Robotics and Computer-Integrated Manufacturing 41 (2016) 115–126

Contents lists available at ScienceDirect

Robotics and Computer-Integrated Manufacturing

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

Reconfigurable industrial robots: A stochastic programming approach for designing and assembling robotic arms

A. Valente

SUPSI, Institute of Systems and Technologies for Sustainable Production, Galleria 2, Manno 6928, Switzerland

ARTICLE INFO

Article history:
Received 23 September 2014
Received in revised form
1 March 2016
Accepted 16 March 2016
Available online 5 April 2016

Keywords:
Reconfigurable Industrial Robotics
Stochastic programming

ABSTRACT

Industrial robots undergo design and re-configuration processes to target extremely challenging precision and reliability performance with agile and efficient architectures. The need for such features currently prevent the exploitation of reconfigurable robotics in manufacturing. The current work presents an approach to design and configure reconfigurable robots for the high precision manufacturing industry. The work proposes a configuration algorithm that enables the identification of the robot architectures and the related reconfigurability features by selecting the type and number of robot modules to be implemented over time in order to better accomplish a number of production requirements. Particularly, assuming the robot will work by utilising a finite set of robotic modules, the algorithm determines the set of modules to form the arm and the ones to be allocated in the robot storage for possible usage over time. Results show a number of benefits such as a robotic chain with customised reaching and degrees of freedom with a reduced cost by performing an accurate module selection and configuration; this should lead the robot users to prefer reconfigurable robots to commercial rigid catalogue solutions proposed by robot manufacturers.

1. Introduction

The reconfigurability feature for industrial robots has the potential to play an instrumental role in manufacturing context suffering rapid evolutions of the product geometric and technological features along with the changes in the production demand. Similarly to machine tools and other mechatronic equipment (fixturing and transportation systems), such robot capabilities constitute an instrumental leverage to efficiently adapt the change of requirements while addressing manufacturing objectives like the maximisation of parts yields and minimisation of resource consumption [1].

In the research and development fields related to robotics, the reconfigurability feature is mostly investigated in relation to applications for stochastic, unstructured, hostile and unpredicted environments such as social, space, underwater, oil&gas and military sectors; these applications are characterized by rapidly and unpredictably changing operating contexts which have revealed to take most advantage from the robot capability to adapt shape, operating modes and self-healing by substituting malfunctioning modules. An extensive literature review proposed in Section 2 will deeply investigate the recent advances in reconfigurable robotics.

The application of these concepts to the manufacturing field poses severe questions in relation to the need for accomplishing demanding precision and accuracy requirements associated to machining and assembly processes [2]. For example, the accuracy, robustness and reliability targets of industrial robots lead to neglect several nesting mechanisms adopted as mechatronic interfaces between the modules of reconfigurable robots presenting either a lattice or chain type structure (such as nesting mechanisms based on small mechanical hooks and electromagnetic devices).

Also, in a manufacturing environment a reconfigurable robot should rely upon a finite (predefined) set of modules which should enable the robot green field configuration and its re-equipment while shifting between groups of different tasks. This implies the need to determine all possible robotic modules which would be necessary for current and future manufacturing tasks, including the ones to be allocated in local storage ready to be used; this ensures the robot autonomy in managing multiple production batches over the time. Changes of product dimensions could require variable reaching space, degrees of freedom or a change of the kinematics structure and changes in the operations can demand for different end-effectors and control logic without sacrificing accuracy, repeatability or stiffness. As a result of this requirement, the research in reconfigurable robotics makes an instrumental use of such approaches which additionally allow the determination of the robot catalogue of modules.

The goal of this research work is to address the reconfigurability concept for industrial robots and propose a design approach to determine the robot configuration over time to ensure its

http://dx.doi.org/10.1016/j.rcim.2016.03.002
0736-5845/

capability to realise complex manufacturing tasks. The rest of the paper is organised as follows: Section 2 overviews the literature related to reconfigurable robots; Section 3 presents the robot configuration approach and Section 4 outlines the mathematical formulation of the model; Section 5 presents some preliminary results related to the application of the proposed approach to a real industrial case study; Section 6 outlines the conclusions and future work.

2. Literature review

Modular robots are conceived as the composition of multiple blocks with uniform interfaces allowing for the transfer of mechanical forces and torque, electrical power, and communication through the robot. Building blocks consist of some primary structural actuated unit and some additional specialized units such as grippers, vision systems and energy storage units [2].

The field of reconfigurable robots has seen significant progress over the last twenty years. The work by Stoy et al. [3] provides a comprehensive overview of the latest solutions and approaches related to self-reconfigurable robots while addressing the mechanical, control, configuration and functional monitoring challenges. Examples of major results in the field of reconfigurable robotics can be found in [4–13]. More recent works deal with: nesting mechanisms for robot modules [14–18]; direct and inverse kinematics [19–22]; communication and control [23–26]; motion planning [27, 28]; optimisation of robot performance [29, 30].

The adoption of modular and reconfigurable robots presents a number of advantages [2], such as:

- *Versatility*: reconfigurable robotic systems are more adaptive than conventional systems. The ability to reconfigure allows a robot to disassemble and reassemble to form new morphologies that are better suited for new tasks.
- *Robustness*: since robot parts are interchangeable, it can also replace faulty parts autonomously, thus leading to self-repair behaviours.
- *Low Cost*: reconfigurable robotic systems can potentially lower overall robot cost by making many copies of one type of modules so economies of scale and mass production come into play. Also, a range of complex solutions can be made from one set of modules, saving costs through reuse.

In manufacturing, these features result essential for users of robotic solutions producing very different families of products for several clients. Relying upon one or more reconfigurable robots could represent a major efficiency leverage as it would allow fast and cost-effectively reaction to short term changes coming from clients. The current practice associated to the need to accomplish future (forecasted) changes of production requirements is fulfilled by oversized robotic structures and, consequently, noticeably increasing the related investment costs.

However, these three advantages have not yet been fully implemented in reality and represent the main reasons for a limited diffusion of reconfigurability for industrial robots adopted in manufacturing. Also, the possibility to change the degrees of freedom make modular robots more versatile in their potential capabilities, but implies the risk to incur a performance trade-off and increased mechanical and control complexities. In fact, existing examples of reconfigurable robots are frequently constituted by a set of almost identical modules and any possible change of reaching and flexibility is managed by adding or subtracting these modules; this practice would be poorly viable in industrial robotics as it would boost the risk to affect accuracy of the manipulator and making more complex the dynamic control of the structure. In manufacturing, the possibility to have several families of robotic modules with very different shape, size and functionalities would be even instrumental dependently to the tasks to be accomplished and the modules' position in the kinematic chain. Examples of currently available research in this field is outlined in [31–33] with preliminary industrial prototypes found in [34–37].

Thus, the open questions the research and industry are facing for this category of reconfigurable robots are:

i) how to guarantee robotic solutions able to operate efficiently and effectively to manufacture high quality products with minimum economical and resource losses;
ii) what is the flexibility of the robot required to change configuration and behaviour starting from a finite set of modules which permit to accomplish demand or technological evolutions of a family of products.

3. The configuration approach

Starting from a set of production requirements, the configuration problem determines the robot architecture and its building block modules. Particularly, the objective of the robot configuration problem is to select the type and number of modules (i.e. links, joints and end-effectors) necessary to generate a robotic arm. This selection is executed to determine a configuration solution able to face present and future manufacturing tasks, i.e. considering a starting robot configuration and, if necessary, a set of possible reconfigurations.

The robot modules are considered as organised in a catalogue and they present variable geometric features, performance and costs. This catalogue includes joint modules, link modules and also the end-effectors; for every one of these families, a variety of models are considered as available so to enable a large number of configuration combinations. The joint and the link modules can have multiple ports; the minimal number of ports for a link or a joint module is two. Joint modules can be classified into active and passive modules. An active joint has motor(s) to produce the motion while a passive joint only provides Degree of Freedoms (DoFs) in a configuration with a parallel or closed-loop structure. Multiple joints can be nested to form a composite joint with enhanced performance; similarly, multiple links can be nested to form longer structures.

Industrial robots must present specific Key Performance Indicators (KPIs) such as accuracy, high speed and low costs. Recently, also the energy efficiency started to play a major role in the robot assembly and management. In configuration problems, this latter KPI is often addressed under the economic perspective when the operational costs are modelled.

The following table lists the KPIs considered at robot level.

The targeted KPIs driving the proposed robot configuration approach are considered concurrently to investment costs of the robot over the robot lifecycle.

The strong competitive leverage of such robots in manufacturing relies upon a new generation of joint and linkage design which present new cutting-edge performance. The current work will not go in detail in the description of such modules which will be presented with dedicated papers [38]. Starting from the catalogue of modules, the configuration algorithm generates a number of robot (re)configurations which are analysed kinematically, statically and dynamically so to ensure the best trade-off in accordance to the objective functions.

