Job Shop Scheduling Application Using Genetic and Branch and Bound Algorithms

Briggs, Tobinson, A.

1Department of Mechatronic Engineering, University of Port Harcourt, Port Harcourt, Nigeria

ABSTRACT

The paper tackled two methodologies in solving job shop scheduling problems, the “genetic and branch and bound” algorithm. The aim is to advance the existing schedule of jobs assigned to machines in a local glass-container factory as a mean of reducing delay in the current schedule so that the mean processing time is minimized. In the study, data used was supplied by the manufacturer along side of the development of model representing each manufacturing line processing time of each job. Their performances are compared for the jobs constructed and appraise each algorithm applied. Technological constraints and other constraints such as precedence relations and resource availability mainly the furnace capacity were considered.

Keywords: Genetic Algorithm, Branch and Bound Algorithm, Scheduling, Processing Time, Job Shop.

INTRODUCTION

Pinedo in his work defined “scheduling as the allocation of distributed assets to tasks over a given period of time” (Pinedo, 2001). Job shop scheduling problem can be illustrated generally by comprises of n jobs in a set \( \{J_i\}_{1 \leq i \leq n} \) and which will be processed on m machines of a set \( \{M_j\}_{1 \leq j \leq m} \).

The problem can be characterized as follows:

1. On each machine each job is processed in an order given by a predefined sequence of operations
2. One job is processed on each machine at a time
3. Job \( J \) is processed on machine \( M \) which is defined by the operation \( O_{ij} \)
4. Each operation \( O_{ij} \) requires an uninterrupted processing on machine \( M \) and preemption is not allowed
5. The processing times for each operation are known in advance

Products or parts are produced in mechanized facility where all necessary operation are performed is known as a shop. Emblematic shops may be divided into single or multiple machines in its operations. Multi-machine shops have machines that may be the same, similar, or different based on the required doling out technique. In a plant all machines are alienated into shops. Plants are decomposed into smaller shops for more specialized and improved solution measures is developed for planning, scheduling and quality control purposes. On the other hand, it gives room for increased need of coordination among shops.

THE STRUCTURE OF GENETIC ALGORITHM

Inspired by the principles of natural evolution, GA’s are one of the well known types of evolutionary algorithms (Holland, 1985). Akin to biological reproduction, sexual GA’s produce offspring from two parents, while asexual GA’s produce offspring from one parent. At the start, a set of possible solutions termed initial population (NPOP) of chromosomes is populated. Each generation of offspring is called iteration. Reproductive phenomena that enable diversity like crossover and mutation are achieved in GA’s by using case-specific operators. In the next few paragraphs, the working of sexual and asexual reproduction processes in GA is presented.
In sexual reproduction the GA’s work similar to that of human genetics. Although mutation which involves the refinement of single chromosomes is important, the crossover operation defines sexual reproduction. Individual chromosomes from the NPOP are assessed and given fitness values which are stored. Using the fitness values as the selection criterion, the crossover operator forms two new offspring from two chromosomes in the first iteration. This iterative process is repeated producing a new generation each time until the loop closes according to set criteria. Finally, the last generation of chromosomes which form the solution to the problem are evaluated.

Asexual reproduction in GA is similar to sexual reproduction in that fitness values are assessed, calculated and stored from the NPOP, but in this case, rearrangement of component genes in each chromosome is performed. Since there is only one parent, mutation is the basis of asexual reproduction like in single-celled species. The last process involves sorting to find the best chromosomes. Various strategies of gene rearrangement have been presented in the literature.

In this study, asexual reproduction with two mutation operations was applied to find best solutions. The following pseudo code below (Holland, 1975), describing the general working of a GA with asexual reproduction, (crossover is not performed) was used.

\[
\begin{align*}
t = 0; \\
\text{Initialize } (K \ (t=0)); \\
\text{Evaluate } (k \ (t =0)); \\
\text{While not Terminated } () \text{ do} \\
Kp(t) = k(t).\text{select parents}(); \\
Kc(t) = \text{Mating}(Kp); \\
\text{Mutate1 } (Kc(t) ); \\
\text{Mutate2 } (Kc(t) ); \\
\text{Evaluate } (Kc(t)); \\
k \ (t+1) = \text{build next generation from } (Kc(t), k(t)); \\
t = t+1; \\
\text{End }
\end{align*}
\]

The Structure of Branch and Bound Algorithm

According to Chinneck, (2010) the terminology used in structuring the branch and bound algorithm are given below:

1. **Node**: it represents incomplete or complete solution. Each task that could be scheduled based upon precedence and resource constraint

2. **Leaf (leaf node)**: it stands for the search consists of traversing the tree until the best root-to-leaf path is found; that is the solution is complete with all known variable values

3. **Bud (bud node)**: a partial solution, either feasible or infeasible. Think of it as a node that might yet grow further, just as on a real tree.

