On a New Modelling Approach for Circular Layouts and its Practical Advantages

Philipp Hungerländer∗ Kerstin Maier† Jörg Pöcher‡ Christian Truden§

June 11, 2017

Abstract

We consider a new facility layout problem. The Directed Circular Facility Layout Problem (DCFLP) seeks to optimally arrange machines on a circular layout with given material flow direction. The DCFLP allows for a wide range of applications and contains several other relevant layout problems as special cases. We model the DCFLP as a Linear Ordering Problem and solve it using an Integer Linear Program and a Tabu Search heuristic. In our computational study we show that the DCFLP is easier to solve for both exact and heuristic approaches than other related layout problems.

Keywords: Facility planning and design; circular layout; linear ordering problem; integer linear programming; tabu search.

1 Introduction

Facility layout is concerned with the optimal location of machines inside a plant according to a given objective function that may reflect transportation costs, the construction cost of a material-handling system, the costs of laying communication wiring, or simply adjacency preferences among machines. Facility layout is a well-known operations research problem and arises in different areas of application.

The survey paper by Meller and Gau [22] divides facility layout research into three broad categories. The first is concerned with models and algorithms for tackling different versions of the basic layout problem that asks for the optimal arrangement of a given number of machines within a facility so as to minimize the total expected cost of flows inside the facility. This includes the well-known special case of the Quadratic Assignment Problem (QAP) in which all the machine sizes are equal. The second category is concerned with extensions of unequal-areas layout that take into account additional issues that arise in real-world applications, such as designing dynamic layouts by taking time-dependency issues into account, designing layouts under uncertainty conditions, and computing layouts that optimize two or more objectives simultaneously. The third category is concerned with specially structured instances of the problem, such as the layout of machines along a

∗Laboratory for Information & Decision Systems, MIT, USA, philipp.hungerlaender@aau.at
†Department of Mathematics, Alpen-Adria-Universität Klagenfurt, Austria, kerstin.maier@aau.at
‡Department of Mathematics, Alpen-Adria-Universität Klagenfurt, Austria, jorg.poecher@aau.at
§Department of Mathematics, Alpen-Adria-Universität Klagenfurt, Austria, christian.truden@aau.at
production line. This article is concerned with finding global upper and lower bounds for one such type of structured instances, namely the Directed Circular Facility Layout Problem (DCFLP) that has recently been suggested by Hungerländer [10].

The DCFLP seeks to arrange the machines on a circular material handling system so as to minimize the total weighted sum of the center-to-center distances between all pairs of machines measured in clockwise direction. The material handling system is assumed to move the parts unidirectionally around the circuit following the sequence specified in its process plan. Each machine is capable of picking up and processing the parts from the material handling system [19].

Circular material handling systems are mostly preferred because of their relative low initial investment costs, high material handling flexibility, their space-saving design and their ability of being easily accommodated to future introduction of new parts and process changes [1, 14].

An instance of the Directed Circular Facility Layout Problem (DCFLP) consists of \( n \) one-dimensional machines with given positive lengths \( \ell_1, \ldots, \ell_n \) and pairwise flows \( f_{ij}, \ i, j \in [n], i \neq j \). The machines are arranged next to each other on a circle. The objective is to find a permutation \( \pi \) of the machines such that the total weighted sum of the center-to-center distances between all pairs of machines (in clockwise direction) is minimized

\[
\min_{\pi \in \Pi_n} \sum_{i,j \in [n], i \neq j} f_{ij} z_{ij}^\pi,
\]

where \( \Pi_n \) is the set of all feasible layouts of the machines \( [n] := \{1, 2, \ldots, n\} \) and \( z_{ij}^\pi \) gives the distance between the centroids of machines \( i \) and \( j \) in the circular layout \( \pi \) (in clockwise direction).

While Hungerländer [10] suggested a semidefinite programming (SDP) method for solving the DCFLP, in this paper we show that the DCFLP can in fact be modelled as a Linear Ordering Problem (LOP). In its matrix version the LOP can be defined as follows. Given an \( n \times n \) matrix \( W = (w_{ij}) \) of integers, find a simultaneous permutation \( \pi \) of the rows and columns of \( W \) such that

\[
\sum_{i,j \in [n], i < j} w_{\pi(i), \pi(j)},
\]

is maximized. Equivalently, we can interpret \( w_{ij} \) as weights of a complete directed graph \( G \) with vertex set \( V = [n] \). A tournament consists of a subset of the arcs of \( G \) containing for every pair of nodes \( i \) and \( j \) either arc \( (i, j) \) or arc \( (j, i) \), but not both. Then the LOP consists of finding an acyclic tournament, i.e. a tournament without directed cycles, of \( G \) of maximum total edge weight.

