CHAPTER 4

ENGINEERING OPTIMIZATION

4.1 INTRODUCTION

Most design optimization problems in engineering are highly non-linear, include many different design variables and complicated constraints on stresses, displacements, load carrying capability, and geometrical configuration. Since non-linearity often leads to multiple local optima only global algorithms should be used (Gandomi et al 2011, Yang 2008). Most nonlinear optimization methods assume that objective function variables are continuous. However, many practical engineering design problems frequently encounter discrete variables as well as continuous variables (Guo et al 2004). In engineering design problem constraints are also very important. They are imposed on the statement of the problems. Sometimes constraints are very hard to satisfy which makes the search difficult and inefficient (Cagnina et al 2008).

Meta-Heuristic methods are quite suitable and powerful for obtaining the solution of constrained non-linear engineering optimization problems. In this chapter, several well-studied engineering design problems taken from the optimization literature are used to show the efficiency of the proposed AWDA. These examples have been previously solved using a variety of other techniques, which is useful to show the validity and effectiveness of the proposed algorithm.
Standard problems in mechanical engineering such as welded beam design, pressure vessel design and spring design are optimized to prove the robustness of the algorithm. An Economic Dispatch Problem (EDP), which is related to the optimum generation scheduling of available generators in an electrical power system, to minimize the total fuel cost while satisfying the load demand and operational constraints is also attempted. A new methodology is proposed to optimize facility layout problems applicable to various branches of engineering. Facility layout optimization problem consists of looking for the best allocation of ‘n’ activities/facilities to m locations, m being greater than or equal to n, where the terms activity and location should be considered in their most general sense.

4.2 PRESSURE VESSEL DESIGN

Pressure vessel is a closed container that holds gases or liquids at a pressure typically significantly higher than the ambient pressure. Pressure vessels can theoretically be almost of any shape, but shapes made of sections of spheres, cylinders, and cones are usually employed. A common design is a cylinder with end hemispherical caps, which are widely used for engineering purposes and this optimization problem was proposed by Sandgren (1990).

The cylindrical pressure vessel capped at both ends by hemispherical heads as shown Figure 4.1, must be designed for minimum cost (Sandgren 1990). The compressed air tank has a working pressure of 3000 psi and a minimum volume of 750 ft³, and must be designed according to the American Society of Mechanical Engineers (ASME) code on boilers and pressure vessels. The total cost results from a combination of welding, material and forming costs. The thickness of the cylindrical skin (T_s), the thickness of the spherical head (T_h), the inner radius (R), and the length of the cylindrical segment of the vessel (L) were included as optimization variables. Thicknesses can only take discrete values which are integer multiples of
0.0625 in. and R and L. have continuous values of \(40 \leq R \leq 80\) in. and \(20 \leq L \leq 60\) in., respectively. The optimization problem can be stated as follows:

Minimize:

\[
f(T_s, T_h, R, L) = 0.6224T_sR + 1.7781T_hR^2 + 3.1661T_s^2L + 19.84T_h^2R \tag{4.1}
\]

Constraints are set in accordance with the ASME design codes; \(g_3\) represents the constraint on the minimum volume of 750 ft\(^3\). The constraints are stated as follows:

\[
g_1 = -T_s + 0.0193R \leq 0 \tag{4.2}
\]

\[
g_2 = -T_h + 0.00954R \leq 0 \tag{4.3}
\]

\[
g_3 = -\pi R^2L - \frac{4}{3}\pi R^3 + (750 \times 1278 \leq 0 \tag{4.4}
\]

\[
g_4 = L - 240 \leq 0 \tag{4.5}
\]

\[
g_5 = 1.1 - T_s \leq 0 \tag{4.6}
\]

\[
g_6 = 0.60 - T_h \leq 0 \tag{4.7}
\]

Figure 4.1 Center and end section of the pressure vessel
This problem was previously analyzed using GA (Wu & Chow 1995a) and HS (Lee & Geem 2005). Optimization results are presented in Table 4.1. With 25 wild dogs, AWDA found the global optimum of 7198.008 within 25,000 function evaluations (i.e. 1000 optimization iterations). Table 4.1 compares the results obtained by AWDA with those reported in the literature.

Table 4.1 Optimal results for the pressure vessel design

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>T_s</td>
<td>1.125</td>
<td>1.125</td>
<td>1.1250</td>
<td>1.1250</td>
</tr>
<tr>
<td>T_h</td>
<td>0.625</td>
<td>0.625</td>
<td>0.6250</td>
<td>0.6250</td>
</tr>
<tr>
<td>R</td>
<td>48.97</td>
<td>58.1978</td>
<td>58.2789</td>
<td>58.2901</td>
</tr>
<tr>
<td>L</td>
<td>106.72</td>
<td>44.293</td>
<td>43.7549</td>
<td>43.6928</td>
</tr>
<tr>
<td>Cost</td>
<td>7980.894</td>
<td>7207.494</td>
<td>7198.4330</td>
<td>7198.0080</td>
</tr>
</tbody>
</table>

Sandgren (1990) achieved the optimal values of $7980.894 using the branch and bound method. Wu and Chow obtained the minimum cost of $7207.494 using the GA-based approach. The best solution obtained by HS algorithm was $7207.494. The AWD algorithm achieves a design with a best solution vector of (1.125, 0.625, 58.2901 and 43.6928) and a minimum cost of $7198.008 without violating any constraint. The results obtained using the AWD algorithms are comparable with earlier solutions.

4.3 OPTIMIZATION OF WEIGHT OF SPRING:

According to Wahl (1963) "A mechanical spring may be defined as an elastic body whose primary function is to deflect or distort under load
(or to absorb energy) and which recovers its original shape when released after being distorted”. The main functions of springs are

- to absorb shock
- to apply force
- to support a structure
- to provide load control

The tension/compression springs as shown in Figure 4.2 are by far the most common type and are useful in the operation of many devices due to several desirable properties, such as a near linear rate (particularly after the first 20% of deflection), the materials that can be used to make them, and the ease of manufacture (Porteiro 2010).

![Figure 4.2 Tension / compression springs](image)

The problem of optimization of spring consists of minimizing the weight $f(x)$ of a tension/compression spring subject to constraints on shear stress, surge frequency and minimum deflection as shown in Figure 4.3.
Figure 4.3 Tension/compression spring to be optimised

The design variables are the mean coil diameter $D (= x_1)$; the wire diameter $d (= x_2)$; and the number of active coils $N_c (= x_3)$. The problem can be stated as (Belegundu 1982):

Minimize:

$$f(x) = (x_3 + 2)x_2x_1^2$$  \hspace{1cm} (4.8)

Subject to:

$$g_1 = 1 - \frac{x_2^3x_3}{71785x_1^4} \leq 0, \hspace{1cm} (4.9)$$

$$g_2 = \frac{4x_2^2 - x_1x_2}{12566(x_2x_1^3 - x_1^4)} + \frac{1}{5108x_1^2} - 1 \leq 0, \hspace{1cm} (4.10)$$

$$g_3 = 1 - \frac{140.45x_1}{x_2^2x_3} \leq 0, \hspace{1cm} (4.11)$$

$$g_4 = \frac{x_2 + x_1}{1.5} - 1 \leq 0, \hspace{1cm} (4.12)$$

using a numerical optimization technique called constraint correction at constant cost (Arora 1989). This problem was also analyzed using GA-based method (Coello 2000a) and improved harmony search algorithm (Mahadavi et al 2007). After 30,000 function evaluations the best solution is obtained at $x = (0.0516558; 0.3559185; 11.336039)$ with corresponding function value equal to $f(x) = 0.0126653$. No constraints are active for this solution.

