2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

Robotic Cycle Shop Control based on deterministic Correlation Maps

Wolfgang Meyer and Claudia Fiedler

***Abstract*— We present a deterministic scheduling approach for robotic cycle shops based on a n-dimensional collision map. The map facilitates the optimization of periodic and non periodic robot schedules. The method rests upon the finding that the release intervals of processes sent to the shop and the auto correlation lengths of robot and machine operations inside the shop are identical. For zero correlation, collisions of robot operations are avoided as well as collisions on any multipurpose or loop machine.**

***Keywords* — Manufacturing, Periodic Scheduling, Robot Control, Collision Avoidance, Discrete Event Systems**

I. Problem Statement

A *cycle shop* is the extension of a flow shop where all jobs (or processes P) obey the same sequence of operations on the processing machines, but in contrast to a flow shop, some operations can be repeated on some (multi purpose) machines a number of times. - We now insert an additional operation on an additional common machine between any two consecutive operations in each process. In general, this common server is called input-output resource R as contrasted to a processing resource [1]. In manufacturing, R is the pickup-delivery resource, the robot which performs transport and loading operations among different machines. A *robotic cycle shop*, therefore, contains at least two loops [2]: a multiple loop for the robot, and one or several loops at one machine. Fig. 1a shows the resource Gantt-chart of a rank 2 (number of loop machines) and multiplicity 7 (maximum number of loops per machine) process P1(t). In this chart, the operations r(t), m1(t), m2(t), … which are processed by the resources R, M1, M2, … , are arranged along the time axis according to the process plan (Table 1). Here, the robot is used for loaded transport only. The robot routing is predetermined and empty moves do not take place.

The situation gets more complicated when a second process P2(t), either identical or different from P1(t), is released to the shop (Fig. 1b). Now the robot is forced to move back and forth unloaded to cope with the transport needs of two processes. Not only that more and more transport operations must be coordinated by the robot controller, depending on the degree of overlap among P1 and P2. Now, additional (empty) move operations must be squeezed into the uppermost row of Fig. 1b without collisions which means without overlap of operations. In this context, the *robotic cycle shop control* problem as dealt with in this paper is formulated as follows: For identical processes released to the shop with release intervals v_i , find the stationary schedule of robot operations which maximizes the shop productivity or, equivalently, minimizes $\Sigma\ v_i$ (Fig. 1c).

Table 1. Process plan P1 (small loop)

resources	M1	M2	M3	M4	M5	M4	R	
operations	m1	m2	m3	m4	m5	m4	tr	mo
processing times	212	191	402	127	127	127	21	10

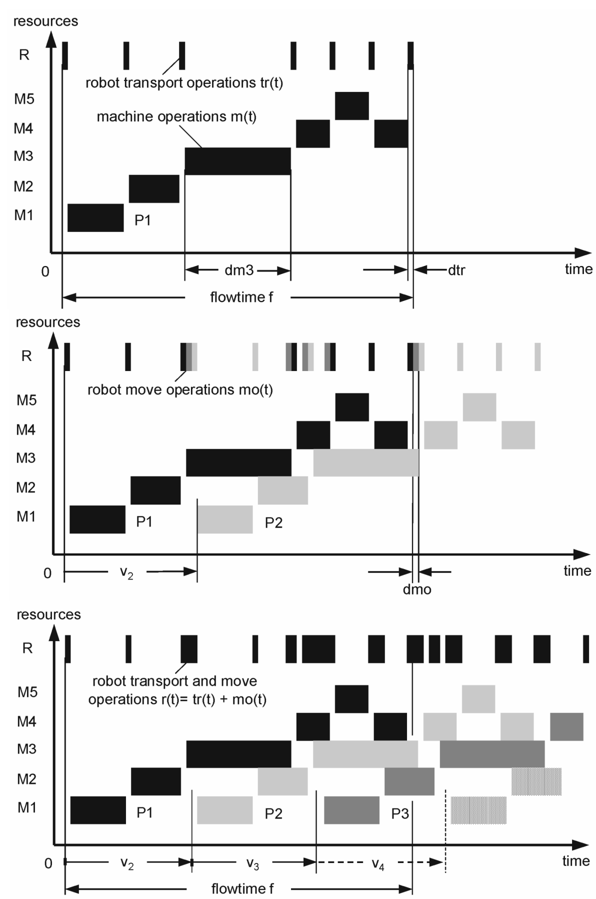

Fig.1. Resource Gantt-chart for a 5 machine robotic cycle shop. (a) Process plan P1(t). dm durations for machine operations ; dtr durations of robot transport operations (b) Two process Gantt-chart P1(t), P2(t). v_2 process release interval for P2. dmo duration of robot move operations (c) Gantt-chart for three processes P1(t), P2(t), P3(t)

