Mathematical Models of Energy-Conscious Bi-Objective Unrelated Parallel Machine Scheduling

Bobby Kurniawan
Department of Industrial Engineering, University of Sultan Ageng Tirtayasa, Indonesia
Jl. Raya Jkt Km 4 Jl. Pakupatan, Panancangan, Kec. Cipocok Jaya, Kota Serang, Banten 42124, Indonesia
*Corresponding author: kurniawan76@gmail.com

1. Introduction

Although economic development contributes to human prosperity, it produces undesirable side effects that will threaten the environment. Accumulated greenhouse gas emissions resulting from fossil fuel combustion are attributed to global warming [1]. Fossil fuel is the primary source in generating energy to drive the economy. Therefore, several issues will arise with the increasing demand for electricity due to industrialization and population growth [2]. First, it produces energy scarcity because fossil fuel depletes over time and cannot be renewed. Second, it makes the quality of the environment deteriorated. Governments have implemented policies to stimulate energy-saving and sustainability practices among energy users to anticipate such high demand. The policies are imposed on the energy suppliers, particularly the manufacturing sector, as an extensive energy user. It has been investigated that the manufacturing industry expedites more than half

of the total world consumption [3]. As a response, the manufacturing industry has implemented a new paradigm for its operational activities. In this paradigm, energy reduction and sustainability must be taken into account as a performance measure. As a result, research on the implementation of energy-saving and sustainability in manufacturing is abundant. Notably, research in scheduling also considers energy consumption as an objective besides the traditional ones.

The manufacturing sector can reduce its consumption by designing and installing energy-efficient machines and equipment [4] [5]. However, this approach requires massive investment. Small companies cannot afford to fund such investments. Therefore, researchers have paid the “soft method”, such as scheduling as an alternative approach. Research that has been conducted in the past demonstrated the importance of scheduling as an energy-reduction and sustainability method [6] [7]. For instance, Mouzon et al. [8] implemented a machine power on/off mechanism. A machine was turned off if it was in an idle state. Machine-speed scaling was another approach to reduce energy consumption. A job can be processed faster but requires more energy [9]. A hybrid sine cosine algorithm was proposed to reduce carbon emissions in a manufacturing company [10]. Recently, energy providers implemented Time-Of-Use tariffs (TOU) to reduce generation and operations costs. The policy provides incentives for consumers that adjust their consumption patterns. In this policy, electricity price in a period is different from those of other periods. Periods are categorized as peak periods (higher prices), middle-peak periods, and off-peak periods (lower prices) [11] [12]. The manufacturer that schedules its operational time from peak to off-peak periods can reduce its operational cost. As a result, the maximum load can be lowered so that energy suppliers can reduce operational and maintenance costs [13], [14]. Therefore, the TOU policy provides the energy-saving measure for energy providers and consumers. Accordingly, the research topic of energy, especially scheduling under TOU, has been abundant [15], [16].

For research of Unrelated Parallel Machine Scheduling (UPMS) under TOU, Moon et al. [17] have studied UPMS under TOU to minimize the sum of the weighted makespan and electricity cost. They proposed a hybrid genetic algorithm (GA) to tackle the problem. Koo and Kim [18] improved Moon et al.’s model. Kurniawan et al. [19] have proposed a GA with a self-adaptive mechanism to solve the same problem as Moon et al. Ding et al. [20] have developed a mixed-integer programming (MIP) model and proposed a column generation method in dealing with UPMS to minimize the electricity consumption. A mathematical model for UPMS under TOU has been presented by Hossein et al. [21]. For research of multi-objective UPMS, Nikabadi and Naderi [22] have proposed a hybrid genetic algorithm (GA) and simulated annealing (SA) to handle the multi-objective UPMS. The objectives are the number of tardy jobs and total tardiness/earliness. Li et al. [23] proposed ten heuristic rules to solve UPMS with energy consideration. A multi-objective UPMS subject to resource constraints was investigated by Wang and Liu [24]. They proposed a meta-heuristic based on an artificial immune system combined with a non-dominated sorting strategy to solve the problem.

