Journal of the Operations Research
Society of Japan
2006, Vol. 49, No. 3, 152-173

AVERSION DYNAMICS - ADAPTIVE PRODUCTION CONTROL HEURISTICS INCORPORATING RISK

Kenneth N. McKay
University of Waterloo

Gary W. Black
University of Southern Indiana

(Received June 9, 2005; Revised January 4, 2006)

Abstract

There has been a well discussed gap between theory and practice in the area of planning and scheduling of dynamic situations such as open job shops. In this paper a number of heuristics and concepts for how to address part of the gap are summarized and discussed. The heuristics and algorithms focus on dynamic and adaptive risk control related to work assignment and work sequencing. A theme connecting the heuristics is that of Aversion Dynamics – the desire to schedule and plan to avoid or minimize perceived problems. We will present the summary of research results on Aversion Dynamics followed by suggestions for further research focusing on risk management.

Keywords: Scheduling, OR practice, aversion dynamics, risk mitigation

1. Introduction

Since the early 1900's, the use of mathematics to help guide production control in factories has been advocated. Early uses included looking at ratios associated with operation times to decide when to start operations within a job [18], to the famous economic order quantity [15], to the use of statistics for quality control [34] and test sampling [32]. The use of algorithms and models has now embraced almost every facet of manufacturing. In some production control areas the use of mathematics has been very successful. For example, line balancing, supply chain management at the aggregate level, scheduling and planning of assembly lines, process oriented factories, and large machine problems such as steel mills. Unfortunately, there has not been similar success in the planning and control of open, dynamic job shops. The gap between theory and practice in this area has been discussed in many papers [9, 13, 14, 16, 19, 26, 30, 33, 36]. In this paper we will discuss a research agenda conducted since the early 1990's on part of the gap: how to include some of the real world dynamics in mathematical algorithms for production control.

There are many issues relating to the perceived gap between theory and practice. Pounds [31] noted the gap developing in the early 1960s and noted how researchers were solving a problem that did not appear to exist in the real world. The seminal work by Conway, Maxwell, and Miller [12] focused on sequencing and placed a limit on the practical nature of the work:

> "*The solution is to extract a problem that ignores the possibility of such changes and considers only the questions of sequence. Such a problem is unrealistic in the sense that it does not exactly represent any individual real situation, but rather gains in generality, since it approximates many situations. The results obtained from the study of this abstract idealized model do not represent a solution to any real sequencing problem; they represent*

> *information that should be available along with judgment and data on other aspects of the real problem." (pp. 2–3)*

During the period 1988-98, Conway and Maxwell repeatedly restated this limitation and have encouraged researchers to investigate the practice of scheduling [11].

It is likely that no single research effort will bridge the gap in its entirety or address all of the issues. The gap is very large and it will probably require the work of many researchers to create the mathematics and systems to address all of the issues. The area we have focused on since the late 1980's has been the area of risk created by operational uncertainty on the factory floor, and the relevance of this to the formulation of the job shop scheduling problem.

One of the earliest definitions of planning and scheduling in a factory was the following:

> *"The schedule man must necessarily be thorough, because inaccurate and misleading information is much worse than useless. It seems trite to make that statement but experience makes it seem wise to restate it. He must have imaginative powers to enable him to interpret his charts and foresee trouble. He must have aggressiveness and initiative and perseverance, so that he will get the reasons underlying conditions which point to future difficulties and bring the matter to the attention of the Department Head or Heads involved and keep after them until they take the necessary action. He is in effect required to see to it that future troubles are discounted" [10].*

Of specific interest in this definition is the sentence pertaining to foreseeing future troubles and discounting them. The view of trouble can be possibly widely defined – for example, trouble with meeting order due dates, trouble having the right person on the right machine at the right time, trouble with yield, trouble with obsolete inventory, and trouble associated with the actual operations caused by material, machine state, or other factors. This theme underlies the concepts behind the line of research called *Aversion Dynamics.* In Aversion Dynamics, observations from empirical studies have been used to enhance and create algorithms and concepts for i) identifying perceived risks, and ii) mitigating the possible impacts associated with the risks. The intent behind Aversion Dynamics is to satisfy and match the objective stated by Coburn, and focuses on special logic or ideas that try to avoid or discount future troubles. The research on Aversion Dynamics has been a collaborative effort involving a number of researchers working at various times with Kenneth McKay: Thomas Morton, Gary Black, John Hollywood, Reha Uzsoy, Ronan O'Donovan, and Smitha Varghese. The Aversion Dynamics research has thus far addressed i) a control framework within which adaptive heuristics can execute, ii) adaptive scheduling engine design, iii) dispatch heuristics, and iv) special batch insertion.

The following sections provide a brief literature review of the traditional approaches to production control heuristics followed by sections describing the various Aversion Dynamics research results. The paper concludes with a discussion about research opportunities and challenges for how the gap between theory and practice can be possibly further reduced with Aversion Dynamics.

2. Literature Review - Traditional Approaches

In the traditional literature on production control, uncertainty and risk are often assumed to be totally random and independent of other problem parameters [3]. In this section, the general view of production control and uncertainty will be discussed. In Section 4, specific literature will be noted for each research area that is addressed.

Typical information in a scheduling algorithm might include quantities, processing time per piece, due date, release date, relationships between operations, penalties for being late or early, setup times, and machine requirements [16, 30]. In inventory, there are the demands, leadtimes, bills of material, transportation and distribution parameters, and production times [35]. Depending on the complexity of the model, randomness is introduced to demands, processing time, setup time, yield, setups, and arrival time. Mean time between failure and mean time to repair might also be introduced. All of these bits of information are necessary for production control, but are they sufficient for applied or realistic production control? If not operationally, at least close enough for insights and guidance? The assumption of basic necessity and sufficiency is implicit or explicit in the traditional research. While valid in some situations, how valid is the assumption when the situation rapidly changes and changes in a significant fashion?

All of the factors included in the traditional models are usually assumed to be independent of each other. That is, the mean time to fail is not dependent upon the type of work processed in the last five operations. Uncertainties associated by these "assumed to be independent and random factors" are usually addressed by starting all work earlier (if possible), adding some safety stock to the equation, adding additional capacity in the form of shifts or new equipment, or specifying slack between operations. The countermeasures are also usually independent of state and manufacturing situation. Safety stock might be dependent on the specific item being produced, but the safety stock will not be dynamically altered based on who the operator of the machine might be, the previous type of work performed on the machine, and so forth. All of the little nuances not addressed are assumed to be random and accounted for by the rough approximation leading to the countermeasure. This approach is fine for truly random perturbations such as machine failure caused by a lightning strike or when the impact of any single perturbation is not significant relative to the bigger picture. There is also not much that can be done for unexpected power outages that affect the whole plant for extended periods of time caused by an accident elsewhere.

This assumption of true randomness being the main cause of variance, or at least that part of variance that should be included in the mathematical formulations, is problematic based on the empirical studies noted in this paper. Not everything that causes variance is truly random. In fact, in many cases the initial event or trigger of variance (the unwanted perturbation) and/or what happens next (the impact associated with the perturbation) can be reasonably predicted. For example, consider a factory that usually operates just-in-time and normally backward loads work from the due date. If you have not made a part recently, not for the last six months, and you have a substantial order due the last day of this month, with a job processing time of two days, what do you do? Do you wait till two days before the end of the month and release the job into the factory? This is what traditional formulations and commercial scheduling software would likely recommend. This is not what a scheduler such as Ralph [22] would do. Ralph was a scheduler who was studied for six months. Ralph would have a number of heuristics to choose from. For example, the work could be started earlier - early enough to allow time to replenish material and restart the job. Alternatively, the water could be tested first - start a small batch of the work earlier to make sure that everything was going to be ok, and then do the full batch at the end of the month. There are other variants that could also be considered. In any event, Ralph would not simply schedule the work two days before the due date. In the documented cases, Ralph theorized that a rather large number of variables might have changed since the last time the part was made and this constituted a risk - *a risk high enough to justify countermeasures.* This is an example of the line of thought behind Aversion Dynamics. Ralph and some other schedulers studied scan the horizon for potential troubles and take countermeasures. In the study centered around Ralph, approximately 10% of the decisions made daily by Ralph pertained to future risks and countermeasures. These decisions were the focal point of decisions, anchored the overall plan, and in effect decomposed the problem space [24]. There are also schedulers who do not scan the horizon, assume everything is random, survive via reacting, and blindly go forward, without thinking about the context or situation. In McKay [22], it was shown that if the situation warranted non-mechanical decision making and it was absent, the situation was high-cost, ineffective, and chaotic.

