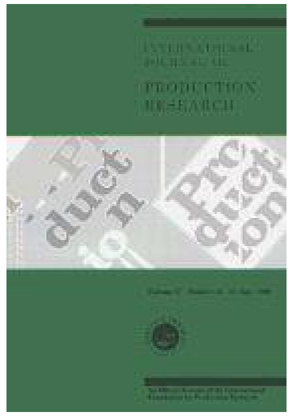
Cross-training workers in Dual Resource Constrained systems with heterogeneous processing times

Jos a C Bokhorst, Gerard J C Gaalman



Cross-training workers in Dual Resource Constrained systems with heterogeneous processing times

Journal:	*International Journal of Production Research*
Manuscript ID:	TPRS-2008-IJPR-0192.R1
Manuscript Type:	Original Manuscript
Date Submitted by the Author:	27-Jun-2008
Complete List of Authors:	Bokhorst, Jos A C; University of Groningen, Faculty of Economics and Business, Department of Operations Gaalman, Gerard J C; University of Groningen, Faculty of Economics and Business, Department of Operations
Keywords:	WORKER ASSIGNMENT, DISCRETE EVENT SIMULATION
Keywords (user):	Cross-training, Dual Resource Constrained systems

Cross-training workers in Dual Resource Constrained systems with heterogeneous processing times

J.A.C. Bokhorst*and G.J.C. Gaalman
Department of Operations,
Faculty of Economics and Business, University of Groningen
PO Box 800, 9700 AV Groningen, the Netherlands

* Corresponding author.
Tel: +31 50 3634693; Fax: +31 50 3632032
E-mail: j.a.c.bokhorst@rug.nl

ABSTRACT

In this paper, we explore the effect of cross-training workers in Dual Resource Constrained (DRC) systems with machines having different mean processing times. By means of queuing and simulation analysis, we show that the detrimental effects of pooling (cross-training) previously found in single resource constrained (SRC) heterogeneous systems are also apparent in DRC heterogeneous systems. Fully cross-training workers in DRC heterogeneous systems is only beneficial if the differences between mean processing times are not too large, otherwise cross-training should be pursued within homogeneous subgroups of machines. Due to the limited machine availability, DRC systems are unable to use some of the potential assignment flexibility from cross-trained workers (pooled queues) that can be used in SRC systems. However, it appears that this restriction in the DRC system may even improve the system mean flow (waiting) time performance compared to the SRC system for relatively large differences in processing time. Finally, in fully flexible multiple server queuing systems, restricting the assignment flexibility by applying a decentral when-rule (i.e. a commonly used labour assignment rule in practice and research) instead of a central when-rule also seems to improve the mean flow time performance under processing time differences.

Key words: Cross-training, Labour assignment, Dual Resource Constrained systems, pooling

****** [Revision June 2008] ******

Word count: 8836 words

Cross-training workers in Dual Resource Constrained systems with heterogeneous processing times

1. Introduction

In practice, most manufacturing systems are not only constrained by machine capacity, but also by labour capacity (see e.g. Wisner and Sifer, 1995). Practical systems can thus often be regarded as “Dual Resource Constrained” (hereafter, DRC) systems. Regularly, Western manufacturing companies have to compete with companies in low wage countries where the number of workers allocated to a process does not have a large impact on the total costs of a process. Competing with these companies involves pursuing the right strategy (e.g. seeking for innovation instead of mass producing standard products) while continuously striving to make better use of the labour force. A key success factor is to appropriately train and deploy the available workers. For instance, Hopp and Van Oyen (2004) show the possible mechanisms by which worker cross-training (and their allocation to tasks) can support strategic objectives such as lower costs, shorter lead times, better quality, and increased production flexibility. Also, Molleman and Van den Beukel (2007) show a positive relationship between worker flexibility in team-based work and its perceived contribution to efficiency and work quality and they show a weakly negative relationship between worker flexibility and its perceived contribution to innovation. Furthermore, the authors indicate that these relationships are moderated by contextual factors such as task autonomy, skill utilisation, and task monotony.

In today’s manufacturing environment, workers thus increasingly need to be flexible (cross-trained)—being able to operate several machines and take over machining tasks or help other workers with their tasks—while remaining efficient and motivated. A manufacturing system without cross-training (without overlapping skills) can be modelled as a collection of separate single-server queuing systems. Each worker (server) processes the jobs assigned to him/her on the machines that are used by this worker only. This can be regarded as a specialised system, in that each machine (and its jobs) can only be handled by one server (worker). By contrast, cross-training creates a flexible system through overlapping skills that enable jobs to be processed by alternative workers (i.e. servers), resulting in a one-queue multiple-server system. Cross-training workers in a DRC system thus resembles the pooling concept, where separate single-server queuing systems are combined into one multiple-server system. Note that we refer to the concept of pooling queues only, instead of to that of pooling servers as well (see Mandelbaum and Reiman 1998). However, as will be explained in detail in section 3, the concept of cross-training workers in a DRC system works out differently than the concept of pooling queues in a 'single resource constrained' (SRC) system. This is caused by machine restrictions (i.e. workers can only process a job if the right machine is available) and specific labour assignment rules in the DRC system.

Most pooling literature relates to SRC systems. For instance, pooling of queues for machines is investigated in systems without labour constraints, or pooling of queues for workers (i.e. cross-training workers to create overlapping skills) is researched in systems without machine constraints. Queuing theory shows that pooling generally greatly improves system performance (see e.g. Kleinrock 1976, Buzacott and Shanthikumar 1993). However, for SRC systems, pooling research has also shown that systems with non-identical servers and/or multiple job types—known as heterogeneous systems—may not benefit from pooling (see section 2.1 for a literature review). Nevertheless, due to the machine restrictions and specific labour assignment rules in DRC systems, the findings on pooling effects in SRC heterogeneous systems alone may not be adequate to support cross-training decisions in DRC heterogeneous systems. This urges the need to carefully study the effect of cross-training (pooling) in DRC heterogeneous environments.

This paper addresses the question whether the performance impact of cross-training workers in DRC systems is influenced by having non-identical machines and/or multiple job types with heterogeneous processing times. This is a highly relevant question, since DRC heterogeneous systems are commonly found in practice, while the effects of cross-training in heterogeneous environments are yet unknown. That is, to the best of our knowledge, the impact of cross-training has not been studied yet when dealing with DRC heterogeneous systems. It is not unusual that a manufacturing team performs more than one type of operation on different machines with unequal average processing times. We refer to one of many examples we have encountered in practice, representing a typical job-shop production unit (PU). The PU manufactures copper bars, where copper strips undergo punching, trimming, bending, drilling, milling and bench working operations. The routing variety of products is large. An analysis of about 3500 orders showed that the number of processing steps after punching lies between 1 and 5, with an average of 1.6. The average processing time per operation ranges from 28 minutes for bench working till 89 minutes for milling. Since workers remain at the machines during operation and are partially cross-trained (including several workers who are trained for both milling and bench working), they will face heterogeneous processing times. The increasing pressure that firms feel to create a flexible workforce and thus increase the level of cross-training makes this topic even more relevant. If cross-training is shown to have negative effects in DRC heterogeneous environments, the impact on cross-training decisions may be large.

The specific practical questions that arise for managers are: ‘Should cross-training be pursued in DRC systems with heterogeneous processing times caused by differences in job types and/or machines? Does it matter where flexibility is added in the manufacturing systems and what assignment rules are set? What is the impact of the size of the system and the extent of the processing time differences?' This paper addresses these questions using queuing theory and simulation.

The paper is structured as follows. First, section 2.1 provides a literature review on research that shows the advantages of specialisation over pooling. In addition, recent literature in the field of cross-training is reviewed in section 2.2. Section 3 contrasts the concept of cross-training in DRC systems with that of pooling in SRC systems. Section 4 formulates and motivates the DRC models used within this study. Section 5 uses queuing theory to explore the impact of heterogeneous processing time distribution characteristics. Section 6 presents a simulation study and section 7 discusses the results. Section 8 is a concluding section.

2. Literature review

The possible negative effects of pooling in heterogeneous environments have been shown in research using SRC systems, not in research using DRC systems. Section 2.1. therefore focuses on central studies on the effect of pooling in SRC heterogeneous environments and positions the current study. Section 2.2. first reviews recent cross-training studies in several SRC and DRC environments and then focuses on cross-training literature in DRC heterogeneous environments.

2.1. When specialisation is superior to pooling

Smith and Whitt (1981) already show that it may be disadvantageous to combine systems with different service-time distributions. Combining queues of customers (jobs) with different processing times will increase the variance of the processing times in the pooled system, which may be detrimental to the average waiting time. Rothkopf and Rech (1987) extend the arguments why combining queues may at times be counterproductive by also including customer behaviour arguments (e.g. jockeying, choosing the shortest queue, etc.). Jouini *et al.* (2008) show that specialisation can even be preferable over pooling in a homogeneous environment. They focus on the specialised team management benefits of agents serving different portfolios of homogeneous customers in call centers. The resulting increased service rate efficiency and/or decreased call-back proportion can outweigh the benefits of pooling all agents to serve all customers. Pinker and Shumsky (2000) show that flexible workers may not gain sufficient experience to provide high-quality service. However, they also find that specialised workers may face low utilisation (since more workers are needed to do the same amount of work as in the flexible system) which also degrades quality. In this paper, however, we will specifically focus on the mean waiting (flow) time effect of combining queues with different processing times without taking into account customer behaviour. Next, research with the same focus will be discussed.

The papers of Benjaafar (1995) and Benjaafar *et al.* (1995) both devote a section to the effect of pooling in a heterogeneous environment. More specifically, both papers consider systems with multiple job types requiring different mean processing times (a condition which they term processing variety). Based on an approximation for an M/G/m queuing system (see e.g. Buzacott and Shanthikumar, 1993) and on a series of simulation experiments, they conclude that pooling will result in a degradation of performance whenever the increase in the squared coefficient of variation in the processing time means of the job types offsets the gains resulting from pooling. They formulate a bound for which this may occur and conclude that a relatively high squared coefficient of variation is needed. Furthermore, they show that an increasing utilisation leads to an increased preference for specialised systems over pooled systems.

Buzacott (1996) compares a 2-machine specialised system and a 2-machine pooled system having to process job types A and B with different mean processing times and equal workloads. In the specialised system, each machine processes one job type. With the pooled system, arriving jobs of both types are allocated cyclically to the machines at their arrival. It is shown that the specialised system will be preferable if the difference in processing times is sufficiently large. Furthermore, it is shown that this effect is stronger under higher system utilisation and under more variability of the processing times of the job types.

Whitt (1999) states that in the case of different service-time distributions of servers, there is a trade-off between the economies of scale gained from a pooled system and the cost of having customers with shorter service times have their quality of service degraded by customers with longer service times. He demonstrates that aggregation (pooling) of classes having different service-time distributions is not always advantageous.

This subsection reviewed central studies showing that pooling in an SRC heterogeneous environment is not always beneficial. More recent related research on the effects of pooling queues in multiple-server systems with variability in processing times includes e.g. Lippolt *et al.* (2003), Van Dijk and Van der Sluis (2006), El-Taha and Maddah (2006). Our research differs from all studies discussed above in that it explores the issue of pooling (cross-training) in a DRC instead of an SRC heterogeneous environment.