The discussion above shows that energy and the environment are essential research topics because they are critical to human life. Hence, the energy-saving method is fostered by the government, energy suppliers, and energy users. Some research has reported the importance of scheduling as an energy-saving measure. Previous research of UPMS under TOU only considered the electricity cost. Similarly, previous studies of multi-objectives did not consider TOU. The current study tries to fill the energy-conscious multi-objective scheduling problem gap, especially the TOU schedule. Therefore, this research addresses multi-objective energy-conscious scheduling for unrelated parallel machine environments under TOU. The objectives are to minimize the total tardiness and...
electricity cost simultaneously. The problem is formulated as two MIP models. The time-index formulation [25] is used to formulate the first model. In contrast, the second model is formulated based on disjunctive formulation [26]. Both models are solved using the weighted sum method (WSM) and calculated using CPLEX. The WMS method possesses several advantages compared to other multi-objective algorithms, such as Nondominated Sorting Genetic Algorithm II (NSGA-II) [27] and Multi-Objective Evolutionary Algorithm Based On Decomposition (MOEA/D) [28]. A multi-objective evolutionary algorithm (NSGA-II and MOEA/D) can generate a set of non-dominated solutions in one run. However, the optimality of the solutions is not yet known and not guaranteed to be optimum.

On the other hand, the WMS is easy to be implemented and high search efficiency, and the ability to find a non-dominated solution [29]. However, a priori preference must be determined manually [30], and it is hard to decide on the weight for each objective. As a result, the WMS must be executed in more than one run to obtain a Pareto frontier. In sum, the contributions of this research are as follows. First, the multi-objective energy-conscious UPMS under TOU that differs from previous research is investigated. Then, two MIP models are developed using the disjunctive and time-indexed formulations. Finally, the sum of the weighted method solved the models.

The remainder of this paper is structured as follows. Section 2 describes the problem formulation and the proposed method. Section 3 presents the results and analysis. Section 4 concludes this research and provides some future works.

2. Methods

2.1 Formulation

The problem considered in this research is stated as follows. A factory has N jobs, j = 1, 2, ..., N, that must be processed in M machines, i = 1, 2, ..., M. Each job has a due date d_j. The processing time of job j in machines i is p_{ij}. All jobs must be finished within a discrete-time horizon [0, U]. The time horizon is divided into U periods with length one. A period t represents an interval [t - 1, t], t = 1, 2, ..., T. Electricity cost incurs when a job is processed on a machine and depends on the job period. The electricity price in each period, a_t, is different from those in other periods. The objectives are to minimize the total tardiness and electricity cost.

Fig. 1 illustrates a daily TOU price in China, and Fig. 2 shows an example of a schedule. Periods 1–8 denote the off-peak in which electricity price is lower compared with other periods. Periods 13–17 and 22–24 denote the mid-peak periods. Periods 9–12 and 22–24 denote the peak periods in which the electricity prices are the highest.

Consider a schedule of 4 jobs and two machines, as shown in Fig. 2. The calculation of the electricity price is demonstrated using job 1. Job 1 starts at time zero and finishes
at time 8. Period 1 starts at period one and finishes at period 1. Therefore, job 1 is processed from period one until period 8. Using the price in Fig. 1, the electricity cost to process job 1 is 0.3075 * 8 = 2.46.

2.2 Assumptions

The problem of UPMS under TOU is formulated to become model 1 ($P_1$) and model 2 ($P_2$). Model $P_1$ is developed based on time-index formulation, whereas model $P_2$ is derived based on disjunctive formulation. In modeling the scheduling problems, the most crucial consideration is how to formulate the decision variable for sequencing. The variable used in the sequencing is binary. The differences in modeling the binary variables lead to different complexity.

Several assumptions for the model $P_1$ and $P_2$ are as follows.

1. No pre-emption is allowed, i.e., all jobs must be processed until finished without interruption.
2. The machines can be idle and require no electricity cost.
3. All jobs are available at time zero.
4. Machines are always available, i.e., no machine breakdowns.
5. There are no maintenance activities.

Notations used by model $P_1$ and $P_2$ are:

- $d_j$ = the due date of job $j$
- $p_{ij}$ = the processing time of job $j$ on machine $i$
- $\omega_1, \omega_2$ = the weight of total weighted tardiness objective and electricity cost objective, respectively, where $\omega_1 + \omega_2 = 1$.

2.3 Model 1 ($P_1$)

The binary variables, integer variables, and parameters used in this model are as follows.

