

Balancing a broomstick with constraint dynamics and *Mathematica*

John W. Perram, Morten E. Andersen, Carsten Panch Pedersen and Adam Sierakowski

*The Maersk Mc-Kinney Moller Institute for Production Technology
University of Southern Denmark*

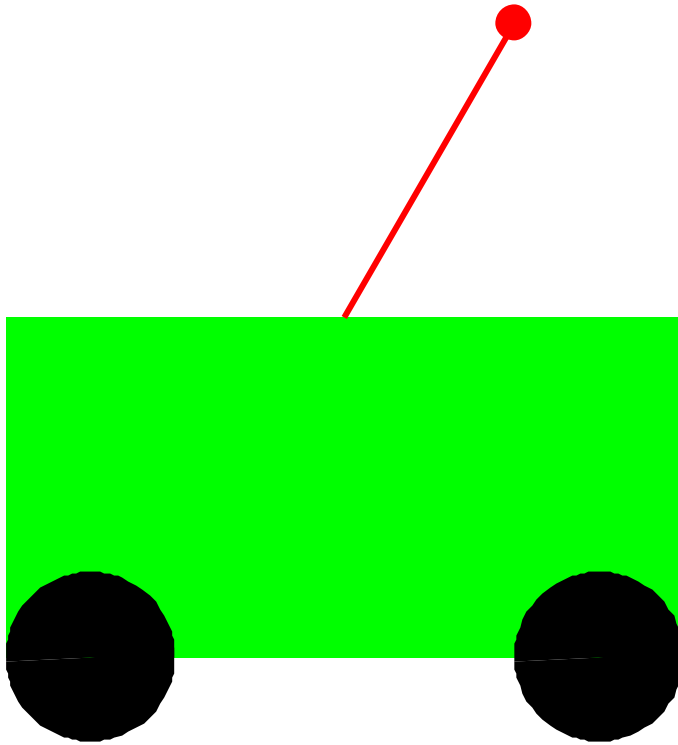
The teaching of analytical mechanics at the Maersk Institute

Analytical mechanics has always been one of the central competences at the Maersk Institute, and its precursor, the Lindo Center for Applied Mathematics. The use of techniques from constraint mechanics was a central feature of the AMROSE project, in which simulation was used to plan collision free paths in cluttered environments by following the motion of a model of the robot in a force field designed to attract it to its goal while repelling its parts from each other and the environment.

The course AM12 (Analytical and Computational Mechanics) is part of the bachelor programs for students of Computer Systems Engineering and Applied Mathematics. The course is currently taught by the first author, using electronic teaching material in the form of interactive *Mathematica* notebooks. Traditional tutorial problems, solved manually, have been replaced by structured projects, in which students learn to solve and document applications by cutting and pasting tools from an electronic toolbox, also a *Mathematica* notebook, into their own project report. The course is part of an experiment taking place at the Institute to evaluate novel forms of grading such projects, described at greater length in the body of this document. Students will be able to follow the course on-line when it is run again in the spring of 2003. The full text of this document can be downloaded in either *Mathematica* .nb or .pdf format from the URL: [http://www.mip.sdu.dk/~jperram/AM12/DIKUart.nb\(pdf\)](http://www.mip.sdu.dk/~jperram/AM12/DIKUart.nb(pdf)). Interested participants should contact the first author by email

Introduction

This paper is based on some of the solution reports for the compulsory project in the course AM12 (analytical and computational mechanics) when it was last run in the spring of 2002. This project deals with the classical problem of stabilising an inverted, planar pendulum by applying a suitable horizontal control force to the arrangement shown below, which starts from rest in a configuration such as that shown below.



