \{Q_{21}, Q_{22}, \cdots, Q_{2j_{max}}\} \\ \vdots & \\ Qm_{max}(j_{max}) & = \{Qm_{max1}, Qm_{max2}, \cdots, Qm_{max}j_{max}\} \end{array} \right) \tag{5}$$

Subject to k : number of process

Qmk : solution node, as job of process= k on machine Mm

(total process=j_{max})

The dispatching rule of new encoding Q of local clustering in the instance of Table 4.1 is as follows.

(step1) A set Q of nodes are given by job number. A job number is generated to each machine.

$$Q1\ (2) = \{1, 2\} \tag{6}$$
$$Q2\ (2) = \{2, 1\}$$
$$Q3\ (2) = \{2, 1\}$$

(step2) Then set Q is constituted to P by pick up from a first jobs of left side to last jobs assigned to each machine in a set. A machine number is assigned by an appearance order of jobs, according to preference relations of machines in a problem definition.

$$P = \{1, 2, 2, 2, 1, 1\} \tag{7}$$

(step3) In making the Gantt-chart of Figure 4.2, the start time of each job considers the previous end time of same job number, and the previous end time of same machine number.

The structured local clustering is applied to new encoding Q. It is iterated local search for the job process on each machine. These are parallel clustering algorithms which are double or complex types based on a new solution structure. In a neighborhood of a job of a same job number or a same process order, parallel search over plural machine can expect an efficient improvement more than operation by the original LCO. The 4 kinds of local clustering are introduced, i.e. dSEM, dSYM, cIM, and dPEM. These operation methods are consisted of SEM, SYM, and IM.

		A_j		
	Pc	*Pj*	*Bj* +	*Bj* −
A_i	*Pi*	SEM, SYM (Swap⊂4-opt)	IM (Insert⊂3-opt)	
	Bi +			
	Bi −			

Figure 4. 10 Positioning of dSEM, dSYM, cIM, and dPEM.

Double - Simple exchange method (dSEM):

"dSEM" operates exchanging 2 nodes like SEM in a job set of one machine. Then, with the set of other machines, it operates SEM in parallel in the neighborhood on the same job. When the evaluation is an equivalent or improved, the sequence is accepted.

(step1) Set *m*=1. Set first line in a set *Q* by machine division to *Qm*.
Set *m'*=2. Set the second line in a set *Q* by machine division to *Qm'*.

(step2) Set one of nodes *Qm·c* is chosen from a set *Qm* of nodes randomly, and a neighborhood *N*(*Qm·c*, *r*) is decided. Set *Qc*'s job number to *j*.

(step3) Set an index of left side in a neighborhood *N*(*Qm·c*, *r*) to *L*, set a right side of a neighborhood *N*(*Qm·c*, *r*) to *R*. thus set an index *L*=c-*r*, and *R*=c+*r*. Set *CL*=*CR*=*c*.

(step4) Search one node having a same job number as *j* from *Qm'*. And a neighborhood *N*(*Qm'·c'*, *r*) is decided.

(step5) Set an index of left side in a neighborhood *N*(*Qm'·c'*, *r*) to *L'*, set a right side to R'. Thus set an index L'=c'-r, and R'=c'+r. Set $CL'=CR'=c'$.

(step6) Set $dL=CL$-1, and $dR=CR$+1.

(step7) Exchange $Qm\cdot_{CL}$ for $Qm\cdot_{dL}$.
Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step8) Set $dL'=CL'$-1, and $dR'=CR'$+1.

(step9) Exchange $Qm'\cdot_{CL'}$ for $Qm'\cdot_{dL'}$.
Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step10) Exchange $Qm'\cdot_{CR'}$ for $Qm'\cdot_{dR'}$.
Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step11) Set $dL'=dL'$-1, and $dR'=dR'$+1. Repeat from (step9) to (step10) until $dL'=L'$ and $dR'=R'$.

