A Fast Estimator of Performance with respect to the Design Parameters of Self Re-entrant Flowshops

Umar Waqas*, Marc Geilen*, Sander Stuijk*, Joost van Pinxten*,
Twan Basten*‡,Lou Somers†§, Henk Corporaal*
* Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands.
† Research and Development, Océ Technologies, Venlo, The Netherlands.
‡ Embedded Systems Innovation, TNO, Eindhoven, The Netherlands.
§ Department of Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands.
Corresponding author email: u.waqas@tue.nl

***Abstract*—Self re-entrant flowshops consist of machines which process jobs several times. They are found in applications like TFT-LCD assembly, LED manufacturing and industrial printing. The structure of a self re-entrant flowshop influences its performance. To get better performance while reducing costs a fast performance estimation method can be used to explore the trade-offs between the structure and the performance during the design process. We present a novel performance estimator that uses the information in the jobs being processed to analyse the trade-offs. We study the impact of the design parameters of an industrial printer using the performance estimator with an average estimation time of 1.1 milliseconds per job and with an average accuracy of not less than 96%.**

I. Introduction

Many Cyber Physical Systems (CPSs) in manufacturing systems [11], [12] and assembly lines [13] consist of self re-entrant flowshops. Performance estimation of a self re-entrant flowshop is crucial during its design. Figure 1 shows a typical arrangement of a self re-entrant flowshop consisting of re-entrant machines (shown as boxes) that process jobs. The arrows indicate the flow of jobs through the machines. In a self re-entrant flowshop jobs are only allowed to re-enter the same machine or move to the next machine. The number on top of the re-entrant loop indicates the number of times a job returns back to the same machine to get reprocessed. The re-entrant loop abstracts from components such as conveyor belts, robotic arms or human operators. Similarly, a machine is an abstraction of a set of embedded systems that collaboratively schedule and control physical and mechanical processes. Decisions about the structure of a flowshop made during the design of a re-entrant loop influence the performance of the flowshop. For example, as later shown by the case study, changing the length of the re-entrant loop of a Large Scale Printer (LSP) impacts its performance. In context of flowshops a sheet is a job and a print request with many sheets is a set of jobs.

A self re-entrant machine consists of several components. The operation of such a machine is shown in

This work was supported in part by the Dutch Technology Foundation STW through project NEST 10346 and through the Robust CPS program, project 12693.

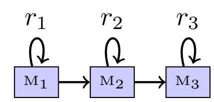

Figure 1: Machines in a self re-entrant flowshop.

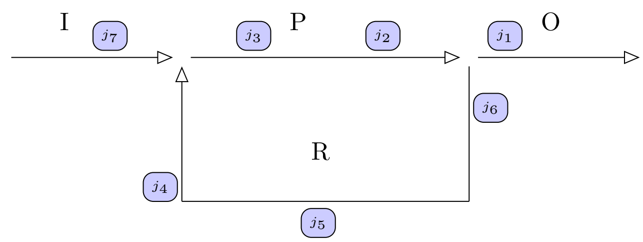

Figure 2: Jobs flowing in a self re-entrant machine.

Figure 2. The jobs (shown as rounded rectangles) enter the machine from the Input component (I) , get processed by the Processor component (P) and return back on the re-entrant loop (R) to get processed again. A job leaves the machine at the Output component (O). The P and the R component form the re-entrant loop of the machine. We assume that there are possibly many jobs in the machine and all of them re-enter the machine at least once (i.e. every job is at least processed twice). For many industrial applications (e.g. wafer sorting [9], LED manufacturing [11] and TFT-LCD assembly [12]) the processor component requires adjustment before processing a job leading to *sequence dependent setup times*. We assume that the setup times are significant and they arise due to the differences between jobs. In a wafer sorter [9], handling memory ICs and logic ICs are examples of different jobs that require setups, i.e., changes in the settings of the machine due to the nature of the process before the processor component, P, switches from one type of job to the other type. The structure of a flowshop (physical part) has an influence over the freedom a scheduler has (cyber part). Thus there is an interplay between the cyber and physical aspects of self re-entrant flowshops.

