Received 5 January 2016
Revised 31 May 2016
Accepted 6 July 2016

Heuristic production line balancing problem solution with MATLAB software programming

Aysun Türkmen and Yalcin Yesil
Department of Textile Engineering, Bartın University, Bartın, Turkey, and
Mahmut Kayar
Department of Textile Engineering, Marmara University, Istanbul, Turkey

Abstract

Purpose – The purpose of this paper is to find the most efficient assembly line balancing solution across many heuristic line balancing methods, in assistance with a developed computer program.
Design/methodology/approach – In this paper, assembly line balancing problem was analyzed using t-shirt and knitted pants data. A computer program using MATLAB software for the solution of assembly line balancing problems has been developed. In this study, following heuristic assembly line balancing methods were applied: Hoffman method; position weight method; COMSOAL method; and Kilbridge and Wester method. A MATLAB program has been developed by taking into account of theoretical solution of all these methods. Later the program is developed further by analyzing solutions made manually and is made to verify the developed program.
Findings – Pre-studies which were conducted in order to decide which programming language would be the best choice for line balancing methods' application came out with the result that MATLAB, from between C, C++, C# and Java, would be the best software choice. The main reason for this choice is that MATLAB is a powerful matrix operation software with a powerful user interface designing tool and has the tools to make development program to be used universally in every computer.
Originality/value – When the researches were investigated, it is clearly seen that, this study is the first research on using computer program for solving assembly line balancing problem.
Keywords MATLAB, COMSOAL method, Heuristic assembly line balancing methods, Hoffman method, Kilbridge and Wester method, Ranked positional weight
Paper type Research paper

1. Introduction

Generally speaking, an assembly line is a special production system, which consists of workstations lined up in order in relation to each other, where work packages are implemented on a product while it passes through these stations. Assembly lines are the places where several products and components are combined together and processed (Asar and Andrew, 2001).

Assembly line balancing is a process which is used for assigning tasks and work packages to workstations, while adapting to technology-based capacity constraints and priority relations, keeping loss of time at a minimum, and increasing line efficiency.

In the textile sector, assembly line balancing methods are used in order to ensure a regular machine flow, to maximize the efficiency of manpower usage and to make use of machine capacities at maximum levels. As the production capacities of garment enterprises are directly related to their sewing capacity, it is extremely important for them to have their assembly lines balanced for efficiency.

The authors would like to thank to Okan Türkmen for his support.

International Journal of Clothing Science and Technology
Vol. 28 No. 6, 2016
pp. 750-779
© Emerald Group Publishing Limited
0955-6222
DOI 10.1108/IJCST-01-2016-0002

In this study, the following heuristic assembly line balancing methods were applied:

- the Hoffman method;
- the positional weight method;
- the computer method of sequencing operations for assembly lines (COMSOAL) method; and
- the Kilbridge and Wester method.

A MATLAB program has been developed that takes account of the theoretical solutions of all these methods. Latterly, the program has been developed further by analyzing solutions made manually to verify the developed program's results. The latest version of the program is a generic tool, which is executable on all operating systems and does not require a MATLAB licence or any dedicated installation. Therefore, it is ready for use in industry to solve assembly line problems, and even people with rudimentary computer knowledge can use it.

The main purpose of this study is to find the most efficient assembly line balancing solution across many heuristic line balancing methods, with the assistance of a developed computer program.

1.1 Literature review

Researchers have studied the subject of balancing assembly lines in many different industrial areas. The first line balancing research was applied in the automotive sector. Assembly line balancing studies have also been conducted previously in the textile industry as well as in other industries.

When the history of the research concerning assembly line balancing is considered, it appears that the idea of assembly line balancing was originally suggested by Bryton (1954). The first published research was called, "The Assembly Line Balancing Problem." It was conducted by Salveson (1955). After this initial study, a great variety of research was conducted by academics who gave the assembly line balancing method its name. The names of the researchers that can be given as examples include Bowman (1960), Kilbridge and Wester (1961), Helgeson and Birnie (1961), Tonge (1961), Hoffman (1963), Moodie and Young (1965), Arcus (1966), Gehrlein and Patterson (1975). In subsequent years, work in this field was undertaken by F.B. Talbot and J.H. Patterson (1984), Talbot *et al.* (1986), Agrawal (1985), El-Sayed and Boucher (1985), Baybars (1986) and Hoffman (1990).

When the studies of assembly line balancing in the ready-to-wear industry are reviewed, it emerges that in a study which was conducted by Baskak (1998), a new method was developed for addressing assembly line balancing problems.

In the studies conducted by Eryuruk and his colleagues, a ready-to-wear assembly line balancing study was carried out by applying the probabilistic line balancing technique developed by El-Sayed and Boucher and the ranked positional weight technique developed by Helgeson and Birnie (Eryuruk *et al.*, 2008, 2011).

In a study conducted by Dundar *et al.* (2012), a ready-to-wear assembly line balancing study was conducted by using graph theory.

In a study conducted by Guner *et al.* (2013) the longest operation time method, the ranked positional weight method, the shortest operation time method, the most following tasks method and the fewest following tasks method were used.

In another study carried out by Eryuruk, a ready-to-wear assembly line study was executed by applying the largest set rule algorithm developed by Agrawal and the probabilistic line balancing technique developed by El-Sayed and Boucher (Eryuruk, 2012).

In a study carried out by Kayar (2008), assembly line balancing was conducted by applying the Hoffman method and the classical method on a ready-to-wear assembly line. Another study by Kayar and his colleagues included the application of different heuristic line balancing methods. For this study, the positional weight method has been used in order to inspect the effects of method analysis on production volume and line efficiency (Kayar and Akyalçın, 2014).

2. Experimental

In this research, the t-shirt and knitted pants productions of a garment company, have been selected and analyzed theoretically. The models of the analyzed t-shirt (a) and knitted pants (b) are shown in Figure 1.

The t-shirt, which is shown in Figure 1(a), consists of five parts including a front (2), a back, sleeves (2), tape (1) and a collar (1). The t-shirt is produced when parts are treated in appropriate machines according to an operation order. Figure 2 shows the production flow that is necessary for producing this t-shirt.

The knitted pants which are shown in Figure 1(b) consist of ten parts including a front (2), a back (2), a back pocket (1), a lower pocket bag (1), an upper pocket bag (1), a welt (1), a waistband (1) and a side band (1). The knitted pants are produced when parts are treated in appropriate machines according to an operation order. Figure 3 shows the production flow that is necessary for producing the knitted pants.

2.1 Time study

All operation durations were measured by using a stopwatch to determine the standard time for the production of a t-shirt and for sewing knitted pants.