- **Bounding function**: by growing a bud node further, it gives the method of calculating approximately the best value of the objective function. Bounding function values are only connected from the bud nodes. Objective function values are contained in the Leaf nodes, these
gives actual values but not estimates. It is imperative that an optimistic estimator should be from the bounding function. Likewise, if you are minimizing, it must underestimate the actual achievable objective function at its best value; then, maximizing it must overestimate the achievable objective function at its best value. You want it to be as accurate an estimator as possible so that the resulting branch and bound tree is as small as possible, but it must err in the optimistic direction. The bounding function is the real magic in branch and bound. It takes ingenuity sometimes to find a good bounding function, however the payoff in amplified efficiency is marvelous. Every problem has its diverse conditions.

- **Branching, growing, or expanding a node:** the process of creating the child nodes for a bud node. One child node is created for each possible value of the next variable. For example, if the next variable is binary, there will be one child node associated with the value zero and one child node associated with the value one.

- **Incumbent:** “is the best complete feasible solution found so far”. When the solution process begins, there may not be an incumbent. In that instance, the first incumbent is taken as the first complete feasible solution found during the solution process.

The below BnB algorithm according to Pinedo, (1995) was used

**Step 1:** (Initial condition)

Ω := {Each job first operation}

\( r_{ij} := 0 \) for all \((i, j) \in \Omega\)

**Step 2:** (Machine selection)

Compute \( t(\Omega) \) for current partial schedule.

\( t(\Omega) := \min \{ r_{ij} + P_{ij} \} \)

\( i_{-} := \) machine on which the minimum is achieved.

**Step 3:** (Branching)

\( \Omega' := \{ (i_{-}, j)|r_{i-} < t(\Omega) \} \)

- For all \((i_{-}, j) \in \Omega'\), extend partial schedule by scheduling \((i, j)\) next on machine \(i\).
- For each such choice, delete \((i_{-}, j)\) from.
- Add job successor of \((i_{-}, j)\) to.
- Return to Step 2

**PROBLEM FORMULATION AND ANALYSIS**

This section provides a detailed description of the job shop scheduling problem dealt in this study. Considering \(n\) jobs \(i = 1, \ldots, n\) and \(m\) machines \(M_1, \ldots, M_m\). Each job \(i\) consists of a set of operations \(O_{ij} (j = 1, \ldots, n_i)\) with processing times \(p_{ij}\). Each operation \(O_{ij}\) must be processed on a machine \(\mu_{ij} \in \{M_1, \ldots, M_m\}\). All jobs operations may have precedence relationships. Each job can be processed by more than one machine at a time while each machine can only process one job at a time. The aim is to minimizes some objective function especially the mean flow time of the finishing times \(C_i\) of a feasible schedule of the jobs \(i = 1, \ldots, n\) of a local glass container factory. Obeying the constraints of precedence and resource availability such as furnace capacity, we got an optimal processing time for five production lines with the feasible tasks schedule.
The problem is denoted by

\[ J \prec \sum_{j=1}^{n} \left( \frac{C_{j}}{n} \right) \]  

(1)

Where;

\( J \) denotes a job shop with \( m \) machines,

\( \prec \) denotes precedence constrains on jobs

\( C_{j} \) denotes the completion time of job \( j \)

The following assumptions are considered while formulating the solution approach for the job shop scheduling problem:

- There exists a feasible schedule.
- All jobs are ready at the time of processing
- Preemption of jobs is not allowed
- Setup times are sequence-independent
- Setup times are added to processing times
- Shift Break times are not considered in processing times

For the purposes of this paper, the manufacturer supplied data were obtained from manufacturer for five actual jobs. These data were well defined. The job build variables in table below have a direct impact on the time required to complete a job.

