Índex

Apèndix B

Codi en C/C++

B.8 Func_geominterface_2.h

func_geominterface_2.h

```
/*********************************************************************
 *    This program is an adaptation of Jan Rosell and Alexander Prez work.  *
*********************************************************************/

/*********************************************************************
 *    Copyright (C) 2008 by Jan Rosell and Alexander Prez                  *
 *    jan.rosell@upc.edu - alexander.perez@upc.edu                         *
 *                                                                         *
 *    This program is free software; you can redistribute it and/or modify  *
 *    it under the terms of the GNU General Public License as published by  *
 *    the Free Software Foundation; either version 2 of the License, or     *
 *    (at your option) any later version.                                   *
 *                                                                         *
 *    This program is distributed in the hope that it will be useful,       *
 *    but WITHOUT ANY WARRANTY; without even the implied warranty of        *
 *    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the         *
 *    GNU General Public License for more details.                          *
 *                                                                         *
 *    You should have received a copy of the GNU General Public License     *
 *    along with this program; if not, write to the                         *
 *    Free Software Foundation, Inc.,                                       *
 *    59 Temple Place - Suite 330, Boston, MA  02111-1307, USA.             *
 *********************************************************************/

#ifndef GEOMINTERFACE_2_H
#define GEOMINTERFACE_2_H

#include "ui_geominterfaceUI_P8.h"
#include <QWidget>
```

```
#include <QObject>

#include <Inventor/Qt/SoQt.h>
#include <Inventor/nodes/SoSeparator.h>


#include <boost/numeric/ublas/vector.hpp>
#include <boost/numeric/ublas/matrix.hpp>



#include "geometry_classes/geomobject.h"

class geomobject;

class geominterface_2 : public QWidget, private Ui::geominterfaceUI_P8
{
        Q_OBJECT

        public:
                //!Non-void constructor
                geominterface_2(QWidget *parent, std::vector<geomobject *> vectptr,
                 const boost::numeric::ublas::matrix<double> config_list);

                geominterface_2(QWidget *parent, std::vector<geomobject *> vectptr,
                 const boost::numeric::ublas::matrix<double> config_list,
                 const boost::numeric::ublas::matrix<double> pod);

                //!Destructor
                ~geominterface_2();

        private:
                //!vector of pointers to the geomobjects of the scene
                std::vector<geomobject *> vectgeomobjects;
                //!Pointer to inventor scene
                SoSeparator* IVScene;
                //!timer for the nterpolation of the motion towards the initial
                // configuration
                QTimer *clearTimer;

                boost::numeric::ublas::matrix<double> config_list_p;
                boost::numeric::ublas::matrix<double> pod_p;

        private slots:
                //!Slot to clear the text box when pushing the clear button
                void clear();
                //!Slot to change the position of the manipulated object
                void changePosition(int t);
                //!Slot to change the orientation of the manipulated object
                void changeOrientationRX(int t);
                //!Slot to change the orientation of the manipulated object
                void changeOrientationRY(int t);
                //!Slot to change the orientation of the manipulated object
                void changeOrientationRZ(int t);
                //!Slot to test collisions
```

```
                void testCollisions();
                //!Slot to test distance
                void testDistance();
                //!Slot to interpolate (and show) the motion towards the initial
                // configuration
                void cleartimeout();

                void numSample(int n);

};


#endif
```

