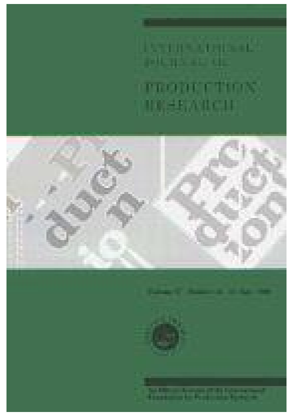
Comparing the impact of different rescheduling strategies on the entropic-related complexity of manufacturing systems

Luisa Delfa Huaccho Huatuco, Janet Smart, Ani Calinescu, Suja Sivadasan, Stella Kariuki



Comparing the impact of different rescheduling strategies on the entropic-related complexity of manufacturing systems

Journal:	*International Journal of Production Research*
Manuscript ID:	TPRS-2006-IJPR-0515.R2
Manuscript Type:	Original Manuscript
Date Submitted by the Author:	02-Nov-2007
Complete List of Authors:	Huaccho Huatuco, Luisa; University of Leeds, Business School Efstathiou, Janet; University of Oxford, Engineering Science Calinescu, Ani; University of Oxford, Computing Laboratory Sivadasan, Suja; University of Oxford, Engineering Science Kariuki, Stella; University of Oxford, Engineering Science
Keywords:	MANUFACTURING SYSTEMS, PERFORMANCE MEASURES
Keywords (user):	entropic-related-complexity, rescheduling

Comparing the impact of different rescheduling strategies on the entropic-related complexity of manufacturing systems

Luisa Huaccho Huatuco[1], Janet Efstathiou[2], Ani Calinescu[3], Suja Sivadasan[2] and Stella Kariuki[2]

[1]Corresponding author: Leeds University Business School, University of Leeds, Woodhouse Lane, LEEDS, LS2 9JT

[2]Department of Engineering Science, University of Oxford, Parks Road, OXFORD, OX1 3PJ

[3]Oxford University Computing Laboratory, University of Oxford, Wolfson Building, Parks Road, OXFORD, OX1 3QD

Comparing the impact of different rescheduling strategies on the entropic-related complexity of manufacturing systems

ABSTRACT

The primary objective of this paper is to compare five rescheduling strategies according to their effectiveness in reducing entropic-related complexity arising from machine breakdowns in manufacturing systems. Entropic-related complexity is the expected amount of information required to describe the state of the system. Previous case studies carried out by the authors have guided computer simulations, which were carried out in Arena 5.0 in combination with MS Excel. Simulation performance is measured by: 1) Entropic-related complexity measures, which quantify: a) the complexity associated with the information content of schedules, and b) the complexity associated with the variations between schedules; and 2) Mean flow time. The results highlight two main points: a) the importance of reducing unbalanced machine workloads by using the least utilised machine to process the jobs affected by machine breakdowns, and b) low disruption strategies are effective at reducing entropic-related complexity, this means that applying rescheduling strategies in order to manage complexity can be beneficial up to a point, which in low disruption strategies is included in their threshold conditions. The contribution of this paper is two-fold. First, it extends the application of entropic-related complexity to every schedule generated through rescheduling, whereas previous work only applied it to the original schedule. Second, recommendations are proposed to schedulers for improving their rescheduling practice in the face of machine breakdowns. Those recommendations vary according to the manufacturing organisations' product type and scheduling objectives. Further work includes: a) preparing a detailed workbook to measure entropic-related complexity at shop-floor level, and b) extending the analysis to other types of disturbances, such as customer changes.

Keywords: complexity; entropy; rescheduling strategies; manufacturing systems.

1. INTRODUCTION

1.1. Background

In an ideal manufacturing organisation, production works perfectly according to schedule. Thus, machines never break down, suppliers deliver the desired product on time and in full, and customers never change their orders. However, in real-world manufacturing organisations the situation described above infrequently occurs as production schedules become inaccurate or unfeasible due to disturbances, such as: customer changes, suppliers' failures to deliver and material problems. These disturbances cause the deviation of the actual production from that scheduled and consequently entropic-related complexity is increased. Entropic-related complexity is the expected amount of information required to describe the state of the system (Calinescu *et al.*, 2000). Such complexity if not properly managed could result in: customer dissatisfaction, losing discounts or preferred treatment by suppliers or less flexibility (Huaccho Huatuco, 2003). Manufacturing organisations find themselves vulnerable to these negative effects of complexity. However, some complexity is necessary to provide the flexibility and responsiveness to satisfy the customer. So, manufacturing organisations need to differentiate how much of that complexity is desirable, *i.e.* Value Adding (VA) and how much of it is non-desirable, *i.e.* Non-Value Adding (NVA) (Huaccho Huatuco *et al.,* 2001). This paper focuses on NVA entropic-related complexity due to machine breakdowns, analyses effectiveness of the rescheduling strategies in reducing entropic-related complexity. Estimating entropic-related complexity can help determine management effort required to overcome uncertainty (a characteristic of complexity) in projects (Bushuyev and Sochnev, 1999). Furthermore, it can help manufacturing organisations prioritise sources of complexity and identify mechanisms to manage or leverage it to their advantage (Sivadasan, 2001a).

The rest of this section is organised as indicated next. Section 1.2 reviews the rescheduling concept for manufacturing systems. Section 1.3 introduces the entropic-related complexity measures, whereas Section 1.4 presents the traditional measures that have been used in previous research and selects mean flow time for the work presented here. Finally, Section 1.5 presents a summary of the case study results that guided the computer simulations in this paper.

1.2. Rescheduling

Rescheduling is also referred in the literature as 'reactive scheduling' (Morton and Pentico, 1993), 'predictive-reactive scheduling' (Vieira *et al.*, 2003), 'real-time scheduling', or 'dynamic production planning' (Song, 2001). Rescheduling is as important as scheduling for the successful operation of manufacturing systems. Bean *et al.* (1991) define rescheduling as a dynamic approach that responds to disturbances, whereas Herrmann (2001) defines rescheduling as the process of updating an existing production schedule in response to those disturbances. In this paper, rescheduling is defined as changing the schedule in terms of time, quantity or product specifications in response to disturbances. A typical horizon for rescheduling is 1-3 days as specified by Morton and Pentico (1993).

Two main types of schedules are defined, as follows. First, the Original Schedule (*OS*), also known as 'first-pass schedule' (Jackson and Browne, 1989), 'initial schedule' (Jain and Elmaraghy, 1997) or 'predictive schedule' (Mehta and Uzsoy, 1998), results from the main production scheduling activity. Typically, it covers a horizon of 2 to 6 weeks (Morton and Pentico, 1993). Second, the Intermediate Schedule (*IS*) which results from rescheduling activities, *i.e.* the more disturbances affecting the *OS*, the more *IS*s are generated. The assumption is that the IS is produced by some reduced form of scheduling activity.