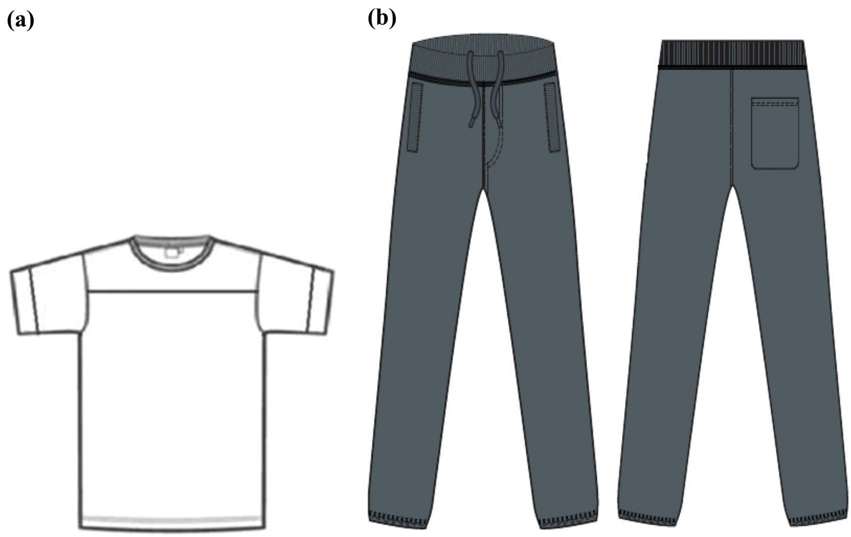

Figure 1. Models of a t-shirt and knitted pants

Notes: (a) T-shirt; (b) knitted pants

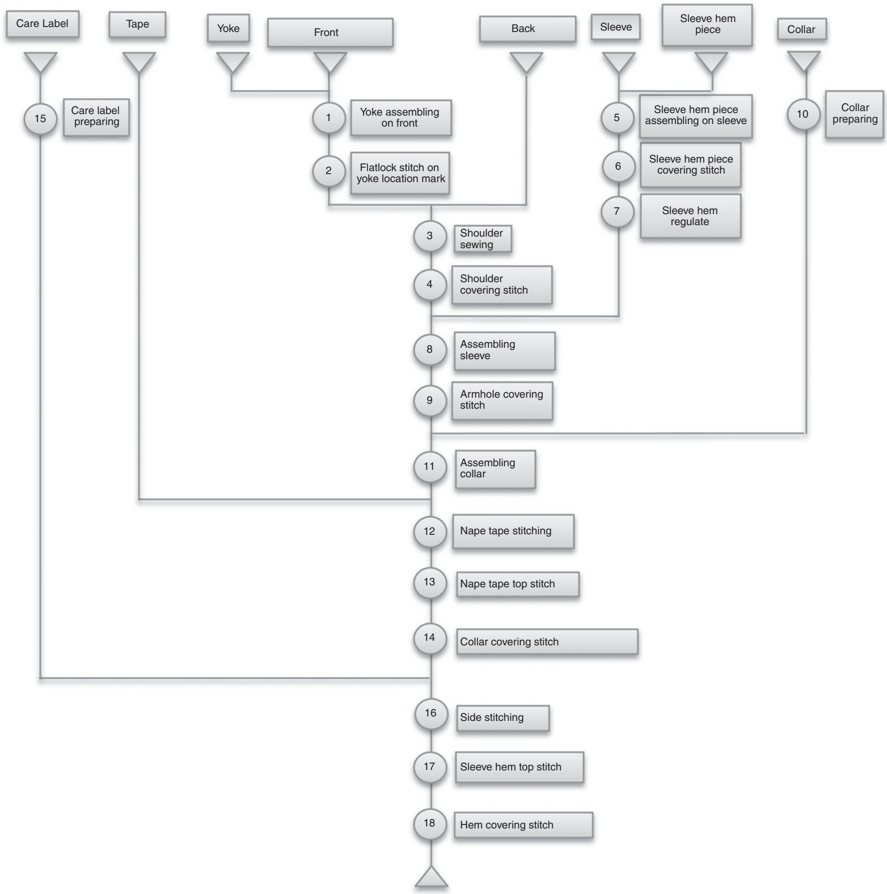

Figure 2.
Operations and flow chart of the operations in t-shirt production

The measurements were made as percentage-minutes (PM) and they were turned into minutes (PM/60) by calculating their arithmetic means.

While operation durations were being measured, a performance assessment was made for each operation.

During the interview that was conducted with executives of the company at which the t-shirts and knitted pants are produced, it was stated that the tolerance share was calculated at 15 percent as a result of previous measurements and this rate was used to estimate the standard time for production.

Afterwards the standard time was calculated for each operation by using the formula shown below:

$$ST = MT \times R + MT \times R \times t$$

where ST is standard time, MT is measured time, R is performance and t is tolerance (Kayar, 2003).

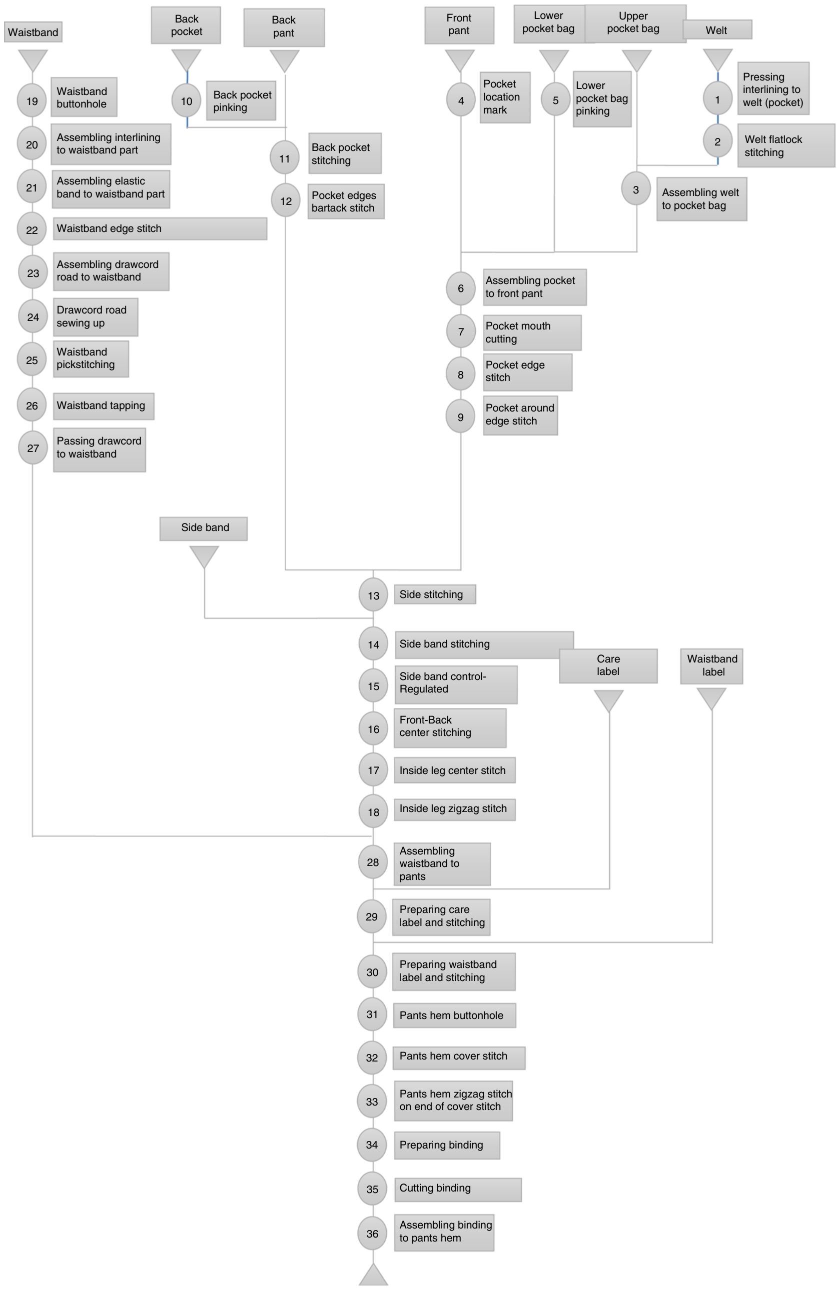

Figure 3.
Flow chart of the operations in knitted pants production

The durations obtained as a result of the measurements which were made for each operation by considering tolerance share, performed performance assessments and the arithmetic mean of performance rates in terms of PM, which were measured by using a stopwatch.

2.2 Line balancing methods with MATLAB programming

It is a fact that line balancing methods are analytical operations and most of them include matrix calculations. These facts make suitable the application of line balancing methods to computer programming. With the use of computer programming, one can secure the following benefits:

- a reduction of error possibilities while method matrices are being produced;
- a reduction of the time needed for creating method matrices;
- in addition to the efficiency improvement that method analysis gives to processes, it also provides efficiencies that enable method analysis itself;
- additional improvements or changes in lines can be inspected and calculated using the program repeatedly without causing any loss of time or productivity; and
- the outputs and documentation of method implementation can be produced without any effort.

In order to transfer line balancing methods to programming, MATLAB software was used because of its mathematical and matrix calculations and its speed. MATLAB, which is an abbreviation of "matrix laboratory", is a multi-paradigm numerical computing application. It is built around the MATLAB scripting language which allows matrix manipulations, the plotting of functions and data, the implementation of algorithms, the creation of user interfaces and interfacing with other programming languages (MATLAB Documentation, 2016).

MATLAB's "Open GUI Layout Editor", the user interface preparation application also known as GUIDE, was used in order to create a user-friendly interface, and facilitate users' inputs and outputs to the program.

What is more, in order to speed up and ease the input of information about processes and the output of the method results, Excel files were utilized. With this utility, users do not have to enter line information more than once, and can change this information easily if required.

The methods' fundamental calculations were located in an "*.m" file which contained MATLAB command lines. This command file was used in order to enter several commands together and allowed the tracking of the program. All methods were programmed in a modular way, and in this way methods could be used separately or together as desired.

Additionally, the program was processed on MATLAB's runtime compiler and an executable file was created which could be used independently without MATLAB. There was no need to have MATLAB installed on a computer and no licence was required to use the program.

The observed advantages of MATLAB over other programming languages used during studies were:

- it is easy to use and has powerful interfaces;
- it is highly documented and it is easy to get feedback for problems;
- it is mathematically flexible, especially with respect to matrix calculations; and
- it is universal and can be used on all operating systems.

In order to describe the algorithm of the program easily and quickly, a flow diagram can be used. The program can be inspected in two separate flow diagrams in

Figures 4 and 5. While the first one presents the user interface, the second one presents the method calculations and uses the Hoffman method's implementation.

