Throughput Optimization in Single and Dual-Gripper Robotic Cells

U.V. Manoj; manojuv@tamu.edu
College of Engineering, Texas A&M University, College Station, TX

Chelliah Sriskandarajah
Mays Business School, Texas A&M University, College Station, TX

Abstract

Throughput optimization is an important problem facing manufacturing industry today due to intense global competition. Automated robotic cell are used to efficiently produce parts to ease this competition. In view of maximizing throughput, practitioners uses a class of cycles known as 1-unit cycles in which the cell returns to the same state after the production of each unit. The complexity of throughput optimization in the class of 1-unit cycles in single and dual-gripper robotic cells is the main focus of this paper. We provide some insights for throughput optimization using two-unit cycles.

Keywords: Robotic cells, Single-gripper and dual-gripper robots, Throughput optimization

1 Introduction

A robotic cell consists of a set of m processing stages, $M = \{M_1, M_2, \ldots, M_m\}$, an input buffer I that holds unprocessed parts, an output buffer O for completed parts, and a robot that performs all material-handling functions – loading parts onto the machines, unloading parts from the machines, and transferring parts from one stage to the next. Figure 1 shows a four-stage cell that has one machine per stage and a dual-gripper robot. If a part that has completed processing on a machine can remain on that machine only for a specified time-window. Such cells are referred to as *interval cells*. For *no-wait cells* this time window is zero. In *free pick-up cells*, there is no time limit on the residence time of a part after it has completed processing on a machine. Each machine can hold only one part, and the cell has no buffers for intermediate storage. Therefore, all parts must be either in I, O, on one of the machines, or on the robot's grippers. We study the problem of scheduling operations in a single and dual-gripper robotic cells processing identical parts in order to maximize the throughput.

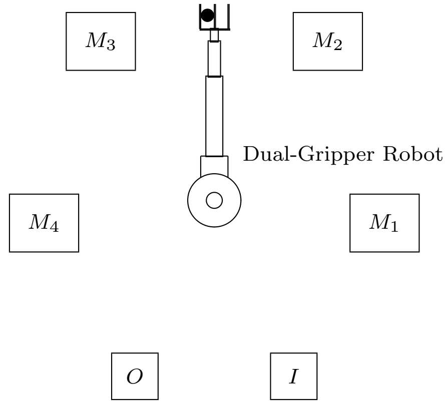

Figure 1: A Four-Stage Dual-Gripper Robot Cell. The input and output buffers are also denoted as machines M_0 & M_{m+1}.

2 Literature Review

We use the classification scheme for robotic cell scheduling problems described in Dawande et al. (2007), which is an extension of that of Graham et al. (1979) for classical scheduling problems. The scheme employs a three-field descriptor $\psi_1|\psi_2|\psi_3$, where ψ_1 specifies the machine environment, ψ_2 describes the part characteristics and/or restrictive requirements, and ψ_3 defines the objective function to be minimized. For example $RF_m^{2,\circ}|(interval,A, cyclic\text{-}1)|C_t$ denote the problem of finding an optimal 1-unit cycle in interval dual gripper cell with circular layout and additive-travel-time. A detailed discussion of various robotic cell problems can be found in Dawande et al (2007). Other reviews of the literature on throughput optimization in robotic cells include Crama et al. (2000), Dawande et al. (2005b), and Brauner (2008).

3 Notation and Definition

In this section, we will provide the notation and definitions used in the paper. We will begin by providing the definition to $T(S)$. $T(S)$ is the cycle time of a robotic cell repeatedly executing a k-unit cycle, S, when the cell operates under a *steady state*. Hence, $\frac{T(S)}{k}$, is the average time required to produce one unit. Note that, minimizing per unit cycle time is equivalent to maximizing throughput.

$[p_i, q_i]$: the processing time window of a part on machine M_i. A part must be processed for p_i time at M_i and must be removed from machine M_i within $(q_i - p_i)$ time units after its completion of processing at M_i, where $p_i \leq q_i, \forall i$. For no-wait cells $p_i = q_i, \forall i$. For free-pickup cells $q_i = \infty, \forall i$.

δ: the time taken by a robot to travel between two adjacent machines.

ϵ: the time taken by the robot to load (unload) a part onto (from) any machine.

θ: the time from the moment one gripper has unloaded a machine until the moment the second gripper is positioned to load the same machine.