For sake of brevity, the current work only addresses the kinematic aspects.

The algorithm refers to a set of hypothesis which has been listed in the following:

- Hp1: serial robots.

• Hp2: the position of the basis of the robot is assigned (origin frame is known).
• Hp3: every module presents one active input port and one active output port.
• Hp4: the robot frames are allocated in accordance to the Denavit–Hartenberg convention.
• Hp5: the kinematic structure will be selected from a set of fundamental structures. The proposed approach is also open to the inclusion of new kinematics chains.

The robot configuration process is required to satisfy a number of constraints which ensure the correct execution of the manufacturing tasks to be performed by the robot with accuracy, stiffness and reliability over time. The modelling of the robot constraints starts from the analysis of the robot tasks to be executed so that every production task is associated to one or more end-effectors (and, consequently, to a set of robotic tasks). Production tasks can evolve over the time as a result of new product variants to be manufactured. Every production task is associated to an identification code and one or more end-effectors capable of performing it. From the end-effector it can be derived the payload to be managed by the robot. The type of task and the end effectors permit to determine the minimum number of degrees of freedom for the manipulator to execute it. Assuming the basis of the robot has a known and fixed position and the working area is assigned (manufacturing cell), all tasks contribute to determine the robot minimum workspace. This information drives the selection of the robot reachability space. Every task is associated to a tolerance range which puts boundaries to the robot accuracy characteristics. In fact, accuracy in exploiting the tasks are meant as the gap between the reference target value and the actual value of the working points. The accuracy will be modelled as an error measure for translation and rotation in accordance to [38].

Additional constraints refer to the need to match the production demand but selecting the robot modules which globally satisfy the throughput targets together with a set of module copies which can utilise the ones undergoing maintenance works.

Summarizing the constraints refer to:
• Satisfaction of the working space by selecting the reachability and dexterity space of the robot.
• Satisfaction of the minimum degrees of freedom DoF.
• Selection of modules able to realise the tasks by respecting the imposed process tolerances (static modelling).
• Satisfaction of production demand.

4. Mathematical formulation

4.1. The stochastic programming model

As anticipated in the previous section, the problem formulation will enable the definition of a robot configuration solution over time, thus proposing a basic configuration and a number of reconfigurations associated to the realisation of specific tasks (i.e. new associations of production tasks to robotic tasks). The reconfiguration feature is motivated by the need to match evolving production requirements over time. The forecasting of future requirements is often affected by uncertainty. As a result of that, the robot configuration approach is modelled as a stochastic problem. In the proposed model, a Two Stage Mixed Integer Linear Stochastic model can be formulated [39–41].

In general, the two stage model corresponds to have two phases in a decision-making process. At the beginning of the first phase, one has to make a decision without precise knowledge about the state of the world in the second stage. However, the uncertain future possibilities should be taken into account in the decision. Thus, as the reality unfolds it is possible to make a recourse decision at the second stage in order to cope with the reality being revealed so far. In mathematical terms, the problem is to find x under the constraints $_{Ax}=b$ and $x \geq 0$. After having made this decision, one of K possible scenarios might occur. Suppose that scenario k will occur with probability π_k ($\pi_k > 0$, $\sum_{k=1}^{K} \pi_k = 1$). In scenario k, the recourse problem with decision variables y_k is formulated as follows:

$$\min q_k^T y_k$$

s.t.

$$W_k y_k = h_k - B_k x$$

$$y_k \geq 0$$

For technical reasons, it is assumed matrices A and W_k have full row ranks. W_k and B_k are real-valued matrices. H_k is a know vector in $\mathbf{R}$. The optimal value of the above problem is a function of x. It is denoted by $Q_k(x)$. Thus, q, W, h, B represent particular realisations of the optimal values for the linear programme. Hence, taking into account every scenario, the expected costs under the decision x are:

$$c^T x + \sum_{k=1}^{K} p_k Q_k(x)$$

Putting the first and second stage decision variables all together, the optimisation problem can be formulated as

min

$$c^T x + \sum_{k=1}^{K} \pi_k q_k^T y_k(x)$$

s.t.

$$Ax = b$$

$$x \geq 0$$

$$W_k y_k = h_k - B_k x$$

$$y_k \geq 0, k = 1, \ldots, K$$

Table 1 lists the indexes considered in the mathematical formulation of the configuration approach.

Tables 2 and 3 outline respectively the list of parameters and variables associated to the configuration problem.

Table 1
List of KPIs.

ID	Description (measured at the robot flange)
Err	Positioning error (distance between desired and current position)
V	Displacement speed
A	Acceleration
J	Jerk
P	Absorbed electrical power (u=tension; i=current)
OP_time	Operation time for a single production task
MTTF_R	Mean Time To Failure Robot
ID	Description
Err	Positioning error (distance between desired and current position)
V	Displacement speed
A	Acceleration
J	Jerk
P	Absorbed electrical power (u=tension; i=current)
OP_time	Operation time for a single production task
MTTF_R	Mean Time To Failure Robot

Table 2
List of indexes.

Symbol	Definition
l	Link
j	Joint
f	Fundamental joint in the chain structure
g	Fundamental link in the chain structure
w	Production tasks
e	End-effector
c	Robotic chain
z	Scenario node
l	Link
j	Joint
f	Fundamental joint in the chain structure
g	Fundamental link in the chain structure
w	Production tasks
e	End-effector
c	Robotic chain
z	Scenario node

4.2. Mathematical formulation

The configuration model is formulated as a linear problem where the objective function (1) selects the robot solution globally minimising the investment costs over the considered time horizon. Investment costs (2) refer to the costs of joints, links and end-effectors. They are formulated as the cost of the module multiplied by the number of the module selected and the impact index that leverages the module selection by considering the module age, the associated carbon footprint and its reliability for the requirements accomplishment. (Eqs. 3-5). Table 4 lists all the model variables

$\forall\ chainc : [1. .C] \in \mathbb{Z},$

$$min \sum_z \frac{Inv_z \cdot pr_z}{(1+r)^{st_z-1}} \tag{1}$$

where:

$\forall\ z$

$$Inv_z = sum\left(InvJoint_z + InvLink_z + InvEE_z\right) \tag{2}$$

$$InvJoint_z = \sum_{c,j} p_j costj_j CopiesJ_{cjz} \tag{3}$$

$$InvLink_z = \sum_{l} p_l costl_l CopiesL_{clz} \tag{4}$$

$$InvEE_z = \sum_{e} p_e coste_e CopiesE_{ez} \tag{5}$$

In detail, every single module of the robot is associated to an impact index p computed as $(1-eff)$ where eff parameter is the module efficiency. This latter value embraces information about:

- module lifecycle is formulated as an ageing parameter, i.e. the difference between the current time and the beginning of life of the module [Tcurrent$-T$BoL].
- Embedded energy consumption is formulated as carbon footprint CO_2.
- Module's Mean Time To Failure MTTF represents the module reliability.

These three components of the impact index are leveraged by coefficients β, γ, η whose value is defined positive $[0,1] \in R+$. These coefficients are set by the user in accordance to the importance of the specific aspect compared to the others.

(Eqs. (6)–8) present the impact formulation.

$$p_j = \left(1 - eff_j\right) = \beta_j A_j + \gamma_j CO2_j + \eta_j MTTF_j \tag{6}$$

$$p_l = \left(1 - eff_l\right) = \beta_l A_l + \gamma_l CO2_l + \eta_l MTTF_l \tag{7}$$

$$p_e = \left(1 - eff_e\right) = \beta_e A_e + \gamma_e CO2_e + \eta_e MTTF_e \tag{8}$$

The configuration problem undergoes a number of constraints. The first group of constraints is related to the assignment problem. Constraint (9) sets the minimum number of degrees of freedom at least equal to the maximum number required to perform the production tasks.

$$\forall\ c, w, z\ \ DoFT_{w,z} \leq DoFR_z \tag{9}$$

Constraint (10) selects the robotic chain whose number of fundamental joints is equal to the requested DoF.

$$\forall\ z\ \ DoFR_z = \sum_{c,f} SCF_{c,f} \cdot RCF_{c,f,z} \tag{10}$$

Constraint (11) imposes the number of fundamental links selected are equal to the (DoFR-1).

$$\forall\ z\ \ DoFR_z - 1 = \sum_{c,g} SCG_{c,g} \cdot RCG_{c,g,z} \tag{11}$$

(Eqs. (12) and 13) state that the fundamental joints and fundamental links are selected from the same chain.

