Energy reduction of stochastic time-constrained robot stations

Citation for the original published paper (version of record):

Hovgard, M., Lennartson, B., Bengtsson, K. (2023). Energy reduction of stochastic time-constrained robot stations. Robotics and Computer-Integrated Manufacturing, 81.
http://dx.doi.org/10.1016/j.rcim.2022.102511

(article starts on next page)

Robotics and Computer–Integrated Manufacturing 81 (2023) 102511

Contents lists available at ScienceDirect

Robotics and Computer-Integrated Manufacturing

journal homepage: www.elsevier.com/locate/rcim

Full length article

Energy reduction of stochastic time-constrained robot stations

Mattias Hovgard *, Bengt Lennartson, Kristofer Bengtsson

Department of Electrical Engineering, Chalmers University of Technology, S-412 96, Gothenburg, Sweden

ARTICLE INFO

Keywords:
Energy optimization
Robot station
Industrial robot
Stochastic scheduling
Time-constrained

ABSTRACT

This paper looks at the problem of reducing the energy use of robot movements in a robot station with stochastic execution times, while keeping the productivity of the station. The problem is formulated as a stochastic optimization problem, that constrains the makespan of the station to meet a deadline with a high probability. The energy use of the station is a function of the execution times of the robot operations, and the goal is to reduce this energy use by finding the optimal execution times and operation order. A theoretical motivation to why the stochastic variables in the problem, under some conditions, can be approximated as independent and normally distributed is presented, together with a derivation of the max function of stochastic variables. This allows the stochastic optimization problem to be approximated with a deterministic version, that can be solved with a commercial solver. The accuracy of the deterministic approximation is evaluated on multiple numerical examples, which show that the method successfully reduces the energy use, while the deadlines of the stations are met with high probabilities.

1. Introduction

In recent years, the pressure on industry by society and governments to reduce the emission of greenhouse gases has increased. One way to achieve this is to reduce the energy use of the automation and robotic systems that exist in many manufacturing industries. For industrial robots, one of the most common approaches to reduce energy use is trajectory optimization [1–3]. Many of the methods, and in particular the ones that are simplest to implement in practice, rely on increasing the execution time of the trajectories (compared to the most time efficient ones) [4–7].

In real production systems, the execution times of the robot stations are often measured closely to ensure that the productivity (the number of produced products per unit of time) is kept. Increasing the execution time of the robot trajectories (to save energy) may increase the execution time of the production system. So, there is a tradeoff between energy reduction and productivity. This is a problem when dealing with real production systems, because normally productivity is prioritized over energy reduction. Therefore, any method, aimed at reducing the energy use in practice, should ideally guarantee that the productivity is not affected, or at least consider or quantify the tradeoff. This problem can be formulated as a scheduling problem of where in the execution of a robot station it is possible to extend the execution times of the robot movements without affecting the productivity.

The tradeoff between energy reduction and productivity becomes especially challenging if, which is the case in most real-life scenarios, there are disturbances or stochastic processing times in the system. This can be, for example, machine breakdowns, arrival of parts and variations of operations performed by human operators etc. As shown by [8], there is a tradeoff between energy reduction and robustness to stochastic disturbances.

Specific problem

Motivated by the discussion above, the aim of this paper is to formulate and present a solution to the optimization problem of reducing the energy use in a multi-robot station. The focus is on robot movements with stochastic processing times, while keeping the productivity of the station. More specifically, each operation in the station contains a segment with uncontrollable stochastic execution time and a robot movement with controllable deterministic execution time. The energy use of each operation is a function of its deterministic execution time. This problem formulation has also been studied in [4,7,8]. Some, but not all, of the operations require shared resources to execute. This creates a scheduling problem to find in what order the operations are allowed to access the shared resources. To guarantee that the productivity is not affected, the optimization problem contains a so-called chance constraint [9] for the station to meet a deadline with a high probability. The decision variables in the optimization problem are the deterministic execution times and the operation order.

* Corresponding author.
E-mail address: hovgard@chalmers.se (M. Hovgard).

https://doi.org/10.1016/j.rcim.2022.102511
Received 1 November 2021; Received in revised form 28 December 2021; Accepted 6 December 2022
Available online 22 December 2022

Literature review

A number of papers have recently been presented on energy reduction of industrial robots or production systems. Carabin et al. [2] present a review of existing solution methods. Vergnano et al. [4] reduce the energy use of a multi-robot station by using so-called energy signatures. Wigström et al. [10] develop a method that combines scheduling and energy optimal trajectory generation of a multi-robot station. Meike et al. [6] present a practical case study, including modeling and reduction of the energy use of a production line in an automotive factory. Gadaleta et al. [7] reduce the energy use of an industrial robot by using a simulation software to find the optimal combination of velocity and acceleration reduction. Pastras et al. [11] present a theoretical investigation on improving the motion profiles of industrial robots. Zhang et al. [12,13] look at the problems of tracking noisy trajectories and obstacle avoidance for robotic manipulators. Glorieux et al. [14] propose a method for energy optimal trajectory optimization of cyclic robot stations. Gadaleta et al. [15] investigate the optimal placement of an industrial robot in order to use the least amount of energy. Chen et al. [16] and Chang et al. [17] look at the energy use of serial production systems with buffers and stochastic machine breakdowns. Salido et al. [18] and Sundström et al. [8] investigate the tradeoff between energy reduction and robustness, while Faraji Amimiri et al. [19] present a method to reduce the energy use of a stochastic flow-shop-problem by reducing the machine velocities. Gürel et al. [20] Look at the problem of scheduling and reducing the energy use of a single robot in a manufacturing cell.

To summarize the existing literature, there are not many papers that look specifically at the problem defined earlier, namely the combination of energy reduction of robot movements, multi-robot scheduling, stochastic processing times and a chance-constraint for meeting a deadline. Therefore, this is a very relevant problem in many practical applications, which motivates the focus of this paper.

Regarding the challenges in solving the defined optimization problem, the main complexity comes from the combination of stochastic scheduling and the chance constraint of the deadline. This added complexity makes the problem hard to solve accurately even for relatively small problems, such as the multi-robot stations that are investigated in this paper. One of the main challenges in solving this optimization problem is to mathematically express the distribution of the makespan of the station, which is required to express the chance constraint. Some interesting analyzes and solution methods related to this are presented by Brucker et al. [21] and Möhring [22]. Ben-Tal et al. [9] and Nemirovski et al. [23] present different ways to reformulate chance constrained optimization problems to convex approximations only containing deterministic variables. These reformulated problems can be solved efficiently by commercial solvers, and are conservative in the sense that the solutions to the reformulated problems are guaranteed to be feasible for the original problem as well. This approach has also been applied to chance constrained scheduling problems [24,25].

Contribution and outline

To the best of the authors knowledge, the specific optimization problem considered in this paper, which we argue is a relevant problem in many practical applications, has not been investigated before. The main contribution of this paper is to model this optimization problem and to find a solvable deterministic approximation of this problem. Specifically, this includes (a) a theoretical motivation to why the stochastic variables in the problem, under some conditions, can be approximated as independent and normally distributed; (b) a derivation of the max function of two stochastic variables; (c) a way to formulate the optimization problem mathematically such that it can be solved with a commercial solver; and (d) the inclusion of the operation order in the optimization formulation. The second contribution is to show that the derived optimization model is able to successfully solve a number of numerical examples that are based on real robot stations.

This is a continuation of the work presented in [26,27]. Compared to [26] this paper contains the addition of stochastic execution times. Compared to [27] this paper contains the addition of more thorough motivations for some of the simplifications that are used, an extended optimization formulation where the operation order is not fixed but instead a part of the optimization problem, and a much more extensive evaluation of the method.

The rest of the paper is organized as follows: the stochastic optimization problem is presented in Section 2. A deterministic approximation of the stochastic optimization problem is then found, first for a fixed operation order in Section 3, and then extended in Section 4 to the case when the operation order is part of the optimization problem. The method is evaluated on three numerical examples in Section 5, and finally some conclusions are given in Section 6.

Table 1
The most important nomenclature.

$\mathcal{I}$: set of all operations, $i \in \mathcal{I}$.
$\mathcal{I}_r$: set of operations requiring resource r to execute, $\mathcal{I}_r \subseteq \mathcal{I}$.
$\mathcal{R}$: set of all resources, $r \in \mathcal{R}$.
$\mathcal{R}_i$: set of resources required by operation i to execute, $\mathcal{R}_i \subseteq \mathcal{R}$.
$\mathcal{K}_r$: set of available time slots for resource r, $k \in \mathcal{K}_r$.
$\mathcal{A}$: operation order.
a_{irk}: binary decision variables $a_{irk} \in \mathcal{A}$, $a_{irk} = 1$ when operation i is executed in time slot k on resource r.
$\mathcal{L}$: set of ending operations.
$\mathcal{U}_i$: set of operations preceding operation i, because of precedence constraints.
$\mathcal{V}_i$: set of operations preceding operation i, because of resource constraints.
$\mathcal{W}_i$: set of all operations preceding operation i, $\mathcal{W}_i = \mathcal{U}_i \cup \mathcal{V}_i$.
S_i: stochastic starting time for operation i, with mean μ_i^S and variance υ_i^S.
$\mathbf{s}_i$: starting time vector of operation i, $\mathbf{s}_i = [\mu_i^S\ \ \upsilon_i^S]^T$.
E_i: stochastic execution time of operation i, with mean μ_i^E and variance υ_i^E.
$\mathbf{e}_i$: execution time vector of operation i, $\mathbf{e}_i = [\mu_i^E\ \ \upsilon_i^E]^T$.
d_i: deterministic execution time of operation i.
$\underline{d}_i$: lower bound on d_i.
C_i: stochastic completion time on operation i, with mean μ_i^C and variance υ_i^C.
$\mathbf{c}_i$: completion time vector of operation i, $\mathbf{c}_i = [\mu_i^C\ \ \upsilon_i^C]^T$.
$\mathbf{q}_{rk}$: completion time vector of time slot k on resource r.
g_i: energy function of operation i, with parameters $\psi_1^i, \psi_2^i, \psi_3^i, \psi_4^i$
t_d: deadline of the station.
β: required probability to meet the deadline.
T: stochastic makespan of the station.

