2012

# New Tabu Search Heuristics for the Dynamic Facility Layout Problem

Alan McKendall

Wen-Hsing Liu


International Journal of Production Research
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tprs20

# New Tabu search heuristics for the dynamic facility layout problem

Alan R. McKendall Jr [a] & Wen-Hsing Liu [b]

[a] Department of Industrial & Management Systems Engineering, West Virginia University, 325A Mineral Resources Building, Morgantown, WV 26506, USA

[b] Department of Industrial Engineering, Texas Tech University, 201 IE Building, Lubbock, TX 79409, USA

Version of record first published: 14 Jun 2011.

To cite this article: Alan R. McKendall Jr & Wen-Hsing Liu (2012): New Tabu search heuristics for the dynamic facility layout problem, International Journal of Production Research, 50:3, 867-878

To link to this article: http://dx.doi.org/10.1080/00207543.2010.545446

*International Journal of Production Research*
Vol. 50, No. 3, 1 February 2012, 867–878

Taylor & Francis
Taylor & Francis Group

# New Tabu search heuristics for the dynamic facility layout problem

Alan R. McKendall Jr[a]* and Wen-Hsing Liu[b]

[a]*Department of Industrial & Management Systems Engineering, West Virginia University, 325A Mineral Resources Building, Morgantown, WV 26506, USA;* [b]*Department of Industrial Engineering, Texas Tech University, 201 IE Building, Lubbock, TX 79409, USA*

(*Received 24 June 2010; final version received 15 November 2010*)

A manufacturing facility is a dynamic system that constantly evolves due to changes such as changes in product demands, product designs, or replacement of production equipment. As a result, the dynamic facility layout problem (DFLP) considers these changes and is defined as the problem of assigning departments to locations during a multi-period planning horizon such that the sum of the material handling and re-arrangement costs is minimised. In this paper, three tabu search (TS) heuristics are presented for this problem. The first heuristic is a simple TS heuristic. The second heuristic adds diversification and intensification strategies to the first, and the third heuristic is a probabilistic TS heuristic. To test the performances of the heuristics, two sets of test problems from the literature are used in the analysis. The results show that the second heuristic out-performs the other proposed heuristics and the heuristics available in the literature.

**Keywords:** dynamic facility layout problem; tabu search; probabilistic tabu search; diversification and intensification strategies; meta-heuristics



## 1. Introduction

The facility layout problem (FLP) is to find the most efficient arrangement of departments within a facility (e.g. manufacturing plants, administrative office buildings, and service facilities). For manufacturing facilities, the most commonly used criterion to determine the efficiency of layouts is the minimisation of material handling cost. It has been estimated that materials handling cost is between 20 to 50% of the total operating cost and effective facility layout planning can reduce material handling costs by 10 to 30% (Tompkins *et al*. 2003). Material handling cost between each pair of departments is defined as the product of the flow of materials, distance, and transportation cost per unit per distance unit between each department pairs. The sum of these products for each pair of departments is defined as the material handling cost of the layout. For a review of the FLP, see Kusiak and Heragu (1987) and Meller and Gau (1996).

A manufacturing facility is a dynamic system that continuously evolves. In order to ensure optimal performance of a facility, the layout should evolve based on changes to the system that may occur over time. Some of the factors, presented by Francis *et al*. (1992), which may cause the modification of the layout of a facility, are as follows:

- Changes in the product design.
- The addition or deletion of a product from the product line.
- Significant increase or decrease in the demand of a product.
- Changes in the process design.
- The replacement of equipment.
- The adoption of new safety standards.
- Bottlenecks in production.
- Unexplainable delays and idle time.
- Excessive temporary storage space.

It is important to note that changing the layout of a facility may be very costly due to the cost of re-arranging the departments, cost of purchasing or leasing equipment for re-arranging the departments, and the cost associated with the loss of production. Therefore, if the total cost of re-arranging the departments (i.e. total re-arrangement cost) is

*Corresponding author. Email: Alan.McKendall@mail.wvu.edu

ISSN 0020–7543 print/ISSN 1366–588X online

http://dx.doi.org/10.1080/00207543.2010.545446
http://www.tandfonline.com

relatively high with respect to total material handling cost, it may be more advantageous not to re-layout the facility or to make small modifications to the layout in order to keep total re-arrangement cost low. In contrast, if total re-arrangement cost is relatively low with respect to total material handling cost, re-layout the facility such that total material handling cost is minimised. Otherwise, re-layout the facility such that the sum of the material handling and re-arrangement costs is minimised. As a result, the dynamic facility layout problem (DFLP) considers these issues and is defined as the problem of assigning departments to locations during a multi-period planning horizon such that the sum of the material handling and re-arrangement costs is minimised. In other words, the DFLP arranges and re-arranges (when there are changes in the amounts of materials flowing between departments) manufacturing facilities such that the sum of the material handling and re-arrangement cost is minimised.

The assumptions for the DFLP presented in this paper are defined as follows.

(1) The distances between locations on the plant floor are known.
(2) There are $T$ periods in the planning horizon where $T > 1$.
(3) For each period, each department requires exactly one location on the plant floor, and only one department can be assigned to a location.
(4) For each period, the material flow between each pair of departments is known.
(5) The objective is to obtain the layout plan (i.e. layout for all periods) which minimises the sum of the material handling and re-arrangement costs.

If $T = 1$ (i.e. the number of periods in the planning horizon is one) in assumption (2), then the problem is defined as a quadratic assignment problem (QAP). For a mathematical formulation of the DFLP, see McKendall *et al.* (2006). Since the DFLP is a generalisation of the QAP and is computationally intractable, three efficient heuristics are presented in this paper for the DFLP. The first heuristic is a simple tabu search (TS) heuristic. The second heuristic adds diversification and intensification strategies to the first, and the third heuristic is a probabilistic TS heuristic. In the next section, the DFLP literature is reviewed. Then the TS heuristics are presented. Following, the computational results are given on the performances of the proposed heuristics with respect to solution quality and computation time on two sets of test problems taken from the literature. Lastly, the paper is concluded and future research directions are discussed.