Manuscript received January 22, 2007.
Wolfgang Meyer and Claudia Fiedler are with the Institute of Automation, Hamburg University of Technology, 21071 Hamburg, Germany
(e-mail: w.meyer@tu-harburg.de; claudia.fiedler@tu-harburg.de)

Obviously, a collision free schedule depends on the proper choice of release times v_i , i standing for the number of processes being released to the shop. In Fig. 1c, v_2 and v_3 have been chosen in such a way as to optimize the plant productivity. It happens that $v_2 = v_3$ here which is not necessarily true in all cases. Neither is there any guaranty about the schedule's periodicity for $i > 3$. What is needed then is a systematic approach for deriving deterministic schedules for coupled discrete processes from first principles, without search. In this paper, we present a theory how to transform deterministic process plans expressed as Gantt-charts into autocorrelation functions and how to construct periodic and non-periodic schedules from the integrated auto correlation maps for robot and machine operations. We elaborate on a representation for collision avoidance recently put forward by us in [3] and extend it to robotic cycle shops including empty moves and multi purpose machines. In our applications, no buffers exist at machines and the robot is not allowed to wait anywhere when being loaded.

II. Robotic Scheduling: State of the Art

Scheduling as a problem solving activity copes with the constraints of the real world by *sequencing* operations and *allocating* resources to operations in such a way as to optimize some performance measures. The robotic scheduling problem is a general shop scheduling problem with special constraints; for a classification and most recent bibliography see [2]. It is already NP-hard for robotic flow shops with more than two machines and for two or more different processes [1] not to speak of robotic cycle shops. Some practical work on robotic cycle shops has been done in the context of hoist scheduling for automated electroplating lines. For an extensive bibliography see [4]; for a recent complex application including important references see [5].

Most schedulers rely on the IP (integer programming) or CP (constraint programming) paradigm or a combination of both, often based on some kind of prohibited-interval rule [6]. Simulation, heuristic and sometimes exhaustive search have been used as well [7]. However, no constructive analytical tool exists for real size NP-hard scheduling problems, of course. In support of short-term planning and look-ahead algorithms, therefore, deterministic correlation theory has been applied for robotic flow shop scheduling [3]. We extend this theory to robotic cycle shops in the following.

III. Basic Idea of Correlation Scheduling

In developing the combined schedule P1+P2 for the running example (Fig. 1b) from the process plan P1 (Fig. 1a), we intuitively performed this reasoning: Create P2 by duplicating P1, and shift P2 to the right along the time axis until no more overlaps exist for any of the black and grey bars for all resources including the robot. Result: the minimum shift or release time v_{min} to avoid parallel operations on each of the (limited) resources. - This reasoning mechanism is mathematically modelled by the collision function $CO_{12}(v)$ [3] :

$$CO_{12}(v) = \int_{-\infty}^{+\infty} P1(t) \cdot P2(t - v)dt \qquad (1)$$

The functionality of Eqn. 1 is visualized with the help of Fig. 2: (a) P2 is duplicated from P1; (b) P2 is shifted to the right according to three different values of v; (c) P1(unshifted) and P2 (shifted) are multiplied; (d) the overlap among P1(t) and P2(t-v) is quantified by integration; (e) for three shift values including v=0 and $v = v_{min}$, $CO_{12}(v)$ is drawn point wise along the v-axis. - Thereby, the collision function measures the overlap of P1 and P2. For $CO_{12}(v) = 0$, P1 and P2 do not interfere. With other words, for the simple example of Fig. 2, every schedule $v \geq v_{min}$ is free of collisions.

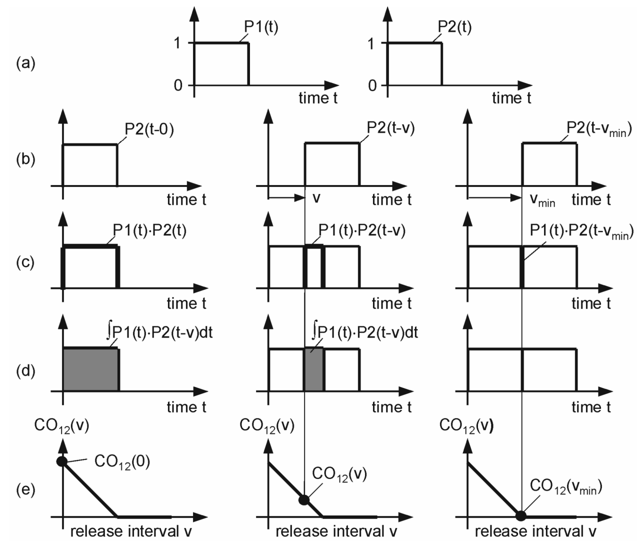

Fig.2. Collision function $CO_{12}(v)$ for three values of the process release interval v.

