Chapter 4  Performance of the WLC release methods

This chapter explores the performance of the classical WLC release methods in more detail. In the previous chapter we formulated three questions which will be researched in this chapter:

1. How do the classical WLC release methods perform with respect to load balancing and the timing of job release?
2. What is the influence of the norm type and the norm levels on the performance and how does this influence relate to routing variety?
3. How sensitive is the performance to other factors, particularly the parameters of the release methods?

These questions will be analysed by means of a simulation study. The next section discusses the experimental design of the study, including the modelling choices and the performance indicators that have been monitored. Section 4.2 overviews the main results of the simulation study, while section 4.3 presents the results of an extended sensitivity analysis. The final section of this chapter assesses the results and relates them to the considerations of the previous chapter in order to find possibilities for improvement.

4.1  Experimental design

The model used in the simulation studies is kept as basic as possible to avoid any noise that might cloud the sight on causes and effects. The number of experimental factors is restricted to four. However, sensitivity of results for a number of other factors is studied as well, which will also verify the appropriateness of their settings in the basic experimental design. These factors will be presented in section 4.3, together with the results of the sensitivity analysis. Regarding the basic experimental design we will first discuss the job and shop characteristics in the next subsection. Subsections 4.1.2 through 4.1.4 detail the methods applied at job entry, release, and priority dispatching respectively. The final subsection overviews the design and the measurements of performance.

4.1.1  Job and shop characteristics

The performance of the classical WLC release methods is investigated by a simulation study. Table 4.1 summarises the characteristics of the simulation model.
Shop: 6 stations, each with unique capacity;
Routing length: discrete uniformly distributed on [1, 6];
Stations in routing: random choice of stations, no re-entrant loops;
Operation processing times: 2-Erlang distributed (mean: 1 time unit);
Inter-arrival times: exponential (mean: 0.648 time unit);

Table 4.1: Model characteristics

The methods are tested in the context of a job shop model with six workstations, each representing a single and unique source of capacity. Capacity is supposed to be invariable during the experiments. Each operation requires one specific workstation. The routing and operation processing time characteristics are known upon entry of the job. The routing length, determined as the number of operations per job, varies between one and six. A station is visited at most once in the routing of a job and all stations have an equal probability of being required in a job routing. For each station the operation processing times are identically and independently distributed according to a 2-Erlang distribution. Processing times are used as a reference measure with the average operation processing time being 1 time unit. The arrival rate of jobs is such that the stations have an average utilisation level of 90 percent.

To cover a wide spectrum of routings, two types of routing sequences have been modelled: (a) a randomly determined routing sequence resulting in undirected flows and (b) visits in order of increasing station number, resulting in directed flows. Routing type a results in a traditional pure job shop model, comparable to the one used in [Melnyk & Ragatz 1989]. Type b represents a shop with a dominant flow direction. Although the set of stations in the routing may still vary, stations are always visited in order of increasing station number (i.e. jobs visiting station 2 and 5 will always visit station 2 first). Consequently, stations with a low number have a more upstream position, and stations with high number have a more downstream position. Enns [1995] argues that this model, though indicated as a general flow shop, comes closer to the reality of job shop production. Also our observations in practice suggest that most shops have stations that perform typical preparative operations (such as sawing) and stations that perform typical finishing operations. Figure 4.1 gives an impression of the flows resulting from each routing structure. The thickness of the arrows roughly indicates the relative volumes of the job flows between the six workstations. The matrices give the transition probabilities between stations. The element \((r,s)\) of each matrix gives the probability that a job will move from station \(r\) to station \(s\), given that station \(r\) is part of its routing. The element \((0,s)\) gives the probability that a job leaving the pool enters the shop at station \(s\). The element \((r,0)\)
gives the probability that a job leaves the shop from station $r$, given that station $r$ is part of its routing.

![Diagram](image)

Figure 4.1: The job flows with (a) undirected routings and (b) directed routings

### 4.1.2 Entry

The entry stage gives the first opportunity to influence the flow of jobs. Controlled acceptance and the determination of due dates could smoothen the time-phased capacity requirements in this early stage.

As noticed earlier, a study of Melnyk et al. [1991] has investigated the influence of load smoothing in the entry stage. Their findings suggest that a pre-smoothed
workload alleviates the task of job release. In our studies we focus on the qualities of
the release methods. Therefore, it makes little sense to diminish the requirements on
controlled release. By simulating a Poisson arrival process, we test the release
methods under the difficult circumstances of uncontrolled job acceptance. The
amount of load imposed by arriving jobs is independent of the state of the shop. After
acceptance the jobs are collected in a pool with infinite capacity.

Also accurate due date setting alleviates the task of job release [Enns 1995,
Bertrand 1983]. Ideally, a due date should be assigned such that the release of the job
is planned at the time when it fits well in the workload norms. This would help to
avoid the negative influences of load balancing requirements on the timing of job
release (see figure 3.12). But again, it makes little sense to alleviate the release task
and to create interaction effects, while testing the qualities of release methods.
Besides, in many practical situations due dates are largely determined externally.
Therefore, we set due dates by just adding a random allowance to the job entry time
to create a variable level of urgency among jobs:

\[ \delta_j = t_{ej} + a, \]

with \( a \) uniformly distributed on \([m, M]\).

A minimum waiting time allowance (\( m \)) of 35 time units and a maximum (\( M \)) of 60
have been used. The minimum is just sufficient to cover a station throughput time of
5 time units for the maximum of 6 operations plus a waiting time before release of 5
time units. These values correspond to the planned station throughput time and the
release interval that will be used in the simulation study (see next subsection). The
value of the maximum waiting time allowance (\( M \)) is chosen such that the basic set
of experiments result in a percentage tardy between 5 and 20%.