Table 4.2 presents the best solution of this problem obtained using the AWD algorithm and compares the AWDA results with solutions reported by other researchers. It is apparent from the Table 4.2 that the result obtained using AWDA algorithm is better than those reported previously in the literature.

### Table 4.2 Optimal results for the minimization of weight of spring

<table>
<thead>
<tr>
<th>Optimal design variables ($x$) and objective function value ($f(x)$)</th>
<th>Numerical optimization technique (Arora 1989)</th>
<th>Mathematical optimization techniques (Belegundu 1982)</th>
<th>GA-Based Method (Coello 2000a)</th>
<th>Improved harmony search algorithm (Mahadavi et al 2007)</th>
<th>Proposed AWDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>0.053396</td>
<td>0.050000</td>
<td>0.051989</td>
<td>0.05115438</td>
<td>0.05165583</td>
</tr>
<tr>
<td>$x_2$</td>
<td>0.399180</td>
<td>0.315900</td>
<td>0.363965</td>
<td>0.34987116</td>
<td>0.35591847</td>
</tr>
<tr>
<td>$f(x)$</td>
<td>0.012730</td>
<td>0.012833</td>
<td>0.012681</td>
<td>0.01267060</td>
<td>0.01266531</td>
</tr>
</tbody>
</table>

### 4.4 DESIGN OPTIMIZATION OF WELDED BEAM

The problem is a simplified example of many complex design issues arising in structural engineering, dealing with designing the form of steel beams and with connecting them to form complex structure like bridges, buildings, etc. The problem of designing an optimal welded beam consists of dimensioning a welded steel beam and the welding length so as to minimize its cost subject to constraints on shear stress, bending stress in the beam, the
buckling load on the bar, the end the deflection of the beam, and side constraints.

The welded beam structure, shown in Figure 4.4, is a practical design problem that has been often used as a benchmark for testing different optimization methods. This includes mathematical optimization algorithms (Ragstell & Phillips 1976) such as APPROX (Griffith and Stewart’s successive linear approximation), DAVID (Davidon–Fletcher–Powell with a penalty function), SIMPLEX (Simplex method with a penalty function), and RANDOM (Richardson’s random method) algorithms. GA-based methods (Deb 1991, Deb 2000, Coello 2000a, Coello 2000b), harmony search method (Lee & Geem 2005), and improved harmony search algorithm (Mahadavi et al 2007) were other methods used to solve this problem.

![Welded beam structure](image)

**Figure 4.4 Welded beam structure**

The objective is to find the minimum fabricating cost of the welded beam subject to constraints on shear stress $\tau$, bending stress $\sigma$, buckling load $P_c$, end deflection $\delta$ and side constraint. There are four design variables: $h(=x_1), l(=x_2), t(=x_3)$ and $b(=x_4)$. The mathematical formulation of the
objective function \( f(x) \), which is the total fabricating cost mainly comprised of the set-up, welding labour, and material costs, is as follows (Ragstell & Phillips 1976):

Minimize

\[
\begin{align}
  f(x) &= 1.10471x_1^2x_2 + 0.04811x_3x_4 (14.0 + x_2) \\
  \text{Subject to} & \\
  g_1(x) &= \tau(x) - \tau_{\text{max}} \leq 0, \\
  g_2(x) &= \sigma(x) - \sigma_{\text{max}} \leq 0, \\
  g_3(x) &= x_1 - x_4 \leq 0, \\
  g_4(x) &= \delta(x_1) - \delta_{\text{max}} \leq 0, \\
  g_5(x) &= P - P_c(x) \leq 0, \\
  0.125 &\leq 0, \\
  0.1 &\leq x_2, x_3 \leq 10, \\
  0.1 &\leq x_4 \leq 5,
\end{align}
\]

where

\[
\begin{align}
  \tau(x) &= \sqrt{(\tau')^2 + 2\tau'\tau''\frac{x_2}{x_2} + (\tau'')^2} \\
  \tau' &= \frac{p}{\sqrt{2x_1x_2}} \\
  \tau'' &= \frac{MR}{J}, \\
  M &= P(L + \frac{x_2}{2}), \\
  R &= \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_2}{2}\right)^2}, \\
  J &= \left\{2\sqrt{2x_1x_2} \left[\frac{x_2^2}{4} + \left(\frac{x_1 + x_2}{2}\right)^2\right]\right\},
\end{align}
\]
\[ \sigma(x) = \frac{6P_L}{x_4x_5^2}, \quad (4.26) \]

\[ \delta(x) = \frac{6P_L}{x_3^4x_4}, \quad (4.27) \]

\[ P_c(x) = \frac{4.013E\sqrt[3]{x_3^2x_4^2/36}}{L^2} \left(1 - \frac{x_3}{2L} \sqrt[3]{\frac{E}{4G}}\right) \quad (4.28) \]

where,

\[ P = 6000 \text{ lb}, \]
\[ L = 14 \text{ in.}, \]
\[ \delta_{max} = 0.25 \text{ in.}, \]
\[ E = 30 \times 10^6 \text{ psi}, \]
\[ G = 12 \times 10^6 \text{ psi}, \]
\[ \tau_{max} = 13,600 \text{ psi}, \]
\[ \sigma_{max} = 30,000 \text{ psi} \]

Deb (2000) and Coello (2000a, 2000b) solved this problem using GA-based methods. Radgisdell & Phillips (1976) compared optimal results of different optimization methods that were mainly based on mathematical optimization algorithms. These algorithms, are APPROX (Griffith and Stewart’s successive linear approximation), DAVID (Davidon–Fletcher–Powell with a penalty function), SIMPLEX (Simplex method with a penalty function), and RANDOM (Richardson’s random method) algorithms. Lee & Geem also solved this problem using HS method. The AWDA is applied for this constrained optimization problem. The comparison of results, are shown in Table 4.3. The AWDA obtained optimum result after 150,000 function evaluations. AWDA is better than the reported results except improved harmony search algorithm.
### Table 4.3 Optimal results for welded beam design

<table>
<thead>
<tr>
<th>Methods</th>
<th>Optimal design variables (x)</th>
<th>Objective function value f(x)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>h</td>
<td>l</td>
</tr>
<tr>
<td>Coello (2000a)</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Coello (2000b)</td>
<td>0.2088</td>
<td>3.4205</td>
</tr>
<tr>
<td>Deb (1991)</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Deb (2000)</td>
<td>0.2489</td>
<td>6.173</td>
</tr>
<tr>
<td>Ragsdell &amp; Phillips (1976)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>APPROX</td>
<td>0.2444</td>
<td>6.2189</td>
</tr>
<tr>
<td>DAVID</td>
<td>0.2434</td>
<td>6.2552</td>
</tr>
<tr>
<td>RANDOM</td>
<td>0.4575</td>
<td>4.7313</td>
</tr>
<tr>
<td>SIMPLEX</td>
<td>0.2792</td>
<td>5.6256</td>
</tr>
<tr>
<td>Lee &amp; Geem (2005)</td>
<td>0.2442</td>
<td>6.2231</td>
</tr>
<tr>
<td>Mahadavi et al (2007)</td>
<td>0.20573</td>
<td>3.47049</td>
</tr>
<tr>
<td>Proposed AWDA</td>
<td>0.2057296</td>
<td>3.4704889</td>
</tr>
</tbody>
</table>

### 4.5 ECONOMIC LOAD DISPATCH PROBLEM

Electrical power industry restructuring has created highly vibrant and competitive market that altered many aspects of the power industry. In this changed scenario, scarcity of energy resources, increasing power generation cost, environment concern, ever growing demand for electrical energy necessitate optimal economic dispatch (Mahor et al 2009).