We have observed mechanistic scheduling caused by several factors. First, the scheduler did not know better, was new to the task (and possibly to the plant), and was using the first-order data without recognizing anything subtle. They honestly did not know any better and could not be expected to do any better. Second, possibly well-meaning management ordered the scheduler to create sequences using hard and fast rules without using any context information that the scheduler actually knew and knew how to use. Third, no one followed the plan, or others changed the plan and the scheduler was depressed and disillusioned - gave up trying to craft better sequences because they were never considered anyway. In all three cases, the output sequence is the same. Based on quantity, due date, machine requirements, machine availability and perhaps several other pieces of data from the MRP or manufacturing database, the sequence is created. This is the same type of output sequence generated by traditional production control methods. There is no context, no anticipation, and no situational knowledge. There is no attempt to foresee problems beyond those associated with simply sequencing, and no attempt to reduce operational risks. Similar issues exist with inventory theory.

The following section briefly describes the history behind Aversion Dynamics and the concepts behind the adaptive and dynamic control it attempts to provide. The focus of the research is the non-mechanistic type of decision that traditional methods do not address.

3. Aversion Dynamics

A series of short case studies on the scheduling task [20] identified a pattern of scheduling behavior that focused on uncertainty and the actions of the schedulers relating to the uncertainty [21]. It was observed that the schedulers spent a large portion of their time on anticipating future problems and crafting special sequences of work assignments to deal with the future problems. This focus appeared to dominate the schedulers' task. Two longitudinal studies then concentrated on this aspect [22]. In this research, non-routine or non-mechanical decisions were specifically studied and analyzed for threat detection and threat response. A mechanical decision was considered to be one that used a heuristic or decision rule blindly, based on typical manufacturing data such as due date, earliest release date, processing time, setup time, and basic resource capability. That is, a decision that a computer could be considered capable of making. A non-mechanical decision was one that included enriched or contextual information pertaining to the work or work situation. These decisions required some kind of human judgment.

Upon reflection and analysis of the data collected through the field studies, a number of heuristics were identified that specifically dealt with risk and risk avoidance - risk being associated with any perceived future trouble. Some of the heuristics related to operational procedures - how to function under periods of rapid change and perceived risks. Other heuristics focused on specific sequences of work that appeared to mitigate the risks. And yet other heuristics related to batching strategies - creating special demand for work that tested for the existence of risk. All of the heuristics shared certain attributes. The heuristics were not active all of the time. The heuristics were triggered by the detection of perceived threat triggers. The decision making adapted to the situation by activating the special logic. By triggering the special logic, the decision making further adapted work flow instructions (what to do when) to the situation - based on the context. The adaptation was not static in the heuristics and would either slowly return to the normal state through a decay process similar to heat loss, or would trigger a discrete change in the logic and return to the normal state. All of the Aversion Dynamic concepts share these traits. Furthermore, all of the heuristics used enriched information or knowledge not normally included in the traditional formulations for production control. The remaining theme that appeared to exist in the various heuristics gathered from the real factories was the existence of implicit utility theories and tradeoffs being made. For example, a special decision was rarely described in solitary terms and was usually made in the context of other possible decisions and why one decision was better than another.

In the early 1990's, these observations were shared with Thomas Morton and the dynamics and utility theory concepts discussed. For the initial research, the aversion ideas were married with Bottleneck Dynamics [27] and new mathematical formulations explored. The concept of Bottleneck Dynamics shared many of the same structural aspects of the empirically derived heuristics, and was considered a reasonable starting point. It was at this time that the term ***Aversion Dynamics*** was created to describe the general effect of risk aversion and trouble avoidance [20, 22].

The Aversion Dynamics research addresses the main traits identified above. At the highest level, a two-stage control theoretic framework was created which was inspired by the hierarchical production planning (HPP) work of Anthony [1, 2]. The Agile Hierarchical Production Planning (A-HPP) model [22, 23] provides the overall framework for exploring how the concepts would work at the tactical and operational levels. A-HPP was subsequently used as the base for an adaptive scheduling engine design [17].

In the case of dispatch heuristics, the formulations developed include the ability to identify or predict time of impact, types of impact, and the recovery from impact. This affects the actual formulation and the creation of hybrid, two-stage heuristics. A number of research activities were performed on dispatch heuristics to explore different types of risk situations and sequencing decisions [4–7, 25, 28, 29]. Special work insertion and batch control was another category of heuristics observed in the studies and this has had preliminary research as well [8, 37]. The following section provides brief overviews of these research efforts.

4. Aversion Dynamics Research

This section summaries six areas of research relating to the concept of Aversion Dynamics. The research has attempted to address the overall control structure within which the concept would execute, tactical and operational adaptation at the heuristic and algorithm level, and adaptive logic for sequencing and batching.

4.1. Risk control structure

The hierarchical production planning (*HPP*) paradigm proposed by Anthony [1] in which information flows downward through structural levels has been used extensively throughout the manufacturing industry. The hierarchical decomposition concepts upon which HPP was based were targeted for stable industries in which the mass production features of standardization, specialization and stabilization can be exploited. However, field studies by McKay [22] have shown that the HPP framework, in its original form, is not well suited for modern rapidly changing industries such as electronics manufacturing, computers, biotech and composites. For instance, in printed circuit board production, the placement machines, processes and production line layouts frequently change due to evolving technologies, changing demand patterns and frequent material/personnel changes. Under such conditions, it is difficult to achieve the state of equilibrium upon which HPP is based. Accordingly, McKay [22] and McKay, et al. [23] proposed an Agile Hierarchical Production Planning (*A-HPP*) paradigm for such industries. The following paragraphs will discuss A-HPP in relation to HPP.

The HPP paradigm presumes that higher levels of decision making will make certain assumptions, aggregate the information and then constrain the lower levels accordingly. In some of the factories studied by McKay [22], lower levels were tightly constrained and forced to make decisions in a routine manner, thus generating a panic-driven perception by personnel that things were out of control. However, in other factories, personnel involved with planning and scheduling were not tightly constrained and were observed to be using a two-stage control mechanism to adjust the decision and control process. They were aware of certain trends, patterns or signals in the manufacturing environment and were using this information to enhance the decision making process. The A-HPP model is a formalization of these observed control mechanisms.

The HPP to A-HPP transformation involves altering four key HPP assumptions:

1. Higher levels constrain lower levels and use aggregated constructs/models of them (HPP) – lower levels can guide and constrain higher levels by using more detailed and specific information about the situation (A-HPP)
2. Higher levels always know what is best for lower levels (HPP) – higher levels do not always know what is best for lower levels, and lower levels have increased scope of authority and control (A-HPP)
3. Higher levels do not know the inner workings of lower levels (HPP) – higher levels may be interested in when lower levels must relax a constraint or take a special action (A-HPP)
4. Levels are stable and have been specialized for the time horizon under consideration (HPP) – levels evolve and are constantly learning about cause-and-effect relationships that may be significant (A-HPP)

The adaptive decision framework for a particular A-HPP level consists of three major logic components:

1. *Active filter* – acquires/analyzes various types of incoming information such as *routine* manufacturing information (e.g., process routings, work orders) and *non-routine* information (e.g., "cues" related to observed/anticipated manufacturing changes or suggestions to assign a complex task to a new machine or operator), blocks/eliminates unnecessary or redundant information, and passes the routine information to the decision controller and the non-routine information to the tactical controller
2. *Tactical controller* – "heart" of the adaptation/learning mechanism, receives "cues" from the active filter and is responsible for knowing when/how much policies/procedures can be relaxed (e.g., a stronger inclination to resist system changes and to follow policies/procedures during non-critical periods than during critical periods)
3. *Decision controller* – makes the decisions and derives the plans (e.g., schedules) based upon logic (e.g., heuristics) and information input from the active filter

Using the personnel, equipment, materials and tools allocated to a particular decision level, that level can adjust or manage its capacity either to avoid the change entirely, reduce the impact associated with the change, alter the domain to create capacity to deal with the change (e.g., work overtime or request an operator from another production line), or knowingly accommodate the situation while attempting to minimize any secondary effects. These four items are proactive, or feed-forward, in nature and provide the control adaptation mechanisms needed to deal with a rapidly changing environment. Although not all manufacturing changes can be anticipated or controlled, it is believed that a large portion can be successfully managed using these concepts. The reader is pointed to McKay [22] and McKay, et al [23] for more information on the A-HPP concept.