4.1.3 Release

The release method is the main subject of our investigations. Both the type of
workload norm and the norm level are experimental factors in this study. Each
release method follows the same procedure without extensions to isolate the
influence of the norm type and the norm level.

The basic procedure to be executed at the beginning of each release period is as
follows:
1. For each job $j$ in the pool, a planned release date $t^R_j$ is determined by back scheduling from the due date $\delta_j$, using a planned throughput time of $T^D_s$ for all stations in the routing of $j$ (the set $S_j$),
that is: $t^R_j := \delta_j - \sum_{s \in S_j} T^D_s$.

2. The job $j$ with the earliest release date is considered first.

3. If the job fits the workload norms,
that is $L^H_{st} + d_{js} \cdot p_{js} \leq \Lambda^D_s \forall s \in S_j$ (for method A), or $L^\Lambda_{st} + p_{js} \leq \Lambda^\Lambda_s \forall s \in S_j$ (for method B),
then the job is selected for release and its load contribution is included,
that is $L^H_{st} := L^H_{st} + d_{js} \cdot p_{js} \forall s \in S_j$ (for method A), or $L^\Lambda_{st} := L^\Lambda_{st} + p_{js} \forall s \in S_j$ (for method B),
else the job must wait in the pool until the next release time.

4. If the pool contains any jobs that have not been considered yet,
then return to step 3 considering the job with the next earliest release date,
else the release procedure is finished and the selected jobs are released.

All variables and parameters in this procedure have been defined in section 3.3; the converted load $L^H_{st}$ is defined according to equation (3.8a), which means that the depreciation factor $d_{js} = 1$ when job $j$ is in the direct load of station $s$ at time $t$.

We use the norms $\Lambda$ as upperbounds for the workloads, which cannot be exceeded. Wiendahl suggests in [Wiendahl 1988] to allow release of the first job that exceeds the norm, and block the release of jobs that visit the station after the norm is exceeded. In that case the criterion for method A in step 3 would be:
$L^H_{st} \leq \Lambda^D_s \forall s \in S_j$.

However, experiments have shown that this alteration has not the suggested influence that average loads stay closer to their norm levels [See chapter 6]. Contrarily, after the norm of one station has been exceeded, the load balancing possibilities diminish with the consequence that other stations may remain far below their norm levels.

Disadvantage of the upperbounds in our simulation context is that theoretically we can never reach a steady state. The 2-Erlang distribution of processing time has no maximum. Thus jobs with a load contribution larger than the norm can never leave the pool. However, the probability that an operation processing time exceeds a norm $\Lambda$ can be calculated as $(1+2 \cdot \Lambda) e^{-2 \cdot \Lambda}$. For the simulated workload norms, this results in such a small probability that our performance measures will hardly be affected. Even for the smallest simulated norm level ($\Lambda = 7.4$), less than 6 operation processing times per million do exceed the norm.
A number of parameter values must be determined for each release method: workload norms and planned throughput times for each station, the release period length and a time limit. If all parameters would be included as experimental factors, this would lead to an immense experimental design. As we want to concentrate on the influence of the workload norms, we determined values for the other parameters by performing some test experiments. The accurateness of chosen parameter values will be verified in the sensitivity analysis of section 4.3.

Release is supposed to take place once every 5 time units ($T=5$). We use an infinite time limit (see section 3.4.3), which permits to consider all jobs in the pool for release. The planned throughput times ($T^{D_s}$) should normally be related to the workload levels, and consequently to the workload norms (see discussion in section 3.3.3). However, we will show in section 4.3 that small deviations from these values hardly affect performance. Therefore, we use a fixed planned throughput time of 5 time units for each station, which will be shown to be a reasonable value within the spectrum of considered workload levels.

The workload norms are the experimental variables. Each workload norm is built up from two components as shown in figure 3.8, a planned output component and a variable component for the planned workload buffer that must remain at the end of period. The planned output ($\rho_s T$) amounts to 4.5 time units per release period for each station. The variable component (respectively $L_s^{D_s}$ and $L_s^{A_s}$) starts at a level of infinity in the first experiment and is decreased in subsequent experiments. In the job shop with undirected flows we use equal norm levels for all stations. For the routings with directed flows, we also use equal norms in case of method A, but the norms in method B should relate to the considered station. Stations with a low station number will have lower aggregate loads than stations with a high number, because the latter are visited more downstream on average (see figure 4.1). Therefore, we derive the norm component for the remaining end-of-period load ($L_s^{A_s}$) from the average aggregate load level that is measured for each station $s$ when using infinite norm levels. We preserve the same ratio between the norms of the stations when lowering the norms. This will be discussed in more detail in 4.1.5.

### 4.1.4 Priority dispatching

At first instance jobs are processed FCFS at each station. The planned release dates are based on the assumption that all jobs have equal station throughput times, and FCFS dispatching will accommodate this assumption. Though interacting with the release method, priority rules may further improve the due date performance by correcting progress disruptions (see section 3.2.3). Therefore we also examine the influence of due date oriented dispatching rules. In this case, we use the operation
due date (ODD) rule, which is consistent with the use of planned release dates within the WLC methods. The operation due date is defined as the job due date minus the planned station throughput times for the downstream operations. Notice that if all jobs were exactly on planned schedule, the ODD rule would result in the same sequence as FCFS.

4.1.5 Performance measurement

The preceding subsections left us with four experimental factors. The factors and their experimental levels are summarised in table 4.2.