The objective of Economic Dispatch (ED) is to schedule the outputs of the online generating units so that the fuel cost of generation can be minimized, while simultaneously satisfying all unit and system equality and
inequality constraints (Chen et al 2011). This can be effectively achieved with help of economic dispatch function. The improvement in economic dispatch function can save several million dollars that are being lost in the fossil fuel cost every year. EDP plays an important role in operation planning and control of modern power systems (Coelho & Mariani 2009, Park et al 2006, Park et al 2010).

Over the past few years, a number of approaches have been developed for solving the EDP using classical mathematical programming methods. Meanwhile, classical optimization methods are highly sensitive to starting points and frequently converge to local optimum solution or diverge altogether. Linear programming methods are fast and reliable but the main disadvantage associated with the piecewise linear cost approximation. Nonlinear programming methods have a problem of convergence and algorithmic complexity. Newton based algorithm have a problem in handling large number of inequality constraints (Coelho & Mariani 2009).

Because of physical limitations of the power generators, a generating unit may have prohibited operating zones between the minimum and maximum power outputs. Generators that operate in these zones may experience amplification of vibrations in their shaft bearings, which should be avoided in practical application. On the other hand, due to the fact that unit generation output cannot be changed instantaneously, the unit in the actual operating processes is restricted by its ramp rate limit. Moreover, the units of real input–output characteristics include higher order nonlinearities and discontinuities owing to the valve point effect. The ED problem with the above considerations is usually a non-smooth/non-convex optimization problem (Wang et al 2007). This kind of optimization problem is very hard to solve using traditionally deterministic optimization algorithms. The proposed meta-heuristic approach African Wild Dog Algorithm (AWDA), is validated here by solving the Economic dispatch problem.
4.5.1 Formulation of Economic Load Dispatch Problem

The main objective of solving the ED problem is to minimize the total generation cost of a power system fulfilling the load demand while satisfying various constraints. The objective function can be formulated as follows (Chen 2011):

\[
\text{Minimize } F_t = \sum_{i=1}^{N_u} F_i(P_i)
\]  

(4.29)

where

\( F_t \) Total fuel cost,

\( Nu \) number of units in the system

\( F_i(P_i) \) fuel cost function of unit i,

\( P_i \) Power output of unit i.

Generally, fuel cost of generation unit will be in second-order polynomial function.

\[
F_i(P_i) = a_i + b_i P_i + c_i P_i^2
\]  

(4.30)

where

\( a_i, b_i, c_i \) cost coefficients of generation unit i.

However, the fuel cost functions of units may be much more complicated due to the physical operation limitations, which actually exist in a practical optimization problem. The generating units with multi-valve steam turbines exhibit a greater variation in the fuel-cost functions. Since the valve point results in the ripples as shown in Figure 4.5, a cost function contains higher order nonlinearity.
Therefore, the Equation (4.30) should be replaced as the Equation (4.31) to consider the valve-point effects. The fuel cost functions taking into account the valve-point effects were expressed as

\[ F_i(P_i) = a_i + b_i P_i + c_i P_i^2 + |e_i \sin(f_i(P_i^{\min} - P_i))| \]  

(4.31)

where \( e_i, f_i \) are constants from the valve-point loading effect of generators.

Subject to following constraints:

**Power balance constraint:** For power balance, an equality constraint should be satisfied. The total generated power should be the same as total load demand plus the total line loss

\[ \sum_{i=1}^{N_{hi}} P_i = P_D + P_{Loss} \]  

(4.32)
where

\[ P_D \quad \text{Total load demand,} \]

\[ P_{Loss} \quad \text{Transmission loss} \]

For simplification purpose, the transmission loss need not be considered.

**Unit capacity constraints:** Generation output of each generator should lie between maximum and minimum limits. The corresponding inequality constraints for each generator are

\[ P_i^{min} \leq P_i \leq P_i^{max} \quad (4.33) \]

where

\[ P_i^{min} \quad \text{Minimum generation limits of unit } i \]

\[ P_i^{max} \quad \text{Maximum generation limits of unit } i \]

### 4.5.2 Optimization of 13 Unit Generator System

An EDP based on a 13-unit test system with incremental fuel cost function taking into account the valve-point loading effects is employed to demonstrate the performance of the AWDA. The feasibility and effectiveness of the proposed algorithm is thus verified. There are many local optimal solutions for this dispatch problem due to valve-point loading effects and the problem is well suitable for testing and validating the developed algorithm. The system unit data is given in Table 4.4 and the load demand is 2520 MW. For comparison purpose the network losses of the system are neglected. The same multiple minimum problem has been solved by the Hybrid Stochastic Search (HSS) (Bhagwandas & Patvarthan 1999), Tabu Search Algorithm (TS)

Table 4.4 Parameters for the thirteen-unit system

<table>
<thead>
<tr>
<th>Unit No</th>
<th>$p_{i}^{max}$</th>
<th>$p_{i}^{min}$</th>
<th>$a_i$</th>
<th>$b_i$</th>
<th>$c_i$</th>
<th>$e_i$</th>
<th>$f_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>680</td>
<td>0</td>
<td>550</td>
<td>8.1</td>
<td>0.00028</td>
<td>300</td>
<td>0.035</td>
</tr>
<tr>
<td>2</td>
<td>360</td>
<td>0</td>
<td>309</td>
<td>8.1</td>
<td>0.00056</td>
<td>200</td>
<td>0.042</td>
</tr>
<tr>
<td>3</td>
<td>360</td>
<td>0</td>
<td>307</td>
<td>8.1</td>
<td>0.00056</td>
<td>200</td>
<td>0.042</td>
</tr>
<tr>
<td>4</td>
<td>180</td>
<td>60</td>
<td>240</td>
<td>7.74</td>
<td>0.00324</td>
<td>150</td>
<td>0.063</td>
</tr>
<tr>
<td>5</td>
<td>180</td>
<td>60</td>
<td>240</td>
<td>7.74</td>
<td>0.00324</td>
<td>150</td>
<td>0.063</td>
</tr>
<tr>
<td>6</td>
<td>180</td>
<td>60</td>
<td>240</td>
<td>7.74</td>
<td>0.00324</td>
<td>150</td>
<td>0.063</td>
</tr>
<tr>
<td>7</td>
<td>180</td>
<td>60</td>
<td>240</td>
<td>7.74</td>
<td>0.00324</td>
<td>150</td>
<td>0.063</td>
</tr>
<tr>
<td>8</td>
<td>180</td>
<td>60</td>
<td>240</td>
<td>7.74</td>
<td>0.00324</td>
<td>150</td>
<td>0.063</td>
</tr>
<tr>
<td>9</td>
<td>180</td>
<td>60</td>
<td>240</td>
<td>7.74</td>
<td>0.00324</td>
<td>150</td>
<td>0.063</td>
</tr>
<tr>
<td>10</td>
<td>120</td>
<td>40</td>
<td>126</td>
<td>8.6</td>
<td>0.00284</td>
<td>100</td>
<td>0.084</td>
</tr>
<tr>
<td>11</td>
<td>120</td>
<td>40</td>
<td>126</td>
<td>8.6</td>
<td>0.00284</td>
<td>100</td>
<td>0.084</td>
</tr>
<tr>
<td>12</td>
<td>120</td>
<td>55</td>
<td>126</td>
<td>8.6</td>
<td>0.00284</td>
<td>100</td>
<td>0.084</td>
</tr>
<tr>
<td>13</td>
<td>120</td>
<td>55</td>
<td>126</td>
<td>8.6</td>
<td>0.00284</td>
<td>100</td>
<td>0.084</td>
</tr>
</tbody>
</table>