Lastly, as the complete program consisted of 850 lines, only core information about the programming can be described by most critical points of the code, which are given below:

- The Hoffman matrix's calculation in the loop is shown below. It uses two main counters, one outer loop for indexing columns and one inner loop for indexing rows. The code flow can also be tracked in Figure 5's left-hand-side of the flow diagram:

```
matrixHoffman=zeros(sizeX,sizeX); % Firstly create the
matrix as all zeroes
for i=1:1:sizeX %Hoffman matrix column is
indexed here
      txtSeperator=txtOnceki{i,1}; %Get the pre op's
      from user input
      [~,sizeTxt]=size(txtSeperator); %
      separate for inspection
      for j=1:1:sizeTxt
          if txtSeperator(j)=='O'
             txtExporter(1)=txtSeperator(j+1);
             txtExporter(2)=txtSeperator(j+2);
```

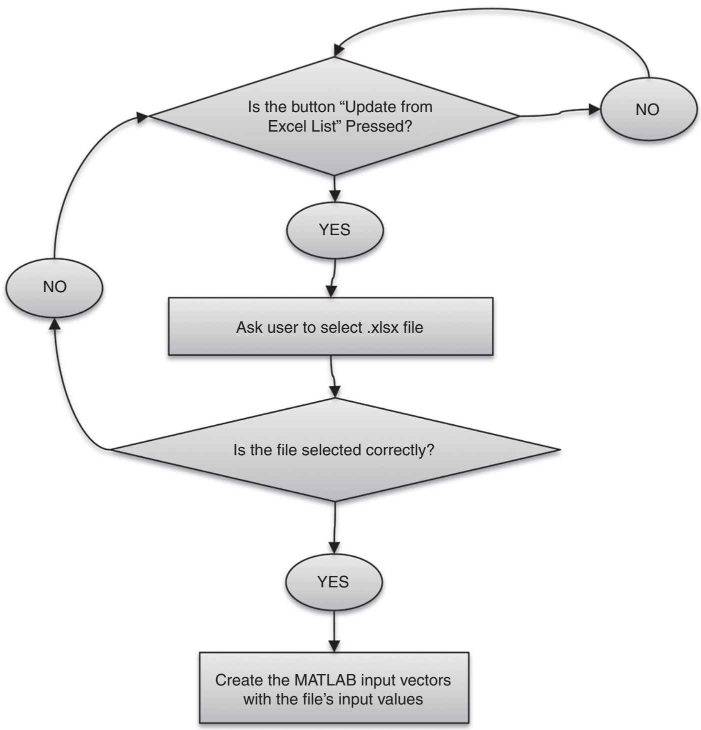

Figure 4.
The user interface flow diagram

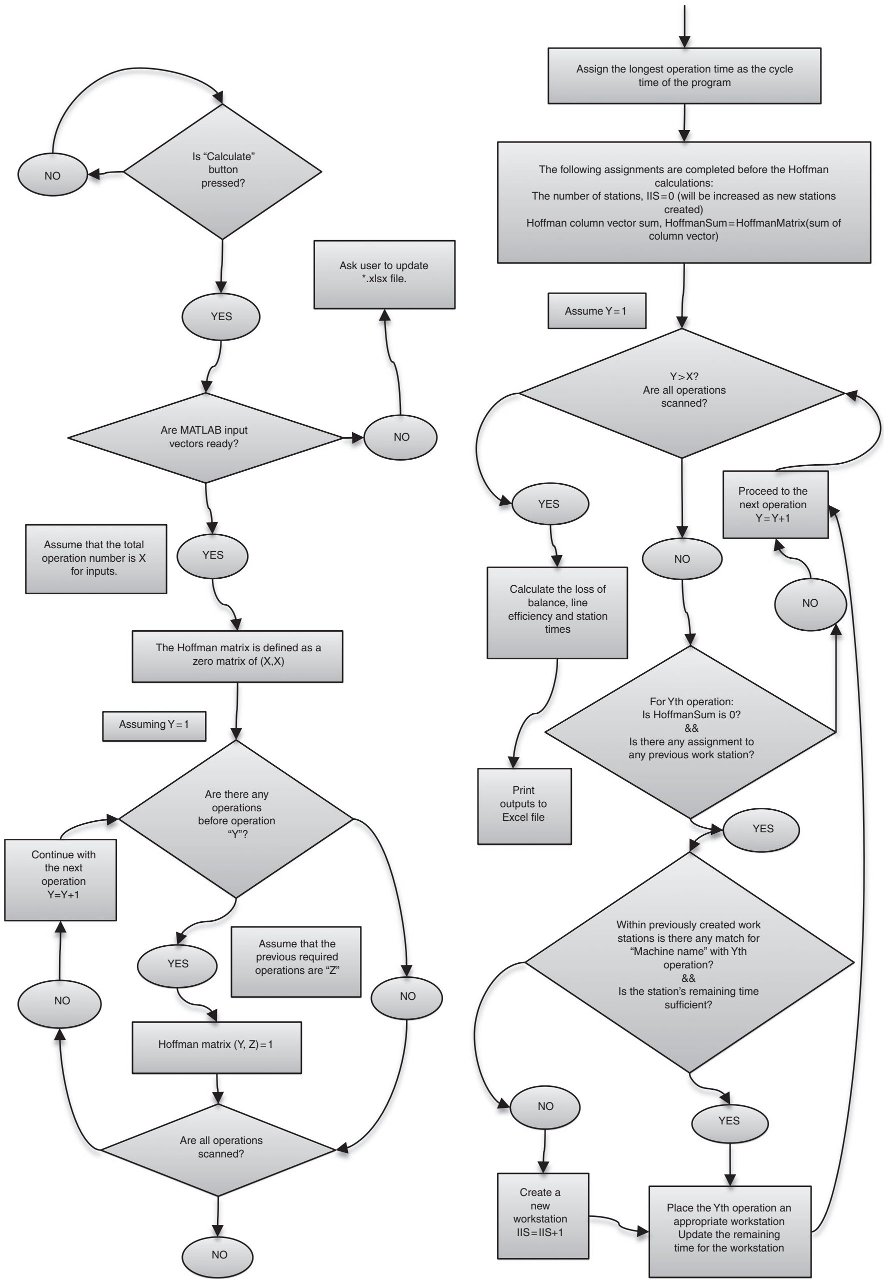

Figure 5.
Method applications' flow diagram

```
numExporter=str2double(txtExporter); %
Convert txt to num
for l=1:1:sizeX % Hoffman matrix row is
indexed here
```

```
                    if numExporter==1 %If scanned op.
                    required before current
                        matrixHoffman(l,i)=1; %Set
                        matrix value to ''1''
                    end
                end
            end
        end
end
```

- Line balancing calculation codes for the Hoffman method are given below. The code flow can also be tracked in Figure 5's right-hand-side of the flow diagram:

```
arrayOperationsInWorkspace=zeros(sizeX,sizeX); %Define
an array for workspace
    matrixDeltaHoffman=matrixHoffman; %Take the hoffman
    matrix to buffer
    numWS=0; %set current number of WS's to ''0''
    sumHoffman=sum(matrixDeltaHoffman); %calculate the
    sum of columns
    counterA=1;%Will be used for matrix shrinking counter
    starting from ''1''
    while sum(sumHoffman)~=-1% Loop for matrix shrinking
    using counterA
        eliminator=transpose(sum(transpose
        (arrayOperationsInWorkspace)));
        createWSonce=1; %For every matrix only one WS to
        be created
        for counterSutun=1:1:sizeX % Scanning
        the matrix.
            if sumHoffman(counterSutun)==0 &&
            eliminator(counterSutun)==0% will
            only consider the columns with the sum
            of 0 and the ones with no assignment has
            been conducted.
                %---1st Stage of Calculations---
                flag=0;
                for counterWS=1:1:numWS % At first we
                will be checking the
                availability of previously created
                WS's. If available assign the
                operation to Ws.
                    if (tRemaining(counterWS)-
                    numOpT(counterSutun,1)>=0) &&
                    (strcmp(txtWSMakina
                    {counterWS,1},txtMakina
                    {counterSutun,1}) || strcmp
                    (txtMakina
                    {counterSutun,1},'Elisi'))
```

```
            tRemaining
            (counterWS)=tRemaining
            (counterWS)-numOpT
            (counterSutun,1);
arrayOperationsInWorkspace(counterSutun,
counterWS)=1;
            matrixDeltaHoffman
            (counterSutun,:)=0;
            flag=1;
            break; % If WS is handwork and
            if we can add a new machine,
            assign.
        elseif (tRemaining(counterWS)-
        numOpT(counterSutun,1)>=0) &&
        (strcmp(txtWSMakina
        {counterWS,1},'Elisi')==1) &&
        (strcmp(txtMakina
        {counterSutun,1},'Elisi')~=1)
            tRemaining
            (counterWS)=tRemaining
            (counterWS)-numOpT
            (counterSutun,1);
            arrayOperationsInWorkspace
            (counterSutun,
            counterWS)=1;
            txtWSMakina
            {counterWS,1}=txtMakina
            {counterSutun,1};
            matrixDeltaHoffman
            (counterSutun,:)=0;
            flag=1;
            break;
        end
    end
            if flag==1%If WS is found,
            continue with remaining
            operations.
        continue;
    end

    %If the WS could not be found create a new WS
    and assing operation there.
    if createWSonce==1
        numWS=numWS+1;
        tRemaining(numWS)=tCycle-numOpT
        (counterSutun,1);
        arrayOperationsInWorkspace
        (counterSutun,numWS)=1;
```

```
                txtWSMakina{numWS,1}=txtMakina
                {counterSutun,1};
                matrixDeltaHoffman
                (counterSutun,:)=0;
                createWSonce=0;
            % Breaking the code/loop is not available
            here, it is required to scan the remaining
            columns. But not to enter to this if
            statement anymore just one WS is required
            in the start phase.
              end
              %---1st Stage of Calculations ends
              here---
              end
          end
          counterA=counterA+1;
          if sumHoffman==0
              break;
          end
          sumHoffman=sum(matrixDeltaHoffman);
      end
```

2.3 Assembly line balancing studies

2.3.1 Application for t-shirt production. Assembly line balancing studies were carried out according to the t-shirt production process which consists of 18 operations that are shown in Figure 6. The operation time for t-shirt production, and the machines which were used during this operation and during previous operations are shown in Table I.