M_i^ℓ/M_i^u: the robot activity *load/unload part onto machine* M_i, $i = 0, 1, 2, \ldots m+1$.

For 1-unit robot move sequences in a dual-gripper cell, we need to consider four distinct ways in which a robot can load and unload a machine. We refer to these as *usages*.

$\mathcal{U}_1$: the robot loads a part P onto a machine M_k, waits at the machine during the entire processing of P, and then unloads P. During this usage, the robot's second gripper is either empty or occupied by a part to be processed at a machine other than M_k.

$\mathcal{U}_2$: the robot arrives with a part P and waits at the machine for the previously loaded part P' to be completed, unloads P' from the machine, rotates the grippers, loads P onto the machine, and moves onto the next machine. $\mathcal{U}_3$: the robot loads a part P onto a machine M_k, leaves M_k to perform other activities at other machines, and then returns to M_k to unload P. During this usage the robot's second gripper is either empty or occupied by a part to be processed at a machine other than M_k.

$\mathcal{U}_4$: the robot arrives with a part P at M_k (to be processed on M_k) and waits at M_k for the previously loaded part P' to be completed, unloads P' from M_k, moves onto the next machine, performs other activities, and returns to M_k to load the part P onto it.

4 Two Extreme Cases of Interval Cells: Free-Pickup Cells and No-Wait Cells

The free-pickup and no-wait criteria represent the two extremes of the general interval pickup criterion. We now discuss the two criteria in detail.

4.1 Free-Pickup Cells

Finding an optimal 1-unit cycle in $RF_m^{2,\circ}|(\textit{free},A,\ \textit{cyclic-1})|C_t$ is the focus of this section. We briefly review the results for $RF_m^{2,\circ}|(\textit{free},A,\ \textit{cyclic-1})|C_t$.

4.1.1 An Optimal Cycle when $\theta \leq \delta$

For free-pickup cells with the practically relevant conditions $p_i \geq \delta, \forall i$; $\delta \geq \theta$, cycle $S_m^{''} = (M_0^u, M_1^u, M_1^\ell, M_2^u,\ M_2^\ell,\ \ldots, M_m^u, M_m^\ell, M_{m+1}^\ell)$ is optimal over 1-unit cycles (Geismar et al.

2006). Note that in cycle S''_m each machine usages is $\mathcal{U}_2$.

We now propose the following algorithm for the problem $RF_m^{2;\circ}|(\textit{free},A,\textit{cyclic}\text{-}1)|C_t$ with arbitrary processing times.

Algorithm Dual-Free
Input: Data for a free-pickup cell with a dual-gripper robot.
Output: An asymptotically optimal 1-unit cycle.
If $\delta \geq \theta$ then
  For $j = 1$ to m
    If $p_j \geq \theta$ then assign M_j usage $\mathcal{U}_2$,
    Else assign M_j usage $\mathcal{U}_1$.
  Construct cycle S''_* by visiting machines in ascending order $(0, 1, \ldots, m, m+1)$ and by performing at each machine the action corresponding to the usage assigned above.
Output constructed cycle.

The following result which is due to Dawande et al (2010), for constant travel time robot cell, is also valid for additive travel time robot cell with circular layout (Drobouchevitch et al 2006). The result below is also valid for the linear layout.

Theorem 1 *Algorithm Dual-Free finds an optimal cycle (S''_*) for problem $RF_m^{2;\circ}|(free,A,cyclic\text{-}1)|C_t$ when $\theta \leq \delta$.*

Proof: Dawande et al (2010).

For linear layout, if $\delta < \theta$ and θ is very large, then a single-gripper 1-unit cycle is optimal and is obtainable in polynomial time by an algorithm due to Crama and ven de Klundert (1997a). However, for the circular layout, if $\delta < \theta$ and θ is very large, then a single-gripper 1-unit cycle is shown to be binary NP-hard (Rajapakshe et al. 2011). In other words, the problem of finding an optimal 1-unit cycle in additive-travel-time free-pickup cell with which a single-gripper robot and circular layout ($RF_m^{1;\circ}|(free, A, cyclic\text{-}1)|C_t$) is binary NP-hard.