The proposed AWD algorithm is applied to solve this 13 unit generator problem to find its performance on non-smooth objective functions. The parameters of the 13 unit problem are tabulated in Table 4.4. Boundary limit strategy, as newly proposed in this research and discussed in chapter 3, is used for handling the constraint of upper and lower bound of variables. Penalty function approach is used for other constraints.
Numerical results obtained with the proposed AWDA approach were compared with other optimization results reported in literature in Table 4.5. The optimum cost of $ 24164.10 achieved using AWDA is better than the results achieved with Hybrid stochastic search (HSS) (Bhagwandas & Patvarthan 1999), Tabu search algorithm (TS) (Khamsawang et al 2002), Hybrid Evolutionary Programming – Sequential quadratic Programming (EP-SQP) (Victoire & Jeyakumar 2004), Particle Swarm Optimization – Sequential quadratic Programming (PSO-SQP) (Victoire & Jeyakumar 2004) and Particle swarm optimization with improved inertia weight (PSO-IIW)(Chen et al 2011).

Table 4.5  Comparison of dispatch results for the load of 2520 MW in the system

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>628.23</td>
<td>628.319</td>
<td>628.3136</td>
<td>628.3205</td>
<td>628.3185</td>
<td>628.3175</td>
</tr>
<tr>
<td>3</td>
<td>299.17</td>
<td>331.8975</td>
<td>299.0474</td>
<td>298.9681</td>
<td>299.1990</td>
<td>294.5105</td>
</tr>
<tr>
<td>4</td>
<td>159.12</td>
<td>159.7305</td>
<td>159.6399</td>
<td>159.4680</td>
<td>159.7330</td>
<td>159.7319</td>
</tr>
<tr>
<td>5</td>
<td>159.95</td>
<td>159.7331</td>
<td>159.6560</td>
<td>159.1429</td>
<td>159.7330</td>
<td>159.7319</td>
</tr>
<tr>
<td>6</td>
<td>158.85</td>
<td>159.7306</td>
<td>158.4831</td>
<td>159.2724</td>
<td>159.7328</td>
<td>159.7320</td>
</tr>
<tr>
<td>7</td>
<td>157.26</td>
<td>159.7334</td>
<td>159.6749</td>
<td>159.5371</td>
<td>159.7328</td>
<td>159.7320</td>
</tr>
<tr>
<td>8</td>
<td>159.93</td>
<td>159.7308</td>
<td>159.7265</td>
<td>158.8522</td>
<td>159.7329</td>
<td>159.7311</td>
</tr>
<tr>
<td>9</td>
<td>159.86</td>
<td>159.7316</td>
<td>159.6653</td>
<td>159.7845</td>
<td>159.7329</td>
<td>159.7327</td>
</tr>
<tr>
<td>10</td>
<td>110.78</td>
<td>40.0028</td>
<td>114.0334</td>
<td>110.9618</td>
<td>77.3996</td>
<td>77.3978</td>
</tr>
<tr>
<td>11</td>
<td>75.00</td>
<td>77.3994</td>
<td>75.0000</td>
<td>75.0000</td>
<td>77.3996</td>
<td>77.3905</td>
</tr>
<tr>
<td>12</td>
<td>60.00</td>
<td>92.3932</td>
<td>60.0000</td>
<td>60.0000</td>
<td>92.3998</td>
<td>92.3993</td>
</tr>
<tr>
<td>13</td>
<td>92.62</td>
<td>92.3986</td>
<td>87.5884</td>
<td>91.6401</td>
<td>87.6868</td>
<td>92.3971</td>
</tr>
<tr>
<td>Cost ($/h)</td>
<td>24275.71</td>
<td>24313</td>
<td>24266.44</td>
<td>24261.05</td>
<td>24169.05</td>
<td>24164.10</td>
</tr>
</tbody>
</table>
4.6 FACILITY LAYOUT PROBLEM

A facility is an entity that facilitates the performance of any job. It may be a machine tool, a work centre, a manufacturing cell, a machine shop, a department, a warehouse, etc (Drira et al 2007). Facility Layout Problem (FLP) is concerned with finding feasible locations for a set of interrelated objects that meet all design requirements and maximize design quality in terms of design preferences such as minimizing the total cost associated with the interactions between these facilities (Yeh 1995, Yeh 2006). These costs may reflect transportation costs (including costs associated with the construction of a material handling system), or preferences regarding adjacencies among departments (Tate & Smith 1995).

The optimum layout of precast yards and Construction Site Layout Planning (CSLP) are examples of FLP in civil engineering construction industry. The layout of site-level facilities can have an important impact on the production time and cost, especially in the case of large projects. In addition, such a problem becomes far from trivial if a construction site is confined due to the lack of available space, or if the site is very large, then traveling between facilities can be considerably time consuming. Therefore, an effective CSLP is utmost important for the success of a construction project (Ning et al 2010). The good placement of facilities contributes to the overall efficiency of operations and can reduce until 50% of the total operating expenses (Tompkins et al 1996). Since the early 1960s, the FLP has been analyzed extensively in the engineering field. However, due to the unique nature of construction sites, models have been developed specifically for the construction domain (Elbeltagi & Hegazy 2001, Elbeltagi et al 2004).

Arranging a set of predetermined facilities into appropriate locations, while satisfying a set of layout constraints, is a difficult problem as
there are many possible alternatives. For example, consider a problem with 10 facilities to be arranged in 10 locations. The number of possible alternative solutions to this problem will be 10! (i.e., 36,288,000).

Early models were based solely on mathematical optimization techniques and, due to the complexity of problem formulation, were successful in laying out only a limited number of facilities. Heuristic approaches and knowledge-based systems also have been used to solve larger size problems of site layout. In general, heuristic solutions attempt to satisfy spatial relationships among facilities and have been reported to produce good but not optimal solutions. Due to the complexity of the site layout problem, non-traditional optimization techniques based on Artificial Intelligence (AI) have been applied to solve the problem. Some other researchers used the powerful evolutionary algorithms to solve optimization problems (Elbeltagi & Hegazy 2001).

4.6.1 Layout Representation

A number of representation schemes such as binary strings, permutation representation, and ordinal representation have been experimented for combinatorial optimization problems such as facility layout problems. In permutation type the representation, each facility layout can be represented by an $n \times n$ permutation matrix ($n$ is the number of facilities, or locations) in which rows and columns are labeled by facilities and locations, respectively. There is only one ‘1’ in each row and column, and the rest of the elements are 0. The corresponding row and column number of ‘1’ indicate the location where the facility is placed. The Table 4.6 shows a permutation matrix with 11 facilities and locations (Li & Love 2000).
Table 4.6 Permutation matrix with 11 facilities

<table>
<thead>
<tr>
<th>Facility</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 2 3 4 5 6 7 8 9 10 11</td>
</tr>
<tr>
<td>1</td>
<td>0 0 0 0 1 0 0 0 0 0 0</td>
</tr>
<tr>
<td>2</td>
<td>0 0 1 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>3</td>
<td>0 0 0 0 0 0 0 1 0 0 0</td>
</tr>
<tr>
<td>4</td>
<td>0 0 0 0 0 0 1 0 0 0 0</td>
</tr>
<tr>
<td>5</td>
<td>0 0 0 1 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>6</td>
<td>0 1 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>7</td>
<td>0 0 0 0 0 0 0 0 0 0 1</td>
</tr>
<tr>
<td>8</td>
<td>1 0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>9</td>
<td>0 0 0 0 0 0 1 0 0 0 0</td>
</tr>
<tr>
<td>10</td>
<td>0 0 0 0 0 0 0 0 1 0 0</td>
</tr>
<tr>
<td>11</td>
<td>0 0 0 0 0 0 0 0 0 1 0</td>
</tr>
</tbody>
</table>

The permutation matrix can also be represented in a string form, as shown in the Table 4.7. In the string representation, the position of a cell represents the facility number, and the value of the cell represents the location of the facility. For example, the location of facility 3 is 8, thus 8 is placed under facility 3 in Table 4.7.