Previous research has focused on the study of different types of disturbances such as cancellation of orders (Jain and Elmaraghy, 1997; Li *et al.*, 1993) and shortage of materials (Li *et al.*, 1993). Another type of disturbance commonly studied in previous research is machine breakdowns (Beskow, 2001; Jain and Elmaraghy, 1997; Li *et al.*, 1993; Raheja and Subramaniam, 2002; Yamamoto and Nof, 1985). This is confirmed by Vieira *et al.* (2003) who report that machine breakdowns constitute the most common and important type of disturbances. Furthermore, machine breakdowns belong to the complexity type of main concern to manufacturing organisations, which is Non-Value Adding and Out of Control (Huaccho Huatuco *et al.*, 2001). Because of these reasons, the focus of the computer simulation experiments in this paper is on rescheduling due to machine breakdowns.

Due to the inherent complexity of the real-world production environment, the need for rescheduling is mandatory (Efstathiou, 1996; Jain and Elmaraghy, 1997, Koenig, 2002). However, previous research on rescheduling has left the following gaps. Firstly, it has measured the rescheduling performance only at the end of the scheduled period. This does not reflect on the dynamic nature of rescheduling with the generation of Intermediate Schedules. Secondly, it has focused mainly on material flows. This neglects the importance of information flows when a machine breakdown occurs in order to get the production back to normal. This paper fills those gaps by measuring entropic-related complexity on the intermediate schedules generated during the rescheduling period and by using an information-flow approach. In doing so, this paper quantifies the following: the amount of information the scheduler faces when rescheduling, the disruption that rescheduling causes to the production schedule as well as the mean flow time.

1.3. Entropic-related complexity

According to the Oxford Dictionary (Hornby, 2000), complexity is: "*The state of being difficult to understand*' or '*The state of being intricate, complicated or complex*".

Manufacturing organisations are dealing with increased complexity that comes from internal sources, such as: machine breakdowns, operator absenteeism; and external sources, such as: suppliers' failure to deliver on time and in full, and customer changes (Sivadasan *et al.,* 2002). Previous research has proposed some measures of complexity including: the different elements in interaction (Weisbuch, 1989), and the number of relationships between elements of the system (Barnes and McKay, 2000). However, in this paper we use entropic-related complexity measures because they provide a common platform on which different aspects of manufacturing operations can be compared (Bushuyev and Sochnev, 1999).

As stated earlier, entropic-related complexity is the expected amount of information required to describe the state of the system (Calinescu *et al.*, 2000). This definition is based on the work of Shannon (1948, 1949), who proposed entropy as a measure of uncertainty in communication channels. The use of Shannon's entropy was adapted to be applied to manufacturing systems by Frizelle (Frizelle and Woodcock, 1995) and Efstathiou (Efstathiou *et al.,* 1999).

Shannon's entropy is defined next. Given a set of n states $E = \{e_1, e_2, \ldots, e_n\}$ and their respective *a priori* probabilities of occurrence $P = \{p_1, p_2, \ldots, p_n\}$, where $p_i \geq 0$ and $\sum_{i=1}^{n} p_i = 1$, entropy (H) can be calculated using Equation (1) (Shannon, 1948, 1949).

$$H = -K \sum_{i=1}^{n} p_i \log_2 p_i \qquad (1)$$

In Equation (1), H is entropy, K is a positive constant (K = 1 throughout this paper), n is the number of states, p_i is the probability of occurrence of state i.

Entropy measures information rate and its units are bits per state. Some important properties of entropy (Calinescu, 2002; Shannon, 1948, 1949) are given next.

1. H is continuous in the p_i.
2. If all the p_i are equal, $p_i = \frac{1}{n}$, then H is maximum and equal to $\log_2 n$. So, any attempt to equalize of the probabilities leads to an increase in the entropy.
3. If the choice is broken down into two successive choices, the original H should be the weighted sum of the individual values of H. For example:

   $$H\left(\frac{1}{2},\frac{1}{3},\frac{1}{6}\right) = H\left(\frac{1}{2},\frac{1}{2}\right) + \frac{1}{2}H\left(\frac{2}{3},\frac{1}{3}\right)$$

   The above holds when there is dependency. However, when there is no dependency, the entropy associated with a number of information flows is the sum of the entropy associated with each independent flow (Sivadasan *et al.*, 2006).
4. H is zero if and only if all the p_i but one are zero, this one having the value of unity. So, only when we are certain of the outcome H disappears.
5. H is a concave function of p_i. This property ensures that a local maximum is also a global maximum.

The entropy measure above takes into account the number of states and the probability of occurrence of each state, which is related to the variety and uncertainty characteristics of complexity, respectively. Variety has been defined as the number of elements that can be distinguished in a set (Ashby, 1958). In the manufacturing context variety can correspond to number of products, number of routes, customers, suppliers, resources, and so on. Uncertainty refers to the inability to accurately define the current state of the system or predict its future state. Uncertainty leads to excess inventory of raw materials, immediate goods and final products that the manufacturing organisation has to maintain as buffer stocks in order to cope with disturbances.

Entropic-related complexity measures have been used previously in order to assess manufacturing organisations problems, such as: the operational complexity of supplier-customer systems (Sivadasan, 2001a; Sivadasan *et al.*, 2002; Sivadasan *et al.*, 2004; Sivadasan *et al.*, 2006), production control (Karp and Ronen, 1992), the risk of managing projects (Bushuyev and Sochnev, 1999), manufacturing flexibility (Deshmukh *et al.*, 1992, 1998; Kumar, 1987; Yao, 1985; Yao and Pei, 1990), product variety (Frenken *et al.,* 1999), and decision making complexity (Calinescu *et al.*, 2001, Calinescu, 2002).

The entropic-related complexity measures applied in this paper are useful to schedulers because they quantify the following:

- *The complexity associated with the information content of schedules***.** This measure determines how much information the scheduler is handling when rescheduling.
- *The complexity associated with the variation between schedules***.** It focuses on the disruptive effects of disturbances on the current production schedule.

The complexity measures presented above will be further explained in Section 3.3.

1.4. Traditional measures

Several performance measures have been commonly used in previous research to compare different rescheduling strategies and dispatching rules, some of them include:

- Mean tardiness (Beskow, 2001; Gindy and Saad, 1998, Jain and Elmaraghy, 1997), overall tardiness (Efstathiou, 1996) or due-date-based objective (Holthaus, 1999; Zhang and Chen, 1999).

- Mean flow time (Bierwith and Mattfeld, 1999; Efstathiou, 1996, Jain and Elmaraghy, 1997; O'Donovan *et al.*, 1999), maximum flow time (Henry and Kafura, 1981).
- Average machine utilisation (Jain and Elmaraghy, 1997; Yao and Pei, 1990; Zhang and Chen, 1999).
- Throughput or productivity (Chiu and Yih, 1995).

In this paper, mean flow time is chosen because it is the traditional measure most commonly used in previous research. Furthermore, it is used in this paper to provide a clear picture of the effects of rescheduling. Mean flow time is the average amount of time that jobs stay in the production system (Morton and Pentico, 1993). It is important that mean flow time is minimised in order to convert working capital into revenue for the manufacturing company as quickly as possible. The mean flow time calculation as used in this paper is given in Section 3.4.

1.5. Case study

This section provides the context for the computer simulations and for validating the findings. The case study carried out by the authors at a manufacturing organisation (here onwards referred as "the company") motivated the computer simulation experiments presented in Section 2. The company is a multinational corporation that produces plastic bottles for the Fast Moving Consumer Goods (FMCG) sector. It integrated with one of its key customers that operated a JIT system, where responsiveness and flexibility are crucial. Details on the results of this case study are given in Huaccho Huatuco (2003).