As is shown in Table I, t-shirt sewing on assembly lines in which manual machines operated involved 18 operations and the total sewing duration time of the t-shirt was 10.02 minutes.

The duration of a workstation cannot be shorter than the longest duration of a work unit and it cannot be longer than the cycle time (Acar and Estas, 1991). Because of this principle, the cycle time in assembly line balancing studies is accepted as 0.97 minutes. Loss of balance on assembly lines, their efficiency and their daily total production amount is estimated by using the formulas which are given below:

$$LB = \left[\left(nC - \sum C_o\right)/nC\right)\right]100$$

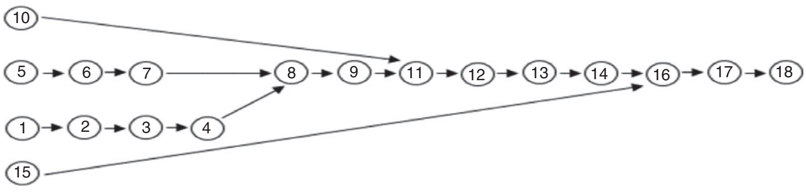

Figure 6.
Priority diagram for t-shirt sewing

Operation number	Operations	Machine type	Operation times (min)	Previous operation
1	Yoke assembling on front	3-thread overlock	0.50	–
2	Flatlock stitch on yoke	Flatlock machine	0.50	1
3	Shoulder sewing	3-thread overlock	0.50	2
4	Shoulder cover seam	Blade cover stitch machine	0.48	3
5	Sleeve hem piece assembling on sleeve	Lock-stitch sewing machine	0.97	–
6	Sleeve hem piece covering stitch	Blade cover stitch machine	0.95	5
7	Sleeve hem regulate	Handmade	0.94	6
8	Assembling sleeve	4-thread overlock	0.60	4-7
9	Armhole covering stitch	Blade cover stitch machine	0.55	8
10	Collar preparing	Lock-stitch sewing machine	0.30	–
11	Assembling collar	4-thread overlock	0.55	9-10
12	Nape tape stitching	Binding cover stitch machine	0.30	11
13	Nape tape top stitch	Lock-stitch sewing machine	0.60	12
14	Collar covering stitch	Blade cover stitch machine	0.55	13
15	Care label preparing	Lock-stitch sewing machine	0.10	–
16	Side seam	4-thread overlock	0.60	14-15
17	Sleeve hem top stitch	Lock-stitch sewing machine	0.48	16
18	Hem covering stitch	Blade cover stitch machine	0.55	17
Total time	10.02			

Table I. Operation times, used machine types and previous operations for t-shirt sewing

$$LE = (1-LB)100$$

$$PA = T/C$$

where LB is loss of balance, LE is line efficiency, C is cycle time, n is the total number of workstations, C_o is the average workstation time, PA is the daily total production amount and T is the daily total production time (Kayar, 2008).

For all the assembly line balancing studies, which were carried out within the scope of this study, it was supposed that handwork operations were done by all operators on condition that operations were done by the same types of machines.

2.3.1.1 Hoffman method. The first step for the Hoffman method's implementation was to define the priority matrix shown in Table II. The priority matrix was built up using the priority diagram.

After the priority matrix had been formed, the first step was to concentrate on the operations columns, which had the sum of "0." In fact, these operations were the ones that did not have any operation being completed before them and so they were candidates for starting points. For the matrix in this case, the first, fifth, tenth and 15th operations did not have any operations before them. The first operation, yoke assembling on the front, was selected to be processed by order. The cycle time of the method was selected to be the duration of the longest operation time, $C = 0.97$ in this case.

When the first operation was assigned to the first workstation, the remaining time for this workstation was $C - t_1 = 0.47$ minutes. As there were no other suitable operations or columns which had the sum of "0" and which used the same machine

Table II.
Solution matrix 1-3

(a)

Op.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	1	1	1	0	1	1	2	1	0	2	1	1	1	0	2	1	1

(b)

Op.	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

(*continued*)

11	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	1	1	0	1	1	2	1	0	2	1	1	1	0	2	1	1

(c)

Op.	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	1	0	1	1	2	1	0	2	1	1	1	0	2	1	1

Table II.

type, the first workstation was left with only the first operation for now. For the next step, the operations used for the first workstation were removed from the priority matrix and a new matrix was formed. The new priority matrix is shown in Table II.

With respect to the second matrix, the second operation's column sum was "0" and was a candidate for the second workstation. After the second operations assignment to the second workstation, the remaining time was calculated as $C-t_2 = 0.47$ minutes and the machine to be used for this workstation was a flatlock. As with the remaining operations with a column sum of "0", there was not any suitable candidate that fit the remaining timing using the same machinery so this workstation was left as it was for the time being. The priority matrix was updated by the deletion of the second operation. The third matrix is also shown in Table II. With respect to this matrix, as for the previous ones, the third workstation's assignment was performed with the third operation. The remaining workstation duration was $C-t_3 = 0.47$ minutes.

As can be seen in the assignment example which was made for the first and second workstations, one could achieve a solution. The solution results, having designed the assembly line by using the Hoffman method, are shown in Table III.

As can be deduced from Table III, the assembly line was designed according to it having a 0.97-minute cycle time and 16 workstations. The loss of balance and the assembly line efficiency of the designed assembly line are shown below:

$$LB = \left[\left(nC - \sum C_o\right)/nC\right)\right]100 = 35.438$$

$$LE = (1 - LB)100 = 64.562$$

The program output screen and results table taken from the program are given below, and these correlated with the manually calculated results (Figure 7).

Table III. Line balancing results – Hoffman Method

Workstation number	Operation number	Machine type	Time (min)	Total time for workstation (x)	Remaining time ($C-x$)
1	1	3-thread overlock	0.50	0.50	0.47
2	2	Flatlock machine	0.50	0.50	0.47
3	3	3-thread overlock	0.50	0.50	0.47
4	4	Blade cover stitch machine	0.48	0.48	0.49
5	5	Lock-stitch sewing machine	0.97	0.97	0
6	6	Blade cover stitch machine	0.95	0.95	0.02
7	7	Handmade	0.94	0.94	0.03
8	8	4-thread overlock	0.60	0.60	0.37
9	9	Blade cover stitch machine	0.55	0.55	0.42
10	10	Lock-stitch sewing machine	0.30	0.88	0.09
	15		0.10		
	17		0.48		
11	11	4-thread overlock	0.55	0.55	0.42
12	12	Binding cover stitch machine	0.30	0.30	0.67
13	13	Lock-stitch sewing machine	0.60	0.60	0.37
14	14	Blade cover stitch machine	0.55	0.55	0.42
15	16	4-thread overlock	0.60	0.60	0.37
16	18	Blade cover stitch machine	0.55	0.55	0.42

Figure 7.
Line balancing
results

On the second page of results, from the programs output file, the following Table IV presents the calculated results for the Hoffman method, which was also in correlation with the manually calculated results.

2.3.1.2 Ranked positional weight method. In order to apply the ranked positional weight method, the first step was to prepare the ranked positional weight table which is shown as Table V.

This table contains the operation numbers in the first column, the operation durations in the second column and the priority factors in columns at its center. The number 1s in the table represent the operations that needed to be completed before the operation defined in that column can be processed. Additionally, keeping in mind the previous operations, "+" symbols have been placed as is relevant.

The positional weight value is calculated for each row by adding the durations of operations which will occur until all the operations are completed. The resulting value is written to the right side of the table for each operation.

For the assignments to workstations, the operation with the highest positional weight value was selected and placed at the first workstation. For this application the operation with the highest value was the fifth operation, which had a weight of 7.64 minutes. After this assignment no other operations could be assigned to this workstation as the remaining duration and machine-type requirements could not be satisfied.

In the next step, operation number one with the highest weighted value of 6.76 minutes was selected and assigned to the second workstation. After this assignment, no other operations could be assigned to that workstation as the remaining duration and machine-type requirements could not be satisfied.

In the following steps the next operation's assignments were performed like the first and second workstations' were until all the operations were completed. After all the assignments were complete the following line balancing results shown in Table VI were obtained.