Theorem 2 *In a free-pickup cell under circular layout with a single-gripper robot, the recognition version of problem $RF_m^{1;\circ}|(free,A,\ cyclic\text{-}1)|C_t$ is binary NP-Complete.*

Proof: Rajapakshe et al. (2011). ■

The above results is true for the problem of finding an optimal 1-unit cycle in additive-travel-time interval cell with with a dual-gripper robot and circular layout ($RF_m^{2;\circ}|(free, A, cyclic\text{-}1)|C_t$) when θ is very large.

Remark 1 *The problem $RF_m^{2;\circ}|(free,A,\ cyclic\text{-}1)|C_t$, is polynomial when $\theta \leq \delta$ and NP-hard when $\theta \geq 3\delta$ (Rajapakshe et al., 2011). The complexity of the following problem is an open question: $RF_m^{2;\circ}|(free,A,\ cyclic\text{-}1)|C_t$ when $\delta < \theta < 3\delta$.*

4.2 No-wait Cells

Dawande et al (2010) show that for No-wait Euclidean travel time robot cell ($RF_m^2|$(*no-wait,E, cyclic-1*)$|C_t$), finding an optimal 1-unit cycle is achieved in polynomial time. This result is proven without any restriction on θ. Note that this result is also valid for constant and additive travel time cells under both linear and circular layouts. Table 1 summarizes the results in this paper and places them with respect to the existing literature.

Table 1: Complexity Map of the Problem of Obtaining an Optimal 1-Unit Cycle (P: Polynomially Solvable, UNP-H: Unary NP-Hard, BNP-H: Binary NP-Hard).

Distance Metric ↓	Single-Gripper Cell			Dual-Gripper Cell ($\theta \leq \delta$)		
	Free	*No-Wait*	*Interval*	*Free*	*No-Wait*	*Interval*
Euclidean $\delta = \min_{0 \leq i < j \leq m+1}\{\delta_{i,j}\}$	UNP-H Brauner et al. (2003)	P Levner et al. (1997)	UNP-H Lei and Wang (1989)	P Dawande et al. (2010)	P Dawande et al. (2010)	UNP-H Dawande et al. (2010)
Constant	P Dawande et al. (2002)	P Levner et al. (1997)	UNP-H Dawande et al. (2010)	P Dawande et al. (2010)	P Dawande et al. (2010)	UNP-H Dawande et al. (2010)
Additive Linear	P Crama and ven de Klundert (1997a)	P Levner et al. (1997)	UNP-H Crama and ven de Klundert (1997b)	P Dawande et al. (2010)	P Dawande et al. (2010)	BNP-H Our Paper
Additive Circular	BNP-H Rajapakshe et al. (2011)	P Levner et al. (1997)	BNP-H Rajapakshe et al. (2011)	P Dawande et al. (2010)	P Dawande et al. (2010)	BNP-H Our Paper

5 Interval Cells under a Circular Layout

We now show that the problem of finding an optimal 1-unit cycle in additive-travel-time interval cell with circular layout ($RF_m^{2,\circ}|(interval, A, cyclic\text{-}1)|C_t$) is binary NP-hard even when $\theta \leq \delta$. We use the Partition problem for our NP-complete reduction (Garey and Johnson 1979). Note that the result below is obtained for 1-unit cycles without usage $\mathcal{U}_4$ as the part must be loaded onto the next machine as soon as it is unloaded from any machine in interval cells. However, we relax this assumption in a later section.

Theorem 3 *In an interval cell under circular layout with a dual-gripper robot, the recognition version of problem* $RF_m^{2,\circ}|(interval,A,\ cyclic\text{-}1)|C_t$ *with usages* $\mathcal{U}_1$, $\mathcal{U}_2$, *and* $\mathcal{U}_3$, *is binary NP-Complete even when* $\theta \leq \delta$.

Proof: Given an arbitrary instance of Partition problem, we now describe a polynomial-time construction of an instance of $RF_m^{2,\circ}|(interval, A, cyclic\text{-}1)|C_t$.

The number of machines is $m = 4n + 5$ with $\delta = 2B$, $\theta = 0$, $\epsilon = 0$, and a constant $L = 2B$. The processing time interval for $M_k, k = 1, 2$, is $[p_k, q_k] = [(4n + 7)\delta + (n + 1)L + 2B, (4n + 7)\delta + (n + 1)L + 2B]$.