This section presents results on the situation after integration, based on live and historical data of changes to the production schedules on a daily basis. The following analysis focuses on the effects of rescheduling on the complexity handled by the company.

The *complexity associated with the information content of schedules* became important to the company because after integration there was more pressure on its scheduling function than before. However, there was no quantitative evidence to prove it. Given that the scheduling task at the company was performed by one scheduler, the calculation of the *complexity associated with the information content of schedules* was used to determine the average amount of information that the scheduler can cope with as compared with previous findings by Miller (1956).

The results of *the complexity associated with the variation between schedules* suggest that rescheduling causes less disruption to the current schedule as the production week goes by. This is not surprising as there are fewer days to consider and therefore less number of jobs to reschedule in the event of machine breakdowns.

The reminder of the paper is organised as follows. Section 2 describes the rescheduling strategies. Section 3 outlines the computer simulations. Section 4 presents the results of the comparison of rescheduling strategies. Finally, Section 5 discusses these results, presents some conclusions and provides further work directions.

2. RESCHEDULING STRATEGIES

Rescheduling strategies are also referred to in the literature as rescheduling 'techniques', 'strategies', 'heuristics', 'algorithms', 'methods' or 'rules'. A rescheduling strategy specifies how and when rescheduling is performed. It also indicates the events and methods used to revise the current schedule (Herrmann, 2001).

The rescheduling strategies used in this paper are: **Priority High strategy** (priority-based strategy with high disruption algorithm), **Priority Low strategy** (priority-based strategy with low disruption algorithm), **Utilisation High strategy** (machine utilisation-based strategy with high disruption algorithm), **Utilisation Low strategy** (machine utilisation-based strategy with low disruption algorithm) and **Right-shift strategy** (delaying jobs). These five rescheduling strategies were derived from literature review, especially from Jain and Elmaraghy (1997), and from the authors' case study experience, which analysed a wider range of disturbances such as customer changes and supplier's failure to deliver on time and in full (Huaccho Huatuco, 2003). However, in Jain and Elmaraghy (1997) the focus was only on traditional performance measures, such as mean tardiness. This paper focuses on entropic-related complexity and mean flow time.

The five rescheduling strategies resulted from the combination of two elements: rescheduling criteria and types of algorithm used for rescheduling production, as discussed next:

1. Rescheduling criteria: The *affected job* refers to the job that is interrupted by the occurrence of a machine breakdown, whereas *remaining time* refers to the affected job's processing time that remains to be completed.

   - *Job priority* (Jain and Elmaraghy, 1997; Smith, 2002; Yamamoto and Nof, 1985): This criterion refers to the case where there are alternative machines to which the *affected job* could be moved, depending on job priorities (*current priority* > *lowest alternative priority*). If the priority of the *affected job* is higher than the one of the job being processed on the alternative machine, then the latter is interrupted and the *affected job* is split, and *remaining time* starts being processed on the alternative machine. In this paper, priorities are defined as indicated by Zhang and Chen (1999) so that the jobs with earlier due dates get higher priorities. After the processing of the *remaining time* is completed, the interrupted job's remaining route processing time is processed on the same alternative machine.
   - *Machine utilisation* (Jain and Elmaraghy, 1997): This criterion refers to the case where there are alternative machines to which the affected job could be moved

depending on machine utilisation. When a machine breakdown affects a job, the alternative machines are checked for availability (*available idle time* > *remaining time*). If there is more than one alternative machine then the machine with the least utilisation is selected. This may cause the delay of the rest of jobs in the chosen alternative machine.

- *Right-shift* (Beskow, 2001; Cheng, 1998; Holthaus, 1999; Yamamoto and Nof, 1985): It can be also called the 'do-nothing' case where no corrective action is taken. Therefore, all the affected jobs run late on the broken machine until the machine breakdown is over.

2. Types of algorithm:

   a. *High disruption* (Yamamoto and Nof, 1985): when the scheduling function reacts as soon as a machine breakdown occurs. In this algorithm the scheduling function always looks for alternative machines, regardless of the downtime and the remaining processing time of the affected job.

   b. *Low disruption* (Smith, 2002): when the scheduling function takes into account a threshold of tolerance before carrying out the rescheduling tasks. In this paper, the scheduling function looks for alternative machines only when the estimated downtime is greater than the remaining processing time of the affected job (*downtime* > *remaining time*).

Combining the first two rescheduling criteria and the algorithm type generates four 'reactive rescheduling strategies' (1 to 4 below), which together with the right-shift strategy give a total of five rescheduling strategies.

The rescheduling strategies are explained next by referring to the Original Schedule, *OS* given in Figure 1, part (a). The shaded areas in Figure 1 represent downtime.

<Figure 1 here>

1. **Priority High strategy**: Job priority with high disruption, see Figure 1, part (b). When there is a machine breakdown that affects any of the scheduled jobs (job 1 on machine M1), the algorithm searches for alternative machines. If the considered alternative machine is busy, but the priority of the affected job is higher than that of the job being processed on the alternative machine (as in this case), then the remaining processing time of the affected job is moved to the alternative machine (machine M2 and right-shifts all the rest of jobs in M2). If the alternative machine is idle, then the jobs are moved straightaway. Otherwise, the *remaining* processing time of the affected job waits until the broken machine becomes operational, and right-shifts the jobs on the broken machine.
2. **Priority Low strategy**: Job priority with low disruption, see Figure 1, part (c). When there is a machine breakdown that affects any of the scheduled jobs (job 1 on machine M1), the algorithm searches for alternative machines as described for the Priority High strategy, but only if the *downtime* > *remaining time* condition is fulfilled (in Figure 1, part (c) this is not the case). Otherwise, the remaining processing time waits until the broken machine becomes operational, and right-shifts the jobs on the broken machine (*remaining time* of job 1 in machine M1).
3. **Utilisation High strategy**: Machine utilisation with high disruption. See Figure 1, part (d). When there is a machine breakdown that affects any of the scheduled jobs (job 1 on machine M1), the algorithm looks for the alternative machines for allocating the *remaining* processing time of the affected job. Then, the algorithm chooses the alternative machine with the least utilisation (in Figure 1, machine M3 is the least utilised). The job is moved to the alternative machine only if there is enough idle space to accommodate the remaining time (job 1 on machine M3). If there are two machines with the same least

utilisation, the machine that is physically closer to the broken machine is chosen. If the distance is the same for both alternatives then the machine is decided randomly. If there are no alternative machines at all, or if the idle time in the alternative machine is insufficient to accommodate the *remaining* processing time of the affected job, then the remaining time of the affected job is not moved to an alternative machine, but waits until the broken machine becomes operational again.

4. **Utilisation Low strategy**: Machine utilisation with low disruption. See Figure 1, part (e). When there is a machine breakdown that affects any of the scheduled jobs (job 1 on machine M1), the algorithm searches for alternative machines as described for the Utilisation High strategy, but only if the condition *downtime* > *remaining time* is fulfilled (in Figure 1, part (e) this is not the case). Otherwise, the *remaining* processing time starts on the same machine when it becomes operational again (job1 on machine M1).
5. **Right-shift strategy**: When there is a machine breakdown, the arrival time at which the processing of the job is resumed is when the machine becomes operational again. This is illustrated in Figure 1, part (f).