## 2. Literature review

Rosenblatt (1986) was the first to present solution techniques (i.e. optimal and heuristic procedures based on dynamic programming) for the DFLP. Lacksonen and Enscore (1993) modified five solution techniques for solving the DFLP. One of the techniques is a modification of the dynamic programming (DP) technique presented by Rosenblatt (1986). Four of them are modifications of solution techniques presented for the QAP: a branch and bound algorithm, computerised relative allocation of facilities technique (CRAFT), cutting planes, and cut trees. Also, Lacksonen and Enscore (1993) generated 32 test problems (contain problems with 6, 12, 20, and 30 departments each with 3 and 5 time periods) to test the performances of the five proposed heuristics. The results show that the modified cutting planes algorithm out-performed the four other heuristics. Urban (1993) also presented a steepest descent pairwise interchange procedure, similar to CRAFT, for the DFLP, but the authors considered forecast windows. Balakrishnan *et al.* (2000) presented two heuristics which improved Urban's heuristic. The first heuristic combines Urban's heuristic with a backward-pass pairwise interchange heuristic, and the second heuristic combines Urban's heuristic with DP. The authors used 48 test problems (contain problems with 6, 15, and 30 departments each with 5 and 10 time periods), presented by Balakrishnan and Cheng (2000), to test the performances of their two heuristics against Urban's heuristic, a modification of Urban's heuristic, and Rosenblatt's two methods. Their two heuristics clearly outperformed the other heuristics.

The more recent solution techniques available in the literature for the DFLP are meta-heuristics and hybrid heuristics (see Table 1). The meta-heuristics presented for the DFLP are: genetic algorithms (GA) as in Conway and Venkataramanan (1994) and Balakrishnan and Cheng (2000); a tabu search (TS) as in Kaku and Mazzola (1997); and simulated annealing (SA) as in Baykasoglu and Gindy (2001) and McKendall *et al.* (2006). The hybrid heuristics presented for the DFLP are: hybrid DP (HDP) as in Erel *et al.* (2003) and Balakrishnan *et al.* (2000) as mentioned above; a hybrid GA (HGA) as in Balakrishnan *et al.* (2003); a heuristic which combines GA, TS, and parallel processing (GATS) as in Rodriguez *et al.* (2006); hybrid ant systems (HAS) as in Baykasoglu *et al.* (2006) and McKendall and Shang (2006); and a SA heuristic with a tabu list (TABUSA) as in Sahin and Turkbey (2009).

Table 1. Meta-heuristics and hybrid heuristics for the DFLP.

| | Meta-heuristics | | | Hybrid heuristics | | | | |
|---|---|---|---|---|---|---|---|---|
| Authors (year) | GA | TS | SA | HDP | HGA | GATS | HAS | TABUSA |
| Conway and Venkataramanan (1994) | √ | | | | | | | |
| Balakrishnan and Cheng (2000) | √ | | | | | | | |
| Kaku and Mazzola (1997) | | √ | | | | | | |
| Baykasoglu and Gindy (2001) | | | √ | | | | | |
| McKendall *et al*. (2006) | | | √ | | | | | |
| Balakrishnan *et al*. (2000) | | | | √ | | | | |
| Erel *et al*. (2003) | | | | √ | | | | |
| Balakrishnan *et al*. (2003) | | | | | √ | | | |
| Rodriguez *et al*. (2006) | | | | | | √ | | |
| Baykasoglu *et al*. (2006) | | | | | | | √ | |
| McKendall and Shang (2006) | | | | | | | √ | |
| Sahin and Turkbey (2009) | | | | | | | | √ |

For a review of the problem assumptions and solution techniques of the DFLP, see Balakrishnan and Cheng (1998) and Kulturel-Konak (2007). Next, three TS heuristics are presented for the DFLP.

## 3. TS heuristics for the DFLP

Glover (1986) was the first to introduce the TS heuristic. Since then, TS has been used to solve many combinatorial optimisation problems related to the DFLP. For instance, Skorin-Kapov (1990) was the first to use TS to solve the QAP. Also, Taillard (1991), Skorin-Kapov (1994), Chiang and Kouvelis (1996), and Chiang and Chiang (1998) presented TS heuristics for the QAP, to mention a few. However, Kaku and Mazzola (1997) presented the only TS heuristic for the DFLP, known to the authors. The differences between their heuristic and the proposed heuristics are discussed later. Since TS heuristics with different variations performed well on related problems, three different TS heuristics are presented for the DFLP. First, a simple basic TS heuristic, called TS1, is developed for the DFLP. The second heuristic, called TS2, adds diversification and intensification strategies to TS1 so that it can obtain high quality solutions even if low quality initial solutions are available. In other words, it uses dynamic tabu tenure length, frequency-based memory, and an intensification strategy similar to the one presented by Chiang and Kouvelis (1996) for the QAP. The third heuristic, called TS3, is a modification of TS1. Instead of accepting the best admissible move while searching neighbourhoods of solutions, it randomly selects a move from the top $M$ admissible moves. This is known as the probabilistic TS heuristic. A similar probabilistic TS heuristic was presented by Chiang and Chiang (1998) for the QAP. Next, the proposed TS heuristics for the DFLP are presented, but first the solution representation, neighbourhood structure, and local search technique are presented for the DFLP.

A feasible solution of the DFLP is represented as follows:

$$\pi = (\pi^1, \pi^2, \ldots, \pi^T)$$

where $\pi$ is the solution (i.e. layout plan) of the DFLP, $\pi^t = \{\pi^t(1), \pi^t(2), \ldots, \pi^t(N)\}$ is the layout in period $t$, $\pi^t(i)$ = location of department $i$ in period $t$, $T$ = number of periods, and $N$ = number of departments. The total material handling cost of $\pi$ is defined as

$$MH(\pi) = \sum_{t=1}^{T} \sum_{i=1}^{N} \sum_{k=1}^{N} c_{tik} f_{tik} d(\pi^t(i), \pi^t(k))$$

where $c_{tik}$ = cost of moving one unit of material per distance unit from department $i$ to department $k$ in period $t$, $f_{tik}$ = units of material flowing from department $i$ to department $k$ in period $t$, and $d(j, l)$ = distance from department $i$ at location $j$ to department $k$ at location $l$ (i.e. $j = \pi^t(i)$ and $l = \pi^t(k)$). Oftentimes it may be difficult to obtain $c_{tik}$; if this is the case, it is assumed that $c_{tik} = 1$ for all $t$, $i$, $k$. Re-arrangement cost occurs when the location of a department in one period is different from that in the preceding or succeeding periods. Thus, the total re-arrangement cost of $\pi$ is defined as

$$R(\pi) = \sum_{t=2}^{T} \sum_{i \in I_t} A_{ti}$$

where $I_t$ = set of all departments in period $t$ with different locations in period $t-1$ and $A_{ti}$ = cost of re-arranging department $i$ at the beginning of period $t$. Therefore, the total cost of solution (layout plan) $\pi$ is defined as

$$TC(\pi) = MH(\pi) + R(\pi) = \sum_{t=1}^{T} \sum_{i=1}^{N} \sum_{k=1}^{N} c_{tik} f_{tik} d(\pi^t(i), \pi^t(k)) + \sum_{t=2}^{T} \sum_{i \in I_t} A_{ti}.$$

