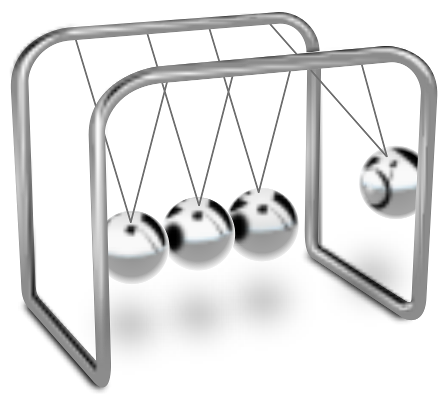
The Step Handbook

Anne-Marie Mahfouf



Contents

Abstract

Step is an interactive physical simulator. It allows you to explore the physical world through simulations. It works like this: you place some bodies on the scene, add some forces such as gravity or springs, then press the **Simulate** icon in the toolbar and Step shows you how your scene will evolve according to the laws of physics. You can change every property of the bodies/forces in your experiment (even during simulation) and see how this will change evolution of the experiment. With Step you cannot only learn but feel how physics works!

Chapter 1

Introduction

Step is an interactive physical simulator.

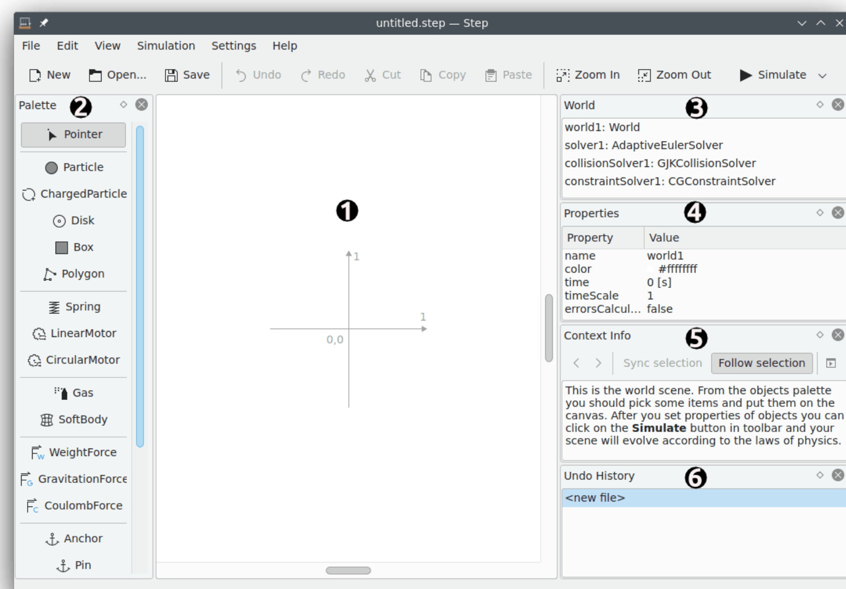
Step features:

- Classical mechanical simulation in two dimensions
- Particles, springs with damping, gravitational and coulomb forces
- Rigid bodies
- Collision detection (currently only discrete) and handling
- Soft (deformable) bodies simulated as user-editable particles-springs system, sound waves
- Molecular dynamics (currently using Lennard-Jones potential): gas and liquid, condensation and evaporation, calculation of macroscopic quantities and their variances
- Units conversion and expression calculation: you can enter something like '(2 days + 3 hours) * 80 km/h' and it will be accepted as distance value (requires libqalculate)
- Errors calculation and propagation: you can enter values like '1.3 ± 0.2' for any property and errors for all dependent properties will be calculated using statistical formulas
- Solver error estimation: errors introduced by the solver is calculated and added to user-entered errors
- Several different solvers: up to 8th order, explicit and implicit, with or without adaptive timestep (most of the solvers require GSL library)
- Controller tool to easily control properties during simulation (even with custom keyboard shortcuts)
- Tools to visualize results: graph, meter, tracer
- Context information for all objects, integrated wikipedia browser
- Collection of example experiments, more can be downloaded with KNewStuff3
- Integrated tutorials

Chapter 2

Using Step

Step simulates a physical world. The main part of Step (1) is the world scene in the center of Step main window where you first place physical objects and where you see the simulation. On the left of this scene a palette (2) let you choose your physical objects. You can freely move this palette anywhere on your desktop by dragging the title bar. On the right of the scene you can see the current world description (3), its properties (4), some help to explain some words (5) and the history of the current world (6). Each of those panels can be placed elsewhere on your screen by dragging the title bar.



To help you get started, Step integrates a series of tutorials which easily teach you how to build an experiment. Please see step by step to start with the first tutorial.