Table 4.7 String layout representation

<table>
<thead>
<tr>
<th>Facility</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 2 3 4 5 6 7 8 9 10 11</td>
</tr>
<tr>
<td>1</td>
<td>5 3 8 7 4 2 11 1 6 9 10</td>
</tr>
<tr>
<td>2</td>
<td>5 3 8 7 4 2 11 1 6 9 10</td>
</tr>
</tbody>
</table>

Samarghandhi (2010) used a new idea of mapping the set of all permutations into a set of factoradic base numbers with a one-to-one mapping and efficiently exploring the generated set is used to find a good quality solution for FLP problem. Finally the permutation will be returned as the final solution for the FLP.
4.6.1.1 Factoradic base

Factoradic is a specially constructed number system. Factoradics provide a lexicographical index for permutations (Knuth 1997). The idea of the factoradic is closely linked to that of the Lehmer code (Knuth 1997). Factoradic is a factorial-based mixed radix numeral system: the ith digit from right side is to be multiplied by i!. In this numbering system, the rightmost digit is always 0, the second 0, or 1, the third 0, 1 or 2 and so on (Knuth 1997). For instance, 38 in decimal base can be shown as \((1_42_31_20_10_0)\) in factoradic base.

\[ (1_42_31_20_10_0) = 1 \times 4! + 2 \times 3! + 1 \times 2! = 38_{10} \]  

(4.34)

The factoradic numbering system is unambiguous. No number can be represented in more than one way because the sum of consecutive factorials multiplied by their index is always the next factorial minus one (Knuth 1997):

\[ \sum_{i=0}^{n} i \times i! = (n + 1)! - 1 \]  

(4.35)

4.6.1.2 Relation between factoradic base and permutations

There is a natural mapping between the integers 0; 1; \ldots; n!-1 (or equivalently the factoradic numbers with n digits) and the permutations of n elements in lexicographical order, when the integers are expressed in factoradic form. This mapping has been termed the Lehmer code. For example, with n = 3, this mapping is shown in Table 4.8.
Table 4.8: Natural mapping between factoradic numbers and permutations when $n=3$

<table>
<thead>
<tr>
<th>Decimal</th>
<th>Factoradic</th>
<th>Permutation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0_{10}$</td>
<td>$0_{2}0_{1}0_{0}$</td>
<td>1,2,3</td>
</tr>
<tr>
<td>$1_{10}$</td>
<td>$0_{2}1_{1}0_{0}$</td>
<td>1,3,2</td>
</tr>
<tr>
<td>$2_{10}$</td>
<td>$1_{2}0_{1}0_{0}$</td>
<td>2,1,3</td>
</tr>
<tr>
<td>$3_{10}$</td>
<td>$1_{2}1_{1}0_{0}$</td>
<td>2,3,1</td>
</tr>
<tr>
<td>$4_{10}$</td>
<td>$2_{2}0_{1}0_{0}$</td>
<td>3,1,2</td>
</tr>
<tr>
<td>$5_{10}$</td>
<td>$2_{2}1_{1}0_{0}$</td>
<td>3,2,1</td>
</tr>
</tbody>
</table>

For mapping factoradic numbers into permutations and vice versa, two straightforward algorithms are presented in McCaffrey (2003) and further used in Behroozi (2009). Computational complexity of these algorithms are $O(n)$ which makes the algorithms able to efficiently map the permutations to factoradic numbers, factoradic numbers to decimal numbers and vice versa. These two algorithms are presented as Algorithm 1 and Algorithm 2.

**Algorithm 1:** Mapping factoradic to permutation.

**Step 1:** Consider a list of possible digits of factoradic base in ascending order $f = \{0,1,\ldots,n-1\}$, as well as a list of possible numbers in permutation $p = \{1,2,\ldots,n\}$

**Step 2:** For $k = 1,2,\ldots,n$ choose the $k$th digit in factoradic representation, and find this digit in $f$. Suppose that this digit is the $i$th digit in $f$. Choose the $i$th element of $p$, set this element as the $k$th element of permutation, and remove this element from $p$.

The Table 4.9 demonstrates the mapping of $(1_{2}1_{1}0_{0}) \rightarrow (2 - 3 - 1)$ based on Algorithm 1.
Table 4.9 Mapping \((1_21_10_0) \rightarrow (2 - 3 - 1)\) based on Algorithm 1

<table>
<thead>
<tr>
<th>Iteration</th>
<th>(k = 1)</th>
<th>(k = 2)</th>
<th>(k = 3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factoradic</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>(F)</td>
<td>{0, 1, 2}</td>
<td>{0, 1, 2}</td>
<td>{0, 1, 2}</td>
</tr>
<tr>
<td>(P)</td>
<td>{1, 2, 3}</td>
<td>{1, 3}</td>
<td>{1}</td>
</tr>
<tr>
<td>Permutation</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

Algorithm 2: Mapping permutation to factoradic.

**Step 1:** Consider a list of possible digits of factoradic base in ascending order \(f = \{0, 1, \ldots, n - 1\}\), as well as a list of possible numbers in permutation \(p = \{1, 2, \ldots, n\}\).

**Step 2:** For \(k = 1, 2, \ldots, n\), choose the \(k\)th digit in permutation, and find this digit in \(p\). Suppose that this digit is the \(i\)th digit in \(p\). Choose the \(i\)th element of \(f\), set this element as the \(k\)th element of factoradic, and remove this element from \(p\).

The mapping of \((2 - 3 - 1) \rightarrow (1_21_10_0)\) based on Algorithm 2 is demonstrated in Table 4.10 (Samarghandi et al. 2010).

Table 4.10 Mapping \((2 - 3 - 1) \rightarrow (1_21_10_0)\) based on Algorithm 2

<table>
<thead>
<tr>
<th>Iteration</th>
<th>(k = 1)</th>
<th>(k = 2)</th>
<th>(k = 3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Permutation</td>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>(P)</td>
<td>{1, 2, 3}</td>
<td>{1, 3}</td>
<td>{1}</td>
</tr>
<tr>
<td>(F)</td>
<td>{0, 1, 2}</td>
<td>{0, 1, 2}</td>
<td>{0, 1, 2}</td>
</tr>
<tr>
<td>Factoradic</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
4.6.2 Optimization of Facility Layout Problems using AWDA

The AWDA developed for constrained optimization problem is slightly modified and an intensification procedure is also added to apply AWDA for FLP problems. Initial solutions which are integer numbers between in the interval 1 and n! are generated in AWDA. Then each number is mapped to respective permutation and the objective function value is calculated. Similarly after end of each iteration, the solutions are rounded to nearest integers. Boundary limit strategy is used to handle the variable upper and lower bound constraints.

