OPKID: Optimal Kinematic Design of Robots Modelling and simulation of a 6 DOF articulated industrial robot in Delmia

Date: Thu 23:59, 16.12.16

Prof. P. Wenger, Prof. S. Caro, Prof. D. Chablat

Rabbia Asghar, BEng, Ernest Skrzypczyk, BSc

Contents

List of Figures

Introduction

The objective of the assignment was to design and model the kinematics of the industrial 6 DOF articulated robot MOTOMAN MH5LS from YASKAWA using Delmia software in order to perform a welding operation on provided elements in a robotic cell. The parts of the robot, the working environment and welding tool were already provided. Also the joints between parts of the robot were also predefined. Since the assembly of parts and joint definition process was performed in previous work, the provided complete robot was used.

The modelling and simulation was carried out in the following steps in Delmia:

- Kinematic modelling of the robot
- Creating the robotic cell and defining the welding task

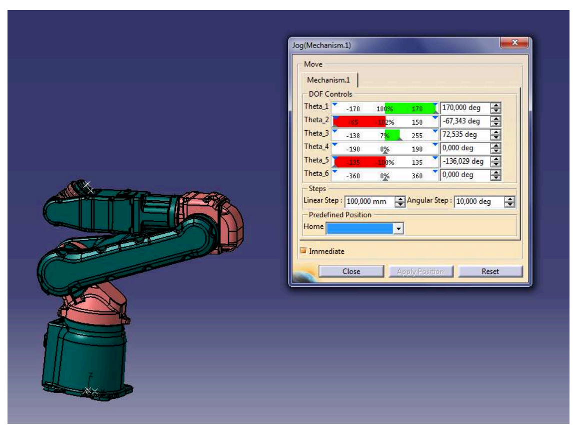
Kinematic modelling of MH5LS

The direct kinematic model of the robot was already defined to some extent, since all the parts of the robots were already assembled and joints were predefined. The six revolute joints in MH5LS were renamed to "theta_n" where $n = 1, .., 6$. Joint limits and joint speeds for each joint were defined as provided in MOTOMAN MH5LS II datasheet flyer, snapshot is shown in (1a). Finally the caution zone for joint limits was kept 0%.

Since the assignment text did not clearly state if the parts needed to be assembled and the joints be defined again, it was assumed, that it was not required. Modelling was done with the provided MOTOMAN product with already defined joint mechanisms. In the end, however, for sake of completion of work the parts were assembled and joints were defined. This robot was not used for the performed task, but since it's the same model, it should be easily replaceable.

Axes	Maximum motion range [°]	Maximum speed [°/sec.]
S	±170	270
L	+150/−65	280
U	+255/−138	300
R	±190	450
B	±135	450
T	±360	720

(a) Joint limits and joint speed limit of MOTOMAN MH5LS II datasheet as provided in datasheet.

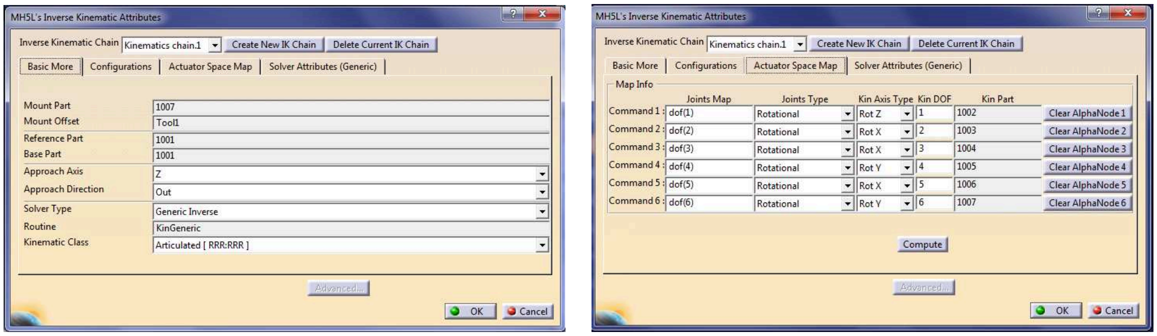


(b) Joints exceeding the limits are shown in red in jog feature.