The LOP is a very well-studied problem for which several high-quality exact methods and heuristics have been proposed. Hence the DCFLP can be solved very efficiently by applying these approaches. In particular we suggest to use an Integer Linear Program (ILP) and a Tabu Search (TS) heuristic for tackling the DCFLP. These approaches are especially interesting for practitioners as they not only outperform the SDP method by Hungerländer [10] but are also very easy to implement.

Next let us consider the well-known Single-Row Facility Layout Problem (SRFLP) that is closely related to the DCFLP. It arises as the problem of ordering stations on a production line where the material flow is handled by an automated guided vehicle travelling in both directions on a straight-line path [9]. An instance of the SRFLP consists of \( n \) one-dimensional machines, with given positive lengths \( \ell_1, \ldots, \ell_n \), and pairwise connectivities \( c_{ij} \). The optimization problem can be written down as

\[
\min_{\pi \in \Pi_n} \sum_{i,j \in [n], i < j} c_{ij} z_{ij}^\pi,
\]
where $\Pi_n$ is the set of permutations of the machines $[n]$ and $\zeta_{ij}^\pi$ is the center-to-center distance between machines $i$ and $j$ with respect to a particular permutation $\pi \in \Pi_n$.

The SRFLP is one of the few layout problems for which strong global lower bounds and even optimal solutions can be computed for instances of reasonable size. The global optimization approaches for the SRFLP are based on relaxations of ILP and SDP formulations. The strongest ones are an LP-based cutting plane algorithm using betweenness variables [3] and an SDP approach using products of ordering variables [12]. In summary we suggest a model for the DCFLP that is significantly easier (linear terms instead of linear-quadratic terms in ordering variables) than all available formulations for the SRFLP that to date is generally considered as the facility layout problem with the easiest structure.

Additional to its practical relevance for the design of flexible manufacturing systems [10], the DCFLP is a very interesting problem as it is a generalization of several other layout problems that have been extensively discussed in the literature. To begin with the DCFLP which is a generalization of the Directed Circular Arrangement Problem (DCAP) that allows the machines to have arbitrary instead of the same lengths. The DCAP was first considered by Liberatore [17] who showed that the problem is NP-hard (hence also the DCFLP is NP-hard). We refer to Bar-Noy et al. [2], Liberatore [17] and Naor and Schwartz [23] for several nice applications of the DCAP in the areas of server design and ring networks.

Furthermore the DCFLP is related to the NP-hard [13] Unidirectional Cyclic Layout Problem (UCFLP) [2]. The UCFLP also considers a circular material handling system with directed flow and the objective is to find an assignment of $n$ machines to $n$ predetermined candidate locations such that the total handling cost is minimized. The UCFLP has two well-known special cases that are at the same time special cases of the DCFLP:

1. In the balanced unidirectional cyclic layout problem (BUCFLP) the material flow is conserved at each machine, i.e. the total inflow is equal to total outflow at each machine.

2. In the equidistant unidirectional cyclic layout problem (EUCFLP) the locations around the unidirectional circular material handling system are assumed to be equally distant to each other.

Bozer and Rim [5] have shown that the EUCFLP and the BUCFLP are in fact equivalent.

In comparison with the UCFLP, the DCFLP considers machine lengths instead of the distances of the locations and hence is an adaption of the SRFLP to circular layouts. Considering machine lengths instead of the location distances (i.e. location lengths) is clearly preferable in many practical applications where the lengths of the machines are the relevant input parameters. Additionally solving the UCFLP with heuristic and exact methods is very hard as it is a special QAP and QAPs are known to be notoriously difficult to solve [18]. Therefore optimizing circular layouts was considered to be clearly harder than optimizing row layouts. In this paper we aim to reveal that the opposite is true, if we determine circular layouts with the help of the DCFLP (modelled as a LOP) instead of the UCFLP (modelled as a QAP).