(step12) Exchange $Qm\cdot_{CR}$ for $Qm\cdot_{dR}$.
Evaluate a scheduling of before exchange $F(P_{prev})$ and after exchange $F(P_{next})$. Compare the evaluation of $F(P_{prev})$ and $F(P_{next})$. If $F(P_{prev}) \geq F(P_{next})$, accept the new sequence.

(step13) Set $CL'=CR'=c'$. Set $dL'=CL'$-1, and $dR'=CR'$+1. Repeat from (step9) to (step10) until $dL'=L'$ and $dR'=R'$. When $dL'=L'$ and $dR'=R'$, go to (step14).

(step14) Set $CL=CL$-1, and $CR=CR$+1. Repeat from (step6) to (step13) until $CL=L$+1 and $CR=R$-1. When $CL=L$ and $CR=R$, go to (step15).

(step15) Set $m'=m'$+1, and $CL=CR=c$. Repeat from (step4) to (step15) while $m \neq m'$.

(step16) Set $m=m$+1. Repeat from (step6) to (step16) until $m=m_{max}$.

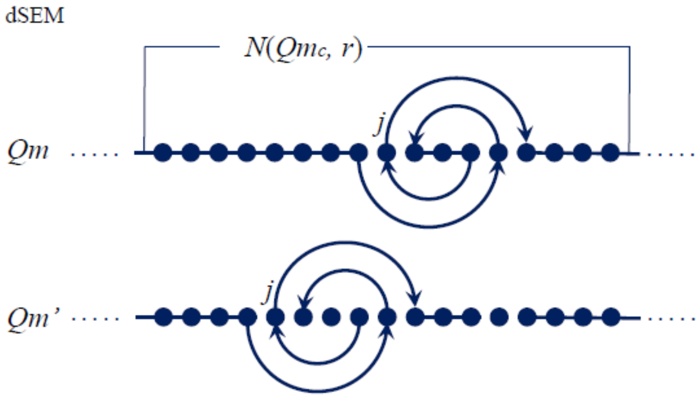


Figure 4. 11 Structured local clustering, dSEM.

Double - Symmetrical exchange method (dSYM):

"dSYM" operates exchanging 2 nodes like SYM in a job set of one machine. Then, with the set of other machines, it operates SYM in parallel in the neighborhood on the same job. To follow the procedure of dSEM, dSYM operates both *Qm* and *Qm'* by SYM. When the evaluation is an equivalent or improved, the sequence is accepted.

Complex –Insert method (cIM):

"cIM" operates the job insertion by IM while to exchange of 2 nodes. The center of neighborhood is determined positioning the node of same job number on each machine. To follow the procedure of dSEM, dIM operates *Qm* that applied SEM, and *Qm'* that applied IM. When the evaluation is an equivalent or improved, the sequence is accepted.

Double - Process exchange method (dPEM):

"dPEM" operates exchanging 2 nodes like dSEM in a job set of one machine with the other machines in parallel. Then, the center of neighborhood is set positioning the node of same process number on each machine. To follow the procedure of dSEM, dPEM operates both *Qm* and *Qm'* by SEM. When the evaluation is an equivalent or improved, the sequence is accepted.

4.8. **Experimentation**

The objective of this experimentation is a performance test of comparing the original LCO with the extended local clustering (ELCO) and the structured local clustering (SELCO). The scale of problem of verification is the problem of "10 Tough Problems" and 15~30 jobs, within 5 minutes as execution time. Also 50~100 jobs and over 500 nodes produced at random, within 10 minutes as execution time. The termination of "Short term" was those times of 5 minutes and 10 minutes are. In addition, "Long term" was set when it is not to be able to improve 2000 steps for comparing the convergence.

The neighborhood was set the upper limit to 1/3~2/3 of total nodes determined randomly. LCO used the mixed clustering of 4 kinds of original methods randomly with the same probability. As an application of extended local clustering, ELCO (extended LCO) used 5 kinds of method added IM in the mixed clustering randomly with the same probability. As an application of structured local clustering, SLCO (structured LCO) used 4 kinds of mixed clustering for a new encoding of solution with the probability of dSEM:dSYM:cIM:dPEM = 2:1:1:1. Table 4.2 shows the best of the evaluation of scheduling and the standard error. The optimal solution is marked by "*". One problem tried 10 times.