<table>
<thead>
<tr>
<th>factor:</th>
<th>experimental levels:</th>
</tr>
</thead>
<tbody>
<tr>
<td>routing sequence</td>
<td>undirected routings, directed routings</td>
</tr>
<tr>
<td>norm type (release method)</td>
<td>converted load (A), aggregate load (B)</td>
</tr>
<tr>
<td>norm level</td>
<td>stepwise down from infinity</td>
</tr>
<tr>
<td>priority rule</td>
<td>FCFS, ODD</td>
</tr>
</tbody>
</table>

Table 4.2: Experimental factors

A full factorial design has been used for the combinations of routing sequence, norm type and priority rule. For each combination, at least 10 norm levels (including infinity) have been simulated. However, each norm type will require its own norm levels. Also the routing sequence will influence the determination of norm levels. Therefore, the norm levels are based on the original (direct or aggregate) average load levels measured for the investigated combination of the routing sequence, norm type and priority rule at infinite norms. More precisely, the planned value for the average end-of-period-load is successively reduced by 15 percent relative to the previous value, starting at 175% of the original level that is measured at infinite norms.

In a given routing context and with a given priority rule, we want to be able to compare two release methods at a certain level of norm tightness. However, it is not possible to define norm tightness in terms of the norm levels themselves when norm types are different. Therefore, we use the average shop floor throughput time as an intermediate variable. We say that two norms of a different type are equally tight, if they result in the same average shop floor throughput time. As introduced in [Oosterman et al. 2000] (chapter 7), we will plot any performance measure against the measured average shop floor time. This results in a performance point for each simulation experiment. By connecting the points resulting from different norm levels, one can easily overview the performance of a norm type at different levels. Figure
4.2 shows an example for gross throughput time performance. The point of convergence at the right end of the curves depicts the performance at infinite norms.

![Performance representation for different levels of norm tightness](image)

**Figure 4.2: Performance representation for different levels of norm tightness**

In the example of figure 4.2 norm type X shows an overall better performance than type Y, as its curve remains uniformly below that of method Y. This means that in the example set of experiments method X results in smaller gross throughput times than method B at each level of norm tightness.

To determine the significance of performance differences between methods, we choose three levels of norm tightness for each combination of experimental factors. We perform additional experiments for these three norm levels. The levels are such that average shop floor throughput time is reduced to respectively 60%, 70% and 80% of its value at infinite norms. We determine the required norm values by interpolation from the original 10 experiments. The precision of the interpolation will be such that the average shop floor throughput time is within 1% of the desired value. Figure 4.2 indicates the three reference levels of norm tightness by dashed vertical lines and open marks on the performance curves (in the presentation of results we will show the vertical lines without added marks). The performance measures at the three reference levels are summarised in tables.

If the norms become too tight, the simulation may become less stable. Therefore, we discard experiments in which the job-average gross throughput time has increased with more than 40 percent relative to the case with infinite norm levels.

Basically we are interested in the lead time and due date performance at different levels of norm tightness. We could also say ‘at different levels of work-in-process’ as
our indicator of norm tightness (average shop floor throughput time) is proportional to the average number of jobs on the floor (see Little’s result in section 3.1.2). Lead time performance will be indicated by the job-average gross throughput time and due date performance will be indicated by the percentage tardy and the standard deviation of job lateness.

To investigate the influences of release methods on work-in-process in more detail, we record a number of load measures. To indicate the ability of methods to create low and steady levels of direct load we measure the time-average of the direct load (in processing time units) and its standard deviation. In the experiments with undirected flows we can restrict ourselves to the direct load of one station, since all stations will show identical load distributions in this symmetric shop. In the experiments with directed flows we record the direct loads for station 1, 3, and 6. A large number of other indicators of workload and workload distributions is recorded for detailed analyses. These indicators will be introduced when necessary.

Table 4.3 summarises the performance indicators.

<table>
<thead>
<tr>
<th>experiments with undirected flows</th>
<th>experiments with directed flows</th>
</tr>
</thead>
<tbody>
<tr>
<td>norm tightness/WIP level</td>
<td>norm tightness/WIP level</td>
</tr>
<tr>
<td>• job-average floor throughput time</td>
<td>• job-average floor throughput time</td>
</tr>
<tr>
<td>lead time performance</td>
<td>lead time performance</td>
</tr>
<tr>
<td>• job-average gross throughput time</td>
<td>• job-average gross throughput time</td>
</tr>
<tr>
<td>due date performance</td>
<td>due date performance</td>
</tr>
<tr>
<td>• percentage of jobs tardy</td>
<td>• percentage of jobs tardy</td>
</tr>
<tr>
<td>• standard deviation of job lateness</td>
<td>• standard deviation of job lateness</td>
</tr>
<tr>
<td>detailed workload control performance</td>
<td>detailed workload control performance</td>
</tr>
<tr>
<td>• time-average direct load (station 1)</td>
<td>• time-average direct load (station 1, 3, 6)</td>
</tr>
<tr>
<td>• direct load std. deviation (station 1)</td>
<td>• direct load std. deviation (station 1, 3, 6)</td>
</tr>
</tbody>
</table>

Table 4.3: Recorded performance measures

We are not interested in absolute values of performance measures, but mainly in performance differences between methods. These differences can be measured most accurately by giving each method the same set of jobs to deal with. Therefore, common random numbers are used, which reduces the variance across experiments. An experiment consists of 100 independent replications, each with a length of 6000 time units including a warming-up period of 3000 time units. The determination of the above parameters is discussed in the appendix of this chapter. When performance differences between two experiments are discussed in the following sections, the significance can be shown by a paired-t test at a 95% confidence level.