Finally let us further clarify the connections and differences of the SRFLP and the DCFLP with the help of a toy example: We consider 4 machines with lengths $\ell_1 = 1$, $\ell_2 = 2$, $\ell_3 = 3$, $\ell_4 = 4$. Additionally we have given the pairwise connectivities $c_{12} = c_{14} = c_{34} = 1$, $c_{13} = c_{24} = 2$ for the SRFLP and pairwise flows $f_{12} = f_{14} = f_{43} = 1$, $f_{13} = f_{42} = 2$ for the DCFLP. Fig. illustrates the optimal layouts and the according costs for both problems.

In summary the two main contributions of this paper are the following:
a.)

b.)

Figure 1: We have given the following data: $\ell_1 = 1$, $\ell_2 = 2$, $\ell_3 = 3$, $\ell_4 = 4$, $c_{12} = c_{14} = c_{34} = 1$, $c_{13} = c_{24} = 2$, $f_{12} = f_{14} = f_{43} = 1$, $f_{13} = f_{42} = 2$. In a.) we display the optimal layout for the SRFLP with associated costs of $3 \cdot 2 + 2 \cdot 5 \cdot 1 + 2 \cdot 2 + 5 \cdot 5 \cdot 1 = 22.5$. In b.) we depict the optimal layout for the DCFLP with associated costs of $2 \cdot 2 + 3 \cdot 2 + 5 \cdot 5 \cdot 1 + 8 \cdot 5 \cdot 1 + 6 \cdot 5 \cdot 1 = 30.5$.

1. We show that the DCFLP can be modelled as a LOP.
2. Building on this model we demonstrate that both heuristic and exact methods for the DCFLP are easier to implement and more efficient than the best current approaches for the SRFLP and the UCFLP.

The paper is structured as follows. In Section 2 we deduce a mathematical formulation for the DCFLP based on binary ordering variables. In Section 3 we briefly recall the best exact and heuristic approaches for the LOP which additionally are easy to implement. In Section 4 we summarize the results of our computational study for the DCFLP and Section 5 concludes the paper.

2 Mathematical Formulation

To model the DCFLP as a LOP we introduce $O(n^2)$ binary ordering variables $x_{ij}$, $i, j \in [n]$, $i < j$:

$$x_{ij} = \begin{cases} 1, & \text{if machine } i \text{ is located before machine } j, \\ 0, & \text{otherwise}. \end{cases}$$

Any feasible ordering of the machines on the circle has to fulfill the 3-cycle inequalities:

$$0 \leq x_{ij} + x_{jk} - x_{ik} \leq 1, \quad i, j, k \in [n], \ i < j < k. \quad (2)$$

It is well-known that the 3-cycle inequalities together with integrality conditions on the variables suffice to describe feasible orderings.

Next we introduce the distance variables $d_{ij}$, $i, j \in [n]$, $i < j$, which give the distance between machines $i$ and $j$, if all machines are arranged on a straight line. This arrangement is in fact a single-row layout of the machines that is defined through the ordering variables. Hence we can compute $d_{ij}$ as the difference of the sums of the lengths of the machines in front of machine $i$ and
machine \( j \) respectively:

\[
d_{ij} = \left( \frac{\ell_i}{2} + \sum_{k \in [n]} \ell_k x_{ki} + \sum_{k \in [n]} \ell_k (1 - x_{kk}) \right) - \left( \frac{\ell_j}{2} + \sum_{k \in [n]} \ell_k x_{kj} + \sum_{k \in [n]} \ell_k (1 - x_{jk}) \right), \quad i < j \in [n].
\]

The \( d_{ij} \) are linear expressions in the ordering variables \( x_{ij} \). To destroy symmetry and reduce the dimensions of the problem, we fix machine 1 to be first in the ordering. Hence we set \( x_{1j} = 1, \ j \in [n], \ j \neq 1 \). This results (after some additional simplifications) in the following adaption of the distance variables

\[
d_{1j} = -\frac{\ell_1 - \ell_j}{2} - \sum_{\substack{k \in [n] \ 1 < k < j}} \ell_k x_{kj} - \sum_{\substack{k \in [n] \ k > j}} \ell_k (1 - x_{jk}),
\]

