ANNALS of the ORADEA UNIVERSITY.
Fascicle of Management and Technological Engineering, Volume VII (XVII), 2008

AN APPLICATION OF ROBOT SIMULATION TO PALLETS

Constantin ISPAS and Bogdan-Ion LUNGU
University POLITEHNICA of Bucharest, Faculty of Engineering and Management of Technological Systems, Department of Machine and Manufacturing Systems
e-mail: ispas1002000@yahoo.fr, lungu_ib@yahoo.com

Keywords: simulation, off-line programming, industrial robot, palletizing

Abstract: This paper presents the environment modeling and the off-line simulation of industrial robots. The analysis starts with the palletizing process which becomes more complicated because of customer's needs. The developing of a modeling environment and an off-line simulation improve the efficiency of the industrial robot programming for palletizing application. The work cell layout will be made using the PC-ROSET software, which operates on a personal computer for the simulation of KAWASAKI industrial robots.

1. INTRODUCTION

Palletizing task is described as an uniform loading process of various products on a pallet. Palletizing is used, in general, for products like boxes, bags, pails and the stacking is made with the help of a pre-determined pattern and a given number of layers.

The use of industrial robots in palletizing processes is an important step for industry because it is necessary to promote efficiency of keeping and shipping tasks. Never the less, palletizing is one of the most monotonous and heavy work in the factory. For this reason, it has been developed an important number of industrial robots with different characteristics to meet the costumers' demands and the continuous process development. Nowadays the types of products and the numbers of case patterns that can be automatic palletized are practically limitless.

Figure 1: Work cell for bag palletizing

Figure 2: Work cell for pail palletizing

The challenges for palletizing process are:

- The cycle time reduction and trajectory optimization for a single cycle of palletizing;
- Maximizing the number of product displacement on a single pallet by modifying the arrangement pattern.
- Calculating the optimal path considering obstacles using a limited configuration of space.
- Minimizing the transition time between two different products in the same manufacturing system.

2. INDUSTRIAL ROBOTS PROGRAMMING

The industrial robots' programming can be made in two ways:

- Manual programming;
- Off-line programming.

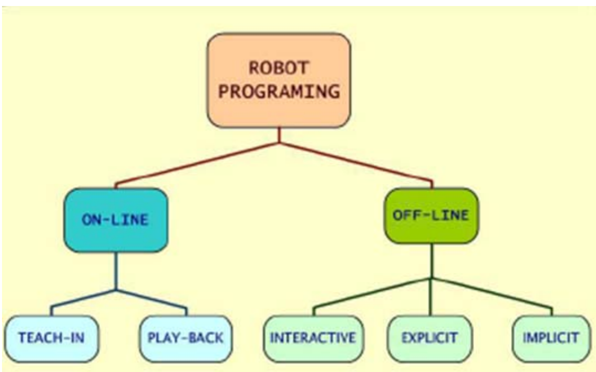


Figure 3: Industrial robots programming

Manual on-line programming refers to physically teaching a robot the required trajectory, through interaction with teach pendant or other similar device (Lee & ElMaraghy, 1990). Robot's controllers are very sophisticated, the commercial programming environments are typically closed systems and the programming languages varies from manufacturer to manufacturer. This type of programming presents the following problems: very slow, it needs for the robot to be available, difficulty in the handling of equipments, need practice in the language used by the robot, and technical knowledge to understand the operation of the equipment. These problems are very expensive in the industry because the productive process needs to stop for the period of time necessary to make the adequate changes.

Off-line method refers to generate task data using a computer and download it to the robot controller. Besides programming the robot, simulation software is a more complex method used for the technological systems' modification and performance checking, evaluation and comparison for various scenarios and obtaining the solutions for an important number of production problems. The robot simulation systems are often referred as CAR (Computer Added Design).

The task is simulated using a virtual layout of the workcell either when only a digital prototype is available or for a physical workcell. In both cases an important component is CAD (Computer Added Design). The modern simulation programs come with a large library of standard components and predefine work cells layouts in order to reduce the time of environment modelling and avoid the risks of errors, like fixtures or clamps being placed in the wrong location or dimension's errors.

This type of programming is very efficient because it provides an interactive environment for the modeling, integration and simulation of the robotic workcell.

Traditional off-line programming does not use the full potential of virtual models and simulation systems in industrial robot applications. The interface between the off-line programming system and the robot controller is today restricted to program transfer. In this way a large amount of information is lost since typical robot programming languages are rather limited. The transfer is made in one direction, from the simulation environment to robot controller, a fact that facilitates the lost of information.