68 Chapter 4
4.2 Results

The results of the basic experimental design are subdivided into three parts. Subsection 4.2.1 gives a broad overview of the performance recorded in this simulation study. As mentioned in the introduction of this thesis, the aim is to understand influences and not just to measure them. Therefore, section 4.2.2 and section 4.2.3 provide an in-depth analysis of the results with respect to load balancing and timing.

4.2.1 Performance overview

The lead time and due date performance of the two release methods is depicted in figure 4.3. Each of the subfigures a, b, c and d represents a different combination of routing sequence and priority-dispatching rule. As explained in the preceding section, the norm tightness is indicated by the average shop floor throughput time, which is set on the horizontal axis. The job-average gross throughput time is set on the primary vertical axis, the percentage of jobs tardy on the secondary vertical axis. The performance points relating to the throughput time are indicated by closed marks, those relating to the percentage tardy by open marks. Connecting the performance points of the successive norm levels results in the curves for each performance measure. The three vertical gridlines respectively represent a 40, 30 and 20 percent reduction of the average shop floor throughput time relative to the infinite norm level (the point at the right end of the curves).

First we focus on the undirected routings and FCFS dispatching (figure 4.3 a). When the workload norms are tightened (going from right to left along the curves), both the gross throughput time and the percentage tardy first decrease and after a certain point the performance starts deteriorating for each of the methods. This point is reached much earlier for method B (aggregate load norms) than for method A (direct load norms). This means that method A shows a better performance than method B in combination with larger shop floor time reductions relative to uncontrolled release. Both curves of method A remain completely below the curves of method B, which means that method A shows a better performance than B at each level of norm tightness.

If we now turn to the directed routings (figure 4.3 b), the situation has reversed. Method B shows an uniformly improved performance in comparison with method A. While method B first shows an improved performance when norms are tightened, the gross throughput time and percentage tardy directly increase for method A.
Figure 4.3 a, b: Performance of release methods A and B with FCFS dispatching
Figure 4.3 c.d: Performance of release methods A and B with ODD dispatching
Combining the release methods with ODD dispatching (figure 4.1.c and d) results in performance reductions as soon as norms are tightened. The percentage tardy at infinite norms with ODD dispatching is already below the percentage that can be reached for FCFS dispatching. Again method A performs better than B in the case of undirected routings. But now the difference between the methods in case of directed routings is small and none of the methods performs uniformly better.

Tables 4.4 to 4.7 show a broader range of performance measures. In these tables the two methods have been compared at four norm levels, infinite norms and norms resulting in respectively 20, 30 and 40 percent reduction of the average shop floor throughput time.
The graphical representation already indicates a correlation between the percentage tardy and the gross throughput time. The tables 4.4 and 4.5 further show that the release methods (in combination with FCFS dispatching) may also reduce the standard deviation of lateness at certain norm levels. For instance, method A reduces the standard deviation from 19.1 at infinite norms to 17.3 when the shop floor throughput time is reduced to 40% of its original value.

In case of undirected routings, the average level of the direct load decreases nearly proportional to the shop floor throughput times. The standard deviation of the direct load decreases even stronger. This suggests that reductions of average gross throughput time can be attributed to load balancing, which will be analysed in more detail in subsection 4.2.2.
One remarkable difference between the load influences of method A and B can be seen in table 4.5. When release is unrestricted (i.e. the 0% columns), the directed routings result in relatively smaller direct loads at more downstream stations. This could have been expected since the coefficient of variation of the processing times is less than one. Now tightening the norms of method B causes the load to spread more evenly between upstream and downstream stations, while method A realises relatively smaller load reductions upstream. This is even more remarkable, as we used equal norms for all stations in method A. One could have argued to use even looser norms for more upstream stations, considering the load levels at infinite norms.

In table 4.6 we skipped the results of method B at 40% shop floor time reduction. This reduction caused such an increase of pool times that the stationarity of the simulation results must be doubted.

Table 4.6 and 4.7 show that the influences of norm tightening on the direct loads are similar for ODD and FCFS dispatching. But, as we already saw in figure 4.3, at the cost of gross throughput time increases. The tables show that norm tightening with ODD dispatching also causes the standard deviation of lateness to increase. This implies a negative interaction between the release method and the ODD dispatching rule. On the one hand the dispatching rule looses effectiveness when the release method reduces the choice of jobs in the queue, on the other hand the unplanned progress of jobs on the floor disturbs the effectiveness of balancing and timing within the release method.

4.2.2 Load balancing

In the previous chapter, we determined to focus particularly on the load balancing and timing qualities of the release methods. It has been discussed that the average gross throughput time, being the sum of pool time and floor time, is the basic indicator of load balancing qualities in simulations with a predetermined utilisation level. To realise the predetermined utilisation level, release methods with poor balancing qualities will require a larger choice of jobs in the pool than methods with good balancing qualities. A workload reduction may cause an increase of the average pool time that exceeds the decrease of the average floor time. This adds up to an increase of the gross throughput time. In that case we suppose a workload control method to lack sufficient balancing qualities for the required workload reduction.

The results presented in the previous subsection show that the balancing qualities of each method depend on the routing mix. Method A performs better for undirected routings and B for directed routings. The balancing qualities of the release methods appear to be sufficient to realise load reductions that make shop floor times decrease
with 40% to 50%, but only when the best of the two methods is combined with FCFS dispatching. Due date oriented dispatching appears to disturb load balancing heavily.