$$\forall\ z, c, f, ff: \mathrm{ff} = \mathrm{f} + 1 \quad RCF_{c,f,z} - RCF_{c,ff,z} = 0 \tag{12}$$

$$\forall\ z, c, g, gg: \mathrm{gg} = \mathrm{g} + 1 \quad RCG_{c,g,z} - RCG_{c,gg,z} = 0 \tag{13}$$

Constraint (14) enables the selection of fundamental joints from only one chain among the ones which present a suitable number of joint.

$$\forall\ f, z \quad \sum_c SCF_{c,f} \cdot RCF_{c,f,z} \leq 1 \tag{14}$$

Constraint (15) enables the selection of fundamental links from only one chain among the ones which present a suitable number of links.

$$\forall\ g, z \quad \sum_c SCG_{c,g} \cdot RCG_{c,g,z} \leq 1 \tag{15}$$

Constraint (16) states that every fundamental joint can be assigned to only one actual joint. The case of composite joints will be not presented in the current model.

$$\forall\ f, z \quad \sum_c RCF_{c,f,z} = 1 \tag{16}$$

Constraint (17) states that every fundamental link can be assigned to only one actual link. The case of composite links will be not presented in the current model.

$$\forall\ g, z \quad \sum_c RCG_{c,g,z} = 1 \tag{17}$$

Constraints (18) and (19) link variables RCF and RCG to RCJF and RCLG responsible for distributing the chain across actual catalogue modules.

$$\forall\ c, f, z \quad \sum_j SCJF_{c,j,f} \cdot RCJF_{c,j,f,z} = RCF_{c,f,z} \tag{18}$$

$$\forall\ c, g, z \quad \sum_l SCLG_{c,l,g,z} \cdot RCLG_{c,l,g,z} = RCG_{c,g,z} \tag{19}$$

Eq. (20) computes the number of joints to be selected for the considered chain.

Table 3
List of parameters.

Symbol	Definition
$Length_{c,j,f}$	Element of the matrix listing – for the chain c – the length of the joint j while operating as fundamental joint f
$Length_{c,l,g}$	Element of the binary matrix listing – for the chain c – the length of the link l while operating as fundamental link g
dl_l	Index ϵ N^+ indicating the duty of the link l.
d_j	Index ϵ N^+ indicating the duty of the joint j.
dl_{lg}	Index ϵ N^+ leveraging the suitability of the position of the link l in the chain structure. The higher is duty the closest to the chain basis the link should be.
d_{jf}	Index ϵ N^+ indicating the duty of the joint j in the chain structure. The higher is duty the closest to the chain basis the joint should be.
$cost_j$	Cost of joint j
$cost_j$	Cost of link l
$cost_j$	Cost of end-effector e
$SCF_{c,f}$	Element of the binary matrix listing the fundamental joint f composing the chain c
$SCJF_{c,j,f}$	Element of the binary matrix listing – for the chain c – the type of joints j which can be suitable for every fundamental joint f
$SCLG_{c,g}$	Element of the binary matrix listing the fundamental link g composing the chain c
$SCLG_{c,l,g}$	Element of the binary matrix listing – for the chain c – the type of links l which can be suitable for every fundamental link g
$SCDoF_c$	c-th element of array SCDoF listing the DoF for the fundamental robot chains considered for the configuration model
$DoFT_{w,z}$	w-th element of array DoFT listing the Degrees of Freedom associated to the realisation of a Production Task types to be executed during scenario node z
$Reach_{w,z}$	Minimum reaching space required to perform production task w (distance between the working area and the robot basis) during scenario node z
$PTET_{w,e,z}$	Incidence Binary Matrix associating Production Tasks to End Effectors capable to executing it. The matrix element $PTET_{w,e}$ values 1 if the task w can be executed by EE e during scenario node z, 0 otherwise
$SCEE_{c,e}$	Binary matrix associating to every chain the suitable end-effector of type e.
$SJEE_{j,e}$	Binary matrix associating to every joint j the suitable end-effector of type e.
$MTTF_L$	Link mean time to failure in sec
$A_L = [t\text{current} - t\text{BoL}]_L$	The link lifecycle formulated as an ageing parameter
$CO2_L$	The link embedded energy consumption formulated as carbon footprint CO_2
$MTTF_j$	Joint mean time to failure in sec
$A_j = [t\text{current} - t\text{BoL}]_j$	The joint lifecycle formulated as an ageing parameter
$CO2_j$	The joint embedded energy consumption formulated as carbon footprint CO_2
$MTTF_e$	End-effector mean time to failure in sec
$A_e = [t\text{current} - \text{tBoL}]_e$	The end-effector formulated as an ageing parameter
$CO2_e$	The end-effector embedded energy consumption formulated as carbon footprint CO_2
pr_z	Realisation probability associated to scenario node z
st_z	Stage of scenario node z
r	Discount rate
$ME_{e,z}$	Working capacity for the end-effector of type e (end-effector daily availability)
$MJ_{j,z}$	Working capacity for the joint j (daily availability)
$ML_{l,z}$	Working capacity for the link l (daily availability)
$Time_{w,e}$	Processing time associated to the end-effector e to perform task w
$MeanTimeT_w$	Average robot processing time associated to production task w
V_e	*Production demand of task w*
m_{Ji}	Mass of Joint i
m_{Li}	Mass of Link i
m_{EF}	Mass of end effector
R_{Ji}	Radius of Joint i
R_{Li}	Radius of Link i
L_{Ji}	Length of Joint i
L_{Li}	Length of Link i
$\dot{\omega}_{maxj}$	Maximum joint angular acceleration
$ERes_j$	Encoder resolution measured at the exit flange of the joint j after the gear box reduction
Acc_w	Minimum accuracy necessary to match production task w
$T_{re\,s_{nomj}}$	Nominal total resistant torque for joint

$$\forall\, c, z \quad \sum_{f} RCF_{c,f,z} = \sum_{j} NJ_{c,j,z} \tag{20}$$

Eq. (21) computes the number of links to be selected for the considered chain.

$$\forall\, c, z \quad \sum_{g} RCG_{c,g,z} = \sum_{l} NL_{c,l,z} \tag{21}$$

(Eqs. (22) and 23) distribute the number of joints across the chain.

$$\forall\, c, z \quad \sum_{j,f} RCJF_{c,j,f,z} = NJC_{c,z} \tag{22}$$

$$\forall\, c, z \quad \sum_{l,g} RCLG_{c,l,g,z} = NLC_{c,z} \tag{23}$$

(Eqs. (24) and 25) compute the total number of joints and links.

$$\forall\, c, z \quad \sum_{j} NJ_{c,j,z} = NJC_{c,z} \tag{24}$$

$$\forall\, c, z \quad \sum_{l} NL_{c,l,z} = NLC_{c,z} \tag{25}$$

Eq. (26) associates to every fundamental joint only one actual joint. Constraint (27) operates similarly for the links.

$$\forall\, f, z \quad \sum_{c,j} SCJF_{c,j,f} \cdot RCJF_{c,j,f,z} \leq 1 \tag{26}$$

$$\forall\, g, z \quad \sum_{c,l} SCLG_{c,l,g} \cdot RCLG_{c,l,g,z} \leq 1 \tag{27}$$

Table 4
List of variables.