The processing time interval for $M_{2k}, k = 2, 3, \ldots, 2n+2$, is $[p_{2k}, q_{2k}] = [0, 2(4n+7)\delta + 2(n+1)L + 4B]$.
The processing time interval for $M_{2k+1}, k = 1, 2, \ldots, 2n$, is $[p_{2k+1}, q_{2k+1}] = [L + a_k, L + a_k]$.
The processing time interval for M_{4n+3} is $[p_{4n+3}, q_{4n+3}] = [L + B, L + B]$ and that for M_{4n+5} is $[p_{4n+5}, q_{4n+5}] = [L + B, L + B]$.

Decision Question (DQ): "Does there exist a 1-unit cycle π_r with cycle time $T(\pi_r) \leq [2(4n + 7)\delta + 2(n + 1)L + 4B]$?"

Since, a "yes" answer to DQ can be verified in polynomial time, DQ is in class NP. We show that there exists a 1-unit cycle π_r with $T(\pi_r) \leq [2(4n + 7)\delta + 2(n + 1)L + 4B]$ if and only if there exist a solution to PARTITION. Without loss of generality, we assume that a 1-unit cycle starts with activity M_1^l.

If part: Suppose that there exists a solution to PARTITION. Without loss of generality, assume that $a_1 + a_3 + \cdots + a_{2n-1} = B$ and $a_2 + a_4 + \cdots + a_{2n} = B$. Consider the 1-unit cycle $\pi_r = (M_1^l, \phi_1, \phi_3, \ldots, \phi_{2n-1}, \phi_{2n+1}, M_2^l, \phi_2, \phi_4, \ldots, \phi_{2n}, \phi_{2n+2})$ where $\phi_k, k = 1, 2, \ldots, 2n + 2$, represents the subsequence (M_{2k+1}^l, M_{2k+2}^l); thus, ϕ_k indicates that the usage at machine M_{2k+1} is $\mathcal{U}_1$. It is easy to verify that π_r is a feasible 1-unit cycle. Also, the contribution of each subsequence $\phi_k, k = 1, 2, \ldots, 2n + 2$, to the cycle time of π_r is $p_{2k+1} + 2\delta$. The robot does not wait to unload a part at any machine except at $M_{2k+1}, k = 1, 2, \ldots, 2n$, M_{4n+3} and M_{4n+5}, where it performs usage $\mathcal{U}_1$. The robot experiences a total waiting time of $2(n + 1)L + 2B + \sum_{i=1}^{2n} a_i = 4B + 2(n + 1)L$. The total time for the robot travel time is $2(4n + 7)\delta$. Thus, the total cycle time of π_r is $T(\pi_r) = [2(4n + 7)\delta + 2(n + 1)L + 4B]$, as required. □

Only If part: Suppose there exist a cycle π_r such that $T(\pi_r) \leq [2(4n+7)\delta+(2n+2)L+4B] = T_{UB}$. Then, we identify the usages for machines $M_j, j = 1, 2, \ldots, m$, through a series of claims.

Claim 1: *In cycle π_r, machines $M_{2k+1}, k = 1, 2 \ldots, 2n + 2$, must perform usage $\mathcal{U}_1$.*

Proof of Claim 1: The time to load and unload in usages $\mathcal{U}_2$ and $\mathcal{U}_3$ violates the assumption. Hence the proof. □

Claim 2: *In cycle π_r, machines M_1 and M_2 must have usage $\mathcal{U}_3$.*

Proof of Claim 2: Suppose one of them, say machine M_1, has usage $\mathcal{U}_1$ in cycle π_r. Then, the robot's total waiting time at machines M_1 and $M_{2k+1}, k = 1, 2 \ldots, 2n + 2$, is

$(4n+7)\delta+3(n+1)L+6B$. Furthermore, in a 1-unit cycle, the robot loads and unloads each of the $(4n+5)$ machines exactly once and visits I and O exactly once; thus, the total travel time incurred is at least $(4n+7)\delta$. Consequently, $T(\pi_r) \geq 2(4n+7)\delta+2(n+1)L+4B$, which violates the upper bound on the cycle time. Cycle $T(\pi_r)$ is, thus, infeasible. The argument for the case when M_2 has usage $\mathcal{U}_1$ and for the case when both M_1 and M_2 have usage $\mathcal{U}_1$ is similar.

