Efficient sequencing of industrial robots through optimal control

N.B. When citing this work, cite the original published paper.

(article starts on next page)

Conference on Assembly Technologies and Systems

Efficient sequencing of industrial robots through optimal control

Staffan Björkenstam*, Domenico Spensieri, Johan S. Carlson, Robert Bohlin, Daniel Gleeson

Fraunhofer-Chalmers Centre, Chalmers Science Park, SE-412 88 Göteborg, Sweden

Abstract

In a production plant for complex assembled products there could be up to several hundred robots used for handling and joining operations. Thus, improvements in robot motion can have a huge impact on equipment utilization and energy consumption. By combining recent algorithms for collision free numerical optimal control and for optimal sequencing, we are able to cut down on energy consumption without sacrificing cycle time. The algorithm has been successfully applied to several industrial cases demonstrating that the proposed method can be used effectively in practical applications to find fast and energy efficient solutions.

1. Introduction

By using Computer Aided Engineering (CAE), physical prototypes can be replaced by simulation, new products can be introduced faster, the efficiency of the production system can be optimized using mathematical methods and algorithms, and it can be done by simulation experts and production engineers in a safe and healthy environment.

The automotive industry is an example of an equipment and energy intensive manufacturing, where up to 28% of the vehicle life cycle energy is spent during production ([1]). For example, a typical automotive car body consists of about 300 sheet metal parts, joined by about 4000 welds. Typical joining methods are spot welding, arc welding, gluing and stud welding. In car body assembly plants, the welds are distributed to several hundred industrial welding robots, which are orga-

nized in up to 100 stations. The body shop is indeed investment intense, with the robots as the main consumer of energy ([1]). In [2] it is highlighted how utilization affects different aspects of sustainable production, the link between utilization and productivity, as well as practical considerations when improving utilization in manufacturing industry. Therefore from a sustainability perspective it is highly motivated to develop new software methods and algorithms for further improvement of equipment utilization and energy efficiency of robotized manufacturing systems.

In [3] it is shown that the balancing of weld work load between the executing stations and robots has a significant influence on achievable production rate and equipment utilization. Robot line balancing is a complex problem, where a number of welding robots in a number of stations are available to execute an overall weld load. Each weld is to be assigned to a specific station and robot, such that the line cycle time is minimized. Line balancing efficiency depends on station load balancing, robot welding sequencing, path planning and

---

*Corresponding author Tel.: +46-31-772-4284, fax: +46-31-772-4260, E-mail address: staffan.bjorkenstam@fcc.chalmers.se

effectiveness of robot coordination for collision free execution within each other's working envelopes. Robot coordination impairs cycle time by inserting waiting positions and signals into the original paths. At Volvo Cars it has been proven that by using automatic path planning and line balancing instead of standard off-line programming the cycle time in welding lines can be improved by as much as 25%. The next step for improving the automatic path planning and line balancing is to include detailed optimization of motion profiles between welds. This choice is also from an energy efficiency aspect motivated. Meike and Ribickis [1] investigated different strategies to operate robots in an energy efficient way. Motion profile optimization was one of the strategies pointed out among others such as automatic shut-down and start-up, reusing braking energy, and brake management.

Global methods for optimal control are in general only applicable to systems of relatively few degrees of freedom or to problems with a special structure, see [4] for an example of such a problem. Most often one has to resort to local methods in order to handle more general problems. That is the path taken in this work where we will be using a direct transcription method called Discrete Mechanics and Optimal Control (DMOC), please refer to [5] for details on direct transcription methods in general and [6] for the DMOC method in particular. Direct transcription methods have been applied to solve optimal control problems for industrial robots before, for example by [7]. In this paper we combine recent advances in sequencing and collision free optimal control of industrial robots to create an efficient algorithm for automatic robotic path planning. In order to compute the travel cost between two locations our algorithm first uses a path planner to create an initial collision free path between the locations. The path is then refined using local numerical optimal control techniques to minimize our cost function. This minimum travel cost between nodes and the cost of visiting the nodes are then used by the sequencer in order to find the best possible sequence to complete the entire operation. Collision avoidance is incorporated into the optimal control problem, in the same way as in [8], by approximating the geometry in configuration space rather than $\mathbb{R}^3$ making the size of the resulting optimization problem independent of the complexity of the geometry which contrasts most existing methods, for example [9] and [10].