func_geominterface_2.cpp

```
/*************************************************************************
 *   This program is an adaptation of Jan Rosell and Alexander Prez work.  *
*************************************************************************/

/*************************************************************************
 *   Copyright (C) 2008 by Jan Rosell and Alexander Perez                 *
 *   jan.rosell@upc.edu - alexander.perez@upc.edu                         *
*                Universitat Politecnica de Catalunya                    *
*                        Barcelona, Spain                                *
*                                                                        *
*                          CLASS  cube                                   *
*                                                                        *
*                           Jan Rosell                                   *
*                            mai 2008                                    *
*                                                                        *
*                                                                        *
*************************************************************************/

/*************************************************************************
 *   Copyright (C) 2008 by Jan Rosell                                     *
 *   jan.rosell@upc.edu                                                   *
 *                                                                        *
 *   This program is free software; you can redistribute it and/or modify *
 *   it under the terms of the GNU General Public License as published by *
 *   the Free Software Foundation; either version 2 of the License, or    *
 *   (at your option) any later version.                                  *
 *                                                                        *
 *   This program is distributed in the hope that it will be useful,      *
 *   but WITHOUT ANY WARRANTY; without even the implied warranty of       *
 *   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the        *
 *   GNU General Public License for more details.                         *
 *                                                                        *
 *   You should have received a copy of the GNU General Public License    *
 *   along with this program; if not, write to the                        *
 *   Free Software Foundation, Inc.,                                      *
 *   59 Temple Place - Suite 330, Boston, MA  02111-1307, USA.            *
 *************************************************************************/
```

```
#ifndef GEOMINTERFACE_2_CPP
#define GEOMINTERFACE_2_CPP

#include <Inventor/nodes/SoCube.h>
#include <Inventor/Qt/SoQt.h>
#include <Inventor/Qt/viewers/SoQtExaminerViewer.h>

#include <sstream>
#include <iostream>
#include "geominterface_2.h"


#include <boost/numeric/ublas/vector.hpp>
#include <boost/numeric/ublas/matrix.hpp>
#include <boost/numeric/ublas/io.hpp>


#include "prog/func_graf_coll_ma.h"


#include "collision_classes/pqpobject.h"


#include "QApplication"
#include "Inventor/Qt/SoQt.h"
#include "QFileInfo"
#include "QDir"
#include "QString"
#include "QtGui"


#include "test_classes/strucparse.h"


//using namespace std;
// including <QtGui> saves us to include every class user, <QString>, <QFileDialog>,...

//The first element of the vector of objects is the one defined as movable
int movingObject=0;
double worldSize = 5.0;

geominterface_2::geominterface_2(QWidget *parent, std::vector<geomobject *> vectptr,
 const boost::numeric::ublas::matrix<double> config_list)
{
        vectgeomobjects = vectptr;
        config_list_p = config_list;
        setupUi(this); // this sets up GUI


        // signals/slots mechanism in action
        connect( ResetButton, SIGNAL( clicked() ), this, SLOT( clear() ) );
        connect( DistanceButton, SIGNAL( clicked() ), this, SLOT( testDistance() ) );
        connect( CollideButton, SIGNAL( clicked() ), this, SLOT( testCollisions() ) );
        //Position sliders
        connect( horizontalSliderTX, SIGNAL( valueChanged (int) ), this,
```

```
        SLOT( changePosition(int) ) );
    connect( horizontalSliderTY, SIGNAL( valueChanged (int) ), this,
        SLOT( changePosition(int) ) );
    connect( horizontalSliderTZ, SIGNAL( valueChanged (int) ), this,
        SLOT( changePosition(int) ) );
    //Orientation sliders
    connect( horizontalSliderRX, SIGNAL( valueChanged (int) ), this,
        SLOT( changeOrientationRX(int) ) );
    connect( horizontalSliderRY, SIGNAL( valueChanged (int) ), this,
        SLOT( changeOrientationRY(int) ) );
    connect( horizontalSliderRZ, SIGNAL( valueChanged (int) ), this,
        SLOT( changeOrientationRZ(int) ) );

    connect( spinBoxConfig, SIGNAL( valueChanged (int) ), this,
        SLOT( numSample(int) ) );


    IVScene = new SoSeparator();
    IVScene->ref();
    for(unsigned int i=0; i<vectgeomobjects.size(); i++)
            IVScene->addChild(vectgeomobjects[i]->getInventorModel());


SoQtExaminerViewer *sceneviewer = new SoQtExaminerViewer(widgetCoin3D);
sceneviewer->setSceneGraph(IVScene);
sceneviewer->show();

    clearTimer = new QTimer( this );
    connect( clearTimer, SIGNAL(timeout()), SLOT(cleartimeout()) );

    numSample(0);
}



geominterface_2::geominterface_2(QWidget *parent, std::vector<geomobject *> vectptr,
 const boost::numeric::ublas::matrix<double> config_list,
 const boost::numeric::ublas::matrix<double> pod)
{
    vectgeomobjects = vectptr;
    config_list_p = config_list;
    pod_p = pod;
    setupUi(this); // this sets up GUI


    // signals/slots mechanism in action
    connect( ResetButton, SIGNAL( clicked() ), this, SLOT( clear() ) );
    connect( DistanceButton, SIGNAL( clicked() ), this, SLOT( testDistance() ) );
    connect( CollideButton, SIGNAL( clicked() ), this, SLOT( testCollisions() ) );
    //Position sliders
    connect( horizontalSliderTX, SIGNAL( valueChanged (int) ), this,
        SLOT( changePosition(int) ) );
    connect( horizontalSliderTY, SIGNAL( valueChanged (int) ), this,
        SLOT( changePosition(int) ) );
    connect( horizontalSliderTZ, SIGNAL( valueChanged (int) ), this,
```

```
        SLOT( changePosition(int) ) );
        //Orientation sliders
        connect( horizontalSliderRX, SIGNAL( valueChanged (int) ), this,
         SLOT( changeOrientationRX(int) ) );
        connect( horizontalSliderRY, SIGNAL( valueChanged (int) ), this,
         SLOT( changeOrientationRY(int) ) );
        connect( horizontalSliderRZ, SIGNAL( valueChanged (int) ), this,
         SLOT( changeOrientationRZ(int) ) );

        connect( spinBoxConfig, SIGNAL( valueChanged (int) ),
         this, SLOT( numSample(int) ) );


        IVScene = new SoSeparator();
        IVScene->ref();
        for(unsigned int i=0; i<vectgeomobjects.size(); i++)
                IVScene->addChild(vectgeomobjects[i]->getInventorModel());

    SoQtExaminerViewer *sceneviewer = new SoQtExaminerViewer(widgetCoin3D);
    sceneviewer->setSceneGraph(IVScene);
    sceneviewer->show();

        clearTimer = new QTimer( this );
        connect( clearTimer, SIGNAL(timeout()), SLOT(cleartimeout()) );

        numSample(0);
}



void geominterface_2::numSample(int n)
{
        std::ostringstream st;

        std::vector<geomobject*> objects;
        objects = vectgeomobjects;
        int n_fil = config_list_p.size1();

        int i;

        boost::numeric::ublas::vector<double> config_mai(30);
        int k;
        if (n >= n_fil)         k = n_fil-1;
        else                    k = n;


        for (i = 0; i < 30; i++)        config_mai(i) = config_list_p(k,i);

                if ( (pod_p.size1() == 3) && (pod_p.size2() == 8) ) {
                set_config_objects_pod(config_mai, pod_p, objects);
                st << "config + pod - NUM_SAMPLE " << k << " of " << n_fil-1 << endl;
        }
        else    {
                set_config_objects(config_mai, objects);
                st << "config - NUM_SAMPLE " << k << " of " << n_fil-1 << endl;
```

```
        }


        textBrowserOutputText->setText(st.str().c_str());
}


void geominterface_2::testCollisions()
{
    std::ostringstream st;

        int n;
        st << "Test Collisions: "<< endl;
        for(int i=0; i<vectgeomobjects.size();i++)
        {
                n = vectgeomobjects[i]->collidemodel->getAllCollisions();
                st << "object " << i << " collides with "<< n << " object(s)" << endl;
        }
        //int n = vectgeomobjects[movingObject]->collidemodel->getAllCollisions();
        //if(n) st << "Test Collisions: \nColliding with " << n << " object(s)" << endl;
        //else  st << "Test Collisions: \nFree configuration" << endl;
        textBrowserOutputText->setText(st.str().c_str());
        //vectgeomobjects[1]->collidemodel->getAllCollisions();
}



void geominterface_2::testDistance()
{
        vector<double> dist;
        vectgeomobjects[movingObject]->collidemodel->getAllDistances(&dist);

    std::ostringstream st;
    st << "Test Distances: " << endl;
        //cout << "p4 " << dist.size()<< endl;
        for(unsigned int i=0; i<dist.size();i++)
        {
                st << dist[i] << "   ";
        }
        st << endl;

        textBrowserOutputText->setText(st.str().c_str());
}


void geominterface_2::clear()
{
        //Home-made interpolation using a timer
        //internaltimer is connected to slot function cleartimeout();
        clearTimer->start( 50 );

        //A possible alternative is to use the
        //inventor interpolation engine (SoInterpolateVec3f)
        //but we lose the control of the exact configuration
```

```
        //of the interpolated path:
        /*
        SbVec3f ini(p[0],p[1],p[2]);
        SbVec3f goal(pini[0],pini[1],pini[2]);
        SoInterpolateVec3f *interp = new SoInterpolateVec3f;
        interp->input0.setValue ( ini  );
        interp->input1.setValue ( goal );
        vectgeomobjects[movingObject]->ivTransf->translation.connectFrom(&interp->output);
        SoOneShot *oneShot = new SoOneShot;
        interp->alpha.connectFrom(&oneShot->ramp);
        oneShot->trigger.touch();
        */
}

void geominterface_2::cleartimeout()
{
        static int i=0;
        static mt::Point3 stept, pini;
        static mt::Rotation qini,q;

        static int maxsteps;
        double sizesteps = 0.1;

        if(i==0)
        {
                mt::Point3 pcurrent=vectgeomobjects[movingObject]->getTranslation();
                mt::Rotation qcurrent=vectgeomobjects[movingObject]->getRotation();
                double dist = qcurrent.distance(qini);
                double dist2 = pcurrent.distance(pini);
                if(dist2>dist) dist = dist2;

                //the 2 is necessary to always reduce the distance...
                maxsteps = (dist/sizesteps)+2;

                //translation
                mt::Point3 p, delta;
                pini = vectgeomobjects[movingObject]->getInitialTranslation();
                p    = vectgeomobjects[movingObject]->getTranslation();
                delta = pini - p;
                stept = delta/maxsteps;
                //rotation
                qini = vectgeomobjects[movingObject]->getInitialRotation();
                q    = vectgeomobjects[movingObject]->getRotation();
        }
        vectgeomobjects[movingObject]->translate(stept);
        vectgeomobjects[movingObject]->setRotation( q.slerp(qini,(double)i/maxsteps) );
        i++;

        char st[80];
        mt::Point3 pcurrent=vectgeomobjects[movingObject]->getTranslation();
        mt::Rotation qcurrent=vectgeomobjects[movingObject]->getRotation();
        double rdist = qcurrent.distance(qini);
        double tdist = pcurrent.distance(pini);
        sprintf(st, "Position (%.1f %.1f %.1f)\n Rot dist = %f\n Trans dist = %f\n",
         pcurrent[0],pcurrent[1],pcurrent[2],rdist,tdist);
```

```
        textBrowserOutputText->setText(st);

        if(vectgeomobjects[movingObject]->collidemodel->getAllCollisions())
        {
                textBrowserOutputText->setText("Colliding...");
                i=0;
                clearTimer->stop();
        }
        else if(i==maxsteps)
        {
                //textBrowserOutputText->setText("End Reset motion");
                i=0;
                clearTimer->stop();
        }
}

void geominterface_2::changePosition(int t)
{
  static float iniX=0.0;
  static float iniY=0.0;
  static float iniZ=0.0;

  float x = (float)horizontalSliderTX->value() - iniX;
  float y = (float)horizontalSliderTY->value() - iniY;
  float z = (float)horizontalSliderTZ->value() - iniZ;

  iniX = (float)horizontalSliderTX->value();
  iniY = (float)horizontalSliderTY->value();
  iniZ = (float)horizontalSliderTZ->value();

  mt::Point3 p(x,y,z);
  p = p*worldSize/1000.0; //1000 is defined by the slider
  vectgeomobjects[movingObject]->translate(p);

  char st[80];
  p=vectgeomobjects[movingObject]->getTranslation();
  sprintf(st, "Position (%.1f %.1f %.1f)\n",p[0],p[1],p[2]);

  textBrowserOutputText->setText(st);
}


void geominterface_2::changeOrientationRX(int t)
{
  static float ini=0.0;
  float rx = (float)horizontalSliderRX->value() - ini;
  ini = (float)horizontalSliderRX->value();


  mt::Unit3 axis(1.0,0.0,0.0);
  mt::Scalar angle = rx*TWO_PI/1000;
  mt::Rotation r(axis,angle);
  vectgeomobjects[movingObject]->rotate(r);

  char st[80];
```

```
  r=vectgeomobjects[movingObject]->getRotation();
  mt::Matrix3x3 m = r.getMatrix();

  sprintf(st,
          "Rotation:\t(%.1f,%.1f,%.1f)\n\t(%.1f,%.1f,%.1f)\n\t(%.1f,%.1f,%.1f)\n",
          m[0][0], m[0][1], m[0][2],
          m[1][0], m[1][1], m[1][2],
          m[2][0], m[2][1], m[2][2]);

  textBrowserOutputText->setText(st);
}

void geominterface_2::changeOrientationRY(int t)
{
  static float ini=0.0;
  float ry = (float)horizontalSliderRY->value() - ini;
  ini = (float)horizontalSliderRY->value();


  mt::Unit3 axis(0.0,1.0,0.0);
  mt::Scalar angle = ry*TWO_PI/1000;
  mt::Rotation r(axis,angle);
  vectgeomobjects[movingObject]->rotate(r);

  char st[80];
  r=vectgeomobjects[movingObject]->getRotation();
  mt::Matrix3x3 m = r.getMatrix();

  sprintf(st,
          "Rotation:\t(%.1f,%.1f,%.1f)\n\t(%.1f,%.1f,%.1f)\n\t(%.1f,%.1f,%.1f)\n",
          m[0][0], m[0][1], m[0][2],
          m[1][0], m[1][1], m[1][2],
          m[2][0], m[2][1], m[2][2]);

  textBrowserOutputText->setText(st);
}

void geominterface_2::changeOrientationRZ(int t)
{
  static float ini=0.0;
  float rz = (float)horizontalSliderRZ->value() - ini;
  ini = (float)horizontalSliderRZ->value();

  mt::Unit3 axis(0.0,0.0,1.0);
  mt::Scalar angle = rz*TWO_PI/1000;
  mt::Rotation r(axis,angle);
  vectgeomobjects[movingObject]->rotate(r);

  char st[80];
  r=vectgeomobjects[movingObject]->getRotation();
  mt::Matrix3x3 m = r.getMatrix();

  sprintf(st,
          "Rotation:\t(%.1f,%.1f,%.1f)\n\t(%.1f,%.1f,%.1f)\n\t(%.1f,%.1f,%.1f)\n",
          m[0][0], m[0][1], m[0][2],
```

```
        m[1][0], m[1][1], m[1][2],
        m[2][0], m[2][1], m[2][2]);

  textBrowserOutputText->setText(st);
}


geominterface_2::~geominterface_2()
{
        IVScene->unref();
}

#endif  // GEOMINTERFACE_2_CPP //
```