Table 4. 2 Comparison of best solution and standard error by LCO, ELCO, and SLCO in 10 Though Problems.

Machine ×Job	Problem	Optimal Value	Short Term						*	Long Term					
			Best Value			Standard Error				Best Value			Standard Error		
			LCO	ELCO	SLCO	LCO	ELCO	SLCO		LCO	ELCO	SLCO	LCO	ELCO	SLCO
5M*15J	la06	926	926	926	926	0	0	0	*	-	-	-	-	-	-
5M*15J	la07	890	890	890	890	0	0	0	*	-	-	-	-	-	-
5M*15J	la08	863	863	863	863	0	0	0	*	-	-	-	-	-	-
5M*20J	ft20	1165	1180	1172	1167	36	47	15		1173	1171	1166	9	9	7
10M*10J	ft10	930	947	965	943	28	15	3		938	937	937	13	3	0
10M*10J	abz5	1234	1250	1253	1234	5	9	0	*	1234	1236	1234	12	10	0
10M*10J	abz6	943	948	943	943	4	27	0	*	948	948	943	4	2	0
10M*15J	la21	1046	1080	1075	1059	20	6	9		1047	1076	1055	31	2	3
10M*15J	la24	935	953	945	939	81	51	20		943	949	945	23	15	6
10M*20J	la26	1218	1219	1242	1220	8	46	21		1218	1218	1218	8	29	0
10M*20J	la27	1235	1265	1308	1269	44	37	38		1264	1261	1238	8	33	31
10M*30J	la31	1784	1785	1784	1784	31	0	0	*	1784	1784	1784	0	0	0
6M*100J	ex6x100a	-	602	602	602	0	0	0	-	602	602	602	0	0	0
8M*100J	ex8x100a	-	4300	4300	4300	0	0	0	-	430	4300	4300	0	0	0
20M*50J	ex20x50a	-	3312	3320	3215	152	98	69	-	323	3188	3049	78	112	45
20M*100J	ex20x100a	-	6308	6253	5949	123	137	112	-	604	6227	5824	43	176	185

Times were measured on a CPU of Intel Pentium(R) M processor 1.80GHz. (Short Term: 5 min/ 10 min, Long Term: Non-improvement counter for the termination=2000, trial=10)

4.9. **Results**

About minimization of the makespan, it is focused to the best solution of short term of Table 4.2. In the problems of 15 jobs×5 machines, the 3 methods reached the optimal solutions with high probability. In larger scale of 15jobs×5 machines, the difference occurred for their best. SLCO led the best most in comparison with LCO and ELCO. About stability, it is focused to the standard error. On the problem of scale larger than 15 jobs 5 machines, the errors of SLCO were smaller in most problems than LCO and ELCO.

Figure 4.9 shows the graph of average and the error in the problem of 50 jobs× 20 machines. In 3 methods, SLCO showed stable performance, and minimized the makespan most. (SLCO→ELCO→LCO: The best was SLCO, and the ELCO followed.) ELCO contributed to the improvement of local minima in LCO by IM. SLCO searched for the job process by the machine division efficiently.

About the convergence of solutions, it compares the best of long term with short term. SLCO improved the solution on the early stage. In addition, the standard error of short term and long term showed SLCO contributed to improvement from the early stage.