2. Method

Here we describe the main steps of our algorithm. The sequencing problem is described in 2.1. In sec-

tion 2.2 we describe how to formulate the optimal control problem associated with the costs used by the sequencer, section 2.3 shows how to discretize and solve this problem, and in section 2.4 we provide details on how to incorporate collision avoidance in the optimal control problem.

2.1. Sequencing

In automotive applications, robots are usually assigned a number M of tasks, consisting of welding and sealing operations, for example. Each task can be done in several ways, and they can be performed in different orders. Thus, minimizing the cycle time requires choosing a robot configuration for each task and deciding the order in which the robot performs the tasks. The problem can be modeled as a Generalized Traveling Salesman Problem (GTSP), which, in this work, is solved exactly for instances up to $M = 20$, and by efficient heuristics for larger problems. The exact algorithm is a straightforward generalization of the dynamic programming approach described in [11] for solving the TSP. When this approach is not suitable, an algorithm based on metaheuristics and local search techniques is used: tour improving operations include 3-opt exchanges, double bridge and others, see [12].

Moreover, it is necessary to find collision-free paths between the chosen configurations for consecutive tasks. This is the most expensive part from the computational point of view, therefore a lazy approach is adopted in order to minimize the number of path planning queries needed. The overall method starts by computing a lower bound for all the robot paths. Thereafter, it iteratively finds a minimum cost sequence of tasks, computes collision free motions, and updates the costs of these paths. The algorithm terminates when the time limit is reached or when the required optimality gap is achieved. This approach is very efficient and can be naturally extended in order to deal with other measures of optimality, as the ones considering energy and robot dynamics. However, a crucial issue for good performance is the tightness of the lower bounds. When the function to be minimized is the time, good lower bounds are easy to compute, for example by considering the maximum speed of the joints and paths computed by linear interpolation in the robot configuration space. In this work, anyway, extensions to minimum energy bounds are not straightforward, therefore we approximate the paths costs by providing estimations that take into account static energy consumptions at the start and end configurations for a given path. These costs are not necessarily strict lower bounds but lead to high convergence rate and good overall quality.

2.2. Optimal Control

The problem we want to solve can be formulated as an optimal control problem where we want to minimize a cost functional

$$J = \Phi(x(t_s), t_s, x(t_f), t_f) + \int_{t_s}^{t_f} \mathcal{L}(x(t), u(t), t) dt \quad (1a)$$

while satisfying the constraints

$$\dot{x}(t) = f(x(t), u(t), t) \quad (1b)$$

$$g(x(t), u(t), t) \geq 0 \quad (1c)$$

$$H(x(t_s), t_s, x(t_f), t_f) = 0 \quad (1d)$$

for $t \in [t_s, t_f]$.

Here the state vector is $x(t) = [q(t)^T, \ \dot{q}(t)^T]^T \in \mathbb{R}^{2n}$ where $q(t)$ belongs to the configuration space i.e. in our robot case q is the vector of joint angles, and the control signal $u(t) \in \mathbb{R}^n$ is the vector of actuator torques applied at the joints. The cost functional to be minimized in (1a) contains two terms, the function $\Phi(x(t_s), t_s, x(t_f), t_f)$ which accounts for costs associated with the initial and terminal state, and the integral of the cost Lagrangian $\mathcal{L}(x, u, t)$ which describes costs incurred along the trajectory. In our problems $\Phi(x(t_s), t_s, x(t_f), t_f)$ is typically just the duration, $t_f - t_s$, and the cost Lagrangian is a measure of the power consumption, modeled by a quadratic function $\mathcal{L}(x, u, t) = x^T Q x + u^T R u$ where Q and R are symmetric positive semi definite matrices.

The differential equations (1b) are called the state equations and describe the dynamics of the system. The path constraints on the state and control to be fulfilled along the trajectory are included in (1c) while (1d) contains the boundary conditions. Note that (1c) can include both equality and inequality constraints.

Using the Lagrange-d'Alembert principle we can write our dynamics (1b) as the forced Euler-Lagrange equation

$$\frac{\partial L}{\partial q}(q(t), \dot{q}(t)) + \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}(q(t), \dot{q}(t))\right)$$
$$+ f_L(q(t), \dot{q}(t), u(t)) = 0$$

where L is the Lagrangian of the mechanical system and f_L is the external force acting on the system.