Symbol	Definition
Inv_z	Investment costs associated to scenario node z
$InvJoint_z$	Joint investment costs associated to scenario node z
$InvLink_z$	Link investment costs associated to scenario node z
$InvEE_z$	End effector investment costs associated to scenario node z
p_j	Impact index for joint j
eff_j	Efficiency of joint module j
P_l	Impact index for link l
eff_l	Efficiency of link module l
p_e	Impact index for end-effector e
eff_e	Efficiency of end-effector e
$NJ_{c,j,z}$	Number of joint j selected for every chain during scenario node z
$CopiesJ_{c,j,z}$	Number of copies of joint j
$NL_{c,l,z}$	Number of link l selected for every chain during scenario node z
$CopiesL_{c,l,z}$	Number of copies of link l
$NJC_{j,z}$	Total number of joint j selected during scenario node z
$NLC_{l,z}$	Total number of link l selected during scenario node z
$NE_{e,z}$	Total number of end-effector e selected during scenario node z
$CopiesE_{c,e,z}$	Number of copies of end-effector e
$DoFR_z$	Required DoF for the robotic chain during scenario node z
RCF_{cfz}	Binary matrix determining – for the scenario z – the size of the chain in terms of required fundamental joints
RCG_{cgz}	Binary matrix determining – for the scenario z – the size of the chain in terms of required fundamental links
$RCJF_{cjfz}$	Binary matrix associating – for every chain – the fundamental joint f of the chain to the actual type of joint j selected from the module database during scenario node z.
$RCLG_{c,l,g,z}$	Binary matrix associating – for every chain – the fundamental link g of the chain to the actual type of link l selected from the module database during scenario node z.
$ReachCJ_{cz}$	Robot arm length due to joint contribution for every scenario z and chain c
$ReachCL_{cz}$	Robot arm length due to link contribution for every scenario z and chain c
$ReachC_{cz}$	Total robot arm length for every scenario z and chain c
$ReachWR_z$	Robotic arm with the maximum length among the all chains
WR_{ijz}	Support variable indicating the number of joints requested by the robot independently from the chain
WR_{lgz}	Support variable indicating the number of links requested by the robot independently from the chain
$RTET_{w,e,z}$	Incidence Binary Matrix associating Production Tasks to End Effectors actually selected to executing it. The matrix element $RTET_{w,e}$ values 1 if the task will be executed by EE e during scenario node z, 0 otherwise
$RJEE_{j,e,z}$	Incidence Binary Matrix associating Joints to End Effectors actually selected. The matrix element $RJEE_{j,e}$ values 1 if the joint j can load the EE e during scenario node z, 0 otherwise
$WR_{j,f,z}$	Incidence Binary Matrix associating Joints to Fundamental Joints.
$WR_{l,g,z}$	Incidence Binary Matrix associating Links to Fundamental Links.
$RFlange_{j,z}$	Selection of joint operating as the flange of the robot, i.e. as the last fundamental joint of the chain
$RFlangeC_{c,j,z}$	Selection of joint operating as the flange of the robot, i.e. as the last fundamental joint of the chain, for every chain
$TotTime_z$	Total time required to process the entire volume of production tasks w
$Tres_{maxj,f,z}$	Maximum total resistant torque for joint j in position f for scenario z
$Tres_{statj,f,z}$	Maximum static resistant torque for joint j in position f for scenario z
$Tres_{dynj,f,z}$	Maximum dynamic resistant torque for joint j in position f for scenario z
$I_{maxj,f,z}$	Maximum inertia momentum of joint j operating in position f for scenario z
Δxtop	Maximum error measured at the robotic chain TCP on the plane (x,y)
Δytop	Maximum error measured at the robotic chain TCP on the plane (x,y)
Δxfront	Maximum error measured at the robotic chain TCP on the plane (x,z)
Δzfront	Maximum error measured at the robotic chain TCP on the plane (x,z)

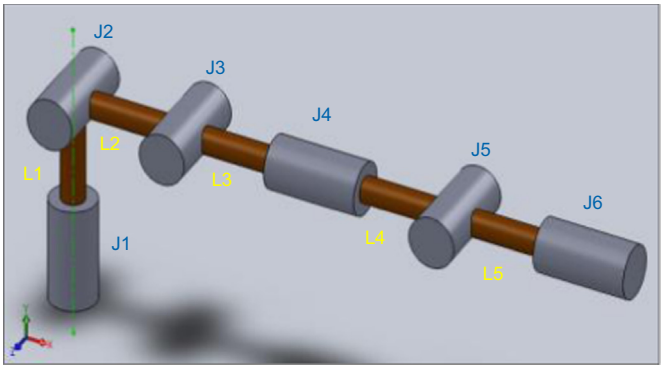

Fig. 1. Robot schematic for the Inertia moment computation.

Eq. (28) computes the reaching space associated to the joints. The reaching is given by length of the joints and the links. Based on the particular position of the joint (i.e. axial or longitudinal),

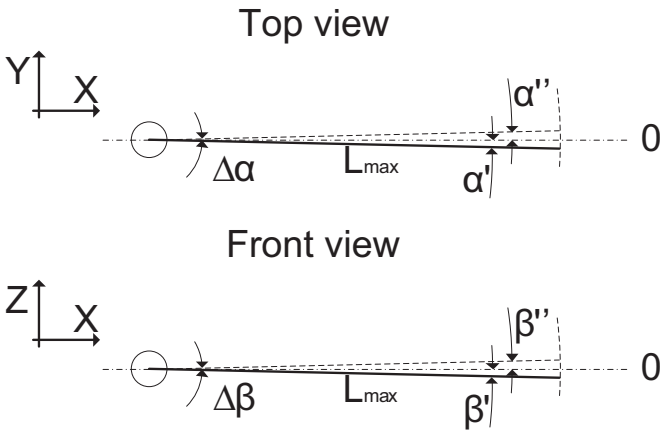

Fig. 2. Minimum displacement of $J1$ (top view) and $J2$ (front view).

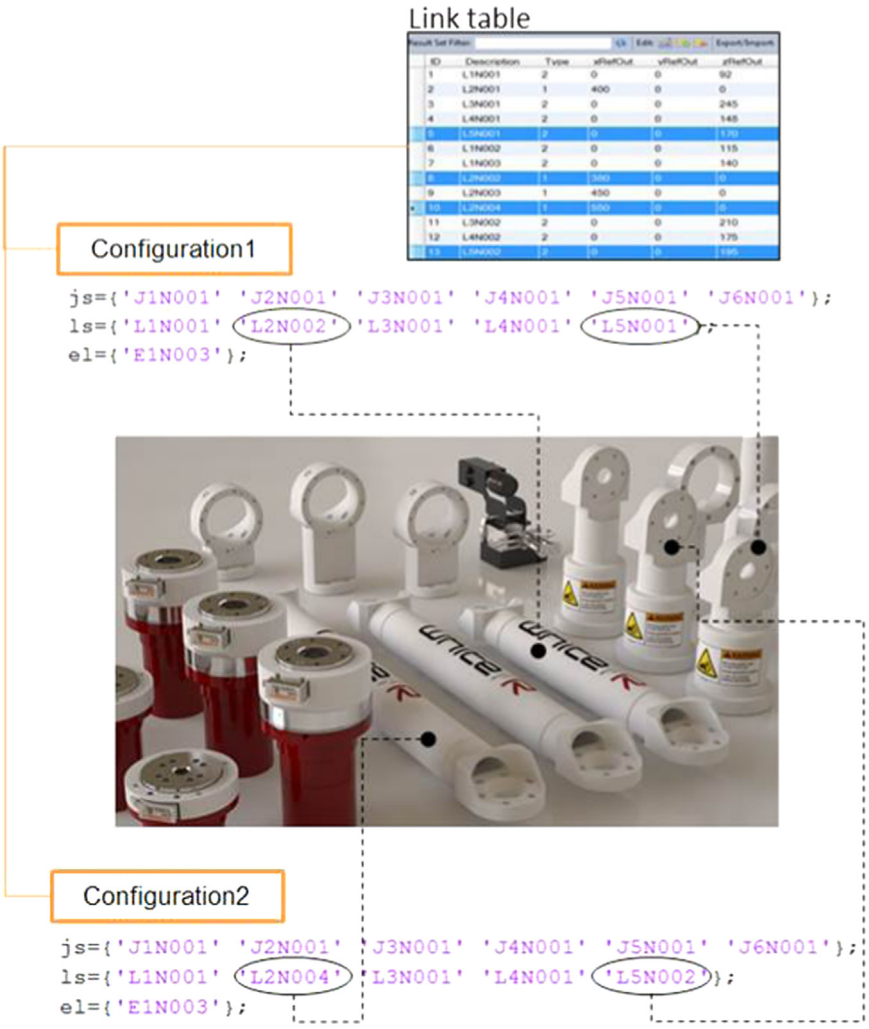

Fig. 3. Set of modules available to configure the robotic chain.

Fig. 4. Robotic joint module.

Table 5
Joint typologies.

Model	Length [mm]	Diameter [mm]	Cost [FEURO]
Type I	100	100	210
Type II	150	120	257
Type III	180	150	271
Type VI	200	180	314

Table 6
Link typologies.

Model	Length [mm]	Diameter [mm]	Cost [FEURO]
Type I	100	100	85.7
Type II	200	110	92.8
Type III	300	110	97
Type VI	400	110	121.4

Table 7
End-effector typologies.

Model	Length [mm]	Envolope Diameter [mm]	Weight [kg]	Cost [FEURO]
Type I	180	80	0.8	142.8
Type II	200	85	1.1	214
Type III	180	80	0.75	150

the value of the joint length changes. For instance, in an anthropomorphic chain, joint 3 length scarcely contributes to the computation of the reaching and only its diameter (if assumed cylindrical) can be considered.

Table 8
Robot solutions.