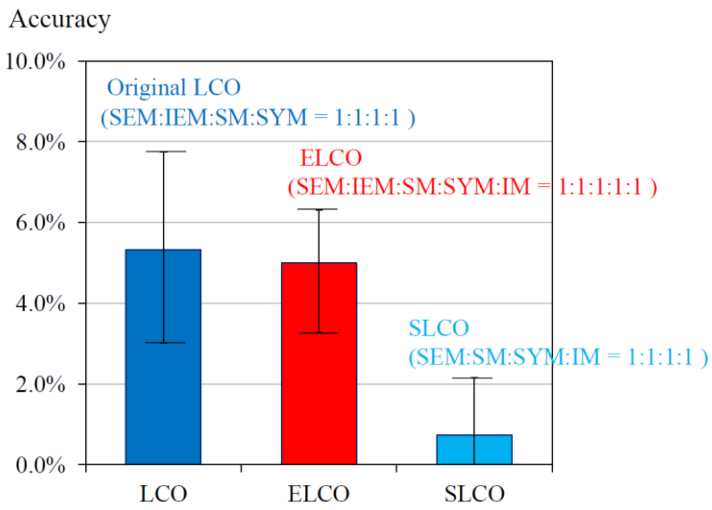


Figure 4. 12 Mean and Range of Error Given by 50Job*20Machine.

Experiment of the case of LCO and ELCO excluded IEM:

It shows about the results of experiments that IEM was excluded. Figure 4.10 is the result of comparison of 5 types of different LCO operation by la24 instance, in the original LCO, expanded LCO, and the structured LCO. The best value showed possibility to improve the local minima by IM, but the quality of solution is uneven by every problems. An error in particular is seen, therefore it is necessary to examine how to combine each local clustering method.

Table 4. 3 Comparison of best solution and standard error by LCO, ELCO, and SLCO in 10 Though Problems (LCO and ELCO excluded IEM).

Machine ×Job	Problem	Optimal Value	Short Term							Long Term					
			Best Value			Standard Error			*	Best Value			Standard Error		
			LCO	ELCO	SLCO	LCO	ELCO	SLCO		LCO	ELCO	SLCO	LCO	ELCO	SLCO
5M*20J	ft20	1165	1178	1192	1167	2	35	15		1178	1175	1166	7	41	7
10M*10J	ft10	930	938	961	943	57	56	3		938	946	937	29	21	0
10M*10J	abz5	1234	1244	1243	1234	26	29	0	*	1242	1242	1234	11	22	0
10M*10J	abz6	943	948	948	943	26	39	0	*	948	948	943	2	5	0
10M*15J	la21	1046	1055	1081	1059	45	37	9		1079	1080	1055	18	13	3
10M*15J	la24	935	963	971	939	12	16	20		952	947	945	35	38	6
10M*20J	la26	1218	1245	1270	1220	40	44	21		1218	1241	1218	13	24	0
10M*20J	la27	1235	1319	1335	1269	39	51	38		1266	1270	1238	25	13	31
10M*30J	la31	1784	1784	1784	1784	0	0	0	*	1784	1784	1784	0	0	0

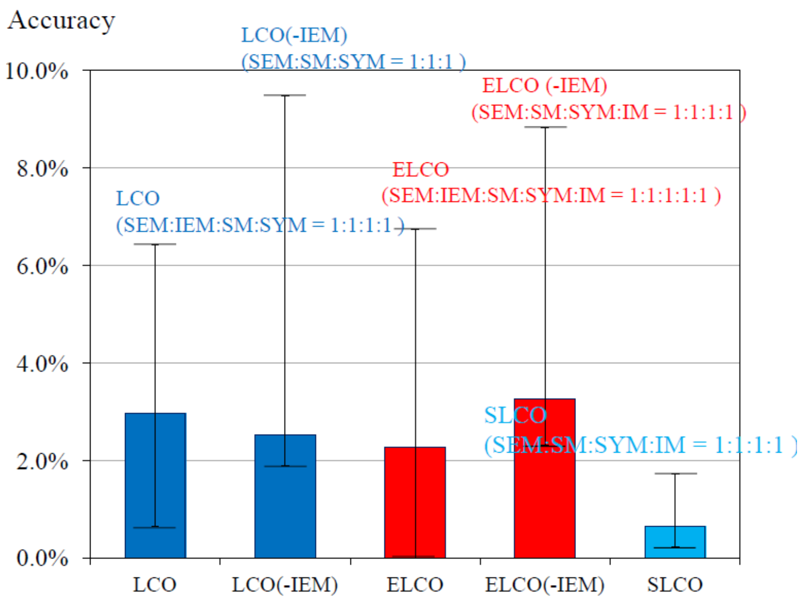


Figure 4. 13 Mean and Range of Error Given by la24 (15Job*10Machine).