\[
d_{ij} = \left( \frac{\ell_i}{2} + \sum_{\substack{k \in [n] \ 1 < k < i}} \ell_k x_{ki} + \sum_{\substack{k \in [n] \ k > i}} \ell_k (1 - x_{ik}) \right) - \left( \frac{\ell_j}{2} + \sum_{\substack{k \in [n] \ 1 < k < j}} \ell_k x_{kj} + \sum_{\substack{k \in [n] \ k > j}} \ell_k (1 - x_{jk}) \right), \quad i < j \in [n],
\]

where \( i, j \in [n], \ 1 < i < j \). Next we determine the distances between machines \( i \) and \( j \) on the circle, denoted by \( z_{ij}, \ i, j \in [n], \ i \neq j \), via the distance variables

\[
z_{1j} = -d_{1j}, \ z_{ij} = L + d_{1j}, \quad z_{ij} = -d_{ij} + (1 - x_{ij})L, \quad z_{ji} = d_{ij} + x_{ij}L, \quad i, j \in [n], \ 1 < i < j,
\]

where \( L = \sum_{k \in [n]} \ell_k \) denotes the sum of the lengths of all machines. Now we can rewrite the objective function \( (1) \) with the help of \( (4) \) as a linear function in \( \binom{n-1}{2} \) ordering variables:

\[
\min_{x \in \{0,1\}^{\binom{n-1}{2}}} f(x)
\]

where

\[
f(x) := L \sum_{i, j \in [n], 1 < i < j} f_{ij} + \sum_{i, j \in [n], 1 < i < j} (f_{ji} - f_{ij}) (-d_{ij} + Lx_{ij}) + \sum_{j \in [n], j \neq 1} [(f_{j1} - f_{1j})d_{1j} + f_{j1}L],
\]

and \( x \) is a vector collecting all the ordering variables.

In summary we have deduced the following formulation of the DCFLP based on ordering variables.

**Theorem 1.** The Linear Ordering Problem \( (5) \) subject to \( (2) \) and \( (3) \) is equivalent to the DCFLP.

**Proof.** The inequalities \( (2) \) together with the integrality conditions on \( x \) suffice to induce a feasible layout on the circle. Equation \( (3) \) connects the ordering with the distance variables and finally the definition of the objective function ensures that the distances between machines are computed correctly and weighted with the appropriate flows.

### 3 Exact and Heuristic Methods for the L0P

The current state-of-the-art Branch-and-Cut algorithm was developed by the working group of Reinelt in Heidelberg and is based on sophisticated cut generation procedures (for details see \[20\]).
It can solve large instances from specific instance classes with up to 150 objects, while it fails on other much smaller instances with only 50 or even less objects.

For our purpose we suggest to use a standard ILP approach for the LOP that shows a very good practical performance and additionally is easy to implement. Our approach is based on the following simple model for the LOP:

\[
\max \left\{ \sum_{i,j \in [n], i < j} (w_{ij} - w_{ji})x_{ij} : x_{ij} \in \{0,1\}, 0 \leq x_{ij} + x_{jk} - x_{ik} \leq 1, i < j < k \in [n] \right\}.
\]

There exist many and several very strong heuristics for both the LOP and the SRFLP, where insert moves are used as the primary mechanism to move from one solution to another. But the costs for these insert moves differ significantly for these two problems. For the LOP an insertion can be done in \(O(n)\) time because only the costs in the row and column of matrix \(W\) have to be considered for the shifted element. For the SRFLP an insert move needs \(O(n^2)\) time because in principle all entries of the cost matrix \(C\) have to be considered. This difference suggests that heuristics for the DCFLP obtain in general stronger solutions with less computational effort than heuristics for the SRFLP. This is also supported by the extensive experiments conducted in the literature: While for the LOP high quality solutions for different types of instances with up to 500 objects are obtained within seconds by various heuristics \([20, 21]\), the strongest heuristics for the SRFLP need up to a few hundred seconds on instances with only 80 objects (on faster computers) to obtain solutions of about the same quality \([7]\).

All relevant metaheuristics for the LOP have been tested on a large variety of benchmark instances with up to 500 objects in \([21]\) and the best ones are the Memetic Algorithm by Schiavinotto and Stützle \([24]\) and the Tabu Search (TS) heuristic by Laguna et al. \([15]\). The differences between these two methods, e.g. with respect to the rank value or the average percentage deviations from the best solutions found, are very small. As there are no implementations of metaheuristics for the LOP (and the SRFLP) available online, we decided to use the TS method from \([15]\), as this heuristic can be implemented very quickly, all relevant parameters are given in the paper, and also the fine-tuning process can be conducted easily.