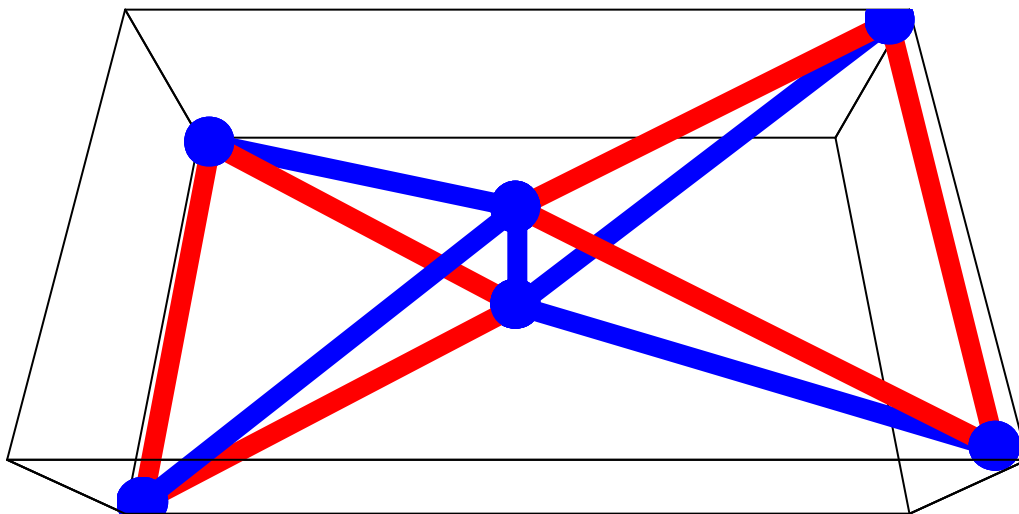
This example has been chosen because it illustrates some of the pedagogical experiments in the teaching of applied mathematics going on at the Maersk Institute, it nicely illustrates the power of the Institute's approach to simulating mechanical systems by calculating the constraint forces rather than generalised coordinates, and it provides a vehicle for presenting an apparently new result for this system which two of the students discovered in their investigations.

This course is part of the research going on in the Institute into IT-enhanced Applied Mathematics education, incorporating features such as:

- **Competence-building rather than pensum-based teaching**
- **Interactive learning material in the form of *Mathematica* notebooks which integrate mathematical text, graphics, executable code fragments and animation**
- **The identification of generic tools and their storage in a personal toolbox, from which they can be cut and pasted into project solution reports**
- **Learning by doing through structured tutorial projects and novel methods for evaluating these projects.**

Constraint dynamics

A central theme of the course AM12 is that the techniques of constraint mechanics offer both computational and pedagogical advantages over the traditional method of finding generalised coordinates. An important class of systems studied in the course are point particle models, such as the one illustrated below



which represents a point particle model of two rigid bodies (the light, red triangles) which are constrained to rotate about a common rotation axis (the heavy, vertical blue line). Such a situation occurs in industrial robots.

Each line (or bond) in the figure represents a distance constraint between the particles at the ends. A constraint reduces the number of degrees of freedom by one, so that this system has 11 constraints and thus 7 degrees of freedom. One triangle can perform rigid body motion (6 degrees of freedom) and the second one is restricted to rotating about the common axis (1 degree of freedom). During the motion, there will be forces directed along the bonds which act to preserve the constraints, so that each point will move according to external forces and the forces of constraint exerted by the particles to which it is bonded. The directions of these constraint forces are known, but not the magnitude. The scalars multiplying the direction of the constraint force are called the constraint force multipliers.

Constraint mechanics is a very compact technique for computing the forces of constraint, which is done by solving a set of linear equations for the constraint force multipliers. In this case, we have to solve a set of 11 by 11 linear equations for these multipliers,

which can be done rapidly because the positive-definite coefficient matrix is sparse. For very low-dimensional examples, such as the one studied here, the computations can be carried out analytically rather than numerically. In many cases, the ability to calculate the explicit form of the forces leads to new physics, such as this example, where an apparently new expression for the critical gain of a non-linear control system is obtained.