First we show that both M_1 and M_2 cannot have usage $\mathcal{U}_2$. Suppose both M_1 and M_2 have usage $\mathcal{U}_2$, then in 1-unit cycle the following activities must be consecutive : M_1^u M_1^l M_2^u M_2^l. Thus between activities M_1^l and M_1^u, machines $M_{2k+1}, k = 1, 2 \ldots, 2n+2$, must perform usage $\mathcal{U}_1$. To perform this the robot has to travel on complete circle requiring time $(4n+7)\delta$ and must wait $2(n+1)L+4B$ time at machines $M_{2k+1}, k = 1, 2 \ldots, 2n+2$, to perform usage $\mathcal{U}_1$. This violates the time-window constraint at M_1: $p_1 = q_1 = (4n+7)\delta + (n+1)L + 2B$. Now it is easy to see either M_1 or M_2 cannot have usage $\mathcal{U}_2$. Consequently, machines M_1 and M_2 must have usage $\mathcal{U}_3$. □

As a result of above claims, we have following usages for the machines.

- Machines M_1 and M_2 have usage $\mathcal{U}_3$.
- Machines $M_{2k+1}, k = 1, \ldots, 2n+2$, have usage $\mathcal{U}_1$.
- Machines $M_{2k}, k = 2, \ldots, 2n+2$, have one of the following usage depending form of 1-unit cycle: $\mathcal{U}_1$, $\mathcal{U}_2$, $\mathcal{U}_3$. Note if a machine M_{2k} has usage $\mathcal{U}_1$ it can be replaced by usage $\mathcal{U}_2$ and vice-versa without increasing the cycle time.

Let $[M_i^l, M_j^u]$ (resp., $[M_i^l, M_j^l]$) refer to the elapsed time between the instant of completion of activity M_i^l and the instant of completion of activity M_j^u (resp., M_j^l).

Claim 3: *In the interval* $[M_1^l, M_1^u]$ *the robot performs usage* $\mathcal{U}_1$ *at exactly* $n+1$ *machines from the set* $\{M_{2k+1} : k = 1, \ldots, 2n+2\}$.

Proof of Claim 3: The processing time window for M_1 is tight: $p_1 = q_1 = (4n+7)\delta + (n+1)L + 2B$. Hence, to achieve the required cycle time, the total time elapsed in the interval $[M_2^l, M_1^l]$ is at most $(4n+7)\delta - \delta + (n+1)L + 2B$.

Suppose the robot performs usage $\mathcal{U}_1$ at $(n+2)$ (or more) machines from the set $\{M_{2k+1} : k = 1, 2, \ldots, 2n+2\}$ in the interval $[M_1^l, M_1^u]$. Then, the robot travels at least $(4n+7)\delta$ time units to visit these $(n+2)$ machines and then return to M_1 for unloading. Also, the total full-waiting time of the robot in the interval $[M_1^l, M_1^u]$ is at least $w_f = (n+2)L + \alpha = (n+1)L + 2B + \alpha > (n+1)L + 2B$ where $\alpha = \sum_{i \in \Lambda} a_i; \Lambda \subseteq A, |\Lambda| = n+2$. Thus, the total time elapsed in the interval $[M_1^l, M_1^u]$ is at least $(4n+7)\delta+(n+1)L+2B+\alpha$, which violates the time window constraint on M_1. Consequently, the robot performs usage $\mathcal{U}_1$ on at most $(n+1)$ machines between the instants of completion of the activities M_1^l and M_1^u.

Furthermore, if the robot performs usage $\mathcal{U}_1$ on fewer than $(n+1)$ machines from the set $\{M_{2k+1} : k = 1, 2, \ldots, 2n+2\}$, then between the instant of completion of activity M_2^l and that of the start of M_1^l, it must perform $\mathcal{U}_1$ on at least $(n+2)$ machines in the interval. Hence, as discussed above, the minimum time elapsed between these two instants is $(4n+7)\delta+(n+1)L+2B+\alpha > (4n+7)\delta-\delta+(n+1)L+2B$, which violates the processing time window of M_1. The result follows. □

The argument used to prove the next two claims is similar to that for Claim 3. We, therefore, state these results without proof.