Once a feasible solution $\pi$ and its total cost, $TC(\pi)$, are obtained, the neighbourhood of solution $\pi$, $N(\pi)$, can be explored so that an improved solution may be found. The neighbourhood $N(\pi)$ consists of all solutions obtained from all possible pairwise interchanges between locations of departments in each period. Therefore, there are $T^*N(N-1)/2$ solutions in $N(\pi)$. Each possible solution in $N(\pi)$ is obtained from each possible move which may be defined as move ($t$, $u$, $v$) where $u$ and $v$ ($u < v$) are the departments which exchange locations in period $t$. For example, consider $\pi = (\{1, 3, 2, 4\}, \{1, 3, 2, 4\}, \{1, 3, 2, 4\})$ where departments 1, 2, 3, and 4 are assigned to locations 1, 3, 2, and 4, respectively, in periods 1, 2, and 3. As a result, there are $3(4)(3)/2 = 18$ solutions in $N(\pi)$. One of the solutions in $N(\pi)$ can be obtained by interchanging the locations of departments 3 and 4 in period 1 (i.e. by performing move $(1, 3, 4)$). As a result, the solution $(\{1, 3, 4, 2\}, \{1, 3, 2, 4\}, \{1, 3, 2, 4\})$ is obtained. When obtaining each solution in $N(\pi)$, its corresponding cost is obtained, and the best solution in $N(\pi)$ is defined as $\pi'$. Then set $\pi = \pi'$, and the process is repeated until a non-improving solution $\pi'$ is obtained (i.e. $TC(\pi') \geq TC(\pi)$). It is obvious that this steepest descent local search technique may often terminate at a poor local optimum, depending on the initial starting solution $\pi$. As a result, a basic TS heuristic is given below which allows iterating through non-improving solutions in search of the global optimum, but first the main components of the TS heuristic are given.

One of the major differences between the steepest descent local search technique discussed above and the TS heuristic (as well as other meta-heuristics) is that the TS heuristic iterates through non-improving solutions. That is, it does not terminate once it converges to a local optimum. More specifically, when exploring $N(\pi)$, even after converging to a local optimum using the steepest descent heuristic, short-term memory is used to forbid the recent moves so that the heuristic can climb out of the valley which contains the local optimum in search of better local optima. A move is defined as a recent move (tabu) if it has been performed within $TL$ iterations ($TL$ = tabu tenure length), and the status of the recent moves are maintained in the tabu list $tabu[t][i][k]$, where $i < k$. If a move $(t, u, v)$ has been performed at the current iteration $iter$ then the move is tabu for $TL$ iterations such that $tabu[t][u][v] = iter + TL$ where $u < v$. That is, move $(t, u, v)$ is defined as recent (or tabu) from iteration $iter$ to iteration $iter + TL$. However, an aspiration criterion may be used to override the tabu restriction. In this paper, if the best move (move* $(t, u, v)$) is recent (i.e. tabu restricted), but its corresponding solution $\pi'$ is better than the best solution $\pi^{best}$ found thus far (i.e. $TC(\pi') < TC(\pi^{best})$), then the tabu restriction is over-ridden and the move is performed. In other words, move* $(t, u, v)$ is defined as the best admissible move which is either the best non-tabu move or tabu move overridden by the aspiration criterion. A simple basic TS heuristic for the DFLP, called TS1, is outlined below.

**Step 1:** Initialise parameters and counters: initialise tabu list $tabu[t][i][k]$ for $i < k$ and current time $curr\_time$. Set tabu tenure length $TL$, iteration counter $iter = 0$, and total run time before terminating heuristic $TRT$.

**Step 2:** Obtain an initial solution $\pi$ by assigning department 1 to location 1, department 2 to location 2, and so on. This layout is used for all periods. Then, determine total cost $TC(\pi)$. Set $\pi^{best} = \pi$.

**Step 3:** Set $iter = iter + 1$, and find the best admissible move move* $= (t, u, v)$, which gives $\pi'$.

**Step 4:** Set $\pi = \pi'$. If $TC(\pi) < TC(\pi^{best})$, then set $\pi^{best} = \pi$.

**Step 5:** Update the tabu list as $tabu[t][u][v] = iter + TL$ where $u < v$.

**Step 6:** If $curr\_time < TRT$, go to Step 3. Else, terminate the heuristic and return the solution $\pi^{best}$ and the total cost $TC(\pi^{best})$.

In the TS2 heuristic, the tabu tenure length $TL$ is dynamic in order to diversify the search (i.e. force the search into unexplored regions of the solution space). In other words, $TL$ at each iteration *iter* vary between a lower bound ($LB$) and an upper bound ($UB$). That is, $LB \le TL \le UB$. The actual value of $TL$ may vary from one iteration to the next depending on the percent reduction of $TC(\pi')$ from $TC(\pi)$. The percent reduction of $TC(\pi')$ from $TC(\pi)$ is defined as follows.

$$PR(\pi) = 100\%[TC(\pi) - TC(\pi')]/TC(\pi)$$

At each iteration *iter*, use the following rules for setting $TL_{iter}$, which were adapted from Chiang and Kouvelis (1996) for the QAP and modified for the DFLP.

(1) If $PR(\pi) < 0\%$, then set $TL_{iter} = TL_{iter-1}$.
(2) If $0\% \le PR(\pi) < \alpha\%$, then set $TL_{iter} = LB + (UB - LB)\ PR(\pi)/\alpha\%$.
(3) If $\alpha\% \le PR(\pi) < \beta\%$, then set $TL_{iter} = UB$.
(4) If $PR(\pi) \ge \beta\%$, then set $TL_{iter} = LV$, for $LV$ = large value. In this paper, $LV = 2NT$.

