Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia CIRP 00 (2014) 000–000

www.elsevier.com/locate/procedia

Variety Management in Manufacturing. Proceedings of the 47th CIRP Conference on Manufacturing Systems

Workstation configuration and process planning for RLW operations

Gábor Erdős[a], Csaba Kardos[a*], Zsolt Kemény[a], András Kovács[a], József Váncza[a,b]

[a]Fraunhofer Project Center on Production Management and Informatics, Institute for Computer Science and Control, Hungarian Academy of Sciences, H-1111 Budapest, Kende u. 13-17, Hungary
b Department of Manufacturing Scence and Technology, Budapest University of Technology and Economics, H-1111 Budapest, Egry József u. 1. Hungary

* Corresponding author. Tel.: +36-1-279-6181; fax: +36-1-466-7503. *E-mail address:* csaba.kardos@sztaki.mta.hu

Abstract

The application of Remote Laser Welding (RLW) has become an attractive assembly technology in various branches of industry, as it offers higher efficiency at lower costs compared to traditional Resistance Spot Welding (RSW) when high volumes of sheet metal assemblies are to be produced. However, the introduction of RLW technology raises multiple new issues in designing the configuration, the layout, and the behavior of the assembly system. Since configuring an RLW workstation and planning the welding process are closely interrelated problems, a hierarchical decision process must be applied where configuration and planning go hand in hand. The paper presents a hierarchical workflow for workstation configuration and process planning for RLW operations, and proposes methods for solving the decision problems related to each step of this workflow. A software toolbox is introduced that has been developed to facilitate a semi-automatic, mixed-initiative workstation design and to guide the expert user throughout the configuration, planning, programming, evaluation, and simulation of the RLW workstation. A case study from the automotive industry is presented, where the software tools developed are applied to configuring and planning the behavior of an RLW workstation that replaces RSW technology in assembling a car door.

Selection and peer-review under responsibility of the International Scientific Committee of "The 47th CIRP Conference on Manufacturing Systems" in the person of the Conference Chair Professor Hoda ElMaraghy.

Keywords: remote laser welding; workstation configuration; task sequencing

1. Introduction

The technology of remote laser welding (RLW) is an emerging option for replacing traditional resistance spot welding (RSW) in industrial applications, as it offers numerous advantages and introduces new opportunities in product design and assembly. The RLW process consists in welding by heat delivered by a laser beam emitted from a laser head. In contrast to earlier laser welding technologies, the focal length of the laser beam in RLW is higher, typically around 1 meter [1], and the beam is deflected and delivered by a scanner system. The scanner system is composed of two mirrors which are mounted to the end effector of an industrial robot via rotary joints [2].

This concise description of the technology comprises the most notable differences between RLW and RSW, from which the advances and difficulties of utilizing RLW stem. While RSW requires contact and access to both sides of the materials to be joined by a large welding gun, RLW enables contactless welding with single-sided access in the narrow line of the laser beam [3, 4]. The better accessibility of the stitches allows higher freedom in part design, which can be turned into end products that serve better the market needs, e.g., lighter yet stiffer car bodies [5]. Combined with increased focal length, the size of the resulting working volume is suitable for enclosing large workpieces, e.g., car body components [2]. Furthermore, the components of the scanner system that are responsible for focusing and guiding the laser beam have low inertia, which allows high-speed beam positioning and

2212-8271 © 2014 The Authors. Published by Elsevier B.V.
Selection and peer-review under responsibility of the International Scientific Committee of "The 47th CIRP Conference on Manufacturing Systems" in the person of the Conference Chair Professor Hoda ElMaraghy.

movement. This not only increases the welding speed but also decreases the non-productive times of the welding process, resulting in a significant reduction in cycle time [2, 4, 6].

Considering the advantages of the technology, application of RLW is prevailing in car body manufacturing. However, in contrast to the technological benefits offered, introducing RLW into manufacturing requires a high initial investment due to the cost of the laser source, the scanner system, and the complex fixture [7]. In order to make RLW a financially feasible alternative of RSW, the higher investment costs have to be returned by cycle time reduction [2, 4, 8].