On the re-entrant loop, we assume that the velocity of a job has a lower and an upper bound that result from mechanisms used to transport the job, such as, motors with programmable (but bounded) rotational velocity. The bounds in conjunction with the length of the loop give

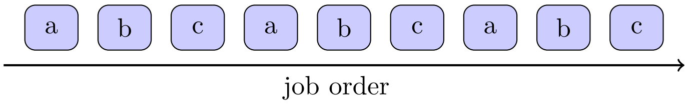

Figure 3: A regular job set with a pattern of jobs.

a deadline, referred to as a *relative due date*, between the time when a job is processed for the first time and the time when the job is processed for the second time leading to a minimum and a maximum loop time. The difference between the minimum and the maximum loop time acts as a virtual buffer that can be used to reduce the number of setups to increase performance. We assume that the cost of a flowshop is directly proportional to the minimum loop time and the buffer time. Reducing the minimum loop time and the buffer time reduces the cost of the flowshop. Hence there exist trade-offs between the cost and the performance of a re-entrant flowshop with setup times and relative due dates.

Many job sets in an industrial operation of a re-entrant flowshop either contain repeating patterns of jobs or only jobs of the same kind. Flowshops that process regular job patterns are for example a wafer sorter or a printer that repeatedly prints the same book. Figure 3 is an example of a job set containing nine jobs with jobs a, b, c as a pattern repeated three times. Note that the pattern repeats in a *tandem* fashion i.e. a pattern immediately follows after the previous pattern ends and the patterns are never partial. The influence of the timing constraints in a job set on the performance of a flowshop is specific to the job set. Given the job sets a challenge is to determine the minimum loop time and the buffer time for maximal performance.

Re-entrant flowshops are found in many industrial applications such as photo-lithography, printed circuit board manufacturing, assembly of circuits ([4], [5] provide detailed lists). In these applications, the jobs re-enter a machine due to the nature of the process performed by the machines; for example, in photo-lithography several layers of silicon are etched on to a wafer one by one. Examples of re-entrant flowshops with sequence dependent setup times are looms in textile industry [1] and wafer sorters [9]. Applications that are self re-entrant with sequence dependent setup times and due dates can be found in TFT-LCD and LED manufacturing [11], [12] and the LSP studied in this work. For these applications the relevant scheduling methods , e.g. [3], [7], [14], [15], are prohibitively slow to be used during the early design phases to explore the design space and therefore there is a need for a fast method to estimate performance. Performance estimation for self re-entrant flowshops with sequence dependent setup times and relative due dates was not studied in literature before.

We present a novel performance estimation method for self re-entrant flowshops with sequence dependent setup times and relative due dates. Section II compares the estimator with approaches from literature. The estimation problem is formally defined in Section III. Section IV describes the estimator. Section V shows how the estimator can be used to perform Design Space Exploration (DSE). A case study performed on a LSP is described in Section VI. We conclude this paper in Section VII.

II. Related work

Since re-entrant flowshop scheduling problems have been shown to be NP-Complete [6], existing exact scheduling algorithms are prohibitively slow be used for fast performance estimation of re-entrant flowshops. Even the heuristic techniques [3], [7], [14], [15] are too slow to explore the design space of a self re-entrant flowshop and thus do not satisfy the need of fast performance estimators. The work of [13] finds an optimal re-entrant loop size for a LSP without sequence dependent setup times. Extending the work of [13] for sequence dependent setup times requires solving an LP program several times which results in a mixed integer program that is compute intensive to solve. Traditionally, performance estimation of re-entrant flowshops has been performed using simulation [2], mean value analysis [8] and probabilistic methods [10]. Simulation based approaches are time consuming [2] which prohibits their usage as fast performance estimators. Mean value based estimators (e.g. [8]) assume that the processing times are known for jobs under consideration. However, in a flowshop with sequence dependent setup times the processing times depend on the state in which a machine is left by the previous job(s) and thus the processing times are not known to allow mean value analysis.

The mean value based approach [8] and the probabilistic method [10] assume scheduling policies which either require priority scheduling for buffers or assume fixed scheduling policies. The assumption of a fixed scheduling policy and buffers with priority allow faster estimation but it does not optimize to reduce sequence dependent setups and thus will be less accurate. In this work, we avoid many setups by incorporating the information from the pattern in a regular job set. The computational complexity of the estimator is $O(n)$ which is similar to the mean value estimates in [8] as they are estimates that use the information about n jobs in a job set.

III. Problem definition

A loop in a self re-entrant flowshop impacts its performance if the length of the loop is altered. Let a re-entrant flowshop with sequence dependent setup times and relative due dates be a tuple $f(M, \phi, \Lambda)$ having a set M of unary machines, a re-entrance vector $\phi =< \gamma_1, \ldots, \gamma_r >$ of machine indices, a vector $\Lambda =< (l_1, b_1), \ldots, (l_{|M|}, b_{|M|}) >$ containing $|M|$ pairs of minimum loop time and buffer time (respectively) for the machines in the flowshop.

