

Khepera **Toolbox**

for use with MATLAB (Release 4.2c)

Richard G. Goyette January 97
goyette@rmc.ca

1

Installation

1.1 Introduction

The Khepera mobile robot is a small, adaptable, scalable robot platform. It is a “standardized” experimental platform upon which to work. It can be interfaced to both desktop computers and to Unix workstations. Its programming interface is C, although LabView can be used on PC's. The Khepera is shown in the following figure.

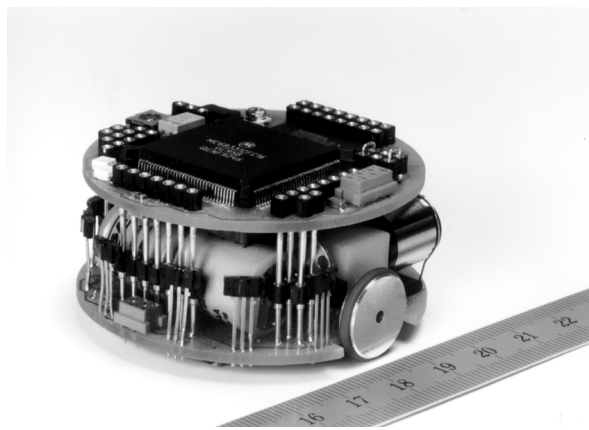


Figure 1
Khepera Miniature Robot

The **Khepera Toolbox** integrates the Khepera robot with MATLAB - a powerful, high level programming environment for linear algebra, matrix operations, and high quality graphical presentation. The **Khepera Toolbox** includes both a robot simulator and the option to control a real Khepera in real time from within the MATLAB environment. Control algorithms can be programmed and “plugged” into the simulator or they can be written to control a real robot directly. The core functionality of MATLAB has been extended by other application specific toolboxes (such as the Control and Neural Network toolbox) which can provide plug-and-play functionality for the creation of robotic control algorithms. A beneficial side effect of integrating the Khepera with MATLAB is that a high proficiency at low level programming (C, Fortran) need not be a prerequisite to investigate autonomous control algorithms. As well, since the programming environment and hardware are standard (they are commercial products), the reproduction and expansion of existing experiments can be accomplished easily by a simple file exchange.

1.2 How to obtain the toolbox

As of this writing, the toolbox may be obtained from <http://www.rmc.ca/~goyette>. Although this location will inevitably change in the next few years, it should still be possible to locate the software somewhere at the site www.rmc.ca.

1.3 Installation and System Requirements

This toolbox has been used extensively on a Pentium 100 MHZ with 16MB of RAM, running Windows 3.1 and DOS 6.2. It has been used less extensively on a 486DX 33Mhz with 8 MB of RAM running the same operating system (with appreciably slower results). The only hardware restrictions placed on operation are that the monitor be capable of supporting a resolution of 1024! 768 and that a serial port be available (preferably port 1 or 2). This package works with Matlab Version 4.2c or higher for Windows 3.1.

To install the program, create the directory **khepera** under **toolbox** in the Matlab directory. Copy the zip file to this directory and unzip it with the `-d` option. The following table describes the function of each installed directory.

Directory Organization Table	
c_files	Contains the C-MEX source for each of the serial commands
mexfiles	Contains the compiled mex functions for the source in c_files
examples	Contains sample m-files cited in this manual
matfiles	Top level contains <code>defmap.mat</code> , the default playpen map
matfiles\scanned	Should be used to hold MAT-files created by the SCAN button in the Playpen Designer and should be accessed when using the OPEN DEFAULT button in the RoboRunner window
matfiles\notscan	Should be used to hold MAT-files created by using the SAVE button in the Playpen Designer. Should be accessed when using the LOAD button in the Playpen Designer
m_files	Master directory for all m-files and functions
m_files\ctrl_alg	Used to store user written Khepera control algorithms
m_files\iface	Master directory for Khepera Toolbox functions. Top level contains <code>template.m</code> file for function integration
m_files\iface\khepgui	Contains all the graphical user interface functions

Directory Organization Table	
m_files\iface\khepplot	Contains all printing related functions
m_files\iface\reakkhep	Contains all functions devoted to running a real Khepera
m_files\iface\simkhep	Contains all functions devoted to running a Khepera simulation
m_files\iface\util_lib	Should be used to deposit application specific utilities or functions which may come in handy to someone else

Table 1
Khepera Directory Organization

Finally, using a text editor, open the file `modmatrc.txt` and copy the contents to the `matlabrc.m` file so that the new directories will be recognized by Matlab at startup.

1.4 Software Capabilities and Limitations

The following list describes the software capabilities and limitations of this Toolbox. You may wish to refer to this list after you have read the rest of this manual.

- The following Khepera serial commands are available for controlling a real Khepera from the Matlab command prompt:
 - ! Configure (A) implemented as **con**.
 - ! Set a position to be reached (C) implemented as **skp**.
 - ! Set speed (D) implemented as **sms**.
 - ! Read speed (E) implemented as **rms**.
 - ! Configure the position PID controller (F) implemented as **cpid**.
 - ! Set position to the position counter (G) implemented as **spc**.
 - ! Read position (H) implemented as **rpc**.
 - ! Read A/D input (I) implemented as **rad**.
 - ! Configure the speed profile controller (J) implemented as **cspc**.
 - ! Read the status of the motion controller (K) implemented as **rmc**.
 - ! Change LED state (L) implemented as **cls**.
 - ! Read proximity sensors (N) implemented as **rps**.
 - ! Read ambient light sensors (O) implemented as **rls**.