3. COMPUTER SIMULATIONS

Based on the observations of the production shop-floor during the case study presented in Section 1.5, the analysis in this paper concentrates on a parallel machine scheduling problem with the following characteristics: 5 loaded machines (some of which are alternative machines according to the job type), 24 scheduled jobs, 0% spare capacity, 4 machine breakdowns, a simulation length (scheduled period) of 60 time units and a total of 130 replications (26 experiments x 5 random arrivals of machine breakdowns) subject to 5 rescheduling strategies. The 26 experiments result from the 3 sets of experiments considered in this paper: *Processing*

Times (Set 1), *Number of Jobs (Set 2)* and *Number of Loaded Machines (Set 3)*, with 7, 9 and 10 experiments, respectively. The corresponding *OS* is presented in Figure 2.

<Figure 2 here>

The assumptions made for the computer simulations are:

- the original schedule (*OS*) is provided,
- jobs have been scheduled so that they meet their due date,
- the job arrivals, processing and departures are deterministic,
- the machine breakdowns arrivals are stochastic,
- the variations are considered between two consecutive schedules, e.g. between *OS* and IS_1 is represented as IS_1-*OS*,
- rescheduling takes place subject to constraints, such as capacity and availability of resources,
- a machine processes one job at a time,
- machines are interchangeable,
- jobs arrive during the scheduled period,
- set-up times are included in processing times, and
- transportation times are negligible.

The base experiment represents the state of the machine as always being "Busy", *i.e.* the machine utilisation is 100%, or equivalently the spare capacity level is 0%. The rest of experiments within each set are generated from the base experiment by increasing the spare capacity, as explained below. It is worth noting that throughout the experiments the jobs start at the original scheduled time.

Processing Times (Set 1): By decreasing the jobs' processing times starting from a base experiment with 0% spare capacity and decreasing gradually the processing time of all jobs by 10% from one experiment to the next until a spare capacity of 60% was reached. See Table 1.

<Table 1 here>

Number of Jobs (Set 2): By decreasing the number of jobs, starting from a base experiment with 24 scheduled jobs and gradually decreasing them until a spare capacity of around 60% was reached, this gave as a result a reduction from 24 to 9 jobs in the *OS*'s for each experiment. See Table 2 and Table 3.

<Table 2 here>

<Table 3 here>

Number of Loaded Machines (Set 3): By decreasing the number of loaded machines (which is the same as increasing the number of idle machines), starting from a base experiment with 5 loaded machines and gradually decreasing them until a spare capacity of 60% was reached, resulting in the number of loaded machines being reduced from 5 to 2. See Table 4.

<Table 4 here>

The experiments consisted of the eight steps noted below.

1. Take the original schedule (*OS*) with the embedded spare capacity according to the definition of the experiment. For example: 10% spare capacity for Experiment 2 in Processing Times (Set 1).
2. The shop floor produces according to the production schedule.
3. A machine breakdown occurs.
4. If the machine breakdown affects any of the scheduled jobs, then the corresponding rescheduling strategy is applied, e.g. Priority High strategy. Next go to step 5. Otherwise, if it is not the end of the scheduled period, then go back to step 2 else go to step 7.
5. A new schedule is generated with the new arrangement of jobs after machine breakdowns.

6. Repeat steps 2 to 5 until the scheduled period is completed.
7. Generate the output schedule and calculate entropic-related complexity measures.
8. Collect, plot and analyse the data.

Next, Section 3.1 briefly presents the input data. Section 3.2 presents the output schedules, and Section 3.3 outlines the complexity formulae and some examples. Finally, Section 3.4 outlines the mean flow time measure.

3.1. Input data

The input data used in this paper is based on the case study presented in Section 1.5, but focussing only on machine breakdowns disturbing the production schedules.

1. Production Schedule: It contains the following data on the jobs to be processed: (a) Arrival time; (b) Machine number; (c) Part number; (d) Job number; (e) Quantity and (f) Departure time.
2. Breakdown data: Describes the occurrence of a machine breakdown: (a) Breakdown arrival time to be randomly distributed in the following intervals: [0,12], [13,25], [26,38], [39,50]; (b) Broken machine number; (c) Downtime.
3. Alternative machines: Lists the machines that can take over the job in case it is interrupted by a machine breakdown: (a) Part number; (b) Alternative machine.
4. State of machines: Contains the data on the current condition of the machines regarding the processing of jobs: (a) State: it describes the current condition of the machine, such as: "Busy" (when the machine is making a product), "Idle" (when the machine is not broken-down and it is not making a product) or "Broken-down" (when the machine is broken-down); (b) State start; (c) State end; (d) Machine; (e) Job number.

5. Machine utilisation: This item of input data is only applicable to the utilisation strategies, *i.e.* Utilisation High and Utilisation Low strategies. (a) Machine; (b) Utilisation: It is calculated using Equation (2), which was derived from Morton (1999):

$$U_i = \frac{b_i}{sp_i} \times 100 \qquad (2)$$

In Equation (2), U_i is the utilisation of machine i; b_i is the time that machine i is scheduled to be busy and sp_i is the length of the scheduled period for machine i.

6. Job priorities: This item of input data is only applicable to the job priority strategies, *i.e.* Priority High and Priority Low strategies. This is only calculated at the beginning of the scheduled period. (a) Part number; (b) Priority: Based on the job's due date. The earlier the due date, the higher the priority. It is calculated using the formula given in Equation (3).

$$Y_i = 100 - (dd_i - EDD) \qquad (3)$$

In Equation (3), $i = \overline{1,n}$; Y_i is the priority of job i; dd_i is the due date of job i and EDD is the earliest due date in the set of scheduled jobs.

The Production Schedule represents the basis for the derivation of the initial values of the states of machines, machine utilisation and job priorities. The number of data items changes depending on the rescheduling strategy chosen. It is worth noting that the methods could be adjusted to include setup and transfer times by adding the corresponding fields to the input data files.

3.2. Output schedules

The output schedules resulting from the experiments include the original schedule (*OS*) and as many Intermediate Schedules (*IS*) generated through rescheduling as are needed to cope with the occurrence of machine breakdowns.

3.3. Entropic-related complexity measures

The complexity measures presented in Section 1.3 are explained in further detail here. The entropic-related measures are used in this paper to quantify the following: a) the complexity associated with the information content of the schedules, and b) the disruption to previous schedules when rescheduling.

The entropic-related complexity associated with the information content of schedules and that associated with the variation between schedules were calculated using Equation (4).

$$H_R = -\sum_{i=1}^{M}\sum_{j=1}^{S} p_{ij} \log_2 p_{ij} \tag{4}$$

In Equation (4), H_R is a rescheduling-related entropic measure of complexity, M is the number of machines, S is the number of states, and p_{ij} is the probability of machine i being in state j.

Taking the data from the *OS* and the *IS*, some states (S) of interest are defined. These states are defined according to the meaning of the information content to the scheduler or person in charge of managing the deviations from the production schedule. A more extensive discussion on state definitions is given in Sivadasan *et al.* (2001b). Then, each occurrence of an event that belongs to each state is counted. After that, the probabilities (p_{ij}) are calculated taking into account the overall number of occurrences across all states. Next, these probabilities are used for calculating the entropic-related complexity measures applying Equation (4).