Figure 1: Joint limits definition in MOTOMAN MH5LS II.

After definition of direct kinematic model, jog feature of the simulation software was used to vary individual angles, reach the set limits and visualize the configuration of robot. Visual indication consisted of the bar of certain joint angle turning red if its limit was crossed (1b).

Before inverse kinematic modelling (IKM) of the robot, frames of interest in the base and end effector were set. The frame of interest in the base can be used to help place the robot in the environment, although other possibilities exist, and helps visualizing its location. The frame of interest in the end effector aids in defining its pose for the inverse kinematic model and later, mounting the welding tool on end effector. Inverse kinematic modelling was done as shown in figure (2).



(a) In basic mode, definition of end-effector(mount part) and base part for definition of inverse kinematic modelling is done.

(b) Kinematic axis types are defined for each joint in reference to global reference axis when all joints are set to 0.

Figure 2: Inverse kinematic modelling definition of MOTOMAN MH5LS II.

From the home position set, where all of the joint angles are at value 0, the appropriate rotational axes for IKM with respect to the absolute frame, could be easily extracted. After the definition of IKM, the jog feature offered option to move the robot in Cartesian space by varying Cartesian coordinates and orientation of the end effector.

Welding process

In order to simulate the welding process in Delmia, tag points were created along the desired path coinciding with the T-shaped plate and the base. Since both elements do not have any curvature and their relative position was orthogonal the resulting contact path was a straight line, as shown in figure (3).

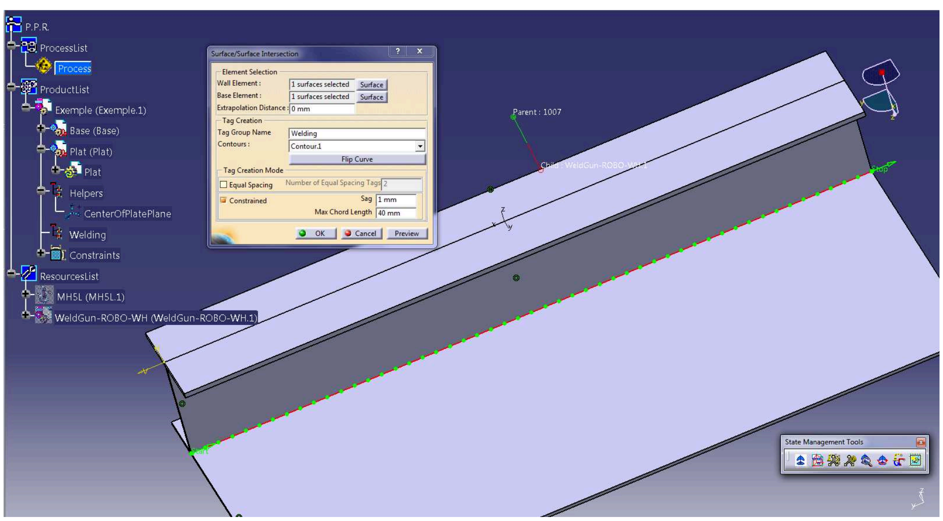


Figure 3: Tag definitions to implement welding process.

Teach a device command was used to analyse the welding process in detail. Using the table format, each operation that is at individual process tag point, can be identified and modified. For the motion, since the trajectory was a straight line, linear motion (LIN) was used for majority of process points. Only the first operation point remained at at joint motion mode (JNT), since when a switch to linear motion is performed several aspects change. First, the interpolation mode is changed from the default mode to *KeepConfig SetTurn* mode. Second, the posture option, ergo the possibility to select different inverse kinematic solution, is disabled. In case of linear motion mode the robot's motion and inverse kinematic solution is determined in the Cartesian space as operational space and based on preceding operation point. This is logical, since the interpolation algorithm is different and dynamic. However any kind of debugging by jumping between non-adjacent operation tags can be misleading or not possible, especially when the analysis mode is turned on with the options to interrupt motion when joint limits are met. The collision detection between all components of the robot itself and the work pieces, as well as joint limit constraints were enabled, so during the welding process, if collision occurred or if any joint limit of the robot was violated, either a warning appeared or the process was interrupted based on specified settings.