2.2. *Worker cross-training*

Worker cross-training is a widely studied topic in various SRC and DRC manufacturing and service environments, including call centers, field service systems, serial production systems, manufacturing cells, job shops, etc (see Hopp and Van Oyen 2004, and Nembhard 2007 for a survey). For an overview of the call center literature including training issues, the reader is referred to Gans *et al.* (2003). More recently, Iravani *et al.* (2007) provided a deterministic solution approach to the complex stochastic problem of designing effective cross-training configurations in call centers based on small world network theory.

In the field service context, cross-training is employed to be able to quickly and successfully deal with equipment failures (for research on cross-training decisions in field service situations, see e.g. Agnihothri and Mishra 2004, Chakravarthy and Agnihothri 2005).

Cross-training in serial production systems is extensively researched including but not restricted to topics such as work sharing (e.g. Zavadlav *et al.* 1996, McClain *et al.* 2000), and bucket brigades (e.g. Bartholdi and Eisenstein 1996, Armbruster and Gel, 2006, Armbruster *et al.* 2007).

DRC studies are found in parallel, serial and job-shop manufacturing environments and an important body of research in this field deals with the issues of cross-training and labour assignment. The extent and division of cross-training impacts the performance of DRC systems, as well as do the assignment rules chosen to assign skilled workers to machines or tasks. Reviews of DRC research can be found in Treleven (1989), Gargeya and Deane (1996), and Hottenstein and Bowman (1998).

Independent of the specific environment in which cross-training is studied, specialisation (no flexibility) and full flexibility (pooling, total flexibility, full cross-training) can be regarded as two extreme configurations with many alternatives in between. It is well known that full flexibility is often not necessary since about the same performance can be obtained with less flexibility (e.g. Malhotra *et al*. 1993, Fry *et al*. 1995, Campbell 1999). Jordan and Graves (1995) studied the effect of process flexibility, which they define as the ability of plants to produce different types of products. They stressed the importance of chaining, which can be regarded as a specific structure to pool a limited number of queues in a system with multiple queues and multiple servers. The concept of chaining has thereafter received much attention, but predominantly in SRC environments (e.g. Sheikhzadeh *et al*. 1998; Gurumurthi and Benjaafar 2004) and more specifically applied to cross-training in SRC environments (e.g. Daniels *et al*. 2004, Hopp *et al*. 2004, Inman *et al*. 2004, Jordan *et al*. 2004, Iravani *et al*. 2005, Iravani *et al*. 2007). The chains in this literature link products to plants (process flexibility), products to machines (routeing flexibility) or workers to tasks (cross-training).

Within DRC environments, Bokhorst *et al*. (2004), Slomp *et al*. (2005) and Yue *et al*. (2008) incorporate the concept of chaining in cross-training configurations linking workers to machines. The skills in a chained cross-training configuration are arranged in such a way that all workers and machines are either directly or indirectly connected. This results in the ability to shift work from a worker with a heavy workload to a worker with a lighter workload, leading—directly or indirectly—to a more balanced workload. Bokhorst *et al*. (2006) compare the effects of inter-cell routing flexibility with those of inter-cell cross-training in a DRC cellular manufacturing system. Future research may incorporate the concept of chaining for linking products to cells as well as for linking workers to (machines within) cells to study its (combined) effects on performance. Even though the performance of chaining configurations is found to come close to that of complete pooling configurations in homogeneous environments, it may be worthwhile to find optimal configurations with limited pooling in heterogeneous environments (i.e. with asymmetric characteristics). In this paper, however, we restrict ourselves to comparing specialised with completely pooled systems and see the study of chaining configurations in heterogeneous environments as an extension to be dealt with in further research.

Most cross-training research in DRC heterogeneous environments with non-identical servers deals with human learning and forgetting behaviors, where human performance varies over time (see e.g. Malhotra *et al.* 1993, McCreery and Krajewski 1999, Yue *et al.* 2008). One of the findings is that increased worker flexibility may not always be beneficial for system-wide performance. Even though workers face heterogeneous processing times in these systems, the relation between the extent of processing time differences and the impact of cross-training remains unclear. This because the level of cross-training by itself impacts the extent of learning and forgetting in these systems. In this paper, we therefore assume workers to be equally proficient (homogeneous) in the machines they are trained for and we do not include learning and forgetting effects. In the current paper, differences in job types and/or machine characteristics will lead to workers facing heterogeneous processing times. The extent of these differences may therefore also be larger than the extent of human performance differences for a single job type or machine. Modelling heterogeneous task proficiencies will be a topic of future research.

Kher and Fry (2001) studied the impact of labour flexibility, labour assignment rules and order dispatching rules on delivery performance of DRC systems with orders from vital customers and normal priority customers. Their results indicated that rules which bias priorities toward vital customers do so at the expense of non-vital customers, while increasing labour flexibility seems to have a beneficial impact on manufacturing lead times for all job types. Even though Kher and Fry (2001) distinguished multiple job types in a DRC system, the job types were homogeneous in their processing time characteristics.

In sum, even though cross-training has been studied in DRC heterogeneous environments, the influence of non-identical machines and/or multiple job types with heterogeneous processing times is yet unknown.

3. Cross-training workers in DRC systems versus pooling servers in SRC systems

This section aims to clarify the differences between the concept of cross-training workers in a DRC system and the concept of pooling queues in an SRC system. For this, we will use the examples in figure 1, representing schematic SRC and DRC queuing systems with multiple job types. The SRC systems (figures 1a and 1b) consist of two machines and two job types. The DRC systems (figures 1c and 1d) consist of three machines, three job types and two workers. Since there are several ways to model a DRC queuing system, we chose a representation that effectively visualises the concept of cross-trained workers. In our DRC queuing systems, a job first enters a queue in order to be coupled to an available machine and then the job and machine enter a queue in order to be processed by an available and skilled worker. The worker can thus be regarded as the server in the queuing system, serving jobs by means of a suitable machine. Note that the coupling of a specific job to a machine may be altered till the moment a worker actually requests another job (and machine). When the worker finishes a job, the finished job leaves the system and the machine and worker become available again.

In both the SRC and DRC systems, jobs arrive according to a Poisson process and are given priority based on the First In System First Served (FISFS) rule, which uses 'time of entry' information that jobs receive upon arrival. In the DRC system, labour assignment rules need to be specified. Labour assignment rules considered in most DRC studies are the when-rule and the where-rule (see e.g. Hottenstein and Bowman 1998). The when-rule determines at what moment labour becomes eligible for transfer, while the where-rule determines to which work centre or machine a worker needs to be transferred once he/she is eligible for transfer. Bokhorst *et al*. (2004) studied the effect of a third assignment decision which they termed the who-rule. Based on worker differences, the who-rule determines which worker should be transferred to a work centre if more than one skilled worker is available. Common when-rules are the 'central' when-rule and the 'decentral' when-rule. With a central when-rule, a worker is eligible for transfer after each job he/she has finished at a machine and with a decentral when-rule, a worker is eligible for transfer after finishing all jobs at a machine. In this example, labour is assigned in the DRC systems according to a central when-rule, a First In System First Served (FISFS) where-rule (which sends a worker to the available machine with the 'oldest' job in queue) and a 'longest idle time' who-rule.

Figure 1 to be inserted about here

Figure 1a shows a specialised SRC system with two separate machines each processing their own job type (x or y) and figure 1b shows a pooled SRC system in which the queues are joined and both machines are able to process both job types (x and y). Compared to the specialised SRC system, which may leave one of the machines idle while a job waits in a queue, the advantage of the pooled system is that both servers always have access to jobs waiting in the queue. Pooling here thus creates more assignment possibilities for job types, which reduces waiting times in homogeneous systems. Furthermore, in the pooled system, all jobs in the combined queue can be considered when making a dispatching decision, while in the specialised system only the dedicated queue can be chosen from. For instance, if job type x with priority 2 (based on the system-wide FISFS rule) is finished, job type x with priority 5 is next in the specialised system (figure 1a) and job type y with priority 4 is next in the pooled system (figure 1b). This may have performance implications. However, as indicated earlier, previous research has shown that heterogeneous systems (those with non-identical servers and/or multiple job types) may frustrate the advantages of pooling even up to the point that pooling becomes disadvantageous.

Figure 1c shows a specialised worker-machine system, in which each machine can only be operated by one worker. Worker 1 is able to operate machines 1 (processing job type a) and 2 (processing job type b), while worker 2 is able to operate machine 3 (processing job type c). Since there is no overlap in the skills of the workers, this system is actually an SRC system (worker-only constrained). Figure 1d shows a cross-trained worker-machine system, in which both workers are able to operate all machines, representing a full flexibility cross-training configuration. Since only one machine is available for each job type and both workers are able to operate all machines, this system is DRC. Equal to the benefits of pooling in an SRC environment, cross-training creates more assignment possibilities for jobs (with their machines). For instance, job types a and b can be processed simultaneously in the flexible DRC system. However, the limited machine availability in the DRC system sometimes causes this benefit to be lost. For instance, if there is no demand for job types b and c, but ample demand for job type a, only one of the workers is able to process these jobs on machine 1 due to a lack of machine capacity. Also, cross-training enables a larger set of jobs to be considered in dispatching and assignment (where-rule) decisions. Where worker 2 had to start with job type c with priority 5 in the specialised system (figure 1c), he/she can work on job type b with priority 3 in the flexible DRC system (figure 1d). Again, however, the limited machine capacity may cause the flexible DRC system to deviate at times from the system-wide job priority sequence. For instance, if worker 1 finishes job type a with priority 2 on machine 1 in figure 1d, he/she has to commence with job type c with priority 5 on machine 3, since machine 2 is already occupied by worker 2 and thus job type b with priority 4 cannot be processed. Intuitively, one expects this to have a negative effect on the performance, but as we will show later, this not always appears to be correct in a DRC heterogeneous context. Furthermore, labour assignment rules may have an impact on the potential flexibility of the flexible DRC system. For instance, under a decentral when-rule, worker 1 first has to finish all jobs that need to be processed on machine 1 (including job priority 7, 8, and possibly other jobs arriving during the time worker 1 is still working on machine 1).

To conclude, the limited machine availability and possibly the labour assignment rules set (e.g. a decentral when-rule) cause the DRC system with cross-trained workers to behave differently than the pooled SRC system, irrespective of the type of environment (homogeneous or heterogeneous). Note that the example models different job types, but does not discuss any characterising differences between these job types. In a homogeneous environment, pooling is known to be beneficial in SRC systems as well as in DRC systems, despite of the differences between the concepts as described above. However, the question remains what the impact of these differences will be in comparing the performance of pooling (cross-training) in DRC heterogeneous systems with that of pooling in SRC heterogeneous systems. This will be examined further in the remainder of this paper.

4. Model characteristics

Our models with M workers and N machines ($M \geq 2$) satisfy a number of basic assumptions, i.e. machines can only be operated by one worker at a time, machines need constant tending, jobs only visit one machine, jobs and workers cannot be interrupted once started, there is unlimited waiting space, transferring jobs or workers does not take any time, there are no machine breakdowns, and there is no worker absenteeism. Note that a parallel system configuration is assumed, meaning there is one routing step per job. The parallel system configuration can be found in many functional departments of manufacturing firms, as in a firm that produces plastic bottle tops using a number of different plastic molding machines.