2. Problem formulation

In this section the optimization problem of reducing the energy use of a robot station with stochastic processing times is presented. The most important nomenclature can be found in Table 1. Regular lower-case letters are used to denote deterministic variables, regular upper-case letters are used to denote stochastic variables and upper-case calligraphy letters are used to denote sets. In the problem there is a set $\mathcal{I}$ of operations $i \in \mathcal{I}$ that need to be executed. The execution time of each operation is a random variable that is a sum of two parts, a stochastic part denoted by E_i and a deterministic part denoted by d_i. Each operation also has a starting time S_i and a completion time C_i. Thus,

$$C_i = S_i + E_i + d_i. \tag{1}$$

2.1. Precedence and resource constraints

Between the operations there are two different types of constraints. The first type is named *precedence constraint* and occurs when there is a fixed order in which some of the operations need to be executed. In the literature this type of constraint is sometimes described by the notion of jobs, where a job contains a set of operations that needs to be executed in a given order. Let $\mathcal{U}_i$ be a set of operations that need to precede operation i. For all operations ℓ in $\mathcal{U}_i$ it must hold that

$$S_i \geq C_\ell. \tag{2}$$

The second type of constraint is named *resource constraint* and is related to which resources an operation needs to execute. The set of resources in the station is denoted by $\mathcal{R}$, each operation needs at least one resource r to execute and each resource can only execute one operation at a time. For a resource r in $\mathcal{R}$ the resource constraint states that for any two operations i and ℓ requiring resource r to execute it must hold that

$$S_i \geq C_\ell \quad or \quad S_\ell \geq C_i. \tag{3}$$

$\mathcal{I}_r$ is the set of operations that require resource r to execute. Examples of resources are robots or so-called shared zones. A shared zone is a physical space that only one robot can access at a time.

The order in which the operations use the resources in the station is named *operation order*. The operation order is set before the station is executed, and it is represented by the set $\mathcal{A}$ (More on this in Section 4). Depending on the operation order, each operation may have an additional set of operations $\mathcal{V}_i(\mathcal{A})$ that need to be finished before it can execute, in addition to $\mathcal{U}_i$. So, for a given operation order $\mathcal{A}$, the total set of operations that precede operations i is given by $\mathcal{W}_i(\mathcal{A}) = \mathcal{U}_i \cup \mathcal{V}_i(\mathcal{A})$. Combining (2) and (3) gives the following constraint for operation i,

$$S_i \geq \max_{\ell \in \mathcal{W}_i(\mathcal{A})} (C_\ell). \tag{4}$$

The makespan of the station (also sometimes referred to as cycle time) is a stochastic variable denoted by T and is defined as,

$$T = \max_{i \in \mathcal{L}} (C_i),$$

where $\mathcal{L}$ is the set of ending operations. An ending operation is an operation that, for at least one operation order, does not have any operations succeeding it. An illustrative example of a robot station is shown in Fig. 1.

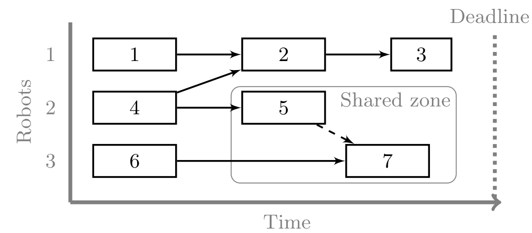


Fig. 1. An illustrative example schedule of a robot station, containing a set of operations and resources. The operations are shown as numbered boxes and the solid arrows between them are precedence constraints. The station contains four resources: three robots and one shared zone. The robots are listed to the left. The operations on the same row as a robot require that robot to execute. Operation 5 and 7 also require the shared zone to execute. Only two operation orders are possible: Operation 5 precedes Operation 7 (which is shown) or the other way around. The additional precedence constraint, that exists because of the operation order, is shown with a dashed arrow.

2.2. Distributions

The stochastic variables in the problem formulation S_i, E_i, C_i, have mean values $\mu_i^S, \mu_i^E, \mu_i^C$ and variances v_i^S, v_i^E, v_i^C. For notational convenience the vectors $\mathbf{s}_i = [\mu_i^S \; v_i^S]$, $\mathbf{e}_i = [\mu_i^E \; v_i^E]$ and $\mathbf{c}_i = [\mu_i^C \; v_i^C]$ are sometimes used. f is used to denote a general probability density function (PDF) and F is used to denote a general cumulative density function (CDF). Out of S_i, E_i and C_i, only E_i can be described by a standard distribution. S_i and C_i are more results of (1) and (4) than something that can be modeled explicitly. Note that all E_i are assumed to be independent from each other.

To model the stochastic execution time E_i, different types of distributions can be used, depending on what type of variation that is being modeled. For example, a common choice when modeling machine breakdowns is to use the exponential distributions. In this paper, E_i represents the natural variations occurring in some types of operations, e.g. welding or human operations. Only some information of the actual distributions is known, e.g. the minimum and maximum values or the mean and an approximation of the standard deviation. For these cases the uniform and normal distributions are suitable. A uniform distribution is denoted by $U(k_1, k_2)$ and the PDF and CDF are defined as [28]

$$f(x) = \begin{cases} \frac{1}{k_2 - k_1}, & \text{if } x \in [k_1, k_2] \\ 0, & \text{otherwise} \end{cases}$$

$$F(x) = \begin{cases} 0, & \text{if } x < k_1 \\ \frac{x - k_1}{k_2 - k_1}, & \text{if } x \in [k_1, k_2] \\ 1, & \text{otherwise.} \end{cases}$$

A Normal Distribution is denoted by $N(\mu, v)$ and the PDF and CDF are defined as [28]

$$\phi(x; \mu, v) = \frac{e^{-\frac{(x-\mu)^2}{2v}}}{\sqrt{2\pi v}},$$
$$\Phi(x; \mu, v) = \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{x-\mu}{\sqrt{2v}}\right)\right), \tag{5}$$

where erf is the error function [29]. For the rest of this paper, if not stated otherwise, it is assumed that E_i are, or safely can be approximated with, a normal distribution.

2.3. Energy use

The energy use of an operation is modeled by a function g_i that depends on the deterministic execution time d_i. The energy functions are found by executing the robot operations with different velocities, recording the energy use and execution times, and fitting functions to the recorded data. The energy functions are parameterized as

$$g_i(d_i) = \psi_1^i \exp(\psi_2^i d_i) + \psi_3^i \exp(\psi_4^i d_i), \tag{6}$$

where $\psi_1^i, \psi_2^i, \psi_3^i, \psi_4^i$ are parameters, and $\psi_1^i, \psi_3^i > 0$ so that the functions are convex. To describe the energy use by a parameterized energy function has been done previously in [4,8], for example. An example of an energy function is $g(d_i) = 15503\exp(-1.3561 d_i) + 4598\exp(0.004212 d_i)$, and some examples can be seen in Fig. 2. Every deterministic execution time is constrained by a lower bound $\underline{d_i}$, which is the shortest possible duration of the operation; this is also the default value of d_i. So to clarify, the energy use of an operation can be reduced by increasing the value of d_i.

Note that in general, operations may have other types of energy functions than what is shown in Fig. 2. What is shown and what is considered in this paper are fast robot movements, typically to move between two locations (more on this in Section 5). Robot movements where a robot simultaneously performs some task, such as glue dispensing, requires relatively slow movements, which probably means that less energy reduction can be made by additional reduction of velocity. These types of operations are not considered in this paper.

2.4. Optimization problem

The purpose of the optimization problem is to consider the tradeoff between productivity and energy reduction. As discussed in the introduction there are many ways for how to formulate such a problem. In this paper, it is assumed that the energy use of the operations should be minimized, under the assumption that the robot station has a deadline when every operation must be completed, and that this deadline must

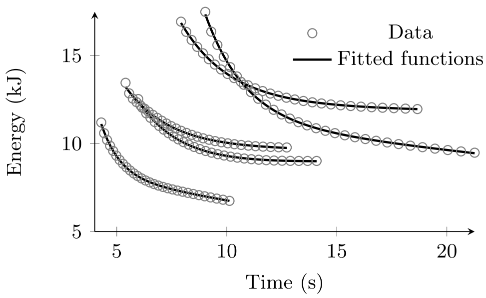


Fig. 2. Some examples of energy functions.

be met with a given high probability. This constraint can be formulated as

$$\mathrm{P}(T < t_d) \geq \beta$$

where P is used to denote probability, t_d is the deadline of the station and β is the required probability by which the deadline must be met. The whole optimization problem can be formulated as follows:

Optimization Problem 1.

$$\min_{\mathcal{A},d} \sum_{i\in\mathcal{I}} g_i(d_i)$$

subject to:

$$C_i = S_i + E_i + d_i \qquad i \in \mathcal{I} \tag{7}$$

$$S_i \geq \max_{\ell\in\mathcal{W}_i(\mathcal{A})}(C_\ell) \qquad i \in \mathcal{I} \tag{8}$$

$$T = \max_{i\in\mathcal{L}}(C_i) \tag{9}$$

$$\mathrm{P}(T < t_d) \geq \beta \tag{10}$$

$$\underline{d_i} \leq d_i \qquad i \in \mathcal{I}$$

where the operation order $\mathcal{A}$ and the deterministic execution times d_i are the optimization variables. □

Note, for a given operation order it is always optimal for every operation to start as early as possible. So, the inequality in (8) can be replaced by an equality. The combined complexity of the scheduling problem to find the operation order together with the chance constraint (10) makes Optimization Problem 1 hard to solve in general.

3. Deterministic approximation

In this section, a deterministic approximation of Optimization Optimization Problem 1 is presented. This is done under the assumption that there is a fixed operation order, which will later be extended to the case with varying operation order in Section 4. The aim is approximate Optimization Problem 1 in a way that it can be expressed only using the mean values and variances of the stochastic variables, which has been found to be accurate enough for the type of robot stations considered in this paper (see Section 5). Three steps will be taken to reach the final approximated optimization problem and also to some extent motivate why the approximation is possible. The three steps are: assume normal distributions, assume independence, and approximate the max-function. The first two steps are relatively common and similar approaches can be found in [21,22]. The third step is more novel, but to get a complete derivation of the proposed optimization method, all three steps are presented in the coming sections.

3.1. Normal distribution

To calculate (7) exactly, the following convolution needs to be solved,

$$f_i^C(x) = \int_{-\infty}^{\infty} f_i^S(x-y) f_i^E(y)\mathrm{d}y + d_i, \tag{11}$$

where f_i^C, f_i^S and f_i^E are the PDFs of C_i, S_i and E_i respectively. Note that S_i and E_i always are independent of each other. To solve (11) is in general too complicated to be practically feasible. If however E_i and S_i are normally distributed, (11) can be simplified [30]. In general, S_i and E_i are not normally distributed; even if every E_i is, S_i is not (because of (8)). However, the central limit theorem [31] can be used to show that for some special cases it can be assumed that E_i and S_i are normally distributed. Otherwise, the assumption can be used as a heuristic. The central limit theorem states that, under some conditions, the distribution of a sum of independent random variables tends to a normal distribution.