Besides using dynamic tabu tenure length as a diversification strategy (i.e. allowing $TL$ to increase) the TS2 heuristic uses long-term memory (i.e. frequency-based memory) as in Skorin-Kapov (1990) and Chiang and Kouvelis (1996) for the QAP. More specifically, the tabu list $tabu[t][i][k]$, for $i > k$, is used to keep track of the frequency of the moves move $(t, i, k)$, and the penalty function defined below is used to penalise non-improving moves. If the best admissible move, move* $= (t, u, v)$ where $u < v$, is a non-improving move, then the penalty value for this move is defined as the product of the penalty parameter $= a$ and the frequency of move move* (i.e. $a*tabu[t][v][u]$). Therefore, the penalty value for either an improving move (i.e. $TC(\pi') < TC(\pi)$) or non-improving move $(t, u, v)$ is given as follows.

$$Penalty(\text{move}(t,u,v)) = \begin{cases} 0 & \text{if } TC(\pi') < TC(\pi) \\ a * tabu[t][v][u] & \text{otherwise} \end{cases}$$

Therefore, when the best admissible move move* $= (t, u, v)$ is a non-improving move, the total cost of the corresponding solution $\pi'$ (as well as all other solutions in $N(\pi)$) is calculated as follows.

$$TCp(\pi') = TC(\pi') + Penalty(\text{move}(t,u,v))$$

Unlike the diversification strategies presented above, the intensification strategy explores promising areas of the solution space more thoroughly. In this paper, besides using dynamic tabu tenure length as an intensification strategy (i.e. allowing $TL$ to decrease), the intensification method described in Chiang and Kouvelis (1996) is modified for the DFLP. The intensification strategy is implemented by fixing pairs of departments after exchanging their locations (i.e. performing move*$(t, u, v)$), if this exchange reduces $TC(\pi^{best})$ by at least $\gamma\%$ (i.e. $PR(\pi^{best}) \ge \gamma\%$). To fix the pair of departments $u$ and $v$ in period $t$, the locations of the departments are not allowed to change in period $t$ until they are freed. The fixed departments can be freed only when the exchange of the locations of the fixed departments with other departments produces a solution $\pi'$ such that $TC(\pi') < TC(\pi^{best})$. This is different from the way Chiang and Kouvelis (1996) freed fixed departments. In contrast, the authors freed the fixed departments only when the exchange of locations of the fixed departments with other free departments produces a percentage reduction better than that at which the departments were fixed. It is important to note that intensification is employed after a certain number ($\eta$) of iterations have been performed (i.e. after a 'good' solution is obtained), since initially $PR(\pi^{best})$ may be large for a number of moves. Thus, exploring 'poor' solutions more thoroughly is a waste of computation time.

Therefore, after adding the above diversification and intensification strategies to TS1, TS2 is obtained by replacing steps 3–5 in TS1 with the following.

**Step 3:** Set $iter = iter + 1$, and find the best admissible move move* $= (t, u, v)$, which gives $\pi'$ according to $TCp(\pi') = TC(\pi') + Penalty(\text{move}(t, i, k))$.
If iter $> \eta$,
  if department $u$ or $v$ in period $t$ is fixed (i.e. $(t, u) \in \boldsymbol{F}$ or $(t, v) \in \boldsymbol{F}$ or both $(t, u) \in \boldsymbol{F}$ and $(t, v) \in \boldsymbol{F}$ where $\boldsymbol{F}$ = set of all fixed department-period pairs)
    if $TC(\pi') < TC(\pi^{best})$, then free the fixed department $u$ or $v$ in period $t$ (i.e. remove $(t, u)$, $(t, v)$, or both $(t, u)$ and $(t, v)$ from $\boldsymbol{F}$).
    Else, find move* where $u$ or $v$ is not fixed in period $t$.

**Step 4:** Set $\pi = \pi'$.
If $TC(\pi) < TC(\pi^{best})$, then set $\pi^{best} = \pi$ and
if iter $> \eta$ and $PR(\pi^{best}) \geq \gamma\%$, then fix department $u$ and $v$ in period $t$ (i.e. set $\boldsymbol{F} = \{\boldsymbol{F} \cup (t, u) \cup (t, v)\}$).
**Step 5:** Update tabu list as $tabu[t][u][v] = iter + TL_{iter}$ where $u < v$ and $TL_{iter}$ is obtained using the rules given above. Also, $tabu[t][v][u] = tabu[t][v][u] + 1$.

As stated earlier, Kaku and Mazzola (1997) presented the only TS heuristic for the DFLP, known to the authors. The main differences between their TS heuristic (called TSH) and our TS heuristic (called TS2) are as follows:

- In TSH, a move is defined to be tabu if it returns two departments to locations that they recently occupied. In contrast, in TS2 a pair of departments that exchange locations recently is defined to be tabu. As a result, each heuristic uses a different tabu list structure.
- In TSH, the stopping rule is to run the heuristic until the maximum number of iterations and maximum number of consecutive non-improving iterations has been achieved. However, in TS2 the heuristic terminates after a specified run time.
- In TSH, the diversification strategy is to use multiple starting solutions 'that are constructed in such a way as to obtain important differences between them'. In TS2, only a single starting solution is used, but frequency based memory and dynamic tabu tenure length (length may increase between iterations), as discussed above, are used to diversify the search.
- In TSH, the main idea of their intensification strategy is to use different tabu tenure length for each run on the same starting solutions. In other words, TSH is executed on each initial solution with different tabu tenure lengths separately, which makes the heuristic computationally expensive. In contrast, TS2 uses the strategies of fixing departments to locations and dynamic tabu tenure length (length may decrease between iterations), as discussed above, to intensify the search.

Last, the probabilistic TS heuristic for the DFLP is presented, called TS3. It is an adaptation of the probabilistic TS heuristics presented by Chiang and Chiang (1998) for the QAP and Lim *et al*. (2004) for a crane sequencing problem. The only difference between TS1 and TS3 is how the admissible move, move*$(t, u, v)$, is selected. Recall in TS1, all the possible moves are evaluated, and the best admissible move, move*$(t, u, v)$, is selected to obtain $\pi'$. However, in TS3, all possible moves are evaluated and ranked, and the top $M$ admissible moves are considered as candidate moves. Then the following procedure is used to select a move move*$(t, u, v)$ from the candidate list of moves such that $\pi'$ is obtained from performing move*$(t, u, v)$.