Using this we can write the optimal control problem in configuration variables as minimize

$$J = \Phi(q(t_s), \dot{q}(t_s), t_s, q(t_f), \dot{q}(t_f), t_f)$$
$$+ \int_{t_s}^{t_f} \mathcal{L}(q(t), \dot{q}(t), u(t), t) dt$$

such that

$$\frac{\partial L}{\partial q}(q(t), \dot{q}(t)) + \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}(q(t), \dot{q}(t))\right)$$
$$+ f_L(q(t), \dot{q}(t), u(t)) = 0$$
$$g(q(t), \dot{q}(t), u(t), t) \geq 0$$
$$H(q(t_s), \dot{q}(t_s), t_s, q(t_f), \dot{q}(t_f), t_f) = 0$$

for $t \in [t_s, t_f]$.

2.3. Discrete Mechanics and Optimal Control

If we divide the time interval $[t_s, t_f]$ into N equidistant sub-intervals of duration h, we can formulate a corresponding discrete optimal control problem. Here we will define our discrete joint trajectory to lie on the $N+1$ grid points while our control is defined on the interval midpoints.

Following [13] we formulate a discrete Lagrangian for our system using the midpoint rule

$$L_d(q_k, q_{k+1}) = hL\left(\frac{q_k + q_{k+1}}{2}, \frac{q_{k+1} - q_k}{h}\right)$$

If we now apply the discrete Lagrange-d'Alembert principle to the action of the system we end up with the following forced discrete Euler Lagrange equations

$$D_2 L_d(q_{k-1}, q_k) + D_1 L_d(q_k, q_{k+1})$$
$$+ f_d^+(q_{k-1}, q_k, u_{k-\frac{1}{2}}) + f_d^-(q_k, q_{k+1}, u_{k+\frac{1}{2}}) = 0$$

for $k = 1 \ldots N - 1$, where $D_1 L_d$ and $D_2 L_d$ are the slot derivatives with respect to the first and second argument and f_d^+ and f_d^- are the left and right discrete forces defined as

$$f_d^+(q_k, q_{k+1}, u_{k+\frac{1}{2}}) = f_d^-(q_k, q_{k+1}, u_{k+\frac{1}{2}})$$
$$= \frac{h}{2} f_L\left(\frac{q_k + q_{k+1}}{2}, \frac{q_{k+1} - q_k}{h}, u_{k+\frac{1}{2}}\right)$$

please refer to [6] about different choices of discrete forces. Using the discrete Legendre transformation we get the boundary conditions

$$D_2 L(q_0, \dot{q}_0) + D_1 L_d(q_0, q_1) + f_d^-(q_0, q_1, u_{\frac{1}{2}}) = 0$$

and

$$- D_2 L(q_N, \dot{q}_N) + D_2 L_d(q_{N-1}, q_N)$$
$$+ f_d^+(q_{N-1}, q_N, u_{N-\frac{1}{2}}) = 0$$

for the initial and final joint velocities, $\dot{q}_0$ and $\dot{q}_N$.

Now we can formulate our discrete optimal control problem as minimize

$$\Phi(q_0, \dot{q}_0, t_0, q_N, \dot{q}_N, t_N)$$
$$+ \sum_{i=0}^{N-1} h\mathcal{L}\left(\frac{q_i + q_{i+1}}{2}, \frac{q_{i+1} - q_i}{h}, u_{i+\frac{1}{2}}, \frac{t_i + t_{i+1}}{2}\right)$$

subject to

$$D_2 L(q_0, \dot{q}_0) + D_1 L_d(q_0, q_1) + f_d^-(q_0, q_1, u_{\frac{1}{2}}) = 0$$
$$D_2 L_d(q_{k-1}, q_k) + D_1 L_d(q_k, q_{k+1})$$
$$\qquad + f_d^+(q_{k-1}, q_k, u_{k-\frac{1}{2}}) + f_d^-(q_k, q_{k+1}, u_{k+\frac{1}{2}}) = 0$$
$$- D_2 L(q_N, \dot{q}_N) + D_2 L_d(q_{N-1}, q_N)$$
$$\qquad + f_d^+(q_{N-1}, q_N, u_{N-\frac{1}{2}}) = 0$$
$$g\left(\frac{q_k + q_{k+1}}{2}, \frac{q_{k+1} - q_k}{h}, u_{k+\frac{1}{2}}, \frac{t_k + t_{k+1}}{2}\right) \geq 0$$
$$H(q_0, \dot{q}_0, t_0, q_N, \dot{q}_N, t_N) = 0$$
$$t_{k+1} = t_k + h$$
$$h \geq 0$$