With respect to Table VI, the assembly lines' cycle time was defined as 0.97 minutes and a set-up with a total of 17 workstations was suggested. The calculations for loss of balance and line efficiency can be found below:

$$LB = \left[\left(nC - \sum C_o\right)/nC\right)\right] 100 = 39.235$$

$$LE = (1 - LB)100 = 60.765$$

2.3.1.3 COMSOAL method. To apply this method, the table which is shown below had to be designed (Table VII). In the first column of the table, operation numbers are shown. In the second column, the amounts of the previous operation are shown. In the third column, operations without previous operations are given.

Table IV. Solution of problem using the Hoffman method

Hoffman results					
Cycle time (min)	Total number of workstations	Total time for all workstations (min)	Total remaining time (min)	Loss of balance (%)	Line efficiency (%)
0.97	16	10.02	5.5	35.43814433	64.56185567

Table V.
Solution of problem using by ranked positional weight method

Operation number	Time (min)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	Ranked positional weight value
1	0.50	0	1	+	+	0	+	+	+	+	0	+	+	+	+	0	+	+	+	6.76
2	0.50	0	0	1	+	0	+	+	+	+	0	+	+	+	+	0	+	+	+	6.26
3	0.50	0	0	0	1	0	+	+	+	+	0	+	+	+	+	0	+	+	+	5.76
4	0.48	0	0	0	0	0	+	+	1	+	0	+	+	+	+	0	+	+	+	5.26
5	0.97	0	0	0	0	0	1	+	+	+	0	+	+	+	+	0	+	+	+	7.64
6	0.95	0	0	0	0	0	0	1	+	+	0	+	+	+	+	0	+	+	+	6.67
7	0.94	0	0	0	0	0	0	0	1	+	0	+	+	+	+	0	+	+	+	5.72
8	0.60	0	0	0	0	0	0	0	0	1	0	+	+	+	+	0	+	+	+	4.78
9	0.55	0	0	0	0	0	0	0	0	0	0	1	+	+	+	0	+	+	+	4.18
10	0.30	0	0	0	0	0	0	0	0	0	0	1	+	+	+	0	+	+	+	3.93
11	0.55	0	0	0	0	0	0	0	0	0	0	0	1	+	+	0	+	+	+	3.63
12	0.30	0	0	0	0	0	0	0	0	0	0	0	0	1	+	0	+	+	+	3.08
13	0.60	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	+	+	+	2.78
14	0.55	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	+	+	2.18
15	0.10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	+	+	1.73
16	0.60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	+	1.63
17	0.48	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1.03
18	0.55	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.55

Workstation number	Operation number	Machine type	Time (min)	Total time for workstation (x)	Remaining time ($C–x$)
1	5	Lock-stitch sewing machine	0.97	0.97	0
2	1	3-thread overlock	0.50	0.50	0.47
3	6	Blade cover stitch machine	0.95	0.95	0.02
4	2	Flatlock machine	0.50	0.50	0.47
5	3	3-thread overlock	0.50	0.50	0.47
6	7	Handmade	0.94	0.94	0.03
7	4	Blade cover stitch machine	0.48	0.48	0.49
8	8	4-thread overlock	0.60	0.60	0.37
9	9	Blade cover stitch machine	0.55	0.55	0.42
10	10	Lock-stitch sewing machine	0.30	0.40	0.57
	15		0.10		
11	11	4-thread overlock	0.55	0.55	0.42
12	12	Binding cover stitch machine	0.30	0.30	0.67
13	13	Lock-stitch sewing machine	0.60	0.60	0.37
14	14	Blade cover stitch machine	0.55	0.55	0.42
15	16	4-thread overlock	0.60	0.60	0.37
16	17	Lock-stitch sewing machine	0.48	0.48	0.49
17	18	Blade cover stitch machine	0.55	0.55	0.42

Table VI. Line balancing results – Ranked Positional Method

(a)			(b)		
Operation number	APO	OWPO	Operation number	APO	OWPO
1	0	1	2	0	2
2	1	5	3	1	5
3	1	10	4	1	10
4	1	15	5	0	15
5	0	–	6	1	–
6	1	–	7	1	–
7	1	–	8	2	–
8	2	–	9	1	–
9	1	–	10	0	–
10	0	–	11	2	–
11	2	–	12	1	–
12	1	–	13	1	–
13	1	–	14	1	–
14	1	–	15	0	–
15	0	–	16	2	–
16	2	–	17	1	–
17	1	–	18	1	–
18	1	–			

Table VII. Solution stages of problems using the COMSOAL method

While assignments for workstations were being made, any operation amongst the operations which were written in the third column was chosen randomly. The selected operation was deleted from the first column and the table was schemed again. Factors which occurred immediately after the chosen operation and had not any other factors following them were added to the third column. This procedure continued until the cycle time at the station and the work factors ran short and they were not able to assign new factors. After that, assignments were made to the next stations.

In Table VII, the first two steps for applying the method are given as in the examples.

In the following steps, as per the first row's elimination, COMSOAL tables for all the workstations were prepared and operation assignments were performed in the same way.

Lastly, with the assignment of the last operation to workstations the following results table was calculated using the COMSOAL method.

With respect to Table VIII, the cycle time of the line was defined as 0.97 minutes. A total of 16 workstations were suggested as the output for the method. The loss of balance and assembly line efficiency of designed the assembly line are shown below:

$$LB = \left[\left(nC - \sum C_o\right)/nC\right)\right]100 = 35.44$$

$$LE = (1 - LB)100 = 64.56$$

2.3.1.4 Kilbridge and Wester method. To apply this method, the figure shown below (Figure 8) was schemed according to the priority diagram in Figure 6.

Workstation number	Operation number	Machine type	Time (min)	Total time for workstation (x)	Remaining time (C–x)
1	1	3-thread overlock	0.50	0.50	0.47
2	2	Flatlock machine	0.50	0.50	0.47
3	3	3-thread overlock	0.50	0.50	0.47
4	4	Blade cover stitch machine	0.48	0.48	0.49
5	5	Lock-stitch sewing machine	0.97	0.97	0
6	6	Blade cover stitch machine	0.95	0.95	0.02
7	7	Handmade	0.94	0.94	0.03
8	8	4-thread overlock	0.60	0.60	0.37
9	9	Blade cover stitch machine	0.55	0.55	0.42
10	10	Lock-stitch sewing machine	0.30	0.88	0.09
	15		0.10		
	17		0.48		
11	11	4-thread overlock	0.55	0.55	0.42
12	12	Binding cover stitch machine	0.30	0.30	0.67
13	13	Lock-stitch sewing machine	0.60	0.60	0.37
14	14	Blade cover stitch machine	0.55	0.55	0.42
15	16	4-thread overlock	0.60	0.60	0.37
16	18	Blade cover stitch machine	0.55	0.55	0.42

Table VIII. Line balancing results – Comsoal Method

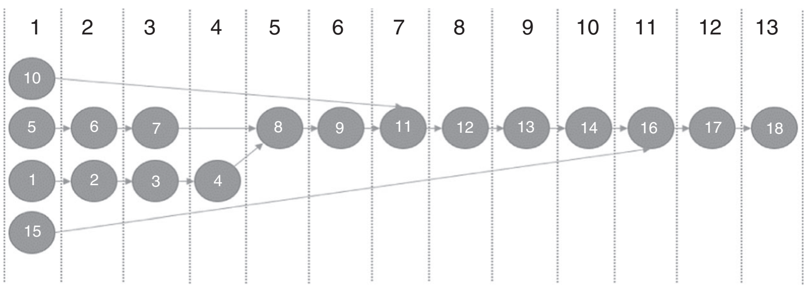

Figure 8. Priority diagram for the Kilbridge and Wester method

As can be interpreted from Figure 8, 13 workstations were required initially. In other words, 13 columns corresponded to 13 workstations. If 13 workstations were used, assembly line productivity decreased since the total time of the workstations was below the cycle time of the workstations. Consequently, the operations which could be carried out in different columns without disordering the process had to be taken into account. In the (c) column of Table IX, the columns to which an operation could be assigned are shown.

To assign elements to workstations, we began with the column 1 elements and continued with the assignment procedure in order of column number until the cycle time was reached. The results of the solution for the assembly line balancing study are shown in Table X.

With respect to Table X, the assembly lines' cycle time was defined as 0.97 minutes and a set-up with a total of 17 workstations was suggested. The calculations for the loss of balance and line efficiency can be found below:

$$LB = \left[\left(nC - \sum C_o\right)/nC\right)\right] 100 = 39.235$$

$$LE = (1 - LB)100 = 60.76$$

2.3.2 Application for knitted pants production. Assembly line balancing studies were carried out according to the knitted pants production which consisted of 36 operations which are shown in Figure 9.

This example had significantly more operations than that for t-shirts and it was obvious that the manual solution was hard to handle and time-consuming. The operation time for knitted pants production, and the machines which were used during this operation and previous operations are shown in Table XI.

As is shown in Table XI, sewing knitted pants in an assembly line in which manual machines are operated involved 36 operations and a total sewing duration of 10.42 minutes.