The current paper specifically considers the effect of cross-training in DRC systems with heterogeneous processing time distributions. For this, we model two groups of machines that receive either H-jobs or L-jobs, representing different processing characteristics. H-jobs will face high mean processing times (b_H) and L-jobs will face low mean processing times (b_L). The squared coefficients of variation of the processing times are c_H and c_L, respectively. In this paper we assume that $c_H = c_L = c$. Furthermore, since we use negative exponentially distributed processing times, $c = 1$. The H-jobs and L-jobs possess Poisson arrival rates λ_H and λ_L per machine. As in Buzacott (1996) for SRC systems, we state that the following relation holds $\lambda_H b_H = \lambda_L b_L$ and thus all M machines have an equal utilisation ρ_M .

In order to be able to create symmetrical configurations and thus be able to allocate an equal number of workers to each machine group, we only consider models for which M equals 2N, where N is an even number. Based on these assumptions, the worker utilisation is $\rho_W = 2\rho_M$. In addition the mean system processing time is

$$b_{LH} = \left(\frac{\lambda_L}{\lambda}\right)b_L + \left(\frac{\lambda_H}{\lambda}\right)b_H = \left(\frac{2b_L b_H}{b_L + b_H}\right) = \left(\frac{2r}{1+r}\right)b_L,$$

with $\lambda = \lambda_L + \lambda_H$ and the processing time (PT) ratio $r = \left(\frac{b_H}{b_L}\right) > 1$. Note: $b_L < b_{LH} < 2b_L$ and b_{LH} is an increasing, concave function of r $(r \geq 1)$. Finally, the mean flow time of the system (MFT) is the sum of the mean waiting and mean processing time of the system: $MFT = w + b_{LH}$.

The smallest symmetrical DRC system that is convenient for our research purposes is one with 4 machines and 2 workers, which we will call a (4-2) system, denoting the number of machines and the number of workers (M-N), respectively. In practice, such small (sub) systems are regularly found. Even for this small (sub) system, several cross-training configurations can be modelled, including those with cross-training (pooling). Figure 2a shows a fully cross-trained configuration, where both workers can operate all four machines. We call this the 4-2HE/P configuration, since there are 4 machines, 2 workers, workers receive HEterogeneous jobs (both H-jobs and L-jobs), and there is 'Pooling' within this configuration. Figures 2b and 2c show two Specialised configurations without pooling within the (4-2) system. In the 4-2HO/S configuration, workers receive HOmogeneous jobs (either H-jobs or L-jobs) and in the 4-2HE/S configuration, workers each operate one machine receiving H-jobs and one machine receiving L-jobs.

We will also include the larger (8-4) system to gain insight into the effect of system size. Figure 2d shows the fully cross-trained configuration within this system (8-4HE/P).

Compared to the full flexibility configuration in the (4-2) system, the redundancy of the machines (i.e. the number of workers able to operate a machine) is now four instead of two, which creates more assignment flexibility. On the other hand, machine constraints may be more apparent. Figure 2e shows a configuration where workers receive homogeneous jobs and are cross-trained (pooled) within the machine groups: 8-4HO/P. The DRC configurations with pooling (cross-trainings) considered in this paper are thus restricted to full flexibility configurations (i.e. 4-2HE/P and 8-4HE/P) and one limited flexibility DRC configuration (i.e. 8-4HO/P). Other limited flexibility configurations (e.g. chaining configurations) in which workers are possibly trained for a different number of machines within the two machine groups are to be considered in future research.

Figure 2 to be inserted about here

5. SRC approximations and closed form expressions of mean waiting (flow) times

The five worker-machine configurations we consider in section 4 are special variants of general systems that can be separated into two classes. Class I is characterized by N workers serving all M (=2N) machines, and includes 4-2HE/S, 4-2HE/P, and 8-4HE/P. In class II, the N machines operating L-jobs are served by ½N workers and the N machines operating H-jobs are served by the other ½N workers. Note that for this reason we require that N is even. Class II includes 4-2HO/S and 8-4HO/P.

The two specialised configurations we consider are SRC (4-2HO/S, 4-2HE/S) and the other configurations are DRC for which analytical results from queuing theory are not known. In the DRC configurations only one worker is allowed to serve the jobs in the queue in front of a machine. By relaxing this physical constraint, all systems in both classes can be 'approximated' by equivalent SRC configurations. For the two classes of SRC systems closed form expressions for the mean waiting (flow) time can be derived. Relaxing the machine constraints can be realised by adding for each machine additional machines in parallel to an amount that is at least equal to the number of workers that are capable of serving the machine. One expects that these SRC configurations would give a lower bound of the mean waiting time for the equivalent DRC configurations. However, generally this appears to be incorrect. We will discuss these results in the simulation section of the paper.

In DRC research the central and decentral when-rule are frequently used. In case of the SRC relaxation the central when-rule is equivalent to the FIFO (or FISFS) rule. For specialised SRC configurations (e.g. 4-2HE/S) the decentral when-rule is equivalent to the alternating (Avi-Itzhak 1965) or the exhaustive rule. In the appendix we provide the closed form expressions of the mean waiting (flow) times of systems of arbitrarily size in the two classes under a central and a decentral when-rule. Here we work out these expressions for the 5 SRC systems under consideration. In addition we assume that $c_H = c_L = 1$ and $\rho_W = 0.85$.

Class I: Mean waiting (flow) times of 4-2HE/S, 4-2HE/P$_{\text{SRC}}$ and 8-4HE/P$_{\text{SRC}}$.

The three configurations differ with respect to the number N of workers (and by this the number of machines). For 4-2HE/S, which can be regarded as 2x[2-1HE/S], $N = 1$, for 4-2HE/P$_{\text{SRC}}$ $N = 2$ and for 8-4HE/P$_{\text{SRC}}$ $N = 4$. Using (A.1) for the central when-rule (FIFO rule) and (A.7) for the decentral when-rule (alternating rule) gives the mean waiting times.

For the central when-rule holds

$$w(\text{4-2HE/S}^{\text{cen}}) = 5.667\left(\frac{1+r}{2}\right)b_L,$$

$$w(4\text{-}2\mathrm{HE/P}_{\mathrm{SRC}}^{\mathrm{cen}}) = 2.6036\left(\frac{1+r}{2}\right)b_L,$$

$$w(8\text{-}4\mathrm{HE/P}_{\mathrm{SRC}}^{\mathrm{cen}}) = 1.1488\left(\frac{1+r}{2}\right)b_L. \quad (1)$$

The expression for the mean flow time can be found from the mean waiting time by adding the mean processing time b_{LH}. It can be noticed that

$$w(8\text{-}4\mathrm{HE/P}_{\mathrm{SRC}}^{\mathrm{cen}}) < w(4\text{-}2\mathrm{HE/P}_{\mathrm{SRC}}^{\mathrm{cen}}) < w(4\text{-}2\mathrm{HE/S}^{\mathrm{cen}}).$$

This property corresponds with the well known pooling concept in queuing theory. The same ordering holds for the mean flow times.

The mean waiting times for the decentral when-rule are

$$w(4\text{-}2\mathrm{HE/S}^{\mathrm{dec}}) = 5.6667\left(\frac{1.15+1.79r+1.15r^2}{2.045(1+r)}\right)b_L,$$

$$w(4\text{-}2\mathrm{HE/P}_{\mathrm{SRC}}^{\mathrm{dec}}) = 2.6036\left(\frac{1.15+1.79r+1.15r^2}{2.045(1+r)}\right)b_L,$$

$$w(8\text{-}4\mathrm{HE/P}_{\mathrm{SRC}}^{\mathrm{dec}}) = 1.1488\left(\frac{1.15+1.79r+1.15r^2}{2.045(1+r)}\right)b_L. \quad (2)$$

The expression for $w(4\text{-}2\mathrm{HE/S}^{\mathrm{dec}})$ is exact, the other two are approximations based on using the fraction of the M/M/s and M/M/1 (see Appendix). Because of this also the pooling concept holds, i.e.

$$w(8\text{-}4\mathrm{HE/P}_{\mathrm{SRC}}^{\mathrm{dec}}) < w(4\text{-}2\mathrm{HE/P}_{\mathrm{SRC}}^{\mathrm{dec}}) < w(4\text{-}2\mathrm{HE/S}^{\mathrm{dec}}) \quad (3)$$

Moreover since for the exact mean values, $w(4\text{-}2\mathrm{HE/S}^{\mathrm{cen}})$ and $w(4\text{-}2\mathrm{HE/S}^{\mathrm{dec}})$, holds

$$w(4\text{-}2\mathrm{HE/S}^{\mathrm{dec}}) \geq w(4\text{-}2\mathrm{HE/S}^{\mathrm{cen}}),$$

also the approximated decentral when-rule waiting times satisfy

$$w(4\text{-}2\mathrm{HE/P}_{\mathrm{SRC}}^{\mathrm{dec}}) \geq w(4\text{-}2\mathrm{HE/P}_{\mathrm{SRC}}^{\mathrm{cen}}),\ w(8\text{-}4\mathrm{HE/P}_{\mathrm{SRC}}^{\mathrm{dec}}) \geq w(8\text{-}4\mathrm{HE/P}_{\mathrm{SRC}}^{\mathrm{cen}}). \quad (4)$$

Note that for the 4-2HE/S configuration, which can be regarded as 2x[2-1HE/S], the decentral when-rule forces each worker to alternate between serving L-jobs and H-jobs. In the appendix, the waiting time of the 2-1HE/S configuration is extrapolated to the multi-server case. This approximation appears to be good in case of G/G/s systems (and thus also in case of M/G/s systems) using FIFO and head of the line priority policies (see e.g. Buzacott and Shanthikumar 1993). However, in the M/G/s systems we consider here, with L-jobs and H-jobs, some workers will serve L-jobs and others will serve H-jobs and thus a pure alternating policy is not followed. For that reason, we expect the approximated waiting times of the 4-2HE/$\mathrm{P_{SRC}}$ and 8-4HE/$\mathrm{P_{SRC}}$ configurations to be too heavily influenced by the 'pure alternating' effect of the 2-1HE/S configuration. In other words, we expect the real mean waiting time of the decentral when-rule to be in between the approximated mean waiting time and the mean waiting time of the central when-rule. In the next section we will simulate these configurations to find out what the mean waiting time really is.

Class II: Mean waiting (flow) times of 4-2HO/S and 8-4HO/$\mathrm{P_{SRC}}$.

The two configurations 4-2HO/S and 8-4HO/$\mathrm{P_{SRC}}$ differ with respect to the number N of workers. For 4-2HO/S $N = 2$, and for 8-4HO/$\mathrm{P_{SRC}}$ $N = 4$. Halve of the operators serve L-jobs and the other halve serve H-jobs. Thus the subsystems of 4-2HO/S or 8-4HO/$\mathrm{P_{SRC}}$ can be seen as an M/G/n system with n=½*N*. Since a subsystem either serves L-jobs or H-jobs the decentral when-rule has no meaning.