4.2. Risk detection and algorithm adaptation

Hollywood and McKay [17] developed a design for an adaptive scheduling engine which exploits the A-HPP paradigm [22, 23]. Its framework was designed for large-scale heterogeneous computer networks facing major changes in job stream and resource availability. It provides context-sensitive scheduling by assigning jobs to hosts (i.e., resources) based upon what the resource is currently doing, what it is capable of doing and the amount of work (i.e., jobs) in queue at the resource. It also provides state-sensitive scheduling using a two-stage control model, operational and tactical, in which incoming jobs pass through *access control*, *cell sequencing* and *dispatch control* modules to be released/denied entry, prioritized and processed, respectively. Each of these *operational* modules transmits information to a *sensing and filtering mechanism* which examines the information for cues or clues to determine whether a scheduling change is necessary and what the change should be. If nothing special is detected, incoming jobs simply follow the normal scheduling path. If a cue or clue is detected, the *tactical logic* for the resource (or resource group) is activated to alter the scheduling and/or sequencing logic within the resource (or resource group). For instance, resources may be reallocated among jobs, sequencing priority heuristics may be modified to prepare for an upcoming urgent job, or a higher-level resource group may be alerted about an overload condition at a particular resource within the group. Thus, system-level changes may change the state of higher-level processes which, in turn, may change the state of lower-level processes which, in turn, control the actual scheduling components. The state transitions may be based upon mathematical formulas, pattern recognition algorithms, decision trees or simple threshold values.

Concurrently while the above processes are being performed, two support modules called *load monitoring* and *learning/trend analysis* operate to provide the resource loading data and evolutionary information, respectively, needed to make tactical decisions and to improve those decisions over time. Examples of decisions that can be improved using learning include the following:

- resource loading, using autoregressive, moving average and regression models
- detection of patterns that may trigger certain states, using data mining and artificial intelligence (AI) techniques
- detection of new patterns or new states
- scheduling responses within a particular state
- job growth (i.e., volume) patterns
- changes in resource processing abilities (e.g., processing times for particular jobs)

The performance of the scheduling framework was tested using a simple simulation model with three levels of adaptive control ability: none, reactive only and proactive control. The performance metric was waiting time in queue. Table 1 displays the results [17].

As can be seen, even basic reactive control was able to respond appropriately to the scenario dynamics by producing major increases in performance. The reader is referred to

Table 1: Comparison of Waiting Times Based on Adaptive Control Level

Adaptive control level	Expected waiting time (min.)	Std. Dev. waiting time (min.)
None	78.5	82.8
Reactive only	17.5	15.3
Proactive	15.5	15.1

Hollywood and McKay [17] for more detailed information on the above example and the adaptive scheduling framework in general.

4.3. Averse-1

The first effort to embed the logic associated with situation-dependent sequencing and risk mitigation within a mathematical formulation was the Averse-1 heuristic [25]. Averse-1 pertained to the single-machine weighted tardiness problem with static job arrivals. It was derived using the R&M (a.k.a. Apparent Tardiness Cost) heuristic as a starting point [27]. R&M is a composite heuristic which combines the weighted shortest processing time (WSPT) rule with an additional term to modify job priority based on urgency (i.e., slack) relative to due date. Since R&M is a dynamic rule, it was a reasonable foundation upon which to develop Averse-1.

Averse-1 combined R&M with an 'aversion term' to mimic how the scheduler adjusts job priorities under the threat of risk due to an event trigger. For instance, an otherwise high priority job may be delayed if it has a particularly high susceptibility to the risk. Since risk is often manifested by increased processing times for many types of events (e.g., machine breakdowns, quality/material problems causing in-line rework, inexperienced operators), extended processing time was used as the surrogate to quantify risk. The Averse-1 heuristic has the following form:

$$\pi_j^*(t) = [(w_j/p_j)e^{-S_j^+/kp_{ave}}]\ [1/(1+\beta\alpha p_j^{**}(t)/p_j^*(t))]. \tag{4.1}$$

$\pi_j^*(t)$ is the Averse-1 priority for job j at time t as a function of its R&M priority term $(w_j/p_j)e^{-S_j/kp_{ave}}$ and an aversion term $[1/(1+\beta\alpha p_j^{**}(t)/p_j^*(t))]$. The R&M priority is a function of the job weight w_j, the (nominal) job processing time p_j and job slack $S_j^+ = \max(d_j - p_j - t,\, 0)$ relative to due date d_j. p_{ave} is the mean processing time and k is a tuning parameter empirically derived based on due date tightness. The R&M priority reduces to the WSPT priority whenever job j will be late (i.e., when $S_j^+ = 0$).

The aversion term is used to adjust the R&M priority in light of the perceived risk. This term is a function of the extra job processing time $p_j^{**}(t)$ due to the disruption, the total processing time $p_j^*(t)$, the risk decay rate α and a tuning parameter β used to adjust the priority correction. The total processing time $p_j^*(t)$ is the sum of the nominal time p_j and the extra processing time $p_j^{**}(t)$. The extra processing time (i.e., risk) is given by:

$$p_j^{**}(t) = p_j\tau_j e^{-\alpha(t-t_{avs})}. \tag{4.2}$$

Thus, risk is proportional to the nominal processing time p_j and the job sensitivity τ_j. Using a concept similar to heat-loss theory, the risk decays exponentially at rate α starting at the time when the risk begins, denoted t_{avs}. This model mimics the real-world situation in which risk is at maximum shortly after the disruptive event occurs and then quickly decreases over time as the machine re-stabilizes. Once completely re-stabilized, the R&M underlying heuristic regains full control of the sequencing process.

Prior research [22] has shown that very good schedulers do exist in practice. However, they are not always perfect. They could predict that the event will have adverse consequences (i.e., risk) when, in fact, production proceeds more smoothly than expected. To study these effects, Averse-1 was tested under two extreme situations – when the risk occurred exactly as planned and when it did not occur at all. What effect will the dynamic re-sequencing of jobs using Averse-1 have in these two cases? Obviously, Averse-1 should be beneficial in the first case but detrimental in the second case, but by how much? If it could be shown that the best-case performance of Averse-1 is far better than its worst-case performance, then it would indeed be a worthy heuristic. Furthermore, what effect will the impact rate, job sensitivity and schedule hardness have?

To examine the above questions, simulation experimentation was conducted. 81 sub-cases were examined in the 'risk occurs as planned' case, namely 3 decay rate (α) levels x 3 job sensitivity (τ_j) levels x 9 schedule hardness levels. 27 corresponding sub-cases were examined at the intermediate τ_j level in the 'risk does not occur' case. The performance metric was total weighted tardiness. Overall, Averse-1 outperformed R&M alone by 7.8% across all cases when the risk occurred. When the risk did not occur, using Averse-1 resulted in an overall performance decrease of 0.7% over R&M. Thus, the best-case benefit of using Averse-1 was much greater than the worst-case cost. Moreover, the benefit was greater at high decay rates than at low rates (9.7% vs. 5.4%), at high job sensitivity levels than at low levels (14.2% vs. 2.5%) and with loose production schedules (i.e., loose due dates) than with tight schedules (11.8% vs. 5.0%).

One may legitimately ask, "What benefit would Averse-1 have over the case where the extra processing time (i.e., risk magnitude) was simply added to the processing time parameter in the R&M heuristic?" To answer this question, note the aversion term in Eq. (4.1) contains not only the risk magnitude but also its marginal rate of change. Thus, Averse-1 would still have an advantage over this "Smart R&M" heuristic. In fact, the experimentation also considered this Smart R&M version and found that Averse-1 outperformed it by 4.8% overall, as compared with 7.8% over the original "Blind" R&M.