The paper investigates the problem of workstation configuration and planning for RLW, and proposes an integrated workflow for solving it, together with efficient methods and a decision support tool for each step of the workflow. The paper is structured as follows. In the next section, a detailed problem statement is given. In Section 3, an integrated decision workflow is introduced, and methods are proposed for solving the decision problems related to each step of the workflow. Section 4 introduces a novel representation that captures the evolution of the workstation configuration and the motion plan throughout the workflow, terminating in the final robot program code. Section 5 provides implementation details and presents the application of the developed tools in a case study. Finally, conclusions are drawn.

2. Problem statement

RLW operations are executed in a dedicated *workstation*. Throughout the paper, we assume that the workstation contains a single RLW robot, and one workpiece is processed at a time. Below, we define the problem of configuring an RLW workstation and planning its processes in order to be able to solve a specified assembly process by RLW [9].

The inputs of the problem are product related: geometric models of the workpiece and the fixture, as well as the structured description of the welding operation. The operation consists of two sub-operations: dimpling, i.e., producing small "bumps" that maintain the gap between the metal sheets assembled; and welding disjoint stitches (linear or circular) that join the sheets. Furthermore, constraints defined by the surrounding manufacturing system of the workstation also have to be provided as an input.

The defined inputs specify constraints (e.g., in terms of size, geometric arrangement) which have to be satisfied and optimization objectives (minimizing costs, cycle time) for the task of configuration and process planning. These together form a complex task composed of a set of subproblems. Solving these subproblems requires different engineering principles to be adopted.

The variety of decisions to be made, such as component selection, placement, and motion planning, requires a decomposition approach, and an adequate workflow that structures these decisions. In our research, the workflow has two main tracks: one responsible for workstation configuration and another for process planning and off-line robot programming.