Chapter 3

Getting familiar with Step: the tutorials

The **File** → **Open Tutorial...** menu item brings you a file dialog where you can load Step built-in tutorials. There are five tutorials and you will progressively learn how to interact with each of Step element. The best is to start with the first tutorial by clicking on the file `tutorial1.step`. This will display Tutorial 1 in Step.

> NOTE
> If you do not see the tutorial properly you can try to zoom in to display it better.

The **World** panel on the right lists all the objects you have on your scene. By clicking on an object here, the **Properties** panel below displays this object properties. You can change the properties here by clicking on the one you want to modify.

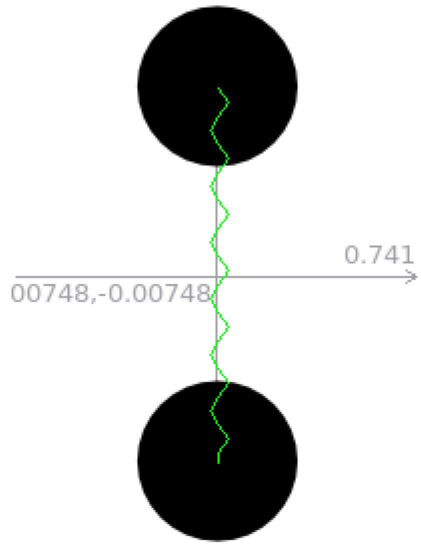
Each tutorial consists in some text presenting the new elements and explaining their properties. Then you are asked to change some properties of the elements in order to achieve a new result of the experiment.

3.1 Tutorial 1: Bodies and springs

This tutorial presents you bodies and springs and how to start your first simulation.

A physical body or body for short is an object which can be described by the theories of classical mechanics, or quantum mechanics, and experimented upon with physical instruments. This includes the determination of position, and in some cases the orientation in space, as well as means to change these, by exerting forces.

A spring is a flexible elastic object used to store mechanical energy.



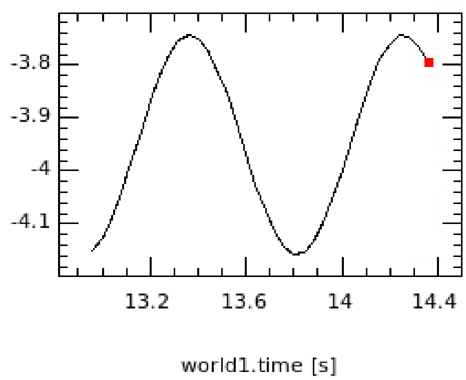
The physical experiment in this tutorial represents two disks linked by a spring. Disks have an initial velocity in a tangential direction (the little blue arrow) and an acceleration (the red arrow) and springs have a stiffness and the length can be changed. Running the experiment you can see the disks being pulled and pushed by the spring. The tutorial invites you to modify the spring stiffness and also to try to change the system experiment.

At the end of this tutorial you should be more familiar with Step interface and you should also be able to easily change bodies properties.

3.2 Tutorial 2: Controllers and graphs

You will learn more about controllers and graphs in this tutorial.

A controller is a device which allows you to graphically modify a property of a body or a spring. In the tutorial, the controller allows you to change the stiffness of the spring ‘spring1’. By moving the slider to the right or using the **W** key you can increase spring1 stiffness value and by moving the slider to the left or using the **Q** key you can decrease it. Right-clicking on the controller brings you several context actions and the **Configure Controller...** item shows a dialog allows you to change each property of the controller.



Graphs allow you to graphically visualize the relationship between two variables. The example in the tutorial prints the evolution of the position of particle1 while time advances in world1. With a right click on a graph you can clear or delete the graph as well as edit the configuration dialog and change here all the properties for this graph.

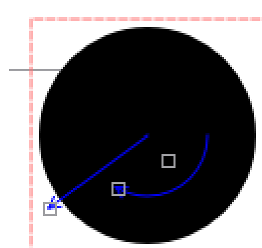
At the end of this tutorial you are able to use controllers to act on your bodies properties and graphs to monitor specific properties in your experiment.