To illustrate the dynamic re-sequencing behavior of Averse-1, Figure 1 [5] depicts a priority graph for two jobs that differ only by weight and risk sensitivity. Suppose Job 1 has a higher weight and sensitivity than Job 2. Using traditional heuristics such as R&M, Job 1 will always be sequenced ahead of Job 2 since sensitivity is ignored. With Averse-1, Job 2 will be sequenced ahead of Job 1 at any decision time before Time 10. At any time after Time 10, Job 1will be sequenced ahead of Job 2. This dynamic re-sequencing of jobs reflects the real-time dependency inherent in real-world job shop sequencing decisions. As time passes after the initial risk event, sequencing decisions gradually revert to those based on the underlying heuristic as the risk diminishes and the machine re-stabilizes.

4.4. Averse-2, Averse-3

In practice, aversion dynamics can be applied in a proactive manner as well as a reactive manner. For instance, jobs can be re-sequenced in anticipation of a disruptive event that is expected to occur soon as well as an event that has already occurred. Often, proactive aversion dynamics applies to disruptive events that can be predicted with some accuracy (e.g., pending machine upgrade or new operator), whereas reactive aversion dynamics applies to unexpected events that cannot easily be predicted (e.g., machine breakdown).

Averse-1 applied only to the reactive aversion dynamics situation. The time horizon before the event occurred (i.e., before time t_{avs}) was not considered and t_{avs} was assumed to be known since the event had already occurred. Also, values of key parameters such as

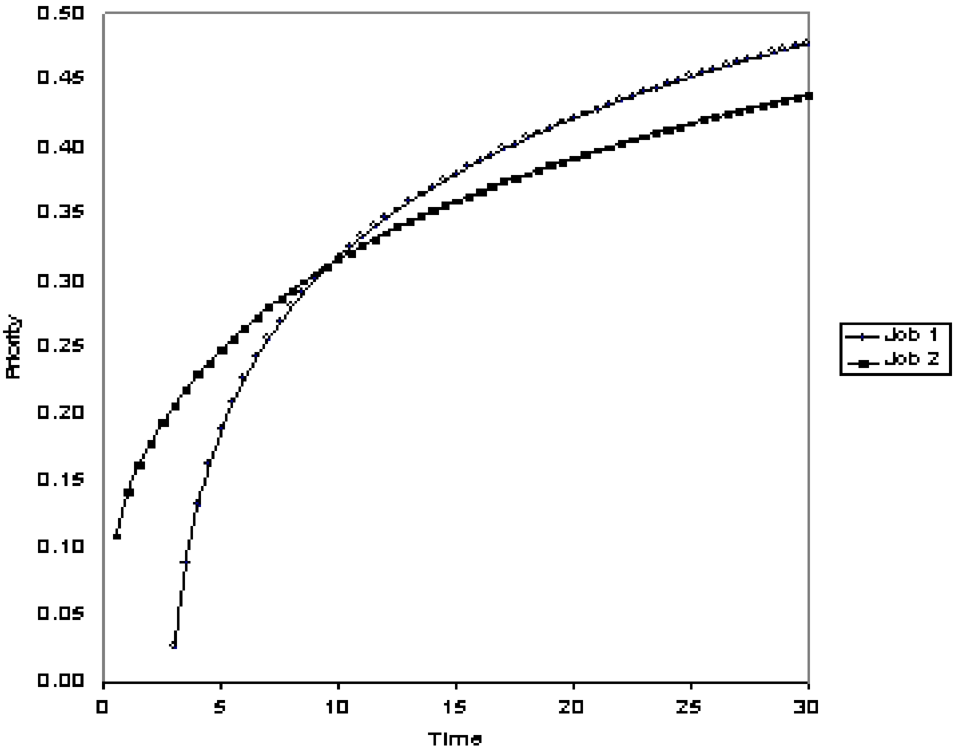

Figure 1: Averse-1 Priority Adjustment Example

risk magnitude and decay rate were assumed known and the job set was assumed as static.

How will aversion dynamics perform in the case where jobs are re-sequenced proactively in anticipation of a pending disruptive event? Furthermore, how will it perform with jobs that arrive dynamically over time? To study these questions, a revised heuristic called Averse-2 was developed [4, 5] containing the following additional features:

1. Predictive and stochastic (i.e., proactive) extension
2. Interval estimate for risk time
3. Dynamic job arrival extension

Each of these extensions will be discussed in the following paragraphs.

In a reactive scenario, the event has occurred and, thus, values of key event/risk parameters are known. However, in a proactive scenario, the scheduler uses expectations about the future to predict these parameter values based upon personal judgment and experience as well as hard and soft information gathered from resources within the organization. The following stochastic parameters were considered:

1. Event time t_e – the predicted start time of the event
2. Event duration t_d – the predicted duration of the event (i.e., downtime)
3. Risk magnitude factor γ – the initial magnitude of the risk
4. Risk decay rate α – the rate at which the machine recovers from the risk

Since these parameters are stochastic, their predictions are subject to error. The predicted value of each parameter is assumed to follow a normal distribution centered about its true (unknown) value. The normal distribution was selected since it models the case where the expected value of the prediction equals the realized value (i.e., unbiased estimate) and the predicted value is symmetric around the true value. The accuracy of predictions depends on the scheduler's skill and affects the success of the proactive schedule adjustments. Skill can be quantified by the dispersion of the predicted values from the realized values. Thus, we define a scheduler's skill level by a coefficient of variation (CV). An accurate scheduler has a low CV. An inaccurate scheduler has a high CV. In summary, each stochastic parameter can be sampled as follows:

$$PredictedValue \sim Normal(RealizedValue,\ RealizedValue\ *\ CV). \tag{4.3}$$

In reality, jobs are processed over *intervals* of time rather than *points* in time. However, Averse-1 uses a point estimate for risk time. Redefining this to an interval estimate will increase the reality of the heuristic and permit a job's processing time to be segmented to reflect that a job can start processing in a non-impacted (i.e., zero risk) state and finish in an impacted state, as would be the case when a disruptive event occurs partway through the job's processing cycle. Thus, we redefine the point estimate in Eq. (4.2) and introduce a risk magnitude factor γ to consider that different events can result in different risk levels:

$$f_j(t) = \gamma\tau_j e^{-\alpha(t-t_{avs})}. \tag{4.4}$$

Whether or not a job incurs risk depends upon whether its processing interval coincides with the "risk zone" (i.e., decay curve anchored at t_{avs}). If job j is processed over time interval $\Delta t = [a_j, b_j]$ in which $a_j, b_j < t_{avs}$, it will not incur risk and, thus, will incur only its nominal time. Conversely, suppose the job is processed over interval $\Delta t = [a_j, b_j]$ in which $a_j, b_j \geq t_{avs}$. Also, suppose it has nominal time p_j and is available at time $t = a_j$. In the *nominal sense*, if it starts at time a_j, then it will finish at time $b_j = a_j + p_j$. However, since its processing interval is within the risk zone, it will incur additional processing time (i.e., risk) $p_j^{**}(\Delta t)$ as defined by the area under its $f_j(t)$ curve:

$$p_j^{**}(\Delta t) = \int_{a_j-t_{avs}}^{b_j-t_{avs}} f_j(t)dt = \int_{a_j-t_{avs}}^{b_j-t_{avs}} \gamma\tau_j e^{-\alpha(t-t_{avs})}dt. \tag{4.5}$$

To derive Averse-2, the original R&M derivation [27] is repeated using Eq. (4.5) to obtain Averse-2 for the *static* arrival case:

$$\pi_j^*(t) = \pi_j(t)/\{1 - \beta\gamma\tau_j[e^{-\alpha(b_j-t_{avs})} - e^{-\alpha(a_j-t_{avs})}]/p_j^*(\Delta t)\}. \tag{4.6}$$

$\pi_j(t)$ represents the R&M priority term, $(w_j/p_j)e^{-S_j^+/kp_{ave}}$. Next, the *X-Dispatch* dynamic arrival priority adjustment term $1 - B(r_j - t)^+/p_{\min}$ [27] is appended to complete the development of Averse-2 for the *dynamic* arrival case:

$$\pi_j^*(t) = \pi_j(t)/\{1 - \beta\gamma\tau_j[e^{-\alpha(b_j-t_{avs})} - e^{-\alpha(a_j-t_{avs})}]/p_j^*(\Delta t)\}\{1 - B(r_j - t)^+/p_{\min}\}. \tag{4.7}$$

X-Dispatch reduces the priority of jobs that have not yet arrived. If job j is not in queue at time t, then its arrival time $r_j > t$ and, thus, the job's priority is reduced to reflect the cost of holding the machine idle. B = tuning parameter. $p_{\min}$ = minimum processing time job in the queue. X-Dispatch expands the feasible job set to both the set of in-queue jobs and the set of jobs scheduled to arrive by time $t + p_{\min}$. Obviously, the machine would never be held idle longer than $p_{\min}$ units since, otherwise, the $p_{\min}$ job could be built entirely within the idle period.