The technique can be extended to systems with thousands of constraints, such as biomolecules. The details are discussed at length in the AM12 electronic learning material, in the form of *Mathematica* notebooks, which can be downloaded from the AM12 home page, accessible from the first author's personal home page <http://www.mip.sdu.dk/~jperram/index.htm>.

Structured projects

The example discussed here is one of the structured tutorial projects studied in the 2002 version of the course AM12. Structured electronic tutorial projects were introduced by Professor E.R. Smith, the pro-vice chancellor responsible for on-line teaching at La Trobe University in Australia. The concept is based on the traditional mathematics tutorial in which students attempt to solve problems, obtaining assistance in the form of hints from a tutor.

The electronic version consists of a document in 4 layers. The top layer, visible to the student provides background information explaining the problem domain. The second layer, also visible to the student contains a brief description of the project deliverables.

The folding properties of the *Mathematica* notebook interface enable a standard solution of the project, structured as sections, subsections and subsubsections, to be hidden from the student. The text in the section headers corresponds to the deliverables in the project summary. The text in the headings of the subsections and subsubsections can be regarded as hints as how to proceed. If these hints are still not sufficient, the student can open the subsections containing the actual technical details of the solution. These procedures are illustrated in the case study below.

Students can thus evaluate their performance by trying to generate their own solution report in which all the deliverables are present and correct, assessing their own performance by keeping track of how many hints were needed to produce all the deliverables correctly.

In his masters thesis "Exercise Publisher" at the Maersk Institute, Morten Andersen has developed software for automatically keeping track of the number of hidden layers accessed by the student. This provides a basis for evaluation of performance in an examination situation, as the student needs to balance the constraint to generate all the deliverables correctly against the need to minimise the purchase of information so as to achieve a higher grade.

The project as it is initially visible to the student is shown below. Following that is a copy of two of the sections which have been partially opened to reveal interesting parts of the model project report.

Project introduction

This project is about using constraint dynamics to stabilize the inverted pendulum hinged to a cart by applying a suitable force to the cart. The setup is shown in the diagram above, where we suppose that the car moves in the plane of the figure.

We suppose that the mass of the cart is M and that the broomstick is modelled by a point mass m placed at the top of the massless stick whose length is L . The stick is smoothly hinged to the top of the cart. Denote the coordinates of the hinge by the vector $\{X, 0\}$ and the coordinates of the point mass m by $\{x, y\}$.

Initially the cart is at rest at the origin of coordinates, with the broomstick making an acute angle with the vertical direction. If we didn't do anything, the broomstick would fall down, because the vertical configuration is unstable. Intuitively, if the initial configuration was as shown, we would try to move the cart to the right to try and reduce this angle. We would probably overshoot, so that now we would pull rather than push the cart. This reasoning suggests that a force of the form

$$M k (x - X)$$

applied to the cart might ensure that the stick doesn't fall over. We might expect that the task of preventing the broomstick from falling over (meaning falling below a horizontal position) becomes more difficult as the initial angular deviation from the vertical direction increases. A force applied by us to achieve some objective is called a control law. The constant k multiplying the force is called the gain.

The system moves subject to the single constraint

$$(x - X)^2 + y^2 = L^2$$

Project summary

Use the toolbox to obtain the equations of motion symbolically.

By considering the initial vertical component of the acceleration of the pendulum mass m , argue that the system will exhibit oscillations about the vertical provided that the gain k satisfies the condition

$$k > \frac{g (m + M) \sec[\theta]}{L M}$$

Verify this formula by visualising numerical solutions of the equations of motion for small (10 degree) and medium (45 degree) initial angular deviations over a time interval of 15 seconds for a system where the length of the pendulum is 1 metre and the values of the masses m and M are 1 and 10 Kg respectively.

For the case when the pendulum exhibits stable oscillations around the vertical position, discuss the motion of cart as the value of the gain is increased above its critical value.

Monitor the constraint satisfaction in the simulations and discuss whether the constraint correction algorithm needs to be applied

Perform animations of the numerical solutions for a suitable case and comment on the results.