For identical processes P1=P2, Eqn. 1 simplifies to

$$CO_{12}(v) = \int_{-\infty}^{+\infty} P1(t) \cdot P1(t - v)dt \qquad (2)$$

By transforming t-v=λ, dt=dλ in Eqn. 2 and renaming λ by t afterwards, we arrive at Eqn. 3 which is the well known auto correlation function $AC_{12}(v)$ [3]:

$$AC_{12}(v) = \int P1(t) \cdot P1(t + v)dt \qquad (3)$$

Therefore, for robotic cycle shops fed by identical consecutive processes, the collision function Eqn. 1 can be replaced by the auto correlation function Eqn. 3. The set of release intervals v_2 for the second process which guaranty collision-free behaviour, is $\mathbf{V_2}$ (Eqn. 4) whereas the set of forbidden intervals is the complement $\mathbf{V_{2^-}}$ (Eqn. 5).

$$\mathbf{V_2} = \{v_2 \,|\, AC_{12}(v_2) = 0\} \qquad (4)$$

$$\mathbf{V_{2^-}} = \{v_2 \,|\, AC_{12}(v_2) \neq 0\} \qquad (5)$$

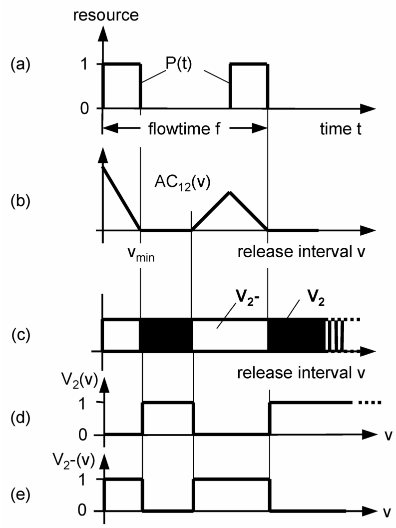

Fig.3. Set representation of the auto correlation function $AC_{12}(v)$. (a) Process plan (b) Auto correlation function of P(t) (c) Set representation of allowed ($\mathbf{V_2}$) and forbidden ($\mathbf{V_2}$-) release intervals (d), (e) Functional representation $V_2(v)$, V_2-(v) of sets $\mathbf{V_2}$ and $\mathbf{V_2}$-

Both sets are illustrated in Fig. 3 for a small loop shop with one machine only. This figure is a nice illustration of a general problem solving paradigm: The difficult (scheduling) problem as it occurs in time space t (Fig. 3a) is transformed into the interval space v (Fig. 3b) and solved now by simpler algorithms based on the new representation (Fig. 3c). The new representation is tailored to the specific scheduling problem and specialized in such a way as to shed some light to the problem from one angle and leaving others in the dark. Scheduling models based on correlation functions are indeed very compact representations which hide some information related to sequence dependent constraints, see Section VII.

IV. PROCESS AUTO CORRELATION FUNCTION

In Fig. 2, two very simple processes were treated. We now return to our running example, the robotic cycle shop and its process plan P1 in Fig. 1a. Each of the six machine processing operations and seven robot transport operations of P1 (neglecting moves for the moment) may lead to a collision at the respective resource when confronted with the second process P2 entering the shop. Therefore, the no-collision condition $AC_{12}(v)=0$ must hold for each individual resource for a valid release time v. With other words, the auto correlation function for the complete process AC_{12} is built from autocorrelations for each individual resource M1, M2, … and for the robot R:

$$AC_{12}(v) = AC_{M1}(v) + AC_{M2}(v) + \ldots + AC_R(v) \qquad (6)$$

$$\begin{aligned} AC_{M1}(v) &= \int m1(t)\cdot m1(t+v)dt \\ AC_{M2}(v) &= \int m2(t)\cdot m2(t+v)dt \\ &\ldots \\ AC_R(v) &= \int r(t)\cdot r(t+v)dt \end{aligned} \qquad (7)$$