Variables:
- $x_{ijt}$ : 1 if job $j$ is processed on machine $i$ in period $t$, 0 otherwise
- $y_{ij}$ : 1 if job $j$ is processed on machine $i$, 0 otherwise
- $z_{ij(t+1)}$: 1 if job $j$ is processed on machine $i$ during period $t$ and $t + 1$, 0 otherwise
- $C_j$ : completion time of job $j$
- $T_j$ : tardiness of job $j$

Parameters
- $a_t$ : the electricity price in period $t$

Model $P_1$ is formulated as follows.

Minimize

\[
\omega_1 \sum_{j=1}^{N} T_j + \omega_2 \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{t=1}^{U} a_t x_{ijt}
\]

\[
\sum_{t=1}^{U} \sum_{i=1}^{M} x_{ijt} \leq 1, \ 1 \leq j \leq N
\] (2)

\[
\sum_{t=1}^{U} x_{ijt} = p_{ij} y_{ij}, \ 1 \leq i \leq M; 1 \leq j \leq N
\] (3)
\[
\sum_{i=1}^{M} y_{ij} = 1, 1 \leq j \leq N \tag{4}
\]

\[
x_{ij} + x_{i(j+1)} \geq 2z_{ij(t+1)}, 1 \leq i \leq M; 1 \leq j \leq N; 1 \leq t \leq U \tag{5}
\]

\[
\sum_{t=1}^{U-1} z_{ij(t+1)} \geq \sum_{t=1}^{U} x_{ijt} - 1, 1 \leq i \leq M; 1 \leq j \leq N \tag{6}
\]

\[
x_{ij} \geq z_{ij(t+1)}, 1 \leq i \leq M; 1 \leq j \leq N; 1 \leq t \leq U - 1 \tag{7}
\]

\[
x_{ij(t+1)} \geq z_{ij(t+1)}, 1 \leq i \leq M; 1 \leq j \leq N; 1 \leq t \leq U - 1 \tag{8}
\]

\[
tx_{ijt} \leq C_{ij} \forall i = 1, 2, ..., M; 1 \leq i \leq M; 1 \leq j \leq N; 1 \leq t \leq U \tag{9}
\]

\[
C_{ij} - d_{j} \leq T_{j}, 1 \leq j \leq N \tag{10}
\]

Eq. (1) expresses the objective functions. Specifically, the first and second term denotes the total tardiness and electricity cost, respectively. Eq. (2) guarantees that each job can be processed only at one machine at a time. Eq. (3) enforces that once job \( j \) is allocated on machine \( i \), its processing time must be equal \( p_{ij} \). Eq. (4) enforces every job at least is allocated only on one machine. Eqs. (5)–(8) guarantee that the job pre-emption is satisfied. Eq. (9) calculates the completion time of each job. Finally, Eq. (10) calculates the tardiness of each job.

### 2.4 Model 2 (P2)

This model introduces a dummy job (job 0); all parameters’ values are zero. The dummy job indicates the first and the last job in the sequence. The binary variables, integer variables, and parameters used in this model are as follows.

**Variables:**
- \( X_{ijk} \): 1 if job \( j \) is processed before job \( k \) on machine \( i \), 0 otherwise
- \( Y_{ij} \): 1 if job \( j \) is processed on machine \( i \), 0 otherwise
- \( Y_{ik} \): 1 if job \( k \) is processed on machine \( i \), 0 otherwise
- \( Z_{ijt} \): 1 if job \( j \) finishes being processed on machine \( i \) on period \( t \), 0 otherwise
- \( \mu_{j}, \mu_{k} \): completion time of job \( j \) and job \( k \), respectively
- \( \tau_{j} \): tardiness of job \( j \)

**Parameters:**
- \( V \): a very big number
- \( a_{v} \): electricity price at period \( v \)
- \( E_{ijt} \): electricity cost incurs if job \( j \) processed on machine \( i \) finishes at period \( t \)

\[
E_{ijt} = \sum_{v=t-p_{ij}+1}^{t} a_{v}, 1 \leq i \leq M; 1 \leq j \leq N; p_{ij} \leq t \leq U
\]

Model \( P_{2} \) is formulated as follows.