Using (A.10) the mean waiting times are

$$w(\text{4-2HO/S}) = 5.6667\left(\frac{2r}{1+r}\right)b_L,$$

$$w(\text{8-4HO/P}_{\text{SRC}}) = 2.6036\left(\frac{2r}{1+r}\right)b_L. \tag{5}$$

Again due to the pooling concept

$$w(\text{8-4HO/P}_{\text{SRC}}) < w(\text{4-2HO/S}). \qquad (6)$$

Comparison between classes

For a given number of workers, the best configurations in each class are compared. If N=2 we have in class I 4-2HE/S (or 2x[2-1HE/S]) and 4-2HE/P_{SRC}, where 4-2HE/P_{SRC} outperforms 4-2HE/S. In class II we only have 4-2HO/S. Thus we compare the mean waiting times of 4-2HE/P_{SRC} and 4-2HO/S. Using the central when-rule in both cases we compare

$$w(\text{4-2HE/P}_{\text{SRC}}^{\text{cen}}) = 2.6036\left(\frac{1+r}{2}\right)b_L, \; w(\text{4-2HO/S}) = 5.6667\left(\frac{2r}{1+r}\right)b_L.$$

The configurations in class I are linearly increasing with r, the configurations in class II are concave increasing and bounded from above as function of r. In addition for $r=1$ holds $w(\text{4-2HE/P}_{\text{SRC}}^{\text{cen}}) < w(\text{4-2HO/S})$. Thus a unique PT ratio (r) exists for which $w(\text{4-2HE/P}_{\text{SRC}}^{\text{cen}}) = w(\text{4-2HO/S})$, giving the condition $r^2 - 6.706r + 1 = 0$ and solution for the crossing point $r^* = 6.55$. Thus for $r > r^*$ we have $w(\text{4-2HO/S}) < w(\text{4-2HE/P}_{\text{SRC}}^{\text{cen}})$. This shows that specialisation is attractive compared to pooling for large differences in job processing times. A notion already known from literature, see e.g. Buzacott (1996).

For N=4 we have in class I the configurations 2x[4-2HE/S], 2x[4-2HE/P_{SRC}], and 8-4HE/P_{SRC}. Here 8-4HE/P_{SRC} outperforms the others. In class II we have the configurations 2x[4-2HO/S] and 8-4HO/P_{SRC}, where 8-4HO/P_{SRC} has the smallest mean waiting time. Consequently we compare the mean waiting (flow) times of 8-4HE/P_{SRC} and 8-4HO/P_{SRC} using the central when-rule

$$w(\text{8-4HE/P}_{\text{SRC}}^{\text{cen}}) = 1.1488\left(\frac{1+r}{2}\right)b_L, \; w(\text{8-4HO/P}_{\text{SRC}}) = 2.6036\left(\frac{2r}{1+r}\right)b_L.$$

Based on the same arguments as for N=2 we have, for $r > r^*$, $w(\text{8-4HO/P}_{\text{SRC}}) < w(\text{8-4HE/P}_{\text{SRC}}^{\text{cen}})$. This shows again that specialisation is attractive compared to pooling for large differences in job processing times. The crossing point r^* can be calculated solving the polynomial $r^2 - 7.067r + 1 = 0$ giving the PT ratio $r^* = 6.92$.

We could also compare the best configurations in class I using the decentral when-rule with the best configurations in class II where no difference exists between the central and decentral when-rule. Since the mean waiting time for the decentral when-rule is also increasing in r there is a unique crossing point. However since the mean waiting time is larger than the waiting time of the central when-rule r^* will be (somewhat) smaller in both cases (N=2 and N=4). Note that the same crossing points are found if the mean flow time is considered instead of the mean waiting time.

Since the comparisons of the configurations are (partly) based on approximations of the DRC configurations, the next section presents the simulation study to find whether these approximations are accurate for the DRC configurations.

6. Simulation study

We use discrete event simulation to gain insights in the impact of machine constraints and the use of different when-rules on the effect of pooling in heterogeneous systems. All simulation models are written in the object-oriented simulation software package Tecnomatix Plant Simulation 7.6 (Texas: UGS Corporation). The replication/deletion approach is used to estimate the steady-state means of the output parameters (see e.g., Law and Kelton 2000: 525).

The general model assumptions and characteristics are described in section 4. Fixed factors are the dispatching rule (FIFO), the where-rule (FISFS), and the who-rule (longest idle time). The dispatching rule chooses the job from the machine queue if a machine and worker request it. An available worker (the one who is idle longest in case more workers are available) is thus sent to an available machine with the oldest job in queue and that job is then dispatched.

The experimental factors are PT ratio, cross-training configurations, and the when-rule. We use seven specific levels of the processing time ratio r (i.e. 1:1, 5:1, 10:1, 15:1, 20:1, 25:1, and 30:1). The three DRC configurations $4\text{-}2HE/P_{DRC}$, $8\text{-}4HE/P_{DRC}$, and $8\text{-}4HO/P_{DRC}$ are simulated, since no analytical queuing formulae exist for these configurations. Since the mean waiting time expressions in section 5 for two SRC configurations $4\text{-}2HE/P_{SRC}$ and $8\text{-}4HE/P_{SRC}$ are approximations, we also simulate these configurations for a fair comparison. Furthermore, we simulate the central and the decentral when-rule. The $4\text{-}2HE/P_{SRC}$ and $8\text{-}4HE/P_{SRC}$ approximations assume that the arrivals in the machine queues are joined into one single queue where a certain queuing discipline is applied (central or decentral in our case). In simulation, especially in the case of a decentral rule, it might be worthwhile to take into account the machine (queue) a job came from. As in DRC systems, these 'machine'-related jobs then belong to one of N different subclasses of L-jobs or H-jobs all to be found in the single queue. Under a decentral when-rule, creating subclasses means that the workers will be eligible for transfer more often than without subclasses. We will thus simulate these SRC configurations under a decentral when-rule with M=2N subclasses, to enable a fair comparison with the DRC systems, and without subclasses, to compare the results with the analytical approximations stated in section 5.

In the next section, where the results are displayed and analysed, we will first use the analytical (approximated or exact) results of the specialised and pooled SRC configurations using the formulae in section 5 and compare these with the outcomes of the simulated DRC configurations under all level combinations of PT ratio and When. Section 7.2 then focuses on the pooled SRC configurations by comparing the analytical approximations with the simulation results.

7. Results and analysis

7.1. Comparing DRC with SRC systems in a heterogeneous environment

In this subsection, we compare the performance of DRC configurations (obtained by simulation) with the performance of SRC configurations (obtained analytically). We will first describe the behaviour of each DRC configuration and compare this with the behaviour of SRC configurations under a central when-rule and then we will discuss the impact of a decentral when-rule. Subsequently, we will show the impact of the differences found on the question whether to pool (cross-train) or not in a heterogeneous environment. Table 1 shows the relevant MFT results for this subsection.

Table 1 to be inserted about here

Figure 3 shows the MFT results for all configurations under a central when-rule, obtained analytically or by means of simulation for all levels of PT ratio. The main observation is that the DRC outcomes show a similar pattern as the SRC outcomes. With a PT ratio of 1:1, representing a homogeneous environment, the specialised configurations (4-2HO/S and $\text{4-2HE/S}^{\text{cen}}$) are equal, since H-jobs and L-jobs are equal. Also, $\text{4-2HE/P}^{\text{cen}}_{\text{SRC}}$ equals $\text{8-4HO/P}_{\text{SRC}}$ and $\text{4-2HE/P}^{\text{cen}}_{\text{DRC}}$ equals $\text{8-4HO/P}_{\text{DRC}}$, since in this case these specific 8-4 configurations are duplications of the 4-2 configurations mentioned. The pooled configurations perform better than the specialised configurations. The $\text{8-4HE/P}^{\text{cen}}_{\text{SRC}}$ and $\text{8-4HE/P}^{\text{cen}}_{\text{DRC}}$ configurations with the highest level of pooling obviously show the best performance in the homogeneous environment. Furthermore, within this homogeneous environment, imposing machine constraints is detrimental to MFT performance.

Figure 3 to be inserted about here

Figure 3 distinctly shows the three configurations in class I (i.e. 4-2HE/S^{cen}, 4-2HE/P^{cen}_{SRC}, and 8-4HE/P^{cen}_{SRC}) for which the MFTs nearly linearly increase for large PT ratios (see Appendix) Also, the two configurations in class II (i.e. 4-2HO/S and 8-4HO/P_{SRC}) can be clearly distinguished from the others since their MFTs are concave and bounded The benefits of pooling within each class, as shown in section 5, are nicely depicted. We expected the simulation results for the DRC configurations (i.e. 4-2HE/P^{cen}_{DRC}, 8-4HE/P^{cen}_{DRC} and 8-4HO/P^{cen}_{DRC}) to always be worse than the results for their SRC counterparts, since the SRC configurations relax the machine restrictions, enabling workers to serve jobs in the same queue. However, this only appears to be correct for 8-4HO/P^{cen}_{DRC}, in which workers only process one job-type. The simulation results of 4-2HE/P^{cen}_{DRC} and 8-4HE/P^{cen}_{DRC} even outperform their SRC counterparts for larger PT ratios, which is a remarkable outcome. Apparently, the changed processing sequence of jobs—due to machine limitations in the DRC systems—turns out to be beneficial compared to the processing sequence of jobs in the SRC system. In going from an SRC to a DRC system, the mean flow times of the H-tasks seem to deteriorate under processing time differences and those of the L-tasks seem to improve.

The decentral when-rule was shown to deteriorate the performance for 4-2HE/S, 4-2HE/P_{SRC} and 8-4HE/P_{SRC} in section 5. The DRC results for 4-2HE/P_{DRC} and 8-4HE/P^{cen}_{DRC} here show that the decentral when-rule does not deteriorate the performance and even seems to slightly improve the MFT performance under large PT ratios for 8-4HE/P_{DRC}.

We will now discuss the impact of the differences found on the question whether to pool (cross-train) or not in a heterogeneous environment. For this, we will first focus on the (4-2) system. 4-2HO/S is the best performing specialised configuration and $\text{4-2HE/P}_{\text{SRC}}$ and $\text{4-2HE/P}_{\text{DRC}}$ are the best performing flexible configurations with respect to SRC and DRC systems, respectively. Figure 4 shows the MFT of these configurations under a central and a decentral when-rule.

Figure 4 to be inserted about here

The crossing point (r^*) at which 4-2HO/S starts to outperform $\text{4-2HE/P}_{\text{DRC}}^{\text{cen}}$ is about the same as that for $\text{4-2HE/P}_{\text{SRC}}^{\text{cen}}$ and 4-2HO/S (which was 6.55, see section 5). Dedicating workers to a job type then thus becomes more advantageous than training them for both job types. Applying a decentral when-rule instead of a central when-rule here only effects (deteriorates) MFT performance in the pooled SRC configuration ($\text{4-2HE/P}_{\text{SRC}}$), where the effect is stronger under higher levels of PT ratio. The effect on the crossing point is minimal.

Within the (8-4) system, 8-4HO/P is the best performing specialised configuration for SRC and DRC systems. This configuration increases the redundancy of machines and the multifunctionality of workers while keeping the workers dedicated to similar machines. It thus incorporates cross-training (pooling) and logically performs better than specialisation without cross-training (i.e. 4x[2-1HE/S]). As mentioned before, $\text{8-4HO/P}_{\text{DRC}}$ performs worse than $\text{8-4HO/P}_{\text{SRC}}$. The $\text{8-4HE/P}_{\text{SRC}}$ and the $\text{8-4HE/P}_{\text{DRC}}$ are the best performing flexible configurations with respect to SRC and DRC systems, respectively. Figure 5 shows the MFT of the configurations under a central and a decentral when-rule.