In Eqn. 7, m1(t), m2(t), ... , r(t) are the time functions of the operations taking place at the respective resources as prescribed by the process plan (Fig. 1a). The sets of admissible release intervals per resource are

$$\begin{aligned} \mathbf{V_{2M1}} &= \{v_2 \mid AC_{M1}(v_2)=0\} \\ \mathbf{V_{2M2}} &= \{v_2 \mid AC_{M2}(v_2)=0\} \\ &\ldots \\ \mathbf{V_{2R}} &= \{v_2 \mid AC_R(v_2)=0\} \end{aligned} \qquad (8)$$

The combined set $\mathbf{V_2}$ of admissible release intervals v_2 for the complete process P1(t) is the intersection of the individual resources from Eqn. 8:

$$\mathbf{V_2} = \mathbf{V_{2M1}} \bigcap \mathbf{V_{2M2}} \bigcap \ldots \bigcap \mathbf{V_{2R}} \qquad (9)$$

Similar considerations hold for the complement $\mathbf{V_2}$- which is the union of forbidden release intervals v_2 for each individual resource:

$$\mathbf{V_{2^-}} = \mathbf{V_{2^-M1}} \bigcup \mathbf{V_{2^-M2}} \bigcup \ldots \bigcup \mathbf{V_{2^-R}} \qquad (10)$$

Eqns. 8, 9 and 10 are illustrated in Figs. 4 and 5 for four fundamental structures of machine operations which constitute the Gantt-chart of Fig. 1 (and any Gantt-chart). These are: a single operation on one machine, one or more loops of different width on one machine, and the combination of several machines into more complex processes. Finally, Fig. 6 shows the sets of possible and forbidden release intervals for the example process from Fig. 1 and Table 1 as they develop from the machine correlations according to Eqns. 6, 7 and 9.

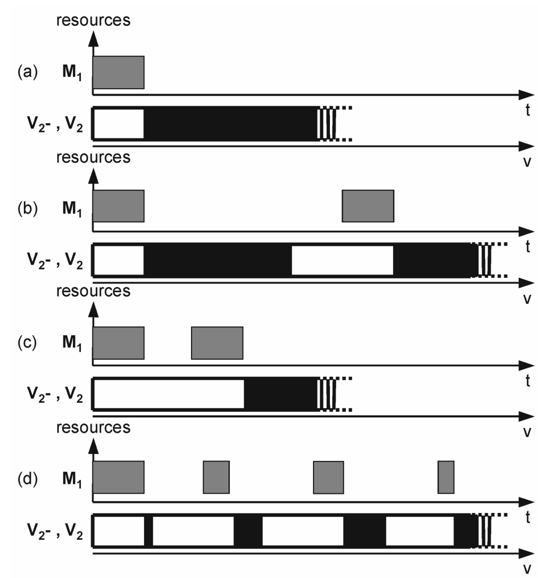

Fig.4. Resource Gantt-chart for single machine processes and respective sets of allowed ($\mathbf{V_2}$; in black) and forbidden ($\mathbf{V_2}$-; in white) release intervals. (a) Single operation (b) Single loop (c) Short loop (d) Three loops of different widths

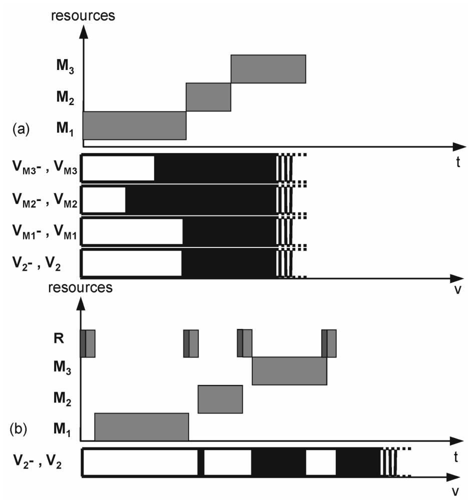

Fig.5. Resource Gantt-chart for multi machine processes and respective sets of allowed (**V_2**; in black) and forbidden (**V_2-**; in white) release intervals. (a) Job shop with 3 operations at 3 machines (b) Robotic cycle shop with 3 machine, and 4 move and transport operations of the robot