\[
\text{Minimize } \omega_{1} \sum_{j=1}^{N} \tau_{j} + \omega_{2} \sum_{t=1}^{U} \sum_{j=1}^{N} \sum_{i=1}^{M} E_{ijt} Z_{ijt} \tag{11}
\]
\[
\sum_{i=1}^{M} Y_{ij} = 1, 1 \leq j \leq N \tag{12}
\]

\[
\sum_{k=0}^{N} X_{ijk} = Y_{ik}, 1 \leq k \leq N; k \neq j; 1 \leq i \leq M \tag{13}
\]

\[
\sum_{k=0}^{N} X_{ijk} = Y_{ij}, 1 \leq j \leq N; k \neq j; 1 \leq i \leq M \tag{14}
\]

\[
\sum_{k=1}^{N} X_{i0k} \leq 1, 1 \leq i \leq M \tag{15}
\]

\[
\mu_k \geq \mu_j + p_{ik} - (1 - X_{ijk})V, 1 \leq i \leq M; 0 \leq j \leq N; 1 \leq k \leq N; j \neq k \tag{16}
\]

\[
\mu_0 = 0 \tag{17}
\]

\[
\mu_j - d_j \leq \tau_j, 1 \leq j \leq N \tag{18}
\]

\[
\sum_{j=1}^{N} Z_{ijt} = 1, 1 \leq i \leq M; 1 \leq t \leq U \tag{19}
\]

\[
\sum_{t=1}^{U} Z_{ijt} \leq Y_{ij}, 1 \leq i \leq M; 1 \leq j \leq N \tag{20}
\]

\[
\sum_{i=1}^{M} \sum_{t=1}^{U} tZ_{ijt} = \mu_j, 1 \leq j \leq N \tag{21}
\]

Eq. (11) states the objective functions; the first term is the total tardiness, whereas the second term is electricity cost. Eq. (12) guarantees that each job can be processed only at one machine at a time. Eq. (13) guarantees that each job is processed only on one machine. Eqs. (14)–(15) ensure each job have only one predecessor and successor. Eq. (16) ensures that a maximum of one dummy job is sequenced as the first job on each machine. Eq. (16) is the disjunctive constraint. Eq. (17) forces the completion time of the dummy job is zero. Eq. (18) expressed the tardiness of each job. Eq. (19) enforces that each job is processed only by one machine and finishes only on a period. Eq. (20) guarantees that if job \( j \) is allocated on machine \( i \), its completion time only occupies a one-time slot. Eq. (21) defines the relationship between binary variables \( Z_{ijt} \) and the completion time \( \mu_j \), i.e., if \( Z_{ijt} = 1 \), then the jobs finished are to be processed on period \( \mu_j \).

### 2.5 Experiment Method

The instances are generated using a combination of the number of jobs and machines. The number of jobs is \( N = \{6, 10, 15, 20\} \), and the number of machines is \( M = \{3, 4, 5\} \). For each job, the processing time is randomly generated follows the uniform distribution of \([1, 10]\). Once the processing time of each job in each instance is determined,
each job’s due date is calculated as \( d_j = \left\lfloor \frac{\sum_{t=1}^{M} P_{ij}}{M} \right\rfloor \) and the number of periods as \( U = \left\lfloor \frac{N \sum_{t=1}^{M} P_{ij}}{M^2} \right\rfloor \). The TOU price follows the pattern Fig. 1 and is repeated after 24 periods.

The computational experiments are performed on a Dual Core Laptop with 3 GB RAM. Model \( P_1 \) and \( P_2 \) are coded using OPL modeling language and solved using CPLEX solver. For each model, each instance is run eleven times. The weight of the total tardiness objective is set to \( \omega_1 = 0.2 \). The maximum time of each run is set to 3,600 seconds. Suppose CPLEX cannot find an optimum solution within 3,600 seconds. In that case, the calculation of CPLEX is stopped. The results (the computation time, the gap, the number of binary variables, integer variables, and constraints) are returned. The gap denotes the duality gap, i.e., the deviation between the upper and lower bound calculated by CPLEX. If the gap is zero, then the solution is optimum. Therefore, the lower the gap, the better the solution.

The procedure of one run of CPLEX is as follows. First, determine the weight of each objective (\( \omega_1 = 0.2 \) and \( \omega_2 = 0.8 \)). Using this combination, model \( P_1 \) is calculated using CPLEX. This process is called one run. The process is repeated for other combinations of weights. The same process is conducted for model \( P_2 \).