Simulation was used to test the performance of Averse-2 against two versions of X-Dispatch: Smart X-Dispatch, which included risk time within processing time, and Blind

X-Dispatch, which did not. Two levels of event realization (event/risk occurs as predicted and does not occur at all), three levels of job arrival tightness (static, tight and loose) and four levels of schedule hardness (high/low combinations of due date mean and dispersion) were considered. The performance metric again was total weighted tardiness. The following key results were noted at the $\alpha = 0.05$ significance level:

1. Averse-2 significantly outperforms both X-Dispatch variants when the event/risk occurred (p<0.0005). Specifically, Averse-2 outperformed Smart and Blind X-Dispatch by 6.13% and 14.06%, respectively.
2. Averse-2 did *not* significantly underperform either X-Dispatch variant when the event/risk did not occur at all, despite being planned for in advance (p=0.9447 and 0.8931, respectively). Smart and Blind X-Dispatch outperformed (insignificantly) Averse-2 by 0.96% and 1.09%, respectively, in these cases.

With respect to job arrival tightness, Averse-2 outperformed Smart X-Dispatch by 8.02% with static arrivals. As arrivals become dispersed, the Averse-2 advantage decreased to 5.04% and 4.69% at the tight and loose arrival levels, respectively. This finding is intuitive since, as arrivals become dispersed, fewer candidate jobs are available in queue.

Regarding schedule hardness, Averse-2 outperformed Smart X-Dispatch by 12.55% in a soft schedule with narrowly dispersed due dates, 10.18% in a soft schedule with widely dispersed due dates, 3.42% in a hard schedule with narrowly dispersed due dates and 3.45% in a hard schedule with widely dispersed due dates. This finding is also intuitive since soft schedules containing slack provide more opportunity to re-sequence jobs. The reader is referred to Black [4] and Black, et al. [5] for a more detailed discussion.

Although Averse-2 tries to schedule the "best" job around a perceived event, it makes no provision to avoid an in-process job at the time of the event. If a job is in process when the event occurs, the job will be interrupted and may have to restart later, thus resulting in a fragmented processing profile. Often, this *fragmentation* can result in duplication of effort/resources, damaged parts or inconsistent batch quality. For instance, after a machine breakdown, an in-process job may need to be scrapped or the machine may need to be set up a second time. Consequently, schedulers may seek to reduce fragmentation in one of two ways [22]. They may intentionally hold the machine idle when a disruption is believed imminent, or they may schedule a lower priority job to bridge the gap between the current time and the predicted event time. This extra aversion logic was embedded in a new heuristic called *Averse-3* [4, 7].

With respect to mathematical formulation and algorithmic implementation, Averse-3 differs from Averse-2 in two ways:

1. It contains an additional priority reduction term that penalizes jobs whose processing intervals may extend into the risk zone.
2. It contains a logical constraint that applies when no job is capable of finishing before the predicted event time.

Two priority reduction profiles are studied: linear and exponential. Suppose a sequencing decision is made at some time before the predicted event time e_p. To compensate for error in the event time prediction, a priority reduction window of width zp_{ave} is used to reduce the priorities of jobs whose completion times will extend into this window as shown in Figure 2 [7]. p_{ave} represents the mean job processing time. z scales the window width in proportion to p_{ave}. Once the event occurs, the Averse-3 heuristic reduces to Averse-2.

As the event draws closer, it is likely *no* job can finish before the predicted event time. What should be done in this case? One solution is to simply hold the machine idle until the

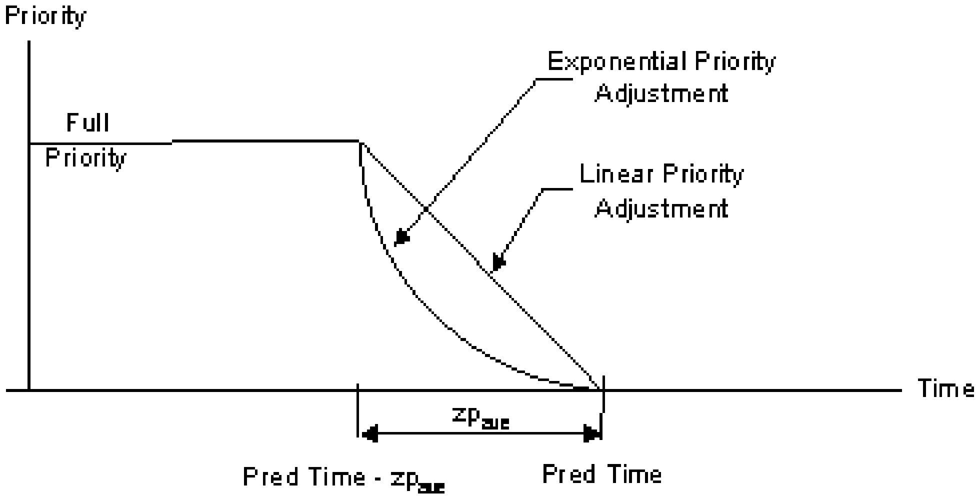

Figure 2: Averse-3 Priority Adjustment Under Imperfect Event Time Predictability

event occurs (or until a shorter job arrives that can finish in time). Imposing this *logical constraint* would reduce fragmentation to a greater extent, although it would also likely degrade weighted tardiness due to the imposed idle time. On the other hand, this constraint could be ignored and the 'best' job could simply be chosen, despite that fragmentation will likely occur. In practice, whether or not to utilize this constraint would be dictated by how 'expensive' fragmentation is.

The experimentation examined fragmentation and weighted tardiness performance in four major cases: exponential or linear priority reduction and the presence or absence of the logical constraint. In each case, we examined eleven levels of event time predictability as measured by a coefficient of variation CV (CV = 0 means perfect predictability), three levels of job arrival time dispersion ('tightness') and four levels of schedule hardness.

The following five propositions were ultimately validated by the experimentation:

1. Averse-3 can significantly reduce fragmentation (> 50% over Averse-2) while degrading weighted tardiness performance to a much lesser extent (< 5%).
2. As event time predictability improves (i.e., as CV decreases), both weighted tardiness performance and fragmentation performance improves.
3. As the width of the priority reduction window increases (i.e., as z increases), fragmentation is further reduced but weighted tardiness performance degrades.
4. When the logical constraint is used, fragmentation is further reduced but weighted tardiness performance further degrades.
5. When the logical constraint is used, the choice between exponential or linear priority adjustment is insignificant since the effect of the constraint overwhelms the priority adjustment.

Results indicated that it is possible to significantly improve (i.e., reduce) fragmentation using Averse-3 while only minimally impacting weighted tardiness. When CV = 0 (perfect event predictability), fragmentation can be eliminated completely using $z = 0$ while incurring a weighted tardiness degradation of 2.13%. When CV > 0, fragmentation can still be reduced significantly while incurring only modest weighted tardiness degradation. For instance, when CV = 0.1, fragmentation can be reduced by 29.3% with only 1.38% degradation in weighted tardiness by using exponential priority adjustment with z =1 without the logical constraint. If the logical constraint is used, fragmentation can be reduced considerably even more, for instance by 68.37% with 4.36% weighted tardiness degradation by using $z = 0.5$ in conjunction with exponential or linear priority reduction. Even more fragmentation reduction is possible by increasing the value of z, albeit at the expense of more weighted tardiness degradation. Even when CV $= 0.25$, it is still possible to reduce fragmentation by over 50% using Averse-3 while incurring weighted tardiness performance degradation of under 5% relative to Averse-2.

Averse-3 fragmentation performance relative to arrival and schedule hardness levels was also examined. For both the exponential and linear profiles *with* the logical constraint, fragmentation reduction was fairly insensitive to arrival and due date profiles and was reduced by 82-83%. For the exponential profile *without* the constraint, fragmentation reduction was also rather insensitive to arrival and due dates and was reduced by 27-29%. For the linear case without the constraint, fragmentation reduction was better than the exponential case, specifically 32-45%, with the largest improvements being realized in the tight arrival scenario.