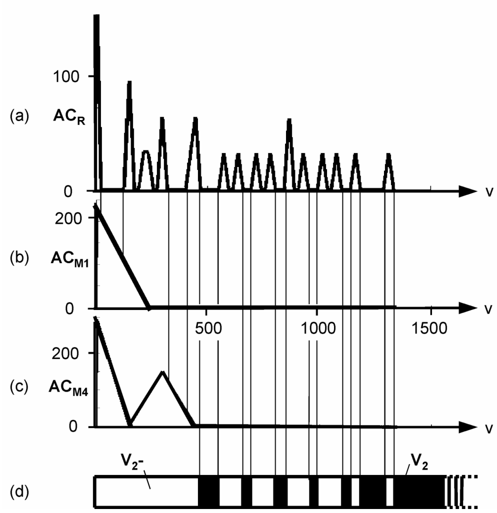

Fig.6. Auto correlations AC of machine operations for process P1 (Table 1). (a) AC of robot R (b) AC of machine M1 (c) AC of machine M4 (d) Allowed (**V_2** ; in black) and forbidden (**V_2**- ; in white) release interval sets for the second process

V. COMBINED AUTO CORRELATIONS

In the preceding sections, we only considered two processes being sent to the system. As indicated in Fig. 1c, we now release a third one of the same kind, P1(t). With the release interval v_3 of the third process as variable, and with $v_2 \in \mathbf{V_2}$ (Eqn. 4) as parameter, the collision function from Eqn. 2 now reads as

$$CO_{123}(v_2, v) = \int [P1(t) + P1(t - v_2)] \cdot P1(t - v_2 - v) dt \quad (11)$$

The set of collision-free release intervals for the third process is $\mathbf{V_3}$ [3]:

$$\mathbf{V_3} = \{ v_3 \, | AC_{12}(v_2) = 0 \wedge AC_{12}(v_3) = 0 \wedge AC_{12}(v_2 + v_3) = 0 \} \quad (12)$$

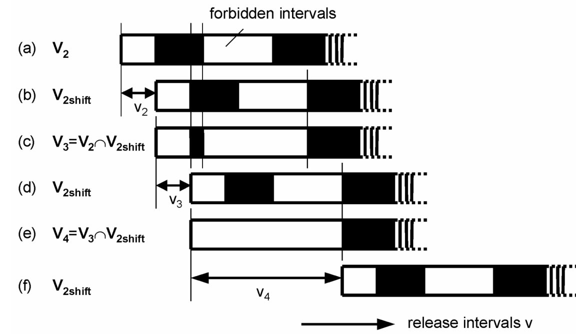

Fig.7. Constraining future release intervals: non periodic schedules. Process example from Fig. 3

Graphically, $\mathbf{V_3}$ is obtained as illustrated in Fig.7. At the uppermost row, Fig 7a, the set $\mathbf{V_2}$ of admissible release intervals from Fig. 3a is duplicated in black. In the second row of Fig. 7b, the smallest possible interval v_2 is chosen for the second process P1 to be released to the shop. The now two processes P1 underway in the system constrain the possible release intervals v_3 for the third process, respectively the set $\mathbf{V_3}$ (Fig. 7c). $\mathbf{V_3}$ is the intersection of the two autocorrelations shifted against each other by v_2 and is calculated from

$$\mathbf{V}_3 = \mathbf{V_2} \bigcap \mathbf{V_{2shift}} \quad (13)$$

Next, the smallest possible value for v_3 is selected from $\mathbf{V_3}$ and the third process is added to the system (Fig. 8d). Again, $\mathbf{V_3}$ is constrained by the shifted auto correlation $\mathbf{V_{2shift}}$ leading to $\mathbf{V_4}$:

$$\mathbf{V}_4 = \mathbf{V_3} \bigcap \mathbf{V_{2shift}} \quad (14)$$

$\mathbf{V_4}$ in Fig. 7e is empty except for admissible release intervals $v_4 > f$, f being the process flow time as defined in Fig. 3a. That means, first, that no more than three jobs can be processed in this shop at a time, and secondly, that the job schedule v_2 v_3 v_4 is non periodic (a periodic schedule requires $v_2 = v_3 = v_4$).

