

January 01, 2004

Metaheuristic technique for the disassembly line balancing problem

Surendra M. Gupta
Northeastern University

Seamus M. McGovern
Northeastern University

Recommended Citation

Gupta, Surendra M. and McGovern, Seamus M., "Metaheuristic technique for the disassembly line balancing problem" (2004). . Paper 68. <http://hdl.handle.net/2047/d1001388x>

This work is available open access, hosted by Northeastern University.



Laboratory for Responsible Manufacturing

Bibliographic Information

McGovern, S. M. and Gupta, S. M., "Metaheuristic Technique for the Disassembly Line Balancing Problem", ***Proceedings of the 2004 Northeast Decision Sciences Institute Conference***, Atlantic City, New Jersey, pp. 223-225, March 24-26, 2004.

Copyright Information

Copyright 2004, Surendra M. Gupta.

Contact Information

Dr. Surendra M. Gupta, P.E.
Professor of Mechanical and Industrial Engineering and
Director of Laboratory for Responsible Manufacturing
334 SN, Department of MIE
Northeastern University
360 Huntington Avenue
Boston, MA 02115, U.S.A.

(617)-373-4846 **Phone**
(617)-373-2921 **Fax**
gupta@neu.edu **e-mail address**

<http://www.coe.neu.edu/~smgupta/> **Home Page**

METAHEURISTIC TECHNIQUE FOR THE DISASSEMBLY LINE BALANCING PROBLEM

Seamus M. McGovern, Northeastern University, Boston, MA 02115, (617)-494-2054, mcgovern.s@neu.edu

Surendra M. Gupta*, Northeastern University, Boston, MA 02115, (617)-373-4846, gupta@neu.edu

(*Corresponding author)

ABSTRACT

The disassembly line is the best choice for automated disassembly of returned products. However, finding the optimal balance for a disassembly line is computationally intensive with exhaustive search quickly becoming prohibitively large. Metaheuristic techniques provide a general algorithmic framework that can be applied to this optimization problem. Although metaheuristics show promise in solving this complex problem, challenges exist in the variety of evaluation criteria available, a lack of disassembly-specific data sets for metaheuristic testing and a lack of performance analysis tools. In this paper, a balance performance measure is reviewed along with a size-independent a priori data set and graphical analysis tools.

INTRODUCTION

One of the first steps in product recovery is disassembly. After disassembly, reusable parts/subassemblies are cleaned, refurbished, tested and directed to the remanufacturing operations. The recyclable materials can be sold to raw-material suppliers, while the residuals are sent to landfills.

Many papers have discussed the different aspects of product recovery. Brennan et al. [2], and Gupta and Taleb [6] investigated the problems associated with disassembly planning and scheduling. Gungor and Gupta [4], [5] presented the first introduction to the disassembly line balancing problem (DLBP) by developing an algorithm for solving it with the goal of assigning tasks to workstations to probabilistically minimize the cycle time. For a review of environmentally conscious manufacturing and product recovery, see Gungor and Gupta [3].

While a disassembly line is the best choice for automated disassembly of end-of-life products, finding its optimal balance is computationally intensive with exhaustive search quickly becoming prohibitively large. Recently developed metaheuristic techniques provide algorithmic frameworks that hold promise for application to the DLBP. Typical metaheuristic techniques include simulated annealing, tabu search, iterated local search, evolutionary algorithms, and ant colony optimization. Because they are often near-optimal techniques, before applying metaheuristics to the DLBP, several items need to be addressed including the determination of a balance measure, generation of a specific data set for metaheuristic testing and finally, definition of performance analysis tools. In this paper, a balance performance measure is presented – which lends itself to

metaheuristic applications due to its non-linear nature – along with an a priori data set and graphical analysis tools.