With regards to schedule classification, Averse-1 produces non-delay schedules since job arrivals are static and, thus, inserted idle time will never be necessary. On the other hand, Averse-2 produces active schedules. Averse-2 accommodates dynamic jobs arrivals using the X-Dispatch term with which inserted idle time is permissible. However, this idle time would never exceed the processing time of the shortest job in queue. Thus, it would not be possible to construct another schedule in which a job could finish earlier without delaying some other job. With regards to Averse-3, it produces neither active nor semi-active schedules since it is possible to hold the machine idle until the event occurs whenever the logical constraint is used. Thus, it may be possible to finish a job earlier (i.e., a local left-shift is possible); however, doing so could result in fragmentation.

4.5. Risky jobs

Schedulers have been observed to be averse to scheduling risky jobs on highly loaded machines, preferring instead to hold them until quieter periods, to mitigate the disruptive impacts on subsequent jobs [22]. In doing so, the scheduler behaves as if the planning processing time used for job prioritization has been inflated for the risky jobs. This situation can be considered the complementary case to the Averse-1, -2 and -3. In that research, jobs were the *recipients* of risk due to a disruptive event. The event occurred at a specific time at which the risk (i.e., extra processing time) commenced and ultimately decayed in an exponential fashion. Conversely, in this *risky jobs* research [6], risk is inherent within the job itself and, thus, does not decay over time.

The objective of this research was to develop simple yet robust approximation policies based on the concept of applying "safety time" to processing times used in dispatching heuristics. We assumed the total actual processing time p_j for job j is the sum of its nominal time n_j plus a random amount of "trouble time" that follows a negative exponential distribution and is scaled by a job risk factor r_j and the nominal time. Thus, the mean trouble time is $n_j r_j$ and the total actual processing time for a job is given by:

$$p_j = n_j[1 + er_j] \tag{4.8}$$

e is a random variable drawn from the negative exponential distribution with mean of 1.0.

For job prioritization purposes, the scheduler was assumed to inflate job processing times by adding "safety time" S to account for the inherent variability in their actual (but unknown) processing times. The safety time impacts the priority processing time in proportion to the job's risk level and nominal processing time. That is, processing times used for priority assignment purposes are:

$$p_j^* = n_j[1 + (1 + S)r_j] \tag{4.9}$$

When $S = -1$, jobs are prioritized using their nominal processing times. When $S= 0$, jobs are prioritized using their expected (mean) processing times. When $S > 0$ or $S < 0$, positive or negative safety time, respectively, is used to compute job priorities.

In the experimentation, the single-machine static-arrival problem was considered. The weighted flowtime and weighted tardiness objectives were investigated along with their respective priority rules, weighted shortest processing time (WSPT) and Rachamadugu & Morton (R&M). The maximum lateness objective was also considered but was found to be indifferent to risky jobs since its corresponding optimum rule, earliest due date, is independent of processing time. The remaining two objectives were studied under nine major cases, namely three risk levels $r_j = 0.5$, 1.0 and 2.0 (low, medium, high) and three risk knowledge levels (true mean risk r_j is known, r_j is unknown and estimated using either one or three actual processing time observations from the past data; the three-observation case yields better information than one observation). The three risk levels $r_j = 0.5$, 1.0 and 2.0 correspond to the cases where the risk time is one-half, equal or twice that of the nominal processing time, respectively. Within each of these nine cases, 24 schedule hardness levels were considered for the weighed tardiness objective.

One analytical result was immediately available. For the weighted flowtime objective with a *known* mean risk level, a safety time $S = 0$ will always be optimal. The reader is referred to Black, et al. [6] for proof.

For the weighted flowtime objective with an unknown, but estimated, risk mean, the best safety time S can differ from zero and was found to be negative in the small risk cases and positive in the high risk cases. However, if the simple *approximation policy* $S = 0$ was used, the percent penalty incurred averaged 0.54% (as opposed to using the best value in each individual case) and did not exceed 2.29% in any specific case. Since the true risk mean is unknown anyway and, thus, can be in error, this simple approximation policy appears to be reasonable.

For the weighted tardiness objective at *high* tardiness levels (i.e., tight schedules), results are fairly similar to the weighted flowtime objective (i.e., R&M reduces to WSPT when all jobs are late). Specifically, the best S value is near zero when the true risk mean is known. It begins to diverge from zero as the risk mean increases and as the quality of information (i.e., number of observations) decreases. At such times, the best S is positive. Consequently, the same robust approximation policy $S = 0$ applies with an average penalty of 0.46% and a maximum penalty not exceeding 2.25% in any specific case.

For the weighted tardiness objective at *low* tardiness levels (i.e., loose schedules), the best S was plotted versus Log_{10}(weighted tardiness) at each risk level. It was found that the best S increases roughly linearly with Log_{10}(Weighted Tardiness). In the low risk case, the best S value is negative. Why negative safety time? In the low risk case with a loose schedule, there is usually some slack in the schedule to absorb the occasional risky job. As the risk level increases from low to medium, the best S increases towards zero. As the mean risk increases from medium to high, positive S is often useful. When the mean risk level is unknown and estimated to be high, positive S becomes even more useful. To develop a simple, robust approximation policy, we asked "What if $S = 0$ is simply used for all low tardiness cases?" To analyze this policy, % Penalty for using $S = 0$ was plotted versus Log_{10}(weighted tardiness) at each risk level. Two results were apparent: % Penalty is lower at high risk values than low risk values, and % Penalty decreases in Log_{10}(weighted tardiness). Accordingly, a robust approximation policy for the weighted tardiness low tardiness case is: If the mean risk is low, use $S = -0.8$. If the mean risk is moderate, use $S = -0.5$. If the mean risk is high, use S = -0.2. This combined policy results in an average penalty of 1.1% and a maximum penalty of 3.8% relative to the best S value in each individual case. These penalties pale in comparison to the case when only nominal times ($S = -1$) are used. Moreover, since in practice many other parameters have error associated with them (e.g., nominal times, weights, risk estimates), the penalties for these practical and simple-to-use policies seem minimal.

How critical is the assumption that risk has a negative exponential distribution? To study this issue, a sensitivity analysis was performed in which risk was sampled uniformly within the interval [0, 2*mean job risk]. It was found that the policy of using expected risk-adjusted processing times ($S = 0$) remained robust for the weighted flowtime and weighted tardiness high tardiness cases with a maximum penalty of only 0.7%. In the weighted tardiness low tardiness case, an alternate value S = -0.3 performed better.

4.6. Anticipatory batch insertion

Anticipatory batch insertion deals with running a small "test batch" prior to the main batch in order to resolve problems that would otherwise affect the entire batch [8, 37]. This research extended McKay's aversion dynamics concepts [22] beyond sequencing to lot sizing applications. The main idea is, that by inserting a test batch in advance of the main batch, the risk will be absorbed by the test batch and, thus, will not affect the remainder of the batch. This strategy should result in reduced scrap, reduced production costs and perhaps reduced weighted tardiness when the risk materializes as anticipated. However, it will result in an extra setup being performed unnecessarily when the risk does not materialize as planned, thus increasing setup costs and perhaps increasing weighted tardiness as well.

The basic problem considered was the single-machine static arrival model with a single test batch of size 10% (of the entire batch) to be run immediately before the main batch in hopes of absorbing risk (i.e., scrap) associated with the potential disruption. The performance of this strategy was measured using cost and weighted tardiness. The cost factor included setup, production and scrap costs. The sensitivity of the strategy was explored with respect to schedule hardness (i.e., due date), job sensitivity to the risk and potential risk magnitude (i.e., % scrap).