The entropic-units of these measures are bits per state. Take for example, one machine showing three states: “Busy” for 80% of the time, “Idle” for 15% of the time and “Broken down” for 5% of the time. In this example, applying the formula above this machine shows a complexity of 0.88 bits per state compared to the maximum entropy of $\log_2 3$= 1.58 bits per state (property 2 in Section 1.3).

a. *The complexity associated with the information content of schedules.* The generated schedules during rescheduling were taken into account, namely: OS and IS_ks, and the following machine states were defined for its calculation: “Busy”, “Idle” and “Broken-down”.

b. *The complexity associated with the variation between schedules.* The difference in production times between consecutive schedules was taken into account, giving as a result the following variations: IS_k-IS_{k-1} and IS_1-OS, where k is an index. The following states were defined for its calculation: “Scheduled” (when the job is being processed according to the schedule), “Busy, wrong job” (when a different job from the schedule is being processed) and “Broken-down” (when a machine is broken down).

3.4. Mean flow time

As stated in Section 1.4, the mean flow time is the average amount of time that jobs stay in the production system (Morton and Pentico, 1993). So, the mean flow time is related to Work in Progress (WIP), lead time and responsiveness of the manufacturing system. Mean flow time was calculated using Equation (5), which was derived from (Morton, 1999) and it is given in time units.

$$F = \frac{1}{n}\sum_{j=1}^{n}\left(c_j - a_j\right) \tag{5}$$

In Equation (5), F is the mean flow time; c_j is the departure time of job j; a_j is the arrival time of job j and n is the number of scheduled jobs.

This section has presented the computer simulations context in terms of sets of experiments, input and output definitions, as well as the performance measures to be assessed. Next section presents the results of those performance measures.

4. RESULTS

This section presents the summary of the results for each rescheduling strategy, which consists of the average values over 130 replications.

1. Entropic-related complexity

a. The complexity associated with the information content of schedules

The method of investigation here consists of measuring the amount of information in each of the schedules across all five rescheduling strategies per set of experiments. An example of the calculation of this measure is given in Table 5 for the *OS* given in Figure 1(a).

<Table 5 here>

The values of this measure were calculated using Equation (4) in the following manner. The states of the machines were monitored during the execution of the schedule. If the machine is making a product then the state was "Busy", if it was broken then the state is "Broken-down" and if it was inactive then the state was "Idle".

The results of this measure are presented in Table 7 (measure 1a). It is worth noting that the maximum value of this measure (3.062 bps) is slightly less than 3.25 bits per state, which is the lower limit for one-dimensional variables that one person can handle according to Miller (1956). This means that the scheduler dealing with this problem can cope with it. However, it is clear that this problem has been simplified so that the scheduler is only taking care of a single disturbance *i.e.* machine breakdowns, rather than the whole range of disturbances that could possibly occur in real life.

< Table 7 here>

In Table 7 (measure 1a), the rescheduling strategy that is the best in reducing this measure is Right-shift and the worst is Priority High. From the four reactive rescheduling strategies, low disruption strategies perform better than high disruption ones. This is because low disruption strategies use a condition to decide whether or not apply rescheduling – which prevents unnecessary disruptions to the *OS*.

b. The complexity associated with the variation between schedules

The method of investigation here consists of comparing the disruption of previously generated schedules (*OS* or *IS*) across all five rescheduling strategies per set of experiments. An example of the calculation of this measure is given in Table 6 corresponding to Figure 1(d).

<Table 6 here>

This measure was calculated using Equation (4) in the following manner. The most recent previous and the current schedule were compared at regular points in time and their variations (IS_1-OS, IS_2-IS_1-IS_3-IS_2, IS_4-IS_3) were recorded. If the job in the current schedule is the same as that of the previous schedule, then this observation corresponded to the "Scheduled" state. If there was a machine breakdown affecting either of the schedules, then that observation was considered in the "Broken down" state. Finally, if a job different that the one stated in the previous schedule was being processed in the current schedule, then this observation was assigned to the "Busy, wrong job" state.

From Table 7 (measure 1b), The Utilisation High strategy is the best at decreasing this measure across all five strategies, whereas the Priority High strategy is the worst. In general, Utilisation strategies are better at reducing this measure than the Priority strategies. The reason is that the condition in Utilisation strategies (*available idle time* > *remaining time*) is less permissive than those of the Priority strategies (*current priority* > *lowest alternative priority*). Out of the four reactive rescheduling strategies, the Utilisation High and Utilisation Low strategies are effective at reducing this measure.

2. Mean flow time

The results of this measure are presented in Table 7 (measure 2). It can be observed that the Right-shift strategy exhibits the worst performance, whereas Priority Low and Priority High show the best and second best, respectively. This is explained by the fact that the Right-shift strategy causes the delay of affected jobs, which makes them stay on the shop floor for longer than the other strategies.

5. DISCUSSION, CONCLUSIONS AND FURTHER WORK

Rescheduling strategies have been studied from an information flow perspective. These rescheduling strategies were compared primarily according to their capacity to reduce the entropic-related complexity and secondarily taking into account the effects on mean flow time. In order to provide a full picture of performance to the scheduler, the five rescheduling strategies are compared in Table 7, as follows: the best (✓✓), second best (✓), neutral (-), second worst (×) and the worst (××).

Based on Table 7, some recommendations are proposed to schedulers for improving their rescheduling practice under machine breakdowns conditions according to the type of objective the manufacturing organisation wants to achieve. Assuming that companies want to satisfy their customers and they would not choose the Right-shift strategy, in order not to risk losing their custom, then:

- If the manufacturing organisation's objective is to minimise *the complexity associated with the information content of schedules*, then the best strategy to choose is the Utilisation Low strategy. The implication of this to the manufacturing organisation is that it would need to invest in additional machines. From the results, Low disruption strategies perform better at reducing this measure than High disruption ones.
- If the manufacturing organisation's objective is to minimise *the complexity associated with the variation between schedules*, *i.e.* the disruption from the previous schedule, then

the best strategy is the Utilisation High strategy, whereas the second best strategy is Utilisation Low.

From the results above it can be inferred that Utilisation strategies are more effective at reducing entropic-related complexity due to machine breakdowns than Priority strategies. Looking at the mean flow time performance, Utilisation High performs slightly better than Utilisation Low.

The ultimate benefit to manufacturing organisations is to save money that is otherwise unnecessarily spent because of poor rescheduling practices, and to reduce stress on the schedulers. The results presented in this section highlight the importance of avoiding unbalanced workloads by using the least utilised machine to process the jobs affected by machine breakdowns.