One may question whether the gross throughput time reductions of the release methods in combination with FCFS dispatching can be fully attributed to the balancing qualities of the release methods. It might be that the use of norms favours the release of small jobs at the cost of retardation for large jobs, small and large relating to processing times and routing lengths. Therefore, we measured the weighted-average routing length and processing times of jobs, using their pool times as weights. We compared these with the unweighted averages. The unweighed average routing length is exactly 3.5 stations and the average operation processing time is 1. A weighted-average processing time exceeding 1 indicates longer pool times for jobs with large processing times, and analogous reasoning holds for the routing lengths. Figure 4.4 shows the weighted averages at the different shop floor time levels for undirected routings and FCFS dispatching.

**Figure 4.4: Processing time and routing length influences on job release.**
Notice that the weighted averages must be equal to the unweighted averages for infinite norms, which is the case. Both measures show a slight increase when norms are tightened, though the increase is so small that it is not reasonable to expect strong effects on performance. According to figure 4.3 the area where the minimal gross throughput times are realised falls between 30 and 40% shop floor time reduction for method A and below 20% for method B. Particularly in these areas figure 4.4 gives no strong indication of favouring jobs with either small processing times or short routings. Thus, there is no reason to attribute the gross throughput time reductions to influences other than load balancing. Contrarily, one may say that workload control even harmonises gross throughput times across jobs with different routing lengths. Notice that the average shop floor throughput time of a job is proportional to the number of stations it must pass. When tightening norms, part of the shop floor time is replaced by a non-proportional pool time, which makes the gross throughput time of a job less dependent on its routing length. Since the due date allowance is independent of the routing length, this may give some advantage for the aspect of timing.

The average and the standard deviation of the direct load give a further indication of load balancing performance. The standard deviation shows to which extent the direct load is kept at a constant level. To realise the predetermined level of utilisation, any reduction of the average direct load must be accompanied by a reduction of the load variations. However, the more than proportional reduction of the standard deviation, as discussed in section 4.2.1, indicates that the load is kept relatively close to the average level. But more can be said about the reduction of load fluctuations.

To improve the understanding of the influences of norms on the direct load, we measured more detailed frequency distributions of the loads. As discussed in section 3.3.2, the release methods are designed to bring the load to the specified norm level by the end of each release period. Therefore, we recorded the loads at the end of each release period. Figure 4.5 shows the resulting frequency distributions (recorded load levels on the horizontal axis, frequencies on the vertical axis) for method A, FCFS dispatching and undirected routings. Norm levels are the same as those in table 4.4, respectively infinite and those leading to 20, 30 and 40% reduction of the shop floor time. For clarity only the distribution for infinite norms is depicted as a histogram, the other frequencies are connected by lines.
Figure 4.5: End of period load distributions related to table 4.4 (method A).

The leftmost frequency in figure 4.5 is that of a zero load, i.e. an idle station, for which we reserved a separate class. The frequency recorded here is higher than the average degree of idleness (10%) because we record it at the end of each release period. The second class represents the observations in the interval (0, 0.5], the third class the observations in (0.5, 1.0], etc. Notice that the class of (0, 0.5] requires at least one job being processed. It contains only few observations because of the 2-Erlang distribution of processing times.

The ‘natural’ distribution of the load at infinite norms has only one mode strongly at the left of the distribution (beside the peak at exactly zero load). The other distributions show how tightening the norms reduces the variability of load. As soon as a finite norm is introduced the distribution shows a double mode, with the second top close to the norm level applied (respectively 9.6, 7.0 and 5.2 for direct load component of the norm $L^{D}_{st}$, see section 3.3.3 ). Notice also that the original mode does not disappear. It gives the impression of a sand dune being shoved from the right, but with a hurdle at the norm level, which hinders the sand in moving too far left. The tightening of norms reduces the long tail of the distribution for infinite norms, which explains the reduction of the standard deviation of the direct loads. Still the load regularly exceeds the planned value for the end-of-period load $L^{D}_{st}$. However, the loads mainly vary between zero and the norm level. This variability is inherent in the use of norms. The utilisation level of 90% forces the load to return to the zero level for 10% of the time; when possible, the release method filling the
norms forces the load to increase to the norm level. This limits the opportunities for realizing a steady direct load by means of workload norms.

### 4.2.3 Timing

The timing qualities of a release method are particularly indicated by the standard deviation of lateness. This standard deviation is affected by two aspects: (1) whether jobs are released according to their planned release dates, and (2) whether the actual shop floor throughput times correspond with the planned value. To isolate these aspects, we compare the final standard deviation of lateness with the standard deviation of the lateness at the time of release. This ‘release lateness’ $\Delta R^j$ is measured by subtracting the planned release date $t^*R^j$ from the realized release time $t^R_j$ (i.e. $\Delta R^j = t^R_j - t^*R^j$), while (final) lateness $\Delta j$ is measured by $\Delta j = \delta_j - t^Z_j$.

A high standard deviation of the release lateness indicates that the workload norms do not allow releasing the jobs relative to their planned release dates. An increase of the standard deviation of final lateness over that of release lateness indicates that insufficient control of throughput times influences the timing performance, assuming that the planned station throughput time corresponds with the recorded average station throughput time.