Serial Commands Not Implemented:

 - ! Read software version (B).
 - ! Send a message to an additional module (T).
 - ! Read a byte on the extension bus (R).
 - ! Write a byte on the extension bus (W).

- The following serial commands are available for controlling a *simulated* Khepera from the Matlab command prompt:

! Set a position to be reached (C) implemented as **sim_skp**.
 ! Set speed (D) implemented as **sim_sms**.
 ! Read speed (E) implemented as **sim_rms**.
 ! Set position to the position counter (G) implemented as **sim_spc**.
 ! Read position (H) implemented as **sim_rpc**.
 ! Read the status of the motion controller (K) implemented as **sim_rmc**.
 ! Read proximity sensors (N) implemented as **sim_rps**.
 ! Read ambient light sensors (O) implemented as **sim_rls**.

Serial Commands Not Implemented:

! Configure (A).
 ! Configure the position PID controller (F).
 ! Read A/D input (I).
 ! Configure the speed profile controller (J).
 ! Change LED state (L).
 ! Read software version (B).
 ! Send a message to an additional module (T).
 ! Read a byte on the extension bus (R).
 ! Write a byte on the extension bus (W).

- The serial commands for controlling a real Khepera from the Matlab prompt were designed using BIOS Int14 calls and have been tested on serial ports 1 and 2. Serial ports 3 and 4 are supported in the code, but communications have not been attempted using these port addresses.
- The serial commands for controlling a real Khepera from the Matlab prompt incorporate minimal error checking. Specifically, except for the port identification, the only error checking performed is to determine that the correct number of arguments has been provided. The port identification argument is the only one on which any range checking is done (the number must be in the range [1 4]). Range verification is not done on any other argument so caution should be exercised.
- When placing a simulated Khepera in the **Virtual Playpen**, no error checking is done to verify the robot patch object is not placed on another object or obstacle. Additionally, although the simulated Khepera can “see” a lamp, it cannot detect it with the proximity sensors. Thus, it is possible for the Khepera to run over light sources.
- The simulation environment currently only supports the operation of one robot. Any

number of lamps can be placed.

- When a playpen environment is created or customized using the Virtual Playpen Designer, it is flipped on its X axis (upside down) when it is scanned.
- The range of motion in the simulation environment is restricted to forwards, backwards, rotate left, and rotate right. That is, the simulated motors do not operate at different speeds (except during rotation in which case the magnitude is the same but the sign is different).
- The simulation environment does not implement the shadowing effect of walls and other obstacles. Lamps can be seen by the Khepera no matter where or on top of what they are placed.

1.5 License Agreement and Standard Disclaimers

The Khepera Toolbox is freeware - public domain software. Permission is hereby granted to copy this package for free distribution. Commercial use of this Toolbox in any way is forbidden. The author's name and copyright notice must be provided in any copy. If you cite this Toolbox in any academic book or paper, please use the following reference:

Captain Rich Goyette. *Khepera Toolbox for Matlab*, Version 1.0,
Free, available at <http://www.rmc.ca/~goyette>

The author is not responsible for any hardware or software damage (however unlikely) caused by the uploading, downloading, backing up, reading, editing, general use, or any other operation remotely connected or unconnected with this software. The author cannot be held responsible for any degeneration of eyesight, carpal tunnel syndrome, obesity, loss of friends, break-up of marriage, or testicular or cervical cancer developed while using this software.

1.6 References

P.I. Corke, "A Robotics Toolbox for Matlab," *IEEE Robotics and Automation Magazine*, vol. 3, pp. 24-32, Mar 1996.

O. Michel, "Khepera Simulator Package, V 2.0," University of Nice Sophia-Antipolis, <http://wwwi3s.unice.fr/~om/khep-sim.html>.

P. Marchand, *Graphics and GUIs with MATLAB*, New York: CRC Press, 1996.

D. Hanselman, B. Littlefield, *The Student Edition of Matlab, Version 4 User's Guide*, Englewood Cliffs, NJ: Prentice Hall, 1995.

I. Harvey, P. Husbands, & D. Cliff, "Issues in evolutionary robotics," in *From Animals to Animats 2*, J.-A. Meyer, H. Roitblat, and S. Wilson, (eds.) Cambridge, MA: MIT Press, pp. 364-373, 1993.

Matlab Reference Guide, Version 4.2c.

Matlab External Interface Guide, Version 4.2c.

Watcom C++ Online User's Manuals, Version 10.0.

Khepera Host Machine Functions located at <ftp://lamiftp.epfl.ch/khepera/softHost/> and, in particular, those written by Jeff Elman and Daniele Denaro.

1.7 Acknowledgements

I would like to acknowledge THE NET and everyone who put their two cents worth on it. Without THE NET, I would probably have done an antenna experiment for my thesis. This was much more interesting.