It is worth mentioning that small differences in the absolute numerical values of the performance measures obtained for different rescheduling strategies are relatively significant. For example, in Table 7 (measure 1.a), the *complexity associated with the information content of schedules* shows a value of 3.000 bps whereas the Utilisation Low strategy shows a value of 3.001 bps. The ranking presented in this paper is based on these results, i.e. the Right-shift rescheduling strategy performs better than Utilisation Low rescheduling strategy at reducing this measure. In this paper the computer simulations were set to reproduce the case study characteristics described in Section 1.5. The data from real-world case studies are based on finite schedules (typical schedule horizon: 2-6 weeks). By contrast, computer simulations which are not based on case studies could assume longer, even infinite, schedule horizons. The relationship between the ranking and the experiments is that the ranking depends on the number of different replications taken into account in the computer simulation experiments, which number is in this paper of 130 replications. However, one of the main contributions of this paper is the study of the impact of different rescheduling strategies on entropic-related complexity measures for specific, rather than general types of manufacturing systems. In this connection, the findings are relevant to organisations that belong to the "*functional product, responsive scheduling*" quadrant (Huaccho Huatuco, 2003), where the key point is to be responsive to customer requests by adapting the current schedule.

The limitations of the research presented in this paper are three-fold. First, the results are only applicable to manufacturing organisations that use any of the specific rescheduling strategies presented here. They have not been tested to provide enough statistical validity for producing more general and definite conclusions across different manufacturing sectors. Second, this paper proposes recommendations for schedulers about their rescheduling practice. The decision to implement these recommendations is left to the management of manufacturing organisations. Third, this paper has dealt with two key approaches to managing complexity: spare capacity (embedded in the production schedules) and decision making (used in the rescheduling strategies algorithms), there are other approaches which include: production flexibility, stock and computer systems, which are outside the scope of this paper.

There are two main areas suggested for further work. First, in order to transfer the knowledge to industry it would be necessary to make the tools available for shop floor use. This can be done in the first instance in a pilot case by providing an education pack with the theory, a set of workbooks that explain step-by-step the calculation of the entropic-related complexity measures and a software suite that assists the scheduler in assessing the advantages and disadvantages of different rescheduling strategies. Second, the work presented in this paper could be extended to investigate the impact of rescheduling due to external disturbances, such as customer changes on the entropic-related complexity of manufacturing systems.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge Engineering and Physical Sciences Research Council (EPSRC) support including grants GR/M52458 and 98316556, and Overseas Research Scholarship (ORS) for award 1999032139. The authors would also like to acknowledge Gerry Frizelle and his research team at the Institute for Manufacturing at the University of Cambridge for their support with the development of the computer simulations. The authors thank the referees for their feedback comments that contributed to the improvement of the paper.

REFERENCES

Ashby, W. R., 1958. Requisite variety and its implications for the control of complex systems. *Cybernetica (Namur)*, 1(2), 83-99.

Barnes, C. J. and McKay, A., 2000. Towards support for complexity in mechanical products and processes. *In*: I. P. McCarthy and T. Rakotobe-Joel, eds. *Proceedings of the Conference on Complexity and Complex Systems in Industry.* Warwick: The University of Warwick, 121-133.

Bean, J. C., Birge, J. R., Mittenthal, J. and Noo, C. E., 1991. Matchup scheduling with multiple resources, release dates and disruptions. *Operations Research,* 39(3), 470-483.

Beskow, G. J., 2001. *Rescheduling of airline pilot training activities following disruptions*. MSc. Thesis. Virginia Polytechnic Institute and State University.

Bierwith, C. and Mattfeld, D. C., 1999. Production scheduling and rescheduling with genetic algorithms. *Evolutionary computation*, 7(1), 1-17.

Bushuyev, S. D. and Sochnev, S. V., 1999. Entropy measurement as a project control tool, *International Journal of Project Management*, 17(6), 343-350.

Calinescu, A., 2002. *Manufacturing complexity: an integrative information-theoretic approach*. Thesis (D.Phil.). University of Oxford.

Calinescu, A., Efstathiou, J., Sivadasan, S., Schirn, J. and Huaccho Huatuco, L., 2000. Complexity in Manufacturing: An Information Theoretic Approach. *In*: I. P. McCarthy and T. Rakotobe-Joel, eds. *Proceedings of the Conference on Complexity and Complex Systems in Industry.* Warwick: The University of Warwick, 30-44.

Calinescu, A., Efstathiou, J., Sivadasan, S. and Huaccho Huatuco, L., 2001. Information-theoretic measures for decision-making in manufacturing. *Proceedings of the 5th World*

Multi-Conference on Systemics, Cybernetics and Informatics (SCI 2001). Orlando: International Institute of Informatics and Systemics, X, 73-78.

Cheng, Y., 1998. Hybrid simulation for resolving resource conflicts in train traffic rescheduling. *Computers in Industry*, 35, 233-246.

Chiu, C. and Yih, Y., 1995. A learning-based methodology for dynamic scheduling in distributed manufacturing systems. *International Journal of Production Research*, 33(11), 3217-3232.

Deshmukh, A. V., Talavage, J. J. and Barash, M. M., 1992. Characteristics of part mix complexity measure for manufacturing systems. *Proceedings of the IEEE International Conference in Systems, Man and Cybernetics*. New York, 2, 1384-1389.

Deshmukh, A. V., Talavage, J. J. and Barash, M. M., 1998. Complexity in manufacturing systems. Part 1: Analysis of static complexity. *IIE Transactions*, 30, 645-655.

Efstathiou, J., 1996. Anytime heuristic schedule repair in manufacturing industry. *IEE Proceedings Control Theory Applications*, 143**,** 114-124.

Efstathiou, J., Tassano, F., Sivadasan, S., Shirazi, R., Alves, J., Frizelle, G. and Calinescu, A., 1999. Information complexity as a driver of emergent phenomena in the business community. *Proceedings of the International Workshop on Emergent Synthesis*. Japan: Kobe University, 1-6.

Frenken, K., Saviotti, P. P. and Tommetter, M., 1999. Variety and niche creation in aircraft, helicopters, motorcycles and microcomputers. *Research Strategy,* 28(5), 469-488.

Frizelle, G. and Woodcock, E., 1995. Measuring complexity as an aid to developing operational strategy. *International Journal of Operations and Production Management,* 15(5), 26-39.

Gindy, N. N. and Saad, S. M., 1998. Flexibility and responsiveness of machining environments. *Integrated Manufacturing Systems*, 9(4)**,** 218-227.

Henry, S. and Kafura, D., 1981. Software structure metrics based on information flow. *IEEE Transactions in Software Engineering*, SE-7(5), 510-518

Herrmann, J. W., 2001. *Improving manufacturing system performance through rescheduling* [online]. Maryland, University of Maryland. Available from: http://www.isr.umd.edu/~jwh2/papers/rescheduling.html [Accessed 18 July 2007].

Holthaus, O., 1999. Scheduling in job shops with machine breakdowns: an experimental study. *Computers & Industrial Engineering,* 36, 137-162.

Hornby, A. S. 2000. *Oxford Advanced Learner's Dictionary of Current English.* 6th ed. S. Wehmeier, ed. Oxford: Oxford University Press.

Huaccho Huatuco, L. D., 2003. *The role of rescheduling in managing manufacturing systems' complexity*. Thesis (D. Phil.), University of Oxford.

Huaccho Huatuco, L., Efstathiou, J., Sivadasan, S. and Calinescu, A., 2001. The value of dynamic complexity in manufacturing systems. *Proceedings of the International Conference of the Production and Operations Management Society. (POMS - Brazil 2001)*, 180-188.