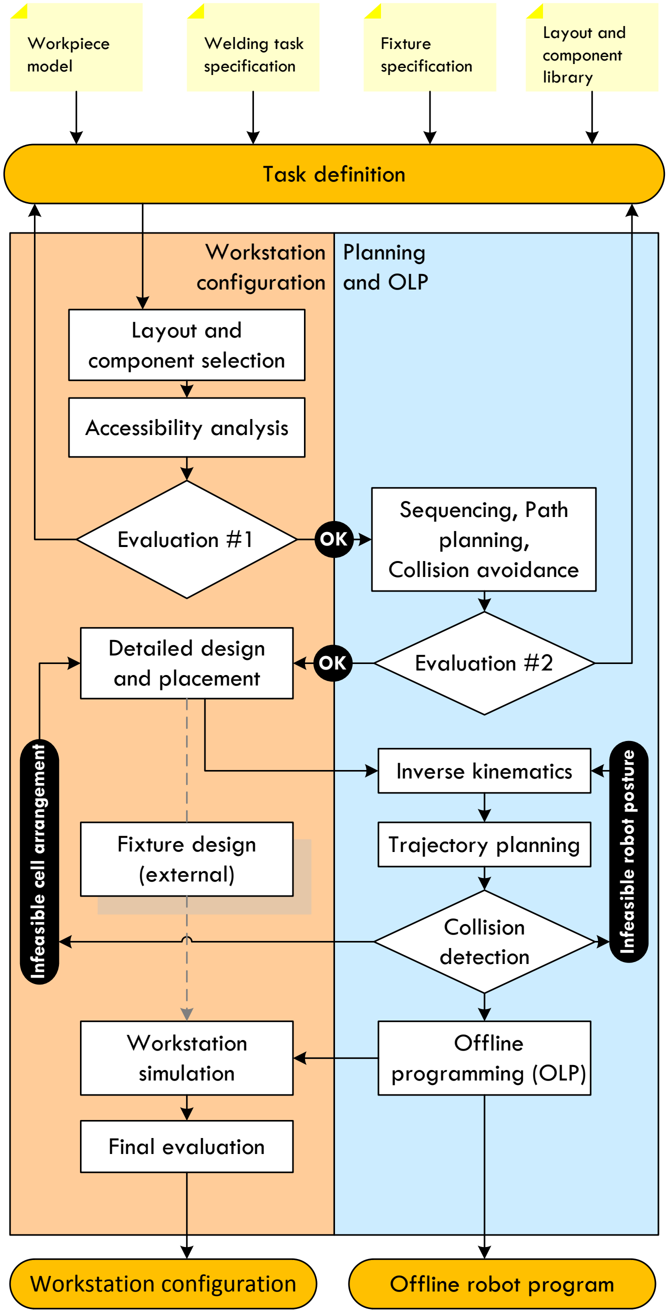

Fig. 1. The defined integrated workflow displays the two main tracks and the components of the solution.

Along with decomposition, defining a workflow which supports the hierarchical refinement of the solution is also desirable as it offers a step-by-step evolution of the solution. Application of generic representation methods provides better connectivity between the components of the solution.

Supporting mixed-initiative problem solving allows human interaction, which is desirable since uncaptured pieces of expert knowledge can also be utilized in the course of a complex solution process. However, this demands a proper graphical user interface and short response times from the solution.

3. Defining the workflow

This section introduces the proposed integrated workflow and details its currently implemented phases. The structure of the workflow is shown in Fig. 1.

3.1. Input data definition

One part of the *input data* refers to the task specification, while the other contains those elements of a component library that are available for the workstation configuration.

Task definition consists of the specification of the welding tasks and the appropriate geometric models. The specifications of the welding tasks describe each stitch or dimple in terms of type, location, normal vector, length, maximal allowed inclination angle, prescribed welding speed and welding power.

The geometric models represent the workpiece and the corresponding fixtures and determine the coordinate frames of the stitch geometries, too. Fig. 2 shows a sample workpiece in its fixture, and an example of a particular welding task.

Since an RLW workstation is, in most cases, incorporated into a manufacturing system, one has to make sure that the workstation fits into the manufacturing system with respect to its physical dimensions, the direction of the material flow and the rough description of operations within the workstation. This information can be formalized by means of *layout patterns*, which are provided along the available elements of the component library. As for these elements, the static and dynamic components have to be distinguished. In order to represent a static workstation component (e.g., a wall) a pure 3D model is eligible, but in order to capture the behavior of dynamic workstation components (e.g., robot, turntable), a more complex representation is needed.

3.2. Workstation configuration

After having a completely defined welding task, the first step is to build up the workstation, as all the upcoming steps of the workflow rely on the parameters of the selected components. Departing from the layout pattern, which defines a rough blue print and generic motion plan of the workstation, as well as the task specification provided by the input, it is possible to translate them into *constraints* imposed on the actual components of the workstation. Specifically, the following constraint types have been identified:

- *Layout type* constraints: stemming from the provided layout pattern, the maximal size and the initial location of the elements, as well as the temporal relations between the operations of the workstation can be constrained.
- *Cycle time*: the upper limit for the cycle time of the workstation.
- *Floor space*: geometric constraint defined by the external environment of the workstation.

Applying the defined constraints, the workstation configuration problem can be formulated as a constrained optimization problem. The decision variables are the "dummy" components of the predefined layout pattern, and the objective function is minimizing the cycle time.

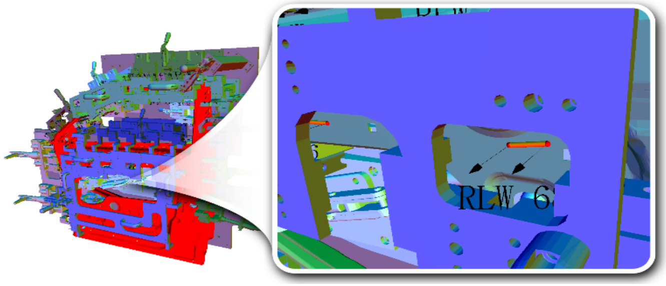

Fig. 2. The geometric representation of the welding task specification displays a workpiece (red), the fixture and the welding stitches.