In our application we will assume that we can control our joint torques directly, in this case the discrete forces are

$$f_d^+(q_k, q_{k+1}, u_{k+\frac{1}{2}}) = f_d^-(q_k, q_{k+1}, u_{k+\frac{1}{2}})$$
$$= \frac{h}{2} f_L\left(\frac{q_k + q_{k+1}}{2}, \frac{q_{k+1} - q_k}{h}, u_{k+\frac{1}{2}}\right) = \frac{h}{2} u_{k+\frac{1}{2}}$$

for $k = 0 \ldots N - 1$.

In the discrete setting our bounds on the angles, velocities and torques become

$$
\begin{array}{rcccl}
q_{lower} & \leq & q_i & \leq & q_{upper}, \qquad\qquad i = 0 \ldots N \\
\dot{q}_{lower} & \leq & \dot{q}_0 & \leq & \dot{q}_{upper} \\
\dot{q}_{lower} & \leq & \frac{q_{i+1} - q_i}{h} & \leq & \dot{q}_{upper}, \qquad i = 0 \ldots N - 1 \\
\dot{q}_{lower} & \leq & \dot{q}_{N+1} & \leq & \dot{q}_{upper} \\
u_{lower} & \leq & u_{i+\frac{1}{2}} & \leq & u_{upper}, \qquad i = 0 \ldots N - 1
\end{array}
$$

The variables in our NLP-problem are the initial and final joint velocities, $\dot{q}_0$, $\dot{q}_{N+1}$, the joint trajectory q_i and times t_i for $i = 0 \ldots N$, and $u_{i+\frac{1}{2}}$ for $i = 0 \ldots N - 1$. Note that since we use equidistant time steps it suffices to use t_0 and t_N as time variables. In our implementation we use t_0, t_N and for convenience also h.

To solve the resulting non-linear optimization problem we use the NLP-solver IPOPT (Interior Point OPTimizer) described in [14].
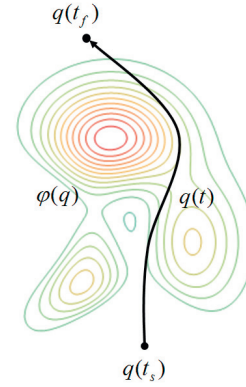

Figure 1. Collision avoiding trajectory and contours of the distance function in configuration space

2.4. Collision avoidance

The initial path given to our optimal control problem is collision free. In this section we describe how this property is maintained while the optimal control problem is solved.

Let us define the distance function as

$$\varphi(q) = \min_{p \in \mathcal{A}(q)} d(p)$$

where $\mathcal{A}(q)$ is the space occupied by the robot at configuration q and $d(p) : \mathbb{R}^3 \to \mathbb{R}$ is the minimum distance to the surrounding geometry $\Gamma \subset \mathbb{R}^3$, i.e. $d(p) = \min_{y \in \Gamma} \|p - y\|_2$.

One way of keeping the solution collision free would be to add a minimum clearance constraint, i.e include

$$\varphi(q(t)) \geq d_c, \ \forall t \in [t_s, t_f] \qquad (4)$$

in (1c), where $d_c \in \mathbb{R}$ is the minimum allowed clearance along the path, this is illustrated in Fig. 1. But $\varphi(q)$ is not generally in C^1 so adding the constraint (4) to the optimal control problem would result in a programming problem which can not be solved using standard NLP-solvers. To overcome this we formulate an iterative method based on local sensitivity analysis.

In the discrete setting the approximate minimum clearance constraints, for iteration k, can be written as

$$n \left\| q_i^{k+1} - q_i^k \right\|_\infty^{w(q_i^k)} \leq \varphi(q_i^k) - d_c \text{ for } i = 0 \ldots N$$

where n is the dimension of the configuration space, q^k is a given feasible configuration, $d_c \geq 0$ is the minimum clearance allowed, and $\| \cdot \|^{w(q)}$ denotes a weighted norm with weight vector $w_i(q) = \max_p \|\frac{\partial p}{\partial q_i}(q)\|_1$ for $i = 1 \ldots n$. These inequalities are simply box constraints
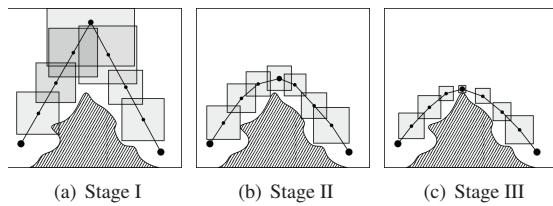
(a) Stage I   (b) Stage II   (c) Stage III