Apèndix C

La cinemàtica directa: Matlab

A continuació es presenten els arxius emprats per a obtenir les expressions explícites de posicions, eixos i orientacions emprats en el projecte, obtinguts mitjançant cinemàtica directa.

Es mostren quatre arxius, que permeten obtenir aquells vectors que es necessitin en funció del dit que es desitji. Per a cada un dels arxius, es carrega a la variable *JBA* el vector, pertanyent a una matriu de transformació concreta, que es vol obtenir, i el resultat és aquest vector en funció de les variables que representen les articulacions que afecten al mateix.

Anular

```
clear all;
close all;


% Declaracio de variables
%Pos Inicial
Xb=sym('Xb','real');
Yb=sym('Yb','real');
Zb=sym('Zb','real');
ALPHAb=sym('ALPHAb','real');
BETAb=sym('BETAb','real');
GAMMAb=sym('GAMMAb','real');
%Anular
TET2A = sym('TET2A','real');
TET3A = sym('TET3A','real');
TET4A = sym('TET4A','real');
TET5IIA = sym('TET5IIA','real');
TET6IIA = sym('TET6IIA','real');
TET7A = sym('TET7A','real');
```

```
% Matrius de transformacio homogenia
%Canell
T1r0=[1 0 0 Xb;0 1 0 Yb;0 0 1 Zb;0 0 0 1];
T2r1=[1 0 0 0; 0 cos(ALPHAb) -sin(ALPHAb) 0; 0 sin(ALPHAb) cos(ALPHAb) 0; 0 0 0 1];
T3r2=[cos(BETAb) 0 -sin(BETAb) 0;0 1 0 0;sin(BETAb) 0 cos(BETAb) 0;0 0 0 1];
T4r3=[cos(GAMMAb) -sin(GAMMAb) 0 0;sin(GAMMAb) cos(GAMMAb) 0 0;0 0 1 0;0 0 0 1];

%Anular
T5r4a=[0 -1 0 0;1 0 0 0;0 0 1 197.55;0 0 0 1];
T6r5a=[cos(TET2A) -sin(TET2A) 0 -67;0 0 -1 -9.5;sin(TET2A) cos(TET2A) 0 0;0 0 0 1];
T7r6a=[cos(TET3A) -sin(TET3A) 0 0;0 0 -1 0;sin(TET3A) cos(TET3A) 0 0;0 0 0 1];
T8r7a=[cos(TET4A) -sin(TET4A) 0 76.66;sin(TET4A) cos(TET4A) 0 0;0 0 1 0;0 0 0 1];
T9r8a=[cos(TET5IIA) -sin(TET5IIA) 0 56;sin(TET5IIA) cos(TET5IIA) 0 0;0 0 1 0;
       0 0 0 1];
T10r9a=[cos(TET6IIA) -sin(TET6IIA) 0 33.62;sin(TET6IIA) cos(TET6IIA) 0 0;0 0 1 0;
       0 0 0 1];
T11r10a=[cos(TET7A) -sin(TET7A) 0 0;0 0 1 0;-sin(TET7A) -cos(TET7A) 0 0;0 0 0 1];
T12r11a=[1 0 0 20;0 1 0 0;0 0 1 0;0 0 0 1];



% CAP AL JACOBIA ————————————————————————————————

% Blocs de frames
%Pos Inicial
T1e0=T1r0;
T2e0=T1e0*T2r1;
T3e0=T2e0*T3r2;
T4e0=T3e0*T4r3;
%Dit
T5e0pa=T4e0*T5r4a;
T6e0pa=T5e0pa*T6r5a;
T7e0pa=T6e0pa*T7r6a;
T8e0pa=T7e0pa*T8r7a;
T9e0pa=T8e0pa*T9r8a;
T10e0pa=T9e0pa*T10r9a;
T11e0pa=T10e0pa*T11r10a;
T12e0pa=T11e0pa*T12r11a;


%Pos Inicial
sA=sym('sA','real');
cA=sym('cA','real');
sB=sym('sB','real');
cB=sym('cB','real');
sG=sym('sG','real');
cG=sym('cG','real');
%Anular
s2A = sym('s2A','real');
c2A = sym('c2A','real');
s3A = sym('s3A','real');
c3A = sym('c3A','real');
s4A = sym('s4A','real');
c4A = sym('c4A','real');
```

```
s5A = sym('s5A','real');
c5A = sym('c5A','real');
s6A = sym('s6A','real');
c6A = sym('c6A','real');
s7A = sym('s7A','real');
c7A = sym('c7A','real');


JBA = T12e0pa(:,1);

% SUBSTITUCIONS
% Canell
JBA = subs(JBA,sin(ALPHAb),sA);
JBA = subs(JBA,cos(ALPHAb),cA);
JBA = subs(JBA,sin(BETAb),sB);
JBA = subs(JBA,cos(BETAb),cB);
JBA = subs(JBA,sin(GAMMAb),sG);
JBA = subs(JBA,cos(GAMMAb),cG);
% Anular
JBA = subs(JBA,sin(TET2A),s2A);
JBA = subs(JBA,cos(TET2A),c2A);
JBA = subs(JBA,sin(TET3A),s3A);
JBA = subs(JBA,cos(TET3A),c3A);
JBA = subs(JBA,sin(TET4A),s4A);
JBA = subs(JBA,cos(TET4A),c4A);
JBA = subs(JBA,sin(TET5IIA),s5A);
JBA = subs(JBA,cos(TET5IIA),c5A);
JBA = subs(JBA,sin(TET6IIA),s6A);
JBA = subs(JBA,cos(TET6IIA),c6A);
JBA = subs(JBA,sin(TET7A),s7A);
JBA = subs(JBA,cos(TET7A),c7A);


JBA = vpa(JBA);
```