Table IX.
Operations arranged according to columns

Column number (a)	Operation(s) number (b)	Transfer (c)	Time (min) (d)	Total time for workstation (e)	Total time (min) (f)
1	1	–	0.50	1.87	1.87
	5	–	0.97		
	10	2-3-4-5-6	0.30		
	15	2-3-4-5-6-7-8-9-10	0.10		
2	2	–	0.50	1.45	3.32
	6	–	0.95		
3	3	–	0.50	1.44	4.76
	7	4	0.94		
4	4	–	0.48	0.48	5.24
5	8	–	0.60	0.60	5.84
6	9	–	0.55	0.55	6.39
7	11	–	0.55	0.55	6.94
8	12	–	0.30	0.30	7.24
9	13	–	0.60	0.60	7.84
10	14	–	0.55	0.55	8.39
11	16	–	0.60	0.60	8.99
12	17	–	0.48	0.48	9.47
13	18	–	0.55	0.55	10.02

Workstation number	Operation number	Machine type	Time (min)	Total time for workstation (x)	Remaining time (C–x)
1	1	3-thread overlock	0.50	0.50	0.47
2	2	Flatlock machine	0.50	0.50	0.47
3	3	3-thread overlock	0.50	0.50	0.47
4	4	Blade cover stitch machine	0.48	0.48	0.49
5	5	Lock-stitch sewing machine	0.97	0.97	0
6	6	Blade cover stitch machine	0.95	0.95	0.02
7	7	Handmade	0.94	0.94	0.03
8	8	4-thread overlock	0.60	0.60	0.37
9	9	Blade cover stitch machine	0.55	0.55	0.42
10	10	Lock-stitch sewing machine	0.30	0.40	0.57
	15		0.10		
11	11	4-thread overlock	0.55	0.55	0.42
12	12	Binding cover stitch machine	0.30	0.30	0.67
13	13	Lock-stitch sewing machine	0.60	0.60	0.37
14	14	Blade cover stitch machine	0.55	0.55	0.42
15	16	4-thread overlock	0.60	0.60	0.37
16	17	Lock-stitch sewing machine	0.48	0.48	0.49
17	18	Blade cover stitch machine	0.55	0.55	0.42

Table X. Line balancing results – Kilbridge and Western Method

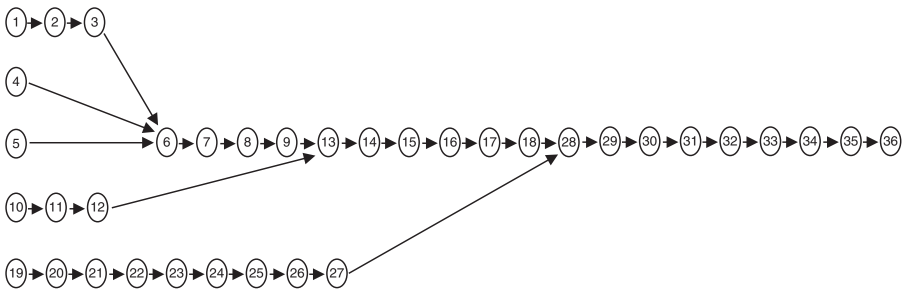

Figure 9. Priority diagram for knitted pants

The cycle time of the application was 0.65 minutes, considering the highest operation time.

The outcomes of the line balancing results for all the methods were the same for manual calculation and computer programming as expected, therefore they could be shared together. On the other hand, it must be noted that the time spent for manual calculations was dramatically higher than for computer calculation, and also several mistakes had to be corrected during the manual calculations process.

2.3.2.1 Hoffman method. Hoffman line balancing results are given in Table XII. With respect to Table XII, the assembly lines cycle time was defined as 0.65 minutes and a set-up with a total of 22 workstations was suggested. The calculations for loss of balance and line efficiency can be found below:

$$LB = \left[\left(nC - \sum C_o\right)/nC\right]100 = 23.663$$

$$LE = (1 - LB)100 = 76.337$$

The program output screen and results table taken from the program are given below, and these were in correlation with the manually calculated results (Figure 10).

Table XI.
Operation times, used machine types and previous operations for knitted pants

Operation number	Operations	Machine type	Operation times (min)	Previous operations
1	Pressing interlining to welt (pocket)	Press	0.20	–
2	Welt flatlock stitching	Flatlock machine	0.12	1
3	Assembling welt to pocket bag	Lock-stitch sewing machine	0.32	2
4	Pocket location mark	Handmade	0.26	–
5	Lower pocket bag pinking	4-thread overlock	0.10	–
6	Assembling pocket to front pant	Lock-stitch sewing machine	0.53	3-4-5
7	Pocket mouth cutting	Handmade	0.41	6
8	Pocket edge stitch	Lock-stitch sewing machine	0.34	7
9	Pocket around edge stitch	Lock-stitch sewing machine	0.65	8
10	Back pocket pinking	4-thread overlock	0.31	–
11	Back pocket stitching	Lock-stitch sewing machine	0.14	10
12	Pocket edges bartack stitch	Bartack machine	0.25	11
13	Side stitching	4-thread overlock	0.56	9-12
14	Side pant stitching	Waistband machine	0.40	13
15	Side pant control – Regulated	Handmade	0.12	14
16	Front – Back center stitching	4-thread overlock	0.27	15
17	Inside leg center stitch	4-thread overlock	0.44	16
18	Inside leg zigzag stitch	Lock-stitch sewing machine	0.06	17
19	Waistband buttonhole	Buttonhole machine	0.17	–
20	Assembling interlining to waistband part	Lock-stitch sewing machine	0.42	19
21	Assembling elastic band to waistband part	Lock-stitch sewing machine	0.36	20
22	Waistband edge stitch	Lock-stitch sewing machine	0.28	21
23	Assembling drawcord road to waistband	Flatlock machine	0.33	22
24	Drawcord road sewing up	Flatlock machine	0.31	23
25	Waistband pickstitching	Lock-stitch sewing machine	0.07	24
26	Waistband tapping	3-thread overlock	0.22	25
27	Passing drawcord to waistband	Handmade	0.30	26
28	Assembling waistband to pants	4-thread overlock	0.46	18-27
29	Preparing care label and stitching	Lock-stitch sewing machine	0.08	28
30	Preparing waistband label and stitching	Lock-stitch sewing machine	0.11	29
31	Pants hem buttonhole	Buttonhole machine	0.35	30
32	Pants hem cover stitch	Cover stitch machine	0.59	31
33	Pants hem zigzag stitch on end of cover stitch	Lock-stitch sewing machine	0.14	32
34	Preparing binding	Piping machine	0.10	33
35	Cutting binding	Handmade	0.10	34
36	Assembling binding to pants hem	Lock-stitch sewing machine	0.55	35
Total time			10.42	

2.3.2.2 Ranked positional weight method. Ranked positional weight line balancing results are given in Table XIII. With respect to Table XIII, the assembly lines cycle time was defined as 0.65 minutes and a set-up with a total of 27 workstations was suggested. The calculations for loss of balance and line efficiency can be found below:

$$LB = \left[\left(nC - \sum C_o\right)/nC\right)\right] 100 = 40.626$$

$$LE = (1 - LB)100 = 59.374$$

Workstation number	Operation number	Machine type	Time (min)	Total time for workstation (x)	Remaining time (C–x)
1	1	Press	0.20	0.58	0.07
	4	Handmade	0.26		
	15	Handmade	0.12		
2	2	Flatlock machine	0.12	0.63	0.02
	7	Handmade	0.41		
	35	Handmade	0.10		
3	3	Lock-stitch sewing machine	0.32	0.59	0.06
	11		0.14		
	18		0.06		
	25		0.07		
4	5	4-thread overlock	0.10	0.41	0.24
	10		0.31		
5	6	Lock-stitch sewing machine	0.53	0.61	0.04
	29		0.08		
6	12	Bartack machine	0.25	0.55	0.10
	27	Handmade	0.30		
7	8	Lock-stitch sewing machine	0.34	0.62	0.03
	22		0.28		
8	9	Lock-stitch sewing machine	0.65	0.65	0
9	13	4-thread overlock	0.56	0.56	0.09
10	14	Waistband machine	0.40	0.40	0.25
11	19	Buttonhole machine	0.17	0.52	0.13
	31		0.35		
12	16	4-thread overlock	0.27	0.27	0.38
13	17	4-thread overlock	0.44	0.44	0.21
14	20	Lock-stitch sewing machine	0.42	0.53	0.12
	30		0.11		
15	21	Lock-stitch sewing machine	0.36	0.50	0.15
	33		0.14		
16	23	Flatlock machine	0.33	0.64	0.01
	24		0.31		
17	26	3-thread overlock	0.22	0.22	0.43
18	28	4-thread overlock	0.46	0.46	0.19
19	32	Cover stitch machine	0.59	0.59	0.06
20	34	Piping machine	0.10	0.10	0.55
21	36	Lock-stitch sewing machine	0.55	0.55	0.10

Table XII. Hoffman line balancing results for knitted pants

2.3.2.3 COMSOAL method. COMSOAL line balancing results are given in Table XIV. With respect to Table XIV, the assembly lines' cycle time was defined as 0.65 minutes and a set-up with a total of 21 workstations was suggested. The calculations for loss of balance and line efficiency can be found below:

$$LB = \left[\left(nC - \sum C_o\right)/nC\right]100 = 23.663$$

$$LE = (1 - LB)100 = 76.337$$

2.3.2.4 Kilbridge and Wester method. Kilbridge and Wester line balancing results are given in Table XV. With respect to Table XV, the assembly lines' cycle time was