Figure 5 to be inserted about here

Since the specialised configuration 8-4HO/P_{DRC} performs worse than 8-4HO/P_{SRC} and the flexible configuration 8-4HE/P_{DRC}^{cen} performs better than 8-4HE/P_{SRC}^{cen} for larger PT ratios (note that we expected it to perform worse), the crossing point at which 8-4HO/P_{DRC}^{cen} starts to outperform 8-4HE/P_{DRC}^{cen} is at a higher PT ratio (around 8.0) then the point at which 8-4HO/P_{SRC} starts to outperform 8-4HE/P_{SRC}^{cen} (which was 6.92, see section 5). Applying a decentral when-rule instead of a central when-rule seems to increase this difference even more.

To conclude, the results show that pooling (cross-training) in a DRC heterogeneous environment does not improve MFT performance compared to dedicating workers to similar machines (specialisation) if the differences between the mean processing times of the H-jobs and L-jobs is large. Note again that pursuing cross-training within homogeneous subgroups of machines remains beneficial. This general pattern is similar to that found within the SRC heterogeneous environment (see also section 2.1). Whereas we expected DRC systems to perform worse than the equivalent SRC systems, the results showed otherwise. The limited machine availability in DRC systems and—to a lesser extent—the use of a decentral when-rule seem to benefit the fully cross-trained DRC configurations. This even tends to shift the point at which specialisation becomes better to heterogeneous environments with larger differences between processing times than in the SRC systems. Still, managers should be aware of the detrimental effect of cross-training in heterogeneous systems with large differences between processing times.

7.2. *Simulation of pooled SRC heterogeneous systems*

Since the 4-2HE/P_{SRC} and 8-4HE/P_{SRC} performance results in the previous section are based on analytical approximations (which also assume no subclasses), we simulated these SRC configurations with a central when-rule and with a decentral when-rule with and without subclasses in an additional experiment. By this, the accuracy of the analytical approximations under a central when-rule and a decentral when-rule without subclasses can be validated and the possible effect of distinguishing subclasses under a decentral when-rule can be shown.

For 4-2HE/P_{SRC} and 8-4HE/P_{SRC}, table 2 shows the analytical results and the simulation results under all level combinations of When and PT ratio. 'Decentral when-no' represents the results under a decentral when-rule without subclasses, 'decentral when-yes' represents the results with subclasses.

Table 2 to be inserted about here

The analytical approximations for 4-2HE/P_{SRC}^{cen} and 8-4HE/P_{SRC}^{cen} show higher mean flow times than the simulation results do under processing time differences. Even though this was expected, since the approximation is known to overestimate the true results for c>1, the difference is quite large for higher PT ratios. It is interesting to see that the simulation outcomes of 4-2HE/P_{SRC}^{dec} and 8-4HE/P_{SRC}^{dec} without subclasses are similar to their simulated counterparts under a central when-rule. Whereas the decentral when-rule showed to deteriorate the performance in a single-server queuing system (i.e. 4-2HE/S) and we expected it to slightly deteriorate the performance in the multiple-server queuing systems (i.e. 4-2HE/P_{SRC} and 8-4HE/P_{SRC}), the simulation results show that the decentral when-rule does not effect performance in the multiple-server queuing systems The transformation of expression (A.6) (see appendix), based on Avi-Itzhak (1965) and Stidman (1972) for the M/G/1 system, to (A.7) for M/G/N systems thus turns out to be inaccurate. In case subclasses are modelled, the decentral when-rule even seems to improve the performance under higher levels of PT ratio. In the case that differences are found between the central and decentral when-rule with subclasses, the mean flow times of the H-tasks deteriorate and those of the L-tasks improve, suggesting that L-jobs are given priority under the decentral when-rule with subclasses.

Since the simulated performance outcomes of 4-2HE/P_{SRC} and 8-4HE/P_{SRC} are better than the analytical approximations, the crossing points with 4-2HO/S and 8-4HO/P_{SRC}, respectively, now come closer to the crossing points of their DRC counterparts.

8. Conclusions and future research

This paper investigated the effect of cross-training in Dual Resource Constrained (DRC) systems with heterogeneous processing time distribution characteristics. The manager's tendency to increase workforce flexibility in DRC manufacturing systems combined with the known detrimental effects of pooling queues within single resource constrained (SRC) systems with non-identical servers and/or multiple job types motivated the study. We showed that the concept of cross-training workers in a DRC system works out differently than the concept of pooling queues in an SRC system. The (positive or negative) effect of the increased assignment flexibility in pooled (cross-trained) SRC systems cannot always be realised in DRC systems due to the limited availability of machines. A decentral when-rule, which is a frequently used assignment rule in DRC systems in practice, further impacts the use of the potential assignment flexibility in pooled systems.

With queuing theory and additional simulations of (DRC) configurations we showed that the detrimental effects of pooling (cross-training) in SRC heterogeneous systems are also apparent in DRC systems. That is, fully cross-training workers in a heterogeneous environment is only beneficial if the difference between the mean processing times is not large (until a PT ratio of between 5:1 and 10:1). Otherwise, cross-training should not be pursued or—if allowed by the size of the system—it should be pursued within homogeneous subgroups of machines. Managers should take this into account when making cross-training decisions.

The simulation results showed that under large processing time differences, DRC configurations with full cross-training even outperform the equivalent SRC configurations. This is a remarkable result, since the relaxation of machine constraints was thought to give a lower bound on the mean flow time performance of the DRC configurations. Limiting the assignment flexibility of a fully flexible SRC configuration in a heterogeneous environment by imposing machine constraints (i.e. turning it into a DRC configuration) apparently changes the processing sequence of jobs beneficially.

The use of a decentral when-rule deteriorates the mean flow time performance in single server queuing systems serving multiple job types (which are SRC by definition), while it seems to improve the performance in multiple server fully flexible queuing systems (modelled with subclasses) under large processing time differences. Again, limiting the assignment flexibility–now by means of a labour assignment rule–is found to improve system performance.

Since the full flexibility configurations in our study encompass the configurations that dedicate workers to similar machines, specifically designed labour assignment rules (where-rule, who-rule) must be able to realise at least the same performance in the full flexibility configurations as in the specialised configurations. The design and evaluation of such labour assignment rules in DRC heterogeneous systems may be a topic for further research. Similarly, the possibilities of enforcing a restricted use of pooling flexibility in

SRC heterogeneous systems may be a research extension. Another extension would be to include limited flexibility configurations (e.g. chaining configurations) in which workers are possibly trained for a different number of machines within the two machine groups. Finally, the effects of differences in worker characteristics as one of the causes for heterogeneous processing times could be considered in future research.

Appendix

In this appendix we show the general expressions for the mean waiting (flow) times of the SRC configurations in the two classes.

<u>Class I: *N workers serving all M machines*</u>

Since the SRC configurations relax the machine constraints, we can join the N arrivals of L-jobs and the N arrivals of H-jobs into a single queue and describe the system as an $M/G/s,\ \ s=N$ system. In this case the central when-rule is equal to the FIFO rule. Except for the M/G/1 system (Pollaczek Khintchine) closed form expressions of the mean waiting (flow) time of an M/G/s system are not known. However many approximated formulae exist (see e.g. Kimura 1994). Here we will use a frequently used approximation, first obtained by Lee and Longton (Kimura 1994) which is based on

$$w_{M/G/s} \approx \left(\frac{w_{M/M/s}}{w_{M/M/1}}\right) w_{M/G/1}.$$

Substituting the closed form expressions of the waiting times result in

$$w^{FIFO} = \left(\frac{\pi_s}{\nu s}\right)\left(\frac{1+c_s}{2}\right)\left(\frac{\nu}{1-\nu}\right) b_s,$$

$$\pi_s = \left(\frac{s^s \nu^s}{s!(1-\nu)\left(\sum_{j=0}^{s-1} \frac{s^j \nu^j}{j!} + \frac{s^s \nu^s}{s!(1-\nu)}\right)}\right),\ \ s = 1, 2, 3, \ldots, \qquad \text{(A.1)}$$

and ν the traffic intensity, b_s the mean and c_s the squared coefficient of variation of the processing time. Since we will also discuss the alternating rule, we have indicated the mean waiting time as w^{FIFO} here.

For our configurations the mean processing time is $b_s = b_{LH}$. The squared coefficient of variation of the processing time $c_s = c_{LH}$ can be found from the second moment of the joint processing time satisfying

$$E\{\tilde{b}^2\} = \left(\frac{\lambda_L}{\lambda}\right) E\{\tilde{b}_L{}^2\} + \left(\frac{\lambda_H}{\lambda}\right) E\{\tilde{b}_H{}^2\}.$$

Using the model characteristics in section 4 and moreover assuming $c_H = c_L = c$ we can write for the coefficient of variation c_{LH}

$$(1 + c_{LH}) = (1+c)\left(\frac{b_L + b_H}{2b_{LH}}\right) = (1+c)\left(\frac{(1+r)^2}{4r}\right) \qquad \text{(A.2)}$$

Note: $c_{LH} > c$ and increases with $r = \left(\frac{b_H}{b_L}\right)$ $(r > 1)$.

Substituting the squared coefficient of variation in (A.1) and using $\nu = \rho_W$, $s = N$ gives

$$w_{M/G/N}^{FIFO} = \left(\frac{\pi_N}{\rho_W N}\right)\left(\frac{1+c}{2}\right)\left(\frac{\rho_W}{1-\rho_W}\right)\left(\frac{1+r}{2}\right) b_L. \qquad \text{(A.3)}$$

Remark: The mean waiting time increases linearly with increasing r (given b_L). This is caused by the multiplication of b_{LH} and $(1+c_{LH})$ in (A.1). The mean flow time adds the mean processing time b_{LH} to the mean waiting time. The mean processing time is an increasing, concave function of r and has an upper bound $2b_L$ for $r \to \infty$. Thus the mean flow time is an increasing, concave function of r and increases nearly linearly for large r.