### 2.6 Analysis of Model Complexity

The number of binary variables, integer variables, and constraints characterizes the complexity of a MIP model. Therefore, we assess model \( P_1 \) and \( P_2 \) by using the three indicators mentioned above. Let us consider an unrelated parallel machine scheduling under TOU of \( N \) jobs, \( M \) machines, and \( U \) periods.

For model \( P_2 \), the binary variables are \( x_{ijt}, y_{ij}, \) and \( z_{ij(t+1)} \). Therefore, the number of binary variables are \( MNU + MN + MNU(U + 1) = MNU^2 + 2MNU + MN \). The integer variables are \( C_j \) and \( T_j \), so the number of integer variables is \( 2N \). We omit Eq. (1) because it does not depend on \( N, M, \) or \( U \). Eqs. (2) and (4) have \( 2N \) constraints. Eqs. (3), (6), and (10) consist of \( 2MN \) constraints. Eq. (5) and (9) have \( 2MNU \) constraints. Finally, Eqs. (7) and (8) constitute \( 2MN(U - 1) \). In total, model \( P_1 \) has \( 2N + 2MN + 2MNU + 2MN(U - 1) = 4MNU + 2N \).

Model \( P_2 \) has \( MN^2 \) (from \( X_{ijk} \)), \( N^2 \) (from \( Y_{ij} \)), and \( MNU \) (from \( Z_{ij(t)} \)). The number of integer variables is \( 2N \). Lastly, the number of constraints is \( MN^2 + 3MN + MU + 2N + M \). The summary of the number of binary variables, integer variables, and constraints of the model \( P_1 \) and \( P_2 \) is shown in Table 1.

<table>
<thead>
<tr>
<th>Model P1</th>
<th>Model P2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary ( MNU^2 + 2MNU + MN ) | ( MN^2 + N^2 + MNU = MN )</td>
<td></td>
</tr>
<tr>
<td>Integer ( 2N ) | ( 2N )</td>
<td></td>
</tr>
<tr>
<td>Constraints ( 4MNU + 2N ) | ( MN^2 + 3MN + MU + 2N + M )</td>
<td></td>
</tr>
</tbody>
</table>

### 3. Results and Discussion

Fig. 3 shows the non-dominated solutions found for the \( 10 \times 3 \) instance. There are four non-dominated solutions found. Fig. 3 is obtained by solving model \( P_1 \) with CPLEX. Fig. 3 shows that the Pareto front is not convex. It can be explained since the objective functions of scheduling problems are not convex. Note that although the instance is run 11 times with different weight values, the number of non-dominated solutions is only 4.
Therefore, even a weight has been assigned and solved by the exact approach. The solution found is not guaranteed to be non-dominated [30].