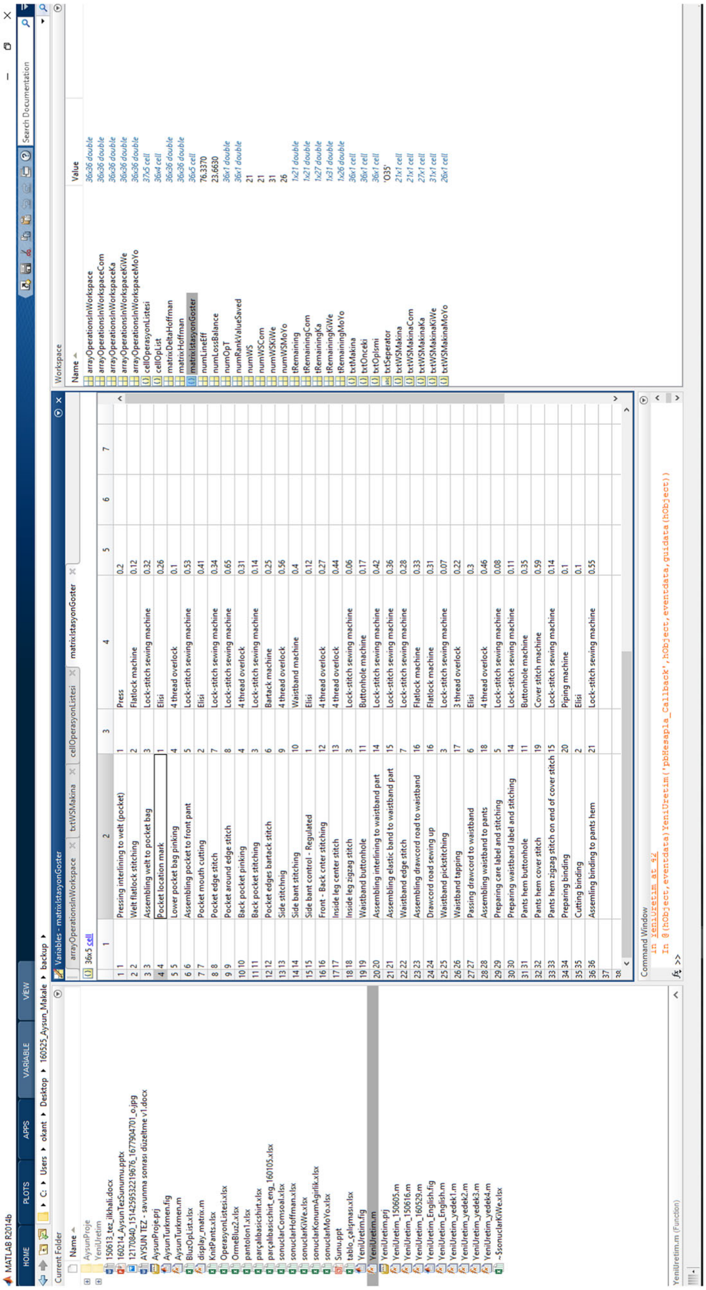

Figure 10.
Line balancing results for knitted pants in MATLAB

Workstation number	Operation number	Machine type	Time (min)	Total time for workstation (x)	Remaining time (C–x)
1	1	Press	0.20	0.46	0.19
	4	Handmade	0.26		
2	2	Flatlock machine	0.12	0.12	0.53
3	3	Lock-stitch sewing machine	0.32	0.32	0.33
4	5	4-thread overlock	0.10	0.41	0.24
	10		0.31		
5	6	Lock-stitch sewing machine	0.53	0.53	0.12
6	7	Handmade	0.41	0.55	0.10
	11	Lock-stitch sewing machine	0.14		
7	8	Lock-stitch sewing machine	0.34	0.34	0.31
8	9	Lock-stitch sewing machine	0.65	0.65	0
9	19	Buttonhole machine	0.17	0.17	0.48
10	20	Lock-stitch sewing machine	0.42	0.42	0.23
11	12	Bartack machine	0.25	0.25	0.40
12	21	Lock-stitch sewing machine	0.36	0.64	0.01
	22		0.28		
13	13	4-thread overlock	0.56	0.56	0.09
14	14	Waistband machine	0.40	0.52	0.13
	15	Handmade	0.12		
15	23	Flatlock machine	0.33	0.64	0.01
	24		0.31		
16	16	4-thread overlock	0.27	0.27	0.38
17	25	Lock-stitch sewing machine	0.07	0.07	0.58
18	26	3-thread overlock	0.22	0.52	0.13
	27	Handmade	0.30		
19	17	4-thread overlock	0.44	0.44	0.21
20	18	Lock-stitch sewing machine	0.06	0.06	0.59
21	28	4-thread overlock	0.46	0.46	0.19
22	29	Lock-stitch sewing machine	0.08	0.19	0.46
	30		0.11		
23	31	Buttonhole machine	0.35	0.35	0.30
24	32	Cover stitch machine	0.59	0.59	0.06
25	33	Lock-stitch sewing machine	0.14	0.14	0.51
26	34	Piping machine	0.10	0.20	0.45
26	35	Handmade	0.10		
27	36	Lock-stitch sewing machine	0.55	0.55	0.10

Table XIII. Ranked positional weight line balancing results for knitted pants

defined as 0.65 minutes and a set-up with a total of 31 workstations was suggested. The calculations for loss of balance and line efficiency can be found below:

$$LB = \left[\left(nC - \sum C_o\right)/nC\right)\right]100 = 48.288$$

$$LE = (1 - LB)100 = 51.712$$

3. Results

The approaches applied for this paper were the Hoffman, ranked position weight, COMSOAL and Kilbridge and Wester methods. For each of these methods, workstations were prepared, operations were assigned to these workstations as per the methods' instructions, and line balancing results were inspected.

Table XIV.
COMSOAL line balancing results for knitted pants

Workstation number	Operation number	Machine type	Time (min)	Total time for workstation (x)	Remaining time (C–x)
1	1	Press	0.20	0.58	0.07
	4	Handmade	0.26		
	15	Handmade	0.12		
2	2	Flatlock machine	0.12	0.63	0.02
	7	Handmade	0.41		
	35	Handmade	0.10		
3	3	Lock-stitch sewing machine	0.32	0.59	0.06
	11		0.14		
	18		0.06		
	25		0.07		
4	5	4-thread overlock	0.10	0.41	0.24
	10		0.31		
5	6	Lock-stitch sewing machine	0.53	0.61	0.04
	29		0.08		
6	8	Lock-stitch sewing machine	0.34	0.62	0.03
	22		0.28		
7	9	Lock-stitch sewing machine	0.65	0.65	0
8	12	Bartack machine	0.25	0.55	0.10
	27	Handmade	0.30		
9	13	4-thread overlock	0.56	0.56	0.09
10	14	Waistband machine	0.40	0.40	0.25
11	16	4-thread overlock	0.27	0.27	0.38
12	17	4-thread overlock	0.44	0.44	0.21
13	19	Buttonhole machine	0.17	0.52	0.13
	31		0.35		
14	20	Lock-stitch sewing machine	0.42	0.53	0.12
	30		0.11		
15	21	Lock-stitch sewing machine	0.36	0.50	0.15
	33		0.14		
16	23	Flatlock machine	0.33	0.64	0.01
	24		0.31		
17	26	3-thread overlock	0.22	0.22	0.43
18	28	4-thread overlock	0.46	0.46	0.19
19	32	Cover stitch machine	0.59	0.59	0.06
20	34	Piping machine	0.10	0.10	0.55
21	36	Lock-stitch sewing machine	0.55	0.55	0.10

The loss of balance and the line efficiency values stand as one of the most important conclusions of the study. The method results on t-shirt application calculated efficiency values can be listed as Hoffman (64.56 percent), positional weight (60.76 percent), COMSOAL (64.56 percent) and Kilbridge and Wester (60.76 percent). For the knitted pants application, the calculated efficiency values can be listed as Hoffman (72.87 percent), positional weight (42.75 percent), COMSOAL (72.87 percent) and Kilbridge and Wester (51.71 percent).

For these case studies, the Hoffman and COMSOAL methods were found to be the most efficient ones. On the other hand it is a known fact that other methods should also be applied in case other line studies can obtain better results. The advantages of the computer programming become important during this phase, as it allows all the methods to be calculated and compared in a fast and reliable way.