Claim 4: *In the interval $[M_1^l, M_1^u]$, the robot cannot perform both subsequences ϕ_{2n+1} and ϕ_{2n+2}.*

Claim 5: *In the interval $[M_2^l, M_1^l]$, the robot cannot perform both subsequences ϕ_{2n+1} and ϕ_{2n+2}.*

Claim 6: *In cycle π_r, the activity M_1^l must precede M_2^u.*

Proof of Claim 6: Suppose not, then, the interval $[M_2^l, M_2^u]$ is disjoint from the interval $[M_1^l, M_1^u]$. Since the time-window constraint is tight for both M_1 and M_2, the elapsed time for each of these intervals is $(4n+7)\delta+(n+1)L+2B$. Also, an additional time δ is required between the completion of activities M_1^u and M_2^l. Consequently, the total cycle time exceeds the required upper bound in DQ. The result follows. □

Claim 7: *If $T(\pi_r) \leq 2(4n+7)\delta+2(n+1)L+4B$, then there exists a solution to* Partition.

Proof of Claim 7: Since the robot performs usage $\mathcal{U}_1$ exactly at $n+1$ machines from the set $\{M_{2k+1} : k = 1, 2, \ldots, 2n+2\}$ in the interval $[M_1^l, M_1^u]$, the total robot activity and waiting time in the interval (using Claims 1-4) is $(4n+7)\delta+(n+1)L+B+\alpha$, where $\alpha = \sum_{k\in\Lambda} a_k; \Lambda \subseteq A, |\Lambda| = n+1$ (the set Λ corresponds to the machines from $\{M_{2k+1} : k = 1, 2, \ldots, 2n+2\}$ with usage $\mathcal{U}_1$ in $[M_1^l, M_1^u]$). Also, the total robot activity and waiting time in the interval $[M_2^l, M_1^l]$ is (using Claims 1-3 and Claims 5-6) $(4n+7)\delta-\delta+(n+1)L+B+\alpha'$, where $\alpha' = 2B-\alpha$. In order to meet the time window constraint of both M_1 and M_2, we have $\alpha \leq B$ and $\alpha' \leq B$. But $\alpha+\alpha' = 2B$. Hence there exist $A_1 \subseteq A$ $\alpha = \sum_{a_k\in A_1} a_k = \alpha' = \sum_{a_k\in A\backslash A_1} a_k = B, k = 1, \ldots, 2n$, and $|A_1| = |A\backslash A_1|$. We, therefore, have a solution to Partition. This completes the proof of Theorem 3. ■

We now show that in an interval cell with a dual-gripper robot ($RF_m^2|$(*interval,A,cyclic-1*)$|C_t$), the problem of finding an optimal cycle among all 1-unit cycles is NP-hard.

Theorem 4 *In an interval cell under circular layout with a dual-gripper robot, the recognition version of problem* $RF_m^{2,\circ}|$*(interval,A, cyclic-1)*$|C_t$ *is binary NP-Complete even when* $\theta \leq \delta$.

Proof: The proof of Theorem 3 can easily be adopted with inclusion of usage $\mathcal{U}_4$.

6 Interval Cells under a Linear Layout

Note that the problem of finding an optimal 1-unit cycle in additive-travel-time free-pickup cell with linear layout ($RF_m^2|(free, A, cyclic\text{-}1)|C_t$) is polynomial solvable under practically relevant condition, $\theta \leq \delta$. For linear layout, if $\delta < \theta$ and θ is very large, then a single-gripper 1-unit cycle is optimal and is obtainable in polynomial time by an algorithm due to Crama and ven de Klundert (1997a).

Theorem 5 *In an interval cell under linear layout with a dual-gripper robot, the recognition version of problem* $RF_m^2|$*(interval,A, cyclic-1)*$|C_t$ *is binary NP-Complete even when* $\theta \leq \delta$.

Proof: Proof is similar to that of Theorem 4 ∎

7 Conclusions and Future Research Directions

A fundamental open question concerning cyclic solutions, in either a single-gripper or a dual-gripper robotic cell with a free-pickup criterion, is that of finding an optimal k-unit cycle, $k \geq 1$. From an algorithmic point of view, for interval dual-gripper cells, we have settled the open questions for 1-unit cycle in this paper. For a circular layout, it is unclear whether the problem is strongly NP-hard or solvable by a pseudo-polynomial time algorithm. Furthermore, the algorithmic analysis available for k-unit cyclic solutions is scant for $k \geq 2$. Table 2 summarizes in the existing literature for 2-unit cycles. There are still many questions remained to be answered. The main one is whether the complexity results obtained in this paper and in Rajapakshe (2011) for 1-unit cycles be generalized to 2-unit cycles.