Mitjer

```
clear all;
close all;


% Declaracio de variables
%Pos Inicial
Xb=sym('Xb','real');
Yb=sym('Yb','real');
Zb=sym('Zb','real');
ALPHAb=sym('ALPHAb','real');
BETAb=sym('BETAb','real');
GAMMAb=sym('GAMMAb','real');
%Cor o mitjer
TET2C = sym('TET2C','real');
TET3C = sym('TET3C','real');
TET4C = sym('TET4C','real');
TET5IIC = sym('TET5IIC','real');
```

```
TET6IIC = sym('TET6IIC','real');
TET7C = sym('TET7C','real');



% Matrius de transformacio homogenia
%Canell
T1r0=[1 0 0 Xb;0 1 0 Yb;0 0 1 Zb;0 0 0 1];
T2r1=[1 0 0 0; 0 cos(ALPHAb) -sin(ALPHAb) 0; 0 sin(ALPHAb) cos(ALPHAb) 0; 0 0 0 1];
T3r2=[cos(BETAb) 0 -sin(BETAb) 0;0 1 0 0;sin(BETAb) 0 cos(BETAb) 0;0 0 0 1];
T4r3=[cos(GAMMAb) -sin(GAMMAb) 0 0;sin(GAMMAb) cos(GAMMAb) 0 0;0 0 1 0;0 0 0 1];

%Cor o mitjer
T5r4a=[0 -1 0 0;1 0 0 0;0 0 1 197.55;0 0 0 1];
T6r5a=[cos(TET2C) -sin(TET2C) 0 0;0 0 -1 -9.5;sin(TET2C) cos(TET2C) 0 0;0 0 0 1];
T7r6a=[cos(TET3C) -sin(TET3C) 0 0;0 0 -1 0;sin(TET3C) cos(TET3C) 0 0;0 0 0 1];
T8r7a=[cos(TET4C) -sin(TET4C) 0 76.66;sin(TET4C) cos(TET4C) 0 0;0 0 1 0;0 0 0 1];
T9r8a=[cos(TET5IIC) -sin(TET5IIC) 0 56;sin(TET5IIC) cos(TET5IIC) 0 0;0 0 1 0;
       0 0 0 1];
T10r9a=[cos(TET6IIC) -sin(TET6IIC) 0 33.62;sin(TET6IIC) cos(TET6IIC) 0 0;0 0 1 0;
       0 0 0 1];
T11r10a=[cos(TET7C) -sin(TET7C) 0 0;0 0 1 0;-sin(TET7C) -cos(TET7C) 0 0;0 0 0 1];
T12r11a=[1 0 0 20;0 1 0 0;0 0 1 0;0 0 0 1];


% CAP AL JACOBIA ----------------------------------------------

% Blocs de frames
%Pos Inicial
T1e0=T1r0;
T2e0=T1e0*T2r1;
T3e0=T2e0*T3r2;
T4e0=T3e0*T4r3;
%Dit
T5e0pa=T4e0*T5r4a;
T6e0pa=T5e0pa*T6r5a;
T7e0pa=T6e0pa*T7r6a;
T8e0pa=T7e0pa*T8r7a;
T9e0pa=T8e0pa*T9r8a;
T10e0pa=T9e0pa*T10r9a;
T11e0pa=T10e0pa*T11r10a;
T12e0pa=T11e0pa*T12r11a;


%Pos Inicial
sA=sym('sA','real');
cA=sym('cA','real');
sB=sym('sB','real');
cB=sym('cB','real');
sG=sym('sG','real');
cG=sym('cG','real');
%Cor o mitjer
s2C = sym('s2C','real');
c2C = sym('c2C','real');
s3C = sym('s3C','real');
```

```
c3C = sym('c3C','real');
s4C = sym('s4C','real');
c4C = sym('c4C','real');
s5C = sym('s5C','real');
c5C = sym('c5C','real');
s6C = sym('s6C','real');
c6C = sym('c6C','real');
s7C = sym('s7C','real');
c7C = sym('c7C','real');


JBA = T12e0pa(:,1);

% SUBSTITUCIONS
% Canell
JBA = subs(JBA,sin(ALPHAb),sA);
JBA = subs(JBA,cos(ALPHAb),cA);
JBA = subs(JBA,sin(BETAb),sB);
JBA = subs(JBA,cos(BETAb),cB);
JBA = subs(JBA,sin(GAMMAb),sG);
JBA = subs(JBA,cos(GAMMAb),cG);
% Cor o mitjer
JBA = subs(JBA,sin(TET2C),s2C);
JBA = subs(JBA,cos(TET2C),c2C);
JBA = subs(JBA,sin(TET3C),s3C);
JBA = subs(JBA,cos(TET3C),c3C);
JBA = subs(JBA,sin(TET4C),s4C);
JBA = subs(JBA,cos(TET4C),c4C);
JBA = subs(JBA,sin(TET5IIC),s5C);
JBA = subs(JBA,cos(TET5IIC),c5C);
JBA = subs(JBA,sin(TET6IIC),s6C);
JBA = subs(JBA,cos(TET6IIC),c6C);
JBA = subs(JBA,sin(TET7C),s7C);
JBA = subs(JBA,cos(TET7C),c7C);


JBA = vpa(JBA);
```