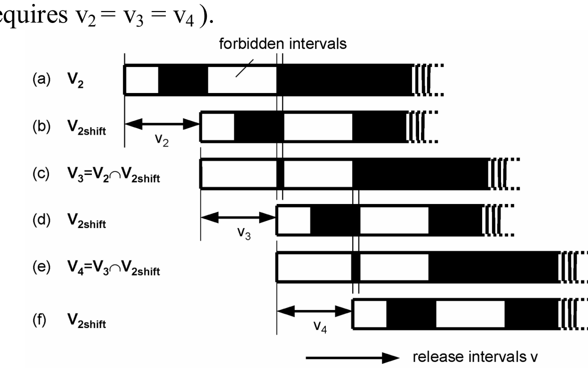

Fig.8. Constraining future release intervals: periodic scheduling. Process example from Fig. 3

However, periodic schedules exist, as exemplified in Fig. 8: $v_2 = v_3 = v_4 = \ldots = v_n$ is fulfilled and $\mathbf{V_3} = \mathbf{V_4} = \ldots = \mathbf{V_n}$ as well. The existence (and size) of periodic schedules can easily be proven by analyzing the auto correlation $\mathbf{V_2}$: if all multiples of $n \cdot v_2$ (n = 1, 2, …) are contained in $\mathbf{V_2}$ then v_2 is a periodic solution, n being the number of parallel processes being persistent in the shop. For n = 3, this statement follows from Eqn. 12 with $v_2 = v_3$:

$$\mathbf{V_3} = \{ v_2 | AC_{12}(v_2) = 0 \wedge AC_{12}(2v_2) = 0 \} \quad (15)$$

For $n = 4$, the statement is proven in [3]. The extension for $n > 4$ is simple.

In Figs. 7 and 8, the non-periodic and periodic schedules are of equal lengths $v_2 + v_3 + v_4$. Both schedules achieve the same (optimal) shop productivity. That is interesting as the Greedy-like algorithm behind Fig. 7 normally is suboptimal as compared to the look-ahead approach in Fig. 8. We elaborate on this finding in the next section with a more realistic process.

VI. Correlation maps

In the preceding section, Figs. 7 and 8 were just a visualization of two different scheduling algorithms based on Eqn. 12: The second process can only enter if $AC_{12}(v_2)=0$ is fulfilled (first row, Fig. 7a). Then, the third process can only enter if no collisions occur with the second process, meaning $AC_{12}(v_3)=0$, and with the first one, meaning $AC_{12}(v_2+v_3)=0$ (third row, Fig. 7c). Then, the fourth process can only enter, if no collisions take place with the third one, $AC_{12}(v_4)=0$, with the second one, $AC_{12}(v_3+v_4)=0$, and with the first one, $AC_{12}(v_2+v_3+v_4)=0$ (fifth row, Fig. 7e). By this repetitive planning procedure, the space of admissible release intervals is constrained stepwise by taking choices, as in any planning. The set notation of the planning algorithm is

$$\mathbf{V_{n+1}} = \mathbf{V_n} \bigcap \mathbf{V_{2shift}} \quad , n \geq 2 \quad (16)$$

with $\mathbf{V_2}$ from Eqn. 4 and $\mathbf{V_{2shift}}$ as illustrated in Figs. 7 and 8. Different means exist to calculate Eqn. 16 and to visualize the results. For instance, the ordered set $\mathbf{V_2}$ may be represented as a function $V_2(v)$, see Fig. 4d. The shifts of Figs. 7 and 8 can then be modelled by correlating $V_2(v)$ similar to Eqn. 3 with the difference that we now correlate auto correlations AC instead of processes P, e.g.

$$V_3(x) = \int V_2(v) \cdot V_2(v+x) dv \quad (17)$$

$$\mathbf{V_3} = \{ v_3 | V_3(v_3) \neq 0 \wedge V_2(v_2) \neq 0 \} \quad (18)$$

However, as a useful decision support for large shops and real planning problems, we adopt two-dimensional correlation or collision maps as recently proposed in [3]. Each map, in a two dimensional array, displays the admissible and forbidden release intervals for the next two processes to be sent to the shop. For example, for the process P1 of Table 1, $\mathbf{V_3}$ according to Eqn. 12 is sketched in Fig. 9 as dependent on v_2 and v_3 . The map is shown as a grey scale image. As throughout this paper, the mostly interesting black areas refer to collision free sequences of release intervals v_2 , v_3 . We therefore restrict our attention to the black and white map of Fig. 10. This figure shows how the final map (bottom right) is built up from individual maps for each single resource, by constraining the black areas with the white (forbidden) ones. Fig. 10 resembles Fig. 7 in Section III, where the process auto correlation AC_{12} is also summed up from resource auto correlations AC_{M1} , …, AC_R .