Figure 4.6 shows the comparison between final lateness and release lateness for the experiments with method A and undirected routings. Starting with figure 4.6a for FCFS dispatching, we observe that at infinite norms the standard deviation of the final lateness is much larger than the standard deviation of release lateness. This implies that, at infinite norms, the required progress of jobs on the shop floor is strongly disturbed by the fluctuating queue lengths. When we tighten the workload norms, the standard deviation of release lateness increases, while the standard deviation of final lateness weakly decreases. The increasing standard deviation of release lateness indicates that balancing increasingly hinders the timing of job releases. But according to the decreasing standard deviation of final lateness, this appears to be more than compensated by the less fluctuating queues on the floor. Thus, we can conclude that the timing qualities of the release method mainly result from controlled throughput times, rather than from the release sequence.
Figure 4.6: Standard deviations of final lateness and release lateness

With ODD dispatching (figure 4.6b), the release method is not able to cause a reduction of the standard deviation of final lateness. Applying a due date oriented dispatching rule on the shop floor reduces the standard deviation of lateness strongly.
at infinite norms. The reduction appears to be larger than can be realised by the timing capabilities of the release method.

Observe that in figure 4.6b both standard deviations increase, when norms are tightened. Apparently the controlled throughput time do no longer compensate for the disturbed release sequence. Nevertheless, the standard deviation of final lateness increases less rapidly than the standard deviation of release lateness, when the norms are tightened. Thus, the control of the throughput times on the shop floor still improves at lower norms, but is insufficient to compensate for the disturbed release sequence. The timing of job releases seems to be hindered too much by load balancing requirements.

We confined ourselves to depicting the results of method A with undirected routings. Other observed results are nearly similar. The main difference between method A and B with undirected routings is that the minimum for method B is reached at a higher level of shop floor time. For directed routings, the performance pattern of method B is roughly comparable to that of method A for undirected routings. With directed routings, method A has its minimum standard deviation of final lateness always at infinite norms, even in case of FCFS dispatching.

4.3 Sensitivity

Four factors have been varied in the basic experimental design: the routing sequence, norm type, norm level and priority dispatching rule. Within the design several other factors have been set as constants. Among these are the utilisation level, which results from the chosen arrival rate, and the other parameters of the release methods. To assess the adequacy of the applied values and to deepen our understanding of their influence, we perform an additional analysis in which we vary these parameters. Unless specified differently, the sensitivity analysis is performed for one combination of the basic experimental variables, which is seen as a reference: undirected routings, release method A and FCFS dispatching. Still, we vary the workload norms. Each of the following subsections presents one investigated factor.
4.3.1 Utilisation level

Within the basic experimental design, the release methods have been tested at a utilisation level of 90%. To give an indication of the influence of utilisation requirements, the reference experiments are repeated for utilisation levels of 88% and 92%.

In general, we observe the same kind of performance patterns at the higher and lower utilisation level. Naturally, gross throughput times and the percentage tardy increase with increasing utilisation levels. But the basic effects of norm tightening do not alter. The most interesting influence of the utilisation level can best be seen in figure 4.7. In figure 4.7 the average shop floor throughput time and gross throughput time are normalised, dividing the values by their respective values at infinite norms. On each axis the normalised values are expressed as a percentage. We can see that the relative influence of norm tightening is exactly the same at each utilisation level, except for the fact that a higher utilisation level allows for a higher percentage reduction of the shop floor throughput time before the performance worsens.

*Figure 4.7: Influence of the utilisation level*

Similar figures can be constructed for indicators of due date performance. These are not depicted as they show the same pattern.
4.3.2 Planned station throughput time

All experiments in the basic design use a planned station throughput time of 5 time units. This parameter affects the relative priorities of jobs at release, and in case of ODD dispatching it also affects the relative priorities of jobs on the floor. An increase of the planned station throughput time will cause jobs with a relatively high number of operations to be considered earlier for release, and jobs with a relatively high number of remaining operations to be processed earlier.

According to Little’s result (section 3.1.2) the average throughput time is related to the average workload level. When the workload norms are tightened and the actual throughput times are reduced, one would normally decrease the planned station throughput times (see section 3.3.3). Notice now that the planned station throughput time of 5 time units corresponds with an average shop floor throughput time of 17.5, since the average number of operations per job is 3.5. This means that the planned station throughput time used in the basic experimental design corresponds with a shop floor throughput time reduction of about 30% for the directed routings and slightly more for the undirected routings. As such, it is in the middle of the simulated spectrum.

To analyse the influence of this parameter choice, we have repeated our experiments for undirected routings with planned station throughput times of 3, 4, 6 and 7 time units. These values correspond to average shop floor throughput times of respectively 10.5, 14, 21 and 24.5, so they cover the full range of simulated shop floor time reductions. The main results of these experiments are shown in figure 4.8. For clarity, only three curves are depicted.

Figure 4.8 shows the influence of the parameter on the standard deviation of lateness, which indicates timing qualities. Low planned values should be more appropriate for low average floor times, i.e. to the left of the figures. High planned values should be more accurate to the right. However, performance appears to be almost equal for all parameter values, and also to the left the higher planned values even lead to slight improvements. A possible explanation for this performance pattern can be derived by comparing figure 4.8 with figure 4.4. Tight norms hinder the release of jobs with long routings, which could be concluded from the slightly increased weighted routing length in figure 4.4. This can be compensated by increasing the priority of jobs with longer routings, which is exactly established by the use of overestimated planned station throughput times. The result is the counterintuitive good performance of high planned throughput times at tight norms. As relative influences are small, this conjecture is not tested further.
Since the planned station throughput time is also the parameter that determines operation due dates in the ODD dispatching rule, we also tested the influence of the parameter in combination with ODD dispatching. Figure 4.9 shows the timing performance. Again the influences are small. We see the same effect as with FCFS dispatching that high planned values perform slightly better at low actual throughput times. In addition we now observe the contra-intuitive result that low planned values now perform slightly better at high actual values. A possible explanation can be found in the progress pattern of jobs (see also [Enns 1994]). The use of ODD with a lower parameter value decreases the priority of a job in the beginning of its routing gives a relatively high priority to a job as it approach the end of its routing. This reduces ‘timing risks’, as closer to the end of a routing the possibilities decrease for downstream compensation of delay. The reduction of these risks is particularly important for the more strongly fluctuating queues at high norm levels (see section 4.2.2). The uneven progress pattern can be concluded from figure 4.10.
Figure 4.9: Planned station throughput time influences (ODD dispatching)