Project report

Derivation of the equations of motion

The critical gain

Numerical simulation for moderate (10 degree) angular displacements

Numerical simulation for large (45 degree) angular displacements

Numerical solution of the equations of motion

Monitoring and correcting constraint satisfaction

Animation of the simulation

Discussion

The above section headers, which hide the subsection hints and the model solution, can be partially opened to reveal a record of the hints used. In this case, we have opened those parts of the model project report which illustrate various novel aspects of the project

The sub-section header hints for the first deliverable are shown below, with the parts of the model report calculating the constraint forces.

Derivation of the equations of motion

The coordinates

The velocities

The accelerations

The inertia matrix

The constraints

The external forces

The constraint forces

According to D'Alembert's principle, the forces of constraint are in the direction of the normal to the constraint manifold, of the form

$$\mathbf{F}_{\text{con}} = \mathbf{J}^T \boldsymbol{\mu} \quad (10)$$

of the transpose of the Jacobian matrix obtained by taking the gradient of the constraints with respect to the coordinates, multiplying a vector of constraint force multipliers.

■ The Jacobian constraint matrix J

The elements of the matrix J are defined using

```
J = Table[ $\partial_{\text{coords}[[j]]} G[[i]]$ , {i, 1, Nc}, {j, 1, Nq}]
```

```
{ { 2 (x - X) , 2 y , -2 (x - X) } }
```

and as functions of time by executing

```
Jt = Table[ $\partial_{\text{coordst}[[j]]} Gt[[i]]$ , {i, 1, Nc}, {j, 1, Nq}]
```

```
{ { 2 (x[t] - X[t]) , 2 y[t] , -2 (x[t] - X[t]) } }
```

■ Time derivative of the matrix J

The time derivatives of the matrix **J** is computed using

```
Jd = Transpose[Table[ $\partial_{\text{coords}[[i]]} J$ , {i, 1, Nq}]] . velt
```

```
{ { 2 u[t] - 2 U[t] , 2 v[t] , -2 u[t] + 2 U[t] } }
```

■ The constraint multiplier matrix

The matrix multiplying the constraint multiplier vector in the equations for them is calculated using

```
Mmat = Simplify[Jt . Inverse[Amat] . Transpose[Jt]]
```

```
General::spell1: Possible spelling error: new symbol  
name "Mmat" is similar to existing symbol "Amat".
```

```
{ { 4  $\left( \frac{(x[t] - X[t])^2}{m} + \frac{(x[t] - X[t])^2}{M} + \frac{y[t]^2}{m} \right) } }$ 
```

which is only zero if $x = X$ and $y = 0$, which can never happen.

■ The right hand side vector of the linear equations for the constraint force multiplier

The right hand side vector in the linear equations for the constraints is calculated using

```
b = -Simplify[Jt . Inverse[Amat] . Fext] - Jd . velt
```

```
{ -u[t] (2 u[t] - 2 U[t]) - U[t] (-2 u[t] + 2 U[t]) - 2 v[t]^2 + 2 k (x[t] - X[t])^2 +  
2 g y[t] }
```

■ Solution for the constraint force multiplier

The constraint multiplier vector μ is then found by executing

```
 $\mu = \text{Simplify}[\text{LinearSolve}[\text{Mmat}, \text{b}]]$ 
```

```
 $\{ - (m M (u[t]^2 - 2 u[t] U[t] + U[t]^2 + v[t]^2 - k x[t]^2 + 2 k x[t] X[t] - k X[t]^2 - g y[t])) / (2 ((m + M) x[t]^2 - 2 (m + M) x[t] X[t] + (m + M) X[t]^2 + M y[t]^2)) \}$ 
```