The result of the component selection is the predefined layout filled up with the selected components, which are tailored to the welding task specification and the operation sequences.

3.3. Accessibility analysis and path planning

Path planning addresses the computation of the robot path for the selected welding robot in the workpiece coordinate system, based on the welding task specification, the geometrical models of the workpiece and the fixture, and the robot parameters.

In order to guarantee that a feasible, collision-free robot path exists, the consistency of the input data must be verified by checking that each welding stitch is accessible. Accessibility analysis detects possible collisions between (1) the scanner head and the workpiece or fixture, and (2) the laser beam and the workpiece or fixture. Further, less critical types of collisions can only be checked for only after computing the inverse kinematics. The welding task specification is accepted as feasible if, for each stitch, there is an adequate access volume that is collision free, and meets the technological constraints on inclination angle and focus range.

Once the accessibility of the stitches is ensured, the sequencing of the welding stitches and the computation of the robot path are solved in an integrated way [10]. Since the technology allows welding complete assemblies with numerous stitches in geometrically complex fixtures in one operation, collision avoidance is one of the key challenges. The implemented planner finds collision-free robot paths while minimizing the welding cycle time.

3.4. Detailed placement

The result of path planning is a collision-free cycle time optimized path of the scanner head. In the next step, an appropriate placement of the workpiece and its fixture has to be found inside the working area of the robot. Note that earlier, during the input data definition, departing from its dynamic model, the kinematic model and the workspace of the robot have already been generated, so at this stage, the working area of the robot is defined and available for solving the placement problem.

Finding a feasible placement for the workpiece plus its fixture within the workstation is a geometrically highly constrained problem. The criteria for a *feasible placement* are as follows:

- The path of the scanner head has to be completely included in the workspace of the robot.
- All potential collisions of the workpiece (plus fixture), the robot and other, both static (e.g., box of the laser source) and dynamic (e.g., rotating table) devices have to be avoided.

In order to solve this problem, the application of an interactive method was developed that supports the manual adjustment of the position and posture of the workpiece, checks for the satisfaction of above constraints by proximity calculations and visualizes the results via intuitive graphics.

Detailed placement does not end here, as it has to be ensured that there is no collision between other workstation components. Repositioning these components (if needed) can be carried out by using a similar procedure that checks for collisions.

3.5. Calculation of inverse kinematics

Having once obtained the path of the laser scanner in the Cartesian space of the workpiece, as well as the placement of this path in the workspace of the robot, the corresponding robot motion—i.e., a motion sequence prescribed for the joint variables—has to be calculated. The calculation of the inverse kinematics is heavily dependent on the kinematic structure of the selected robot. A detailed description of the inverse calculation for a typical RLW robot used in our current application can be found in [3].

4. Modeling the structure and behavior of the workstation

Following the steps of the defined workflow, new results are generated in each step, which enrich and add new details to the model of the workstation under configuration. However, the results are in various formats and have different representations (e.g., textual parameter lists, graphics and kinematic models), thus it is desirable to have a workstation representation which captures all these aspects within a single model of the workstation. The requirements on this representation are the following:

- It should be able to represent the static structure of the workstation (i.e., the layout of the workstation).
- It should be able to represent the dynamic behavior of the workstation (i.e., the kinematic models and operations of the robot and other dynamic components).
- In order to follow the steps of the workflow, the representation needs to be dynamically extendable.
- It should be able to provide a presentation of the workstation to the user at any stage of the workflow.

In order to meet the above requirements, a novel workstation model was developed. This is based on the application of *parametric linkage models* which are built up by defining parameterized kinematic pairs. These kinematic pairs represent reference frames connected by parametric homogeneous transformation matrices. By using rotational and translational kinematic pairs the dynamic components of a workstation can be modeled as traditional mechanisms. Moreover, the application of parametric transformation matrices allows extending the linkage definition beyond traditional mechanisms: by using fixed kinematic pairs static structures can be represented (i.e., the layout of a workstation). Consequently, with the application of various kinematic pairs, a workstation linkage is able to capture both the static and the dynamic structure of the workstation and can be enriched step by step in parallel with the evolution of the workflow by adding new kinematic pairs. Fig. 3 shows the linkage mechanism of a typical RLW workstation.

