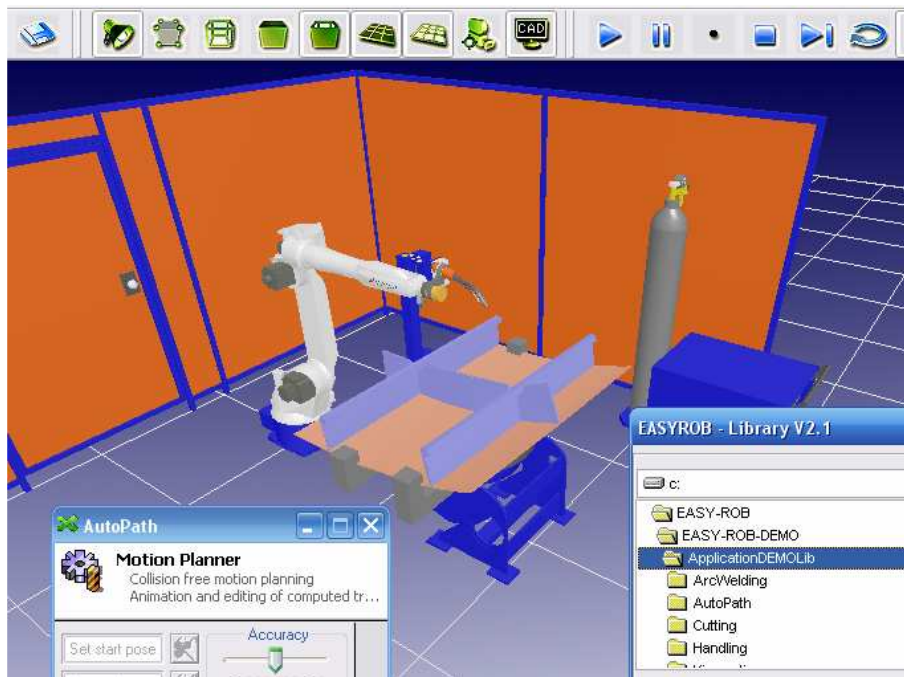


Update

EASY-ROB™ V4.307



August 2007

Version 1.01

EASY-ROB™

Table of contents

Introduction	3
General.....	4
Dual-Core - capability.....	4
Windows VISTA™ - capability.....	4
Kinematics with less than 6 independent axis.....	4
API Functions	4
New Libraries	5
ApplicationLib	5
DeviceLib.....	5
TrainLib.....	5
ERPL-/ ERCL-Examples	5
New Functions / ERCL-Commands	6
If – While - Goto.....	6
New Option	8
AutoPath™ - collision free path planning	8
Notes.....	9

EASY-ROB™ V4.307

Introduction

The version EASY-ROB™ V4.307 is the result of the previous month work and also the last interim version of V4.3.

We invested a lot of time to provide Dual-Core capability for EASY-ROB™ and we started already the first tests with EASY-ROB™ on Windows Vista™. The results are very satisfying. But all that reduced the time for development of new functions.

In spite of everything we extend the amount of ERPL-/ERCL- commands, we implemented new API-functions and we improved the numerical solution. Furthermore we removed some bugs.

We proudly present the new option AutoPath™, which has been developed in cooperation with the University of Darmstadt. AutoPath™ can generate collision free paths automatically. The calculation by RRTs (Rapidly Random Trees) can take sometimes a few minutes, but it will make the users life a lot easier. The generated solution (path) can be quite surprising – but will be always collision free. The combinations of the users' process know how and the functionality of AutoPath™ will provide an optimal result.

The current version is - as usual – just a further milestone. The list of (new) required functions is long, but we are working constantly on it. The customer specific EASY-ROB™-DLL-version and the EASY-ROB™ Simulations Kernel (used by the offline programming system Famos robotic®) need as well our attention. For a short time the kernel is available as a version for Linux.

The development will go on - the goal is to provide a tool which makes planning and simulation tasks easier for the user.

The future version V4.6 will provide finally the capability of "multi-program". Simulate more than one robot – every robot with its own program. The robots will communicate via I/O-signals. The "multi-program"-functionality requires major changes in the software structure, which has been started already.

We would like to thank our customers and users, who send suggestions and requirements for further development.

Thank you

A handwritten signature in blue ink, appearing to read "Stefan Anton".

Stefan Anton

EASY-ROB
3D Robot Simulation Tool

General

Dual-Core - capability

The development to improve the hardware is going on to fulfill the increasing requirements of the customers. But changes in the hardware architecture require as well changes in the software to avoid undesirable effects.

One of these effects is „jumping of geometries” while the graphic update in EASY-ROB™. That phenomenon appeared on PCs with Dual-Core- and Multi-Core-CPU's. The reason for the phenomenon has been the not synchronized threads while program simulation.

That was as well a topic with „EASY-ROB™ on Dual-Core“.

But the investment of many hours of the development department was worth:

1. no more „jerky“ simulations with “jumping geometries” and on top of that
2. we can provide a major improvement of the graphical performance

Windows VISTA™ - capability

Beside the changes of the hardware, the operating systems are changing as well. The question how far the current OpenGL version will be supported by VISTA™ is not answered yet. The results of the first tests with EASY-ROB™ on Windows Vista™ are very satisfying.