Let $J = \{j_1, \ldots, j_n\}$ be a job set processed by the flowshop f. Each job $j_i \in J$ has a set of r operations $O_i = \{o_{i,1}, \ldots, o_{i,r}\}$. For each operation the re-entrance vector describes the machine on which it executes and only allows self re-entrance i.e. $\phi(\gamma_{i+1}) \geq \phi(\gamma_i)$. The set $O = O_1 \cup \dots \cup O_n$ consists of all operations in the job set. Let $p_{i,j}$ be the processing time of an operation $o_{i,j} \in O$. Similarly, let $s_{i,j,x,y}$ be the setup time required between the end of the operation $o_{i,j}$ and the start of the operation $o_{x,y}$. The relative due date between two operations $o_{i,j}$ and $o_{x,y}$ is $d_{i,j,x,y}$ and defines the maximum permitted time between the start time of the operation $o_{i,j}$ and the operation $o_{x,y}$.

A valid schedule for the job set J on a re-entrant flowshop f has the start times ($start(o_{i,j})$) for all operations $o_{i,j} \in O$ such that the following three constraints hold. (1) A machine processes at most one operation at a time as a machine is a unary resource. (2) Between any two operations $o_{i,j}$ and $o_{x,y}$ the processing and setup time constraint holds i.e. $start(o_{x,y}) \geq start(o_{i,j}) + p_{i,j} + s_{i,j,x,y}$. (3) The due date constraint between two operations $o_{i,j}$ and $o_{x,y}$ holds i.e. the $start(o_{x,y}) \leq start(o_{i,j}) + d_{i,j,x,y}$.

Let the throughput of a machine m in a flowshop f for a job set J be denoted by $n_{m,f,J}$. Then, the throughput of the re-entrant flowshop f for the job set J is $n_{f,J} = \min_{m \in M} n_{m,f,J}$ assuming that the machine with least throughput is the bottleneck. Let n_J^* denote the maximum of throughput estimates for the job set J over flowshops considered in the DSE.

The performance estimation method defined in this work assumes a steady state behaviour i.e. the job sets are large and the pattern in a job set tandemly repeats. Formally, we define the performance of a job set J on a flowshop f by $\xi_{f,J} = \frac{n_{f,J}}{n_J^*}$.

Definition 1. *A job set J is regular if it consists of tandem repeats of the form w^k where $k > 1$ and $w \in \Sigma^*$ is a word over the alphabet set Σ.*

The example job set in Figure 3 has pattern $w = abc$ with $k = 3$ and alphabet set $\Sigma = \{a, b, c\}$. Note that there are infinitely many words in the language Σ^*. The performance estimation method assumes that a job set consists of one and only one pattern $w \in \Sigma^*$ repeated a finite number of times in a job set.

Given the performance of a job set, a challenge is to find the performance of the flowshop on the job sets that are grouped in different categories. Let c be a category where $c = \{J_1, \dots, J_u\}$ of u job sets. Then the performance of a re-entrant flowshop is defined as follows.

Definition 2. *The performance of a re-entrant flowshop f on the job sets in a category c is the weighted sum $\xi_{c,f} = \sum_{J \in c} w_J \times \xi_{f,J}$ where w_J is the weight of the job set.*

The weight w_J denotes the relative importance of the job set J in the category c modelled by a statistical distribution as shown in the LSP case study in Section VI. We assume that the weights for the job sets in a category sum to one i.e. $\sum_{J \in c} w_J = 1$. The higher the importance of a job set the more it contributes to the overall performance of the flowshop for the category. Let C be the set of all

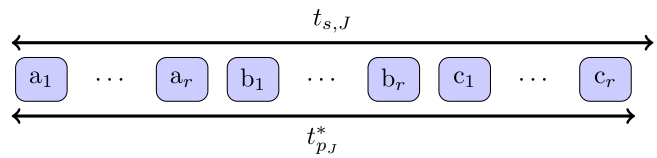

Figure 4: Place allocation for the regular job set $\{a, b, c\}$.

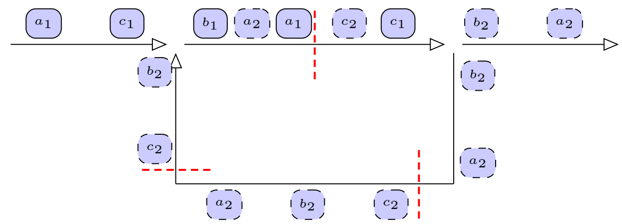

Figure 5: A loop with 3 slots with single re-entrance.

categories of job sets of interest. Then the performance of a re-entrant flowshop (called the *total performance*) over a set of categories is defined as follows.