5. Conclusion

This chapter elucidated that the analysis of the mechanism related to the improvement to solve JSP, and the design of structured local clustering for LCO.

(A) The LCO algorithm to solve JSP was introduced. The encoding expresses by a set of multiplexed job numbers meaning different machine number. The operator of improvement schedule has 4 kinds of local clustering operations based on exchanging elements randomly in selected neighborhood. Those are SEM (Simple exchange method), IEM (Inverse exchange method), SM (Smoothing method) and SYM (Symmetrical exchange method).

(B) IM (Insert method) that is moving job by insertion was applied to LCO as the extended local clustering operation. The schedule in JSP is evaluated by the job process on machines based on Gantt chart. The original operator was not considered this mechanism.

(C) ELCO that operates 5 kinds of local clustering (original methods and IM) showed the possibility to get optimal, avoiding local minima of LCO.

(D) The encoding was extended the structure of machine division. For new structure, the operator was examined 4 kinds of operation method that improve job process in parallel on each machine as structured local clustering. Those are dSEM (Double-SEM), dSYM, cIM (Complex-IM) and dPEM (Double Process exchange method). The structured local clustering was able to search job process effectively.

(E) The experimentation was compared in 3 methods, LCO, ELCO and SLCO that operates structured local clustering by “10 Though Problems” and the scale of JSP is the problem has 15 ~ 30 jobs, also the problems having 50 ~ 100 jobs. The results of performance were that SELCO was best in the reach the optimal solution, the improvement speed, the standard error.

Chapter 5

6. **Summary**

In this dissertation, a design methodology for LCO developed based on SOM was constructed to make a performance enhancement of the local clustering which was LCO search operator as one of heuristic included in approximate algorithms. The design method was considered about an diversification and intensification of search as meta-heuristic' strategies. The discussion was divided into two phases. (1) Extended local clustering was a case of extension of variation of the operation. (2) Structured local clustering was a case of intensive search for a search neighborhood of a solution by the extension of an operation. Each mission was approached as follows. (1) For an expansion local site clustering, it approached by introduction of an operation based on insertion method. (2) For a structured local clustering, it approached by extracting a structural partial solution element depending on a characteristic of the mechanism of improvement. In (2), as a frame catching solution structurally, it considered solution expression to be comprised of the subset (node-set) of the solution element (simple node). They were applied to TSP and JSP, and the effectiveness was experimented by using their benchmark problems. Those results were explained in Chapter 3 and Chapter 4.

Chapter 1 described the basic algorithm of LCO, discussed a positioning of LCO in an approximate algorithm, and the issues in the conventional study and a related study. Chapter 2, suggested a framework for a design of LCO operation for diversification and intensification of search. This is a design method to take in structural operation for LCO with meta-heuristics strategy necessary for the search performance advancement and the concept of structural operation. Chapter 3 showed the suggestion of structured local clustering in TSP and the results of inspection experiments. In structured local clustering operator, LCO extracted the partial tour that is immobilized locally on a search process. Then the solution was considered that was divided to node-sets and bridges. Then the insertion method was introduced for the recombination. Chapter 4 showed the suggestion of structured local cluster ring in JSP and the results of inspection experiment. In structured local clustering, the solution structure was extended in a phenotype of the resource division, the search was made intensive to each job sequence on each resource.

The technique that was adopted for the expansion is slightly, also, demonstrated methods are some methods with LCO. However, their behavior and strategy are related to a lot technique of local search algorithms and population based search algorithms in approximate algorithms. Finally what this dissertation hope to insist on is that the definition and positioning its operation of the method for the study of approximate algorithm are important. The structured concept should be discussed for meta-heuristics extensions.