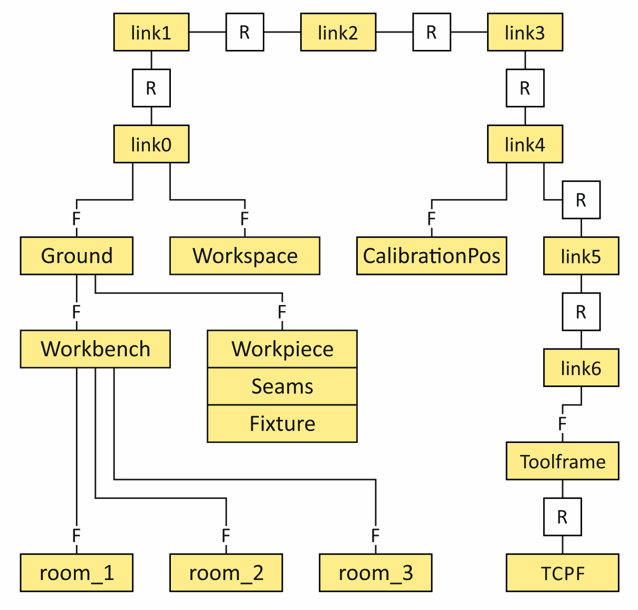

Fig. 3. The structure of a linkage which represents a complete workstation. The elements "link1" to "link4" denote the arms of the robot joined by rotational joints (R). The mirrors of the scanner system are "link5" and "link6". The static elements of the workstation are connected by fixed joints (F).

4.1. Presentation of the linkage

The workstation is subject to configuration and planning decisions as one advances through the overall workflow. Hence, the actual linkage model should be properly visualized, even if it captures a partial solution. Furthermore, it should be accessible for interactive manipulation, too. Specific requirements on the presentation of the workstation linkage are as follows:

- The linkage is a complex object, comprising numerous elements of mixed types. The presentation should keep this original structure and facilitate the selection and manipulation of specific (types of) elements.
- Rendering any element of the linkage in a prescribed pose, as well the animation of poses is a must, if one is going to have feedback about the dynamic behavior of the workstation.
- Feedback from presentation to representation is to be provided for cases when the user makes changes to the presentation of the linkage (e.g., when changing the position of a robot in a workstation, or solving the placement problem via interactive manipulation of the workpiece).
- Visualization should use some standard format.

In this work, VRML 97 (Virtual Reality Modelling Language) rendering was selected as the way for presenting the linkages. VRML 97 is a standard rendering file format which can be visualized by third party software even in a Web browser. Also, it is a standard text based file that can easily be converted to other rendering file formats such as X3d or HTML5 that have been designed for carrying multimedia content [11]. (See Fig. 4 for a VRML example.)

5. Implementation and case study

Supporting different stages of the workflow calls for the application of a number of specialized software modules. However, in order to satisfy the requirements of the initial problem statement, integration of the applied tools is fundamental. Hence, the following system design requirements have to be complied with:

- The system should support the complete integrated workflow, even though there are two modules which have not been implemented yet.
- It should allow the use of external software, such as optimization or geometric computation tools.
- The system should have an extendable architecture.
- It should provide a graphical environment for interactive, mixed-initiative problem solving.

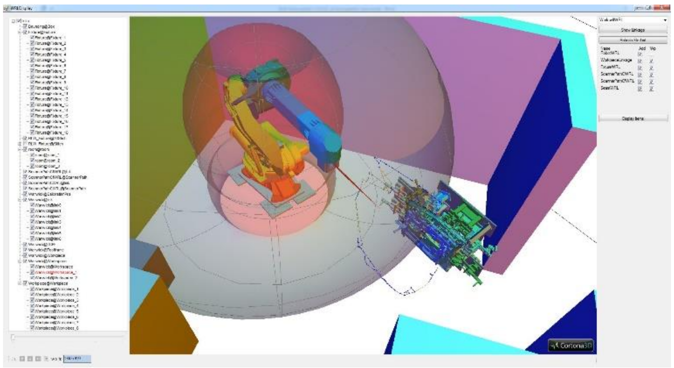

Fig. 4. Screen showing the VRML representation of the workstation.