Definition 3. *(Problem definition) The total performance of a re-entrant flowshop f over set of categories of job sets C is defined as $\xi_{T,f} = \sum_{c \in C} \xi_{c,f} \times w_c$ where w_c is the weight for category c.*

The *total performance* indicates how productive a flowshop is over all job sets. The higher the total performance, the more productive the flowshop is.

Definition 4. *A flowshop f_o has maximum total performance over the set of flowshops F under consideration if the total performance $\xi_{T,f_o} = \max_{f \in F} \xi_{T,f}$.*

The following section describes a performance estimation method which is used to find f_o in Section V.

IV. Performance estimation

The method in this section estimates the performance by dividing the loop in a self re-entrant machine into segments called *slots* that process repetitions of a pattern. Within a slot, for each job an allocation where it will be processed is determined based on the sequence of the jobs in the job set and the timing constraints. Figure 4 shows the allocation of places for jobs with pattern a, b, c with every job re-entering r times. The first place in the slot is allocated for a_1 (the subscript denotes that the allocation is fixed for the first occurrence of a in the slot), followed by the re-entering occurrences of job a. Similarly, for job b the allocations for the first and the re-entrant occurrences are determined. The performance estimation is performed by arranging multiple slots in a re-entrant loop. Figure 5 shows the slots in the re-entrant loop of a flowshop for the pattern shown in Figure 3 that has jobs a, b and c re-entering the loop once. The scheduling behaviour is modelled by assigning the positions for the jobs which are processed for the first time (shown as solid outlines) and for the second time (shown as dashed outline).

A slot s produces a number of jobs equal to the length $|p_J|$ of the pattern p_J in the job set J. For example in Figure 5 from every slot three jobs leave the flowshop assuming the flowshop is full of jobs (steady state). Then the performance of a re-entrant flowshop is the ratio between the work processed in a slot and the time span $t_{s,J}$ of the slot. The *time span* of slot is the time that a slot occupies from the total loop time (minimum loop time + buffer time).

Definition 5. *The estimated performance of a job set J on a re-entrant machine m in a flowshop f is $\eta_{m,f,J} = \frac{|p_J|}{t_{s,J}}$ where $|p_J|$ is the number of jobs in pattern p_J and $t_{s,J}$ is the time span of the slot s.*

Given a job set for a machine in a flowshop the performance is estimated using Definition 5. The estimate is the ratio of the amount of work done in a slot and the time duration of the slot. The time span of a slot depends on the timing constraints between jobs in a pattern and thus in many cases will result in slot times which do not completely fit in the loop. For such cases the slot time is adjusted such that the loop time becomes an integer multiple of the time span of the slot as the estimation method assumes that the complete re-entrant loop in a machine m can be sub-divided into slots of identical time spans such that the loop time is an integer multiple of the time span of a slot. The assumption ensures that slots do not interfere with each other.

Let the pattern time t_{p_J} be the sum of the timing constraints between the jobs in the pattern p_J. Let the re-entrant pattern time $t^*_{p_J}$ be the sum of the timing constraints between the jobs in a pattern p_J considering the preallocated locations for re-entrant jobs as well. Then, for a given minimum loop time $t_l > 0$ and buffer time $t_b > 0$ there are 3 possible cases (as shown in Equation 1).

The first case is when the re-entrant pattern time is such that it fits an integer number of times in the loop. In this case no adjustment is required. This case happens when the minimum number of slots $n_{min} = \lfloor \frac{t_l}{t^*_{p_J}} \rfloor$ that fit in the loop is not equal to the maximum number of slots $n_{max} = \lfloor \frac{t_l + t_b}{t^*_{p_J}} \rfloor$ that fit in the loop indicating that the range of the loop back times allow the re-entrant loop to be split into integer number of slots.

$$t_{s,J} = \begin{cases} t^*_{p_J} & \text{if } n_{min} \neq n_{max} \\ \frac{t_l}{n_{min}} & \text{if } n_{min} \neq 0 \text{ and } n_{min} = n_{max} \\ t_{p_J} + t_l \times |p_J| & \text{otherwise} \end{cases} \tag{1}$$

The second case is when at least one pattern fits in the re-entrant loop but the loop time is not an integer multiple of the re-entrant pattern time. Thus the time span of a slot, after adjustment is $t_{s,J} = \frac{t_l}{n_{min}}$. An intuitive argument, for the second case, is that the loop time will always be an integer multiple of the slot time as follows. Replacing n_{min} makes $t_{s,J} = \frac{t_l}{\lfloor \frac{t_l}{t^*_{p_J}} \rfloor}$ and shows that the ratio of the loop time and the slot time results in an integer number i.e. $\frac{t_l}{t_{s,J}} = \lfloor \frac{t_l}{t^*_{p_J}} \rfloor$.