**Table 1. Line Performance Parameters**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Line Performance Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>( W )</td>
<td>Average Operating Weight (g)</td>
</tr>
<tr>
<td>( V_s )</td>
<td>Machine Sectional Speed (BPM)</td>
</tr>
<tr>
<td>( T_p )</td>
<td>Actual Production Time (min)</td>
</tr>
<tr>
<td>( G_m )</td>
<td>Melted Gross Pull (Tons)</td>
</tr>
<tr>
<td>( G_p )</td>
<td>Packed Gross Pull (Tons)</td>
</tr>
<tr>
<td>( B_T )</td>
<td>Theoretical Gross Production</td>
</tr>
<tr>
<td>( B_A )</td>
<td>Actual Gross Production</td>
</tr>
<tr>
<td>( E )</td>
<td>Efficiency (%)</td>
</tr>
</tbody>
</table>

Decision Variables

\( p_{ik} \) processing time of job \( j \) on machine \( i \) (tonnes per day)

\( C_j \) Total Completion Time

\( m \) Number of Machines

\( n \) Number of jobs

\( O_{ij} \) set of operations \( O_{ij} \)

\( C_g \) Capacity of furnace in period \( t \) (tones)

\( I_k \) Stock of product \( k \) at the end of period \( t \)

\( I_k \) Backlog of product \( k \) at the end of period \( t \)

The optimal processing time after a pre-specified number of generation(s) was determined using Matlab Release 2007b (R2007b) running on an AMD Turion (tm) X2 Dual Core Mobile RM – 72 210GHz processor with 32 – bit operating system Windows 7 Home Premium.
EXPERIMENTAL RESULTS AND DISCUSSION

The results of the computational methods are given below for ten, twenty, thirty and forty operational runs. A run is a completed number of generation(s) or branches for genetic algorithm and branch and bound algorithm respectively. The results obtained here are not reproducible, that is, two runs of identical characteristics may not necessarily produce the same results. This is due to the random generators and the nodes pruning used in the techniques employed. Also, the nature of asexual genetic algorithm also increases the irreproducibility of the results.