Simulation helps solving some of the on-line programming problems presented above, but at the same time adds newer problems, such as:

- The accuracy of the modeled workcell is low so it requires a calibration when developed in a physical world;
- The software's errors and programming bugs;
- Difficult and time consuming to create an accurate digital model;
- The world has to be static with high accuracy in pre-manufacturing and clamping.

The differences between the on-line and off-line programming and the practical characteristics of off-line programming are shown in Table 1.

Table 1

ON-LINE	OFF-LINE	OFF-LINE PROGRAMMING ADVANTAGES	OFF-LINE PROGRAMMING DISADVANTAGES
Sequential operation mode	Parallel working mode	Increases robot's efficiency	High initial costs
Operational robots requested	No physical robot and workcell's components	Provides a safe environment for simulation	Fast information exchanges between engineering departments
Attention with errors	Early examinations and optimizations.	Integrated CAD-CAM systems	Reorganization
Requires staff for supervising	Quality information regarding the process	Simplification of complex tasks	Necessity of robot's calibration in real working environment
Extra time for workcell's physic arrangement	Compound vision of the simulation.	Verification of programs before loading it into robot controller Fast and easy optimization	Low precision
	Saving costs	Analysis provided by simulation software	Software errors and programming bugs

3. PALLETIZING PROCESS SIMULATION USING PC-ROSET KAWASAKI SOFTWARE

PC-ROSET is a PC based off-line teaching tool for KAWASAKI robots. Teaching data can be sent to robot's controller and it can be executed with a precise simulation and an accurate cycle time calculation. It uses the same internal software as in an actual KAWASAKI robot controller that allows a precise simulation.

Major functions of PC-Roset are:

- Teaching procedures: there are two ways of teaching robot's poses: from Teaching Panel on AS Control Panel and from Data Viewer;
- Process simulation: the AS Control Panel is linked with Scene Viewer and the simulation results are reflected in the Scene Viewer graphics, so the interface between robot arm, works, tool and surroundings can be checked.

The modeling and task simulation using PC-ROSET are made as presented in Figure 3.

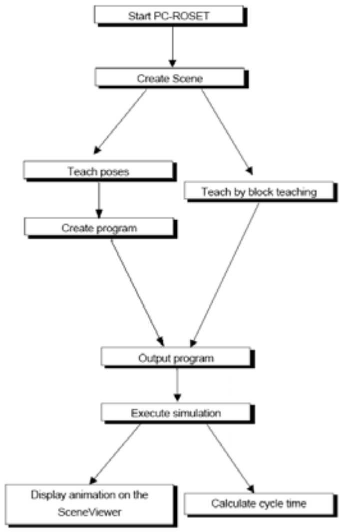


Figure3: PC-ROSET's work procedure

The Scene Creation step consists in choosing the robot type and the workcells components. This operation can be made using either PC-ROSET's Simple Modeler function or any other CAD software that uses the VRML extension type. Simple modeler can add new models to the workcell like: cylinder, cuboids and regular prism.

The robots are chosen considering technical characteristics and specific payloads. Different tools can be attached to the robot depending on palletizing process. For example, in figure 4 is shown a tool for pails handling and figure 5 is shown a tool for glass handling.

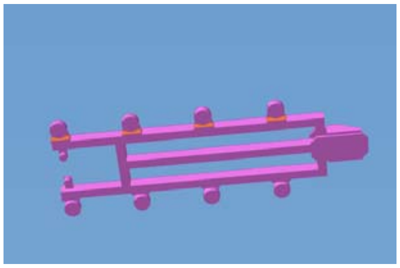


Figure 4: Glass handling toll

Figure 5: Pail handling tool

In figure 6 is presented a workcell for box palletizing using a KAWASAKI ZD130S robot designed for payloads up to 130 kilograms.