The third case is when $n_{min} = 0$ i.e. the re-entrant pattern time is larger than the given minimum loop time thus dividing the re-entrant loop into slots is not possible. In such a case, the re-entrant machine will process each job separately i.e. without interleaving the slots. The time $t_{p_J} + t_l \times |p_J|$ denotes the total time to process the job set (where only one job is in the machine at a time).

The computational complexity of the estimator for n jobs in a job set is $O(n)$ which is explained with an intuitive argument. Equation 1 has time complexity $O(n)$ which is due to the computation of t_{p_J} and $t^*_{p_J}$. The remaining variables in the equation are either derived from others in constant time or are parameters coming from the description of the considered job. Once the equation is computed, the performance is estimated through Definition 5 which has constant time complexity. Thus the total complexity of the estimator is $O(n)$. As the estimator is fast, it is suitable for DSE. The usage of the estimator is described in the following section.

V. DSE method

The method described in the previous section estimates the performance of a flowshop for a given job set. In this section we demonstrate how the estimator can be used to perform DSE to find a flowshop which performs the best on all job sets considering the importance of the job sets. The relative importance of a job set is a weight that is a variable from a probability distribution χ. The variable is drawn based on a specific property of a given job-set. For example, in many application areas (such as the LSP case study) large job sets are rare and thus the number of jobs in a job set could be used as a property to find the relative importance of the job sets. Moreover, application specific distributions can be used where the relative importance is a more complicated function of a given job set.

The estimator is used by the DSE method as described in Algorithm 1. The DSE method explores the design space of a flowshop given four sets: a set of minimum loop times, a set of buffer times, a set of machines and a set of categories of job sets. The output of the algorithm are estimates for the performance of job sets over the provided loop and buffer times on all machines. Then the estimated performance of maximally productive flowshop f_o can be computed using the job performance estimates, the distribution χ and Definitions 2 and 3.

VI. Case study

This section describes a case study to find the trade-offs between the performance of a LSP, the minimum loop time and the buffer time of the re-entrant loop in the printer. A LSP is a system printing sheets of paper in a re-entrant loop where every sheet is processed twice, i.e. to print the first side and the second side of a sheet. A job set consists of many jobs where each job is a sheet. The performance

Algorithm 1: The design space exploration method

Input: Four sets: minimum loop times L, buffer times B, machines M and categories C
Output: Estimates of performance $\eta_{m,f,J}$
for $m \in M$, $l \in L$, $b \in B$ **do**
  Let f be the flowshop with machine m having minimum loop time l and buffer time b.
  for $c \in C$ **do**
    for $J \in c$ **do**
      Compute $\eta_{m,f,J}$ by Definition 5.
      Record $\eta_{m,f,J}$.
    end
  end
end
return recorded values of $\eta_{m,f,J}$.

of the LSP is the length of sheets (in millimetres) printed per second by the printer (to normalize for different paper sizes). The length of the loop is measured in the number of seconds spent by a sheet travelling from the merge point (the merge point is where the input component and the return loop meet) and to come back to the merge point. Similarly, the buffer time is the amount of seconds spent by a sheet in the buffer region of the re-entrant loop.

The case study is performed over 3 categories of job sets with: *homogeneous* patterns, *booklet* patterns and *unstructured* randomly generated patterns. These categories represent the jobs observed in an industrial operation of the printer. The details of the test set are described in Section VI-A. Section VI-B describes the experimental setup. The accuracy of the estimation is described in Section VI-C followed by Section VI-D that describes the results of the case study.

A. Test set

The test set to explore the design space of a LSP consists of job sets which represent print requests of industrial importance. The job sets are categorized based on the pattern in each category. The *homogeneous* job sets are with all jobs having the same sheet characteristics. Then the pattern in a homogeneous job set consists of a single sheet only. The homogeneous category is of most importance as the majority of industrial job sets are homogeneous in nature. Examples of homogeneous job sets are found in printing of forms and informational pamphlets. The second category is of the *booklet* type where there is a cover sheet followed by several body sheets. Examples of booklets are books and brochures with many sheets. The cover sheet in a booklet is thicker than a body sheet requiring a sequence dependent setup between printing a cover and a body sheet and vice versa. The third category consists of job sets having *unstructured* randomly generated patterns repeating a number of times in the job set.