DISASSEMBLY LINE MEASURE OF BALANCE

Following notation are used in the remainder of the paper:

CT	cycle time; time available at each workstation
F	M ^c Govern-Gupta balance measure for a solution
I_{max}	maximum possible total idle time
I_{min}	minimum possible total idle time
j	workstation count (1, ..., NWS)
k	part identification (1, ..., n)
n	number of parts for removal
NWS	number of workstations required for a solution
NWS_{max}	maximum possible number of workstations
NWS_{min}	minimum possible number of workstations
PRT_k	part removal time required for k th part
WS_j	elapsed time in workstation j

Line balancing seeks to achieve Perfect Balance (all idle times equal to zero). When this is not achievable, either Line Efficiency (IE) or the Smoothness Index (SI) is commonly used as a performance evaluation tool [1]. We use a measure of balance that combines the two, is easier to calculate, and lends itself to ready evaluation by metaheuristics. SI rewards similar idle times at each workstation, but at the expense of allowing for a large (sub-optimal) number of workstations. This is because SI compares workstation elapsed times to the largest WS_j instead of the CT . IE rewards the minimum number of workstations, but allows unlimited variance in idle times between workstations because no comparison is made between WS_j s. This paper makes use of the balancing method developed by M^cGovern and Gupta [7]. The M^cGovern-Gupta measure of balance simultaneously minimizes the number of workstations while ensuring that idle times at each workstation are similar, though at the expense of the generation of a non-linear objective function. The method is computed based on the minimum number of workstations required as well as the sum of the square of the idle times for all the workstations. This penalizes solutions where, even though the number of workstations may be minimized, one or more have an exorbitant amount of idle time when compared to the other workstations. It provides for leveling the workload between different workstations on the disassembly line. Therefore, a resulting minimum performance value is the more desirable solution, indicating both a minimum number of workstations and similar idle times across all workstations. The M^cGovern-Gupta balancing measure is represented as:

$$F = \sum_{j=1}^{NWS} (CT - WS_j)^2 \quad (1)$$

A PRIORI KNOWN OPTIMAL BENCHMARK SET

Any developed DLBP metaheuristics will first be used on a collection of test cases to demonstrate their performance as well as their limitations. Benchmark data sets are common for many NP-complete problems, such as Oliver30 and RY48P for application to the Traveling Salesman Problem (TSP) and Nugent15/20/30, Elshafei19 and Krarup30 for the Quadratic Assignment Problem (QAP). Unfortunately, because of their size and their design, most of these existing data sets have no known optimal answer and new solutions are not compared to the optimal solution, but rather the best-known solution to date. In addition, since DLBP is a recently defined problem, no benchmark data sets exist. It was therefore necessary to develop a set of instances for the DLBP in order to evaluate DLBP metaheuristics. We propose an a priori known optimal solution benchmark line balance data set to determine its efficacy.

This size-independent a priori benchmark data set was generated based on the following. Since, in general, solutions to larger and larger instances cannot be verified as optimal (due to the time complexity of exhaustive search), it is proposed that instances be generated in such a way to always provide a known solution. This was done by using part times consisting exclusively of prime numbers. They were further selected to ensure that no combinations of these part removal times allowed for any equal summations in order to reduce the number of possible optimal solutions. For example, part removal times 1, 3, 5 and 7 and $CT = 16$ would have minimum idle time solutions of not only one 1, one 3, one 5 and one 7 at each workstation, but various additional combinations of these as well since $1 + 7 = 3 + 5 = \frac{1}{2} CT$. Subsequently, the chosen instances are made up of parts with removal times of 3, 5, 7 and 11 and $CT = 26$. As a result, the optimal balance for all subsequent instances will consist of a perfect balance of combinations of 3, 5, 7 and 11 at each workstation with idle times of 0. To further complicate the data (i.e., provide a large, feasible search space), only one part is listed as hazardous and this would be one of the parts with the largest part removal time. This is done so that only the hazardous sequencing will be demonstrated, while providing no solution sequence advantage to any greedy-based methods since it does not allude to the fact that each of the three smaller part removal time tasks needs to be placed in the first workstation for Perfect Balance. In addition, one part (a middle listed, second smallest part removal time component) is listed as being demanded. The demand is also made equal for all parts except for one in order to demonstrate demand sequencing,

again while providing no advantage for any greedy-based methods. Furthermore, no precedence constraints are placed on the sequence, a deletion that further challenges a metaheuristics methods' ability to attain an optimal solution (precedence constraints can significantly shrink the search space).