To illustrate this phenomenon, consider a simple robot station with a set of resources $\mathcal{R}$, each resource executes a sequence of operations $i \in \mathcal{I}_r$ and the operations of each resource have precedence constraints between them so that the operation order is fixed. None of the operations has precedence constraints related to operations of other resources. The execution times E_i of the operations are either normally or uniformly distributed with mean values μ_i^E and variances v_i^E. For this example, $\mathcal{L}$ contains the last operation of every resource, so (7)–(9) can be simplified to:

$$C_r = \sum_{i\in\mathcal{I}_r} E_i + d_i \qquad r \in \mathcal{R} \tag{12}$$

$$T = \max_{r\in R}(C_r)$$

Because of the central limit theorem the following identity holds for every $r \in R$,

$$\sum_{i\in\mathcal{I}_r} E_i + d_i \stackrel{d}{=} \lim_{|\mathcal{I}_r|\to\infty} N(\bar{\mu}, \bar{v}) \tag{13}$$

where $\stackrel{d}{=}$ is used to denote equality in terms of distribution [28], $|\cdot|$ is used to denote cardinality and

$$\bar{\mu} = \sum_{i\in\mathcal{I}_r} \mu_i^E + d_i,$$

$$\bar{v} = \sum_{i\in\mathcal{I}_r} v_i^E.$$

So, for this example, by substituting (13) in (12), only the mean values and variances of E_i are needed to solve the problem, as long as $|\mathcal{I}_r|$ is large enough. How large $|\mathcal{I}_r|$ needs to be for the approximation to be accurate varies, but for the distributions considered in this paper the number is not so big. For example: if all E_i are uniformly distributed with similar mean values and variances, the approximation is accurate enough for $|\mathcal{I}_r| = 4$. For more complex robot stations the result presented above does not hold, but for the type of robot stations considered in this paper it has been found to be accurate enough to be used as a heuristic.

Finally, by treating $S_i + E_i$ as the sum of normally distributed random variables, (7) can be approximated by

$$\mathbf{c}_i = \mathbf{s}_i + \mathbf{e}_i + \begin{bmatrix} d_i \\ 0 \end{bmatrix}, \tag{14}$$

where $\mathbf{c}_i, \mathbf{s}_i, \mathbf{e}_i$ are two-dimensional vectors containing the mean values and variances of the completion time, starting time, and execution time, respectively.

3.2. Independence

To express Optimization Problem 1 in a deterministic way, it is assumed that the stochastic variables involved are independent from each other. In general, that does not hold, but the optimization problem can be reformulated such that this independency property is satisfied. Similar approaches has already been presented, for example by Möhring [22]. But for completeness, this reformulation is shown in detail for the specific type of problem considered in this paper.

Combining the left-hand side of (10) with (9) it can be rewritten as

$$\mathrm{P}(T < t_d) = \mathrm{P}\Big(\bigcap_{i\in\mathcal{L}} C_i < t_d\Big), \tag{15}$$

and expanded using the chain rule (which can be derived using Bayes formula [32]),

$$\mathrm{P}(T < t_d) = \prod_{i\in\mathcal{L}} \mathrm{P}\Big(C_i < t_d \Big| \bigcap_{\ell\in\mathcal{L}_i} C_\ell < t_d\Big), \tag{16}$$

where $\mathrm{P}(\cdot|\cdot)$ is used to denote conditional probability and $\mathcal{L}_i = \{\ell \in \mathcal{L} : \ell > i\}$. Assuming that every C_i is independent from each other, the conditioning in (16) can be removed:

$$\prod_{i\in\mathcal{L}} \mathrm{P}\Big(C_i < t_d \Big| \bigcap_{\ell\in\mathcal{L}_i} C_\ell < t_d\Big) \xrightarrow{\text{independence}} \prod_{i\in\mathcal{L}} \mathrm{P}(C_i < t_d).$$

In general some of the $C_i \in \{C_\ell : \ell \in \mathcal{L}\}$ are not independent; how they depend on each other is discussed below.

Every $C_i \in \{C_\ell : \ell \in \mathcal{L}\}$ can be expressed as a max-function on a set of sums of execution times, so that

$$C_i = \max\left\{\sum_{\ell\in\mathcal{I}_j} E_\ell \, : \, j \in \mathcal{J}_i\right\} \tag{17}$$

where j is a "path" from a starting operation to operation i, $\mathcal{J}_i$ is the set of all such possible paths, and $\mathcal{I}_j$ is the set of operations contained in path j. For example, in Fig. 1, $C_3 = \max(E_1 + E_2 + E_3, E_4 + E_2 + E_3)$ and $C_5 = \max(E_4 + E_5, E_6 + E_7 + E_5)$. Examining their dependency, C_3 and C_5 are positively correlated, because they both contain E_4. In general, looking at (17) and comparing any pair of $C_i \in \{C_\ell : \ell \in \mathcal{L}\}$, it is clear that they will either be positively correlated if they share one or more E_i or independent if they do not; they will never be negatively correlated. Because of this, the following identity holds

$$\mathrm{P}\Big(C_i < t_d \Big| \bigcap_{\ell\in\mathcal{L}_i} C_\ell < t_d\Big) \geq \mathrm{P}(C_i < t_d) \qquad i \in \mathcal{L},$$

and it follows that

$$\prod_{i\in\mathcal{L}} \mathrm{P}\Big(C_i < t_d \Big| \bigcap_{\ell\in\mathcal{L}_i} C_\ell < t_d\Big) \geq \prod_{i\in\mathcal{L}} \mathrm{P}(C_i < t_d). \tag{18}$$

Combining (16) with (18), it is clear that if

$$\prod_{i\in\mathcal{L}} \mathrm{P}(C_i < t_d) \geq \beta, \tag{19}$$

is satisfied, (10) will also be satisfied. Hence, using (19) instead of (9) and (10) when solving Optimization Problem 1 will result in a solution that is feasible also for the original problem.

Proposition 1. *Assuming normal distributions of the stochastic variables in (9), approximating (9) and (10) in Optimization Problem 1 with*

$$\prod_{i\in\mathcal{L}} \Phi(t_d; \mathbf{c}_i) \geq \beta,$$

results in an optimization problem whose feasible solutions will be feasible for the original optimization problem as well.

Proof. This proposition is proved by combining (19) and the derivation of it with the definition of the CDF of a normal distribution (5). □

3.3. Max approximation

The last step is to find an approximation of (8) that can be expressed with mean values and variances. This will be done first for the case when $|\mathcal{W}_i(\mathcal{A})| = 2$ and then be extended to the case when $|\mathcal{W}_i(\mathcal{A})| > 2$. If $|\mathcal{W}_i(\mathcal{A})| = 1$, it means that only one operation precedes operation i. Then (8) simplifies to $\mathbf{s}_i = \mathbf{c}_\ell$, where ℓ is the operation preceding operation i.

Let x and y be two independent operations with normally distributed completion times defined by $C_x \sim N(\mu_x, v_x)$ and $C_y \sim N(\mu_y, v_y)$ respectively, and with vectors $\mathbf{c}_x = [\mu_x \; v_x]^T$ and $\mathbf{c}_y = [\mu_y \; v_y]^T$. Let z be an operation succeeding x and y. Its starting time is defined as

$$S_z = \max(C_x, C_y), \tag{20}$$

with mean μ_z and variance v_z. The PDF f_z^S of S_z is given by

$$f_z^S(z) = \phi(z; \mathbf{c}_x)\Phi(z; \mathbf{c}_y) + \phi(z; \mathbf{c}_y)\Phi(z; \mathbf{c}_x).$$

This equation can be derived intuitively. If Z is to take the value z, then either X or Y must take the value z, the relative likelihood of which is calculated by $\phi(z; \mathbf{c}_x)$ or $\phi(z; \mathbf{c}_y)$. At the same time, the one that does not take the value z must take a value that is smaller than z, the probability of which is calculated by $\Phi(z; \mathbf{c}_x)$ or $\Phi(z; \mathbf{c}_y)$.

Expressions for calculating μ_z and v_z will be derived below. μ_z is calculated as

$$\begin{aligned}\mu_z &= \int_{-\infty}^{\infty} f_z(z) z \mathrm{d}z \\ &= \int_{-\infty}^{\infty} \phi(z; \mathbf{c}_x)\Phi(z; \mathbf{c}_y) z \mathrm{d}z + \int_{-\infty}^{\infty} \phi(z; \mathbf{c}_y)\Phi(z; \mathbf{c}_x) z \mathrm{d}z. \end{aligned}\tag{21}$$

The expression in (21) contains two copies of the same function (with different arguments for μ and v). Hence, only one of them will be derived. Define a function $\bar{\mu}(\mathbf{c}_x, \mathbf{c}_y)$

$$\begin{aligned}\bar{\mu}(\mathbf{c}_x, \mathbf{c}_y) &= \int_{-\infty}^{\infty} \phi(z; \mathbf{c}_x)\Phi(z; \mathbf{c}_y) z \mathrm{d}z \\ &= \int_{-\infty}^{\infty} \frac{e^{-\frac{(z-\mu_x)^2}{2v_x}}\left(1 + \mathrm{erf}\left(\frac{z-\mu_y}{\sqrt{2v_y}}\right)\right) z}{2\sqrt{2\pi v_x}} \mathrm{d}z,\end{aligned} \tag{22}$$

so that (21) can be written as

$$\mu_z = \bar{\mu}(\mathbf{c}_x, \mathbf{c}_y) + \bar{\mu}(\mathbf{c}_y, \mathbf{c}_x). \tag{23}$$

The integration in (22) cannot be performed directly. Instead, the expression has to be transformed to a product of Gaussian functions, by taking the derivative and integral with respect to μ_y [33]

$$\bar{\mu} = \int \frac{\mathrm{d}}{\mathrm{d}\mu_y} \int_{-\infty}^{\infty} \frac{e^{-\frac{(z-\mu_x)^2}{2v_x}}\left(1 + \mathrm{erf}\left(\frac{z-\mu_y}{\sqrt{2v_y}}\right)\right) z}{2\sqrt{2\pi v_x}} \mathrm{d}z \mathrm{d}\mu_y.$$

The order of differentiation and integration can be changed, and performing the differentiation results in the following product of Gaussian functions, which now can be solved.

$$\bar{\mu} = \int \int_{-\infty}^{\infty} \frac{e^{-\frac{(z-\mu_x)^2}{2v_x} - \frac{(z-\mu_y)^2}{2v_y}} z}{2\pi\sqrt{v_x v_y}} \mathrm{d}z \mathrm{d}\mu_y.$$

Performing the integrations result in

$$\bar{\mu} = -\frac{1}{2}\mu_x \mathrm{erf}\left(\frac{\mu_y - \mu_x}{\sqrt{2}\sqrt{v_x + v_y}}\right) + \frac{v_x e^{-\frac{(\mu_x-\mu_y)^2}{2(v_x+v_y)}}}{\sqrt{2\pi}\sqrt{v_x + v_y}} + k_{\mu 1}.$$

where $k_{\mu 1}$ is the integration constant. Using (23), the whole expression for μ_z then becomes

$$\mu_z = \frac{\mu_x - \mu_y}{2}\mathrm{erf}\left(\frac{\mu_x - \mu_y}{\sqrt{2}\sqrt{v_x + v_y}}\right) + \frac{e^{-\frac{(\mu_x-\mu_y)^2}{2(v_x+v_y)}}\sqrt{v_x + v_y}}{\sqrt{2\pi}} + k_\mu, \tag{24}$$

where k_μ is the total integration constant. By looking at the edge case when $\mu_x \gg \mu_y, v_x, v_y$ in (24), we find that $\mu_z = \frac{\mu_x - \mu_y}{2} + k_\mu$. At the same time it is clear from a physical interpretation of the problem that the mean value of $Z = \max(X, Y)$ is equal to the mean value of X, if the mean value of X is much greater than the mean value of Y, and the variances of X and Y are small enough so that $\mathrm{P}(X > Y) \approx 1$. In other words, if $\mu_x \gg \mu_y, v_x, v_y$, then (24) should simplify to $\mu_z = \mu_x$; so $k_\mu = \frac{\mu_x + \mu_y}{2}$.