Table I lists, for each category, the parameters of the test set. The number of jobs in a job set is the first parameter which indicates what is the minimum and maximum number of jobs in a job set (sheets in a print request). The second parameter indicates how many job sets are created with a sheet count randomly chosen between the minimum and maximum number of sheets. The jobs in a job set are sheets of specific length and thickness. The pattern length indicates the range of the number of jobs in a pattern. The total job sets indicates the total number of job sets generated by the combination of different parameters for each category. In total there are 2100 job sets in the test set. The design space of the LSP consists of the minimum loop time between $1-18$ seconds and $1-5$ seconds for the buffer time (in steps of 1 second) resulting in 199500 test cases to be explored. Minimum loop time of 10 seconds and buffer time of 2 seconds are considered as default values in the experiments if not mentioned explicitly. The ranges of the minimum loop time, the buffer time and the default values are of interest to the designers of the LSP and are preferred for commercial purposes.

Parameter	**homogeneous**	**booklet**	**unstructured**
Jobs in a pattern/job set	between 15-115 sheets	between 3-12 sheets	between 3-12 sheets
No. of random job set size selection	70	5	5
Sheet length (mm)	{177.8, 210, 420, 431.8, 500}	c $\in$ {355.6, 420} and b $\in$ {177.8, 210}	{177.8, 210, 420,500}
Sheet thickness (mm)	{0.1, 0.2}	b $\in$ {0.1}, c $\in$ {0.2}	{0.1, 0.2}
Pattern length	-	between 2 -36 sheets	between 1-8 sheets
Total job sets	700	700	700

Table I: Parameters for the test set used in the case study.

B. Experimental setup and run-time

The DSE method and the performance estimation were implemented in Python. The experiments presented in the next section were performed on a Windows 7 Enterprise Edition running on an Intel i7 at 2.90 GHz. The estimator took on average 1.1 ms and maximum 32 ms per test case. The heuristic scheduler used in the LSP (proprietary scheduler used by the company which manufactures the LSP) requires on average 200 ms and minimum 150 ms per test case. Thus the estimator is relatively fast and suitable for DSE.

C. Accuracy of performance estimation

The accuracy of the performance estimation influences the outcomes of the DSE. Incorrect performance estimation might lead to a design which has worse performance

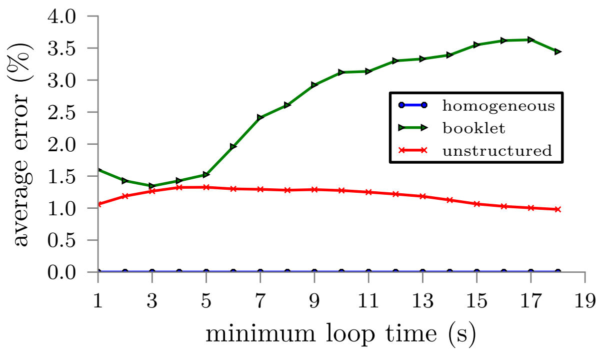

Figure 6: The absolute error between estimation and measured performance of the LSP.

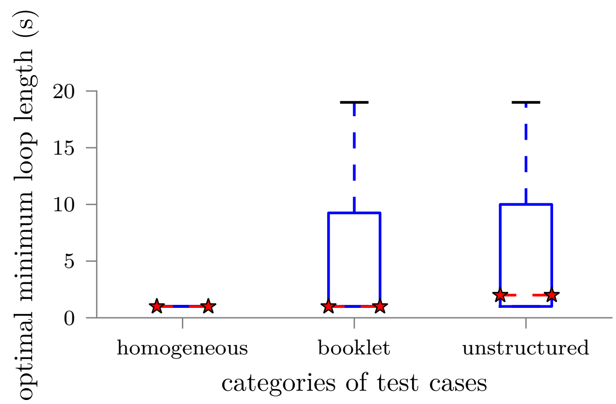

Figure 7: The variation in the optimal minimum loop time.

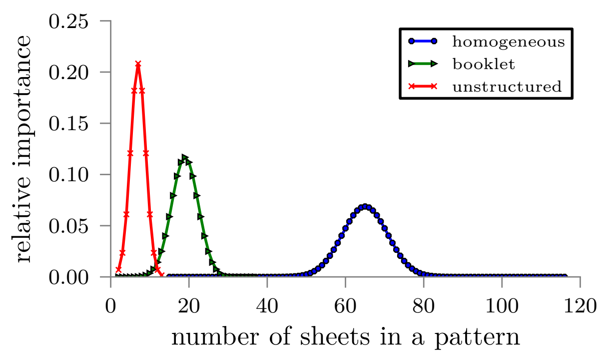

Figure 8: Distributions used to model the importance of job sets in different categories.