Robot solution	DoF	Joint type	Link type	End-effector type
First stage	6	2 Type III (position 1,2), 3 Type II (position 3,4,5), 1 Type I (position 6)	4 Type I, 1 Type II	Type 2
Second stage	6	3 Type III (position 1,2,3), 2 Type II (position 4,5), 1 Type I (position 6)	3 Type I, 1 Type II, 1 Type III	Type 2

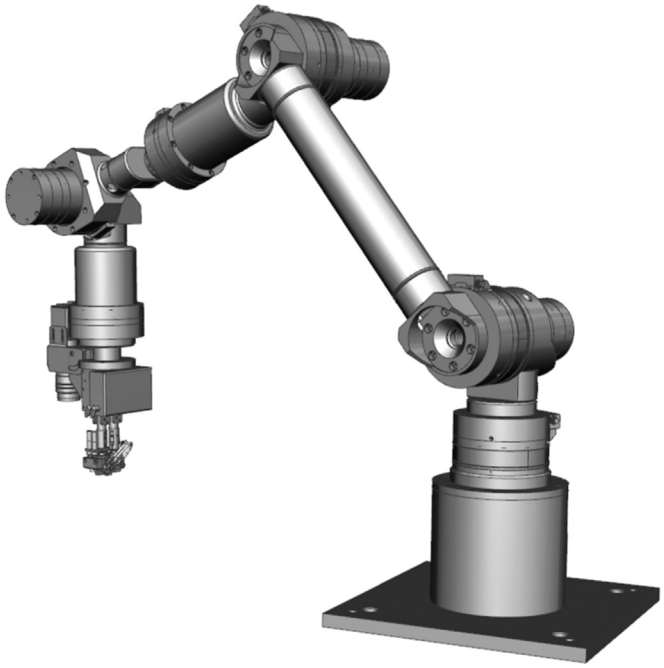

Fig. 5. Robotic chain configured at the first stage.

$\forall\ c, z$

$$\sum_{j,f}\left(Lenght_{c,j,f}\cdot RCJF_{c,j,f,z}\right)=ReachCJ_{c,z} \tag{28}$$

Eq. (29) computes the reaching space associated to the links.

$\forall\ c, z$

$$\sum_{l,g}\left(LenghtL_{c,l,g}\cdot RCLG_{c,l,g,z}\right)=ReachCL_{c,z} \tag{29}$$

Eq. (30) computes the total reaching space resulting from the sum of joints and links for every chain.

$\forall\ c, z$

$$\left(ReachCJ_{c,z}\right)+\left(ReachCL_{c,z}\right)=\ ReachC_{c,z} \tag{30}$$

Eq. (31) selects the maximum reaching space value.

$$\forall\ z\quad max\left(ReachC_{c,z}\right)=ReachWR_{z} \tag{31}$$

Eq. (32) imposes the maximum reaching space determined in the previous constraint is higher than the one required for every production task. This value of reaching space for the production tasks is derived from the layout configuration model.

$$\forall\ w, z\quad \sum_{w} Reach_{w,z} \le ReachWR_{z} \tag{32}$$

(Eqs. (33) and 34) state the reaching is always positive for joints and links.

$$\forall\ c, z\quad ReachCJ_{c,z}\ge 0 \tag{33}$$

$$\forall\ c, z\quad ReachCL_{c,z}\ge 0 \tag{34}$$

Constraints (35) and (36) determine synthesise the types of joints and links selected to compose the final chain. *WRs* are support variables independent from the chain type.

$$\forall\ j, f, z\quad \sum_{c} RCJF_{c,j,f,z}=WR_{j,f,z} \tag{35}$$

$$\forall\ l, g, z\quad \sum_{c} RCLG_{c,l,g,z}=WR_{l,g,z} \tag{36}$$

Constraint (37) imposes the joints are ordered in accordance to their duty across the chain (across all fundamental joints).

$\forall\ j, z$

$\forall\ f, ff : ff=f+1$

$$WR_{j,f,z}\cdot djf_{j,f}-WR_{j,ff,z}\cdot djf_{j,ff}\ge 0 \tag{37}$$

Constraint (38) imposes the links are ordered in accordance to their duty across the chain (across all fundamental links).

$\forall\ l, z$

$\forall\ g, gg : gg=g+1$

$$WRL_{l,g,z}\cdot dlg_{l,g}-WRL_{l,gg,z}\cdot dlg_{l,gg}\ge 0 \tag{38}$$

Constraint (39) selects the joint operating as the robot flange in the chain.

$$\forall\ z, j, f : f=DoF\quad WR_{j,f,z}=RFlange_{j,z} \tag{39}$$

Eq. (40) considers only the joint positioned on the robot flange (last Fundamental Joint) for every chain and joint.

$$\forall\ c, z, j, f : f=DoF\quad RCJF_{c,j,f,z}=RFlangeC_{c,j,z} \tag{40}$$

Constraint (41) assigns production tasks only to end-effectors capable to execute them.

$$\forall\ w, z\quad \sum_{e} PTET_{w,e,z}\cdot RTET_{w,e,z}=1 \tag{41}$$

Constraint (42) stays only one type of end-effector can be selected for every task.

$$\forall\ w, z\quad \sum_{e} RTET_{w,e,z}=1 \tag{42}$$

Eq. (43) computes the total time required to process all the production tasks *w*.

$$\forall\ z\quad \sum_{w,e} RTET_{w,e,z}\cdot Time_{w,e}\cdot V_{w}=TotTime_{z} \tag{43}$$

Constraint (44) imposes the joint positioned on the flange must be compliant with the selected end-effector.

$$\forall\ j, z\quad \sum_{e} SJEE_{j,e}\cdot RJEE_{j,e,z}-RFlange_{j,z}=0 \tag{44}$$

Constraint (45) imposes the end-effector selected for the production task is the same of the one selected to be nested to the flange joint.

$$\forall\ e, w, z\quad RTET_{w,e,z}=\sum_{j}\cdot RJEE_{j,e,z} \tag{45}$$

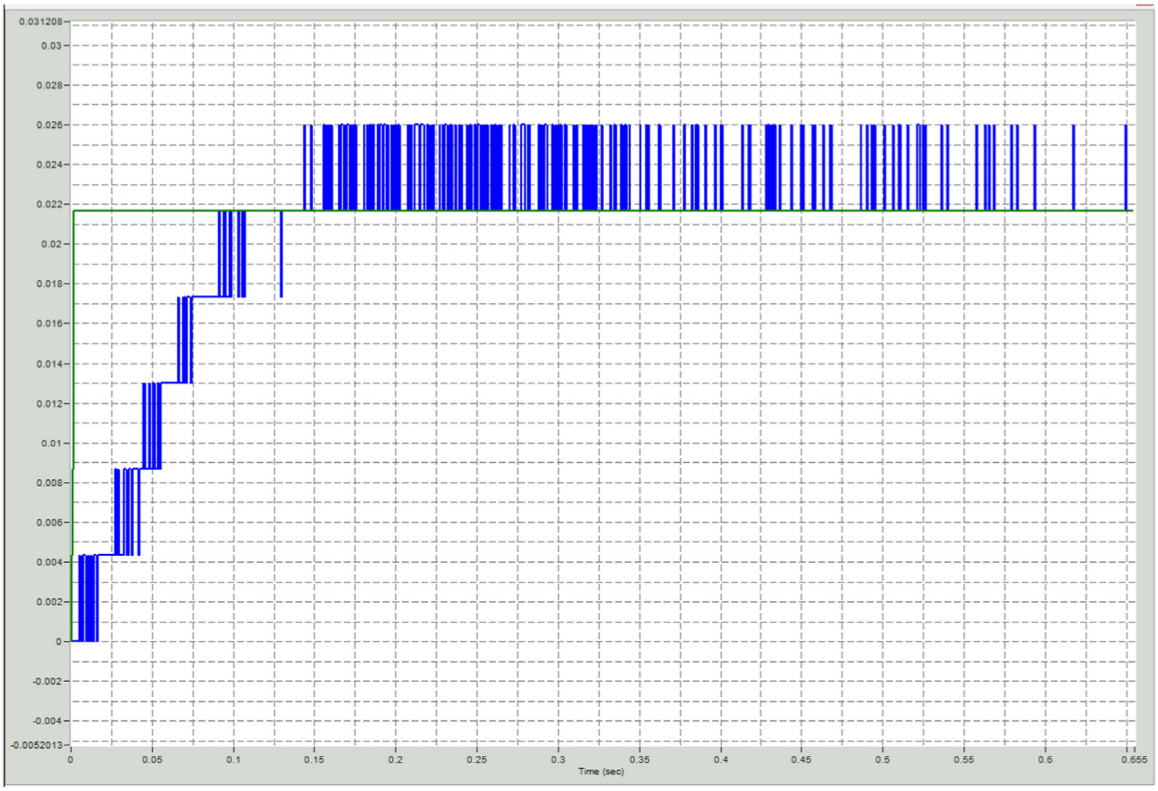

Fig. 6. Robotic chain positioning error of Joint 2.