The variance of S_z, v_z is defined as,

$$\begin{aligned} v_z &= \int_{-\infty}^{\infty} f_z(z) z^2 \mathrm{d}z - \mu_z^2 \\ &= \int_{-\infty}^{\infty} \phi(z; \mathbf{c}_x)\Phi(z; \mathbf{c}_y) z^2 \mathrm{d}z + \int_{-\infty}^{\infty} \phi(z; \mathbf{c}_y)\Phi(z; \mathbf{c}_x) z^2 \mathrm{d}z - \mu_z^2, \end{aligned}$$

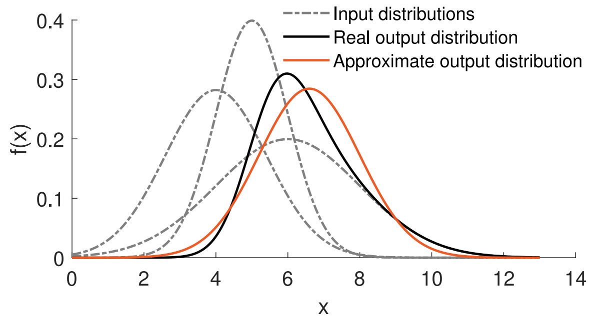


Fig. 3. An example of the inputs and output of a max-function, compared to the approximate distribution found by using H in (26).

where μ_z is the mean calculated according to (24). The two integrations can be solved in a similar way as for (21), resulting in

$$v_z = \frac{\left(\mu_x^2 + v_x - \mu_y^2 - v_y\right)}{2} \operatorname{erf}\left(\frac{\mu_x - \mu_y}{\sqrt{2}\sqrt{v_x + v_y}}\right) + \frac{(\mu_x + \mu_y)\sqrt{v_x + v_y}e^{-\frac{(\mu_x - \mu_y)^2}{2(v_x + v_y)}}}{\sqrt{2\pi}} + k_v - \mu_z^2, \tag{25}$$

where k_v is the integration constant. It is found in a similar way as for k_μ. By letting $\mu_x \gg \mu_y, v_x, v_y$ in (25), we find that $v_z = \frac{(\mu_x^2 + v_x - \mu_y^2 - v_y)}{2} + k_v - \mu_x^2$. Based on the physical interpretation of the problem it should result in $v_z = v_x$, which finally gives $k_v = \frac{\mu_x^2 + v_x + \mu_y^2 + v_y}{2}$.

The result from (24) and (25) can be expressed as

$$\mathbf{s}_z = h(\mathbf{c}_x, \mathbf{c}_y)$$

where h is a function that calculates μ_z using (24) and v_z using (25), and $\mathbf{s}_z = [\mu_z \ v_z]^T$. This function is well defined as long as $v_x + v_y \neq 0$ and is nonconvex.

In the general case, when $|\mathcal{W}_i(\mathcal{A})| > 2$, (8) can be expressed recursively as $S_i = \max(C_1, \max(C_2, \max(C_3, \ldots)))$. The h-function can be used in a similar way, by defining a function H as

$$H(\{\mathbf{c}_\ell : \ell \in \mathcal{W}_i(\mathcal{A})\}) = h(\mathbf{c}_1, h(\mathbf{c}_2, h(\mathbf{c}_3, \ldots))). \tag{26}$$

The output of H is a two-dimensional vector, which is an approximation of $\mathbf{s}_i = [\mu_i^S \ v_i^S]$, i.e. the mean and variance of S_i. S_i is in general not normally distributed, but approximating it as $N(\mu_i^S, v_i^S)$, the following proposition can be formulated.

Proposition 2. *Constraint (8) in Optimization Problem 1 can be approximated as*

$$\mathbf{s}_i = H(\{\mathbf{c}_l : l \in \mathcal{W}_i(\mathcal{A})\}),$$

where the set function $H(\cdot)$ is defined in (26). ☐

In Fig. 3, an example of the input distributions and output distribution of a real max-function is shown, compared to the approximate normal output distribution whose mean and variance is obtained by using H.

Below follows a short discussion on the potential errors of using Proposition 2. For ideal conditions, i.e. when $\{C_\ell : \ell \in \mathcal{W}_i(\mathcal{A})\}$ are independent and normally distributed, Proposition 2 is completely accurate for $|\mathcal{W}_i(\mathcal{A})| = 1$. For $|\mathcal{W}_i(\mathcal{A})| = 2$, the calculated values in $\mathbf{s}_i$ are the real mean and variance of S_i. However, using $\mathbf{s}_i$ to approximate S_i as a normal distribution, which is done implicitly by using Proposition 2, is not always accurate. For $|\mathcal{W}_i(\mathcal{A})| > 2$, also the calculated values in $\mathbf{s}_i$ are approximations.

Additional errors may be introduced for non-ideal conditions, i.e. when $\{C_\ell : \ell \in \mathcal{W}_i(\mathcal{A})\}$ are not independent and not normally distributed, which is the case in practice. How big the errors are depend on a large number of factors, such as the size of $|\mathcal{W}_i(\mathcal{A})|$, how the elements in $\{C_\ell : \ell \in \mathcal{W}_i(\mathcal{A})\}$ depend on each other and how the means and variances of these elements are distributed. An example where Proposition 2 will introduce large errors is when $\{C_\ell : \ell \in \mathcal{W}_i(\mathcal{A})\}$ have similar means but large differences in variances. This case is illustrated in Fig. 3.

Despite the potential errors, Proposition 2 is still useable for the applications considered in this paper. If one of the C_ℓ has a mean and variance such that the probability of it being larger than the others is close to one, then (8) will behave as if $|\mathcal{W}_i(\mathcal{A})| = 1$, for which Proposition 2 introduces no errors. It turns out that this is the case for many of the operations (see Section 5). But for which of the operations (8) will behave in this simplified way depends on the optimization result. So, it is in general not possible to simplify (8) prior to the optimization, and the full formulation in Proposition 2 is still needed.

Using Eq. (14) and Propositions 1 and 2, Optimization Problem 1 can be expressed in an approximate way, using only the means and variances of the stochastic variables. For a fixed operation order $\mathcal{A}$, the complete approximation of Optimization Problem 1 can then formulated as follows:

Optimization Problem 2.

$$\min_{d} \sum_{i \in \mathcal{I}} g_i(d_i)$$

subject to:

$$\mathbf{c}_i = \mathbf{s}_i + \mathbf{e}_i + \begin{bmatrix} d_i \\ 0 \end{bmatrix} \qquad i \in \mathcal{I}$$

$$\mathbf{s}_i = H(\{\mathbf{c}_\ell : \ell \in \mathcal{W}_i(\mathcal{A})\}) \qquad i \in \mathcal{I} \tag{27}$$

$$\prod_{i \in \mathcal{L}} \Phi(t_d; \mathbf{c}_i) \geq \beta \tag{28}$$

$$\underline{d_i} \leq d_i \qquad i \in \mathcal{I}$$

where the optimization variables are d_i. ☐

The complexity of solving Optimization Problem 2 varies depending on the type of robot station, but in general Optimization Problem 2 is not easy to solve, because of Φ and H that are nonconvex.

4. Operation order

Optimization Problem 2 assumes that the operation order is fixed. If that is not the case, i.e. there exist resource constraints (3), some additions are needed. An approach would be to find the operation order by solving a separate optimization problem that does not include the energy use or stochastic variations and then use that solution to solve Optimization Problem 2, and perhaps do this iteratively. In this paper a simple version of that idea will be used for comparison. It can be stated as:

Optimization Problem 3. Find the operation order that minimizes the makespan of the deterministic problem, i.e. where the stochastic variables of the execution times E_i are replaced with the mean values μ_i^E and the values of the deterministic execution times d_i are on their lower bounds $\underline{d_i}$. Then use this operation order to solve Optimization Problem 2. ☐

However, solving the scheduling problem and the stochastic energy reduction problem separately may result in the solution being suboptimal. In this section an approach to include the operation order into Optimization Problem 2 is presented. First, $\mathcal{R}_i$ is the set of resources required by operation i to execute; it can contain more than one element. Every resource $r \in \mathcal{R}_i$ has a set $\mathcal{K}_r$ of time slots $k \in \mathcal{K}_r$ in which the operations can be executed, where $\mathcal{K}_r = \{1, 2, \ldots, |\mathcal{I}_r|\}$ and $\mathcal{I}_r$ is the set of operations that requires resource r. Let $\mathbf{q}_{rk}$ be the completion time of the operation being processed on resource r and in time slot k. Let $a_{irk} \in \mathcal{A}$ be a binary decision variable that is 1 if operation i is processed on resource r in time slot k and 0 otherwise. The completion time $\mathbf{q}_{rk}$ can then be expressed as

$$\mathbf{q}_{rk} = \sum_{i \in \mathcal{I}_r} a_{irk}\mathbf{c}_i. \tag{29}$$

Instead of (27), the expression for $\mathbf{s}_i$ now becomes

$$\mathbf{s}_i = H(\mathcal{G}_i \cup \mathcal{Q}_i) \tag{30}$$

where

$$\mathcal{Q}_i = \{\mathbf{c}_l : l \in \mathcal{U}_i\} \tag{31}$$

is the contribution from the precedence constraints, and

$$\mathcal{G}_i = \left\{ \sum_{k \in \mathcal{K}_r \setminus \{1\}} \mathbf{q}_{r(k-1)} a_{irk} \ : \ r \in \mathcal{R}_i \right\} \tag{32}$$

is the contribution from the resource constraints. For a given operation order, $\mathcal{G}_i$ will contain the completion times of the operations that are directly preceding operation i on every resource r that is required by resource i. (32) is accurate if no operations share more than one resource. The constraints

$$\sum_{k \in \mathcal{K}_r} a_{irk} = 1 \qquad i \in \mathcal{I}, r \in \mathcal{R}_i \tag{33}$$

$$\sum_{i \in \mathcal{I}_r} a_{irk} = 1 \qquad r \in \mathcal{R}, k \in \mathcal{K}_r \tag{34}$$

ensure that every operation is only scheduled once and that only one operation can be executed by a resource at a time. Note that for some resources there is only one possible operation order, because of precedence constraints. Resource constraints for these resources do not need to be included in the optimization problem. For example, in Fig. 1 only the resource constraint of the shared resource needs to be included.