Acknowledgments

The author expresses to thank Professor Keiji Suzuki of Hokkaido University for instruction and guidance from the inception to the completion of the present research. Deep thanks are also due to Associate Professor Hidenori Kawamura of Hokkaido University. He wishes to thank Professor Masahito Yamamoto, Professor Masahito Kurihara, and Professor Tetsuo Ono of Hokkaido University for profitable discussions and suggestions. He wishes to extend deep appreciation to Professor Masashi Furukawa of Hokkaido Information University and Sadayoshi Mikami of Future University Hakodate for valuable discussions and suggestions. Many pieces of affectionate advice and discussion given by them were very useful to improve this research. He would like to thank President Hideyuki Nakashima and all stuff of Future University Hakodate, and all students of the office, Harmonious Systems Engineering. Finally, he appreciates all that his family had done for supporting his life.

Bibliography

[Frukawa 05-1] M. Furukawa, M. Watanabe, and Y. Matsumura,
"Local Clustering Organization (LCO) Solving a Large-Scale TSP", Journal of Robotics and Mechatronics, Vol.17, No.5, pp.560-567, 2005.

[Frukawa 05-2] M. Furukawa, M. Watanabe, and Y. Matsumura,
"Development of Local Clustering Organization (LCO) for Solving TSP", Transactions of the Japan Society of Mechanical Engineers. C, Vol.71, No.117, pp.83-89, 2005. (in Japanese)

[Frukawa 06] M. Furukawa, M. Watanabe, and Y. Matsumura,
"Development of Local Clustering Organization Applied to Job-shop Scheduling Problem", Journal of the Japan Society for Precision Engineering, Contributed Papers, Vol.72, No.7, pp. 867-872, 2006. (in Japanese)

[Konno 09] Y. Konno, K. Suzuki,
"Performance of Extended Local Clustering Organization for Large Scale Job-Shop Scheduling Problem", IEEJ Transactions on Electronics, Information and Systems, Vol.129, No.7, pp.1363-1370, 2009. (in Japanese)

[Iwasaki 13] Y. Iwasaki, I. Suzuki, M. Yamamoto, and M. Furukawa,
"Job-shop Scheduling Approach to Order-picking Problem" Transactions of the Institute of Systems, Control and Information Engineers Vol.26, No.3, pp. 103-109, 2013.

[Kohonen 89] T. Kohonen,
"Self-organization and associative memory", (3rd edition) Springer-Verlag, Berlin, pp.68-209, 1989.

[Kohonen 01] T. Kohonen,
" Self-Organization Maps ", (3rd edition) Springer-Verlag, Berlin, pp.71-176, 2001.

[Kohonen 07] T. Kohonen,
"Self-Organization Maps", (3rd edition) Springer Japan, Berlin, pp.74-182, 2007.

[TSPLIB] TSPLIB,
http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95.

[NTSP] National Traveling Salesman Problems,
http://www.math.uwaterloo.ca/tsp/world/countries.html.

[Lin 73] S. Lin, and B. W. Kernighan,
"An effective heuristic algorithm for the traveling-salesman problem", Operations research 21.2, pp.498-516, 1973.

[Helsgaun 00] K. Helsgaun,
"An effective implementation of the Lin–Kernighan traveling salesman heuristic", Eur. J. Oper. Res., Vol.126, No.1, pp.106–130, 2000.

[Helsgaun 09] K. Helsgaun,
"General λ-opt submoves for the Lin–Kernighan TSP heuristic", Mathematical Programming Computation, Vol.1, Issue 2-3, pp. 119-163, 2009.