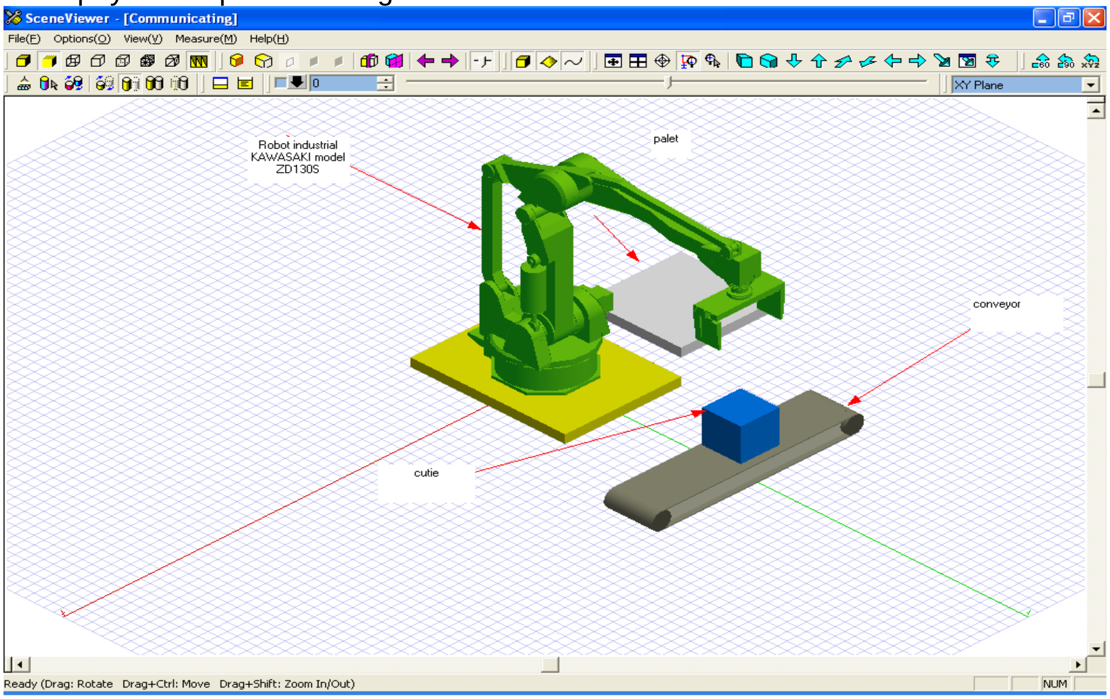


Figure 6: Box palletizing workcell layout

The robotic workcell is composed by the following elements:

- KAWASAKI ZD130S industrial robot;
- Conveyor;
- Box;
- Pallet.

In this workcell a box is brought to robot's workplace station by a conveyor. The KAWASAKI ZD130S uses a two-finger gripper to pick-up the box and place it on a pallet. The pattern for placing the boxes on the pallet can be easily made by modifying the program. A real process is shown in figure 7 and in figure 8 is shown the virtual arrangement of the boxes on a pallet.

Figure 7: Real environment box palletizing

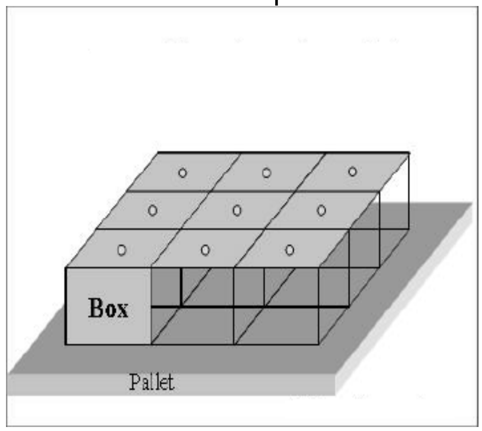
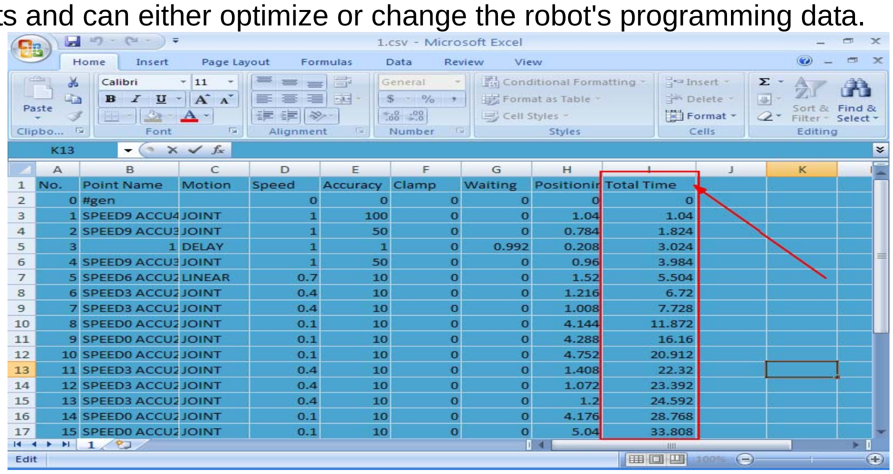


Figure 8: Virtual layout display of boxes on the pallet

When the overall system has been simulated, the defined robotic points and motions can be translated into actual robotic control programs.

Cycle time calculation function is useful after the teaching of robot's movements. It outputs the list of cycle times for each motion step, as shown in figure 9. Simulated results are displayed in CSV format and it can be exported in Excel format, providing information about the speed, accuracy, waiting, positioning time, total time of the process, etc. In this way the engineer has control on simulation results and can either optimize or change the robot's programming data.