Workstation number	Operation number	Machine type	Time (min)	Total time for workstation (*x*)	Remaining time (*C*–*x*)
1	1	Press	0.20	0.46	0.19
	4	Handmade	0.26		
2	2	Flatlock machine	0.12	0.12	0.53
3	3	Lock-stitch sewing machine	0.32	0.32	0.33
4	5	4-thread overlock	0.10	0.41	0.24
	10		0.31		
5	6	Lock-stitch sewing machine	0.53	0.53	0.12
6	7	Handmade	0.41	0.55	0.10
	11	Lock-stitch sewing machine	0.14		
7	8	Lock-stitch sewing machine	0.34	0.34	0.31
8	9	Lock-stitch sewing machine	0.65	0.65	0
9	12	Bartack machine	0.25	0.25	0.40
10	13	4-thread overlock	0.56	0.56	0.09
11	14	Waistband machine	0.40	0.40	0.25
12	15	Handmade	0.12	0.29	0.36
	19	Buttonhole machine	0.17		
13	16	4-thread overlock	0.27	0.27	0.38
14	17	4-thread overlock	0.44	0.44	0.21
15	18	Lock-stitch sewing machine	0.06	0.48	0.17
	20		0.42		
16	21	Lock-stitch sewing machine	0.36	0.36	0.29
17	22	Lock-stitch sewing machine	0.28	0.28	0.37
18	23	Flatlock machine	0.33	0.33	0.32
19	24	Flatlock machine	0.31	0.31	0.34
20	25	Lock-stitch sewing machine	0.07	0.07	0.58
21	26	3-thread overlock	0.22	0.22	0.43
22	27	Handmade	0.30	0.30	0.35
23	28	4-thread overlock	0.46	0.46	0.19
24	29	Lock-stitch sewing machine	0.08	0.08	0.57
25	30	Lock-stitch sewing machine	0.11	0.11	0.54
26	31	Buttonhole machine	0.35	0.35	0.30
27	32	Cover stitch machine	0.59	0.59	0.06
28	33	Lock-stitch sewing machine	0.14	0.14	0.51
29	34	Piping machine	0.10	0.10	0.55
30	35	Handmade	0.10	0.10	0.55
31	36	Lock-stitch sewing machine	0.55	0.55	0.10

Table XV. Kilbridge and Wester line balancing results for knitted pants

Ultimately, the Hoffman method was found to be the most convenient method for computer programming as it can be calculated using matrix operations.

4. Conclusion

The aim of this study was to design assembly lines which had the highest performance and to compare the classical method, which is used widely by ready-to-wear companies with other methods which are used for assembly line balancing and then examine their applicability.

In this context the methods, which have been named, were applied and the results were analyzed.

When the results of the study were examined, it could be stated that almost all of the heuristic assembly line balancing methods could be applied on ready-to-wear assembly lines, though some methods were of lower efficiency because of basic obstacles to their applicability. When the classical method used in ready-to-wear companies these days is considered, the Hoffman and COMSOAL methods gave nearly the same results.

One of the most important pieces of intelligence from the application of line balancing methods is that, as the number of operations and workstations gets higher, applying methods also gets harder, more complex and becomes more open to suffer manual calculation faults. In order to overcome these difficulties, the use of computer-aided design has been found to be a requirement and other studies have also focused on this.

Pre-studies which were conducted in order to decide which programming language would be the best choice for the line balancing methods' application suggested that MATLAB, rather than C, C++, C# or Java, would be the best software choice. The main reason for this choice is that MATLAB is a powerful matrix operation software with a powerful user interface designing tool, which has the wherewithal to be used universally on every computer.

With the know-how gathered during the studies, the program was developed as a generic tool, which can be used on all operating systems and it does not require a MATLAB licence or any specific installation. It is very user-friendly with a helpful GUI, and even people with limited computer experience can use it. Therefore, it is ready to be used by industries to solve assembly line problems.

References

Acar, N. and Estas, S. (1991), "Kesikli Seri Üretim Sistemlerinde Planlama ve Kontrol Çalışmaları", Milli Prodüktivite Yayınları: 309, Baskı, MPM Endüstri Şubesi, Ankara, 3.

Agrawal, P.K. (1985), "The related activity concept in assembly line balancing", *International Journal of Production Research*, Vol. 23 No. 2, pp. 403-421.

Arcus, A.L. (1966), "COMSOAL: a computer method of sequencing for assembly lines", *International Journal of Production Research*, Vol. 4 No. 4, pp. 259-277.

Asar, K. and Andrew, J.D. (2001), "A knowledge based design methodology for manufacturing assembly lines", *Journal of Computers & Industrial Engineering*, Vol. 41 No. 4, pp. 441-467.

Baskak, M. (1998), "Çok Modelli/Ürünlü Montaj Hatlarının Dengelenmesi İçin Yeni Bir Model ve Çözüm Yöntemi", PhD thesis, Istanbul Technical University, İstanbul.

Baybars, I. (1986), "A survey of exact algorithms for the simple assembly line balancing problem", *Management Science*, Vol. 32 No. 8, pp. 900-932.

Bowman, E.H. (1960), "Assembly line balancing by linear programming", *Operations Research*, Vol. 8 No. 3, pp. 385-389.

Bryton, B. (1954), "Balancing of a continuous production line", MS thesis, Northwestern University, Evanson, IL.

Dundar, P., Guner, M. and Colakoglu, O. (2012), "An approach to the modular line balancing problem for an apparel product manufacturing with graph theory", *Tekstil ve Konfeksiyon*, Vol. 22 No. 4, pp. 369-374.

El-Sayed, E.A. and Boucher, T.O. (1985), *Analysis and Control of Production Systems*, Prentice Hall, Englewood Cliffs, NJ.

Eryuruk, S.H. (2012), "Clothing assembly line design using simulation and heuristic line balancing techniques", *Tekstil ve Konfeksiyon*, Vol. 22 No. 4, pp. 360-368.

Eryuruk, S.H., Kalaoglu, F. and Baskak, M. (2008), "Assembly line balancing in a clothing company", *Fibres & Textiles in Eastern Europe*, Vol. 16 No. 1, 66, pp. 93-98.

Eryuruk, S.H., Kalaoglu, F. and Baskak, M. (2011), "Assembly line balancing by using statistical method in clothing production", *Tekstil ve Konfeksiyon*, Vol. 22 No. 1, pp. 65-71.

Gehrlein, W.V. and Patterson, J.H. (1975), "Sequencing for assembly lines with integer task times", *Management Science*, Vol. 21 No. 9, pp. 1064-1070.

Guner, M., Yucel, O. and Unal, C. (2013), "Applicability of different line balancing methods in the production of apparel", *Tekstil ve Konfeksiyon*, Vol. 23 No. 1, pp. 77-84.

Helgeson, W.P. and Birnie, D.P. (1961), "Assembly line balancing using the ranked positional weight technique", *Journal of Industrial Engineering*, Vol. 12 No. 6, pp. 384-398.

Hoffman, T.R. (1963), "Assembly line balancing with a precedence matrix", *Management Science*, Vol. 9 No. 4, pp. 551-562.

Hoffman, T.R. (1990), "Assembly line balancing: a set of challenging problems", *International Journal of Production Research*, Vol. 28 No. 10, pp. 1807-1815.

Kayar, M. (2003), "Kot Pantolon Dikiminde Farklı Üretim Teknolojileri Verimliliğinin Simülasyon Yöntemiyle Karşılaştırılması", MSc thesis, University of Marmara, İstanbul.

Kayar, M. (2008), "Hazır Giyim İşletmelerinde Verimsizliği Ortaya Çıkaran Nedenlerin Araştırılması Ve Bunların Çözümüne Yönelik Alan Çalışması", PhD thesis, University of Marmara, İstanbul.

Kayar, M. and Akyalçın, C. (2014), "Applying different heuristic assembly line balancing methods in the apparel industry and their comparison", *Fibres & Textiles in Eastern Europe*, Vol. 22 No. 6, pp. 8-19.

Kilbridge, M.D. and Wester, L. (1961), "A heuristic method for assembly line balancing", *Journal of Industrial Engineering*, Vol. 12 No. 4, pp. 292-298.

MATLAB Documentation (2016), *Getting Started-Language Fundamentals – Mathematics – Graphics – Data Import and Analysis – Programming Scripts and Functions – App Building*, available at: www.mathworks.com/help/matlab/ (accessed April 14, 2016).

Moodie, C.L. and Young, H.H. (1965), "A heuristic method of assembly line balancing for assumptions of constant or variable work element times", *Journal of Industrial Engineering*, Vol. 16 No. 1, pp. 23-29.

Salveson, M.E. (1955), "The assembly line balancing problem", *Journal of Industrial Engineering*, Vol. 6 No. 3, pp. 18-25.

Talbot, F.B. and Patterson, J.H. (1984), "An integer programming with network cuts for solving the assembly line balancing problem", *Management Science*, Vol. 30 No. 4, pp. 85-99.

Talbot, F.B., Patterson, J.H. and Gehrlein, W.V. (1986), "A comparative evaluation of heuristic line balancing techniques", *Management Science*, Vol. 32 No. 4, pp. 430-454.

Tonge, F.M. (1961), *A Heuristic Program of Assembly Line Balancing*, Prentice-Hall, Englewood Cliffs, NJ.

Corresponding author
Yalcin Yesil can be contacted at: yalcinyesil@gmail.com