Instead of the FIFO rule (central when-rule) we will also consider the alternating rule, because of the equivalence with the decentral when-rule in DRC systems. In the literature this rule has also been called the exhaustive rule (see the polling model literature for example Takagi, H., 1988). Also in this case no closed form expressions for the mean waiting time of the M/G/s system exist, except for the M/G/1 case with two job classes 1 and

2 (Avi-Itzhak 1965, Stidman 1972). We will use an approximation used by Buzacott and Shanthikumar (1993, page 88) for the head of the line priority policy

$$w_{1:M/G/s} \approx \left(\frac{w_{M/M/s}}{w_{M/M/1}}\right) w_{1:M/G/1}, w_{2:M/G/s} \approx \left(\frac{w_{M/M/s}}{w_{M/M/1}}\right) w_{2:M/G/1}$$

where $w_{1:M/G/s}$ and $w_{2:M/G/s}$ are the mean waiting times of job class 1 and 2. Since the ratio is the same as for the $w^{FIFO}_{M/G/N}$ case we work out here Avi-Itzhak's result for $w_{1:M/G/1}$ and $w_{2:M/G/1}$

$$w_{1:M/G/1} = \left(\frac{1+c_1}{2}\right)\left(\frac{\rho_1}{(1-\rho_1)}\right) b_1 + \left(\frac{\rho_2[(1+c_2)(1-\rho_1)^2 b_2 + (1+c_1)\rho_1\rho_2 b_1]}{2(1-\rho_1)(1-\rho)(1-\rho+2\rho_1\rho_2)}\right),$$

and by symmetry

$$w_{2:M/G/1} = \left(\frac{1+c_2}{2}\right)\left(\frac{\rho_2}{(1-\rho_2)}\right) b_2 + \left(\frac{\rho_1[(1+c_1)(1-\rho_2)^2 b_1 + (1+c_2)\rho_1\rho_2 b_2]}{2(1-\rho_2)(1-\rho)(1-\rho+2\rho_1\rho_2)}\right),$$

with $\rho = \rho_1 + \rho_2$, $\rho_1 = \lambda_1 b_1, \rho_2 = \lambda_2 b_2$ and c_1, c_2 the squared coefficient of variation of the processing times. In our case we have equal utilisations and squared coefficient of variations giving

$$w^{Alt}_{L:M/G/1} = \left(\frac{1+c}{2}\right)\left(\frac{\rho_W}{(1-\rho_W)}\right)\left(\frac{((2-3\rho_W+2(\rho_W)^2)+r(2-\rho_W))}{2(2-2\rho_W+(\rho_W)^2)}\right) b_L \qquad \text{(A.4)}$$

$$w^{Alt}_{H:M/G/1} = \left(\frac{1+c}{2}\right)\left(\frac{\rho_W}{1-\rho_W}\right)\left(\frac{(2-\rho_W)+r(2-3\rho_W+2(\rho_W)^2)}{2(2-2\rho_W+(\rho_W)^2)}\right) b_L \qquad \text{(A.5)}$$

The system mean waiting time is found from the weighted sum

$$w^{Alt}_{M/G/1} = \left(\frac{\lambda_L}{\lambda}\right) w^{Alt}_{L:M/G/1} + \left(\frac{\lambda_H}{\lambda}\right) w^{Alt}_{H:M/G/1}.$$

Thus results in the expression

$$w^{Alt}_{M/G/1} = \left(\frac{1+c}{2}\right)\left(\frac{\rho_W}{1-\rho_W}\right)\left(\frac{2r(2-3\rho_W+2(\rho_W)^2)+(1+r^2)(2-\rho_W)}{2(1+r)(2-2\rho_W+(\rho_W)^2)}\right) b_L \qquad \text{(A.6)}$$

The alternating rule for this case has the properties (for $r > 1$) :

1: $w^{Alt}_{L:M/G/1} > w^{Alt}_{H:M/G/1}$ though this is not generally true for the alternating rule (see Avi-Itzhak 1965) L-jobs here have larger waiting times than H-jobs;

2: $w^{Alt}_{L:M/G/1} > w^{FIFO}_{L:M/G/1}$, $w^{Alt}_{H:M/G/1} < w^{FIFO}_{H:M/G/1}$; $w^{FIFO}_{L:M/G/1} = w^{FIFO}_{H:M/G/1} = w^{FIFO}_{M/G/1}$

3: $w^{Alt}_{M/G/1} > w^{FIFO}_{M/G/1}$.

Knowing the waiting time for $s=1$ the approximated mean waiting time for $s=N$ can be found by the multiplication with the ratio $(\pi_N / \rho_W N)$ giving

$$w^{Alt}_{M/G/N} = \left(\frac{\pi_N}{N\rho_W}\right)\left(\frac{1+c}{2}\right)\left(\frac{\rho_W}{1-\rho_W}\right)\left(\frac{2r(2-3\rho_W+2(\rho_W)^2)+(1+r^2)(2-\rho_W)}{2(1+r)(2-2\rho_W+(\rho_W)^2)}\right)b_L \qquad \text{(A.7)}$$

<u>Class II: *N machines operating L-jobs (H-jobs) are served by ½N workers*</u>

The configurations in this class can be seen as an M/G/s system with s=½*N*. Since a subsystem either serves L-jobs or H-jobs the alternating rule has no meaning. The mean waiting times of both systems become

$$w^{FIFO}_{L:M/G/n} = \left(\frac{\pi_n}{\rho_W n}\right)\left(\frac{1+c}{2}\right)\left(\frac{\rho_W}{1-\rho_W}\right)b_L, n = N/2, \qquad \text{(A.8)}$$

$$w^{FIFO}_{H:M/G/n} = \left(\frac{\pi_n}{\rho_W n}\right)\left(\frac{1+c}{2}\right)\left(\frac{\rho_W}{1-\rho_W}\right)rb_L. \qquad \text{(A.9)}$$

These results in the system mean waiting time of

$$w^{FIFO}_{M/G/n} = \left(\frac{\pi_n}{\rho_W n}\right)\left(\frac{1+c}{2}\right)\left(\frac{\rho_W}{1-\rho_W}\right)b_{LH} \qquad \text{(A.10)}$$

Remark 1: Contrary to the performance of configurations in class I the main waiting time here is concave increasing having an upper bound as function of r $(r \ge 1)$. The upper bound is reached if $r \to \infty$ and by this $b_{LH} \to 2b_L$.

Remark 2: For $c_L = c_H = 1$ both subsystems in this class are M/M/s systems and the closed form expression of the mean waiting time is exact.

References

Agnihothri, S.R. and Mishra, A.K., 2004. Cross-training decisions in field services with three job types and server-job mismatch. *Decision Sciences,* 35 (2), 239-257.

Armbruster, D. and Gel, E.S., 2006. Bucket brigades revisited: are they always effective? *European Journal of Operational Research,* 172, 213-229.

Armbruster, D., Gel, E.S. and Murakami, J., 2007. Bucket brigades with worker learning. *European Journal of Operational Research,* 176, 264-274.

Avi-Itzhak, B., Maxwell, W.L. and Miller, L.W., 1965. Queuing with Alternative Priorities. *Operations Research*, 13 (2), 306-318.

Bartholdi III, J.J. and Eisenstein, D.D., 1996. A production line that balances itself. *Operations Research*, 44 (1), 21-34.

Benjaafar, S., 1995. Performance bounds for the effectiveness of pooling in multi-processing systems. *European Journal of Operational Research*, 87, 375-388.

Benjaafar, S., Talavage, J. and Ramakrishnan, R., 1995. The effect of routeing and machine flexibility on the performance of manufacturing systems. *International Journal of Computer Integrated Manufacturing*, 8 (4), 265-276.

Chakravarthy, S.R. and Agnihothri, S.R., 2005. Optimal workforce mix in service systems with two types of customers. *Production and Operations Management*, 14 (2), 218-231.

Bokhorst, J.A.C., Slomp, J. and Gaalman G.J.C., 2004. On the who-rule in Dual Resource Constrained (DRC) manufacturing systems. *International Journal of Production Research*, 42 (23), 5049-5074.

Bokhorst, J.A.C., Slomp, J. and Molleman, E., 2004. Development and evaluation of cross-training policies for manufacturing teams. *IIE Transactions*, 36 (10), 969-984.

Bokhorst, J.A.C., Slomp, J. and Gaalman G.J.C., 2006. Cooperation in a distributed dual resource constrained manufacturing system. *International Journal of Computer Applications in Technology*, 26 (1/2), 83-90.

Buzacott, J.A., 1996. Commonalities in reengineered business processes: models and issues. *Management Science*, 42 (5), 768-782.

Buzacott, J.A., 1999. The structure of manufacturing systems: insights on the impact of variability. *The International Journal of Flexible Manufacturing Systems*, 11, 127-146.

Buzacott, J.A. and Shanthikumar, J.G., 1993. *Stochastic models of manufacturing systems.* Englewood Cliffs, NJ: Prentice-Hall.

Campbell, G.M., 1999. Cross-Utilization of workers whose capabilities differ. *Management Science*, 45 (5), 722-732.

Daniels, R.L., Mazzola, J.B. and Shi, D., 2004. Flow shop scheduling with partial resource flexibility. *Management Science*, 50 (5), 658-669.

El-Taha, M. and Maddah, B., 2006. Allocation of service time in a multiserver system. *Management Science*, 52 (4), 623-637.

Fry, T.D., Kher, H.V. and Malhotra, M.K., 1995. Managing worker flexibility and attrition in dual resource constrained job shops. *International Journal of Production Research*, 33, 2136-2179.

Gans, N., Koole, G. and Mandelbaum, A., 2003. Telephone call centers: tutorial, review, and research prospects. *Manufacturing & Service Operations Management*, 5 (2), 79-141.

Gargeya, V. B. and Deane, R. H., 1996. Scheduling research in multiple resource constrained job shops: a review and critique. *International Journal of Production Research*, 34 (8), 2077-2097.

Gurumurthi, S. and Benjaafar, S., 2004. Modeling and analysis of flexible queueing systems. *Naval Research Logistics*, 51, 755-782.

Hopp, W.J., Tekin, E. and Van Oyen, M.P., 2004. Benefits of skill chaining in serial production lines with cross-trained workers. *Management Science*, 50 (1), 83-98.

Hopp, W.J. and Van Oyen, M.P., 2004, Agile workforce evaluation: a framework for cross-training and coordination. *IIE Transactions*, 36, 919-940.

Hottenstein, M.P. and Bowman, S.A., 1998. Cross-training and worker flexibility: A review of DRC system research. *The Journal of High Technology Management Research*, 9 (2), 157-174.

Inman, R.R., Jordan, W.C. and Blumenfeld, D.E., 2004. Chained cross-training of assembly line workers. *International Journal of Production Research*, 42 (10), 1899-1910.

Iravani, S.M., Van Oyen, M.P. and Sims, K.T., 2005. Structural flexibility: a new perspective on the design of manufacturing and service operations. *Management Science*, 51 (2), 151-166.

Iravani, S.M.R., Kolfal, B. and Van Oyen, M.P., 2007. Call-center labor cross-training: it's a small world after all. *Management Science*, 53 (7), 1102-1112.

Jordan, W.C. and Graves, S.C., 1995. Principles on the benefits of manufacturing process flexibility. *Management Science*, 41, 577-594.

Jordan, W.C. Inman, R.R. and Blumenfeld, D.E., 2004. Chained cross-training of workers for robust performance. *IIE Transactions*, 36, 953-967.

Jouini, Q., Dallery, Y. and Nait-Abdallah, R., 2008. Analysis of the impact of team-based organizations in call center management. *Management Science*, 54 (2), 400-414.

Kher, H.V. and Fry, T.D., 2001. Labour flexibility and assignment policies in a job shop having incommensurable objectives. *International Journal of Production Research*, 39 (11), 2295-2311.

Kimura, T., 1994. Approximations for Multi-server Queues: System Interpolations. *Queueing Systems*, 17, 347-382.

Kleinrock, L., 1976. *Queueing Systems, Vol. II: Computer Applications.* New York: Wiley.

Law, A.M. and Kelton, W.D., 2000. *Simulation Modeling and Analysis*. USA: McGraw-Hill.

Lippolt, C.R., Arnold, D. and Dörrsam, 2003. Analysis of a single stage production system with heterogeneous machines. *OR Spectrum*, 25, 97-107.