For this data set, it was desired not only that the optimal solution be known but also that by varying size, known solution instances be generated. In this developed data set, the smallest practical problem instance is $n = 8$ (the first data set size that is larger than the trivial $n = 4$ and $NWS = 1$ example). There is no design limit to the largest instance to be generated. This benchmark set provides a variety of data not easily solved by general-purpose metaheuristics. The developed benchmark has instance size evenly distributed in steps of $\|PRT_U\|$ (i.e., 4). This provides numerous instances of predetermined, calculable solutions of various sizes. Small n decreases the NWS and tends to exaggerate less than optimal performance while large n demonstrates time complexity growth and efficacy changes. To summarize, the a priori known optimal solution benchmark line balance data set consists of n parts with 4 unique part removal times of 3, 5, 7 and 11. The disassembly line is operated at a speed that allows 26 seconds ($CT = 26$) for each workstation. In general, for any n parts consisting of this type of data:

$$NWS_{\max} = n \quad (2)$$

$$NWS_{\min} = \frac{n}{\|PRT_U\|} \quad (3)$$

$$I_{\max} = \frac{nCT(\|PRT_U\| - 1)}{\|PRT_U\|} \quad (4)$$

$$I_{\min} = 0 \quad (5)$$

Since $\|PRT_U\| = 4$, each part removal time is generated by:

$$PRT_k = \begin{cases} 3, 0 < k \leq \frac{n}{4} \\ 5, \frac{n}{4} < k \leq \frac{n}{2} \\ 7, \frac{n}{2} < k \leq \frac{3n}{4} \\ 11, \frac{3n}{4} < k \leq n \end{cases} \quad (6)$$

DLBP DATA ANALYSIS

The proposed data analysis of the results consists of a numerical or graphical comparison of known best and worst case results with the metaheuristic-generated results. These may include: number of workstations, total idle time, balance measure, and time complexity. Although the a priori benchmark has an intuitive, perfect balance solution, experience has shown that it is very unusual for a metaheuristic technique to find the optimal solution, even for

sets as small as $n = 12$. These near-optimal metaheuristic-generated solutions coupled with the known optimal and worst case solutions for all problem sizes under study, provides a method for not only comparing the metaheuristic to the best and worst cases, but to other metaheuristics as well.

An example using DLBP-specific greedy, 2-Opt and hill climbing (AEHC) techniques was selected to demonstrate the solution challenges with the a priori benchmark data and use of the data analysis tools to measure efficacy. Various performance measures are demonstrated with the analysis of the a priori benchmark data sets of $8 \leq n \leq 80$. The benchmark data set regularly resulted in the DLBP greedy algorithm obtaining the near-optimal solution of $NWS = NWS_{min} + 1$ workstations (figures 1 and 2).

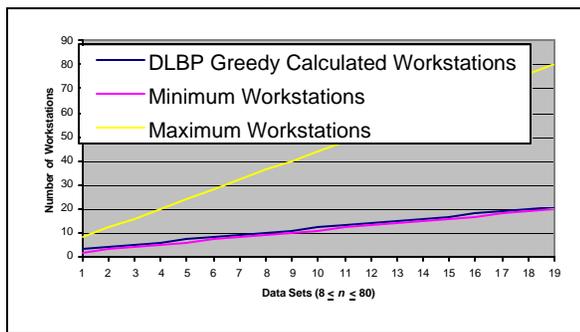


Fig. 1: Data analysis of workstation calculation.

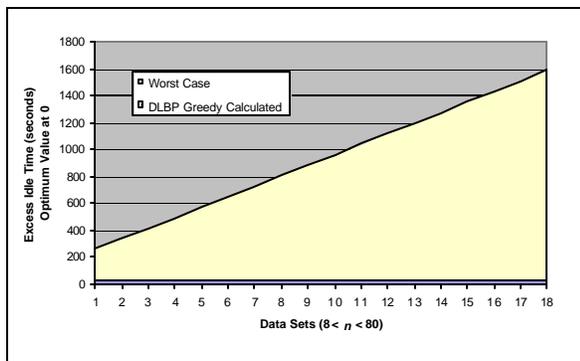


Fig. 2: Data analysis of excess idle time at all workstations.