Different postures, ergo different IKM solutions, were studied and for different segments of the welding path collision free as well as within joint limit postures were identified. A helpful tool for smoothing the motion of the robot was the interpolate tag orientation command. Weld angle and rake angle defined in the orient tag feature had to be respected, therefore only 1 DOF was available at the end-effector, namely the welding gun could be only rotated around the electrode axis. Still, this was sufficient to manually find solutions at different placements of the robot base that were outside of collisions and yet reached the tag. The solutions proposed by Delmia were sometimes misleading, even though a correct solution was available.

The maximum reach D_M, in the case of MOTOMAN MH5LS defined as maximum working range $D_M = R = 895\ mm$, limits the workspace just enough, so that when the robot is placed on the centre of the plate platform is cannot reach the welding points on the both ends. When collision detection is not activated the manipulator can reach the second and fifth last welding points. Figure (4) shows the workspace of the robot when it is placed on top of T-shaped plate.

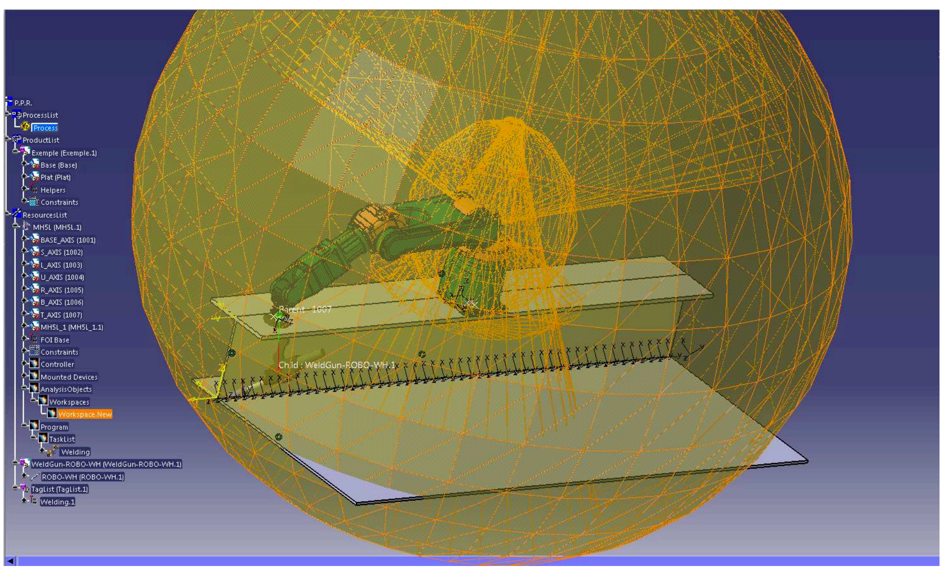


Figure 4: Workspace of MOTOMAN MH5LS.

In order to increase the number of reachable welding points, the position and base orientation of the robot can be modified. In this case the base orientation θ_0 was already optimal, the robot stayed within limits executing motions for the welding process. The position was changed, so that the manipulator was closer to the edge of the platform and had more available reach. Should the placement on the bottom platform be possible, then a position at a distance sufficient to allow the manipulator to maintain its aspect throughout the welding trajectory. This was the assumption for the provided optimal solution.

Question 1: Without moving the robot, on which parameters can we act to increase the number of reachable welded points ?

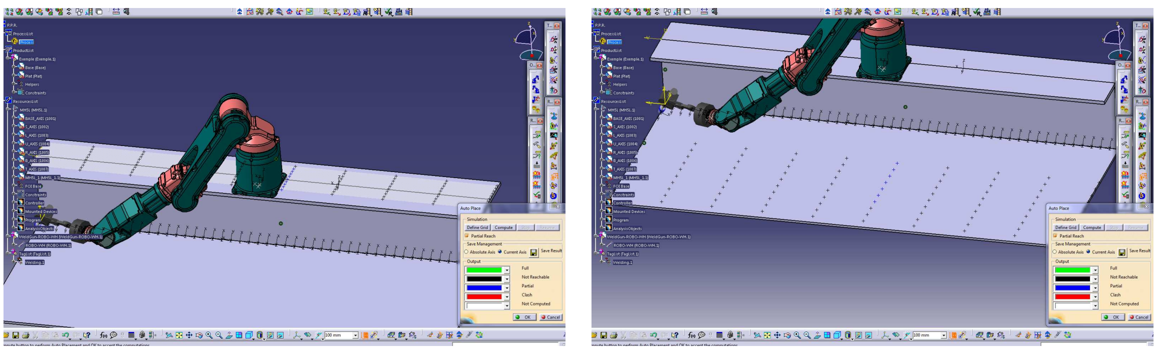
Assuming the possibility of changing the design parameters exists, following considerations should be taken into account.