Table 9
Commercial robot comparison.

Model	Reaching [mm]	DoF	Payload [kg]	Repeatability in pose [mm]	Cost [FEURO]
Kuka KR16	1611	6	16	0.05	7857
ABB IRB 1600.6	1200	6	6	0.02	5000
DENSO VM-6083G	1298	6	13	0.05	8000
Universal robots	1300	6	10	0.1	4286

Eq. (46) computes the number of end-effectors to be acquired for every type.

$$\forall\, e, z \quad \sum_{w} RTET_{w,e,z} * TimeTE_{w,e} * Vt_{w,e} \leq ME_{e,z} \cdot CopiesE_{c,e,z} \tag{46}$$

Eq. (47) computes the copies of joint j to be acquired for every type in order to accomplish the production demand.

$$\forall\, c, j, z \quad \sum_{f,w} RCJF_{c,j,f,z} MeanTimeT_w * Vt_{w,e} \leq MJ_{j,z} \cdot CopiesJ_{c,j,z} \tag{47}$$

Eq. (48) computes the copies of link l to be acquired for every type in order to accomplish the production demand.

$$\forall\, c, l, z \quad \sum_{g,w} RCLG_{c,l.g,z} MeanTimeT_w * Vt_{w,e} \leq ML_{l,z} \cdot CopiesL_{c,l,z} \tag{48}$$

The robotic arm preliminarily configured is far from being the final solution as it still needs to accomplish the maximum resistant torque requirement and the accuracy requirements.

In detail, based on the modules selected to be allocated across the robotic chain, the first step is to evaluate if all the rotating parts (i.e. the joints) possess the correct resistant torque for enabling all the movements of the robot. From the end-effector the payload to be managed by the robot can be derived. Every module is fully described in terms of functionalities, geometry and material. Joints' motors are also labelled with their torque nominal values ($T_{re\ s_{nomj}}$). Such torque analysis requires the robot manipulator to be put in the most stressed kinematic pose and only then the actual resisting torque for all the joints of the chain is computed and compared to the nominal one. In the case the actual torque would be higher than the nominal value, the specific robot configuration would be discarded and the immediate next good solution would be examined.

The maximum resistant torque results from the sum of the torque static and dynamic components (Eq. 49). The static component (Eq. 50) of the resisting torque is given by a static load distribution balance (mostly related to the robotic elements' weights) where all joints and links are evaluated with regards to a reference pivot point. The contribution of every element is its weight measured in the centre of mass multiplied by the distance between joint and the considered element. The dynamic contribution is computed in Eq. (51) and involves the maximum required angular acceleration of joint j ($\dot{\omega}_{maxj}$) times the maximum inertia momentum for joint j positioned as a fundamental joint f ($I_{maxj,f,z}$). The inertia moments of the generic joint j results from the sum of the inertia associated to each rotating element transported to the j motor axis (with Huygens–Steiner Theorem or Parallel Axes Theorem). Eq. (52) illustrates the computation of the inertia moment for a 6 axes anthropomorphic chain computed in the robot basis (joint 1) in the most critical pose of the robot is put (Fig. 1).

$$\forall\, c, j, f, z T_{re\ s_{maxj,f,z}} = T_{re\ s_{statj,f,z}} + T_{re\ s_{dynj,f,z}} \tag{49}$$

$$T_{re\ s_{statj,f,z}} = Weight \cdot distancefrompivot \tag{50}$$

$$\forall\, j, f, z T_{re\ s_{dynj,f,z}} = \dot{\omega}_{maxj} \cdot I_{maxj,f,z} \tag{51}$$

$$I_{J1}=\frac{1}{2}m_{J1}R_{J1}^2+\frac{1}{2}m_{L1}R_{L1}^2+\frac{1}{12}m_{J2}\left(3R_{J2}^2+L_{J2}^2\right)+\frac{1}{12}m_{L2}\left(3R_{L2}^2+L_{L2}^2\right)$$
$$+m_{L2}\left(R_{J2}+\frac{1}{2}L_{L2}\right)^2+\frac{1}{12}m_{J3}\left(3R_{J3}^2+L_{J3}^2\right)$$
$$+m_{J3}\left(R_{J2}+L_{L2}+R_{J3}\right)^2+\frac{1}{12}m_{L3}\left(3R_{L3}^2+L_{L3}^2\right)$$
$$+m_{L3}\left(R_{J2}+L_{L2}+2R_{J3}+\frac{1}{2}L_{L3}\right)^2+\frac{1}{12}m_{J4}\left(3R_{J4}^2+L_{J4}^2\right)$$
$$+m_{J4}\left(R_{J2}+L_{L2}+2R_{J3}+L_{L3}+\frac{1}{2}L_{J4}\right)^2+m_{EF}$$
$$(R_{J2}+L_{L2}+2R_{J3}+L_{L3}+L_{J4}+L_{L4}+2R_{J5}+L_{L5}+L_{J6})^2 \quad (52)$$

Once checked the maximum resistant torque of the robotic chain, the following step is to check the robot accuracy. At this stage, it is purely done as a static analysis assuming the robot is a perfect rigid body; that is by keeping the robot still in the most critical pose and evaluating how the minimum appreciable joint rotation (corresponding to the encoder resolution) becomes an error at the TCP placed on the flange.

As an example, in the following it is possible to evaluate the misplacement respectively in relation to $J1$ and $J2$ (pose represented in Fig. 1). It is assumed $L_{\max}$ being the maximal length of the robot and Fig. 2 illustrating the schematic of the robot.

In the case of $J1$, it is possible to appreciate a displacement in the (x,y) Cartesian plane:

$$\Delta xtop=|L_{max}[\cos(\Delta\alpha)]| \quad (52)$$

$$\Delta ytop=|L_{max}[\sin(\Delta\alpha)]| \quad (53)$$

In the case of $J2$, it is possible to appreciate a displacement in the (x,z) Cartesian plane:

$$\Delta xfront=|L_{max}[\cos(\Delta\beta)]| \quad (54)$$

$$\Delta zfront=|L_{max}[\sin(\Delta\beta)]| \quad (55)$$

$\Delta\alpha$ and $\Delta\beta$ will be read by respectively $J1$ and $J2$ encoders (ERes$j1$ and ERes$j2$) whose measurement is realized after applying the reduction rate of the gearbox. Therefore, the accuracy check demands to compare the TCP error with the necessary accuracy associated to the most precise production task of the list.

$$J1: \Delta xtop \le Acc_w \quad (56)$$

$$J1: \Delta ytop \le Acc_w \quad (57)$$

$$J2: \Delta xfront \le Acc_w \quad (58)$$

$$J2: \Delta zfront \le Acc_w \quad (59)$$

In the case the accuracy constrain is violated, a different joint type will be selected to operate in that fundamental position, likely one with an increased reduction rate of the gear box.

The model also includes constraints ensuring the temporal sequence between first and second stage decisions. These constraints are not reported for sake of brevity.

The mathematical model has been implemented in IBM ILOG CPLEX and consists of 256 variables, 185 of which are binary, 43 are integer and 28 are float. The model implementation and the related experimental tests have been run on a Dell Precision M4800 Workstation. The average execution time is 7 seconds utilising 17% of local memory to solve 171 nodes.

5. Industrial case study

The proposed configuration approach has been applied to an industrial use case related to the field of solar cell manufacturing. Particularly, the analysed manufacturing processes are the tabbing and stringing of solar cells for both specialist low power and Concentrated Photovoltaic (CPV) systems [42]. While tabbing and stringing systems are commonplace in standard flat-plate module manufacture, the large variety in size and shape of CPV cells means specialist systems or manual processes are required.

For concentrator systems using custom designed Laser Grooved Buried Contact LGBC cells, the need to manually produce strings adds to an already costly, time-consuming process.

The use case, considered interesting in the current analysis, is to evaluate the scenario in which the cell manufacturers would decide to switch from a manual based production to a robotic based assembly. The very aspect to be considered during the robot selection process would be the investment costs of the resource while addressing the necessary precision for the assembly process as well as the great product variability over the time.

So far, such evaluation always led to keep the production as manual based. However, the production requirements envisage the reconfigurability feature as an essential leverage for the involved equipment. In fact, the automation of the tabbing and stringing represents an opportunity to significantly increase production capabilities and decrease CPV module costs, resulting in a reduction in the overall system cost.

The family of products considered as a proof of concept for the configuration model is composed by three part types to be manufactured at the present time and one new variant forecasted to be possibly committed in four months time. All the part types are characterized by 0.10 mm precision level to be ensured in manufacturing (both affecting the cell and string positioning). This value will represent the benchmark to test the robot accuracy. The targeted production volumes are on average 100 cellarrays per day for a very heterogeneous product mix (and reaches 200 when the mix of part is quite reduced) and it can slightly change based on the cell type. Every cell array can host a number of cell ranging between 5 and 12.