**Step 1:** Consider the first move $m$ in the candidate list.

**Step 2:** Accept the move $m$ with probability $p$. If the move is accepted, then this move is selected as move*$(t, u, v)$, and exit procedure. Else, go to step 3.

**Step 3:** Go to the next move in the candidate list and set as $m$. If there are no more candidate moves in the list, select the best move from the list of $M$ admissible moves with respect to total cost. Else, go to step 2.

## 4. Computational results

Two sets of test problems available in the literature will be used to test the performances of the proposed heuristics. The first set of test problems (32-problem data set), called data set 1, presented by Lacksonen and Enscore (1993), contains problems with 6, 12, 20, and 32 departments each with 3 and 5 time periods. The second set of test problems (48-problem data set), called data set 2, presented by Balakrishnan and Cheng (2000), contains problems with 6, 15, and 30 departments each with 5 and 10 time periods. The proposed TS heuristics were programmed using the C++ programming language, and the two data sets were solved on an AMD Athlon 2600+ 1.92 GHz PC. Each test problem was solved once using TS1 and TS2. However, each test problem was solved three times using TS3.

All the parameter settings for the proposed TS heuristics were obtained experimentally. For data set 1, $TL = NT/2$ for TS1 and TS3; $M = 10$ and $p = 0.33$ for TS3; and additional heuristic parameters for TS2 as well as the run time ($TRT$) for each heuristic are given in Table 2. For data set 2, $TL = NT/4$ for TS1 and TS3; $M = 5$ and $p = 0.33$ for TS3; and additional heuristic parameters for TS2 as well as the run time ($TRT$) for each heuristic are given in Table 3.

Table 2. TS2 heuristic parameters for data set 1.

| Problem size | | Diversification | | | | | Intensification | | |
|---|---|---|---|---|---|---|---|---|---|
| *N* | *T* | *LB* | *UB* | $\alpha\%$ | $\beta\%$ | *a* | $\eta$ | $\gamma\%$ | *TRT* (sec) |
| 6 | 3 | NT/8 | 7NT/8 | 0.065 | 0.13 | 3 | 12 | 0.035 | 1 |
| | 5 | NT/8 | 7NT/8 | 0.065 | 0.13 | 3 | 16 | 0.028 | 1 |
| 12 | 3 | NT/8 | 7NT/8 | 0.065 | 0.13 | 3 | 30 | 0.018 | 3 |
| | 5 | NT/8 | 7NT/8 | 0.065 | 0.13 | 3 | 40 | 0.013 | 60 |
| 20 | 3 | NT/8 | 7NT/8 | 0.065 | 0.13 | 3 | 50 | 0.016 | 150 |
| | 5 | NT/8 | 7NT/8 | 0.066 | 0.132 | 3 | 90 | 0.012 | 270 |
| 30 | 3 | NT/8 | 7NT/8 | 0.06 | 0.12 | 3 | 100 | 0.009 | 360 |
| | 5 | NT/8 | 7NT/8 | 0.0662 | 0.132 | 3 | 175 | 0.005 | 510 |

Table 3. TS2 heuristic parameters for data set 2.

| Problem size | | Diversification | | | | | Intensification | | |
|---|---|---|---|---|---|---|---|---|---|
| *N* | *T* | *LB* | *UB* | $\alpha\%$ | $\beta\%$ | *a* | $\eta$ | $\gamma\%$ | *TRT* (sec) |
| 6 | 5 | NT/16 | 15NT/16 | 0.0082 | 0.0164 | 15 | 10 | 0.008 | 1 |
| | 10 | NT/16 | 15NT/16 | 0.0082 | 0.0164 | 15 | 20 | 0.004 | 1 |
| 15 | 5 | NT/16 | 15NT/16 | 0.00799 | 0.01598 | 15 | 200 | 0.002 | 10 |
| | 10 | NT/16 | 15NT/16 | 0.00811 | 0.01622 | 15 | 400 | 0.001 | 35 |
| 30 | 5 | NT/16 | 15NT/16 | 0.0082 | 0.0164 | 15 | 500 | 0.0007 | 180 |
| | 10 | NT/16 | 15NT/16 | 0.0082 | 0.0164 | 15 | 900 | 0.0005 | 720 |

Table 4. Summary of results for data set 1.

| Problem size | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| *N* | *T* | TS1 | TS2 | TS3 | TS Best | SA | HAS | TS-KM |
| 6 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| | 5 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 2 | 3 | 3 | 4 | 4 | 4 | 4 | 4 | 4 |
| | 5 | 3 | 4 | 4 | 4 | 4 | 4 | 4 |
| 20 | 3 | 3 | 4 | 3 | 4 | 3 | 3 | 2 |
| | 5 | 3 | 4 | 4 | 4 | 3 | 3 | 2 |
| 30 | 3 | 0 | 3 | 0 | 3 | 3 | 2 | 2 |
| | 5 | 0 | 2 | 2 | 4 | 0 | 1 | 1 |
| Total | | 20 | 29 | 25 | 31 | 25 | 25 | 23 |

Table 4 summarises the number of best solutions obtained by TS1, TS2, and TS3, as well as the TS heuristic (TS-KM) presented by Kaku and Mazzola (1997) and HASs (HAS) presented by McKendall and Shang (2006) for data set 1. TS-KM and HAS are compared to the proposed heuristics, since these heuristics obtained the best solutions (available in the literature) for data set 1. Since the SA heuristics presented in McKendall *et al*. (2006) performed well for data set 2, both heuristics were programmed in C++, and the best results for data set 1 are also given in the table as (SA), after running each heuristic five times. Overall, TS2, TS3, SA, HAS, TS-KM, and TS1 obtained the best solutions for 29, 25, 25, 25, 23, and 20 problems, respectively, out of the 32 test problems. Hence, the TS2 heuristic out-performed the other heuristics with respect to solution quality. Also, the proposed heuristics, obtained the best solutions for 31 of the 32 test problems. For the total costs of the best solutions obtained for the TS1 and the TS2 heuristics as well as the best solution and the average of the 3 solutions obtained for the TS3 heuristic, see in the Appendix columns 4 to 7 of Tables A1 to A4, respectively. The solutions with an asterisk are the solutions Lacksonen and Enscore (1993) verified as optimal. Also, the total costs of the best solutions obtained from