When the position of the robot is fixed and movement of its base is not possible, one could increase lengths of the links. In this particular scenario the links 2 and 4 are of interest. When increasing the lengths of these links, the original ratio should be taken into account in order to try maintaining a similar workspace. Alternatively the offsets of the joints could be modified. Especially in the case of the offset of the tool on the last link in the chain, an increase, so that the maximum reach D_M of the system is higher could be performed. This might be in fact the most resource efficient solution.

Another possibility would be to introduce a translation of the base along the work piece, for example on a rail. This would allow to keep the orientation of the welding gun without the need of introducing via points as in the proposed solution.

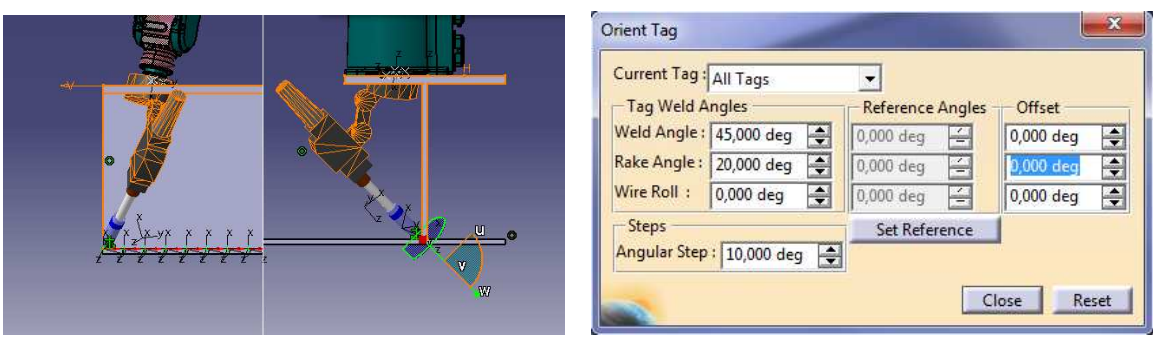
Question 2: Can we change the robot placement to improve it?

Different placements of the robot were tested that included points in the centre of the T-shaped plate, one-third length, in both centre and edge of the plate as well as different placements on the base. The optimal robot placement command in the Delmia was used. The placement grid was defined on the T-shaped plate and also on the base plate to run the algorithm. Optimal placement for partial reachability criteria of spot welds was found when the robot was placed on the base plate. Figure (5) shows the obtained results.



(a) Optimal placement results for T-shaped plate. (b) Optimal placement results for base.

Figure 5: Optimal placement results for potential placements across the T-shaped plate and base.



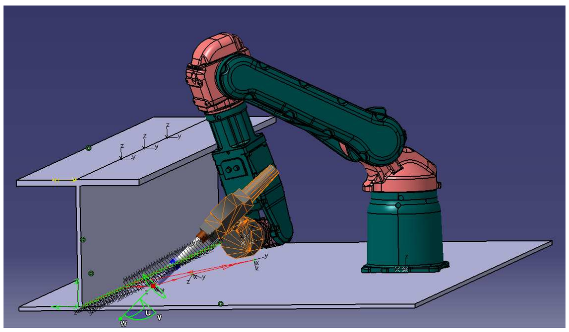
(a) (b)

Figure 6: Weld and rake angles configuration and visualization in the 3D environment.