Figure 4.10 compares the upstream and downstream loads of a station for planned station throughput times of respectively 3, 5 and 7 time units, using ODD dispatching and undirected routing. For the lowest parameter value the upstream load exceeds the downstream load across the full range of simulated shop floor throughput time levels. For the highest parameter value we observe the opposite. These different load distributions result from the progress patterns of jobs. A low planned throughput time suggests no urgency in the beginning of the routing. The consequence is that jobs move faster near the end of their routing, resulting in lower downstream loads, while upstream loads get higher. As long as the realised throughput times strongly exceed the planned value, jobs move on average slowly through the first part of their routings and are speeded up in the second part. The planned station throughput time of 5 results in a more even load distribution in the area around the corresponding shop floor throughput time of 17.5, which suggests an even progress pattern of jobs.
Figure 4.10: Planned station throughput times and job progress patterns
4.3.3 Length of the release period

Where performance is rather insensitive to the value of planned station throughput times, the length of the release period $T$ appears to be a more delicate parameter choice. In our steady-state simulations, the average amount of newly released workload must be equal to the average output $\rho T$. For the release period length $T$ of 5 time units, combined with the utilisation level $\rho$ of 90%, 4.5 units of workload will on average be released for each station. The average available room between the end-of-period load and the workload norm will even exceed these 4.5 units of workload. The probability that an operation processing time exceeds 4.5 time units is 0.0012. Therefore, a job will nearly always be released when it becomes the most urgent job in the pool. A release period shorter than 5 will speed up the release of small jobs, at the cost of large pool delays for large jobs. Experiments have been repeated with release period lengths varying between 1 and 9 time units.

Figure 4.11 shows the influence of this parameter on the gross throughput time. Clearly, a longer release period always leads to higher delay in the gross throughput times at infinite norms, since jobs just have to wait until the end of a release period before being released. But also relative to the situation at infinite norms, a shorter release period leads to stronger gross throughput time reductions when norms are tightened. Thus, gross throughput time results seem to plead for a short release period.

However, it must be concluded from the standard deviations of lateness in figure 4.12 that the timing of release may worsen, if the release period is set below a level of 5 time units. As discussed above, this can be explained from the decreased probability that large jobs can be released timely in case of a shorter release period. This leads to a more uneven distribution of due date deviations among jobs for shorter release periods.

The preceding explanation is supported by the results in figure 4.13, which shows the weighted-average processing times and routing length of jobs using their pool times as weights (comparable to figure 4.4). When norms are tightened, jobs with large routing lengths and processing times appear to get relatively larger waiting times in the pool, routing lengths particularly for release period lengths of 1 and 3 time units.
Figure 4.11: Release period: influences on gross throughput times

Figure 4.12: Release period length: influences on standard deviation of lateness
Figure 4.13: Processing time and routing length influences for different $T$
4.3.4 Time limit

The time limit restricts the set of jobs in the pool that can be selected for release. Only jobs with a planned release date that falls within the time limit are considered. Introducing a time limit should improve the timing of release at the cost of decreased balancing possibilities. The basic experimental design used an infinite time limit. Time limits of 10, 20 and 30 time have been additionally included to test sensitivity.

Figure 4.14a shows its influence on timing, as indicated by the standard deviation of lateness. We see that a modest time limit improves the standard deviation of lateness. When the time limit is decreased to 10 time units (only jobs with a planned release falling within the next 2 release periods are considered for release) performance worsens for larger shop floor time reductions.

Figure 4.14b shows the influence of the time limit on balancing, as indicated by the gross throughput time. As might be expected from the decreased balancing possibilities, gross throughput times increase for smaller time limits. However, the negative influences on the average gross throughput time appears to undo the positive influences on the standard deviation of lateness, if we consider the resulting percentage of jobs which is tardy (see figure 4.15). This explains the choice for the infinite time limit in the basic experimental design.
Figure 4.14: Time limit influences on timing and balancing
4.4 Summary and assessment of results

The simulation study in this chapter has been performed to answer three crucial questions:

1. How do the classical WLC release methods perform with respect to load balancing and the timing of job release?
2. What is the influence of the norm type and the norm levels on the performance and how does this influence relate to routing variety?
3. How sensitive is the performance to other factors, particularly the parameters of the release methods?

The simulation results show that the release methods do have capabilities to balance loads. With the best performing norm type, the average shop floor time could be reduced with about 30 to 40 percent relative to unrestricted (periodic) release before gross throughput time started to increase. Checks in order to exclude other possible influences on the gross throughput times indicate that the above results can be attributed to load balancing. Contrarily, the frequency distributions of realised load levels show that there are still opportunities for improvement. Although the standard deviation of direct loads reduces significantly when norm are tightened, there are clear fluctuations between a zero load level imposed by the inevitable idle

Figure 4.15: Time limit influences: the percentage of jobs tardy
time of stations and the level imposed by the workload norm. This phenomenon will be studied in more detail in the next chapter.

The simulations also show that the classical WLC release methods do contribute to timing performance, as indicated by a reduced standard deviation of lateness. This timing performance appears to result mainly from controlled station throughput times. The planned sequence to release jobs according to their relative urgency, which is based on these controlled throughput times, is disturbed as soon as norms are tightened. Besides, the release methods interact poorly with a due date oriented priority-dispatching rule. Both aspects deserve further attention within chapter 5.