in reality. To assess the error, for each test case, the performance estimate is compared to the performance reported by the print scheduler (written as *real*) of the LSP. For all test cases, Figure 6 shows the percentage of average error illustrating how good the estimator performs on the entire test set. For instance, for a minimum loop time l and for a job set J the ratio of error is $e_{l,J} = \frac{real_{l,J} - estimate_{l,J}}{real_{l,J}}$. Then the percentage of average error for a category C and minimum loop time l is $error_{l,C} = \frac{\sum_{J \in C} e_{l,J}}{|C|}$. As the figure shows, the homogeneous category has the least error because of the predictability of the scheduling behaviour of the LSP for a homogeneous job set. The error increases with the increase in the dynamic structure in a job set. The error in the booklet category increases with the increase in the minimum loop time. The increase is because, for test cases with few jobs and a large loop it is less likely to reach a steady state assumed by the estimation method. However, the average accuracy of at least 96% is promising to evaluate early design decisions and hence the estimation method will sufficiently facilitate the trade-off analysis.

D. *Trade-off analysis*

The trade-offs between the length, the buffer time and the performance of a LSP arise because of the sequence dependent setup times, the relative due dates and the buffering behaviour of the re-entrant loop. Furthermore the trade-offs are specific to job sets. Figure 7 shows the variation in the optimal minimum loop time for all job sets as a box-plot. The height of a box shows the span of the variation for 50% of the cases closest to the median in a category. The span of a whisker (the lines going out of the boxes) indicates the variation for 25% of the cases. The two stars inside a box is the median for a given category and splits the variation into two halves with each half representing variation for 50% of the cases. For the homogeneous category, the optimal minimum loop time is the smallest possible minimum loop time because, due to the homogeneity of the jobs, the minimum loop time does not affect the performance and the DSE method chooses the shortest re-entrant loop as the optimal one. The optimal minimum loop time for the booklet and the unstructured category varies over almost the entire design space. Over all categories, the experiment shows that the optimal minimum loop time is job and category specific. Thus, fixing the minimum loop time to a certain value would make the LSP productive for some job sets but will lose performance for other job sets.

The relative importance of job sets in different categories can be modelled using distributions as shown in Figure 8. The relative importance is normally distributed and was derived from the profiles of the customers of the LSP. For a given number of sheets in a pattern (x-axis) its relative importance is given by the height of the normal curve which is used to compute the total performance.

Figure 9 shows the total performance for the categories of the job sets. For a given minimum loop time and for all job sets, the performance is the weighted sum of the relative importance of a job set and the ratio of performance with the maximum performance. The booklet category performs relatively poorly on small minimum loop times because a booklet does not fit and then the setups cannot be avoided thus losing performance. Furthermore, the weighted sum of the total performance over all categories (according to the importance of a category i.e $booklet = 0.7$, $unstructured = 0.1$, $homogeneous = 0.2$) indicates the total performance of the LSP. The printer has maximum performance at 13 seconds of minimum loop

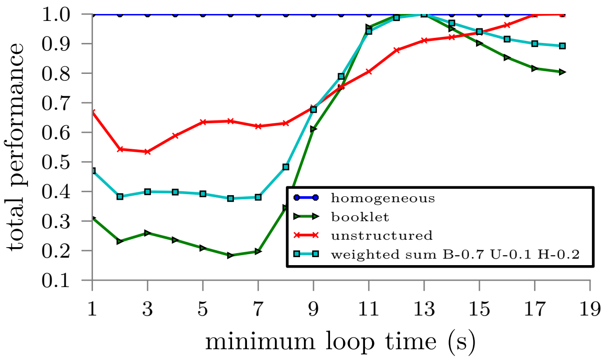

Figure 9: Total performance for different categories.

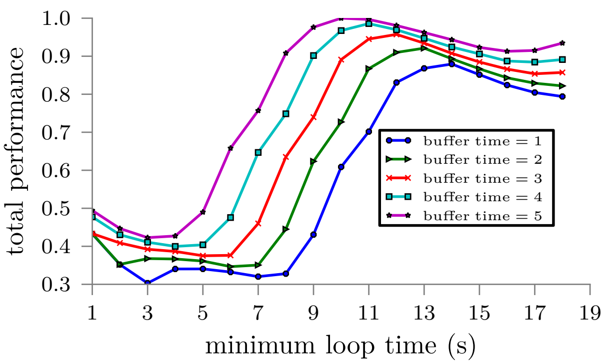

Figure 10: Total performance over the design space.

time and default buffer time.