Four major cases were considered in the experimentation:

1. Batch insertion *was not* used and the disruption (i.e., risk) *did not* materialize
2. Batch insertion *was not* used and the disruption *did* materialize
3. Batch insertion *was* used and the disruption *did* materialize
4. Batch insertion *was* used and the disruption *did not* materialize

The difference between Case 2 & 3 results yield insight into the benefit of using batch insertion when the disruption and risk occurs, and the difference between Case 1 & 4 results yield insight into the cost of using batch insertion unnecessarily. Within each major case, 81 sub-cases were studied consisting of 9 schedule hardness levels, 3 product sensitivity levels and 3 risk magnitude levels. Table 2 [8, 37] displays aggregate results for each case averaged across schedule hardness, product sensitivity and risk magnitude levels:

As expected, the lowest (i.e., best) values for cost and weighted tardiness occur in Case 1 when no disruption occurred and batch insertion was not used. Also as expected, the highest values for cost and weighted tardiness occur in Case 2 when a disruption occurred

Table 2: Aggregate Results for the Four Major Experimental Cases

Case	Total Cost	Weighted Tardiness
1. No Insertion-No Disruption	2282	129,660
2. No Insertion-Disruption	2540	164,030
3. Insertion-Disruption	2367	146,660
4. Insertion-No Disruption	2321	141,560

but batch insertion was not used. In Case 3, the disruption occurred and the batch insertion concept was used to mitigate the risk. Here, cost improved by 6.8% and weighted tardiness improved by 10.6% relative to Case 2 where no batch insertion was used. Although these values will fluctuate somewhat based on the specific hardness, sensitivity and risk magnitude level, these aggregate results serve to validate the overall usefulness of the batch insertion strategy. In Case 4, no disruption occurred but batch insertion was used in anticipation of a disruption. Here, cost increased by 1.7% and weighted tardiness increased by 9.1% relative to Case 1 where no batch insertion was used. Overall, the degradation associated with using batch insertion when the disruption does not occur is less than the benefit achieved by using it when the disruption does occur. It was found that this degradation could be reduced by selectively choosing whether to use batch insertion within specific situations related to hardness, sensitivity and risk magnitude.

As expected, the performance advantages of using batch insertion when the disruption occurs are highest when the schedule is loose, when multiple products are sensitive to the disruption and when risk magnitude is high. These three relationships appear to be close to linear. The reader is referred to Varghese [37] and Black, et al. [8] for more detailed information.

5. Discussion

In section 4, six research efforts on Aversion Dynamics were summarized. All of the research focuses on perturbations or changes in the situation that warrant extraordinary decisions. As the preliminary and exploratory research has illustrated, it is possible to formulate and use extraordinary reasoning with results which are mathematically significant. This observation is interesting for three reasons. First, the mathematical studies suggest that human schedulers who routinely use such strategies are actually quite insightful and are not naïve sequencers. This is encouragement for further studies on the human contribution and the heuristics deployed. Second, in order to formulate and control the aversion heuristics, the information used in the heuristic is extended. Information such as sensitivity to risk, degree of sensitivity, and predictability of risks are necessary to capture the real world phenomena. Third, special algorithms or mathematical extensions are necessary to trigger the state-dependent logic and to control the phase-out or recovery phase of the logic. These extensions are necessary for the results to match what happens in the real world. As the heuristics and algorithms were inspired by real world problems, it is important that the research has a high fidelity to the original context.

The research can only be claimed to be exploratory and preliminary. It is not claimed that any of the formulations or parameter settings are *optimal* or *best*. The initial challenge was to see if formulations could be constructed and if the formulations would be robust. Towards those goals, we believe that the research has made progress. There are still challenges though. For example, would different formulations provide better control of the aversion logic, or provide learning mechanisms? What are optimal settings for different situations? The formulations have different settings for sensitivity, reactions, and recovery. What are the tradeoffs? What are the dynamic relationships? For example, in the inventory concepts, how many batches and what would the timing of batches be for different types of risk? In manufacturing, there are secondary impacts and risks associated with a primary event. For example, a perturbation concerning one machine might indirectly affect another machine or another subset of work elsewhere in the plant. How can the single or two-machine problems be generalized? What would be the best way to incorporate the concept of aversion into meta-heuristics such as genetic algorithms or tabu search? We hope that the ideas and challenges are intriguing enough for other researchers to contemplate solutions.

Another area to explore is the application of Aversion Dynamics in system wide heuristics (e.g., not a myopic, single machine focus) and meta-heuristics (e.g., tabu search, genetic algorithms). The present formulations are designed as myopic dispatch rules for single machine situations. Research has started on a parallel machine formulation, but this is also limited to the simple dispatch logic. Heuristics with a broader view might be able to take the risk information and perform more proactive routing and risk mitigation. This is a research area with high potential because of the options available for alternative routing. While it has not been explored, it is also possible to consider the wider application of the Averse dispatching heuristics in a larger job shop problem (e.g., used for local dispatch control at each machine). While the negative impact on the global performance is expected to be relatively small, the precise effect is not known. The impact is expected to be minimal because the risks being modeled are associated with the machine and not the job. In a risky job situation, the impact might be greater. The negative impact is also mitigated because of the inherent aversion logic that decays and reverts back to the underlying heuristic. It would be interesting and useful research to explore the global impact in a larger job shop problem.

In addition to the deeper exploration of the concepts already developed, there are other forms of risk and issues to be considered. Thus far, we have used time as a surrogate for the risk impact; how could quality and yield be directly modeled? Specific models of risk are not used. What would risk models of material, processes, job characteristics and resources look like? How could they be incorporated into production control concepts?

Risk management is obviously the underlying theme behind Aversion Dynamics and some of these questions. While the human scheduler is possibly sensitive to a wide range of possible risks, it is not reasonable to consider all of the possible risks as being suitable for inclusion in algorithms or systems. McKay [22] specifically considered what risks would be reasonable and feasible for inclusion in modern systems for production control. There are three general classes of risks - those possible to anticipate with relative accuracy, those which can be anticipated with a zone of possibility, and those not possible to predict and are unknown till they actually occur.

An example of the first class is the scheduled upgrade of a machine on a specific date. There is a risk associated with a machine upgrade and such events are usually planned in advance and are known. This class would include planned changes in material, changes in vendors, line or machine upgrades or introductions, scheduled product or process changes, or scheduled changes in crews. New or prototype work is also of this type since it can be scheduled and anticipated. In general, risks associated with planned changes in the physical system are the most reasonable to consider. These would be changes that could be detectable through the material, routing, or maintenance records found in most modern manufacturing control systems. While the precise impact of the risk may not be known, it is reasonable to conclude that risks to processing time and yield might occur for a shift or two and have a destabilizing affect. A rough heuristic is: the larger the change, the larger and longer the possible risk.

The second classification includes those events or triggers which are suspected to occur sometime within a time zone, but the precise time is not known. For example, there is a change planned for a specific month, but the precise day within the month is not known. These types of risks can still be anticipated for and it is possible to build heuristics sensitive to them. The human's heuristic scans the work to be done, identifies work possibly sensitive to the expected change, and alters the priority of the work so that the work is planned to start earlier than originally planned. Simply put - do not wait till the last moment for these jobs. The concept is to start the work before it needs to be done, but not too early either. If the event happens in the same time period, then the work can be safely delayed. This is in contrast to the situation where the work is scheduled as late as possible and then the event happens at the same time. To make this classification feasible in a real situation, additional encoding is likely necessary for the 'change will happen sometime next month' information instead of 'change will happen on the 15th of the month' data found on most effective date records.

The third classification is the reactive scheduling situation. For example, a machine breaks down and a repair is needed, a material does not arrive and a substitute is needed, the preferred machine is not available and a rarely used machine is considered. In these cases, it is not possible to anticipate and the planners are restricted to intelligent reactions. While hard to include in a planning module, these situations are feasible to consider for a realtime scheduling tool. The realtime tool can detect these unexpected changes in the plan and make sequence choices which will reduce or avoid the possible impact. For example, a cheaper or more bountiful part from the job queue can be selected instead of the next job scheduled if the next job has a higher repair cost. It might also be possible to split batches and instead of committing a whole batch of material, a short or smaller batch is made and checked before starting the whole batch.

If it is possible to track historical data, it is then possible to consider additional risks in each of the three categories, or to fine-tune the reactions. For example, if the part has not been made for several months and the same machines and same crew is involved without any changes, the risk is perhaps low. However, if the machines have been upgraded since the part was made, a different vendor is supplying the material, and a different operator is on the machine - it might not be wise to assume that all will happen as planned. In theoretical modeling, it is possible to consider different strategies for controlling the risk in these cases. It is also important to consider the parameters that will guide the compromises. For example, a small number of changes since the last build might dictate a rather optimistic reaction (e.g., one extra batch of a small size one week before the desired batch). A large number of changes might dictate two or three test runs starting several weeks in advance of the final batch. Research investigating these trade-offs is suggested as a later phase of work after the basic relationships have been explored and modeled.