Kinematics with less than 6 independent axis

To solve the inverse kinematics problems for robots with less than 6 independent axis (global degenerated kinematics) EASY-ROB™ provides the capability to write an own solution (in the language “C”) in the DLL „er_kin.dll“. For the first attempt the numerical solution „Numerical Inv.Kin.+ Data“ is sufficient, but not all possible configurations are considered. For global degenerated kinematics the procedure is improved. The preferred tool axis (in most cases the Z „Approach“-Axis) coincides now with the target frame. Here is for example the rotation around Z-axis not determined. For these kinematics you select the Sub-ID = 1.

API Functions

Many new API functions have been added. Please see the header-files „./er_dvlp/er_dvlp.h“ and „./er_dvlp/er_dvlp_ext.h“.

New Libraries

The new libraries are available on the new [EASY-ROB™-CD](#) and in the customer area of the EASY-ROB™ webpage.

ApplicationLib

The examples of the ApplicationLib are applications from different areas - for instance handling, welding or painting. All these applications can be planned with EASY-ROB™.

DeviceLib

The DeviceLib contains several devices like conveyors, gripper, weld guns, tables, chairs, etc. All devices are defined and saved as ROB-Files and that means they are automatically included in the collision chain.

TrainLib

The TrainLib (including the tutorial) will support and guide the user while doing the first steps in EASY-ROB™. The user "will walk along a red line" from an empty workcell to a complete simulation.

The files are located in the customer area of the webpage of EASY-ROB:
<http://www.easy-rob.com/en/download/easy-rob/manual-examples>

ERPL-/ ERCL-Examples

ERCL (EASY-ROB Command Language) is an extension of the ERPL (EASY-ROB Programming Language) to automate nearly all user interactions inside a robot program. Examples are: Switch ON/OFF the TCP trace, change the rendering or change the simulation step size. This feature is useful to create more advanced and effective simulations.

The File „Proj_Example_ERPL.zip“ contains many explaining examples of the functionality of ERCL/ERPL. Just load the workcell and start the simulation. In the program window you follow the execution.

The file „Proj_Example_ERPL.zip“ (including documentation) is located in the download area:
<http://www.easy-rob.com/en/download/easy-rob/manual-examples>

New Functions / ERCL-Commands

If – While - Goto

The new IF, WHILE and GOTO –Statements providing more options within a robot program. The statements IF and WHILE will check a condition for „True“ and „False“. The condition is a mathematical expression which is true when greater zero.

By using an **If-statement** the robot is able to decide while executing the program which target (e.g. a conveyor) is next.

Example IF-Statement: $xx > yy$, thus the robot will move to Tag 'T_1'

```
xx=3; yy=1
IF gt(xx,yy)
    LIN T_1
ELSEIF lt(xx,yy)
    LIN T_2
ELSEIF eq(xx,yy)
    LIN T_3
ELSE
    ! enter ERPL,ERCL here
ENDIF
```

Using a **WHILE-loop** the robot will move along a path until the condition is false.

Example WHILE-Statement: The robot will move along path 'path01' three times.

```
xx=4; yy=1
WHILE gt(xx,yy)
    xx = xx-1
    ALONG PATH01 T_1 T_5
ENDWHILE
```

See examples in „./Tutorial/Proj_example_erpl/“
functions_if_while_goto_01.cel
functions_if_while_goto_02.cel
functions_if_while_goto_03.cel
functions_if_while_goto_04.cel

The **GOTO** can be used - like in other languages – to jump to another line in the program. It is up to the user to use the GOTO in a “proper way”.

```
Example:  
GOTO LABEL my_label  
...  
! skip all the lines  
...  
LABEL my_label
```

A detailed documentation of the commands and the examples is located in the customer area:
<http://www.easy-rob.com/en/download/easy-rob/manual-examples>

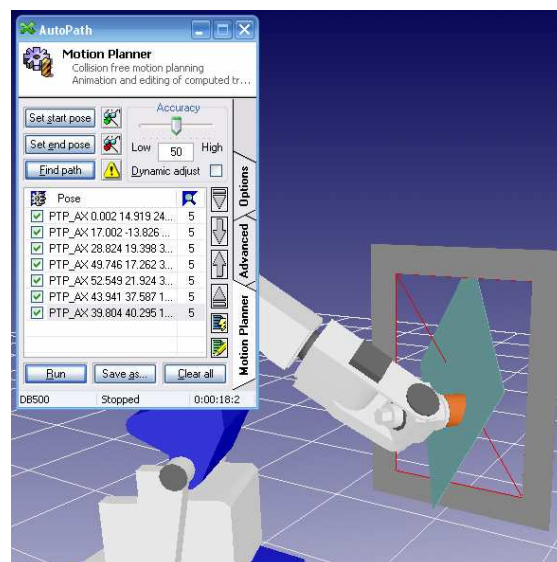
New Option

AutoPath™ - collision free path planning

To engineer a workpiece into the fixture or to move around an interfering geometry is a time-consuming task.

The EASY-ROB™ option **AutoPath™** will support the user to plan and create collision free paths.

The examples of the collision free path planning are located into the subdirectory „AutoPath“ in the „ApplicationLib“ respectively the „ApplicationDEMOLib“.



How does AutoPath™ work ?

First the user has to select the collision free start- and end position without exceeding the axis limits of the robot. By using the „Find path“-button the system will determine a collision free path. Several axis specific positions will be generated and displayed in the dialog. The positions can be copied into the teach window.

AutoPath™ was developed in cooperation with the University of Darmstadt, faculty Informatics (Prof. Dr. T. Horsch, Rudi Scheitler).

A description of the examples is located in the download area:
<http://www.easy-rob.com/en/download/easy-rob/manual-examples>



Update EASY-ROB™ V4.307

Notes