The aforementioned production requirements set the objectives for the robot configuration process. Along with the capability to perform the manufacturing process (driving precision, accuracy and reliability performance) on evolving product types, a major driver for the reconfigurable robot is the ability to change configuration and set-up extremely fast. All these aspects influence the design of the robotic modules, i.e. joint and links.

The configuration model relies upon a module catalogue which comprehends alternative models of joints, links and end-effectors (Fig. 3). The various models differ in dimensions, performance, capabilities and costs. Everywhere in the current section, costs are considered scaled by a constant factor f.

The joints referred in this work are compact and embed the actuation controllers along with the electricity, pneumatics and communication cables (Fig. 4). The joint models vary in the maximum torque and size; such a variety enables for example the possibility to select heavier structures close to the basis of the robotic arm and lighter model in proximity of the robot nose.

As an example, Table 5 lists four types of joints considered in the catalogue. Similarly to the joints, also the links are available in various typologies which differ by shape and size. The considered link types are listed in Table 6. Tables 5–7 report costs scaled by the factor f. Based on the process specifications, three types of end-effectors are considered as available for the configuration. Among the three, only Type III is capable of realizing all the operations; the other two can realise only a subset of them. Joints, links and end-effectors available in the catalogue are also associated to economic data together with impact and duty information.

At this stage of the configuration process, all these robotic modules do not exist in reality; they are virtually available in a catalogue. The robot builder will exploit the configurator software

to generate the best solution to suite the customer needs. The user of the robot configuration approach will only acquire (or build) the modules which will be selected as an outcome of the configuration model.

A set of robotic chains are available to the module, from 4 DoF planar robots to anthropomorphic robots.

The presented results refer to a small production problem characterized by four major families of production tasks (picking components, wire handling and soldering, placing components, vision check) associated to the three part types; tasks must be realized multiple times in order to ensure the targeted production. The four families of production tasks require a different number of minimum DoF, respectively 6 DoF for the wire handling and soldering, picking and placing and 4 DoF for the vision check tasks. The reference production problem is expected in the near future to undergo a change with regards to one part type of the reference product family; a new variant of the product will be possibly committed with a quite larger size of the cells. This will imply an increased reaching for the robotic arm from 500 mm to 1150 mm.

Considering current and future production requirements, the proposed configuration problem generates two sets of solutions for the robotic arm. In both cases, the manipulator presents 6 DoF. The first stage configuration of the robotic arm is constituted by medium size joints (type III) located in the position 1 and 2 of the fundamental chain and joints of type II for the position 3–4–5; the smallest joint (type I) has been selected for the flange position (position 6). These joints ensure to the robot the correct acceleration and speed to execute the tasks quickly and with precision. This configuration is later modified by reconfiguring one link of the robot and adopting another larger size joints (type III) with higher torque in position 3 of the fundamental chain. The choice is driven by the need to support an increased length of the arm which causes higher forces applied to the first three joints. The selected end-effector in every case and scenario is the Type II; it reasonably represents the best trade-off between cost and productivity as with one single investment it is possible to execute all the production tasks.

The following table summarizes the resulting configurations. (Table 8)

As a result of the production requirements and the module availability during the day, including maintenance, the configuration solution also suggests the acquisition of one extra copy of Joint III. This would globally ensure the robot capability to perform the amount of production tasks in the considered observing horizon. Fig. 5 shows the resulting robotic arm configured for the second stage, i.e. when it requires higher reach.

All the robotic joints are equipped with encoders whose resolution is 3.74E-7 rad. Based on the accuracy constraints, this makes the robot accuracy close to 5 μm. Of course, this accuracy value must be leveraged by additional aspects which have not been considered at this stage (such as material deformation) but will be addressed in future works. Nevertheless, a preliminary study of the robot behaviour executed with an on-board camera, shows extremely cutting edge performance, accuracy and repeatability of the robot also in motion.

The robot accuracy is assessed by performing an experimental plan based on data gathered by the joint encoders and an on-board camera closing the loop with the actual working environment and verifying the correctness of the encoders' measurements. The robot during the experimentation is asked to execute a specific set of production tasks repeatedly over 8 h. Fig. 6 highlights an example of measurement of the rotation of Joint 2 (the most stressed of the chain) before the gearbox reduction (101 rate of reduction). The plot represents the joint position when it is commanded to the motor a rotation of 0.0217 degrees (0.000217 degrees on the outer flange of the joint) corresponding to 5 μm on the robot's flange TCP. It can be noticed the robot is able to successfully reach and keep the commanded position. In fact, once reached the targeted rotation, it presents only the 0.004 μm of positioning error. The ramp to target the steady state rotation is performed by executing 5 intermediate compensation steps. This can be interpreted as the ability to target even smaller rotations, thus higher accuracy thresholds at the TCP.

If the resulting robot solution would be compared to existing commercial robots, it would be recognised as being very competitive in costs and performance. Also, the robot presents accuracy characteristics which are far above the performance of top class commercial robots used for high precision manufacturing industry.

In the case the robot user would be asked to purchase a standard rigid robot solution for the considered use case, he would reasonably select the robot model which has a range of reach that could accomplish the current and future requirements. Thus, focusing on solutions which have a reach higher than 1150 mm.

Table 9 lists few examples of commercial robots ensuring such requirement. It reports costs scaled by a factor f which must be evaluated with regard to the investment cost of reconfigurable robot, that is 2172.6 fEuros.

The examples of commercial robot solutions match the reach requirement but they are over dimensioned in relation to the actual necessary payload and robot performance. Of course, the price of the robot always embeds additional factors – such as the personnel and equipment costs of the company – which cannot be considered when comparing a commercial solution price to a lab prototype.

However, such comparison gives a first proof of concept that the possibility to exploit reconfigurability in manufacturing presents an economic leverage and allows the user to tune the robot features based on the requirements. In fact, the user can decide to face the investment costs in two separate decision stages, thus starting with the purchase of the robot modules necessary to match the current production requirements and buying the modules required to reconfigure the robot only when the actual production requirements occur. On the other hand, the adoption of a commercial solution would force him to anticipate the cost of the second stage configuration and risking to acquire more flexibility than the necessary if the forecasted scenario would not actually occur.

The very last benefit associated to the adoption of reconfigurable robots compared to the manual assembly can be appreciated – for the presented use case – by analysing the great improvement in production efficiency in terms of yields. An experimental campaign has been designed to assess the yields of the robotic based assembly by acting on the variety of cell types to be assembled over a working shift. The introduction of the robot leads on average to increased yields ranging between 5% and 21%. Despite the reconfigurable robot has been designed for ensuring the capability to cope with the production changes and accuracy rather than the reduced lead times, the robot reliability and repeatability contributes to lower the scraps and concurrently running some production tasks which cannot be parallelized by a human operator.

6. Conclusions and future works

The current research presents a configuration approach for Reconfigurable Industrial Robots. The proposed model is formulated as a Two Stage Mixed Integer Linear Programming Model and allows to select and configure the modules type and numbers to form a robotic arm. Such modules are identified to match evolving production requirements which typically characterize the manufacturing sector.

The major contribution of this research is to investigate and proof the competitive benefits of reconfigurability for industrial robots operating in manufacturing. Reconfigurable robot solutions

present significant advantages under the economic perspective as well as reliability and accuracy features which are profoundly better than the current state of the art. With reference to the presented industrial case, the stochastic formulation of the configuration model provides the ability to determine a robot configuration which can be on average close to the 46% cheaper than traditional industrial robot solutions; this is possible by building a robotic arm composed by the set of modules which better suites the current and future production requirements. This economical benefit becomes even larger if it is considered that the extra modules cover possible maintenance activities that in traditional robotic applications would prevent the ability to work.

If compared to other reported prototypes of reconfigurable robots designed for industrial applications, the proposed approach is more than a feasibility study as it conceived as preliminary turnkey solution and tool for the robot designer to generate a solution that is robustly capable to operate in a manufacturing environment for a defined planning horizon.

Future work will specifically focus on the study of the dynamic behaviour analysis and control of this new generation of industrial reconfigurable robots.

Acknowledgements

The research has been partially funded by FP7 European Project entitled white'R - white room based on Reconfigurable robotic island for optoelectronics.