Table 2. Computational Results for five operational runs

<table>
<thead>
<tr>
<th>No. of Run</th>
<th>GA</th>
<th>BnB</th>
<th>No. of Run</th>
<th>GA</th>
<th>BnB</th>
<th>No. of Run</th>
<th>GA</th>
<th>BnB</th>
<th>No. of Run</th>
<th>GA</th>
<th>BnB</th>
<th>No. of Run</th>
<th>GA</th>
<th>BnB</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>550</td>
<td>375</td>
<td>2</td>
<td>350</td>
<td>375</td>
<td>3</td>
<td>600</td>
<td>380</td>
<td>4</td>
<td>350</td>
<td>390</td>
<td>5</td>
<td>600</td>
<td>375</td>
</tr>
<tr>
<td>2</td>
<td>350</td>
<td>375</td>
<td>3</td>
<td>350</td>
<td>370</td>
<td>4</td>
<td>350</td>
<td>390</td>
<td>4</td>
<td>350</td>
<td>370</td>
<td>5</td>
<td>351</td>
<td>370</td>
</tr>
<tr>
<td>3</td>
<td>600</td>
<td>380</td>
<td>4</td>
<td>350</td>
<td>370</td>
<td>5</td>
<td>351</td>
<td>370</td>
<td>5</td>
<td>350</td>
<td>370</td>
<td>6</td>
<td>355</td>
<td>370</td>
</tr>
<tr>
<td>4</td>
<td>600</td>
<td>375</td>
<td>5</td>
<td>351</td>
<td>370</td>
<td>6</td>
<td>355</td>
<td>375</td>
<td>5</td>
<td>351</td>
<td>370</td>
<td>7</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>5</td>
<td>350</td>
<td>370</td>
<td>6</td>
<td>350</td>
<td>380</td>
<td>7</td>
<td>350</td>
<td>370</td>
<td>8</td>
<td>350</td>
<td>370</td>
<td>8</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>6</td>
<td>350</td>
<td>370</td>
<td>7</td>
<td>350</td>
<td>370</td>
<td>9</td>
<td>355</td>
<td>380</td>
<td>9</td>
<td>350</td>
<td>380</td>
<td>10</td>
<td>355</td>
<td>380</td>
</tr>
<tr>
<td>7</td>
<td>350</td>
<td>370</td>
<td>8</td>
<td>350</td>
<td>370</td>
<td>10</td>
<td>350</td>
<td>380</td>
<td>11</td>
<td>355</td>
<td>380</td>
<td>12</td>
<td>350</td>
<td>370</td>
</tr>
<tr>
<td>8</td>
<td>350</td>
<td>370</td>
<td>9</td>
<td>355</td>
<td>380</td>
<td>11</td>
<td>350</td>
<td>370</td>
<td>12</td>
<td>600</td>
<td>375</td>
<td>13</td>
<td>550</td>
<td>370</td>
</tr>
<tr>
<td>9</td>
<td>350</td>
<td>370</td>
<td>10</td>
<td>600</td>
<td>380</td>
<td>12</td>
<td>600</td>
<td>380</td>
<td>13</td>
<td>600</td>
<td>380</td>
<td>14</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>10</td>
<td>355</td>
<td>380</td>
<td>11</td>
<td>600</td>
<td>380</td>
<td>13</td>
<td>350</td>
<td>375</td>
<td>14</td>
<td>350</td>
<td>375</td>
<td>15</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>11</td>
<td>350</td>
<td>370</td>
<td>12</td>
<td>900</td>
<td>400</td>
<td>14</td>
<td>350</td>
<td>370</td>
<td>15</td>
<td>350</td>
<td>370</td>
<td>16</td>
<td>351</td>
<td>375</td>
</tr>
<tr>
<td>12</td>
<td>350</td>
<td>370</td>
<td>13</td>
<td>350</td>
<td>375</td>
<td>16</td>
<td>351</td>
<td>375</td>
<td>17</td>
<td>350</td>
<td>375</td>
<td>18</td>
<td>550</td>
<td>375</td>
</tr>
<tr>
<td>13</td>
<td>350</td>
<td>370</td>
<td>14</td>
<td>350</td>
<td>370</td>
<td>17</td>
<td>350</td>
<td>375</td>
<td>19</td>
<td>350</td>
<td>380</td>
<td>19</td>
<td>550</td>
<td>375</td>
</tr>
<tr>
<td>14</td>
<td>350</td>
<td>370</td>
<td>15</td>
<td>350</td>
<td>370</td>
<td>18</td>
<td>350</td>
<td>375</td>
<td>20</td>
<td>351</td>
<td>370</td>
<td>20</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>15</td>
<td>350</td>
<td>370</td>
<td>16</td>
<td>350</td>
<td>370</td>
<td>21</td>
<td>350</td>
<td>375</td>
<td>21</td>
<td>350</td>
<td>375</td>
<td>22</td>
<td>350</td>
<td>370</td>
</tr>
<tr>
<td>16</td>
<td>350</td>
<td>370</td>
<td>17</td>
<td>350</td>
<td>370</td>
<td>22</td>
<td>788</td>
<td>375</td>
<td>23</td>
<td>350</td>
<td>370</td>
<td>23</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>17</td>
<td>350</td>
<td>370</td>
<td>18</td>
<td>350</td>
<td>370</td>
<td>24</td>
<td>788</td>
<td>375</td>
<td>24</td>