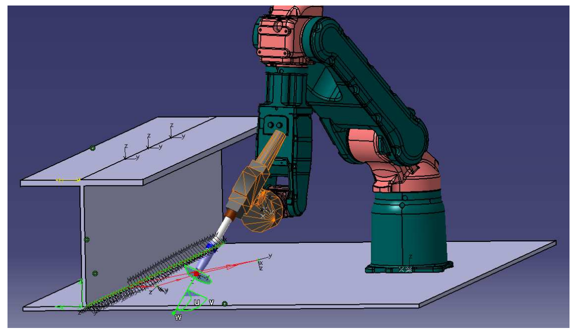
As the robot was placed on the base and moved further from the T-shaped plate, the collision frequency reduced. Moving the robot too far resulted in inability of the robot to reach the start and end tag points. One possibility is to move the robot above the base, however even at the height of T-shaped plate that is

325 mm, the robot arm collides with the base. This is due to the weld and rake angle of the weld gun, the geometry of the robot and especially the welding gun itself.

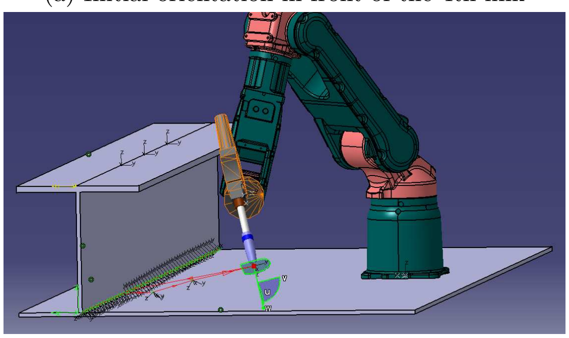
Because of the orientation of the welding gun determined by θ_6, rotation of link 4 determined by θ_4 has to be concluded between the position of the robot and the T-shaped plate. The shape of welding gun prevents free rotation of θ_4, which is the reason additional operation steps with via points have been included. A maximum of all tag points for the predefined welding trajectory were covered, the robot stayed within the joint limits and without collision using the proposed solution. Since the objective was to perform a spot welding task, it was not necessary as in arc welding to continuously follow the trajectory. With help of additional via points the maneuver of the robot was performed, after which the the constraints of welding and rake angles shown in figure (6) were satisfied again. The sequence to perform this action can be seen in provided movie material inside the Videos folder and is illustrated in figure (7).



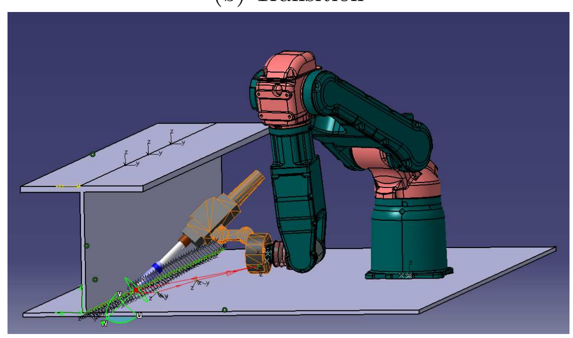
(a) Initial orientation in front of the 4th link



(b) Transition



(c) Transition



(d) Final orientation behind the 4th link

Figure 7: Sequence of additional motions in order to change the orientation of the welding gun without collision.

Conclusions

The practical sessions addressed the challenges involved in the welding process inside the Delmia software. The welding needed to be performed at a specified welding angle and rake angle. In presence of the obstacles, in this case the welding gun itself was often the obstacle, this can be a complex problem for a long path, especially in the case of arc welding, which requires t-connected region of the robot in order to follow the continuous trajectory. Even though the required task was not difficult to perform, a great effort was made trying to consider the objective for an arc welding task, so that a solution would serve both cases of spot and arc welding. Many different placement and postures were explored with the MH5LS robot, but a complete reachability without further user interaction to overcome joint limits, self collision and collision with the plate or base, was not found. The objective required task definition for a spot welding process, which consists of discrete trajectory following, where even the time steps between the welding points can vary. Therefore an optimal robot placement was determined where all trajectory points were reachable. In order to avoid collisions additional via points were created for the task outside of the desired trajectory with the goal of changing the robots postures manually.

Weaknesses of the Delmia software were noted, again the inability to find optimal placement for a given task in particular. Overall the Delmia software framework provides a wide variety of features relevant to modelling of robots and simulation of tasks, like spot welding, in robotics.