Laguna et al. \([15]\) complemented the basic TS procedure with a long-term intensification based on the path relinking methodology, and with a long-term diversification based on the REVERSE operation proposed by Chanas and Kobyłanski \([6]\). Both long-term strategies incorporate frequency information recorded during the application of the short-term phase. Due to clear description of the method, we could facilely reproduce the results of the computational experiments in \([20]\). This easy reproducibility is an important feature for practitioners looking for an efficient layout in their production plant. We can conclude from the extensive computational studies in the literature that high-quality DCFLP layouts for instances with up to 500 machines can be obtained by state-of-the-art heuristics for the LOP. In our computational study we additionally apply both our TS heuristic and our ILP approach to well-known facility layout benchmark instances with up to 80 machines.

## 4 Computational Experiments

In order to set up a benchmark library for the DCFLP we use well-known benchmark instances for the SRFLP and adapt them to the DCFLP as follows: We directly transfer the number of machines and their lengths. For adapting the flows we propose two variants. Either we just set \(f_{ij} := c_{ij}\) and \(f_{ji} := 0\) or we decide randomly (with the same probability for both cases) if \(f_{ij} := c_{ij}, f_{ji} := 0\) or \(f_{ij} := 0, f_{ji} := c_{ij}\). We denote the first variant as one-way, the second one as random.
We restrict ourselves to cases where either $f_{ij}$ or $f_{ji}$ is zero. If both $f_{ij}$ and $f_{ji}$ are greater than zero, then we can remodel the problem by setting $f_{ij}^n := f_{ij} - \min(f_{ij}, f_{ji})$, $f_{ji}^n := f_{ji} - \min(f_{ij}, f_{ji})$ and adding the constant $\min(f_{ij}, f_{ji}) \cdot L$ to the objective function. Notice that for such problems the additional constant in the objective function just reduces the relative gap.

We report the results for the DCFLP using our TS heuristic implemented in Java and our ILP approach using Gurobi 6.5.1 [8]. All computations were conducted on an Intel Xeon E5160 (Dual-Core) with 24 GB RAM, running Debian 5.0 in 64-bit mode. All DCFLP instances and the corresponding best layouts found as well as the according ordering problems in standard format together with the best orderings obtained can be downloaded from http://tinyurl.com/dcflp-lib.

We summarized the results of our experiments with our (ILP) approach and our TS heuristic in Tables 1 and 2 respectively. Due to space restrictions we only include detailed computational results for one instance of each instance type. There exist up to five instances of each instance type, but the results for instances of the same type are very homogeneous.

First we notice that the one-way variant is a lot easier to solve for both our exact and heuristic approaches than the random variant. In particular our ILP approach determines the optimal solution within seconds for all benchmark instances considered and our TS provides close-to-optimal layouts with a deviation of at most 0.5 % from the optimal objective value, expect for the instance ste36-1.

For the random variant our ILP approach rather quickly determines the optimal solutions for all instances with up to 42 machines. For larger instances it provides reasonable bounds within the given time limit of 24 hours. Our TS heuristic again solves each instance within one second and provides better feasible layouts than the ILP approach for all instances with at least 60 machines, except for the instance sko4-1.

We also compared our ILP approach with the SDP method from [10]. While our ILP approach is clearly preferable on the one-way instances and the random instances with up to 50 machines, the SDP method obtains the stronger bounds for random instances with more than 50 machines. We will provide a more detailed comparison between these two exact approaches together with further experiments on DCAP and BUCFLP instances in a future journal paper.

In summary our computational study supports the claim that the DCFLP is easier to solve than the SRFLP and the UCFLP. Especially the meta-heuristics for the LOP are by several orders of magnitude faster than comparable heuristics for the SRFLP and the QAP and additionally provide layouts of higher quality most of the time.

5 Outlook and Conclusions

We have considered the Directed Circular Facility Layout Problem (DCFLP) that represents a new modelling approach for circular layouts. The DCFLP allows for a wide range of applications and can be solved very efficiently, e.g. via Integer Linear Programming and Tabu Search algorithms, because the DCFLP can be modelled as a Linear Ordering Problem (LOP). We provided theoretical arguments and a computational study supporting our claim that the DCFLP can be solved more efficiently than the Single-Row Facility Layout Problem and the Unidirectional Cyclic Layout Problem by both exact and heuristic methods.