Table 5. Summary of results for data set 2.

| Problem size | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| *N* | *T* | TS1 | TS2 | TS3 | TS Best | TABUSA | GATS | SA | HAS |
| 6 | 5 | 4 | 8 | 6 | 8 | 8 | 8 | 8 | 8 |
| | 10 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| 15 | 5 | 5 | 4 | 5 | 6 | 6 | 6 | 4 | 4 |
| | 10 | 0 | 2 | 1 | 3 | 5 | 1 | 1 | 1 |
| 30 | 5 | 0 | 5 | 1 | 6 | 1 | 1 | 0 | 0 |
| | 10 | 0 | 7 | 0 | 7 | 0 | 0 | 1 | 0 |
| Total | | 17 | 34 | 21 | 38 | 28 | 24 | 22 | 21 |

the proposed heuristics as well as the best solutions available in the literature are given in columns 8 and 9, respectively. Hence, the proposed TS heuristics obtained 5 new best solutions for data set 1.

Similarly, Table 5 summarises the number of best solutions obtained by TS1, TS2, and TS3, as well as the heuristics (TABUSA) presented by Sahin and Turkbey (2009), the GATA heuristic (GATA) presented by Rodriguez *et al*. (2006), the SA heuristics (SA) presented by McKendall *et al*. (2006), and the HASs (HAS) presented by McKendall and Shang (2006) for data set 2. These heuristics are compared to the proposed heuristics, since they obtained the best solutions (available in the literature) for this data set. Overall, TS2, TABUSA, GATS, SA, TS3, HAS, and TS1 obtained the best solutions for 34, 28, 24, 22, 21, 21, and 17 problems, respectively, out of the 48 test problems. Hence, the TS2 heuristic out-performed the other heuristics with respect to solution quality. Also, the proposed heuristics, obtained the best solutions for 38 of the 48 test problems. For the total costs of the best solutions obtained for the TS1 and the TS2 heuristics as well as the best solution and the average of the 3 solutions obtained for the TS3 heuristic, see in the Appendix columns 4 to 7 of Tables A5 to A7, respectively. Also, the total costs of the best solutions obtained from the proposed heuristics as well as the best solutions available in the literature are given in columns 8 and 9, respectively. Hence, the proposed TS heuristics obtained 14 new best solutions for data set 2.

Although TS2 out-performed TS1, TS3, and the heuristics available in the literature, with respect to solution quality, it has two major drawbacks. First, TS2 is more complex since it consists of diversification and intensification strategies. As a result, it is more difficult to code and require many more operations (more computation time) per iteration than TS1 or TS3, which indicate less iterations are needed to obtain high quality solutions. Second, there are many more heuristic parameters to set in TS2, which required a tremendous amount of effort.

## 5. Conclusion

In this paper, three TS heuristics were presented to solve the DFLP. The first (i.e. TS1) is a simple TS heuristic which uses static tabu tenure length. The second heuristic (i.e. TS2) uses dynamic tabu tenure length, frequency-based memory, and an intensification strategy, and the third heuristic (i.e. TS3) is a probabilistic TS heuristic. The proposed heuristics performed well on two data sets available in the literature. More importantly, the TS2 heuristic out-performed all the heuristics available in the literature for the DFLP. The following recommendations are given for future research.

- Consider re-arrangement costs data that are not fixed such that costs would depend on time periods (i.e. time-value of money), locations of the departments being interchanged, etc.
- Develop heuristics for the DFLP which require less heuristic parameters than the proposed TS2 heuristic but perform equally as well or better.

## References

Balakrishnan, J. and Cheng, C.H., 1998. Dynamic layout algorithm: A state-of-the-art survey. *OMEGA*, 26 (4), 507–521.

Balakrishnan, J. and Cheng, C.H., 2000. Genetic search and the dynamic layout problem. *Computers and Operations Research*, 27 (6), 587–593.