Jackson, S. and Browne, J., 1989. An interactive scheduler for production activity control. *Computer Integrated Manufacturing,* 2(1), 2-14.

Jain, A. K. and Elmaraghy, H. A., 1997. Production scheduling/rescheduling in flexible manufacturing. *International Journal of Production Research*, 35(1)**,** 281-289.

Karp, A. and Ronen, B., 1992. Improving shop floor control: An entropy model approach. *International Journal of Production Research*, 30(4)**,** 923-938.

Koenig S., 2002. Greedy On-line Planning. *Tutorial of the 6th International conference on Artificial Intelligence Planning and Scheduling (AIPS'02)*, American Association for Artificial Intelligence/European Association of Excellence in AI Planning (AAAI/PLANET).

Kumar, V., 1987. Entropic measures of manufacturing flexibility. *International Journal of Production Research*, 25(7)**,** 957-966.

Li, R. K., Shyu, Y. T. and Adiga, S., 1993. A heuristic rescheduling algorithm for computer-based production scheduling systems. *International Journal of Production Research,* 31(8)**,** 1815-1826.

Mehta, S. V. and Uzsoy, R. M., 1998. Predictable scheduling of a job shop subject to breakdowns. *IEEE Transactions on Robotics and Automation,* 14, 365-378.

Miller, G. A., 1956. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review,* 63, 81-97.

Morton, T. E., 1999. *Production Operations Management*. International Thomson Publishing.

Morton, T. E. and Pentico, D. W., 1993. *Heuristic Scheduling Systems: with applications to production systems and project management*. Wiley Series in Engineering and Technology Management. John Wiley and Sons.

O'Donovan, R., Uzsoy, R. and McKay, K., 1999. Predictable scheduling of a single machine with breakdowns and sensitive jobs. *International Journal of Production Research,* 37(18), 4217-4233.

Raheja, A. S. and Subramaniam, V., 2002. Reactive schedule repair of job shops [online]. Massachussets, MIT. Available from: http://hdl.handle.net/1721.1/4038 [Accessed on 18 July 2007].

Shannon, C. E., 1948. A mathematical theory of communication, *Bell System Technical Journal,* 27, 379-423, 623-656.

Shannon, C. E., 1949. The mathematical theory of communication. *In*: C. E. Shannon and W. Weaver, eds. *The mathematical theory of communication*. Illinois: University of Illinois Press, 3-91.

Sivadasan, S., 2001a. *Operational Complexity of Supplier-Customer Systems.* Thesis (D.Phil.). University of Oxford.

Sivadasan, S., Efstathiou, J., Calinescu, A. and Huaccho Huatuco, L., 2001b. A discussion of the issues of state definition in the entropy-based measure of operational complexity across supplier-customer systems. *Proceedings of the 5th World Multi-Conference on Systemics, Cybernetics and Informatics (SCI 2001).* Orlando: International Institute of Informatics and Systemics, II, 227-232.

Sivadasan, S., Efstathiou, J., Frizelle, G., Shirazi, R. and Calinescu, A., 2002. An information-theoretic methodology for measuring the operational complexity of supplier-customer systems. *The International Journal of Operations and Production Management*, 22(1), 80-202.

Sivadasan, S., Efstathiou, J., Calinescu, A. and Huaccho Huatuco, L., 2004. Supply Chain Complexity. *In*: S. New and R. Westbrook, eds. *Understanding Supply Chains*. Oxford: Oxford University Press, 133-163.

Sivadasan, S., Efstathiou, J., Calinescu, A. and Huaccho Huatuco, L., 2006. Advances on Measuring the Operational Complexity of Supplier-Customer Systems. *European Journal of Operational Research,* 171, 208-226.

Smith, S. F., 2002. Technologies for dynamic scheduling. *Workshop on On-line Planning and Scheduling of the 6th International Conference on Artificial Intelligence Planning and*

Scheduling (AIPS'02). American Association for Artificial Intelligence/European Association of Excellence in AI Planning (AAAI/PLANET).

Song, D., 2001. Stochastic Models in Planning Complex Engineer-To-Order Products. Thesis (PhD). University of Newcastle upon Tyne.

Vieira, G. E., Herrmann, J. W. and Lin, E., 2003. Rescheduling manufacturing systems: a framework of strategies, strategies and methods, *Journal of Scheduling*, 6(1), 35-58.

Weisbuch, G., 1989. *Complex systems dynamics: an introduction to automata networks.* Wokingham: Addison-Wesley Advanced Book Program.

Yamamoto, M. and Nof, Y., 1985. Scheduling/rescheduling in the manufacturing operating system environment. *International Journal of Production Research*, 23(4)**,** 705-722.

Yao, D. D., 1985. Material and information flows in flexible manufacturing systems. *Material flow*, 2, 143-149.

Yao, D. D. and Pei, F. F., 1990. Flexible parts routing in manufacturing systems. *IIE Transactions*, 22, 48-55.

Zhang, Y. and Chen, H., 1999. A knowledge-based dynamic job-scheduling in low-volume/high-variety manufacturing. *Artificial Intelligence in Engineering*, 13, 241-249.

Table 1: Processing Times (Set 1) experiments

Experiment	Decrease in processing times from base experiment	Resulting spare capacity
1	0%	0%
2	10%	10%
3	20%	20%
4	30%	30%
5	40%	40%
6	50%	50%
7	60%	60%

Table 2: Number of jobs (Set 2) experiments for Priority High strategy

Experiment	Number of jobs taken out from base experiment	Resulting number of jobs	Resulting spare capacity
1	0	24	0%
2	2	22	10%
3	4	20	17%
4	6	18	25%
5	8	16	32%
6	10	14	38%
7	12	12	47%
8	14	10	53%
9	15	9	60%

Table 3: Jobs that were taken out for each experiment in Number of Jobs (Set 2)

Job	Experiment								
	1	2	3	4	5	6	7	8	9
14		x	x						
62		x	x	x					
43			x						
37			x	x					
40				x					
38				x					
39				x					
60				x					
11					x	x	x	x	x
13					x	x	x	x	x
41					x	x	x	x	x
51					x	x	x	x	x
61					x	x	x	x	x
12					x	x	x	x	x
21					x	x	x	x	x
42					x	x	x	x	x
34						x	x	x	x
22						x	x	x	x
35							x	x	x
31							x	x	x
33								x	x
53								x	x
32									x

Where "x" represents that the job in the table row has been taken out from the experiment in the table column.