References

[1] Y. Koren, U. Heisel, F. Jovane, T. Moriwaki, G. Pritschow, G. Ulsoy, H. Van Brussel, Reconfigurable manufacturing systems, CIRP Ann. - Manuf. Technol. 48 (2) (1999) 527–540.
[2] M. Yim, W. Shen, B. Salemi, D. Rus, M. Moll, H. Lipson, E. Klavins, G. Chirikjian, Modular self-reconfigurable robot systems, IEEE Robot. Autom. Mag. 14 (1) (2007) 43–52.
[3] K. Stoy, D. Brandt, D.J. Christensen, Self-Reconfigurable Robots, An Introduction, The MIT Press, 2010.
[4] T. Fukuda, S. Nakagawa, Y. Kawauchi, M. Buss, Structure decision method for self organizing robots based on cell structures-CEBOT, in: Proceedings of the IEEE International Conference on Robotics and Automation, Scottsdale, AZ, 1989, pp. 695–700.
[5] M. Yim, New locomotion gaits, in: Proceedings of the IEEE International Conference on Robotics and Automation, San Diego, CA, 1994, pp. 2508–2514.
[6] S. Murata, E. Yoshida, A. Kamimura, H. Kurokawa, K. Tomita, S. Kakaji, M-TRAN: self-reconfigurable modular robotic system, IEEE/ASME Trans. Mechatron. 7 (4) (2002) 431–441.
[7] A. Castano, A. Behar, P. Will, The CONRO modules for reconfigurable robots, IEEE/ASME Trans. Mechatron. 7 (4) (2002) 403–409.
[8] M. Winkler, J. Brgensen, H.E.H. Hautop, M. McKinney, Modular ATRON: modules for a self-reconfigurable robot, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan, 2004, pp. 2068–2073.
[9] A. Lyder, R. Mendoza Garcia, K. Stoy, Mechanical design of Odin, an extendable heterogeneous deformable modular robot, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, 2008, pp. 883–888.
[10] A. Sproevitz, A. Billard, P. Dillenbourg, A. Ijspeert, Roombots-mechanical design of self reconfiguring modular robots for adaptive furniture, in: Proceedings of the IEEE International Conference on Robotics and Automation, Kobe, Japan, 2009, pp. 4259–4264.
[11] J. Davey, N. Kwok, M. Yim, Emulating self-reconfigurable robots – design of SMORES system, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, Portugal, 2012, pp. 4464–4469.
[12] K. Gilpin, K. Kotay, D. Rus, Miche: Modular shape formation by self-disassembly, in: Proceedings of the IEEE International Conference on Robotics and Automation, Rome, Italy, 2007, pp. 2241–2247.
[13] M. Yim, D. Duff, K. Roufas, Polybot: a modular reconfigurable robot, in: Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, CA, 2000, pp. 514–520.
[14] J. Liedke, H. Worn, Heinz, CoBoLD — A bonding mechanism for modular self-reconfigurable mobile robots, in: Proceedings of the IEEE International Conference on Robotics and Biomimetics, Phuket, Thailand, 2011, pp. 2025–2030.
[15] C.E. Thorne, N. Skorodinski, H. Tipton, T. Van Schoyck, M. Yim, Brake design for dynamic modular robots, in: Proceedings of the IEEE International Conference on Robotics and Automation, Anchorage, AK, 2010, pp. 3135–3140.
[16] M. Vespignani, E. Senft, S. Bonardi, R. Moeckel, A.J. Ijspeert, An experimental study on the role of compliant elements on the locomotion of the self-reconfigurable modular robots Roombots, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 2013, pp. 4308–4313.
[17] M.G. Catalano, G. Grioli, M. Garabini, F. Bonomo, M. Mancinit, N. Tsagarakis, A. Bicchi, VSA-CubeBot: A modular variable stiffness platform for multiple degrees of freedom robots, in: Proceedings of the IEEE International Conference on Robotics and Automation, Shanghai, China, 2011, pp. 5090–5095.
[18] F. Aghili, K. Parsa, A reconfigurable robot with lockable cylindrical joints, IEEE Trans. Robot. 25 (4) (2009) 785–797.
[19] P. Xinan, W. Hongguang, J. Yong, Y. Cen, Research on kinematics of modular reconfigurable robots, in: Proceedings of the IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems, Kunming, China, 2011, pp. 91–96.
[20] T. Huang, M. Li, X.M. Zhao, J.P. Mei, D.G. Chetwynd, S.J. Hu, Conceptual design and dimensional synthesis for a 3-DOF module of the TriVariant-a novel 5-DOF reconfigurable hybrid robot, IEEE Trans. Robot. 21 (3) (2005) 449–456.
[21] W. Wenqiang, G. Yisheng, L. Huaizhu, S. Manjia, Z. Haifei, Z. Xuefeng, Z. Hong, Task-oriented Inverse Kinematics of Modular Reconfigurable Robots, in: Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Wollongong, Australia, 2013, pp. 1187–1192.
[22] R. Koker, A genetic algorithm approach to a neural-network-based inverse kinematics solution of robotic manipulators based on error minimization, Inform. Sci. 222 (2013) 528–543.
[23] Q. Guifang, S. Guangming, Z. Jun, S. Hongtao, G. Jian, W. Weiguo, Design of a self-reconfigurable wireless network system for modular self-reconfigurable robots, in: Proceedings of the IEEE International Conference on Robotics and Biomimetics, Guangzhou, China, 2012, pp. 1337–1342.
[24] M. Bordignon, K. Stoy, U.P. Schultz, Generalized programming of modular robots through kinematic configurations, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, California, 2011, pp. 3659–3666.
[25] S. Castro, S. Koehler, H. Kress-Gazit, High-level control of modular robots, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, California, 2011, pp. 3120–3125.
[26] M. Hofbaur, M. Brandstotter, S. Jantscher, C. Schörghuber, Modular re-configurable robot drives, in: Proceedings of the IEEE International Conference on Robotics Automation and Mechatronics, Singapore, 2010, pp. 150–155.
[27] V. Vonasek, M. Saska, K. Kosnar, L. Preucil, Global motion planning for modular robots with local motion primitives, in: Proceedings of the IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 2013, pp. 2465–2470.
[28] H. Feili, S. Wei-Min, On the complexity of optimal reconfiguration planning for modular reconfigurable robots, in: Proceedings of the IEEE International Conference on Robotics and Automation, Anchorage, AK, 2010, pp. 2791–2796.
[29] Y. Lin, F. Xi, R. Mohamed, X. Tu, Calibration of modular reconfigurable robots based on a hybrid search method, ASME J. Manuf. Sci. Eng. 132 (6) (2010) 061002.
[30] R. Mohamed, F. Xi, D. Finistauri, Module-based static structural design of a modular reconfigurable robot, ASME J. Mech. Des. 132 (1) (2010) 014501.
[31] J. Sulzer, I. Kovač, Enhancement of positioning accuracy of industrial robots with a reconfigurable fine-positioning module, Precis. Eng. 34 (2) (2010) 201–217.
[32] W. Yao, J.S. Dai, T. Medland, G. Mullineux, A reconfigurable robotic folding system for confectionery industry, Ind. Robot: Int. J. 37 (6) (2010) 542–551.
[33] S. Tabandeh, W. Melek, C.M. Clark, An adaptive niching genetic algorithm approach for generating multiple solutions of serial manipulator inverse kinematics, Robotica 28 (4) (2010) 493–507.
[34] ⟨www.traclabs.com⟩.
[35] ⟨http://www.schunk.com/schunk_files/attachments/ModularRobotics_2010-06_EN.pdf⟩.
[36] ⟨www.modbot.com⟩.
[37] Z.M. Bi, W.J. Zhang, Concurrent optimal design of modular robotic configuration, J. Robot. Syst. 18 (2) (2000) 77–87.
[38] A. Valente, I. Brugnetti, M. Zamboni, M. Colla, Design and optimization of compact and lightweight modular joints for High Accuracy Reconfigurable Industrial Robots, in: Proceedings of the ICRA 2016 – IEEE International Conference on Robotics and Automation (to be published).
[39] J.R. Birge, F.V. Louveaux, Introduction to Stochastic Programming, Springer, London, 1997.
[40] T. Tolio, A. Valente, A stochastic programming approach to design the production system flexibility considering the evolution of the part families, IJMTM Int. J. Manuf. Manag. Special Issue Reconfig. Manuf. Syst. 17 (1) (2009) 42–67.
[41] W. Terkaj, T. Tolio, A. Valente, Design of focused flexibility manufacturing systems (FFMSs), In: Design of Flexible Production Systems – Methodologies and Tools, Berlin, Heidelberg, 2009, pp. 137–190 (Chapter 7).
[42] S. Dunnill, D.J. Morrison, I. Brugnetti, M. Colla, A. Valente, H. Bikas, Reconfigurable automated island for (dis)assembly of a variety of silicon solar cell geometries for specialist low power and CPV applications, in: Proceedings of the European PV Solar Energy Conference and Exhibition, Amsterdam, Netherlands, 2014, Available on-line on September 2014.