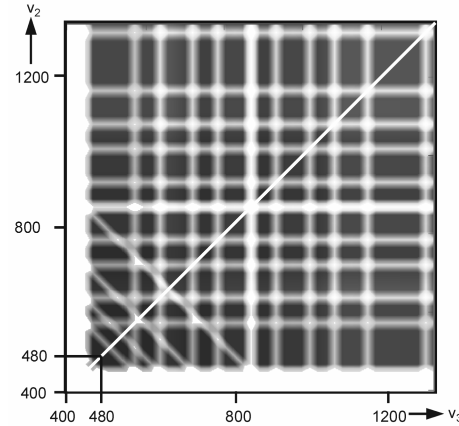

Fig.9. Collision map for process P1 (Table 1; no moves)

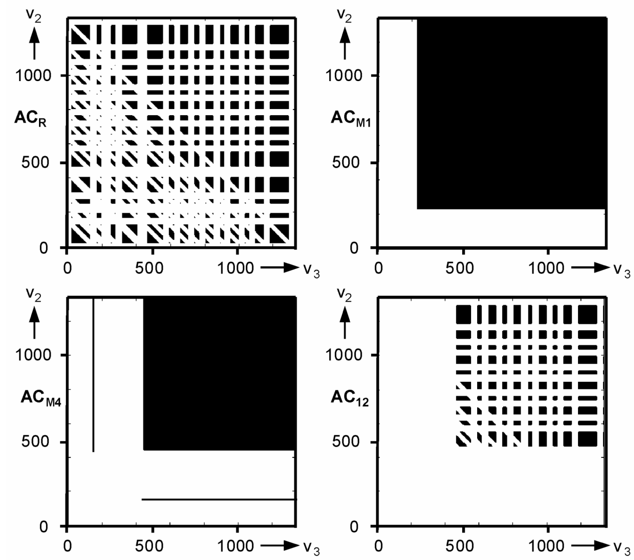

Fig.10. Collision maps for P1 (Table 1; no moves). Individual maps for robot, machines M1 and M4, and combined map.

Fig. 10 resembles Fig. 7 in Section III, where the process auto correlation AC_{12} is summed up from resource auto correlations AC_{M1} , …, AC_R in a similar way.

In Figs. 9 and 10, the horizontal structures refer to $AC_{12}(v_2)\neq 0$ and collisions among P1, P2; the vertical ones to $AC_{12}(v_3)$)$\neq 0$ and collisions among P2, P3; and the diagonal ones to $AC_{12}(v_2+v_3)$)$\neq 0$ and collisions among P1 and P3. Following the same line of reasoning, higher order maps can be constructed as well. The four quadrants of Fig 11 show the sequence of $\mathbf{V_2}$, $\mathbf{V_4}$, $\mathbf{V_6}$, $\mathbf{V_8}$ for a rank 1, multiplicity 11 process. The trajectory refers to a Greedy-like, apparently non-periodic schedule of 0, 480, 480, 540, 531, 489, … (which turns out to be 6-part periodic). Most important, though, are the 1-part periodic schedules. By inspection of the correlation maps we obtain 0, 516, 516, 516, … as the optimum , i.e. shortest periodic schedule.

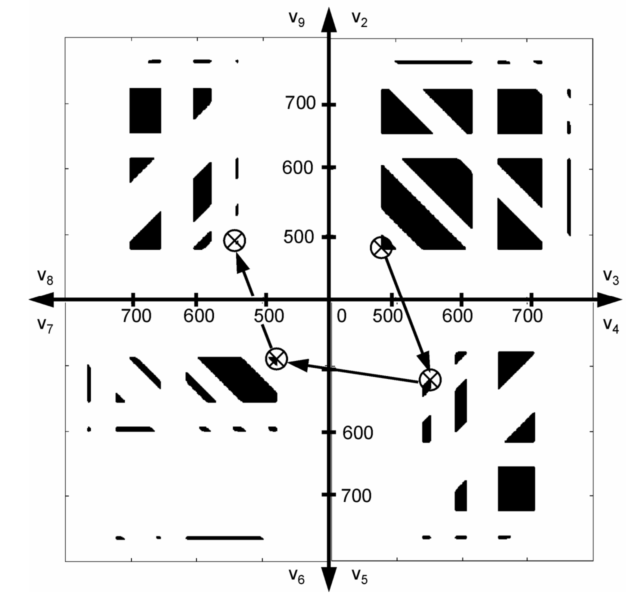

Fig.11. Four collision maps $\mathbf{V}_2$, $\mathbf{V}_4$, $\mathbf{V}_6$, $\mathbf{V}_8$ arranged in a clockwise manner. The trajectory shows the first 8 release intervals v_2, v_3, v_4, … , v_9 of a non periodic schedule

Here, with an average release period of v = 504, the 6-periodic solution is slightly better than the 1-periodic one. However, more investigations of the dominance or non-dominance of 1-periodic schedules (the '1-cycle conjecture' [1]) are needed.