Fulfilling these requirements have led to the development of a *software toolbox* which acts as a component integrator and provides the necessary user interfaces as well. The software was developed using Microsoft .NET framework's Visual C# language, as it offers creation of graphical user interfaces and is suitable for integrating different software tools. In the following section the components of the software toolbox are introduced. They are all fully integrated into the toolbox and can be operated through a common graphic user interface where their results are displayed as well.

5.1. Components of the software toolbox

The implementation of the linkage mechanism is relying on *LinkageDesigner*, an application package of *Mathematica* for virtual prototyping of linkages [12]. *LinkageDesigner* is designed to analyze, synthesize and simulate linkages with serial chain, tree and graph structures. By making use of the symbolic calculation capabilities of *Mathematica*, it is able to handle fully parametric models, thus providing sufficient capabilities for modelling a complete workstation. In addition, since the workstation linkage relies on using transformation matrices, the resulting model contains all the necessary information that is required to perform the calculation of the inverse kinematic solution within the *LinkageDesigner* package.

In order to fulfill the requirements on the presentation of the linkage, an enhanced VRML display was implemented, which is built around the ActiveX component of Cortona3D Viewer [13]. The display was, however, tailored for displaying the workstation linkage.

Workstation configuration, as described in Section 3.2, is solved using a constraint programming model developed in the ILOG CP constraint solver engine. For solving performing accessibility analysis and computing the collision-free robot path (see Section 3.3), a customized solver has been developed in C++.

Both the path planning and the workpiece placement rely heavily on using collision detection and proximity queries, implemented in the Proximity Query Package (PQP) [14]. PQP represents 3D objects using triangle mesh models, and performs distance computation among pairs of such models. PQP is used for collision detection between each pair of elements in the workstation.

5.2. Case study

In order to verify and validate the methods, a physical demonstrator case study is currently under implementation. The goal is replacing RSW with RLW in the assembly of a car door in an existing workstation. This demands several changes in the product design, thus resulting in completely new process design and fixtures for the product. (These aspects were, however, out of the scope of the current research.) Our goal is computing a detailed workstation configuration and an optimal robot path, as well as automatically generating the corresponding robot program by the developed software toolbox.

Since the workstation components and their arrangement in the demonstration environment are predefined, component selection was reduced to the re-implementation of the existing workstation in our representation. After calculating the optimal robot path, workpiece placement was executed in the interactive environment of the developed software toolbox.

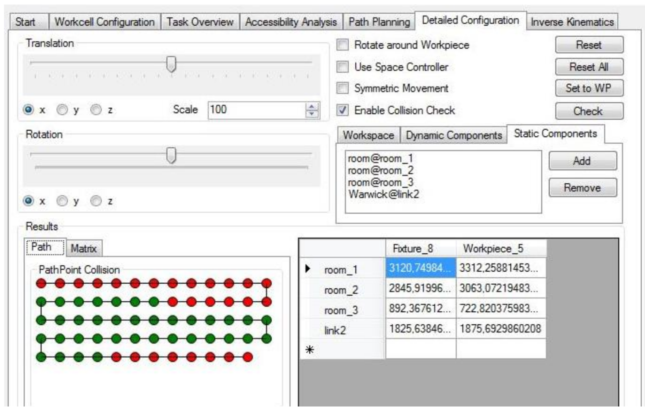

Fig. 5. Workpiece placement supported by collision checks.

Fig. 5 shows a screenshot of interactive workpiece placement supported by automated collision checks. The system computes and displays actual collisions (green circles stand for collision-free sections of the path, red circles denote collisions) and informs the user about the distance of the selected pairs of objects.

Having found a feasible workpiece placement, the calculation of the robot inverse kinematics was executed. The computed joint coordinates are presented in a structured textual format, and the corresponding postures of the robot can be displayed in the VRML display. The automated generation of the robot program code form the motion plan is subject of future work.