Balakrishnan, J., Cheng, C.H., and Conway, D.G., 2000. An improved pair-wise exchange heuristic for the dynamic plant layout problem. *International Journal of Production Research*, 38 (13), 3067–3077.
Balakrishnan, J., Cheng, C.H., Conway, D.G., and Lau, C.M., 2003. A hybrid genetic algorithm for the dynamic plant layout problem. *International Journal of Production Research*, 86 (2), 107–120.
Baykasoglu, A., Dereli, T., and Sabuncu, I., 2006. An ant colony algorithm for solving budget constrained and unconstrained dynamic facility layout problems. *OMEGA*, 34, 385–396.
Baykasoglu, A. and Gindy, N.N.Z., 2001. A simulated annealing algorithm for dynamic layout problem. *Computers and Operations Research*, 28 (14), 1403–1426.
Chiang, W. and Chiang, C., 1998. Intelligent local search strategies for solving facility layout problems with the quadratic assignment problem formulation. *European Journal of Operational Research*, 106 (2), 457–488.
Chiang, W. and Kouvelis, P., 1996. An improved tabu search heuristic for solving facility layout design problems. *International Journal of Production Research*, 34, 2565–2585.
Conway, D.G. and Venkataramanan, M.A., 1994. Genetic search and the dynamic facility layout problem. *Computers and Operations Research*, 21 (8), 955–960.
Erel, E., Ghosh, J.B., and Simon, J.T., 2003. New heuristic for the dynamic layout problem. *Journal of the Operational Research Society*, 54 (12), 1275–1282.
Francis, R.L., McGinnis, L.F., and White, J.A., 1992. *Facility layout and location: An analytical approach*. New Jersey: Prentice Hall.
Glover, F., 1986. Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 13 (5), 533–549.
Kaku, B.K. and Mazzola, J.B., 1997. A tabu search heuristic for the dynamic plant layout problem. *INFORMS Journal on Computing*, 9 (4), 374–384.
Kulturel-Konak, S., 2007. Approaches to uncertainties in facility layout problems: Perspectives at the beginning of the 21st century. *Journal of Intelligent Manufacturing*, 18, 273–284.
Kusiak, A. and Heragu, S.S., 1987. The facility layout problem. *European Journal of Operational Research*, 29 (3), 229–251.
Lacksonen, T.A. and Enscore, E.E., 1993. Quadratic assignment algorithms for the dynamic layout problem. *International Journal of Production Research*, 31 (3), 503–517.
Lim, A., Rodrigues, B., Xiao, F., and Zhu, Y., 2004. Crane scheduling with spatial constraints. *Naval Research Logistics*, 51, 386–406.
McKendall, A.R. and Shang, J., 2006. Hybrid ant systems for the dynamic facility layout problem. *Computers and Operations Research*, 33 (3), 790–803.
McKendall, A.R., Shang, J., and Kuppusamy, S., 2006. Simulated annealing heuristics for the dynamic facility layout problem. *Computers and Operations Research*, 33 (8), 2431–2444.
Meller, R.D. and Gau, K.Y., 1996. The facility layout problem: Recent and emerging trends and perspectives. *Journal of Manufacturing Systems*, 15 (5), 351–366.
Rodriguez, J.M., MacPhee, F.C., Bonham, D.J., and Bhavsar, V.C., 2006. Solving the dynamic plant layout problem using a new hybrid meta-heuristic algorithm. *International Journal of High Performance Computing and Networking*, 4 (5/6), 286–294.
Rosenblatt, M.J., 1986. The dynamics of plant layout. *Management Science*, 32 (1), 76–86.
Sahin, R. and Turkbey, O., 2009. A new hybrid tabu-simulated annealing heuristic for the dynamic facility layout problem. *International Journal of Production Research*, 47 (24), 6855–6873.
Skorin-Kapov, J., 1990. Tabu search applied to the quadratic assignment problem. *ORSA Journal on Computing*, 2, 33–45.
Skorin-Kapov, J., 1994. Extensions of a tabu search adaptation to the quadratic assignment problem. *Computers and Operations Research*, 21, 855–865.
Taillard, E., 1991. Robust tabu search for quadratic assignment. *Parallel Computing*, 17, 443–455.
Tompkins, J.A., White, J.A., Bozer, Y.A., Frazelle, E.H., Tanchoco, J.M.A., and Trevino, J., 2003. *Facilities Planning*. New York: John Wiley & Sons.
Urban, T.L., 1993. A heuristic for the dynamic facility layout problem. *IIE Transactions*, 25 (4), 57–63.

**Appendix 1: Proposed heuristic solutions and best solutions from the literature for data set 1**

Table A1. Results for data set 1 where $N = 6$.

| Problem size | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| *N* | *T* | Pb # | TS1 | TS2 | TS3 best | TS3 avg | TS best | Best available |
| 6 | 3 | P01 | **267*** | **267*** | **267*** | 280 | **267*** | **267*** |
| | | P02 | **260*** | **260*** | **260*** | 263 | **260*** | **260*** |
| | | P03 | **363*** | **363*** | **363*** | **363*** | **363*** | **363*** |
| | | P04 | **299*** | **299*** | **299*** | **299*** | **299*** | **299*** |
| 6 | 5 | P05 | **442*** | **442*** | **442*** | **442*** | **442*** | **442*** |
| | | P06 | **586** | **586** | **586** | 604 | **586** | **586** |
| | | P07 | **424*** | **424*** | **424*** | 466 | **424*** | **424*** |
| | | P08 | **428*** | **428*** | **428*** | 436 | **428*** | **428*** |

Table A2. Results for data set 1 where $N = 12$.

| Problem size | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| *N* | *T* | Pb # | TS1 | TS2 | TS3 best | TS3 avg | TS Best | Best available |
| 12 | 3 | P09 | **1624** | **1624** | **1624** | 1636 | **1624** | **1624** |
| | | P10 | **1973** | **1973** | **1973** | 1978 | **1973** | **1973** |
| | | P11 | **1661** | **1661** | **1661** | 1670 | **1661** | **1661** |
| | | P12 | 2102 | **2097** | **2097** | 2100 | **2097** | **2097** |
| 12 | 5 | P13 | **2930** | **2930** | **2930** | **2930** | **2930** | **2930** |
| | | P14 | **3701** | **3701** | **3701** | **3701** | **3701** | **3701** |
| | | P15 | 2779 | **2756** | **2756** | **2756** | **2756** | **2756** |
| | | P16 | **3364** | **3364** | **3364** | **3364** | **3364** | **3364** |

Table A3. Results for data set 1 where $N = 20$.

| Problem size | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| *N* | *T* | Pb # | TS1 | TS2 | TS3 best | TS3 avg | TS best | Best available |
| 20 | 3 | P17 | **2758** | **2758** | **2758** | **2758** | **2758** | **2758** |
| | | P18 | **5318** | **5318** | **5318** | **5318** | **5318** | **5318** |
| | | P19 | **3034** | **3034** | **3034** | 3048 | **3034** | **3034** |
| | | P20 | 5873 | **5869** | 5873 | 5884 | **5869** | 5873 |
| 20 | 5 | P21 | **4554** | **4554** | **4554** | 4571 | **4554** | **4554** |
| | | P22 | 9746 | **9724** | **9724** | 9735 | **9724** | **9724** |
| | | P23 | **4654** | **4654** | **4654** | **4654** | **4654** | **4654** |
| | | P24 | **8979** | **8979** | **8979** | **8979** | **8979** | **8979** |

Table A4. Results for data set 1 where $N = 30$.

| Problem size | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| *N* | *T* | Pb # | TS1 | TS2 | TS3 best | TS3 avg | TS best | Best available |
| 30 | 3 | P25 | 7131 | **7130** | 7131 | 7133 | **7130** | **7130** |
| | | P26 | 14528 | **14478** | 14528 | 14528 | **14478** | **14478** |
| | | P27 | 8100 | **8065** | 8068 | 8106 | **8065** | 8066 |
| | | P28 | 14933 | 14915 | 14957 | 14968 | 14915 | **14901** |
| 30 | 5 | P29 | 13446 | **13372** | 13468 | 13500 | **13372** | 13374 |
| | | P30 | 25515 | 25517 | **25462** | 25495 | **25462** | 25472 |
| | | P31 | 12163 | 12163 | **12148** | 12167 | **12148** | 12163 |
| | | P32 | 24307 | **24200** | 24247 | 24258 | **24200** | **24200** |