Other graphical data analysis techniques show how the DLBP Greedy/2-Opt consistently obtained the best-balanced solutions (figure 3) while being only slightly better than those obtained by DLBP Greedy/AEHC and DLBP Greedy alone and at a significant time complexity cost (figure 4). This analysis also shows how, in general, the efficacy of these techniques actually improved with problem size (figure 3, this can be attributed to DLBP Greedy's ability to consistently generate solution sequences with no worse than $NWS_{min} + 1$ workstations). This data analysis also shows that the solutions found, while not optimal, were

consistently within a few percent of optimal solution regardless of n .

The number of workstations required by DLBP Greedy went from 17.4 percent of optimal (minimum) balance (at $n = 8$) to 0.4 percent (at $n = 80$). In terms of balance, DLBP Greedy/2-Opt went from generating the optimal balance initially (i.e., balance value of 0) to being within 0.1 percent of optimal at $n = 80$, while DLBP Greedy/AEHC improved from 9.2 to 0.2 percent of the optimal balance. Finally, a comparison of calculated NWS with NWS_{max} and NWS_{min} shows that these DLBP methods improved from 11.1 percent of optimal NWS at $n = 12$, to 1.7 percent at $n = 80$.

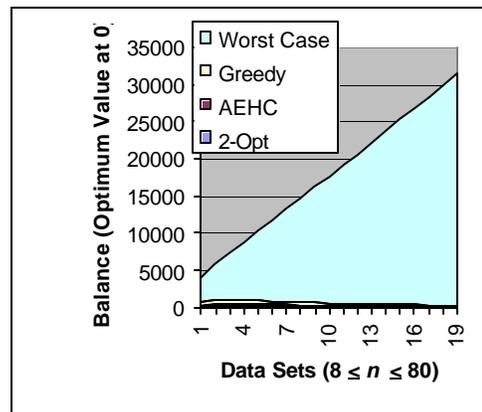


Fig. 3: Data analysis of balance performance.

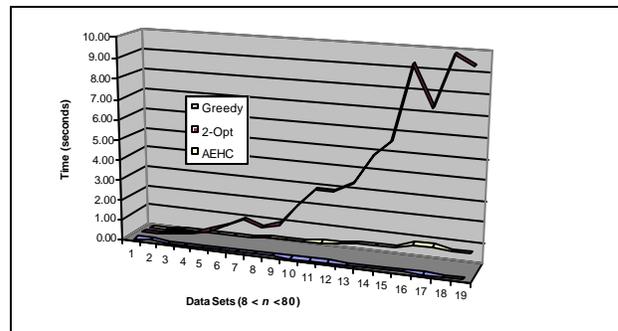


Fig. 4: Data analysis of time complexity.

REFERENCES

- [1] Elsayed, E.A. and Boucher, T.O., *Analysis and Control of Production Systems*, Prentice Hall, NJ, 1994..
- [2] Brennan, L., Gupta, S.M. and Taleb, K.N., 1994, Operations planning issues in an assembly/disassembly environment. *International Journal of Operations and Production Planning*, 14(9), 57-67.
- [3] Gungor, A. and Gupta, S.M., 1999, Issues in environmentally conscious manufacturing and product recovery: a survey. *Computers and Industrial Engineering*, 36(4), 811-853.
- [4] Gungor, A. and Gupta, S.M., 2001, A solution approach to the disassembly line problem in the presence of task failures, *International Journal of Production Research*, 39(7), 1427-1467.

- [5] Gungor, A. and Gupta S.M., 2002, Disassembly line in product recovery, *International Journal of Production Research*, 40(11), 2569-2589.
- [6] Gupta, S.M. and Taleb, K.N., 1994, Scheduling disassembly. *International Journal of Production Research*, 32, 1857-1866.
- [7] M^cGovern, S.M. and Gupta, S.M., "Greedy Algorithm for Disassembly Line Scheduling", *Proceedings of the 2003 IEEE International Conference on Systems, Man, and Cybernetics*, Washington, DC, pp. 1737-1744, October 2003.