References

Brauner, N. 2008. Identical Part Production in Cyclic Robotic Cells: Concepts, Overview and Open Questions, *Discrete Applied Mathematics*, **156**(13), 2480–2492.

Table 2: Complexity Map of the Problem of Obtaining an Optimal 2-Unit Cycle under Circular Layout and the Additive Travel-Time Metrics (P: Polynomially Solvable, UNP-H: Unary NP-Hard, BNP-H: Binary NP-Hard).

	Single-Gripper Cell			**Dual-Gripper Cell** ($\theta \leq \delta$)		
Distance Metric ↓	*Free*	*No-Wait*	*Interval*	*Free*	*No-Wait*	*Interval*
Constant	Open	P Che et al. (2003)	UNP-H Dawande et al. (2010)	Open	Open	UNP-H Dawande et al. (2010)
Additive Linear	Open	P Che et al. (2003)	Open	Open	Open	Open
Additive Circular	Open	P Che et al. (2003)	Open	Open	Open	Open

Che, A., Chu, C., and Levner, E. 2003. A Polynomial Algorithm for 2-degree Cyclic Robot Scheduling, *European Journal of Operational Research*, **145**(1), 31-44.

Crama, Y., Kats, V., van de Klundert, J., and Levner, E. 2000. Cyclic scheduling in robotic flowshops, *Annals of Operations Research: Mathematics of Industrial Systems*, **96**, 97-124.

Crama, Y. and van de Klundert, J. 1997b. Robotic Flowshop Scheduling is strongly NP-complete, In W.K. Klein Haneveld et al. (ed.), Ten Years LNMB, CWI Tract, Amsterdam, 277-286.

Dawande, M., Geismar, N., Pinedo, M., Sriskandarajah, C. 2010. Throughput Optimization in Dual-Gripper Interval Robotic Cells, *IIE Transactions*, **42**(1), 1-15.

Dawande, M., Geismar, H.N., Sethi, S.P., and Sriskandarajah, C. 2007. *Throughput Optimization in Robotic Cells*, Springer.

Dawande, M., Geismar, N., and Sethi, S. 2005a. Dominance of Cyclic Solutions and Some Open Problems in Scheduling Bufferless Robotic Cells, *SIAM Review*, **47**, 709-721.

Dawande, M., Geismar, N., Sethi, S., and Sriskandarajah, C. 2005b. Sequencing and Scheduling in Robotics Cells: Recent Developments, *Journal of Scheduling*, **8**, 387–426.

Drobouchevitch, I., Sethi, P., and Sriskandarajah, C. 2006. Scheduling Dual-Gripper Robotic Cells: 1-unit Cycles, *European Journal of Operational Research*, **171**, 598–631.

Garey, M.R. and Johnson, D.S. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Francisco.

Geismar, H.N., Chan, L.M.A., Dawande, M., and Sriskandarajah, C. 2008. Approximations to Optimal k-Unit Cycles for Single-Gripper and Dual-Gripper Robotic Cells, *Production and Operations Management*, **17**(5), 551-563.

Geismar, H.N., Dawande, M., and Sriskandarajah, C. 2006. Throughput Optimization in Constant Travel-Time Dual-Gripper Robotic Cells with Parallel Machines, *Production Operations Management*, **15**, 311–328.

Graham, R.L., Lawler, E.L., Lenstra, J.K., and Rinnooy Kan, A.H.G. 1979. Optimization and approximation in deterministic sequencing and scheduling: a survey, *Annals of Discrete Mathematics*, **5**, 287-326.

Levner, E., Kats, V., and Levit, V. 1997. An Improved Algorithm for Cyclic Flowshop Scheduling in a Robotic Cell, *European Journal of Operational Research*, **97**, 500-508.

Lei, L. and Wang, T.J. 1989. A Proof: The cyclic hoist-scheduling probelm is NP-complete, Graduate School of Management, Rutgers University, Working paper, #89-0016.

Rajapakshe, T., Dawande, M., Sriskandarajah, C. 2011. Quantifying the Impact of Layout on Productivity: An Analysis from Robotic-Cell Manufacturing. *Operations Research*, **59**(2), 440–454.

Sethi, S., Sidney, J., and Sriskandarajah, C. 2001. Scheduling in dual-gripper robotic cells for productivity gains, *IEEE Transactions on Robotics and Automation*, **17**, 324-341.