[Yoshihara 09] I. Yoshihara, A. Oishi, M. Kuroda, K. Yamamori, M. Aikawa, "A Rapid Solution of Lin-Kernighan Algorithm for TSP", Memoirs of the Faculty of Engineering, Miyazaki University 38, pp.277-282, 2009. (in Japanese)

[Kubo 09] M. Kubo, J. P. Pedoroso,
"Metaheuristics: a programming gide", Kyoritsu Shuppan CO. LTD., pp.1-67, 2009. (in Japanese)

[Nonobe 08] K. Nonobe, M. Yagiura,
"Local Search and its Extensions-Focusing on Tabu Search", Journal of the Society of Instrument and Control Engineers Vol.47, No.6, pp.493-499, 2008. (in Japanese)

[Nagata 07] Y. Nagata,
"Fast Implementation of Genetic Algorithm by Localized EAX Crossover for the Traveling Salesman Problem", Transactions of the Japanese Society for Artificial Intelligence, Vol.22, No.5, pp.542-552, 2007. (in Japanese)

[Nguyen 07] H. D. Nguyen, I. Yoshihara, K. Yamamori, and M. Yasunaga,
"Implementation of an Effective Hybrid GA for Large-Scale Traveling Salesman Problems", IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, Vol.37, No.1, pp.92-99, 2007.

[Nguyen 02] H. D. Nguyen, I. Yoshihara, K. Yamamori, K. Yamamori, M. Yasunaga, "Greedy Genetic Algorithms for Symmetric and Asymmetric TSPs", IPSJ Trans. Mathematical Modeling and Its Applications 43, SIG_10(TOM_7)), pp.165-175, 2002.

[Schleuter 97] M. G. Schleuter,
"Asparagos96 and the traveling salesman problem", Evolutionary Computation 1997, IEEE International Conference, pp.171-174, 1997.

[Karp 72] R. M. Karp,
Reducibility among combinatorial problems.In Complexity of Computer Computations R E.Miller and J.W. Thatcher,eds.Advances in Computer Research, Plenum Press,New York,pp.85-103, 1972.

[Karp 77] R. M. Karp,
"Probabilistic Analysis of Partitioning Algorithms for the Traveling-Salesman Problem in the Plane", Mathematics of Operations Research, Vol.2, pp.209-224, 1977.

[Arora98] S. Arora.,
"Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems", Journal of the ACM 45,p.753-782, 1998.

[Yamamura 9] M. Yamamura, T. Ono, S. Kobayashi,
"Character-Preserving Genetic Algorithms for Traveling Salesman Problem", Journal of Japanese Society for Artificial Intelligence, Vol.7, No.6, pp.1049-1059 , 1992. (in Japanese)

[Boussaïd 13] Boussaïd, I., Lepagnot, J., & Siarry, P,
"A survey on optimization metaheuristics", Information Sciences, Vol.237, pp. 82-117, 2013.

[Ou 08] G. Ou, H. Tamura, K. Tanno, Z. Tang,
"A Method of Solving Scheduling Problems Using Improved Guided Genetic Algorithm", IEEJ Transactions on Electronics, Information and Systems, Vol.128, No.8, pp. 1351-1357, 2008. (in Japanese)

[Sait 01] S. M. Sait, H. Youssef,
"Iterative Computer Algorithm with Application in Engineering", Maruzen Co., Ltd., 2001. (in Japanese)

[Kuroda 02] M. Kuroda, K. Matsumura, "Production Scheduling", Asakura Publishing Co., Ltd., pp.1-99, 2002. (in Japanese)

[JSP 00] B. Massey, "CS410/510SS Project JobShop Scheduling", http://web.cecs.pdx.edu/~bart/cs510ss/project/jobshop/, 2000.

[JSP 98] K. Morikawa, "HUGE (Interactive Scheduling Support System)", http://home.hiroshima-u.ac.jp/mkatsumi/open/HUGE/data/, 1998.

[Hartmann 05] A.K. Hartman, M. Weigt,
"Phase Transitions in Combinatorial Optimization Problems", WILEY-VCH Verlag CmbH&Co., 2005.

[Pardalos 10] P. M. Pardalos, J. D. Pinter, S. M. Robinson, T. Terlaky, M. T. Thai, “Data Correcting Approach in Combinatorial Optimization”, Springer New York Heidelberg Dordrecht London, 2010.

[Mladenovic 97] N. Mladenovic, P. Hansen,
“Variable neighborhood search”, Computers and Operations Research 24, pp.1097–1100, 1997.