Figure 2. Three different stages in the iterative procedure. Using the collision free path from the previous iteration an improved path is found by optimization constrained to the trust regions. The iteration is initialized by a piecewise linear collision free path from a path planner (Stage I) and terminated when no progress is being made (Stage III).

on the optimization variables at each time step, as illustrated in Fig. 2, for details on the derivations please refer to [8]. Since we use an approximate clearance constraint we can not guarantee that the new configuration q^{k+1} is above our clearance threshold. Hence we need to check the new iterate and perform a backtracking step if necessary. We stop the iteration when the reduction in the objective is below some given threshold.

Starting with an initial collision free path from the path planner, Fig. 2 illustrates how the trajectory is iteratively improved using the clearance constraints.

3. Results

In order to validate our method we have applied it to a virtual stud welding station. The case consists of a car body with 20 stud welding points to be welded by an ABB 6400 industrial robot, the setup can be seen in Figure 3. Each welding takes 2 seconds to perform. The welding points can be visited in any order and each welding point can be reached by the robot from several alternative joint configurations.

To evaluate the performance of our solutions we will use the composite cost function

$$J = \int_{t_s}^{t_f} c_t + c_e u(t)^T u(t) + c_v \dot{q}(t)^T \dot{q}(t) dt$$

i.e. a linear combination of the time, the control effort and a regularizing term to keeping the kinetic energy of the system low. In our test case we have used $c_t = 1.0$, $c_v = 1.0$ and $c_e = 10^{-6}$ which tend to give fast and smooth trajectories with low actuator torques.

The number of alternatives for each welding point and the range of the costs associated with each welding are given in Table 1, note the large variations in cost depending on the alternative joint configurations. The main difference between the method in this paper and
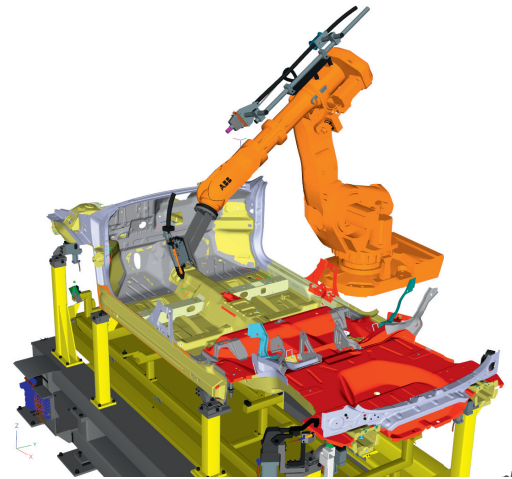

Figure 3. Welding station

Stud	Alt.	Cost
1	6	[41.16, 41.44]
2	8	[79.11, 80.73]
3	16	[110.8, 223.3]
4	22	[95.69, 240.0]
5	27	[33.78, 330.0]
6	18	[58.36, 339.6]
7	10	[240.8, 350.8]
8	16	[176.9, 448.6]
9	20	[89.64, 307.2]
10	30	[48.63, 282.2]
11	16	[35.49, 295.5]
12	36	[50.55, 471.6]
13	16	[73.29, 228.7]
14	13	[140.1, 447.8]
15	16	[44.07, 417.3]
16	9	[173.1, 324.9]
17	9	[181.1, 329.3]
18	7	[174.5, 181.5]
19	6	[241.5, 242.5]
20	6	[222.9, 223.3]

Table 1. Number of alternatives for each weld and the corresponding range of cost associated with performing each stud welding operation

	Travel	Welding	Total
Old sequence	2212.5	4590.4	6802.9
New sequence	1560.9	2323.1	3884.0

Table 2. The cost of completing a cycle of the stud welding station using sequences computed with and without optimal control

our previous method, described in [8], is that we now incorporate optimal control already in the sequencing step. This also means that we can include the cost occurring while performing the welding operation, which increase the potential saving even further.