Índex

```
clear all;
close all;


% Declaracio de variables
%Pos Inicial
Xb=sym('Xb','real');
Yb=sym('Yb','real');
Zb=sym('Zb','real');
ALPHAb=sym('ALPHAb','real');
BETAb=sym('BETAb','real');
GAMMAb=sym('GAMMAb','real');
%Index
TET2i = sym('TET2i','real');
```

```
TET3i = sym('TET3i','real');
TET4i = sym('TET4i','real');
TET5IIi = sym('TET5IIi','real');
TET6IIi = sym('TET6IIi','real');
TET7i = sym('TET7i','real');



% Matrius de transformacio homogenia
%Canell
T1r0=[1 0 0 Xb;0 1 0 Yb;0 0 1 Zb;0 0 0 1];
T2r1=[1 0 0 0; 0 cos(ALPHAb) -sin(ALPHAb) 0; 0 sin(ALPHAb) cos(ALPHAb) 0; 0 0 0 1];
T3r2=[cos(BETAb) 0 -sin(BETAb) 0;0 1 0 0;sin(BETAb) 0 cos(BETAb) 0;0 0 0 1];
T4r3=[cos(GAMMAb) -sin(GAMMAb) 0 0;sin(GAMMAb) cos(GAMMAb) 0 0;0 0 1 0;0 0 0 1];
%Index
T5r4a=[0 -1 0 0;1 0 0 0;0 0 1 197.55;0 0 0 1];
T6r5a=[cos(TET2i) -sin(TET2i) 0 67;0 0 -1 -9.5;sin(TET2i) cos(TET2i) 0 0;0 0 0 1];
T7r6a=[cos(TET3i) -sin(TET3i) 0 0;0 0 -1 0;sin(TET3i) cos(TET3i) 0 0;0 0 0 1];
T8r7a=[cos(TET4i) -sin(TET4i) 0 76.66;sin(TET4i) cos(TET4i) 0 0;0 0 1 0;0 0 0 1];
T9r8a=[cos(TET5IIi) -sin(TET5IIi) 0 56;sin(TET5IIi) cos(TET5IIi) 0 0;0 0 1 0;
       0 0 0 1];
T10r9a=[cos(TET6IIi) -sin(TET6IIi) 0 33.62;sin(TET6IIi) cos(TET6IIi) 0 0;0 0 1 0;
       0 0 0 1];
T11r10a=[cos(TET7i) -sin(TET7i) 0 0;0 0 1 0;-sin(TET7i) -cos(TET7i) 0 0;0 0 0 1];
T12r11a=[1 0 0 20;0 1 0 0;0 0 1 0;0 0 0 1];



% CAP AL JACOBIA ----------------------------------------------------

% Blocs de frames
%Pos Inicial
T1e0=T1r0;
T2e0=T1e0*T2r1;
T3e0=T2e0*T3r2;
T4e0=T3e0*T4r3;
%Dit
T5e0pa=T4e0*T5r4a;
T6e0pa=T5e0pa*T6r5a;
T7e0pa=T6e0pa*T7r6a;
T8e0pa=T7e0pa*T8r7a;
T9e0pa=T8e0pa*T9r8a;
T10e0pa=T9e0pa*T10r9a;
T11e0pa=T10e0pa*T11r10a;
T12e0pa=T11e0pa*T12r11a;


%Pos Inicial
sA=sym('sA','real');
cA=sym('cA','real');
sB=sym('sB','real');
cB=sym('cB','real');
sG=sym('sG','real');
cG=sym('cG','real');
%Index
```

```
s2I = sym('s2I','real');
c2I = sym('c2I','real');
s3I = sym('s3I','real');
c3I = sym('c3I','real');
s4I = sym('s4I','real');
c4I = sym('c4I','real');
s5I = sym('s5I','real');
c5I = sym('c5I','real');
s6I = sym('s6I','real');
c6I = sym('c6I','real');
s7I = sym('s7I','real');
c7I = sym('c7I','real');



JBA = T12e0pa(:,1);

% SUBSTITUCIONS
% Canell
JBA = subs(JBA,sin(ALPHAb),sA);
JBA = subs(JBA,cos(ALPHAb),cA);
JBA = subs(JBA,sin(BETAb),sB);
JBA = subs(JBA,cos(BETAb),cB);
JBA = subs(JBA,sin(GAMMAb),sG);
JBA = subs(JBA,cos(GAMMAb),cG);
% Index
JBA = subs(JBA,sin(TET2i),s2I);
JBA = subs(JBA,cos(TET2i),c2I);
JBA = subs(JBA,sin(TET3i),s3I);
JBA = subs(JBA,cos(TET3i),c3I);
JBA = subs(JBA,sin(TET4i),s4I);
JBA = subs(JBA,cos(TET4i),c4I);
JBA = subs(JBA,sin(TET5IIi),s5I);
JBA = subs(JBA,cos(TET5IIi),c5I);
JBA = subs(JBA,sin(TET6IIi),s6I);
JBA = subs(JBA,cos(TET6IIi),c6I);
JBA = subs(JBA,sin(TET7i),s7I);
JBA = subs(JBA,cos(TET7i),c7I);



JBA = vpa(JBA);
```

Polze

```
clear all;
close all;

% Declaracio de variables
%Pos Inicial
Xb=sym('Xb','real');
Yb=sym('Yb','real');
Zb=sym('Zb','real');
ALPHAb=sym('ALPHAb','real');
BETAb=sym('BETAb','real');
```

```
GAMMAb=sym('GAMMAb','real');
%Polze
TET2P = sym('TET2P','real');
TET3P = sym('TET3P','real');
TET4P = sym('TET4P','real');
TET5IP = sym('TET5IP','real');
TET6IP = sym('TET6IP','real');
TET7P = sym('TET7P','real');


% Matrius de transformacio homogenia
%Canell
T1r0=[1 0 0 Xb;0 1 0 Yb;0 0 1 Zb;0 0 0 1];
T2r1=[1 0 0 0; 0 cos(ALPHAb) -sin(ALPHAb) 0; 0 sin(ALPHAb) cos(ALPHAb) 0; 0 0 0 1];
T3r2=[cos(BETAb) 0 -sin(BETAb) 0;0 1 0 0;sin(BETAb) 0 cos(BETAb) 0;0 0 0 1];
T4r3=[cos(GAMMAb) -sin(GAMMAb) 0 0;sin(GAMMAb) cos(GAMMAb) 0 0;0 0 1 0;0 0 0 1];
%Polze
T5r4a=[cos((-44.56)*(pi/180)) -sin((-44.56)*(pi/180)) 0 0;
       sin((-44.56)*(pi/180)) cos((-44.56)*(pi/180)) 0 0;0 0 1 264;0 0 0 1];
T6r5a=[cos(TET2P) -sin(TET2P) 0 7.56;
       cos(14.11*(pi/180))*sin(TET2P) cos(14.11*(pi/180))*cos(TET2P)
              -sin(14.11*(pi/180)) 203.32*sin(14.11*(pi/180));
       sin(14.11*(pi/180))*sin(TET2P) sin(14.11*(pi/180))*cos(TET2P)
              cos(14.11*(pi/180)) -203.32*cos(14.11*(pi/180));
       0 0 0 1];
T7r6a=[cos(TET3P) -sin(TET3P) 0 0;0 0 -1 0;sin(TET3P) cos(TET3P) 0 0;0 0 0 1];
T8r7a=[cos(TET4P) -sin(TET4P) 0 76.66;sin(TET4P) cos(TET4P) 0 0;0 0 1 0;0 0 0 1];
T9r8a=[cos(TET5IP) -sin(TET5IP) 0 66;sin(TET5IP) cos(TET5IP) 0 0;0 0 1 0;
       0 0 0 1];
T10r9a=[cos(TET6IP) -sin(TET6IP) 0 39.17;sin(TET6IP) cos(TET6IP) 0 0;0 0 1 0;
       0 0 0 1];
T11r10a=[cos(TET7P) -sin(TET7P) 0 0;0 0 1 0;-sin(TET7P) -cos(TET7P) 0 0;0 0 0 1];
T12r11a=[1 0 0 20;0 1 0 0;0 0 1 0;0 0 0 1];


% CAP AL JACOBIA ----------------------------------------------

% Blocs de frames
%Pos Inicial
T1e0=T1r0;
T2e0=T1e0*T2r1;
T3e0=T2e0*T3r2;
T4e0=T3e0*T4r3;
%Dit
T5e0pa=T4e0*T5r4a;
T6e0pa=T5e0pa*T6r5a;
T7e0pa=T6e0pa*T7r6a;
T8e0pa=T7e0pa*T8r7a;
T9e0pa=T8e0pa*T9r8a;
T10e0pa=T9e0pa*T10r9a;
T11e0pa=T10e0pa*T11r10a;
T12e0pa=T11e0pa*T12r11a;


%Pos Inicial
```

```
sA=sym('sA','real');
cA=sym('cA','real');
sB=sym('sB','real');
cB=sym('cB','real');
sG=sym('sG','real');
cG=sym('cG','real');
%Polze
s2P = sym('s2P','real');
c2P = sym('c2P','real');
s3P = sym('s3P','real');
c3P = sym('c3P','real');
s4P = sym('s4P','real');
c4P = sym('c4P','real');
s5P = sym('s5P','real');
c5P = sym('c5P','real');
s6P = sym('s6P','real');
c6P = sym('c6P','real');
s7P = sym('s7P','real');
c7P = sym('c7P','real');


JBA = T12e0pa(:,1);

% SUBSTITUCIONS
% Canell
JBA = subs(JBA,sin(ALPHAb),sA);
JBA = subs(JBA,cos(ALPHAb),cA);
JBA = subs(JBA,sin(BETAb),sB);
JBA = subs(JBA,cos(BETAb),cB);
JBA = subs(JBA,sin(GAMMAb),sG);
JBA = subs(JBA,cos(GAMMAb),cG);
% Polze
JBA = subs(JBA,sin(TET2P),s2P);
JBA = subs(JBA,cos(TET2P),c2P);
JBA = subs(JBA,sin(TET3P),s3P);
JBA = subs(JBA,cos(TET3P),c3P);
JBA = subs(JBA,sin(TET4P),s4P);
JBA = subs(JBA,cos(TET4P),c4P);
JBA = subs(JBA,sin(TET5IP),s5P);
JBA = subs(JBA,cos(TET5IP),c5P);
JBA = subs(JBA,sin(TET6IP),s6P);
JBA = subs(JBA,cos(TET6IP),c6P);
JBA = subs(JBA,sin(TET7P),s7P);
JBA = subs(JBA,cos(TET7P),c7P);

JBA = vpa(JBA);
```

Apèndix D

El model simplificat de la mà: fitxers *inventor*

Tot seguit es mostren els fitxers que contenen els elements del model tridimensional de la mà i dels punts de contacte que s'empra per al testeig de col·lisions. De fet s'han creat dos models, un que conté la mà únicament, i un segon que conté el model tridimensional de la mà i els punts de contacte.