As a future research topic we suggest to enhance the exact approaches and heuristics for the LOP, and thus also for the DCFLP, with the ability to handle dynamic and stochastic aspects as well as the possibility to consider multiple objectives. Then these approaches would be applicable for optimizing layouts in many additional complex real-world applications.
Table 1: Summary of the results produced by our ILP approach when applied to DCFLP instances with 15 to 80 machines. The running times are given in sec, min:sec or h:min:sec respectively. We set a time limit of 24 hours. Gap indicates the relative gap between the best lower and upper bound determined by our ILP approach.

<table>
<thead>
<tr>
<th>Instance</th>
<th></th>
<th>ILP one-way</th>
<th></th>
<th></th>
<th>ILP random</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Optimal</td>
<td></td>
<td></td>
<td>Best</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>layout</td>
<td></td>
<td></td>
<td>layout</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P15</td>
<td></td>
<td>8284.0</td>
<td>0.01</td>
<td></td>
<td>8041.0</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>P17</td>
<td></td>
<td>12717.0</td>
<td>0.01</td>
<td></td>
<td>13105.0</td>
<td>0.02</td>
<td>0.00</td>
</tr>
<tr>
<td>P18</td>
<td></td>
<td>14450.5</td>
<td>0.01</td>
<td></td>
<td>14726.5</td>
<td>0.01</td>
<td>0.00</td>
</tr>
<tr>
<td>N25_01</td>
<td></td>
<td>6221.0</td>
<td>0.01</td>
<td></td>
<td>5921.0</td>
<td>2.09</td>
<td>0.00</td>
</tr>
<tr>
<td>N30_01</td>
<td></td>
<td>11067.0</td>
<td>0.10</td>
<td></td>
<td>10216.0</td>
<td>0.11</td>
<td>0.00</td>
</tr>
<tr>
<td>ste36-1</td>
<td></td>
<td>13476.0</td>
<td>0.22</td>
<td></td>
<td>19548.0</td>
<td>0.30</td>
<td>0.00</td>
</tr>
<tr>
<td>sko42-1</td>
<td></td>
<td>34112.0</td>
<td>8.08</td>
<td></td>
<td>31993.0</td>
<td>13.58</td>
<td>0.00</td>
</tr>
<tr>
<td>sko49-1</td>
<td></td>
<td>50697.0</td>
<td>1.90</td>
<td></td>
<td>51031.0</td>
<td>24:00:00</td>
<td>0.73</td>
</tr>
<tr>
<td>sko56-1</td>
<td></td>
<td>81074.0</td>
<td>4.32</td>
<td></td>
<td>82634.0</td>
<td>24:00:00</td>
<td>4.30</td>
</tr>
<tr>
<td>AKV-60-01</td>
<td></td>
<td>2042653.0</td>
<td>1.59</td>
<td></td>
<td>2625986.0</td>
<td>24:00:00</td>
<td>9.61</td>
</tr>
<tr>
<td>sko64-1</td>
<td></td>
<td>122479.0</td>
<td>11.17</td>
<td></td>
<td>124693.0</td>
<td>24:00:00</td>
<td>6.48</td>
</tr>
<tr>
<td>AKV-70-01</td>
<td></td>
<td>2185508.0</td>
<td>6.33</td>
<td></td>
<td>2835266.0</td>
<td>24:00:00</td>
<td>10.90</td>
</tr>
<tr>
<td>sko72-1</td>
<td></td>
<td>168547.0</td>
<td>12.10</td>
<td></td>
<td>181929.0</td>
<td>24:00:00</td>
<td>8.16</td>
</tr>
<tr>
<td>AKV-75-01</td>
<td></td>
<td>3457598.5</td>
<td>5.74</td>
<td></td>
<td>4304658.5</td>
<td>24:00:00</td>
<td>12.30</td>
</tr>
<tr>
<td>AKV-80-01</td>
<td></td>
<td>2837168.5</td>
<td>6.65</td>
<td></td>
<td>3896864.5</td>
<td>24:00:00</td>
<td>15.30</td>
</tr>
</tbody>
</table>

Table 2: Summary of the results produced by our TS heuristic when applied to DCFLP instances with 15 to 80 machines. The running times are given in seconds. Gap indicates the relative gap between the best lower bound determined by the ILP approach and the best layout found by the TS heuristic.