<td>350</td>
<td>370</td>
<td>25</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>18</td>
<td>350</td>
<td>370</td>
<td>19</td>
<td>350</td>
<td>370</td>
<td>25</td>
<td>350</td>
<td>370</td>
<td>25</td>
<td>788</td>
<td>375</td>
<td>26</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>19</td>
<td>350</td>
<td>370</td>
<td>20</td>
<td>350</td>
<td>370</td>
<td>26</td>
<td>788</td>
<td>375</td>
<td>27</td>
<td>350</td>
<td>370</td>
<td>27</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>20</td>
<td>350</td>
<td>370</td>
<td>21</td>
<td>350</td>
<td>370</td>
<td>27</td>
<td>350</td>
<td>370</td>
<td>28</td>
<td>350</td>
<td>370</td>
<td>28</td>
<td>788</td>
<td>375</td>
</tr>
<tr>
<td>21</td>
<td>350</td>
<td>370</td>
<td>22</td>
<td>350</td>
<td>370</td>
<td>29</td>
<td>350</td>
<td>375</td>
<td>29</td>
<td>600</td>
<td>375</td>
<td>29</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>22</td>
<td>350</td>
<td>370</td>
<td>23</td>
<td>350</td>
<td>370</td>
<td>30</td>
<td>350</td>
<td>370</td>
<td>30</td>
<td>350</td>
<td>370</td>
<td>30</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>23</td>
<td>350</td>
<td>370</td>
<td>24</td>
<td>350</td>
<td>370</td>
<td>31</td>
<td>350</td>
<td>370</td>
<td>31</td>
<td>550</td>
<td>370</td>
<td>31</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>24</td>
<td>350</td>
<td>370</td>
<td>25</td>
<td>350</td>
<td>370</td>
<td>32</td>
<td>550</td>
<td>370</td>
<td>32</td>
<td>350</td>
<td>375</td>
<td>32</td>
<td>350</td>
<td>370</td>
</tr>
<tr>
<td>25</td>
<td>350</td>
<td>370</td>
<td>26</td>
<td>350</td>
<td>370</td>
<td>33</td>
<td>350</td>
<td>370</td>
<td>33</td>
<td>350</td>
<td>370</td>
<td>33</td>
<td>350</td>
<td>370</td>
</tr>
<tr>
<td>26</td>
<td>350</td>
<td>370</td>
<td>27</td>
<td>350</td>
<td>370</td>
<td>34</td>
<td>350</td>
<td>370</td>
<td>34</td>
<td>788</td>
<td>375</td>
<td>34</td>
<td>788</td>
<td>375</td>
</tr>
<tr>
<td>27</td>
<td>350</td>
<td>370</td>
<td>28</td>
<td>350</td>
<td>370</td>
<td>35</td>
<td>350</td>
<td>370</td>
<td>35</td>
<td>350</td>
<td>370</td>
<td>35</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>28</td>
<td>350</td>
<td>370</td>
<td>29</td>
<td>350</td>
<td>370</td>
<td>36</td>
<td>350</td>
<td>370</td>
<td>36</td>
<td>350</td>
<td>370</td>
<td>36</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>29</td>
<td>350</td>
<td>370</td>
<td>30</td>
<td>350</td>
<td>370</td>
<td>37</td>
<td>350</td>
<td>370</td>
<td>37</td>
<td>350</td>
<td>370</td>
<td>37</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>30</td>
<td>350</td>
<td>370</td>
<td>31</td>
<td>350</td>
<td>370</td>
<td>38</td>
<td>350</td>
<td>370</td>
<td>38</td>
<td>350</td>
<td>370</td>
<td>38</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>31</td>
<td>350</td>
<td>370</td>
<td>32</td>
<td>350</td>
<td>370</td>
<td>39</td>
<td>350</td>
<td>370</td>
<td>39</td>
<td>350</td>
<td>370</td>
<td>39</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>32</td>
<td>350</td>
<td>370</td>
<td>33</td>
<td>350</td>
<td>370</td>
<td>40</td>
<td>350</td>
<td>370</td>
<td>40</td>
<td>350</td>
<td>370</td>
<td>40</td>
<td>350</td>
<td>375</td>
</tr>
<tr>
<td>33</td>
<td>350</td>
<td>370</td>
<td>34</td>
<td>350</td>
<td>370</td>
<td>41</td>
<td>800</td>
<td>380</td>
<td>35</td>
<td>350</td>
<td>370</td>
<td>41</td>
<td>800</td>
<td>380</td>
</tr>
<tr>
<td>34</td>
<td>350</td>
<td>370</td>
<td>35</td>
<td>350</td>
<td>370</td>
<td>42</td>
<td>600</td>
<td>380</td>
<td>36</td>
<td>350</td>
<td>370</td>
<td>42</td>
<td>600</td>
<td>380</td>
</tr>
</tbody>
</table>
Briggs, Tobinson, A. “Job Shop Scheduling Application Using Genetic and Branch and Bound Algorithms”