En primer lloc es mostra el fitxer que carrega els fitxers *inventor* del model de la mà; en segon lloc, el que carrega la mà i els punts de contacte de l'objecte. Aquests dos fitxers són fitxers XML.

Finalment es presenten els fitxers *inventor* (amb extensió *.iv*) que contenen les dimensions i color de cada falange i tou dels dits, del palmell i dels punts de contacte de l'objecte.

D.1 Fitxers XML de càrrega

Mà: mai_v2.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<Scene>


  <!-- ANULAR -->

  <Geomobject>
```

```
    <File objectfile="objects_ma/anular/cylinder_f1_a.iv">
    </File>
    Object File
    <Type T="mesh">
      Type of geomobject
    </Type>
    <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
      Object Position and Orientation
    </Configuration>
    <Properties p1="␣" p2="␣" p3="␣">
      not used
    </Properties>
    <Scale sf="1.0">
      Scale factor to be applied to the ivfile
    </Scale>
  </Geomobject>

  <Geomobject>
         <File objectfile="objects_ma/anular/cylinder_f2_a.iv">
    </File>
    Object File
    <Type T="mesh">
      Type of geomobject
    </Type>
    <Configuration TH="1.7" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
      Object Position and Orientation
    </Configuration>
    <Properties p1="␣" p2="␣" p3="␣">
      not used
    </Properties>
    <Scale sf="1.0">
      Scale factor to be applied to the ivfile
    </Scale>
  </Geomobject>

<Geomobject>
        <File objectfile="objects_ma/anular/cylinder_f3_a.iv">
    </File>
    Object File
    <Type T="mesh">
      Type of geomobject
    </Type>
    <Configuration TH="0.8" WZ="0.0" WY="0.0" WX="1.0" Z="0.0" Y="0.0" X="0.0">
      Object Position and Orientation
    </Configuration>
    <Properties p1="␣" p2="␣" p3="␣">
      not used
    </Properties>
    <Scale sf="1.0">
      Scale factor to be applied to the ivfile
    </Scale>
  </Geomobject>

  <Geomobject>
         <File objectfile="objects_ma/anular/sphere_ft_a.iv">
```

```
        </File>
        Object File
        <Type T="sphere">
                Type of geomobject
        </Type>
        <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Initial Configuration of the robot
        </Configuration>
        <Properties p1="20.0" p2=" " p3=" ">
                p1 contains Sphere radius - p2 and p3 not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
        </Scale>
</Geomobject>



<!-- COR -->

<Geomobject>
        <File objectfile="objects_ma/middle/cylinder_f1_m.iv">
  </File>
  Object File
  <Type T="mesh">
    Type of geomobject
  </Type>
  <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
    Object Position and Orientation
  </Configuration>
  <Properties p1=" " p2=" " p3=" ">
    not used
  </Properties>
  <Scale sf="1.0">
    Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>

<Geomobject>
        <File objectfile="objects_ma/middle/cylinder_f2_m.iv">
  </File>
  Object File
  <Type T="mesh">
    Type of geomobject
  </Type>
  <Configuration TH="1.7" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
    Object Position and Orientation
  </Configuration>
  <Properties p1=" " p2=" " p3=" ">
    not used
  </Properties>
  <Scale sf="1.0">
    Scale factor to be applied to the ivfile
  </Scale>
```

```
 </Geomobject>

<Geomobject>
        <File objectfile="objects_ma/middle/cylinder_f3_m.iv">
   </File>
   Object File
   <Type T="mesh">
     Type of geomobject
   </Type>
   <Configuration TH="0.8" WZ="0.0" WY="0.0" WX="1.0" Z="0.0" Y="0.0" X="0.0">
     Object Position and Orientation
   </Configuration>
   <Properties p1="␣" p2="␣" p3="␣">
     not used
   </Properties>
   <Scale sf="1.0">
     Scale factor to be applied to the ivfile
   </Scale>
 </Geomobject>

 <Geomobject>
         <File objectfile="objects_ma/middle/sphere_ft_m.iv">
         </File>
         Object File
         <Type T="sphere">
                 Type of geomobject
         </Type>
         <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                 Initial Configuration of the robot
         </Configuration>
         <Properties p1="20.0" p2="␣" p3="␣">
                 p1 contains Sphere radius - p2 and p3 not used
         </Properties>
         <Scale sf="1.0">
                 Scale factor to be applied to the ivfile
         </Scale>
 </Geomobject>




 <!-- INDEX -->

 <Geomobject>
         <File objectfile="objects_ma/index/cylinder_f1_i.iv">
   </File>
   Object File
   <Type T="mesh">
     Type of geomobject
   </Type>
   <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
     Object Position and Orientation
   </Configuration>
   <Properties p1="␣" p2="␣" p3="␣">
     not used
```

```
    </Properties>
    <Scale sf="1.0">
      Scale factor to be applied to the ivfile
    </Scale>
  </Geomobject>

  <Geomobject>
          <File objectfile="objects_ma/index/cylinder_f2_i.iv">
    </File>
    Object File
    <Type T="mesh">
      Type of geomobject
    </Type>
    <Configuration TH="1.7" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
      Object Position and Orientation
    </Configuration>
    <Properties p1=" " p2=" " p3=" ">
      not used
    </Properties>
    <Scale sf="1.0">
      Scale factor to be applied to the ivfile
    </Scale>
  </Geomobject>

<Geomobject>
        <File objectfile="objects_ma/index/cylinder_f3_i.iv">
    </File>
    Object File
    <Type T="mesh">
      Type of geomobject
    </Type>
    <Configuration TH="0.8" WZ="0.0" WY="0.0" WX="1.0" Z="0.0" Y="0.0" X="0.0">
      Object Position and Orientation
    </Configuration>
    <Properties p1=" " p2=" " p3=" ">
      not used
    </Properties>
    <Scale sf="1.0">
      Scale factor to be applied to the ivfile
    </Scale>
  </Geomobject>

  <Geomobject>
          <File objectfile="objects_ma/index/sphere_ft_i.iv">
          </File>
          Object File
          <Type T="sphere">
                  Type of geomobject
          </Type>
          <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                  Initial Configuration of the robot
          </Configuration>
          <Properties p1="20.0" p2=" " p3=" ">
                  p1 contains Sphere radius - p2 and p3 not used
          </Properties>
```

```
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
        </Scale>
</Geomobject>



<!-- POLZE -->

<Geomobject>
        <File objectfile="objects_ma/thumb/cylinder_f1_t.iv">
  </File>
  Object File
  <Type T="mesh">
    Type of geomobject
  </Type>
  <Configuration TH="0.0" WZ="0.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
    Object Position and Orientation
  </Configuration>
  <Properties p1="" p2="" p3="">
    not used
  </Properties>
  <Scale sf="1.0">
    Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>

<Geomobject>
        <File objectfile="objects_ma/thumb/cylinder_f2_t.iv">
  </File>
  Object File
  <Type T="mesh">
    Type of geomobject
  </Type>
  <Configuration TH="0.0" WZ="0.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
    Object Position and Orientation
  </Configuration>
  <Properties p1="" p2="" p3="">
    not used
  </Properties>
  <Scale sf="1.0">
    Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>

<Geomobject>
        <File objectfile="objects_ma/thumb/cylinder_f3_t.iv">
  </File>
  Object File
  <Type T="mesh">
    Type of geomobject
  </Type>
  <Configuration TH="0.0" WZ="0.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
    Object Position and Orientation
```

```
  </Configuration>
  <Properties p1="␣" p2="␣" p3="␣">
    not used
  </Properties>
  <Scale sf="1.0">
    Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>