■ Forces on the cart and broomstick

The forces are thus given by

```
 $\text{Forces} = \text{Fext} + \text{Simplify}[\text{Transpose}[\text{Jt}] . \mu]$ 
```

```
 $\{ - (m M (x[t] - X[t]) (u[t]^2 - 2 u[t] U[t] + U[t]^2 + v[t]^2 - k x[t]^2 + 2 k x[t] X[t] - k X[t]^2 - g y[t])) / ((m + M) x[t]^2 - 2 (m + M) x[t] X[t] + (m + M) X[t]^2 + M y[t]^2), -g m - (m M y[t] (u[t]^2 - 2 u[t] U[t] + U[t]^2 + v[t]^2 - k x[t]^2 + 2 k x[t] X[t] - k X[t]^2 - g y[t])) / ((m + M) x[t]^2 - 2 (m + M) x[t] X[t] + (m + M) X[t]^2 + M y[t]^2), k M (x[t] - X[t]) + (m M (x[t] - X[t]) (u[t]^2 - 2 u[t] U[t] + U[t]^2 + v[t]^2 - k x[t]^2 + 2 k x[t] X[t] - k X[t]^2 - g y[t])) / ((m + M) x[t]^2 - 2 (m + M) x[t] X[t] + (m + M) X[t]^2 + M y[t]^2) \}$ 
```

The equations of motion

The accelerations are then computed using


```
acceleration = Simplify[Inverse[Amat] . Forces]
```

```
{ - (M (x[t] - X[t]) (u[t]^2 - 2 u[t] U[t] +
      U[t]^2 + v[t]^2 - k x[t]^2 + 2 k x[t] X[t] - k X[t]^2 - g y[t])) /
  ( (m + M) x[t]^2 - 2 (m + M) x[t] X[t] + (m + M) X[t]^2 + M y[t]^2) ,
  -g - (M y[t] (u[t]^2 - 2 u[t] U[t] +
      U[t]^2 + v[t]^2 - k x[t]^2 + 2 k x[t] X[t] - k X[t]^2 - g y[t])) /
  ( (m + M) x[t]^2 - 2 (m + M) x[t] X[t] + (m + M) X[t]^2 + M y[t]^2) ,
  ( (x[t] - X[t]) (m u[t]^2 - 2 m u[t] U[t] + m U[t]^2 + m v[t]^2 +
      k M x[t]^2 - 2 k M x[t] X[t] + k M X[t]^2 - g m y[t] + k M y[t]^2) ) /
  ( (m + M) x[t]^2 - 2 (m + M) x[t] X[t] + (m + M) X[t]^2 + M y[t]^2) }
```

The result is a vector containing the symbolic forms of the accelerations associated with the coordinates x, y, X respectively.

The critical gain

The initial vertical acceleration of the pendulum mass m is

$$\frac{2 (g (m + M) - k L M \cos[\theta]) \sin[\theta]^2}{-m - 2 M + m \cos[2 \theta]} \quad (11)$$

which can be computed by evaluating the vertical acceleration in the initial configuration using

```
Simplify[acceleration[[2]] /.
  {U[t] -> u[t], v[t] -> 0, x[t] -> X[t] + L Sin[θ], y[t] -> L Cos[θ]}]
```

$$\frac{2 (g (m + M) - k L M \cos[\theta]) \sin[\theta]^2}{-m - 2 M + m \cos[2 \theta]}$$

If this is positive, we would expect that the initial motion of the mass is upwards, towards the vertical position. If this is negative, we would expect that the initial motion is downwards, so that the upward position will be unstable.

Thus we expect that the limit of stability occurs when this is zero, which can be regarded as an equation for the critical value of the gain k. This value can be found by setting the initial acceleration to zero and solving the resulting equation for k, noting that the denominator is negative. The result is

$$k_{\text{crit}} = \frac{g (m + M) \sec[\theta]}{L M} \quad (12)$$

which can be obtained by executing

```
kcrit = Flatten[Solve[Simplify[acceleration[2]] /.
  {U[t] -> u[t], v[t] -> 0, x[t] -> X[t] + L Sin[θ], y[t] -> L Cos[θ]}] == 0,
  k]
```

$$\left\{ k \rightarrow \frac{g (m + M) \sec[\theta]}{L M} \right\}$$

We have been unable to find a reference to this formula, which is a generalisation of the known expression

$$k_{\text{crit}} > \frac{g (m + M)}{L M} \quad (13)$$

valid for infinitesimal initial displacements.