Constraint (28) also needs to be modified if the operation order is no longer fixed. For every operation $i \in \mathcal{L}$, $\tilde{a}_i$ is defined as

$$\tilde{a}_i = \prod_{r \in \mathcal{R}_i} a_{ir|\mathcal{I}_r|}, \tag{35}$$

so that $\tilde{a}_i$ is 1 only if operation i is the last operation on all of its resources. Eq. (28) can then be modified as

$$\prod_{i \in \mathcal{L}} ((1 - \tilde{a}_i) + \tilde{a}_i \Phi(t_d; \mathbf{c}_i)) \geq \beta. \tag{36}$$

The expression $((1 - \tilde{a}_i) + \tilde{a}_i \Phi(t_d; \mathbf{c}_i))$ becomes $\Phi(t_d; \mathbf{c}_i)$ if $\tilde{a}_i = 1$ and 1 if $\tilde{a}_i = 0$. In other words, an operation will only affect the calculations of the probability to meet the deadline if it does not have any operations succeeding it.

4.1. Deterministic optimization formulation

Together with the expressions in (26), (31), (32) and (35), the complete deterministic approximation of Optimization Problem 1 is given below:

Optimization Problem 4.

$$\min \sum_{i \in \mathcal{I}} g_i(d_i) \tag{37}$$

subject to:

$$\mathbf{c}_i = \mathbf{s}_i + \mathbf{e}_i + \begin{bmatrix} d_i \\ 0 \end{bmatrix} \qquad i \in \mathcal{I} \tag{38}$$

$$\mathbf{s}_i = H(\mathcal{G}_i \cup \mathcal{Q}_i) \qquad i \in \mathcal{I} \tag{39}$$

$$\mathbf{q}_{rk} = \sum_{i \in \mathcal{I}_r} a_{irk}\mathbf{c}_i \qquad r \in \mathcal{R}, k \in \mathcal{K}_r \tag{40}$$

$$\sum_{k \in \mathcal{K}_r} a_{irk} = 1 \qquad i \in \mathcal{I}, r \in \mathcal{R}_i \tag{41}$$

$$\sum_{i \in \mathcal{I}_r} a_{irk} = 1 \qquad r \in \mathcal{R}, k \in \mathcal{K}_r \tag{42}$$

$$\prod_{i \in \mathcal{L}} ((1 - \tilde{a}_i) + \tilde{a}_i \Phi(t_d; \mathbf{c}_i)) \geq \beta \tag{43}$$

$$\mathbf{c}_i \leq \begin{bmatrix} t_d \\ (t_d/3)^2 \end{bmatrix} \qquad i \in \mathcal{L} \tag{44}$$

$$\underline{d}_i \leq d_i \qquad i \in \mathcal{I} \tag{45}$$

$$a_{irk} \in \{0, 1\} \qquad i \in \mathcal{I}, r \in \mathcal{R}_i, k \in \mathcal{K}_r \tag{46}$$

where the optimization variables are $\mathbf{s}_i$, $\mathbf{q}_{rk}$, a_{irk} and d_i. □

Table 2
Parameters and solution of the simple example shown in Fig. 1.

Setup				Result	
i	$\underline{d}_i$	g_i $(\psi_1^i, \psi_2^i, \psi_3^i, \psi_4^i)$	E_i	d_i	$\mathcal{V}_i$
4	0	1000, −0.5, 500, 0.4	N(5.5, 0.4)	1.226	{}
5	0	2000, −0.5, 500, −0.2	N(5, 0.4)	1.018	{7}
6	0	1000, −0.5, 500, 0.4	N(5, 0.4)	0	{}
7	0	2000, −0.5, 500, −0.2	N(5, 0.4)	1.226	{}

Note that only d_i and a_{irk} are needed to completely describe the solution, since it is assumed that every operation is starting as early as possible. Also note that (44) is not strictly necessary to formulate the problem. It is added to prevent the solver from getting stuck at an unfeasible solution with zero gradient. This may happen since the error function $\mathrm{erf}(x)$ has a gradient close to zero for $x \gg 0$. Optimization Problem 4 is a nonlinear and nonconvex mixed integer programming problem. It is modeled in CasADi [34] and solved using the branch-and-bound algorithm of BONMIN [35] together with IPOPT [36]. Since the problem is nonconvex the solution from BONMIN is not exact but a heuristic, and the solution depends heavily on the starting point. Therefore, the result might be improved if the optimization is rerun a number of times with different random starting points. How many that is suitable depends on the application, how long timeframe that is available for the optimization and how good the solution needs to be. For the numerical examples in Section 5 the best solution of five optimization runs with different random starting points is used.

5. Numerical examples

In this section Optimization Problems 3 and 4 are applied to some numerical examples of robot stations.

5.1. Simple example

The first example is a simple one to show the basic principles. It is the robot station shown in Fig. 1 but without Robot 1 and Operations 1, 2 and 3. The precedence and resource constraints are shown in that figure, and the rest of the parameters can be found in the left side of Table 2. Solving this with $t_d = 20$ and $\gamma = 0.99$ using Optimization Problem 4 gives 5802 as the value of the cost function (37) and the rest of the solution is shown in the right side of Table 2.

Below a theoretical derivation of the optimal solution is shown to compare with the solution of Optimization Problem 4. First, there are two possible operation orders, Operation 5 preceding Operation 7 or the other way around, which will be denoted Operation orders 57 or 75 respectively. Studying the energy functions, $g_4(d) = g_6(d)$, and they have minima at $d = 1.018$. For Operation order 57 there is enough room in the schedule after Operation 6 for it to have $d_6 = 1.018$ and achieve the minimum energy use without affecting the starting time of Operation 7. The same is true for Operation order 75 but for Operation 4 and 5 instead. This means that the distribution that is relevant regarding meeting of the deadline (43) is either $E_4 + E_5 + E_7$ or $E_6 + E_7 + E_5$ for operation orders 57 or 75 respectively. These sums of distributions and sets of energy functions of the corresponding operations are the same apart from the former having a greater mean value of the sum of distributions. This means that operation order 75 is the best choice because that leaves more room in the schedule to use for energy reduction. With this operation order the deadline constraint (43) becomes

$$\Phi(20, \mu_6^E + d_6 + \mu_7^E + d_7 + \mu_5^E + d_5, v_6^E + v_7^E + v_5^E) \geq 0.99. \quad (47)$$

Using the values of μ and v from Table 2 to solve (47) (with = instead of ≥) for d_6+d_7+d_5 gives an upper limit on d_6+d_7+d_5 that is approximately equal to 2.452. Comparing the energy functions $g_6(d), g_7(d)$ and $g_5(d)$: because the gradients of $g_5(d)$ and $g_7(d)$ are negative and more negative than $g_6(d)$ for any d below 2.452 it is clear that $d_6 = 0$ in the optimal solution. Furthermore, since $g_5(d) = g_7(d)$ and that their gradients become less negative for increasing values of d, the optimal solution is $d_5 = d_7 = 2.452/2$. The value of the cost function at the optimal solution is therefore $g_4(1.018) + g_5(1.226) + g_6(0) + g_7(1.226) = 5802$. To summarize, this theoretical optimal solution is the same as the solution to Optimization Problem 4 presented earlier.

5.2. Problem setup

This section describes the setup of the main numerical examples. There are three of them and they are taken from a production line in Volvo Cars in Gothenburg, Sweden (see Fig. 4). The stations each consists of four ABB robots of the type IRB 6700. The robots primarily perform a series of spotwelds on the body of a car. Because the stations are used in production it was not possible to perform the energy optimization on the real stations. It was done in the simulation software Robotstudio from ABB, by using simulation models of the stations. It has been shown previously that the simulated energy use in Robotstudio is comparable to the actual measured energy use of a robot [26]. Time schedules of the simulation models can be seen in Figs. 5–7.

The operations of the robots can be divided into welding operations, that represents the actual welding, and movement operations, that represents the movements between the different welding locations. The welding operations cannot be affected, their energy use is not considered in this context and they have stochastic execution times. Most of the data in the numerical examples are taken from the real robot stations but detailed data about the distributions of the execution times of the welding operations were not available. An approximation of the execution time of a welding operation is 1.4 s and according to available data it can roughly wary ± 0.15 s of that. So, the distribution of the welding times are modeled as uniform distributions $U \sim (1.25, 1.55)$. The variation in execution time of the welding operations have not been investigated in this paper but one possible contributing factor is how clean the electrodes on the welding tool are. As the tool is used, dirt is accumulated on the electrodes which may increase how long it takes to complete a welding operation.

For the movement operations, their execution times can be controlled, they have no stochastic part and they use energy according to the type of energy functions described earlier and illustrated in Fig. 2. The energy functions are found in the way that is described in [26], by executing the operations with different velocity settings and recording the energy use; this is done in Robotstudio. Note that it is only the energy use of the movements that are included, not the energy use of the controllers or the brakes. In the original settings the robots are programmed to move with the maximum velocities, so their execution times are on the lower bounds.

The stations also contain shared zones. Two (groups of) operations requiring the same shared zone cannot be executed at the same time, and an operation may require more than one shared zone to be executed. In the context of this paper both the robots and the shared zones can be modeled as resources. Station 3 have additional constraints because it includes some additional operations other than welding. At one point Robot 2 picks up and places a part on the car body, which Robot 1 and 3 welds into place. Therefore, there are some precedence constraints between these robots, as indicated by the arrows in Fig. 7.

The task is then to reduce the energy use of these stations while not causing a delay in the production line. To create the exact optimization problem that will be solved for these stations there are some limitations

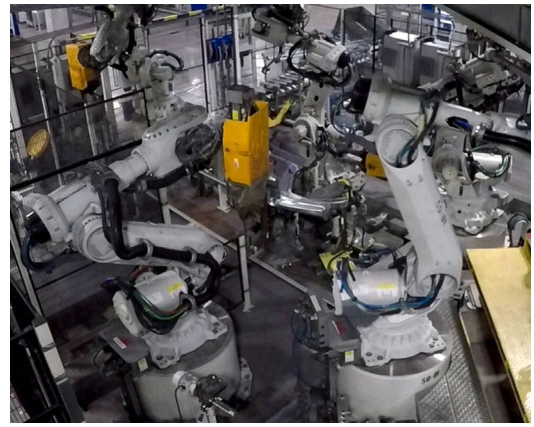


Fig. 4. Station 3, a welding station with four robots.

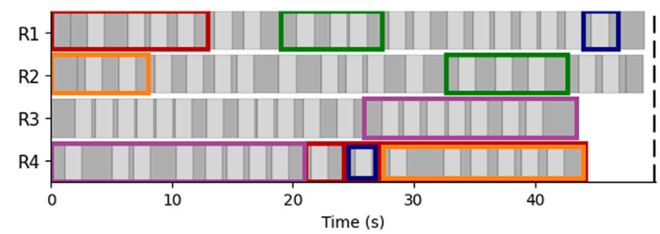


Fig. 5. A schedule of Station 1 with the original operation order and execution times. The robots are listed to the left and their respective operations are shown to the right. The light gray boxes are welding operations and the dark gray boxes are movement operations. Their execution times in the schedule are their mean values and lower bounds $\underline{d}_i$ respectively. The colored borders represent shared zones, one color per shared zone. The operations with the same color cannot be executed at the same time. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

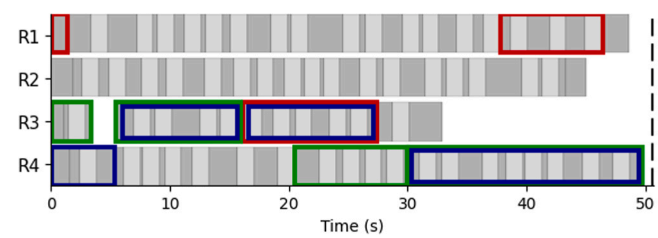


Fig. 6. A schedule of Station 2 with the original operation order and execution times.