<table>
<thead>
<tr>
<th>Instance</th>
<th></th>
<th>TS one-way</th>
<th></th>
<th></th>
<th>TS random</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Best</td>
<td></td>
<td>Gap (%)</td>
<td>Best</td>
<td></td>
<td>Gap (%)</td>
</tr>
<tr>
<td>P15</td>
<td></td>
<td>8284.0</td>
<td>0.01</td>
<td>0.00</td>
<td>8041.0</td>
<td>0.01</td>
<td>0.00</td>
</tr>
<tr>
<td>P17</td>
<td></td>
<td>12717.0</td>
<td>0.01</td>
<td>0.00</td>
<td>13105.0</td>
<td>0.01</td>
<td>1.51</td>
</tr>
<tr>
<td>P18</td>
<td></td>
<td>14450.5</td>
<td>0.01</td>
<td>0.00</td>
<td>14726.5</td>
<td>0.01</td>
<td>0.00</td>
</tr>
<tr>
<td>N25_01</td>
<td></td>
<td>6221.0</td>
<td>0.01</td>
<td>0.09</td>
<td>5921.0</td>
<td>2.09</td>
<td>0.00</td>
</tr>
<tr>
<td>N30_01</td>
<td></td>
<td>11067.0</td>
<td>0.10</td>
<td>0.00</td>
<td>10216.0</td>
<td>0.11</td>
<td>0.00</td>
</tr>
<tr>
<td>ste36-1</td>
<td></td>
<td>13476.0</td>
<td>0.22</td>
<td>0.00</td>
<td>19548.0</td>
<td>0.30</td>
<td>0.00</td>
</tr>
<tr>
<td>sko42-1</td>
<td></td>
<td>34112.0</td>
<td>8.08</td>
<td>0.00</td>
<td>31993.0</td>
<td>13.58</td>
<td>0.00</td>
</tr>
<tr>
<td>sko49-1</td>
<td></td>
<td>50697.0</td>
<td>1.90</td>
<td>0.00</td>
<td>51031.0</td>
<td>24:00:00</td>
<td>0.73</td>
</tr>
<tr>
<td>sko56-1</td>
<td></td>
<td>81074.0</td>
<td>4.32</td>
<td>0.00</td>
<td>82634.0</td>
<td>24:00:00</td>
<td>4.30</td>
</tr>
<tr>
<td>AKV-60-01</td>
<td></td>
<td>2042653.0</td>
<td>1.59</td>
<td>0.00</td>
<td>2625986.0</td>
<td>24:00:00</td>
<td>9.61</td>
</tr>
<tr>
<td>sko64-1</td>
<td></td>
<td>122479.0</td>
<td>11.17</td>
<td>0.00</td>
<td>124693.0</td>
<td>24:00:00</td>
<td>6.48</td>
</tr>
<tr>
<td>AKV-70-01</td>
<td></td>
<td>2185508.0</td>
<td>6.33</td>
<td>0.00</td>
<td>2835266.0</td>
<td>24:00:00</td>
<td>10.90</td>
</tr>
<tr>
<td>sko72-1</td>
<td></td>
<td>168547.0</td>
<td>12.10</td>
<td>0.00</td>
<td>181929.0</td>
<td>24:00:00</td>
<td>8.16</td>
</tr>
<tr>
<td>AKV-75-01</td>
<td></td>
<td>3457598.5</td>
<td>5.74</td>
<td>0.00</td>
<td>4304658.5</td>
<td>24:00:00</td>
<td>12.30</td>
</tr>
<tr>
<td>AKV-80-01</td>
<td></td>
<td>2837168.5</td>
<td>6.65</td>
<td>0.00</td>
<td>3896864.5</td>
<td>24:00:00</td>
<td>15.30</td>
</tr>
</tbody>
</table>
Another worthwhile direction of research is to combine and extend the approaches for the DCFLP and the SRFLP to tackle the Combined Cell Layout Problem (CCLP) [11]. The CCLP is concerned with the minimization of the material-handling costs in a cellular manufacturing system with two or more cells in the presence of parts that require processing in more than one cell. The machines of each cell can be arranged in a row or on a circle. Hence the CCLP allows to model more complex layout types, but still builds on well-studied and efficiently solvable basic problems.
References