Intensification is the search in the neighborhood of the good-quality solution for constructing better solution closer to optimum. There are numerous intensification algorithm developed in the literature; namely, pairwise exchange algorithm families (Francis et al 1998, Kusiak & Heragu 1987). In this study, the exhaustive intensification procedure used by Samarghandhi et al (2010) is adopted. It is more exhaustive and searches the current neighbourhood of all of the solutions once triggered. Intensification is performed on all solutions after a iterations 2a iterations and so on, where a is a parameter set by the user. The proposed algorithm uses a straightforward procedure as its intensification sub-algorithm. First integer numbers- or particles – are mapped into their respective permutation. Then the algorithm exchanges the location of the first two adjacent facilities and evaluates the objective function value of the new permutation. If the objective function of the new permutation has improved by the exchange, algorithm accepts this exchange and restarts the exchange sub-procedure. Nevertheless, if this exchange does not improve the objective function of the solution, the two facilities will move back to their original location and exchange will be applied to the next two adjacent facilities (Samarghandhi et al 2010).
The three benchmark problems from literature are considered to validate the efficiency of AWDA in solving FLP problems.

4.6.3 Layout Planning of a Site Precast Yard

In this model, there are equal number of facilities and locations under settlement. This benchmark problem has already been solved by many researchers using Genetic algorithm (Cheunget al 2002), Tabu search (Liang & Chao 2008) and Mixed integer programming (Wong et al 2010).

There are 11 facilities located to 11 locations in this example. The 11 facilities are as follows:

1. Main Gate
2. Side Gate
3. Batching Plant
4. Steel Storage Yard
5. Formwork Storage Yard
6. Bending Yard
7. Cement/ Sand/ Aggregate Storage Yard
8. Curing Yard
9. Refuse Dumping Area
10. Casting Yard
11. Lifting Yard

The location coordinates of the above facilities indicating their relative positions are indicated in Table 4.11 (Liang & Chao 2008).
Table 4.1: Location coordinates of layout planning of a site precast yard

<table>
<thead>
<tr>
<th>Location no</th>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>15</td>
<td>40</td>
</tr>
<tr>
<td>2</td>
<td>13</td>
<td>30</td>
</tr>
<tr>
<td>3</td>
<td>22</td>
<td>30</td>
</tr>
<tr>
<td>4</td>
<td>25</td>
<td>20</td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>12</td>
<td>10</td>
</tr>
<tr>
<td>7</td>
<td>40</td>
<td>10</td>
</tr>
<tr>
<td>8</td>
<td>48</td>
<td>20</td>
</tr>
<tr>
<td>9</td>
<td>48</td>
<td>35</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>20</td>
</tr>
<tr>
<td>11</td>
<td>32</td>
<td>42</td>
</tr>
</tbody>
</table>

Their frequencies of trips made between facilities were based on the type of resources as listed in the following:

(a) Aggregate, sand and cement, $F_A(C_{mk} = 5)$

\[
F_A = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 20 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 15 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 35 & 0 & 0 & 35 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
20 & 15 & 35 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 35 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]
(b) Reinforcement, $F_{R}(C_{MK} = 4)$

$$F_{R} = \begin{bmatrix}
0 & 0 & 0 & 30 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 20 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
30 & 20 & 0 & 0 & 0 & 50 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 50 & 0 & 0 & 0 & 0 & 0 & 50 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 50 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}$$

(c) Formwork, $F_{f}(C_{MK} = 8)$

$$F_{f} = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 48 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 48 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}$$

(d) Complete precast units, $F_{c}(C_{MK} = 8.5)$

$$F_{c} = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 28 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 20 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 48 & 48 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 48 & 0 & 0 & 0 \\
28 & 20 & 0 & 0 & 0 & 0 & 0 & 48 & 0 & 0 & 0 \\
\end{bmatrix}$$
The distances of the 11 locations are in meter unit.

\[
\begin{array}{ccccccccccc}
0 & 12 & 17 & 30 & 35 & 33 & 55 & 53 & 38 & 30 & 19 \\
12 & 0 & 9 & 22 & 27 & 21 & 47 & 45 & 40 & 18 & 31 \\
17 & 9 & 0 & 13 & 22 & 30 & 38 & 36 & 31 & 27 & 22 \\
30 & 22 & 13 & 0 & 15 & 23 & 25 & 23 & 38 & 20 & 29 \\
35 & 27 & 22 & 15 & 0 & 8 & 20 & 38 & 53 & 25 & 44 \\
33 & 21 & 30 & 23 & 8 & 0 & 28 & 46 & 61 & 17 & 52 \\
55 & 47 & 38 & 25 & 20 & 28 & 0 & 18 & 33 & 45 & 40 \\
53 & 45 & 36 & 23 & 38 & 46 & 18 & 0 & 15 & 43 & 38 \\
38 & 40 & 31 & 38 & 53 & 61 & 33 & 15 & 0 & 58 & 23 \\
30 & 18 & 27 & 20 & 25 & 17 & 45 & 43 & 58 & 0 & 49 \\
19 & 31 & 22 & 29 & 44 & 52 & 40 & 38 & 23 & 49 & 0 \\
\end{array}
\]

The total transportation cost is defined (Liang & Chao 2008) as follows:

\[
\text{Total Cost} = \text{minimize} \left( \sum_{k=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} TLC_{MK_i,j} \right) 
\]

Subject to

\[
TCL_{MK_i,j} = M_{LM_{ij}} \times C_{Mk} 
\]

\[
M_{LM_{ij}} = FL_{MK_i,j} \times d_{ij} 
\]

\[
FL_{MK_i,j} = \begin{bmatrix}
FL_{MK_{1,1}} & FL_{MK_{1,2}} & \cdots & FL_{MK_{1,q}} \\
FL_{MK_{2,1}} & FL_{MK_{2,2}} & \cdots & FL_{MK_{2,q}} \\
\vdots & \vdots & \ddots & \vdots \\
FL_{MK_{g,1}} & FL_{MK_{g,2}} & \cdots & FL_{MK_{g,q}}
\end{bmatrix}
\]

where,

\[d_{ij}\] is the distance between location i and j
\( C_{MK} \) is the cost per unit distance for resources Mk flow

\( TCL_{MKij} \) is the total cost of resource Mk flow between locations i and j

\( M_{LMIK,ij} \) is the distance travelled by resource Mk flow per unit time between locations i and location j

\( TCL_{MKij} \) is the frequency of resource Mk flow between location I and j per unit time.

With the coordinates, a matrix of distance \( d_{ij} \) can be calculated in the following formula:

\[
d_{ij} = \left| X_{Lj} - X_{Li} \right| + \left| Y_{Lj} - Y_{Li} \right|
\]

where \( L_i \) and \( L_j \) are the coordinates of the locations within the site area.