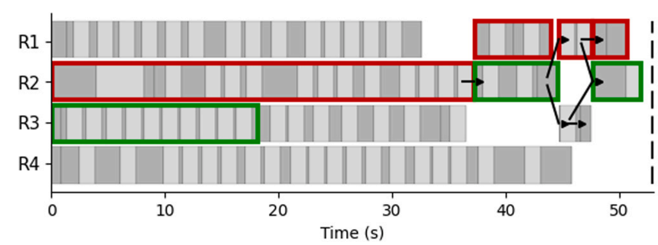


Fig. 7. A schedule of Station 3 with the original operation order and execution time. The arrows show precedence constraints between some of the operations.

and assumptions that depends both on practical limitations on what would be possible to change in the real stations and what makes for an interesting optimization problem. For example, it is assumed that the welding locations have been divided between the robots and that the paths between them have been set, because changing the paths introduces an unnecessary risk for collision. It is further assumed that the operation order is partly set. In the real stations the welding operations (and the movements between them) are divided in groups based on where on the car body they are operating. In this paper a group of operations is defined as follows. All operations in a group require the same resources and only the first and last operation in the group have precedence constraints with operations other than the ones in the same group. Finally, the operations in a group are next to each other in the original operation order (see Figs. 5–7). Based on this definition there are 16, 13, 12 groups in Station 1,2 and 3 respectively. It is assumed that the operation orders within each of these groups are fixed but not the order in which these groups are executed on the robots, and in which order they gain access to the shared zones, that is part of the optimization problem.

Based on the assumptions above, each group of operations can be modeled as one operation where the distribution E_i, the mean μ_i and the variance υ_i of the new operation simply is the sums of the distributions, mean values and variances of the individual operations in the group. The new energy function g_i of each new operation is found by solving the following optimization problem for some values of d_k where $\overline{d}_k \geq \sum_{i\in\mathcal{I}} \underline{d}_i$

$$
\begin{aligned}
& x_k := \min \sum_{i\in\mathcal{I}} g_i(d_i) \\
& \text{subject to:} \\
& \sum_{i\in\mathcal{I}} d_i \leq \overline{d}_k \\
& \underline{d}_i \leq d_i \qquad\qquad i \in \mathcal{I}
\end{aligned}
$$

The values of $\overline{d}_k$ and the corresponding optimal values of the optimization problem x_k are then used as data points and the new g_i is found by fitting functions to these data points in the same way as the original g_i were found. The station models using these merged operations will be referred to as the simplified station models, as opposed to the original station models shown in Figs. 5–7. They are the same apart from the number of operations. The simplified station models are what will be used in the optimization problem and the complete setup of them can be found in Tables 4–6 in Appendix.

5.3. Experimental setup

A total of 48 experiments have been conducted, each with a different combination of station, deadline and optimization method. The deadlines of the experiments were determined in the following way. The stations were simulated 100 000 times with the initial settings, i.e. $d_i = \underline{d}_i$ and the original operation orders. The deadlines that corresponded to 99.5 % of the simulation runs of the respective stations meeting them was then found. These deadlines were used in the experiments together with 7 higher values with 0.5 s difference between them. Both Optimization Problems 3 and 4 were used. Each experiment was conducted in the following way. The station was optimized using one of the optimization methods with $\beta = 0.99$. The resulting operation order and deterministic execution times were set, and the station was simulated using the Monte Carlo method 100 000 times. A simulation consists of drawing samples of the distributions and setting the starting times (and completion times) of the operations as low as possible while obeying the precedence constraints and operation order. The resulting data of the experiments are the energy use, which is the sum of the energy functions evaluated at the optimized deterministic execution times, and the probability to meet deadline, which is the fraction of the simulations where the deadline were met.

5.4. Results

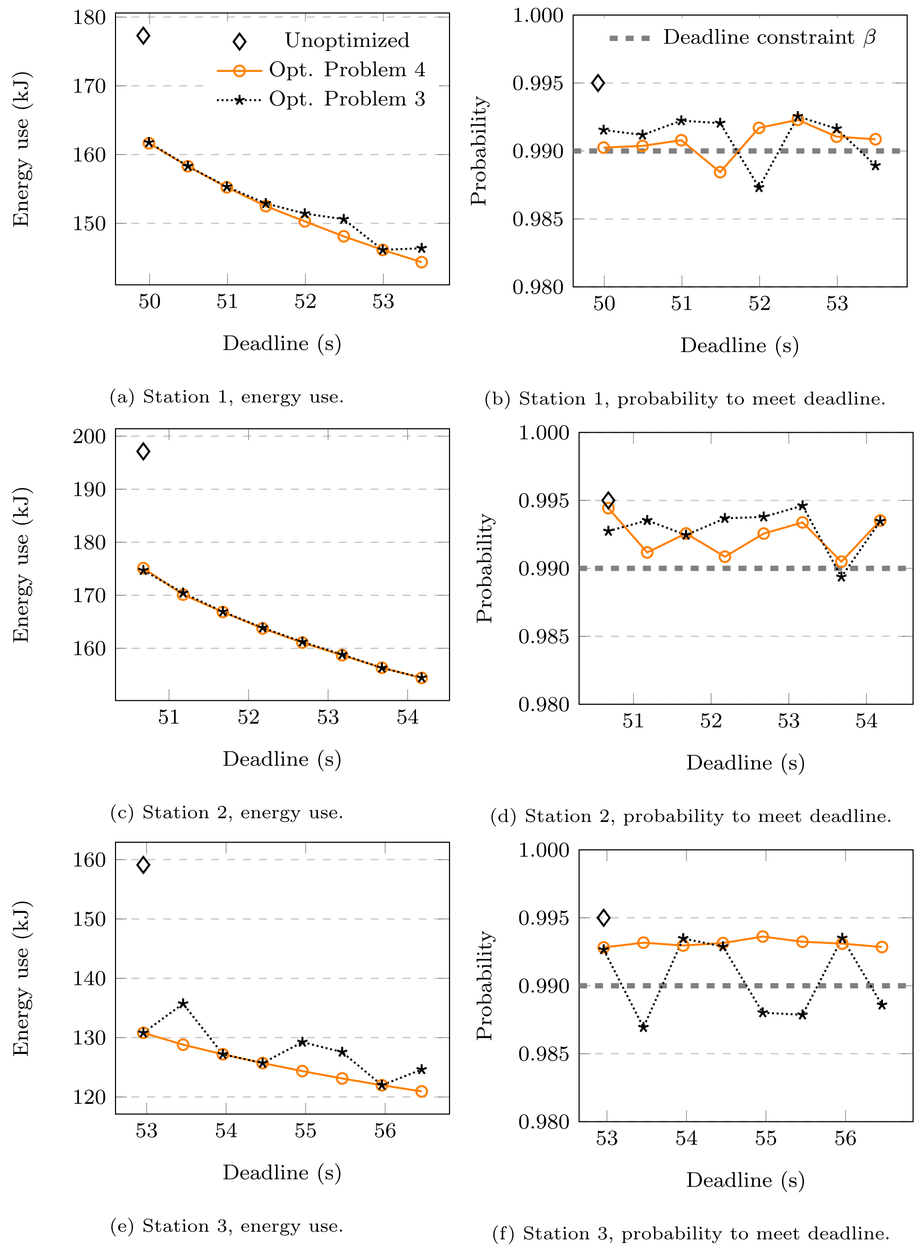
The results of the experiments can be seen in Fig. 8. Examples of solutions to Optimization Problem 4 can be found in Tables 4–6 in the Appendix and they are also shown graphically in Figs. 9, 10, 11 (after translating the results to the original station model).

The energy use of the unoptimized stations, when $d_i = \underline{d}_i$, are 177 kJ, 197 kJ and 159 kJ for Stations 1, 2 and 3 respectively. Comparing this with the optimized energy use shown in Figs. 8(a), 8(c) and 8(e) there are some interesting things to note. Firstly, the result shows that significant energy reduction can be made, even with the lowest deadlines. This energy reduction is roughly 9, 11 and 18 percent for the three stations respectively. The reason why the reduction is so big is because of a combination of the steep gradients of the energy functions around $\underline{d}_i$ (see Fig. 2) together with the "free time" (sometimes known as slack) in the schedule, which allows extensions of the execution times of the operations without causing an extension of the makespan of the station. Secondly, Station 3 has a bigger reduction than the other two. A contributing factor to this is that it has more slack than Station 1 and that its slack is more evenly divided between the robots than for Station 2. The fact that the gradients of the energy functions are steepest around $\underline{d}_i$ and also that the energy functions of the robots are relatively similar means that the total energy reduction is bigger if the slack is evenly divided between the robots. Thirdly, the energy use of each station is further reduced with around 2–3 percent per additional second that the deadline is extended (compared to the unoptimized energy use). However, this effect is declining as the deadline is increasing.

Comparing the two optimization methods: the simpler version sometimes finds the same solution as Optimization Problem 4 but not always. When there is a difference it is because of different operation orders. Specifically looking at Station 3 (see Fig. 7), the bad solutions are when the green operation group of Robot 3 are executed directly before the first green operation group of Robot 2, instead of being executed in the beginning. This prevents the slack in the schedule to be utilized fully, which illustrates a possible problem with separating the scheduling and energy reduction part of the optimization problem.

Figs. 8(b), 8(d) and 8(f) show the simulated probability to meet deadline for the optimized stations. The result shows that the probability to meet deadline is very close to, and most of the time above, the constraint of 0.99. This is a good result considering the heavy approximations used when deriving the optimization method and the fact that the distributions of the execution times of the operations are not normal (as modeled) but uniform. This is partly because there are enough operations and the uniform distribution is close enough to the normal distribution for the central limit theorem to hold. It is also partly because many of the max-functions (that exist because of precedence and resource constraints), are not "active". For example, looking at Fig. 5 and the first operation of Robot 1 using the blue zone (that temporarily will be denoted Operation x). Its starting time is the max-function of the completion times of the operations directly preceding it of Robot 1 (Operation y) and the last operation of Robot 4 using the blue zone (Operation z). Since the latter finishes significantly later than the former (and their variances are not unreasonably large), then $S_x \approx C_y$. So, for Operation x (and other similar operations) the approximation of the max function does not introduce any errors.

Looking at the examples of solutions shown in Figs. 9–11, it can be noted that the operations executed by robots that have large amounts of slack show the most energy reduction. It can also be noted that the execution time extensions are unevenly divided even between operations of one robot. It is the operation that have the largest execution time to begin with that shows the most execution time extensions. This is because they correspond to longer robot movements that use more energy and that therefore have the most energy reduction potential.