Fig. 3 The Pareto frontier of 10 × 3 instance

<table>
<thead>
<tr>
<th>Instance</th>
<th>Period</th>
<th>Time (s)</th>
<th>Gap (%)</th>
<th>Binary</th>
<th>Integer</th>
<th>Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>6×2</td>
<td>24</td>
<td>3.15</td>
<td>3.7</td>
<td>0</td>
<td>0</td>
<td>300 386 13 14 1202 138</td>
</tr>
<tr>
<td>10×2</td>
<td>28</td>
<td>99.51</td>
<td>*</td>
<td>0</td>
<td>25.99</td>
<td>580 802 21 22 2298 779</td>
</tr>
<tr>
<td>15×2</td>
<td>38</td>
<td>*</td>
<td>*</td>
<td>32.53</td>
<td>59.43</td>
<td>1170 1652 31 32 4638 606</td>
</tr>
<tr>
<td>20×2</td>
<td>49</td>
<td>*</td>
<td>*</td>
<td>42.35</td>
<td>61.85</td>
<td>2000 2842 41 42 7940 2797</td>
</tr>
<tr>
<td>6×3</td>
<td>24</td>
<td>3.78</td>
<td>0.66</td>
<td>0</td>
<td>0</td>
<td>450 579 13 14 1796 193</td>
</tr>
<tr>
<td>10×3</td>
<td>24</td>
<td>182</td>
<td>136</td>
<td>0</td>
<td>0</td>
<td>930 1263 21 22 3682 437</td>
</tr>
<tr>
<td>15×3</td>
<td>24</td>
<td>*</td>
<td>*</td>
<td>23.55</td>
<td>20.13</td>
<td>1980 2703 31 32 7856 2655</td>
</tr>
<tr>
<td>20×3</td>
<td>30</td>
<td>*</td>
<td>*</td>
<td>30.48</td>
<td>43.23</td>
<td>1800 3123 41 42 7272 3066</td>
</tr>
<tr>
<td>6×4</td>
<td>24</td>
<td>0.22</td>
<td>0.36</td>
<td>0</td>
<td>0</td>
<td>600 772 13 14 2390 248</td>
</tr>
<tr>
<td>10×4</td>
<td>24</td>
<td>1.53</td>
<td>11.7</td>
<td>0</td>
<td>0</td>
<td>1000 1444 21 22 3918 568</td>
</tr>
<tr>
<td>15×4</td>
<td>24</td>
<td>1520</td>
<td>*</td>
<td>0</td>
<td>28.43</td>
<td>1500 2464 31 32 5828 1148</td>
</tr>
<tr>
<td>20×4</td>
<td>30</td>
<td>*</td>
<td>*</td>
<td>24.11</td>
<td>45.32</td>
<td>2480 4164 41 42 9682 3949</td>
</tr>
</tbody>
</table>

Table 2 shows the results of computational experiments using CPLEX. Table 2 only shows the experiments using $\omega_1 = 0.2$ and $\omega_2 = 0.8$. CPLEX requires an average of 1,651 seconds to solve all instances using model $P_1$. For solving all instances using model $P_2$, CPLEX needs 2,113 seconds on average. Therefore, model $P_1$ is more efficient than model $P_2$ because it requires about 80% percent of model $P_2$ to solve the same number of instances. The average solution gap resulted from a model $P_1$ is also better than those of $P_2$. Model $P_1$ requires fewer binary variables than $P_2$. The addition of dummy jobs makes model $P_2$ has more one integer variables than model $P_1$. As for the constraints, we can see from Table 2 model $P_1$ needed more constraints than model $P_2$. Therefore, we can say that time-indexed formulation is more compact than the disjunctive formulation. For other combinations of $\omega_1$ and $\omega_2$, the results are similar to those shown in Fig. 2. The results support the findings of Moon et al. [17] and Ding et al. [20]. Although their model has one objective (electricity cost), they showed that a single objective UPMS under TOU is hard to solve. Therefore, they proposed a hybrid Genetic Algorithm [17] and Column Generation [20] to solve UPMS under TOU.
From the results in Table 2, several findings are as follows. First, the time-indexed formulation is more suitable for solving the unrelated parallel machine scheduling under TOU than the disjunctive formulation. We imply this finding based on the average solutions and gap resulted from both formulations. Second, the time-indexed formulations provide a more compact model than the disjunctive formulation, based on the number of binary variables. Lastly, the performance of a MIP model largely depends on the number of binary variables. Although the time-indexed formulation is efficient than the disjunctive formulation, both cannot solve more than 15 jobs and 24 periods. It is not practical in a real-world application where the number of jobs is significant. Hence, other approaches, such as heuristic [31] and metaheuristics [32], can be applied in a real-world application.

4. Conclusion

This study addressed the bi-objective unrelated parallel machine scheduling under TOU tariffs. The problem is modeled as two MIP models using time-indexed and disjunctive formulations. We analyzed the complexity of both models based on the number of binary variables, integer variables, and constraints. After that, we performed computational experiments to compare the effectiveness of both models. The models were solved using the weighted sum method and computed using CPLEX. The results showed that the time-indexed formulation is more efficient than the disjunctive formulation to solve the bi-objective scheduling problem. However, both models cannot find an optimum solution for more than 15 within a reasonable time. It indicates that the problem needs a suitable approach to solve the problem of large instances. Metaheuristics such as NSGA-II, Non-dominated Sorting Genetic Algorithm III, the Strength Pareto Evolutionary Algorithm 2 (SPEA2), and MOEA/D are good candidates as a solution approach.

Acknowledgments

The authors wish to thank anonymous referees for providing valuable and constructive feedback.

References