Figure 10 shows the weighted sum total performance (normalized by maximum of estimated performances) for the printers with different buffer times. An increase in the buffer time increases the total performance because of the increase in the scheduling freedom to minimize the number of setups. However, as shown in the figure, at minimum loop time of 11 seconds the performance is almost the same for buffer time of 4 and 5 seconds. That means a cost reduction could be achieved by having a smaller buffer without significant loss of performance. Similarly, for buffer time of 1 second the minimum loop time of 13 seconds performs better than 15 seconds thus a shorter loop is beneficial. However, as shown in the figure, an increase of a single second is very beneficial between 7 to 11 seconds. The designer of the LSP can assess, using these trade-offs unveiled by the DSE, whether adding additional cost to the LSP brings performance gains.

VII. Conclusion

The design parameters of a self re-entrant flowshop influence its performance. Traditional approaches to estimate performance are either too slow or do not take sequence dependent setup times into account. The performance estimator described in this work, for common regular job sets, allows to explore the relation between the design parameters and performance without using compute intensive algorithms. As shown by the LSP case study, the maximally productive length of the re-entrant loop highly depends on the job sets under consideration. For the LSP the estimator has an average compute time of 1.1 milliseconds with an average accuracy of not less than 96%. The accuracy and fast computation of the estimator allow a designer to evaluate structural decisions during the design of self re-entrant flowshops with sequence dependent setup times and due dates.

The performance estimator assumes that a job set consists of repeating patterns of jobs. The assumption holds in many industrial applications but estimation of performance for job sets without a pattern is left as future work. Furthermore, the design parameters explored in this work were the loop time and buffer time because they significantly influence the performance. The study of which other design parameters influence the performance is another interesting topic for future work.

References

[1] A. Allahverdi, C. Ng, T. E. Cheng, and M. Y. Kovalyov. A survey of scheduling problems with setup times or costs. *European Journal of Operational Research*, 187(3):985–1032, 2008.

[2] M. Chincholkar and J. W. Herrmann. Estimating manufacturing cycle time and throughput in flow shops with process drift and inspection. *International Journal of Production Research*, 46(24):7057–7072, 2008.

[3] S.-W. Choi and Y.-D. Kim. Minimizing total tardiness on a two-machine re-entrant flowshop. *EJOR*, 199(2):375–384, 2009.

[4] L. Danping and C. K. Lee. A review of the research methodology for the re-entrant scheduling problem. *International Journal of Production Research*, 49(8):2221–2242, 2011.

[5] H. Emmons and G. Vairaktarakis. *Flow shop scheduling: theoretical results, algorithms, and applications*, volume 182. Springer, 2012.

[6] M. R. Garey, D. S. Johnson, and R. Sethi. The complexity of flowshop and jobshop scheduling. *Mathematics of operations research*, 1(2):117–129, 1976.

[7] M. Nawaz, E. E. Enscore Jr, and I. Ham. A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. *Omega*, 11(1):91–95, 1983.

[8] Y. Park, S. Kim, and C.-H. Jun. Performance analysis of re-entrant flow shop with single-job and batch machines using mean value analysis. *Production Planning & Control*, 11(6):537–546, 2000.

[9] W. Pearn, S. Chung, M. Yang, and K. Shiao. Solution strategies for multi-stage wafer probing scheduling problem with reentry. *Journal of the Operational Research Society*, 59(5):637–651, 2008.

[10] S. Pradhan, P. Damodaran, and K. Srihari. Predicting performance measures for markovian type of manufacturing systems with product failures. *European Journal of Operational Research*, 184(2):725–744, 2008.

[11] H. J. Shin. A dispatching algorithm considering process quality and due dates: an application for re-entrant production lines. *The International Journal of Advanced Manufacturing Technology*, 77(1-4):249–259, 2015.

[12] H. J. Shin and Y. H. Kang. A rework-based dispatching algorithm for module process in tft-lcd manufacture. *International Journal of Production Research*, 48(3):915–931, 2010.

[13] L. Swartjes, L. Etman, J. van de Mortel-Fronczak, L. Somers, and J. Rooda. *Model-based Constrained Optimization for Paper Path Layout and Timing Design of Printers (Master Thesis)*. Eindhoven University of Technology, 2012.

[14] E. Taillard. Some efficient heuristic methods for the flow shop sequencing problem. *EJOR*, 47(1):65–74, 1990.

[15] D.-L. Yang, W.-H. Kuo, and M.-S. Chern. Multi-family scheduling in a two-machine reentrant flow shop with setups. *EJOR*, 187(3):1160–1170, 2008.