<Geomobject>
        <File objectfile="objects_ma/thumb/sphere_ft_t.iv">
        </File>
        Object File
        <Type T="sphere">
                Type of geomobject
        </Type>
        <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Initial Configuration of the robot
        </Configuration>
        <Properties p1="20.0" p2="␣" p3="␣">
                p1 contains Sphere radius - p2 and p3 not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
        </Scale>
</Geomobject>


<!-- PALMELL -->

<Geomobject>
<File objectfile="objects_ma/palmell.iv">
  </File>
  Object File
  <Type T="cube">
    Type of geomobject
  </Type>
        <Configuration TH="0.0" WZ="1.578" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Initial Configuration of the robot
        </Configuration>
        <Properties p1="␣" p2="␣" p3="␣">
                not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>


</Scene>
```

Mà i punts de contacte: mai_pod_v2.xml

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<Scene>


  <!-- ANULAR -->

  <Geomobject>
    <File objectfile="objects_ma/anular/cylinder_f1_a.iv">
    </File>
    Object File
    <Type T="mesh">
      Type of geomobject
    </Type>
    <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
      Object Position and Orientation
    </Configuration>
    <Properties p1="␣" p2="␣" p3="␣">
      not used
    </Properties>
    <Scale sf="1.0">
      Scale factor to be applied to the ivfile
    </Scale>
  </Geomobject>

  <Geomobject>
          <File objectfile="objects_ma/anular/cylinder_f2_a.iv">
    </File>
    Object File
    <Type T="mesh">
      Type of geomobject
    </Type>
    <Configuration TH="1.7" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
      Object Position and Orientation
    </Configuration>
    <Properties p1="␣" p2="␣" p3="␣">
      not used
    </Properties>
    <Scale sf="1.0">
      Scale factor to be applied to the ivfile
    </Scale>
  </Geomobject>

 <Geomobject>
         <File objectfile="objects_ma/anular/cylinder_f3_a.iv">
    </File>
    Object File
    <Type T="mesh">
      Type of geomobject
    </Type>
    <Configuration TH="0.8" WZ="0.0" WY="0.0" WX="1.0" Z="0.0" Y="0.0" X="0.0">
      Object Position and Orientation
    </Configuration>
    <Properties p1="␣" p2="␣" p3="␣">
      not used
    </Properties>
```

```
  <Scale sf="1.0">
    Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>

<Geomobject>
        <File objectfile="objects_ma/anular/sphere_ft_a.iv">
        </File>
        Object File
        <Type T="sphere">
                Type of geomobject
        </Type>
        <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Initial Configuration of the robot
        </Configuration>
        <Properties p1="20.0" p2=" " p3=" ">
                p1 contains Sphere radius - p2 and p3 not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
        </Scale>
</Geomobject>



<!-- COR -->

<Geomobject>
        <File objectfile="objects_ma/middle/cylinder_f1_m.iv">
  </File>
  Object File
  <Type T="mesh">
    Type of geomobject
  </Type>
  <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
    Object Position and Orientation
  </Configuration>
  <Properties p1=" " p2=" " p3=" ">
    not used
  </Properties>
  <Scale sf="1.0">
    Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>

<Geomobject>
        <File objectfile="objects_ma/middle/cylinder_f2_m.iv">
  </File>
  Object File
  <Type T="mesh">
    Type of geomobject
  </Type>
  <Configuration TH="1.7" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
    Object Position and Orientation
```

```
    </Configuration>
    <Properties p1=" " p2=" " p3=" ">
      not used
    </Properties>
    <Scale sf="1.0">
      Scale factor to be applied to the ivfile
    </Scale>
  </Geomobject>

<Geomobject>
        <File objectfile="objects_ma/middle/cylinder_f3_m.iv">
    </File>
    Object File
    <Type T="mesh">
      Type of geomobject
    </Type>
    <Configuration TH="0.8" WZ="0.0" WY="0.0" WX="1.0" Z="0.0" Y="0.0" X="0.0">
      Object Position and Orientation
    </Configuration>
    <Properties p1=" " p2=" " p3=" ">
      not used
    </Properties>
    <Scale sf="1.0">
      Scale factor to be applied to the ivfile
    </Scale>
  </Geomobject>

  <Geomobject>
          <File objectfile="objects_ma/middle/sphere_ft_m.iv">
          </File>
          Object File
          <Type T="sphere">
                  Type of geomobject
          </Type>
          <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                  Initial Configuration of the robot
          </Configuration>
          <Properties p1="20.0" p2=" " p3=" ">
                  p1 contains Sphere radius - p2 and p3 not used
          </Properties>
          <Scale sf="1.0">
                  Scale factor to be applied to the ivfile
          </Scale>
  </Geomobject>




  <!-- INDEX -->

  <Geomobject>
          <File objectfile="objects_ma/index/cylinder_f1_i.iv">
    </File>
    Object File
    <Type T="mesh">
```

```
    Type of geomobject
  </Type>
  <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
    Object Position and Orientation
  </Configuration>
  <Properties p1="␣" p2="␣" p3="␣">
    not used
  </Properties>
  <Scale sf="1.0">
    Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>

<Geomobject>
      <File objectfile="objects_ma/index/cylinder_f2_i.iv">
  </File>
  Object File
  <Type T="mesh">
    Type of geomobject
  </Type>
  <Configuration TH="1.7" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
    Object Position and Orientation
  </Configuration>
  <Properties p1="␣" p2="␣" p3="␣">
    not used
  </Properties>
  <Scale sf="1.0">
    Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>

<Geomobject>
      <File objectfile="objects_ma/index/cylinder_f3_i.iv">
  </File>
  Object File
  <Type T="mesh">
    Type of geomobject
  </Type>
  <Configuration TH="0.8" WZ="0.0" WY="0.0" WX="1.0" Z="0.0" Y="0.0" X="0.0">
    Object Position and Orientation
  </Configuration>
  <Properties p1="␣" p2="␣" p3="␣">
    not used
  </Properties>
  <Scale sf="1.0">
    Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>

<Geomobject>
      <File objectfile="objects_ma/index/sphere_ft_i.iv">
      </File>
      Object File
      <Type T="sphere">
            Type of geomobject
```

```
        </Type>
        <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Initial Configuration of the robot
        </Configuration>
        <Properties p1="20.0" p2=" " p3=" ">
                p1 contains Sphere radius - p2 and p3 not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
        </Scale>
</Geomobject>



<!-- POLZE -->

<Geomobject>
        <File objectfile="objects_ma/thumb/cylinder_f1_t.iv">
  </File>
  Object File
  <Type T="mesh">
    Type of geomobject
  </Type>
  <Configuration TH="0.0" WZ="0.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
    Object Position and Orientation
  </Configuration>
  <Properties p1=" " p2=" " p3=" ">
    not used
  </Properties>
  <Scale sf="1.0">
    Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>

<Geomobject>
        <File objectfile="objects_ma/thumb/cylinder_f2_t.iv">
  </File>
  Object File
  <Type T="mesh">
    Type of geomobject
  </Type>
  <Configuration TH="0.0" WZ="0.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
    Object Position and Orientation
  </Configuration>
  <Properties p1=" " p2=" " p3=" ">
    not used
  </Properties>
  <Scale sf="1.0">
    Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>

<Geomobject>
        <File objectfile="objects_ma/thumb/cylinder_f3_t.iv">
```

```
  </File>
  Object File
  <Type T="mesh">
    Type of geomobject
  </Type>
  <Configuration TH="0.0" WZ="0.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
    Object Position and Orientation
  </Configuration>
  <Properties p1=" " p2=" " p3=" ">
    not used
  </Properties>
  <Scale sf="1.0">
    Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>

<Geomobject>
        <File objectfile="objects_ma/thumb/sphere_ft_t.iv">
        </File>
        Object File
        <Type T="sphere">
                Type of geomobject
        </Type>
        <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Initial Configuration of the robot
        </Configuration>
        <Properties p1="20.0" p2=" " p3=" ">
                p1 contains Sphere radius - p2 and p3 not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
        </Scale>
</Geomobject>


<!-- PALMELL -->

<Geomobject>
<File objectfile="objects_ma/palmell.iv">
  </File>
  Object File
  <Type T="cube">
    Type of geomobject
  </Type>
        <Configuration TH="0.0" WZ="1.578" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Initial Configuration of the robot
        </Configuration>
        <Properties p1=" " p2=" " p3=" ">
                not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
  </Scale>
</Geomobject>
```

```
<!-- POD -->
<Geomobject>
        <File objectfile="objects_ma/pod/sphere_pod_a.iv">
      </File>
                Object File
        <Type T="sphere">
                Type of geomobject
        </Type>
        <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Initial Configuration of the robot
        </Configuration>
        <Properties p1="20.0" p2=" " p3=" ">
                p1 contains Sphere radius - p2 and p3 not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
        </Scale>
</Geomobject>


<Geomobject>
        <File objectfile="objects_ma/pod/cylinder_pod_a.iv">
        </File>
        Object File
        <Type T="mesh">
                Type of geomobject
        </Type>
        <Configuration TH="0.0" WZ="0.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Object Position and Orientation
        </Configuration>
        <Properties p1=" " p2=" " p3=" ">
                not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
        </Scale>
</Geomobject>


<Geomobject>
        <File objectfile="objects_ma/pod/sphere_pod_m.iv">
        </File>
        Object File
        <Type T="sphere">
                Type of geomobject
        </Type>
        <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Initial Configuration of the robot
        </Configuration>
        <Properties p1="20.0" p2=" " p3=" ">
                p1 contains Sphere radius - p2 and p3 not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
```

```
        </Scale>
</Geomobject>


<Geomobject>
        <File objectfile="objects_ma/pod/cylinder_pod_m.iv">
        </File>
        Object File
        <Type T="mesh">
                Type of geomobject
        </Type>
        <Configuration TH="0.0" WZ="0.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Object Position and Orientation
        </Configuration>
        <Properties p1=" " p2=" " p3=" ">
                not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
        </Scale>
</Geomobject>


<Geomobject>
        <File objectfile="objects_ma/pod/sphere_pod_i.iv">
        </File>
        Object File
        <Type T="sphere">
                Type of geomobject
        </Type>
        <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Initial Configuration of the robot
        </Configuration>
        <Properties p1="20.0" p2=" " p3=" ">
                p1 contains Sphere radius - p2 and p3 not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
        </Scale>
</Geomobject>


<Geomobject>
        <File objectfile="objects_ma/pod/cylinder_pod_i.iv">
        </File>
        Object File
        <Type T="mesh">
                Type of geomobject
        </Type>
        <Configuration TH="0.0" WZ="0.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Object Position and Orientation
        </Configuration>
        <Properties p1=" " p2=" " p3=" ">
                not used
        </Properties>
```

```
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
        </Scale>
</Geomobject>


<Geomobject>
        <File objectfile="objects_ma/pod/sphere_pod_t.iv">
        </File>
        Object File
        <Type T="sphere">
                Type of geomobject
        </Type>
        <Configuration TH="0.0" WZ="1.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Initial Configuration of the robot
        </Configuration>
        <Properties p1="20.0" p2=" " p3=" ">
                p1 contains Sphere radius - p2 and p3 not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
        </Scale>
</Geomobject>


<Geomobject>
        <File objectfile="objects_ma/pod/cylinder_pod_t.iv">
        </File>
        Object File
        <Type T="mesh">
                Type of geomobject
        </Type>
        <Configuration TH="0.0" WZ="0.0" WY="0.0" WX="0.0" Z="0.0" Y="0.0" X="0.0">
                Object Position and Orientation
        </Configuration>
        <Properties p1=" " p2=" " p3=" ">
                not used
        </Properties>
        <Scale sf="1.0">
                Scale factor to be applied to the ivfile
        </Scale>
</Geomobject>



</Scene>
```