(a) Station 1, energy use.
(b) Station 1, probability to meet deadline.
(c) Station 2, energy use.
(d) Station 2, probability to meet deadline.
(e) Station 3, energy use.
(f) Station 3, probability to meet deadline.

Fig. 8. The simulated energy use and probability to meet deadline for each of the three stations. The data show a comparison between the stations before optimization and the stations after optimization using Optimization Problems 3 and 4.

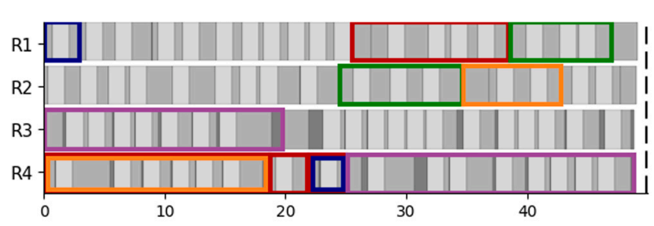


Fig. 9. A schedule of Station 1 with the optimized operation order and execution times for the lowest of the deadlines, c.f. Fig. 5. The darkest gray areas show the extensions of execution times of the movement operations compared to the lower bound $\underline{d}$.

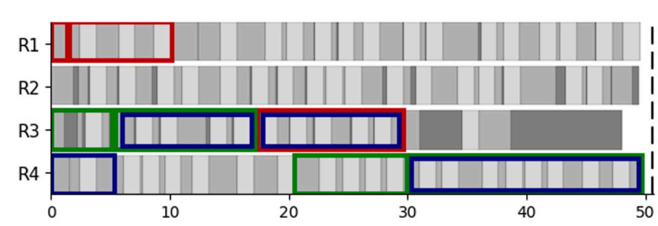


Fig. 10. A schedule of Station 2 with the optimized operation order and execution times for the lowest of the deadlines, c.f. Fig. 6.

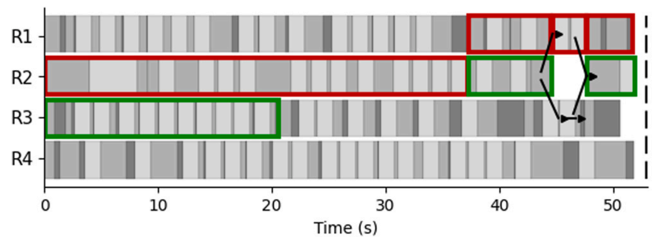


Fig. 11. A schedule of Station 3 with the optimized operation order and execution times for the lowest of the deadlines, c.f. Fig. 7.

6. Discussion and conclusion

In this paper, the stochastic optimization problem of reducing the energy use of a robot station subject to stochastic processing times has been investigated. A deterministic approximation of the stochastic optimization problem has been presented, and it has been verified on three numerical examples. The result showed that the proposed method was able to find solutions to all the problem instances and that the solutions resulted in reduced energy use and a high probably to meet deadline. The method is a heuristic and it cannot be guaranteed that it works within satisfying accuracy for a general robot station. However, the result shows that under some conditions, which we believe holds for many types of robot stations in practice, the proposed simplifications makes the problem possible to solve within reasonable accuracy. Because of the complexity of the problem, it was not possible to compare the obtained solutions with the true optimum of the initial optimization problem. However, since the proposed method was able to approximate the distributions of the makespans reasonably well (see Fig. 8 and the fact that the probability of the optimized stations were close to the constraint) the obtained solutions can be expected to be close to the true optima.

Future work includes improving the optimization model and method to not require so heavy approximations, making it faster to compute, finding a way to compare the obtained solutions to the true optima and testing the method on other types of robot stations.

CRediT authorship contribution statement

Mattias Hovgard: Conceptualization, Methodology, Software, Investigation, Writing – original draft, Writing – review & editing, Visualization. **Bengt Lennartson:** Conceptualization, Writing – review & editing, Supervision. **Kristofer Bengtsson:** Conceptualization, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research was funded by Vinnova FFI through the project SmoothIT and the ITEA3 project SPEAR. The support is gratefully acknowledged. The authors would also like to thank Volvo Cars in Gothenburg for providing us with the numerical examples.

Appendix

Detailed setups and results for the three stations.

Table 4
Detailed setup and result of the simplified model of Station 1 for $t_d = 49.992$.

Setup						Result	
i	$\mathcal{U}_i$	$\mathcal{R}_i$	$\underline{d}_i$	g_i $(\psi_i^i, \psi_2^i, \psi_3^i, \psi_4^i)$	E_i	d_i	$\mathcal{W}_i$
0	{}	{R1, z3}	7.512	30 960 000.0, −1.228, 11 510.0, 0.0007215	$\sum_{i=1}^{4} U(1.25, 1.55)$	7.512	{3, 14}
1	{}	{R1}	3.072	22 850 000.0, −3.11, 6373.0, −0.05055	$\sum_{i=1}^{2} U(1.25, 1.55)$	3.14	{4}
2	{}	{R1, z1}	4.296	172 400.0, −0.9024, 7882.0, −0.0123	$\sum_{i=1}^{3} U(1.25, 1.55)$	4.296	{0, 8}
3	{}	{R1}	5.28	19 970 000.0, −1.727, 4700.0, 0.002706	$\sum_{i=1}^{8} U(1.25, 1.55)$	5.28	{1}
4	{}	{R1, z7}	1.632	237 700.0, −3.764, 1728.0, 0.02348	$\sum_{i=1}^{1} U(1.25, 1.55)$	1.632	{}
5	{}	{R1}	2.04	24 280.0, −1.223, 3306.0, 0.003842		2.04	{2}
6	{}	{R2, z4}	5.4	44 140.0, −0.4559, 9285.0, 0.003057	$\sum_{i=1}^{2} U(1.25, 1.55)$	5.4	{15, 8}
7	{}	{R2}	11.76	210 500.0, −0.3037, 18 400.0, −0.003754	$\sum_{i=1}^{9} U(1.25, 1.55)$	11.83	{}
8	{}	{R2, z1}	5.976	78 070.0, −0.5097, 8511.0, 0.003581	$\sum_{i=1}^{3} U(1.25, 1.55)$	6.015	{7}
9	{}	{R2}	3.36	18 290.0, −0.6, 3266.0, 0.006462	$\sum_{i=1}^{2} U(1.25, 1.55)$	3.36	{6}
10	{}	{R3}	9.024	419 600.0, −0.454, 10 550.0, −0.002905	$\sum_{i=1}^{12} U(1.25, 1.55)$	12.16	{11}
11	{}	{R3, z5}	7.92	161 200.0, −0.449, 12 280.0, −0.0004844	$\sum_{i=1}^{7} U(1.25, 1.55)$	10.07	{}
12	{}	{R4, z5}	9.792	201 000.0, −0.3422, 14 500.0, −0.0003558	$\sum_{i=1}^{8} U(1.25, 1.55)$	12.56	{11, 14}
13	{}	{R4, z3}	1.896	66 870 000.0, −6.898, 1731.0, −0.04803	$\sum_{i=1}^{1} U(1.25, 1.55)$	1.907	{15}
14	{}	{R4, z3, z7}	1.416	482 200.0, −4.491, 1312.0, −0.004648	$\sum_{i=1}^{1} U(1.25, 1.55)$	1.718	{13, 4}
15	{}	{R4, z3, z4}	8.736	108 200.0, −0.3622, 11 070.0, −0.0005268	$\sum_{i=1}^{6} U(1.25, 1.55)$	10.23	{}

Table 5
Detailed setup and result of the simplified model of Station 2 for $t_d = 50.677$.

Setup						Result	
i	$\mathcal{U}_i$	$\mathcal{R}_i$	$\underline{d}_i$	g_i $(\psi_i^i, \psi_2^i, \psi_3^i, \psi_4^i)$	E_i	d_i	$\mathcal{W}_i$
0	{}	{R1, z9}	1.536	3524.0, −0.5987, 3876.0, 0.01876		1.536	{}
1	{}	{R1}	16.63	1 825 000.0, −0.3135, 27 030.0, −0.001908	$\sum_{i=1}^{14} U(1.25, 1.55)$	17.56	{3}
2	{}	{R1, z9}	4.584	566 600.0, −1.262, 6015.0, 0.0001171	$\sum_{i=1}^{3} U(1.25, 1.55)$	4.584	{0}
3	{}	{R1}	2.04	7924.0, −0.7211, 2605.0, 0.006942		2.04	{2}
4	{}	{R2}	21.19	236 700.0, −0.1223, 34 430.0, 0.0001344	$\sum_{i=1}^{17} U(1.25, 1.55)$	25.68	{}
5	{}	{R3, z2}	2.064	54 740.0, −1.58, 2741.0, −0.004668	$\sum_{i=1}^{1} U(1.25, 1.55)$	3.917	{}
6	{}	{R3, z2, z3}	5.016	3 505 000.0, −1.321, 7745.0, −0.01598	$\sum_{i=1}^{4} U(1.25, 1.55)$	6.265	{9, 5}
7	{}	{R3, z3, z9}	4.392	518 700.0, −1.166, 7473.0, −0.02741	$\sum_{i=1}^{5} U(1.25, 1.55)$	5.398	{2, 6}
8	{}	{R3}	4.032	10 910.0, −0.1617, 6485.0, 0.01252	$\sum_{i=1}^{1} U(1.25, 1.55)$	16.97	{7}
9	{}	{R4, z3}	4.032	116 300.0, −0.8353, 7719.0, −0.002558	$\sum_{i=1}^{1} U(1.25, 1.55)$	4.032	{}
10	{}	{R4}	6.624	2 615 000.0, −0.9252, 10 290.0, −0.004638	$\sum_{i=1}^{6} U(1.25, 1.55)$	6.624	{9}
11	{}	{R4, z2}	3.96	15 820 000.0, −2.358, 7921.0, −0.04329	$\sum_{i=1}^{4} U(1.25, 1.55)$	3.96	{10, 6}
12	{}	{R4, z2, z3}	8.496	902 300.0, −0.5832, 10 700.0, −0.001559	$\sum_{i=1}^{8} U(1.25, 1.55)$	8.496	{7, 11}

Table 6
Detailed setup and result of the simplified model of Station 3 for $t_d = 52.959$.