Malhotra, M.K., Fry, T.D., Kher, H.V. and Donohue, J.M., 1993. The impact of learning and labor attrition of worker flexibility in dual resource constrained job shops. *Decision Sciences*, 24 (3), 641-663.

Mandelbaum, A. and Reiman, M.I., 1998. On pooling in queueing networks. *Management Science*, 44 (7), 971-981.

McClain, J.O., Schultz, K.L. and Thomas, L.J., 2000. Management of worksharing systems. *Manufacturing & Service Operations Management*, 2 (1), 49-67.

McCreery, J.K. and Krajewski, L.J., 1999. Improving performance using workforce flexibility in an assembly environment with learning and forgetting effects. *International Journal of Production Research*, 37 (9), 2031-2058.

Molleman, E. and Van den Beukel, A., 2007. Worker flexibility and its perceived contribution to performance: the moderating role of task characteristics. *Human Factors and Ergonomics in Manufacturing*, 17 (2), 117-135.

Nembhard, D.A., ed., 2007. *Workforce cross training*. Boca Raton, FL: CRC Press, Taylor&Francis Group.

Pinker, E.J. and Shumsky, R.A., 2000. The efficiency-quality trade-off of cross-trained workers. *Manufacturing & Services Operations Management*, 2 (1), 32-48.

Rothkopf, M.H. and Rech, P., 1987. Perspectives on queues: Combining queues is not always beneficial. *Operations Research*, 35 (6), 906-909.

Sheikhzadeh, M., Benjaafar, S. and Gupta, D., 1998. Machine sharing in manufacturing systems: total flexibility versus chaining. *The International Journal of Flexible Manufacturing Systems*, 10, 351-378.

Slomp, J., Bokhorst, J.A.C. and Molleman, E., 2005. Cross-training in a cellular manufacturing environment. *Computers and Industrial Engineering*, 48 (3), 609-624.

Smith, D.R. and Whitt, W., 1981. Resource sharing for efficiency in traffic systems. *The Bell System Technical Journal*, 60 (1), 39-55.

Stidman, S., 1972. Regenerative Processes in the Theory of Queues, with Applications to the Alternating-Priority Queue. *Advances in Applied Probability*, 4 (3), 542-577.

Takagi, H., 1988. Queuing Analysis of Polling Models. *ACM Computing Surveys*, 20 (1), 5-10.

Treleven, M.D., 1989. A review of dual resource constrained system research. *IIE Transactions*, 21 (3), 279-287.

Van Dijk, N.M. and Van der Sluis, E., 2006. Check-in computation and optimization by simulation and IP in combination. *European Journal of Operational Research*, 171, 1152-1168.

Wisner, J.D. and Sifer, S.P., 1995. A survey of US manufacturing practices in make-to-order machine shops. *Production & Inventory Management Journal*, 36 (1), 1-7.

Whitt, W., 1999. Partitioning customers into service groups. *Management Science*, 45 (11), 1579-1592.

Yue, H., Slomp, J., Molleman, E. and Van Der Zee, D.J., 2008. Worker flexibility in a parallel dual resource constrained job shop. *International Journal of Production Research*, 46 (2), 451-467.

Zavadlav, E., McClain, J.O. and Thomas L.J., 1996. Self-buffering, self-balancing, self-flushing production lines. *Management Science,* 42 (8), 1151-1164.