D.2 Fitxers *inventor*

Palmell: palmell.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        1.0 1.0 1.0
    }

    Cube {
     width 174.4
     height 200.0
     depth 40.0
    }
}
```

Anular - Falange proximal: cylinder_f1_a.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.0 1.0 0.2
    }

    Cylinder {
     radius 20.0
     height 76.6
    }
}
```

Anular - Falangeta: cylinder_f2_a.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.0 1.0 0.2
    }

    Cylinder {
     radius 20.0
     height 56.0
    }
}
```

Anular - Falangina: cylinder_f3_a.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.0 1.0 0.2
    }
```

```
    Cylinder {
     radius 20.0
     height 33.62
    }
}
```

Anular - Tou: sphere_ft_a.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.0 1.0 0.2
    }

    Sphere {
     radius 20.0
    }
}
```

Mitjer - Falange proximal: cylinder_f1_m.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.0 0.75 1.0
    }

    Cylinder {
     radius 20.0
     height 76.6
    }
}
```

Mitjer - Falangeta: cylinder_f2_m.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.0 0.75 1.0
    }

    Cylinder {
     radius 20.0
     height 56.0
    }
}
```

Mitjer - Falangina: cylinder_f3_m.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.0 0.75 1.0
    }

    Cylinder {
     radius 20.0
     height 33.62
    }
}
```

Mitjer - Tou: sphere_ft_m.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.0 0.75 1.0
    }

    Sphere {
     radius 20.0
    }
}
```

Índex - Falange proximal: cylinder_f1_i.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        1.0 0.35 0.0
    }

    Cylinder {
     radius 20.0
     height 76.6
    }
}
```

Índex - Falangeta: cylinder_f2_i.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        1.0 0.35 0.0
    }

    Cylinder {
```

```
     radius 20.0
     height 56.0
    }
}
```

Índex - Falangina: cylinder_f3_i.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        1.0 0.35 0.0
    }

    Cylinder {
     radius 20.0
     height 33.62
    }
}
```

Índex - Tou: sphere_ft_i.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        1.0 0.35 0.0
    }

    Sphere {
     radius 20.0
    }
}
```

Polze - Falange proximal: cylinder_f1_t.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        1.0 1.0 0.0
    }

    Cylinder {
     radius 20.0
     height 76.6
    }
}
```

Polze - Falangeta: cylinder_f2_t.iv

```
#Inventor V2.1 ascii
```

```
Separator {
    Material {
            diffuseColor        1.0 1.0 0.0
    }

    Cylinder {
     radius 20.0
     height 66.0
    }
}
```

Polze - Falangina: cylinder_f3_t.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        1.0 1.0 0.0
    }

    Cylinder {
     radius 20.0
     height 39.17
    }
}
```

Polze - Tou: sphere_ft_t.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        1.0 1.0 0.0
    }

    Sphere {
     radius 20.0
    }
}
```

Punts de contacte de l'objecte - Anular - Esfera: sphere_pod_a.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.0 0.6 0.0
    }

    Sphere {
     radius 20.0
    }
```

```
}
```

Punts de contacte de l'objecte - Anular - Cilíndre: cylinder_pod_a.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.0 0.6 0.0
    }

    Cylinder {
     radius 5.0
     height 40.0
    }
}
```

Punts de contacte de l'objecte - Mitjer - Esfera: sphere_pod_m.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.0 0.35 0.6
    }

    Sphere {
     radius 20.0
    }
}
```

Punts de contacte de l'objecte - Mitjer - Cilíndre: cylinder_pod_m.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.0 0.35 0.6
    }

    Cylinder {
     radius 5.0
     height 40.0
    }
}
```

Punts de contacte de l'objecte - Índex - Esfera: sphere_pod_i.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
```

```
            diffuseColor        0.7 0.05 0.0
    }

    Sphere {
     radius 20.0
    }
}
```

Punts de contacte de l'objecte - Índex - Cilíndre: cylinder_pod_i.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.7 0.05 0.0
    }

    Cylinder {
     radius 5.0
     height 40.0
    }
}
```

Punts de contacte de l'objecte - Polze - Esfera: sphere_pod_t.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor         0.6 0.6 0.0
    }

    Sphere {
     radius 20.0
    }
}
```

Punts de contacte de l'objecte - Polze - Cilíndre: cylinder_pod_t.iv

```
#Inventor V2.1 ascii

Separator {
    Material {
            diffuseColor        0.6 0.6 0.0
    }

    Cylinder {
     radius 5.0
     height 40.0
    }
}
```