Setup						Result	
i	$\mathcal{U}_i$	$\mathcal{R}_i$	$\underline{d}_i$	g_i $(\psi_i^i, \psi_2^i, \psi_3^i, \psi_4^i)$	E_i	d_i	$\mathcal{W}_i$
0	{}	{R1}	12.98	943 000.0, −0.3696, 19 670.0, −0.002883	$\sum_{i=1}^{14} U(1.25, 1.55)$	17.46	{}
1	{}	{R1, z1}	4.152	61 890.0, −0.6411, 4291.0, −0.005965	$\sum_{i=1}^{2} U(1.25, 1.55)$	4.588	{4, 0}
2	{5}	{R1, z1}	0.192	0, 1368.0, 0, 1368.0	$\sum_{i=1}^{2} U(1.25, 1.55)$	0.192	{1, 5}
3	{2}	{R1, z1}	3.12	62 410.0, −1.037, 8479.0, −0.004496		4.135	{2}
4	{}	{R2, z1}	16.18	590 900.0, −0.2631, 20 920.0, 0.00101	$\sum_{i=1}^{15} U(1.25, 1.55)$	16.18	{}
5	{4}	{R2, z2}	4.68	175 400.0, −0.9639, 5813.0, −0.004209	$\sum_{i=1}^{2} U(1.25, 1.55)$	4.68	{7, 4}
6	{2, 9}	{R2, z2}	2.904	31 820.0, −0.8284, 7156.0, 0.005265	$\sum_{i=1}^{1} U(1.25, 1.55)$	2.904	{2, 5, 9}
7	{}	{R3, z2}	4.152	1 179 000.0, −1.415, 4465.0, −0.01134	$\sum_{i=1}^{10} U(1.25, 1.55)$	6.571	{}
8	{}	{R3}	8.496	474 100.0, −0.4971, 14 270.0, −0.003665	$\sum_{i=1}^{7} U(1.25, 1.55)$	14.8	{7}
9	{5}	{R3}	0.576	157 300.0, −9.409, 114.1, 0.6499	$\sum_{i=1}^{1} U(1.25, 1.55)$	0.9223	{8, 5}
10	{9}	{R3}	0.84	191 300.0, −5.723, 2268.0, −0.08994		3.096	{9}
11	{}	{R4}	19.18	1 331 000.0, −0.2699, 25 800.0, −0.001129	$\sum_{i=1}^{19} U(1.25, 1.55)$	25.19	{}

References

[1] M. Ratiu, M. Adriana Prichici, Industrial robot trajectory optimization- a review, MATEC Web Conf. 126 (2017) 02005, http://dx.doi.org/10.1051/matecconf/201712602005.

[2] G. Carabin, E. Wehrle, R. Vidoni, A review on energy-saving optimization methods for robotic and automatic systems, Robotics 6 (4) (2017) 39, http://dx.doi.org/10.3390/robotics6040039.

[3] Y. Zhao, Y. Wang, M.C. Zhou, J. Wu, Energy-optimal collision-free motion planning for multiaxis motion systems: an alternating quadratic programming approach, IEEE Trans. Autom. Sci. Eng. 16 (1) (2019) 327–338, http://dx.doi.org/10.1109/TASE.2018.2864773.

[4] A. Vergnano, C. Thorstensson, B. Lennartson, P. Falkman, M. Pellicciari, F.

Leali, S. Biller, Modeling and optimization of energy consumption in cooperative multi-robot systems, IEEE Trans. Autom. Sci. Eng. 9 (2) (2012) 423–428, http://dx.doi.org/10.1109/TASE.2011.2182509.

[5] M. Pellicciari, G. Berselli, F. Leali, A. Vergnano, A method for reducing the energy consumption of pick-and-place industrial robots, Mechatronics 23 (3) (2013) 326–334, http://dx.doi.org/10.1016/j.mechatronics.2013.01.013.

[6] D. Meike, M. Pellicciari, G. Berselli, Energy efficient use of multirobot production lines in the automotive industry: detailed system modeling and optimization, IEEE Trans. Autom. Sci. Eng. 11 (3) (2014) 798–809, http://dx.doi.org/10.1109/TASE.2013.2285813.

[7] M. Gadaleta, M. Pellicciari, G. Berselli, Optimization of the energy consumption of industrial robots for automatic code generation, Robot. Comput.-Integr. Manuf. 57 (2019) 452–464, http://dx.doi.org/10.1016/j.rcim.2018.12.020.

[8] N. Sundström, O. Wigström, B. Lennartson, Conflict between energy, stability, and robustness in production schedules, IEEE Trans. Autom. Sci. Eng. 14 (2) (2017) 658–668, http://dx.doi.org/10.1109/TASE.2016.2643621.

[9] A. Ben-Tal, L. El Ghaoui, A. Nemirovski, Robust Optimization, Princeton University Press, 2009.

[10] O. Wigstrom, B. Lennartson, A. Vergnano, C. Breitholtz, High-level scheduling of energy optimal trajectories, IEEE Trans. Autom. Sci. Eng. 10 (1) (2013) 57–64, http://dx.doi.org/10.1109/TASE.2012.2198816.

[11] G. Pastras, A. Fysikopoulos, G. Chryssolouris, A theoretical investigation on the potential energy savings by optimization of the robotic motion profiles, Robot. Comput.-Integr. Manuf. 58 (2019) 55–68, http://dx.doi.org/10.1016/j.rcim.2019.02.001.

[12] Z. Zhang, S. Chen, X. Zhu, Z. Yan, Two hybrid end-effector posture-maintaining and obstacle-limits avoidance schemes for redundant robot manipulators, IEEE Trans. Ind. Inform. 16 (2) (2020) 754–763, http://dx.doi.org/10.1109/TII.2019.2922694.

[13] Z. Zhang, S. Chen, J. Liang, Discrete-time circadian rhythms neural network for perturbed redundant robot manipulators tracking problem with periodic noises, IEEE Trans. Ind. Inform. 18 (1) (2022) 242–251, http://dx.doi.org/10.1109/TII.2021.3065715.

[14] E. Glorieux, S. Riazi, B. Lennartson, Productivity/energy optimisation of trajectories and coordination for cyclic multi-robot systems, Robot. Comput.-Integr. Manuf. 49 (2018) 152–161, http://dx.doi.org/10.1016/j.rcim.2017.06.012.

[15] M. Gadaleta, G. Berselli, M. Pellicciari, Energy-optimal layout design of robotic work cells: potential assessment on an industrial case study, Robot. Comput.-Integr. Manuf. 47 (2017) 102–111, http://dx.doi.org/10.1016/j.rcim.2016.10.002.

[16] G. Chen, L. Zhang, J. Arinez, S. Biller, Energy-efficient production systems through schedule-based operations, IEEE Trans. Autom. Sci. Eng. 10 (1) (2013) 27–37, http://dx.doi.org/10.1109/TASE.2012.2202226.

[17] Q. Chang, G. Xiao, S. Biller, L. Li, Energy saving opportunity analysis of automotive serial production systems (march 2012), IEEE Trans. Autom. Sci. Eng. 10 (2) (2013) 334–342, http://dx.doi.org/10.1109/TASE.2012.2210874.

[18] M.A. Salido, J. Escamilla, F. Barber, A. Giret, D. Tang, M. Dai, Energy efficiency, robustness, and makespan optimality in job-shop scheduling problems, Artif. Intell. Eng. Des. Anal. Manuf. 30 (3) (2016) 300–312, http://dx.doi.org/10.1017/S0890060415000335.

[19] M. Faraji Amiri, J. Behnamian, Multi-objective green flowshop scheduling problem under uncertainty: estimation of distribution algorithm, J. Clean. Prod. 251 (2020) 119734, http://dx.doi.org/10.1016/j.jclepro.2019.119734.

[20] S. Gürel, H. Gultekin, V.E. Akhlaghi, Energy conscious scheduling of a material handling robot in a manufacturing cell, Robot. Comput.-Integr. Manuf. 58 (2019) 97–108, http://dx.doi.org/10.1016/j.rcim.2019.02.002.

[21] P. Brucker, A. Drexl, R. Möhring, K. Neumann, E. Pesch, Resource-constrained project scheduling: notation, classification, models, and methods, European J. Oper. Res. 112 (1999) 3–41.

[22] R.H. Möhring, Scheduling under uncertainty: bounding the makespan distribution, in: H. Alt (Ed.), Computational Discrete Mathematics: Advanced Lectures, Springer Berlin Heidelberg, Berlin, Heidelberg, 2001, pp. 79–97, http://dx.doi.org/10.1007/3-540-45506-X_7.

[23] A. Nemirovski, A. Shapiro, Convex approximations of chance constrained programs, SIAM J. Optim. 17 (4) (2007) 969–996, http://dx.doi.org/10.1137/050622328.

[24] X. Lin, S.L. Janak, C.A. Floudas, A new robust optimization approach for scheduling under uncertainty: I. bounded uncertainty, Comput. Chem. Eng. 28 (6–7) (2004) 1069–1085, http://dx.doi.org/10.1016/j.compchemeng.2003.09.020.

[25] S.L. Janak, X. Lin, C.A. Floudas, A new robust optimization approach for scheduling under uncertainty. II. Uncertainty with known probability distribution, Comput. Chem. Eng. 31 (3) (2007) 171–195, http://dx.doi.org/10.1016/j.compchemeng.2006.05.035.

[26] M. Hovgard, B. Lennartson, K. Bengtsson, Applied energy optimization of multi-robot systems through motion parameter tuning, CIRP J. Manuf. Sci. Technol. 35 (2021) 422–430, http://dx.doi.org/10.1016/J.CIRPJ.2021.07.012.

[27] M. Hovgard, B. Lennartson, K. Bengtsson, Energy-optimal timing of robot stations subject to gaussian disturbances, in: 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), 2019, pp. 1441–1444, http://dx.doi.org/10.1109/ETFA.2019.8869250.

[28] X. Cai, X. Wu, X. Zhou, Optimal Stochastic Scheduling, Springer, Boston, MA, 2014, http://dx.doi.org/10.1007/978-1-4899-7405-1.

[29] M. Abramowitz, S. Irene, Handbook of Mathematical Functions, US Department of Commerce, 1972.

[30] E.W. Weisstein, Normal sum distribution, in: From MathWorld–A Wolfram Web Resource, URL https://mathworld.wolfram.com/NormalSumDistribution.html.

[31] E.W. Weisstein, Central limit theorem, in: From MathWorld–A Wolfram Web Resource, URL https://mathworld.wolfram.com/CentralLimitTheorem.html.

[32] M. Grigoriu, Stochastic Calculus, Birkhäuser, Boston, MA, 2002, http://dx.doi.org/10.1007/978-0-8176-8228-6.

[33] D. Constantin, Gaussian integral of an error function, 2012, URL http://blitiri.blogspot.com/2012/11/gaussian-integral-of-error-function.html.

[34] J.A.E. Andersson, J. Gillis, G. Horn, J.B. Rawlings, M. Diehl, CasADi – A software framework for nonlinear optimization and optimal control, Math. Program. Comput. 11 (1) (2019) 1–36, http://dx.doi.org/10.1007/s12532-018-0139-4.

[35] P. Bonami, L.T. Biegler, A.R. Conn, G. Cornuéjols, I.E. Grossmann, C.D. Laird, J. Lee, A. Lodi, F. Margot, N. Sawaya, A. Wächter, An algorithmic framework for convex mixed integer nonlinear programs, Discrete Optim. 5 (2) (2008) 186–204, http://dx.doi.org/10.1016/j.disopt.2006.10.011.

[36] A. Wächter, L.T. Biegler, On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming, Math. Program. 106 (2006) 25–57, http://dx.doi.org/10.1007/s10107-004-0559-y.