<table>
<thead>
<tr>
<th>Number of generations</th>
<th>Processing time</th>
</tr>
</thead>
<tbody>
<tr>
<td>43</td>
<td>355</td>
</tr>
<tr>
<td>44</td>
<td>350</td>
</tr>
<tr>
<td>45</td>
<td>788</td>
</tr>
<tr>
<td>46</td>
<td>350</td>
</tr>
<tr>
<td>47</td>
<td>550</td>
</tr>
<tr>
<td>48</td>
<td>350</td>
</tr>
<tr>
<td>49</td>
<td>600</td>
</tr>
<tr>
<td>50</td>
<td>350</td>
</tr>
</tbody>
</table>

**Figure 1.** Plot of Processing Time Vs ten generations for GA

**Figure 2.** Plot of Processing Time Vs ten branches for BB
Briggs, Tobinson, A. “Job Shop Scheduling Application Using Genetic and Branch and Bound Algorithms”

**Figure 3.** Plot of Processing Time Vs twenty generations for GA

**Figure 4.** Plot of Processing Time Vs twenty branches

**Figure 5.** Plot of Processing Time Vs thirty generations
Briggs, T., Robinson, A. “Job Shop Scheduling Application Using Genetic and Branch and Bound Algorithms”

**Figure 6.** Plot of Processing Time Vs thirty branches for BB

**Figure 7.** Plot of Processing Time Vs forty generations for GA

**Figure 8.** Plot of Processing Time Vs forty branches for BB
From the computational analysis, genetic algorithm, GA has proved to obtain near optimal solutions with reasonable computational time than the branch and bound algorithm, BnB. With the high relative speed of computational GA possesses over the BnB algorithm, results were easily obtain.

BnB algorithm was found not able to solve large size problems unlike genetic algorithm which operates within large range of possible solutions. The operational runs for both techniques show that GA often gives the best minimum processing times for the various parallel machines. Also, it was observed that the minimum processing times given by GA for the five parallel machines were always same while the minimum processing times given by BnB were always slightly different with variations ranging from 0.5 to 5.0.

Therefore the research work has indicated the meta-heuristics nature of GA for solving hard optimization problems than BnB which is an exact method.
CONCLUSION AND RECOMMENDATION

This paper presents a model for optimizing production schedule with the objective of minimizing the “mean processing time” of various jobs in a current schedule for five uniform parallel machines based on the development and analysis of “genetic algorithm” and “branch and bound algorithm”.

Analyses were coded in genetic algorithm and branch and Bound algorithm and programmed with the aid of matlab software. Both computational methods improve on the current schedule by optimizing the mean processing time.

The two resource constraints were precedence and furnace capacity. The techniques calculated the optimal duration of the glass manufacturing process to be about three hundred and fifty (350) minutes for GA and three hundred and seventy (370) minutes for BnB for specific quantity of bottle products. One of the major advantages of using computational techniques is their relative speed of computation. With relatively low processing time, computational techniques can solve problems with a very large number of jobs in seconds. Also, other methods of solving scheduling problems (Critical Path and Project Evaluation Review Technique) cannot account for resource constraints.

Several comparisons were made for the minimum processing time and duration generated for ten, twenty, thirty and twenty operational runs. We observed that the optimal minimum processing time with the existing schedule was always obtained for ten and higher generational runs. The reproducibility of the G.A was calculated to be about 60%.

Many recommendations have been suggested for most works on genetic algorithm, including local searches in the components of the genetic algorithms to optimizing the operators of the algorithm to using an effective replacement strategy; several strategies exist for improving the results of a genetic algorithm. For this particular work, the following recommendations are been made for further studies:

- Using a complex task schedule with large number of jobs as this will emphasize the effectiveness of the algorithms.
- Including cross over operation in the analysis of genetic algorithm
- Including tardiness and earliness of activities in the analysis
- Compelling job shop scheduling with various computational techniques.

ACKNOWLEDGEMENT

I will like to appreciate students who spend time and help to carry out the field work. Also thank the Indigenous Glass Company who gives assistance in carrying out the study.

REFERENCES


Briggs, Tobinson, A. “Job Shop Scheduling Application Using Genetic and Branch and Bound Algorithms”


AUTHOR’S BIOGRAPHY

Dr. T. A. Briggs, obtained his Doctoral degree (Ph.D.) in Mechanical Engineering and a Postgraduate certificate in Advanced Studies in Academic Practice from University of Newcastle. Dr. Briggs has published some research papers in learned reputable journals both here and abroad. He has a high sense of research curiosities with a great intellectual drive. Currently the Head of Department and a lecturer of Mechatronic Engineering and Acting Director of Intellectual Property Technology Transfer Office (IPTTO).