## Appendix 2: Proposed heuristic solutions and best solutions from the literature for data set 2

Table A5. Results for data set 2 where $N = 6$.

| Problem size | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| *N* | *T* | Pb # | TS1 | TS2 | TS3 best | TS3 avg | TS best | Best available |
| 6 | 5 | P01 | **106,419** | **106,419** | **106,419** | **106,419** | **106,419** | **106,419** |
| | | P02 | **104,834** | **104,834** | **104,834** | **104,834** | **104,834** | **104,834** |
| | | P03 | 104,520 | **104,320** | **104,320** | 104,387 | **104,320** | **104,320** |
| | | P04 | **106,399** | **106,399** | **106,399** | **106,399** | **106,399** | **106,399** |
| | | P05 | 105,737 | **105,628** | **105,628** | 105,701 | **105,628** | **105,628** |
| | | P06 | **103,985** | **103,985** | **103,985** | **103,985** | **103,985** | **103,985** |
| | | P07 | 106,447 | **106,439** | 106,447 | 106,447 | **106,439** | **106,439** |
| | | P08 | 106,152 | **103,771** | 106,152 | 106,152 | **103,771** | **103,771** |
| 6 | 10 | P09 | **214,313** | **214,313** | **214,313** | 215,761 | **214,313** | **214,313** |
| | | P10 | **212,134** | **212,134** | **212,134** | 212,699 | **212,134** | **212,134** |
| | | P11 | **207,987** | **207,987** | **207,987** | 208,335 | **207,987** | **207,987** |
| | | P12 | **212,530** | **212,530** | **212,530** | 213,011 | **212,530** | **212,530** |
| | | P13 | **210,906** | **210,906** | **210,906** | 211,676 | **210,906** | **210,906** |
| | | P14 | **209,932** | **209,932** | **209,932** | 210,094 | **209,932** | **209,932** |
| | | P15 | **214,252** | **214,252** | **214,252** | 214,519 | **214,252** | **214,252** |
| | | P16 | **212,588** | **212,588** | **212,588** | 213,099 | **212,588** | **212,588** |

Table A6. Results for data set 2 where $N = 15$.

| Problem size $N$ | $T$ | Pb # | TS1 | TS2 | TS3 best | TS3 avg | TS best | Best available |
|---|---|---|---|---|---|---|---|---|
| 15 | 5 | P17 | **480,453** | **480,453** | **480,453** | 480,468 | **480,453** | **480,453** |
| | | P18 | **484,761** | **484,761** | **484,761** | 484,786 | **484,761** | **484,761** |
| | | P19 | 489,335 | **488,748** | 489,265 | 489,312 | **488,748** | **488,748** |
| | | P20 | 484,621 | 484,446 | 484,621 | 484,893 | 484,446 | **484,405** |
| | | P21 | 487,822 | 487,911 | 487,753 | 487,901 | 487,753 | **487,722** |
| | | P22 | **486,493** | **486,493** | **486,493** | 487,179 | **486,493** | **486,493** |
| | | P23 | **486,268** | 486,592 | **486,268** | 487,087 | **486,268** | **486,268** |
| | | P24 | **490,551** | 490,812 | **490,551** | 491,572 | **490,551** | **490,551** |
| 15 | 10 | P25 | 983,061 | 981,412 | 980,906 | 981,374 | 980,906 | **978,848** |
| | | P26 | 978,874 | 978,004 | 978,815 | 979,398 | 978,004 | **977,338** |
| | | P27 | 982,944 | 983,109 | 983,898 | 984,298 | 982,944 | **978,027** |
| | | P28 | 972,325 | **971,720** | 972,019 | 972,915 | **971,720** | **971,720** |
| | | P29 | 978,033 | 977,100 | 977,534 | 978,103 | 977,100 | **976,310** |
| | | P30 | 969,124 | 971,287 | **967,617** | 968,722 | **967,617** | **967,617** |
| | | P31 | 979,881 | **978,576** | 979,513 | 980,018 | **978,576** | 978,660 |
| | | P32 | 985,105 | 983,341 | 985,105 | 985,567 | 983,341 | **982,888** |

Table A7. Results for data set 2 where $N = 30$.

| Problem size $N$ | $T$ | Pb # | TS1 | TS2 | TS3 best | TS3 avg | TS best | Best available |
|---|---|---|---|---|---|---|---|---|
| 30 | 5 | P33 | 576,269 | 574,657 | **574,577** | 575,682 | **574,577** | 574,624 |
| | | P34 | 569,119 | **567,481** | 567,691 | 569,553 | **567,481** | 567,992 |
| | | P35 | 573,930 | **571,462** | 573,307 | 573,705 | **571,462** | 572,865 |
| | | P36 | 565,637 | 564,868 | 565,849 | 566,811 | 564,868 | **564,726** |
| | | P37 | 556,946 | **555,628** | 557,098 | 557,381 | **555,628** | 557,138 |
| | | P38 | 565,559 | **565,100** | 565,670 | 566,200 | **565,100** | 565,388 |
| | | P39 | 574,278 | **566,993** | 571,085 | 572,568 | **566,993** | 567,131 |
| | | P40 | 573,873 | 573,023 | 574,854 | 575,101 | 573,023 | **572,992** |
| 30 | 10 | P41 | 1,160,941 | **1,159,589** | 1,160,196 | 1,161,712 | **1,159,589** | 1,161,751 |
| | | P42 | 1,160,273 | **1,157,942** | 1,159,088 | 1,161,423 | **1,157,942** | 1,160,656 |
| | | P43 | 1,158,212 | **1,154,799** | 1,155,280 | 1,157,220 | **1,154,799** | 1,155,406 |
| | | P44 | 1,149,047 | **1,143,110** | 1,146,881 | 1,147,592 | **1,143,110** | 1,144,345 |
| | | P45 | 1,127,721 | **1,123,446** | 1,125,429 | 1,126,650 | **1,123,446** | 1,125,968 |
| | | P46 | 1,143,559 | **1,141,144** | 1,144,625 | 1,144,990 | **1,141,144** | 1,141,344 |
| | | P47 | 1,150,130 | 1,145,951 | 1,146,200 | 1,147,172 | 1,145,951 | **1,140,744** |
| | | P48 | 1,166,646 | **1,160,484** | 1,163,528 | 1,165,437 | **1,160,484** | 1,161,437 |