No.	Point Name	Motion	Speed	Accuracy	Clamp	Waiting	Positionir	Total Time
0	#gen		0	0	0	0	0	0
1	SPEED9 ACCU4	JOINT	1	100	0	0	1.04	1.04
2	SPEED9 ACCU3	JOINT	1	50	0	0	0.784	1.824
3	1	DELAY	1	1	0	0.992	0.208	3.024
4	SPEED9 ACCU3	JOINT	1	50	0	0	0.96	3.984
5	SPEED6 ACCU2	LINEAR	0.7	10	0	0	1.52	5.504
6	SPEED3 ACCU2	JOINT	0.4	10	0	0	1.216	6.72
7	SPEED3 ACCU2	JOINT	0.4	10	0	0	1.008	7.728
8	SPEED0 ACCU2	JOINT	0.1	10	0	0	4.144	11.872
9	SPEED0 ACCU2	JOINT	0.1	10	0	0	4.288	16.16
10	SPEED0 ACCU2	JOINT	0.1	10	0	0	4.752	20.912
11	SPEED3 ACCU2	JOINT	0.4	10	0	0	1.408	22.32
12	SPEED3 ACCU2	JOINT	0.4	10	0	0	1.072	23.392
13	SPEED3 ACCU2	JOINT	0.4	10	0	0	1.2	24.592
14	SPEED0 ACCU2	JOINT	0.1	10	0	0	4.176	28.768
15	SPEED0 ACCU2	JOINT	0.1	10	0	0	5.04	33.808

Figure 5: The table of simulation's cycle time results (cycle time values and characteristics are for exemplification purposes only)

For the analyzed palletizing process, PC-ROSET software offers important functions like the robot's cycle time calculation, analyzing robot's position for fast application planning, collision check, changing model installation position, conveyor synchronization, etc.

Challenges appear at products like bags because it depends on the bag type and how the product in the bag is packed. There are many differences between case palletizing and bag palletizing but these two elements will affect the cost and efficiency in operation. In simulation environment the problems consist in the way the bag is picked-up from the conveyor because it can result in collision errors. Also, the bag modeling is nearly impossible in simulation environment but the problem can be solved through the use of a proper end-of-arm tooling type which works with every type of bags.

4. CONCLUSIONS

The simulation method varies from an application to another. Starting from the palletizing process, the use of simulation software offers bigger advantages then disadvantages. Even with a complementary cost for the program acquisition and a robot physical calibration, the time for a specific task and the transition between different tasks have been drastically reduced along with the use of robot simulation systems. PC-ROST simulation software provides important function and tools for workcell modeling and off-line robot simulation.

Regarding the palletizing cell design, it requires careful consideration of product shape, size, packaging and consistency. Evaluating these components, software requirements and manufacturing layout during the design phase will increase the overall palletizing efficiency. Other considerations for optimizing the palletizing cell are:

- For cell productivity, the product on the in-feed conveyor should be close to the build pallet but at the same time without interfering with the robot or with its tool;
- The software needs to remain simple and easy to learn and use;
- Bag palletizing, with its unique shapes and contents, does not lend well to column stacking; a simple solution consists in interlocking pallet patterns for a more stable pallet load;
- When building a pallet, boxes tend to be placed on the pallet where as bags tend to be dropped. Additionally, because robots work in more fluid motions cycle times are increased by this stacking method.

5. REFERENCES

[1] Bottazzi, V., Fonseca, V.: *Off-line Programming Industrial Robots Based in the Information Extracted From Neutral Files Generated by the Commercial CAD Tools*. Pro Literatur Verlag, 2006. ISBN 3-86611-286-6.

[2] Chen, D., Cheng, F. (2000): *Integration of Product and Process Development Using Rapid Prototyping and Workcell Simulation Technologies*. Journal of Industrial Technology, Volume 16, Number 1

[3] Lim, S., Yu, M., Kang, M., Han, C., (2006), *Robot Palletizing Simulation Using Heuristic Pattern Generation and Trajectory Optimization*. SICE-ICASE International Joint Conference, Busan, Korea;

[4] Olsson, M., (2002), *Simulation and execution of autonomous robot systems*. PhD thesis, Div of Robotics, Department of Mechanical Engineering, Lund University.

[5] http:// www.kukarobotics.com/ palletizing/Bag palletizing: a design guide.

[6] ***, Olsson, M., Cederberg, P., Bolmsjö, G (2006): *Integration of Simulation and Execution in Industrial Robot Systems*. Division of Robotics, Lund University.

[7] ***, Sánchez, C., A., (2004): *Robot handling application made by off-line programming robot handling application made by off-line programming.*