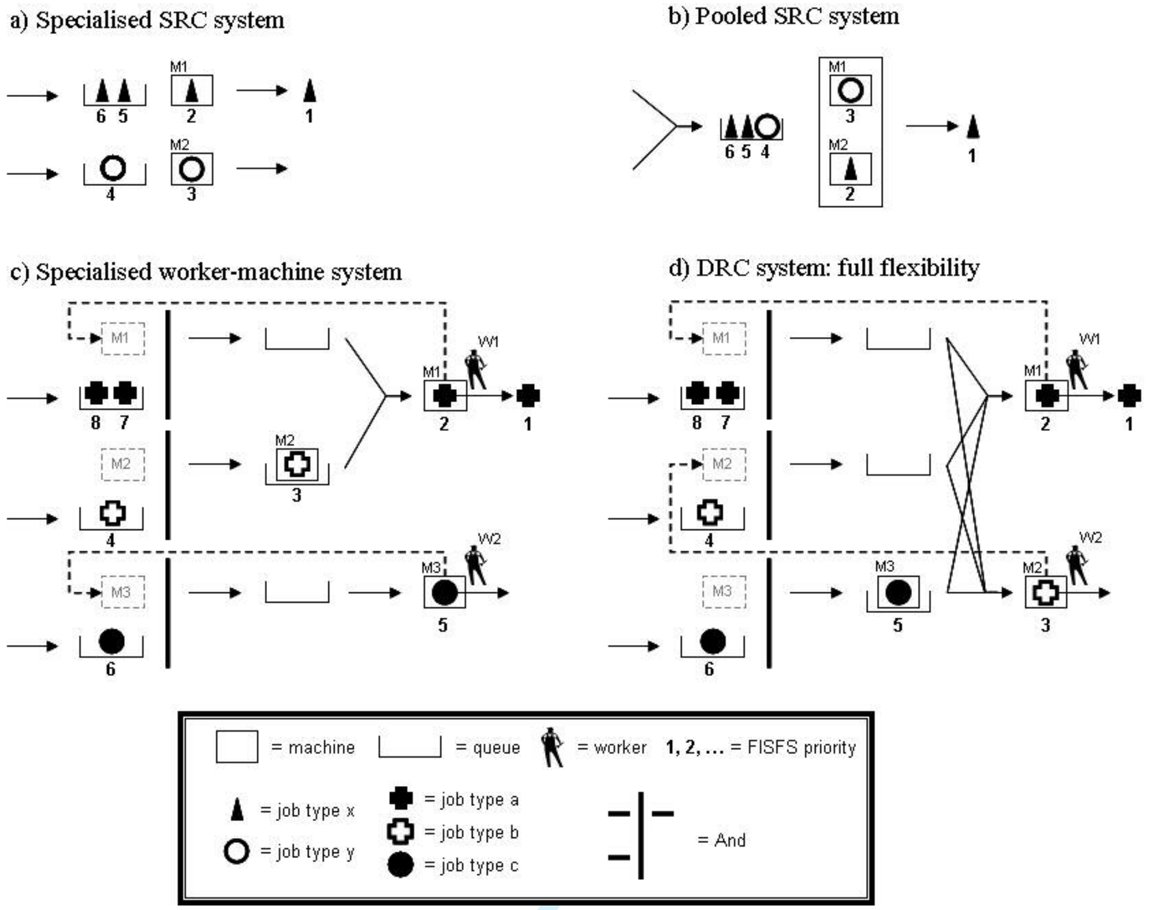


Figure 1. Comparing specialisation and pooling in SRC and DRC systems

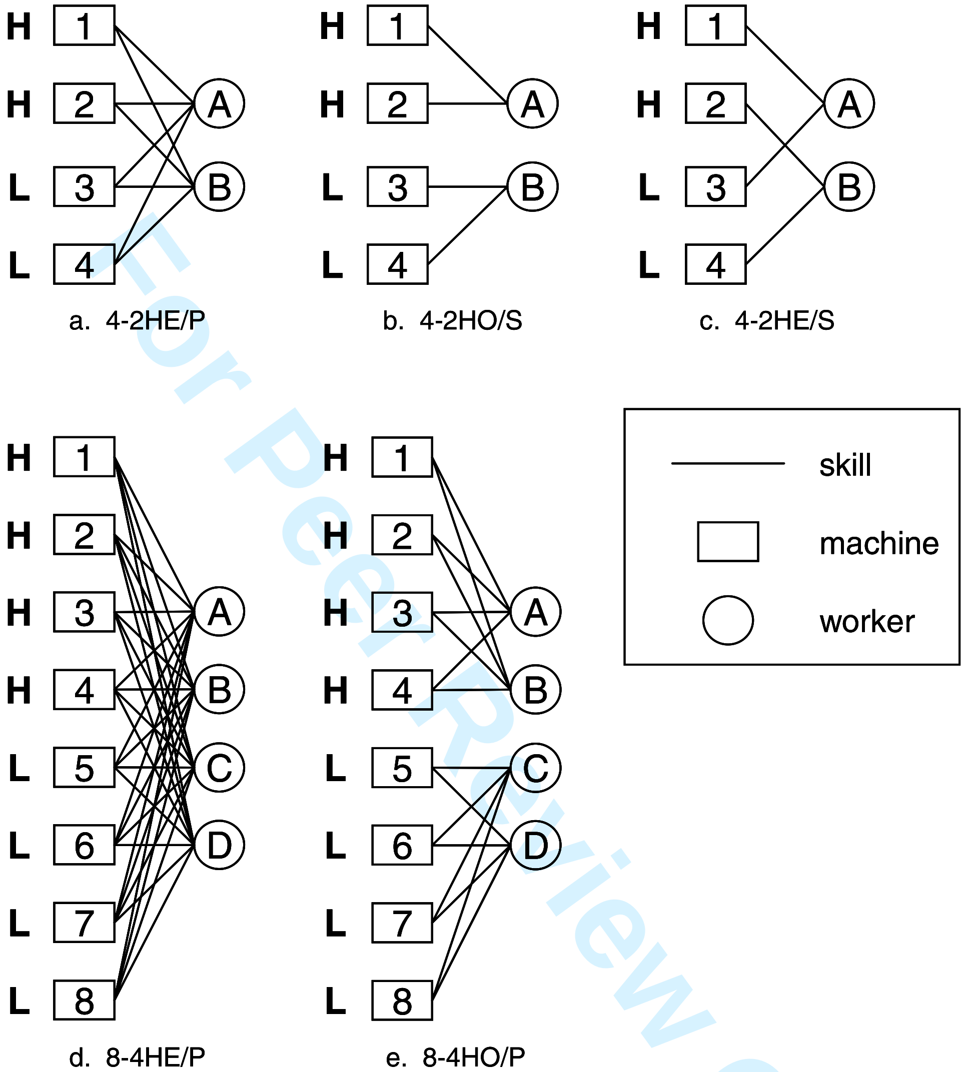


Figure 2. Cross-training configurations

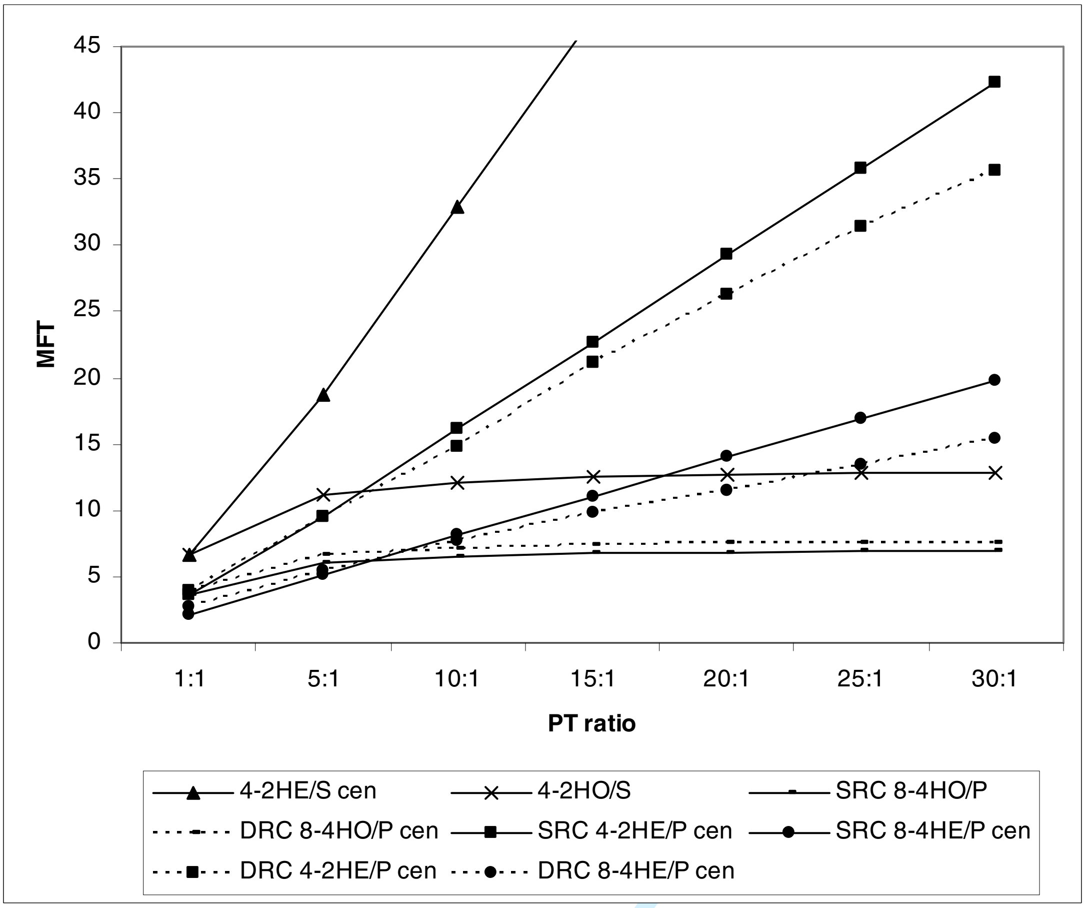


Figure 3. MFT results under a central when-rule

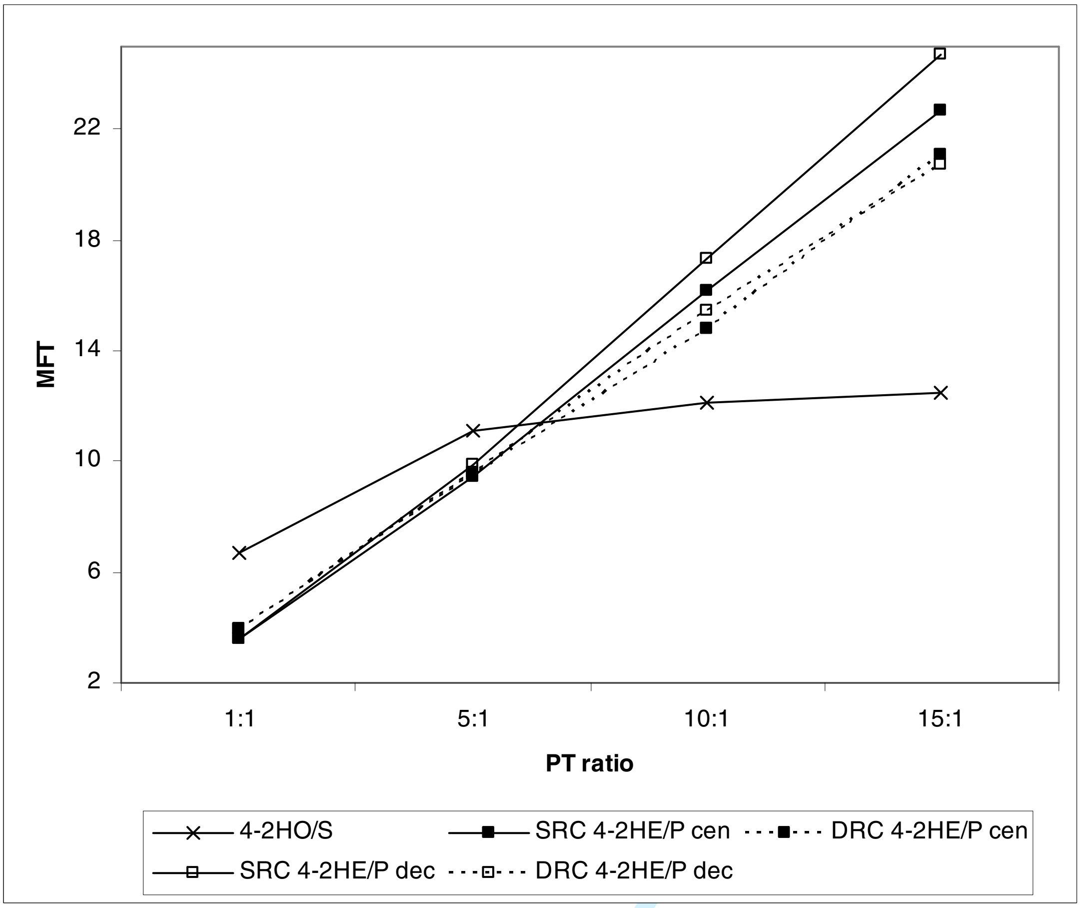


Figure 4. Crossing point of specialised and pooled configurations within the (4-2) system

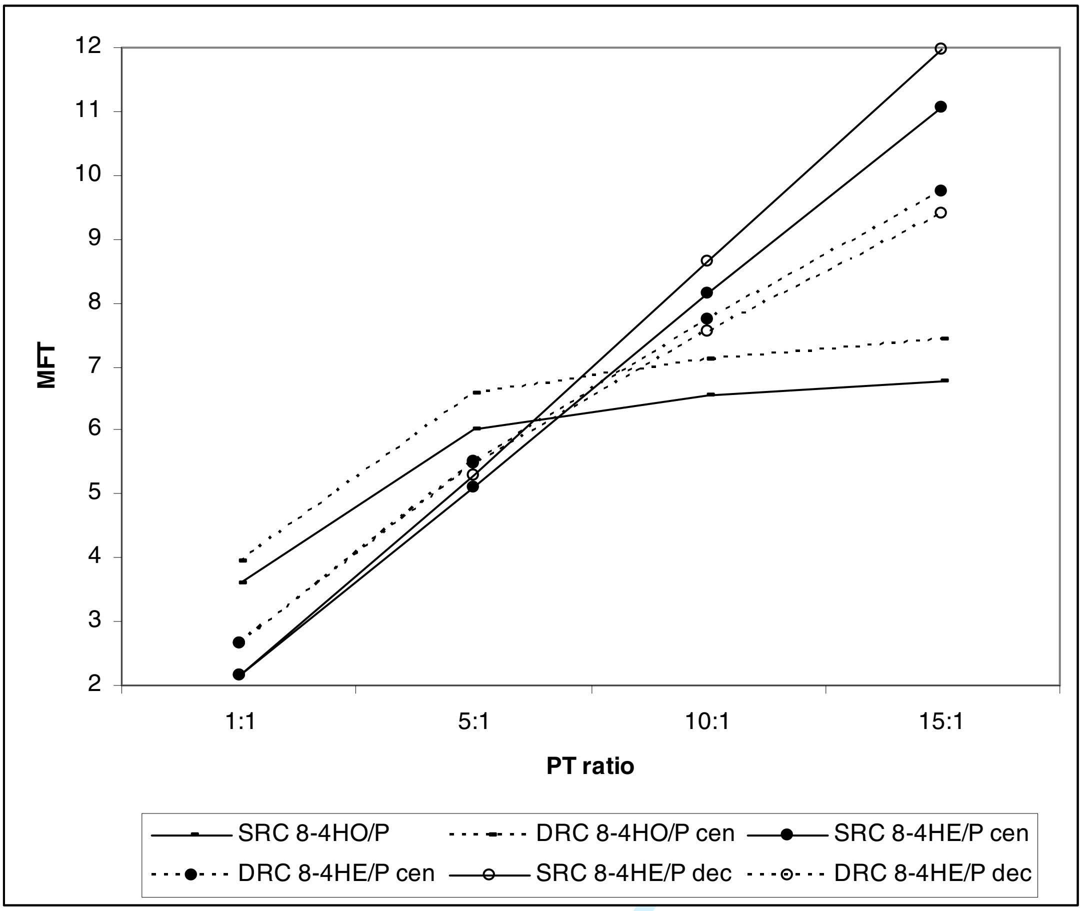


Figure 5. Crossing point of specialised and pooled configurations within the (8-4) system

Table 1. MFT results

		PT ratio						
		1:1	5:1	10:1	15:1	20:1	25:1	30:1
4-2HE/S	analytical							
	central when	6.67	18.67	32.98	47.21	61.40	75.59	89.77
	decentral when	6.67	19.61	35.59	51.54	67.48	83.42	99.35
4-2HO/S	analytical	6.67	11.11	12.12	12.5	12.70	12.82	12.90
4-2HE/P	SRC analytical							
	central when	3.60	9.48	16.14	22.70	29.24	35.77	42.29
	decentral when	3.60	9.91	17.33	24.69	32.03	39.37	46.70
	DRC simulation							
	central when	3.95	9.49	14.81	21.09	26.28	31.36	35.61
	decentral when	3.98	9.58	15.44	20.72	26.08	32.19	36.82
8-4HE/P	SRC analytical							
	central when	2.15	5.11	8.14	11.07	13.97	16.86	19.74
	decentral when	2.15	5.30	8.66	11.97	15.20	18.44	21.68
	DRC simulation							
	central when	2.67	5.51	7.74	9.74	11.49	13.51	15.38
	decentral when	2.67	5.47	7.55	9.41	11.08	12.91	14.95
8-4HO/P	SRC analytical	3.60	6.01	6.55	6.76	6.86	6.93	6.97
	DRC simulation							
	central when	3.94	6.57	7.11	7.43	7.51	7.60	7.52
	decentral when	3.97	6.63	7.27	7.42	7.61	7.68	7.68

Table 2. Mean flow time results for the $4\text{-}2HE/P_{SRC}$ and $8\text{-}4HE/P_{SRC}$ configurations

		PT ratio						
		1:1	5:1	10:1	15:1	20:1	25:1	30:1
4-2HE/P	SRC analytical							
	central when	3.60	9.48	16.14	22.70	29.24	35.77	42.29
	decentral when	3.60	9.91	17.33	24.69	32.03	39.37	46.70
	SRC simulation							
	central when	3.60	9.33	15.72	21.44	27.71	33.98	39.52
	decentral when-no	3.60	9.35	15.84	21.73	27.79	33.97	40.26
	decentral when-yes	3.60	9.18	15.22	21.42	26.91	33.13	38.20
8-4HE/P	SRC analytical							
	central when	2.15	5.11	8.14	11.07	13.97	16.86	19.74
	decentral when	2.15	5.30	8.66	11.97	15.20	18.44	21.68
	SRC simulation							
	central when	2.16	4.99	7.76	10.33	12.86	15.41	17.92
	decentral when-no	2.16	4.98	7.71	10.29	12.85	14.99	17.78
	decentral when-yes	2.15	4.81	7.30	9.62	11.84	13.84	16.36