In Table 2 we compare a sequence computed using our old sequencer, based on approximate travel time minimization, to a sequence computed using our new optimal control based sequencer. In this particular case we see that by using the costs from the optimal control problem already when selecting the sequence we can reduce our cost with over 40 percent. It is particularly interesting to note that the greatest saving is made during the welding operation while the robot is standing still.

4. Conclusion

We have presented a method for efficient sequencing of industrial robots. The method extends our previous method, presented in [8], by incorporating the optimal control problem in the sequence optimization. Our optimal control solver has also been improved by using a discrete variational principle to derive the equations of motion. This gives our solver the nice characteristics, regarding conservation of momentum and energy, usually associated with variational integrators [15].

From our experiments we conclude that our method can indeed give considerable savings under realistic conditions and that the static poses are as important as the motions connecting them.

While this approach can be applied to any robot in a station, collisions among robots require geometrical and/or timing modifications of the paths, see [16] and [17]. This extension is to be included in future work.

References

[1] D. Meike, L. Ribickis, Analysis of the energy efficient usage methods of medium and high payload industrial robots in the automobile industry, 2011, 10th International Symposium on Topical Problems in the Field of Electrical and Power Engineering, Pärnu, Estonia, January 10-15.

[2] P. Almström, C. Andersson, A. Muhammad, M. Winroth, Achieving sustainable production through increased utilization of production resources, 2011, the 4th Swedish Production Symposium, Lund, May 3-5.

[3] J. Segeborn, D. Segerdahl, J. S. Carlson, F. Ekstedt, A. Carlsson, R. Söderberg, A generalized method for weld load balancing in multi station sheet metal assembly lines, 2011, proceedings of the ASME 2011 International Mechanical Engineering Congress & Exposition, Denver, Colorado, USA, November 11-17.

[4] S. Björkenstam, M. Ji, M. B. Egerstedt, C. F. Martin, Leader based multi-agent coordination through hybrid optimal control, in: 44th Allerton Conference on Communication, Control, and Computing, 2006, pp. 1352–1357.

[5] J. T. Betts, Practical methods for optimal control and estimation using nonlinear programming, 2nd Edition, Society for Industrial and Applied Mathematics, 2010.

[6] J. E. Marsden, M. West, Discrete mechanics and variational integrators, Acta Numerica 10 (1) (2001) 357–514.

[7] O. von Stryk, M. Schlemmer, Optimal control of the industrial robot manutec r3, Computational optimal control, International series of Numerical Mathematics 115 (1994) 367–382.

[8] S. Björkenstam, D. Gleeson, R. Bohlin, J. S. Carlson, B. Lennartson, Energy efficient and collision free motion of industrial robots using optimal control, in: Automation Science and Engineering (CASE), 2013 IEEE International Conference on, 2013, pp. 510–515.

[9] A. Müller, Energy optimal control of serial manipulators avoiding collisions, in: Mechatronics, 2004. ICM'04. Proceedings of the IEEE International Conference on, IEEE, 2004, pp. 299–304.

[10] A. El Khoury, F. Lamiraux, M. Taix, Optimal motion planning for humanoid robots (2012).
URL hal.archives-ouvertes.fr

[11] M. Held, R. M. Karp, A dynamic programming approach to sequencing problems, in: Proceedings of the 1961 16th ACM National Meeting, ACM '61, ACM, New York, NY, USA, 1961, pp. 71.201–71.204.

[12] G. Gutin, D. Karapetyan, A memetic algorithm for the generalized traveling salesman problem 9 (1) (2010) 47–60.

[13] S. Ober-Blöbaum, O. Junge, J. E. Marsden, Discrete mechanics and optimal control: an analysis, arXiv preprint arXiv:0810.1386.

[14] A. Wächter, L. T. Biegler, On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming, Mathematical Programming 106 (1) (2006) 25–57.

[15] E. Hairer, C. Lubich, G. Wanner, Geometric numerical integration: structure-preserving algorithms for ordinary differential equations, Springer, 2006.

[16] D. Spensieri, F. Ekstedt, J. Torstensson, R. Bohlin, J. S. Carlson, Throughput maximization by balancing, sequencing and coordinating motions of operations in multi-robot stations, in: Proceedings of the 8th International NordDesign Conference 2010, 2010, pp. 455–465.

[17] D. Spensieri, R. Bohlin, J. S. Carlson, Coordination of robot paths for cycle time minimization, in: Automation Science and Engineering (CASE), 2013 IEEE International Conference on, 2013, pp. 522–527.