Numerical simulation for moderate (10 degree) angular displacements

Assignment of parameter values

The superstable case

The unstable case

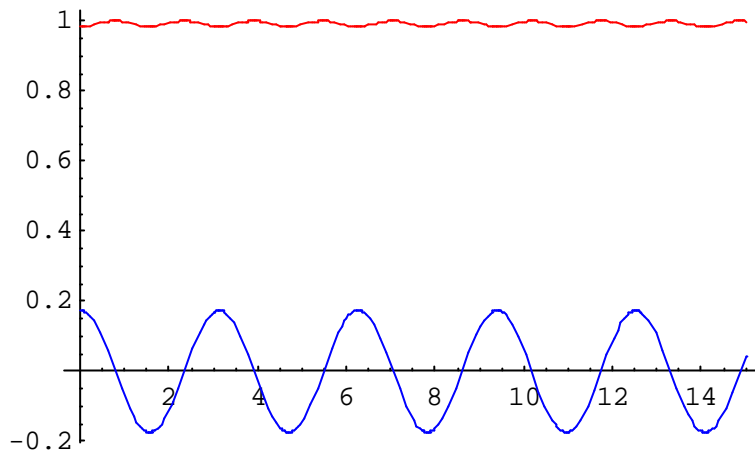
Effect of increasing the gain beyond the critical value

It might be interesting to try to reduce the horizontal motion of the cart in the stable case, by setting the value of k considerably higher (15) than the critical value, using

```
kval = {k -> 15}
```

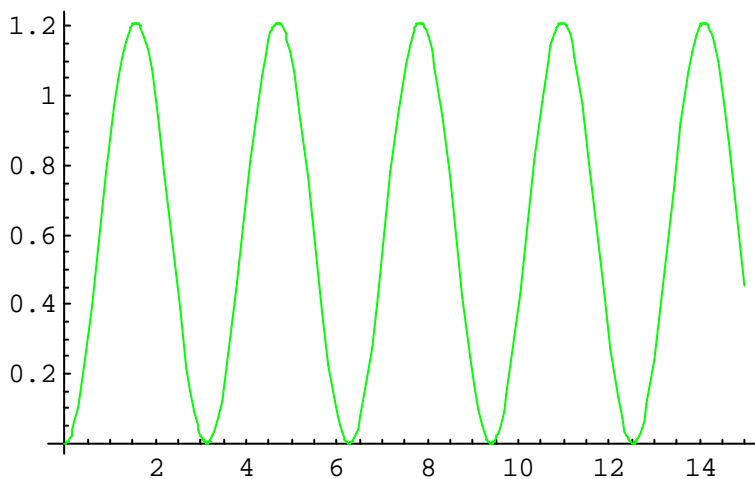
```
{k -> 15}
```

The vertical and horizontal displacements of the pendulum relative to the cart are shown below



We see that the frequency of the relative movement has increased.

The motion of the cart is shown below



We see the cart only moves back and forward about 1.2 metres. We choose this as the case to animate.

Numerical simulation for large (45 degree) angular displacements

Numerical solution of the equations of motion

For the current set of parameter values, the numerical solution and its visualisation can be generated by executing the commands in this section

Input to NDSolve

The numerical solution

Visualisation of the solution

Monitoring and correcting constraint satisfaction

Animation of the simulation

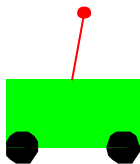
Graphics components for visualising the system

Generating the animation

Viewing the animation of a solution

The animation frames can be generated by executing

```
ShowAnimation[%]
```



To see the animation, double click on the graphic above. To stop the animation, click once on it