VII. Further Applications and Constraints

Most problems encountered with cycle shop control arise from collisions caused by routing loops. These loops are clearly visible as diagonals in the correlation maps, see Fig. 10 for example. Size and location of the diagonals sensibly depend on the loop characteristics: Fig. 10 (upper left map) shows multiple small cycles of the robot whereas in Fig. 12 (right hand map) the big loop at machine M1 of process P2 (Table 2) exempts many release intervals (white areas).

Table 2. Process plan P2 (big loop)

resources	M1	M2	M3	M4	M5	M1	R
operations m(t)	m1	m2	m3	m4	m5	m1	tr
processing times dm	212	191	402	341	437	263	21

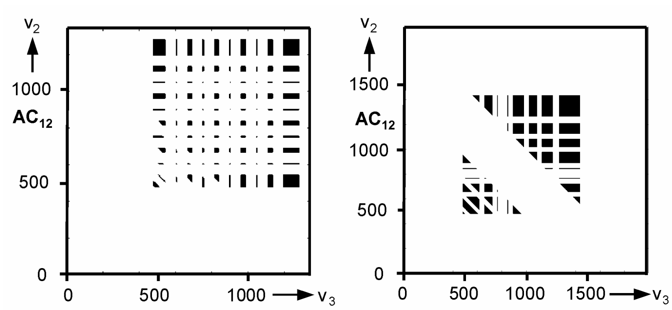

Fig.12. Collision maps for short loop process P1 from Table 1 including moves (left); big loop process P2 from Table 2 (right)

The previous examples did not consider varying transport times and empty moves of the robot. According to Fig. 5b, these can easily be incorporated into the Gantt-chart and the robot auto correlation function. Fig. 12 (left hand side) takes into account empty moves of process P1 (Table 1). As expected, the resulting collision map is slightly more constrained than the previous one without moves (Fig.10, bottom right). Similarly, precedence or other constraints posed by the process plan, can be modeled by proper combination of auto correlations. The method was successfully employed for the complex 2-robot 24-machine rank-5 multiplicity-28 cycle shop described in [5].

What remains a problem for this and almost all other analytical approaches, are sequence dependent set-up and move times. It is an inherent feature of correlation methods that some absolute time and sequence information is suppressed in favour of a compact representation, compare Figs. 4 and 5.

VIII. Conclusion

Main characteristics of a robotic cycle shop are multi purpose machines and loops in the transport routing. Each loop is permanently endangered by collisions and therefore needs to be controlled. By collision maps as presented in this paper, the impact of each individual loop on future planning decisions can be conveniently analysed, displayed and quantized. The underlying deterministic correlation theory was applied to periodic and non periodic schedules of identical processes but can be adopted for the general robotic job shop problem as well. The extension to probabilistic processes is possible, too.

References

[1] J. Blazewicz, N. Brauner and G. Finke, "Scheduling with Discrete Resource Constraints" in: *Handbook of Scheduling*. Boca Raton: Chapman & Hall, 2004, pp. 23-1–23-18.

[2] V. G. Timkovsky, "Cycle Shop Scheduling" in: *Handbook of Scheduling*. Boca Raton: Chapman & Hall, 2004, pp. 7-1–7-22.

[3] W. Meyer, C. Fiedler, "Auto Correlation and Collision Avoidance in Robotic Flow Shops", Proc. 45th *IEEE Conf. Decision and Control CDC 2006,* San Diego, December 2006

[4] M.-A. Manier, C. Bloch, "A Classification for Hoist Scheduling Problems", *International Journal of Flexible Manufacturing Systems*, vol. 15, no.1, pp. 37 – 55, Jan 2003.

[5] C. Fiedler, W. Meyer and A. v. Drathen, "Designing Transitory Routing Controllers for Hybrid Plants", Proc. *IEEE Conf. Control Applications*, Toronto, pp. 934-939, July 2005.

[6] V. Kats, E. Levner, L. Meyzin, "Multiple-Part Cyclic Hoist Scheduling Using a Sieve Method", *IEEE Transactions on Robotics and Automation*, vol. 15, no. 4, pp. 704-713, Aug. 1999.

[7] C. Fiedler, "Event-driven Generation of Periodic Hoist Schedules", Proc. *IEEE Conf. Systems, Man, and Cybernetics*, Taipei, Oct. 2006.