The AWDA algorithm is applied and the results are tabulated in Table 4.12. In order to get a better appreciation of the performance in AWDA, some comparisons between the costs produced by the GA, Tabu search and Mixed integer programming methodologies are summarised. Comparing with the results from Table 4.12, it is found that the AWDA technique resulted in lower costs than the other techniques. The development of AWDA is capable of dealing with discrete variables and modified its strategy based on problem solving progress. The analysis has shown that the AWDA does perform 7020 saving better than GA (Cheung et al 2002) and 2100 better than TS (Liang & Chao 2008). It has shown that the AWDA performs 2.2% saving better than TS in this experiment.
### Table 4.12: Comparison of best layout for site precast yard

<table>
<thead>
<tr>
<th>Location</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>Total optimum cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Genetic algorithm (Cheung et al 2002)</td>
<td>1</td>
<td>11</td>
<td>8</td>
<td>10</td>
<td>6</td>
<td>4</td>
<td>9</td>
<td>5</td>
<td>3</td>
<td>2</td>
<td>7</td>
<td>99,788</td>
</tr>
<tr>
<td>Tabu search (Liang &amp; Chao 2008)</td>
<td>5</td>
<td>10</td>
<td>8</td>
<td>11</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>7</td>
<td>3</td>
<td>1</td>
<td>9</td>
<td>94,858</td>
</tr>
<tr>
<td>Mixed integer programming (Wong et al 2010)</td>
<td>1</td>
<td>10</td>
<td>8</td>
<td>11</td>
<td>6</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>7</td>
<td>2</td>
<td>9</td>
<td>98,424</td>
</tr>
<tr>
<td>Proposed AWDA</td>
<td>5</td>
<td>10</td>
<td>8</td>
<td>11</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>7</td>
<td>3</td>
<td>6</td>
<td>9</td>
<td>92,758</td>
</tr>
</tbody>
</table>

Figure 4.6 shows the facility arrangement achieved by using the AWDA solution. The main gate and side gate are located on the southern side. Cement, sand aggregate, reinforcement bars and formwork materials are stored in their respective storage areas before they are transported to their production units. Concreting of the pre-cast units is carried out at the casting yard. The concreted units will undergo a curing process in the curing yard before being placed in the lifting yard. Therefore, the casting yard is close to the curing yard and the bending yard is close to the steel storage yard. The facility arrangement achieved by using the AWD solution is similar to Tabu search solution (Liang & Chao 2008) as shown in Figure 4.7, except the locations of main gate, bending yard and steel storage yard. Relocation of these facilities in AWDA leads to saving of 2100 compared Tabu search solution.
Figure 4.6 Optimum layout solution achieved using AWDA

Figure 4.7 Optimum layout solution achieved using Tabu search (Liang & Chao 2008)
4.6.4 Construction Site Unequal Area Layout

If each of the predetermined places is capable of accommodating any of the facilities, then the facility layout problem can be modeled as an equal-area facility layout problem. If some of the predetermined places are only able to accommodate some of the facilities, then the problem becomes an unequal-area facility layout problem, where predetermined places have differing areas. Generally, unequal-area layout problems are more difficult to solve than equal-area layout problems, primarily because unequal-area layout problems introduce additional constraints into the problem formulation (Li & Love 2000). An unequal-area layout of a construction site with 11 facilities is solved using AWDA. This benchmark problem has already been solved using Genetic algorithm (Li & Love 2000) and Particle swarm optimization (Zhang & Wang 2008).

The 11 facilities are:

1. Site Office
2. Falsework workshop
3. Labor residence
4. Storeroom 1
5. Storeroom 2
6. Carpentry workshop
7. Reinforcement steel workshop
8. Side gate
9. Electrical, water and other utilities control room
10. Control Batch workshop
11. Main gate
In a construction site, the main gate and side gate are important for transport and access. Thus the relative position to the gates affects the facility layout decision. However, usually the gates are determined before the construction starts, and they are not subject to change during the allocation process. Thus, the two gates are treated as special facilities which have been ‘clamped’ on the predetermined locations. However, as the relative position to the gates will affect the layout of other facilities, they are included in the facility list as a special constraint to the layout problem (Li & Love 2000).

Among the 11 facilities, 3 are too large to be accommodated in the 2 smaller locations. The smaller locations are 7 and 8. The 3 large facilities include Site office, Labor residence and Concrete Batch workshop. In AWDA, the unequal area constraint, for each solution after every iteration is verified to ensure that no large facilities are allocated to the smaller locations. If not, the solutions which violate the constraints are suitability penalized.

The frequency of trips made between facilities is listed below:

\[
F = \begin{bmatrix}
0 & 5 & 2 & 2 & 1 & 1 & 4 & 1 & 2 & 9 & 1 \\
5 & 0 & 2 & 5 & 1 & 2 & 7 & 8 & 2 & 3 & 8 \\
2 & 2 & 0 & 7 & 4 & 4 & 9 & 4 & 5 & 6 & 5 \\
2 & 5 & 7 & 0 & 8 & 7 & 8 & 1 & 8 & 5 & 1 \\
1 & 1 & 4 & 8 & 0 & 3 & 4 & 1 & 3 & 3 & 6 \\
1 & 2 & 4 & 7 & 3 & 0 & 5 & 8 & 4 & 7 & 5 \\
4 & 7 & 9 & 8 & 4 & 5 & 0 & 7 & 6 & 3 & 2 \\
1 & 8 & 4 & 1 & 1 & 8 & 7 & 0 & 9 & 4 & 8 \\
2 & 2 & 5 & 8 & 3 & 4 & 6 & 9 & 0 & 5 & 3 \\
9 & 3 & 6 & 5 & 3 & 7 & 3 & 4 & 5 & 0 & 5 \\
1 & 8 & 5 & 1 & 6 & 5 & 2 & 8 & 3 & 5 & 0
\end{bmatrix}
\]

The distances of 11 locations are listed in the following equation. The distances are measured in meters.
The objective of the problem is to minimize the total travelling distance of site personnel between facilities. The total distance is defined as:

\[
\text{Minimize } \text{TD} = \sum_{i=1}^{n} \sum_{x=1}^{n} \sum_{j=1}^{n} \delta_{xixj} f_{ij} d_{ij}
\]  \hspace{1cm} (4.41)

Subject to,

\[
\sum_{x=1}^{n} \delta_{xii} = 1, \{i = 1, 2, 3, \ldots, n\}
\]  \hspace{1cm} (4.42)

where

\(n\) is the number of facilities

\(\delta_{xi}\) is the permutation matrix variable

\(f_{ij}\) is the frequency trips made by construction personnel between facilities \(i\) and \(j\)

\(d_{ij}\) is the distances between locations \(m\) and \(n\)

The two gates (facility 8 and 11) are clamped during generation of initial wild dog pack and are not subjected to relocation during iterations. The optimal results obtained by AWDA are compared in Table 4.13 with the optimal layout of Li & Love (2000) obtained by genetic algorithm and Zhang & Wang (2008) obtained by Particle swarm optimization. It is found that the

\[
D = \begin{bmatrix}
0 & 15 & 25 & 33 & 40 & 42 & 47 & 55 & 35 & 30 & 20 \\
15 & 0 & 10 & 18 & 25 & 27 & 32 & 42 & 50 & 45 & 35 \\
25 & 10 & 0 & 8 & 15 & 17 & 22 & 32 & 52 & 55 & 45 \\
33 & 18 & 8 & 0 & 7 & 9 & 14 & 24 & 44 & 49 & 53 \\
40 & 25 & 15 & 7 & 0 & 2 & 7 & 17 & 37 & 42 & 52 \\
42 & 27 & 17 & 9 & 2 & 0 & 5 & 15 & 35 & 40 & 50 \\
47 & 32 & 22 & 14 & 7 & 5 & 0 & 10 & 30 & 35 & 40 \\
55 & 42 & 32 & 24 & 17 & 15 & 10 & 0 & 20 & 25 & 35 \\
35 & 50 & 52 & 44 & 37 & 35 & 30 & 20 & 0 & 5 & 15 \\
30 & 45 & 55 & 49 & 42 & 40 & 35 & 25 & 5 & 0 & 10 \\
20 & 35 & 45 & 53 & 52 & 50 & 40 & 35 & 15 & 10 & 0
\end{bmatrix}
\]
AWDA technique resulted in lower costs than the GA and PSO. It has shown that the AWDA performs 18.3% saving better than GA in this problem.