The approach so far in the Aversion Dynamics research has been to explore the basic phenomena and to understand the various factors to include. This has led to the general approach of large scale computational experiments with tests for sensitivity and robustness to such factors as schedule hardness, probability of impact, and duration of effect. Once the research enters the normative or predictive phase, the optimal or best choices for solution formulation will be sought. This suggests that the need will created for specific benchmarks and standards for such research components as: job problems (e.g., number of machines, jobs, processing time distributions, etc.), baseline distributions for risk, risk impact, and risk recovery. Until such benchmarks exist, it is recommended that sensitivity and robustness tests using large scale experiments continue to be used. Different distributions for risk and impact can be incorporated in this fashion and if the same parameters are used, comparisons can be made. It is the convention in such studies to publish parameters and experiment particulars and these are useful for replicating a previous result before extending the concept.

6. Conclusion

Part of the gap between theory and practice is the ability of theory to capture and represent necessary and sufficient aspects of the problem and solution. Based on empirical studies, it is suggested that concepts such as Aversion Dynamics are necessary for bridging the gap. The Aversion Dynamics research suggests that it is feasible to create such concepts and that the concepts can have a significant impact on the objective functions. These are not sufficient to completely bridge the gap, but we believe that they are a necessary part of the problem. Such decisions dominate the schedulers' day and anchor (and hence decompose) the scheduling or plan. If the theoretical solutions ignore such decisions and disregard their importance to feasible and operational schedules in the real world, the theoretical concepts will remain theoretical.

We hope that others will investigate risk aversion concepts and other real world heuristics. Not everything a scheduler does is right or good, but it is also possible that what they are doing makes sense mathematically and that they are actually creating a very good *executable* sequence.

Acknowledgements

This research has been supported in part by NSERC grant OGP 0121274 on 'Adaptive Production Control'.

References

[1] R.N. Anthony: *Planning and Control Systems* (Harvard Business School Press, Boston, 1965).

[2] R.N. Anthony: *The Management Control Function* (Harvard Business School Press, Boston, 1988).

[3] H. Aytug, M. A. Lawley, K.N. McKay, S. Mohan and R. Uzsoy: Executing production schedules in the face of uncertainties: a review and some future directions. *European Journal of Operations Research*, **165** (2005), 86-110.

[4] G.W. Black: *Predictive, Stochastic and Dynamic Extensions to Aversion Dynamics Scheduling.* Ph.D. Thesis, University of Alabama in Huntsville (2001).

[5] G.W. Black, K.N. McKay and S.L. Messimer: Predictive, stochastic and dynamic extensions to aversion dynamics scheduling. *Journal of Scheduling*, **7** (2004), 277-292.

[6] G.W. Black, K.N. McKay and T.E. Morton: Aversion scheduling in the presence of risky jobs, forthcoming. *European Journal of Operations Research,* (2004).

[7] G.W. Black, K.N. McKay and S.L. Messimer: Anti-fragmentation in aversion dynamics scheduling. *International Journal of Production Research*, **43** (2005), 109-129.

[8] G.W. Black, K.N. McKay and S.E. Varghese: 'Anticipatory batch insertion' to mitigate perceived processing risk. *International Journal of Production Research*, forthcoming.

[9] G. Buxey: Production Scheduling: Practice and Theory. *European Journal of Operational Research*, **39** (1989), 17-31.

[10] F.G. Coburn (circa 1918): Scheduling: The Coordination of Effort, In I. Mayer (ed.), *Organizing for Production and Other Papers on Management 1912-1924* (Hive Publishing, 1981).

[11] R.N. Conway: *A New Research Agenda For Scheduling*, Presentation (William L. Maxwell Symposium, Cornell University, 1998).

[12] R.N. Conway: W.L. Maxwell and L.W. Miller: *Theory of Scheduling* (Addison Wesley, Reading, MA, 1967)

[13] S. Crawford and V.C.S. Wiers: From anecdotes to theory: reviewing the knowledge of the human factors in planning and scheduling. In B.L. MacCarthy & J.R. Wilson (Eds.), *Human Performance in Planning and Scheduling.* (Taylor & Francis, London, 2001).

[14] R.A. Dudek, S.S. Panwalker and M.L. Smith: The lesson of flowshop scheduling research. *Operations Research*, **40** (1992), 7-13.

[15] D. Erlenkotter: An early classic misplaced; Ford W. Harris's economic order quantity model of 1915. *Management Science*, **35** (1989), 898-900.

[16] S.C. Graves: A review of production scheduling. *Operations Research*, **29** (1981), 646-675.

[17] J.H. Hollywood and K.N. McKay: An adaptive scheduling framework for heterogeneous computer networks. *Control Engineering In Practice*, **12** (2004), 725-734.

[18] C.E. Knoeppel: *Installing Efficiency Methods* (The Engineering Magazine, New York, 1915).

[19] B.L. MacCarthy and J. Liu: Addressing the gap in scheduling research: a review of optimization and heuristic methods in production scheduling. *International Journal of Production Research*, **31** (1993), 59-79.

[20] K.N. McKay: *Conceptual Framework For Job Shop Scheduling.* MASc Dissertation, University of Waterloo (1987).

[21] K.N. McKay, J.A. Buzacott and F. Safayeni: The scheduler's knowledge of uncertainty: the missing link, In *Knowledge Based Production Management Systems*, J. Browne (ed), Conference on Knowledge Based Planning Systems (Galway, Ireland 1988), North-Holland, Amsterdam, (1989), 171-189.

[22] K.N. McKay: *Production Planning and Scheduling: A Model for Manufacturing Decisions Requiring Judgement.* Ph.D. Thesis, University of Waterloo (1992).

[23] K.N. McKay, F. Safayeni and J.A. Buzacott: An information systems based paradigm for decision making in rapidly changing industries. *Control Engineering Practice*, **3** (1995), 77-88.

[24] K.N. McKay and J.A. Buzacott: The application of computerized production control systems in job shop environments, *Computers In Industry*, **42** (2000), 79-97.

[25] K.N. McKay, T.E. Morton, P. Ramnath and J. Wang: Aversion dynamics – scheduling when the system changes, *Journal of Scheduling*, **3** (2000), 71-88.

[26] K.N. McKay, M. Pinedo and S. Webster: A practice-focused agenda for production scheduling research. *Production and Operations Management*, **11** (2002), 249-258.

[27] T.E. Morton and D.W. Pentico: *Heuristic Scheduling Systems* (John Wiley & Sons, New York, 1993).

[28] R. O'Donovan: *Predictable Scheduling and Aversion Dynamics for a Single Machine.* Master's thesis, Purdue University (1997).

[29] R. O'Donovan, R. Uzsoy and K. McKay: Predictable scheduling and rescheduling on a single machine in the presence of machine breakdowns and sensitive jobs, *International Journal of Production Research*, **37** (1999), 4217-4233.

[30] M. Pinedo: *Scheduling Theory, Algorithms and Systems*, (Prentice-Hall, New Jersey, 1995).

[31] W.F. Pounds: The Scheduling Environment. Muth, J.F. and G.L. Thompson (eds.), In *Industrial Scheduling* (Prentice-Hall, Englewood Cliffs, 1963), 5-12.

[32] W.L. Robertson: Quality control by sampling, *Factory and Industrial Management.* **76** (1928), 503-505.

[33] F.A. Rodammer and K.P. White: A recent survey of production scheduling. *IEEE Transactions on Systems, Man, and Cybernetics*, **18** (1988), 841-851.

[34] W.A. Shewhart: *Economic Control of Quality of Manufactured Product*, (Van Nostrand, 1931).

[35] E.A. Silver, D.F. Pike and P. Peterson: *Inventory Management and Production Planning and Scheduling* (John Wiley, New York, 1998).

[36] P.P.M. Stoop and V.C.S. Wiers: The complexity of scheduling in practice. *International Journal of Operations and Production Management*, **16** (1996), 37–53.

[37] S.E. Varghese: *Dynamic Batch Insertion to Mitigate Perceived Processing Risk,* MASc Dissertation, University of Waterloo (2004).

Kenneth N. McKay
University of Waterloo
200 University Avenue West
Waterloo, ON
Canada N2L 3G1
E-mail: kmckay@uwaterloo.ca