Table 4: Number of Loaded Machines (Set 3) experiments

Experiment	**Number of machine loads taken out**	**From machine number**	**Resulting number of loaded machines**	**Resulting spare capacity**
1	0	Not Applicable	5	0%
2	1	1	4	20%
3	1	2	4	20%
4	1	3	4	20%
5	2	1,3	3	40%
6	2	1,2	3	40%
7	2	1,4	3	40%
8	3	1,2,3	2	60%
9	3	1,3,4	2	60%
10	3	1,2,4	2	60%

Table 5: Calculation example for the complexity associated with the information content of schedules

OS in Figure 1(a)

Occurrences

Machine	Busy	Idle	Broken down
1	60	0	0
2	60	0	0
3	60	0	0
Total			180

Probabilities

Machine	Busy	Idle	Broken down
1	0.333	0	0
2	0.333	0	0
3	0.333	0	0
Total			1

Entropic-related complexity

Machine	Busy	Idle	Broken down
1	0.528	0	0
2	0.528	0	0
3	0.528	0	0
Total			1.585

Table 6: Calculation example for the Complexity associated with the variation between schedules

IS1-OS (Utilisation High strategy) in Figure 1(d)

Occurrences

Machine	Scheduled state	Busy, wrong job	Broken-down
1	50	0	10
2	60	0	0
3	30	30	0
Total			180

Probabilities

Machine	Busy	Idle	Broken down
1	0.278	0	0.056
2	0.333	0	0
3	0.167	0.167	0
Total			1

Entropic-related complexity

Machine	Busy	Idle	Broken down
1	0.513	0	0.232
2	0.528	0	0
3	0.431	0.431	0
Total			2.135

Table 7: Performance measure vs. rescheduling strategy (average over 130 replications)

Performance measure	Rescheduling strategy				
	Priority High	**Priority Low**	**Utilisation High**	**Utilisation Low**	**Right-shift**
Entropic-related					
1a. The complexity associated with the information content of schedules	×× 3.062	× 3.028	- 3.013	✓ 3.001	✓✓ 3.000
1b. The complexity associated with the variation between schedules	×× 2.731	× 2.722	✓✓ 2.709	- 2.713	✓ 2.712
Traditional					
2. Mean flow time	✓ 13.678	✓✓ 13.650	- 13.802	× 13.992	×× 14.002

Key: ✓✓: this strategy is the best; ✓: this strategy is the second best; -: this strategy is neutral; ×: this strategy is the second worst; ××: this strategy is the worst.

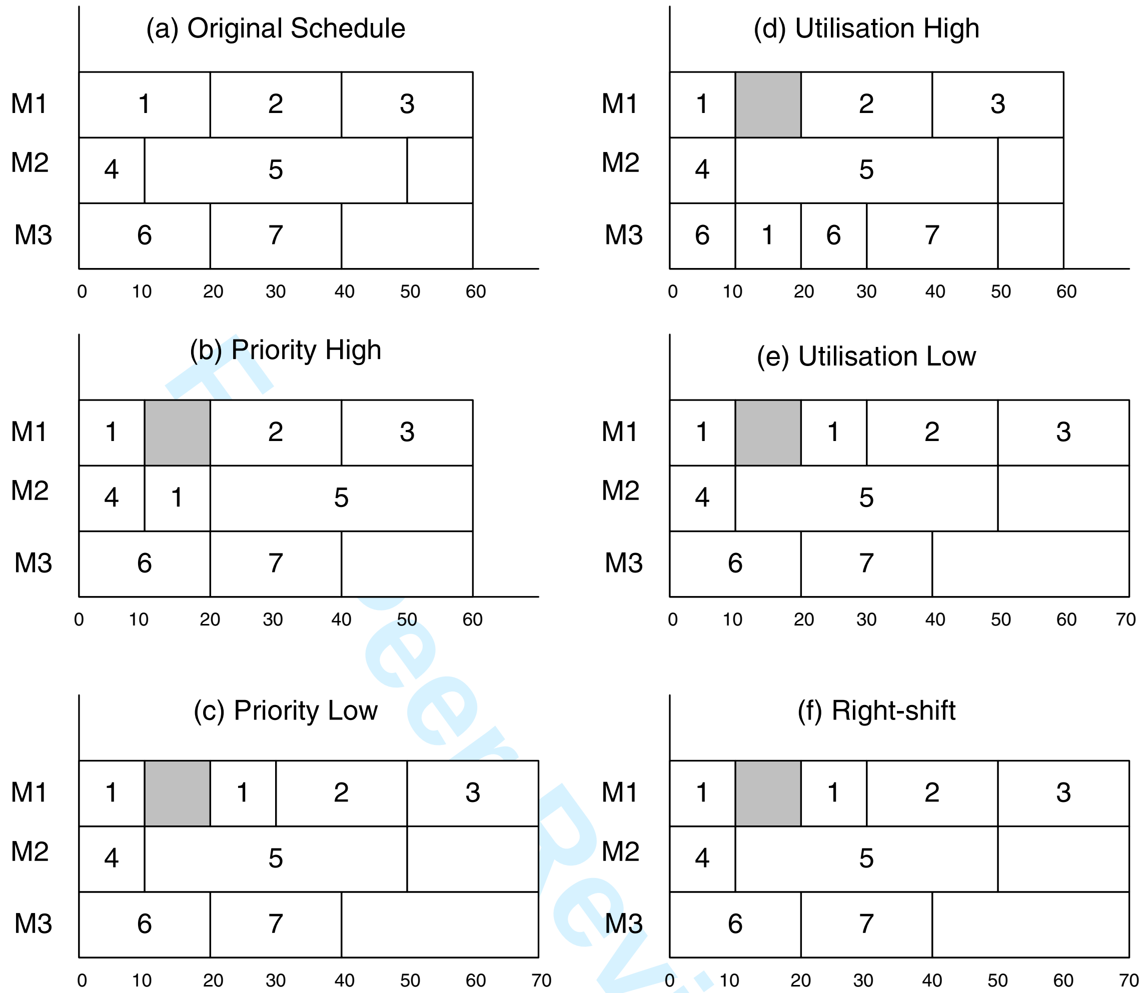


Figure 1: Examples of rescheduling strategies operation

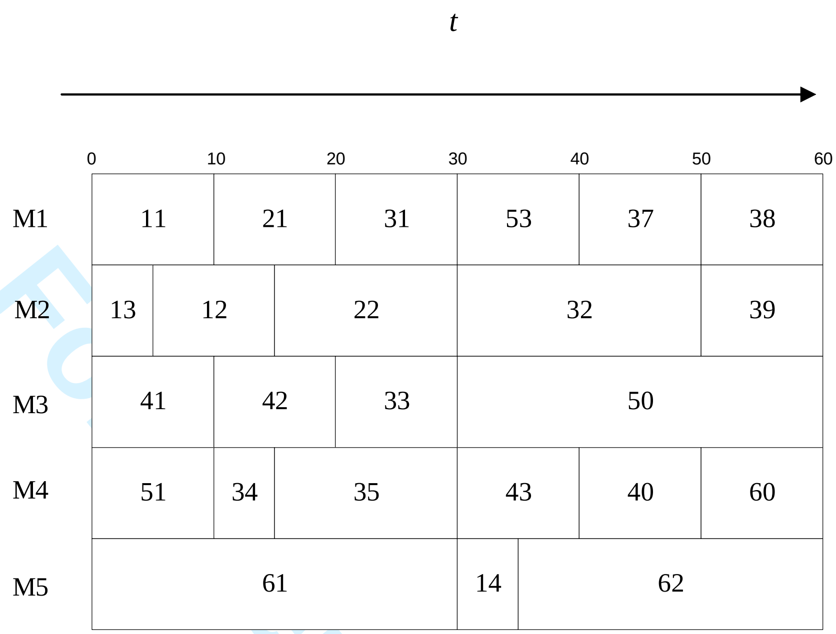


Figure 2: Original Schedule