Table 4.13  Comparison of optimal results of unequal area construction site layout

<table>
<thead>
<tr>
<th>Location</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>Total optimum cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Genetic algorithm (Li &amp; Love 2000)</td>
<td>8</td>
<td>5</td>
<td>7</td>
<td>10</td>
<td>2</td>
<td>9</td>
<td>4</td>
<td>6</td>
<td>3</td>
<td>11</td>
<td>1</td>
<td>15610</td>
</tr>
<tr>
<td>Particle swarm optimization (Zhang &amp; Wang 2008)</td>
<td>8</td>
<td>5</td>
<td>7</td>
<td>10</td>
<td>2</td>
<td>9</td>
<td>4</td>
<td>6</td>
<td>3</td>
<td>11</td>
<td>1</td>
<td>16060</td>
</tr>
<tr>
<td>Proposed AWDA</td>
<td>8</td>
<td>1</td>
<td>10</td>
<td>9</td>
<td>3</td>
<td>7</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>11</td>
<td>2</td>
<td>12756</td>
</tr>
</tbody>
</table>

4.6.5  Construction Site Layout Planning

Lam et al (2009) used the traditional GA and a modified GA (Max-min ant system – Genetic algorithm) to solve a construction site layout planning problem with nine facilities of equal area and twenty four possible locations. In this example number of predetermined locations is greater than the number of predetermined facilities. In FLP, if the number of predetermined locations is greater than the number of predetermined facilities, then a number of ‘dummy’ fictitious facilities will be added to make both numbers equal. By assigning both the distance and frequency as 0, the ‘dummy’ facilities will not affect the layout results (Li & Love 2000).
The objective of this example is to minimize the transportation of work flows between the site facilities, for which depends mainly on two attributes (Lam et al 2009):

1. Closeness index of work flow \( C_{ij} \) between site facilities
2. Distance \( d \) between the site locations.

In order to minimize the transportation of work flows, Ning et al (2009) used closeness index \( C_{ij} \) to show the closeness relation between the facilities. The work flows include transportation of materials, equipment, etc. from one facility to the other. The higher the frequency of the transportation of work flows between facilities, the closer the distance from the facilities to the others.

The objective function of the problem can be defined as follows:

\[
\min \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} C_{ij} d_{kl} x_{ik} x_{jl} \tag{4.43}
\]

Subject to the following constraints:

\[
\sum_{i=1}^{n} x_{ij} = 1 \tag{4.44}
\]

\[
\sum_{j=1}^{n} x_{ij} = 1 \quad \text{and}
\]

\[X \in \{0,1\}\]

where,

\( C_{ij} \) is the closeness index of work flow between facilities \( i \) and \( j \)

\( D_{kl} \) is the distance between the facilities \( k \) and \( l \)

\( x_{ik} \) and \( x_{jl} \) means when facility \( I \) is assigned to location \( k \) and facility \( j \) is assigned to location \( l \) respectively.

The constraint of \( x_{ij} \) will be a binary variable which takes value 1 if facility \( i \) is assigned to location \( j \) and 0 otherwise (Lam et al, 2009).
The following nine essential facilities are considered in this problem.

1. Refuse Chute
2. Debris storage area
3. Reinforcement bending yard
4. Carpentry workshop and store
5. Labor hunt
6. Materials Storage area
7. Main Gate
8. Material hoist
9. Site office

Figure 4.8 Simplified layout of the hypothetical construction site
The three fixed facilities are site office, material hoist and main gate in the construction site. The location of the main gate is usually predetermined in terms of external transportation system. Site office is always near the main gate for the safety of the working staff. The materials hoist is used for the transportation of materials to the superstructure and its location is dependent on the structural element to which it is tied, and thus site planners always freeze this facility in a certain location. Based on the above reasons, site office, main gate and material hoist are predetermined by the planner (Lam et al 2009). The twenty four free locations available for other six facilities are shown in Figure 4.8. Table 4.14 shows the closeness index between the facilities calculated and adopted by Lam et al (2009) based the frequency of the transportation of work flows between facilities.

### Table 4.14 Closeness index of work flows between the site facilities

<table>
<thead>
<tr>
<th>Site facilities</th>
<th>Facility number</th>
<th>Closeness index of work flow between facilities</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Facility number</td>
<td>1</td>
</tr>
<tr>
<td>Refuse chute</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Debris storage area</td>
<td>2</td>
<td>0.2</td>
</tr>
<tr>
<td>Rebar bending yard</td>
<td>3</td>
<td>2.2</td>
</tr>
<tr>
<td>Carpentry workshop and store</td>
<td>4</td>
<td>2.3</td>
</tr>
<tr>
<td>Labor hut</td>
<td>5</td>
<td>2.3</td>
</tr>
<tr>
<td>Materials storage area</td>
<td>6</td>
<td>2.4</td>
</tr>
<tr>
<td>Main gate</td>
<td>7</td>
<td>4.2</td>
</tr>
<tr>
<td>Materials hoist</td>
<td>8</td>
<td>2.7</td>
</tr>
<tr>
<td>Site office</td>
<td>9</td>
<td>2.5</td>
</tr>
</tbody>
</table>

The results from the African Wild Dog Algorithm are compared with the best result found by other algorithms GA and MMAS-GA in Table 4.15. The comparative results are based on the optimal work flow objective function values of the facility layout arrangements. Compared to
other methodologies, the AWDA performs well with reduced objective function value. The optimum layout solutions achieved using AWDA, GA (Lam et al 2009) and MMAS-GA (Lam et al 2009) are shown in Figures 4.9, 4.10 and 4.11 respectively.

Table 4.15 Optimal results of construction site layout planning locations

<table>
<thead>
<tr>
<th>Facilities</th>
<th>Location of facilities and optimal work flow</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GA (Lam et al 2009)</td>
</tr>
<tr>
<td>Refuse chute</td>
<td>18</td>
</tr>
<tr>
<td>Debris storage area</td>
<td>2</td>
</tr>
<tr>
<td>Reinforcement bending yard</td>
<td>15</td>
</tr>
<tr>
<td>Carpenter workshop and store</td>
<td>16</td>
</tr>
<tr>
<td>Labor hunt</td>
<td>5</td>
</tr>
<tr>
<td>Materials Storage area</td>
<td>12</td>
</tr>
<tr>
<td>Optimal work flow</td>
<td>2063.10</td>
</tr>
</tbody>
</table>

Figure 4.9 Optimum layout solution achieved using GA (Lam et al 2009)
Figure 4.10 Optimum layout solution achieved using MMAS - GA (Lam et al 2009)

Figure 4.11 Optimum layout solution achieved using AWDA
4.7 PERFORMANCE OF AWDA IN ENGINEERING OPTIMIZATION PROBLEMS

In this chapter, several well-studied engineering design problems taken from the optimization literature are used to show the efficiency of the proposed AWDA. These examples have been previously solved using a variety of other techniques, which is useful to show the validity and effectiveness of the proposed algorithm.

Standard problems in mechanical engineering such as welded beam design, pressure vessel design and spring design are optimized to prove the robustness of the algorithm. An economic dispatch problem (EDP), which is related to the optimum generation scheduling of available generators in an electrical power system, to minimize the total fuel cost while satisfying the load demand and operational constraints is also attempted. A new methodology is proposed to optimize facility layout problems applicable to various branches of engineering. It is found that the proposed AWD algorithm performs better than other meta-heuristics such as GA, PSO, HS and TS for the problems considered.