As expected from the analysis in chapter 3, the functioning of the different norm types depends on the routing characteristics of jobs. Method A using norms for estimated direct loads performs well for high routing variety, but is less suitable for handling directed job flows with typical upstream and downstream routings. Method B using norms for aggregate loads shows the opposite qualities. The problems typical result in a reduced scope to tighten the norms. Performance starts to deteriorate at lower levels of shop floor time reduction. The opposite qualities of the two methods provide a challenge to combine the strengths of both methods.

The results of the sensitivity analysis confirm that the release methods are also sensitive to parameters other than the norm type and norm levels. The length of the planned station throughput time appears of less influence. Planned values both longer and shorter than realised average throughput times show certain advantages. Consequently, the net effect of overestimating and underestimating throughput times is negligible. In the basic experimental design we used an infinite time limit, which allowed considering each possible job for release. Though timing performance is slightly improved by restricting the set of considered jobs to the more urgent ones, the resulting reduction of balancing opportunities appears to be critical. Likewise, the choice of an appropriate release period length is shown to be a delicate decision. A long release period delays jobs before release and results in increased gross throughput times, while a short release period hinders the release of large jobs, large regarding routing length and/or processing time. In order to improve the robustness of the release methods the latter point should be addressed as well in the redesign.
Appendix chapter 4: simulation parameters

The simulations have been performed in Desimp, a simulation tool for Pascal and Delphi, developed by E.J. Stokking at the University of Groningen. Each experiment consists of 100 runs of 6000 time units, all runs including a warming-up period of 3000 time units. The common random number technique has been used to reduce the variance among experiments. The same warming-up period, the same run length and the same number of runs are used in all experiments to enable the use of these common random numbers.

The warming-up period for each run is based on the required time to reach a state that can be considered as steady state. Several methods have been proposed to determine an exact length for the warming-up period (see e.g. [Law & Kelton 1991]). Most of these methods focus on efficient use of computer time. However, computer time is not the restrictive resource anymore. We determined a suitable length of the warming up period by performing test experiment with a very high number of runs (i.e. 2500). We measured the value of performance indicators at fixed time intervals (and always at the exact end of a release period) during each run. The level of the indicator (averaged across the 2500 runs) was plotted against time. Figure 4A.1 shows such a picture of all subsets of loads (in processing time units of station 1) for undirected routings, norm type A, FCFS dispatching and a norm level resulting in about 35% floor time reduction.

Within WLC the pool size acts as a mechanism to reach a stationary state: a larger pool should allow for sufficient balancing to reach the predetermined utilisation level. Thus the warming-up period could end at the time where a steady-state level of the pool load seems to be reached. Loads on the shop floor are controlled by WLC and as such these loads show less fluctuations. These loads cannot reach a steady state level before the pool load has reached its steady state, because more load in the pool will enable the release decision to bring the workloads closer to their norm levels. Thus, other loads can be used as indicators too.
Figure 4A.1: Transient pattern of load levels (average of 2500 runs)

The decision on the warming up period could be formalised by for instance using the average of the last $x$ values to estimate the steady state level (see dashed line in figure 4 A.1) and then determining the first time after which this level remains within the confidence intervals for a fraction $y$ of time. The 95% confidence intervals around the pool loads are included in the figure (though hardly observable, as the intervals are small for this experiment). However, such an approach would suggest an exactness which is not reasonable. We would come to a period in-between 1000 and 1500 time units. But our warming-up period has to include a significant buffer, since it should be based on the worst of all experiments. The worst case is hardly possible to predict beforehand, as we do numerous strongly different experiments. Considering the speed of existing computers it makes little sense to spend much time in finding the shortest possible warming-up period to reach a certain accurateness, with the risk of having to redo all experiments if the level appears to be too low for just one of the experiments. Therefore, we use a warming up period of 3000 time
units. We still verified whether the warming up period was sufficient for some extreme experiments.

Figure 4A.2 shows an example of an extreme experiment. Settings are similar to those in figure 4A.1, only the workload norm is decreased to result in more than 55% shop floor time reduction, which is outside the ranges discussed in this chapter. With such a tight norm the transient period increases strongly, but even here the warming-up period of 3000 time units seems sufficient.

![Figure 4A.2: Transient pattern of load levels (average of 2500 runs): extreme exp.](image)

The run lengths can be specified in terms of either a number of jobs or a time-period. Our simulation environment best supports the use of time-periods. Regarding the job-related statistics, this requires a further decision whether to include jobs that arrived or jobs that have been completed in this period. The use of common random numbers strongly favours the inclusion of completed jobs. Otherwise runs would have to be extended for an indefinite period until all jobs to be included have been
completed. Under the assumption that a steady-state has been reached, the approaches should not result in significant differences.

There is a trade-off between this run length and the number of runs. A shorter run will require a larger number of runs. We decided to measure all completions between time 3000 and 6000. Given this length of the run, the number of runs must be sufficiently high to prove the significance of a difference between two experiments (using a pair-wise T-test) for each of the considered performance statistics. We want to be able to prove the significance of differences at a 95% confidence level, as soon as the measured difference for a performance statistic between two experiments is more than 5%. Also here a significant buffer must be included, because it is difficult to predict which pairs of experiments will result in low or high T-values. Again we preferred low risk to efficiency and performed 100 runs per experiment. As a consequence the results across 300 000 time units, resulting in about 460 000 job completions, were included in each experiment.