6. Conclusions and future work

The paper discussed the problems and requirements of designing a RLW workstation configuration and presented a suitable decomposition of the problem. Based on the decomposition, a workflow has been defined to solve the tasks in a hierarchical and generic way. Following the workflow, the implementation of its elements has been carried out, up to the task of inverse kinematics calculation (the trajectory planning and the offline robot programming have not yet been implemented), and were integrated into a self-developed software toolbox supporting mixed-initiative problem solving. The applied solution was compared to existing task sequencing and path planning methods and was found to outperform them [10]. In addition, the case study involving the aforementioned task sequencing and path planning methods showed a reduction of 30% of the total cycle time compared to the existing RLW planning algorithms.

According to the current stage of development, an obvious target of future work is to complete the implementation of the workflow, with special attention to trajectory planning and the automatic generation of an off-line robot program. Once these remaining steps are completed, the physical realization and testing of the demonstrator workstation will follow.

As explained, the current implementation still requires a significant amount of user interaction. Expecting further improvement in this regard, the possibilities of enhancing the automated decision making/support functions of the system are worth further investigation as well.

Acknowledgment

This work has been supported by EU FP7 grant RLW Navigator No. 285051, and the NFÜ grant ED_13-2-2013-0002.

References

[1] Zäh MF, Munzert UK, Oefele F. Offline concept for optimization of robot paths for remote laser welding without scanners. 10th International Research/Expert Conference "Trends in the Development of Machinery and Associated Technology" 2006; 601–604.

[2] Tsoukantas G., Salonitis K, Stournaras A, Stavropoulos P, Chryssoulouris G. On optical design limitations of generalized two-mirror remote beam delivery laser systems: the case of remote welding. International Journal of Advanced Manufactrung Technology 2007; 32:932–941.

[3] Erdős G, Kemény Zs, Kovács A, Váncza J. Planning of remote laser welding processes, Procedia CIRP 2013; 7:222–227.

[4] Zäh MF, Moesl J, Musiol J, Oefele F. Material processing with remote technology revolution or evolution?, Physics Procedia 2010; 5:19-33.

[5] ElMaraghy H, AlGeddawy T, Azab A, ElMaraghy W. Change in manufacturing – research and industrial challenges. In: ElMaraghy H, editor. Enabling manufacturing competitiveness and economic sustainability; Berlin: Springer; 2011; p. 2–7.

[6] Reinhart G, Munzert U, Vogl W. A programming system for robot-based remote-laser-welding with conventional optics, CIRP Annals - Manufacturing Technology 2008; 57:37–40.

[7] Um J, Stroud IA, Total energy estimation model for remote aaser welding process, Procedia CIRP 2013; 7:658–663.

[8] Ceglarek D, Huang W, Zhou S, Ding Y, Kumar R, Zhou Y. Time-based competition in manufacturing: stream-of-variation analysis (SOVA) methodology – review. International Journal of Flexible Manufacturing Systems 2004; 16:11–44.

[9] Remote Laser Welding (RLW) System navigator for eco and resilient automitve factories, FoF-ICT-1011.7, No. 285051, http://www.RLWnavigator.eu/ , accessed on 06.12.2013.

[10] Kovács A, Task sequencing for remote laser welding in the automotive industry, Proceedings of the Twenty-Third International Conference on Automated Planning and Scheduling 2013; 457–461.

[11] Specification of the Virtual Reality Modeling Language, (VRML), ISO/IEC DIS 14772

[12] Wolfram's Linkage Designer. http://www.wolfram.com /products/applications/linkagedesigner/, accessed on 12.12.2013.

[13] Cortona3D Viewers, http://www.cortona3d.com/cortona3d-viewers, Accessed on 12.12.2013.

[14] Larsen E, Gottschalk S, Lin MC, Manocha D. Fast proximity queries with swept sphere volumes. Proceedings of the IEEE International Conference on Robotics and Automation 2000. 3719–3726.