3.3 Tutorial 3: Rigid bodies and tracers

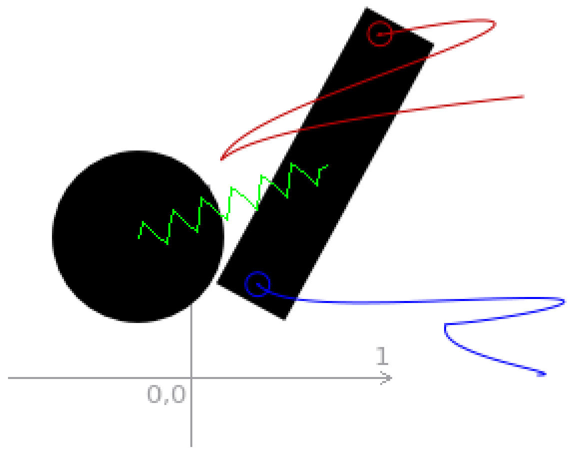
Tutorial 3 presents you rigid bodies and tracers.

A rigid body is an idealization of a solid body of finite size in which deformation is neglected. In other words, the distance between any two given points of a rigid body remains constant in time regardless of external forces exerted on it.

A tracer is a tool which shows the trajectory of a given point on a rigid body.



When a rigid body (here a disk) is selected you see three grey handlers on it. Using them by clicking on them and moving them, you can change the velocity, the angle and the angular velocity of the body.

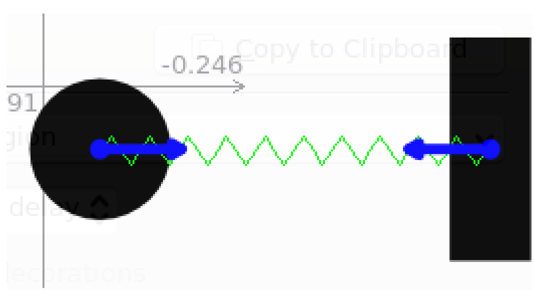


The experiment in Tutorial 3 shows a disk and a box linked by a spring. A tracer (the blue one) is already on the box. You can add a second one: select the **Tracer** button in the **Palette** panel then click on the box on the point where you want the tracer to be. In the **Properties** panel, click on the **color** line and on the right of this line you can click on the blue square and a color palette appears: you can choose a new color for the tracer. The screenshot above shows two tracers after the simulation is run for a few seconds.

3.4 Tutorial 4: Motors and forces

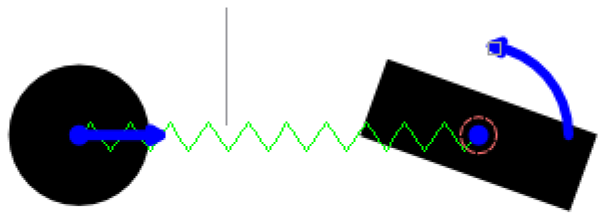
You have two sorts of motors available in Step: linear motors and circular motors. A linear motor applies a constant force to a given point on a body while a circular motor applies a constant angular momentum to a body.

Three different forces can be added to bodies: the weight force, the gravitation force and the Coulomb force. By default all forces are turned off in Step. Coulomb force is a force which existed intrinsically between two charges.



In the experiment you have a disk and a box linked by a spring. A flat box at the bottom will make a boundary. The disk and the box both have a linear motor applied to them. Two controllers allow you to change the force value of each motor. Start the simulation and play with the controllers. Then stop the simulation and add a weight force in the world (forces are global and apply to the whole world). Restart the simulation and analyze the difference.

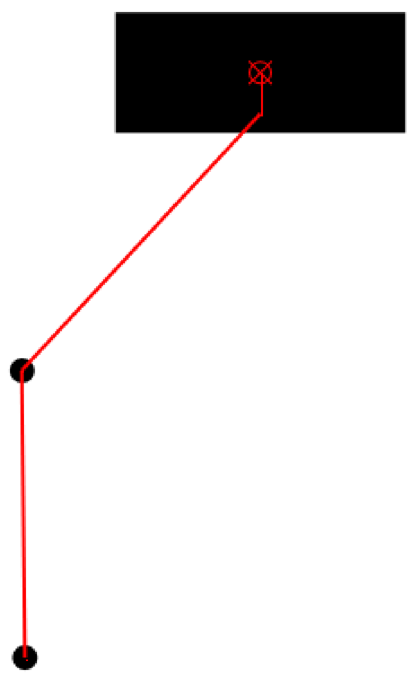
You can also remove the linear motor on the box and add a circular motor instead. Press on the  **CircularMotor** button in the **Palette** panel and then click on the box. The circular motor is applied to the box. You then need to set the torque value by clicking and moving the grey handler of the motor.



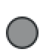
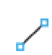
This tutorial introduced you with motors and forces and you should now be able to add those to bodies.

3.5 Tutorial 5: Joints

Joints are objects that attaches bodies to each other or to the background. You have a the following joints in Step: anchors, pins and sticks. An anchor is a joint that fixes position of the body. The body cannot move when it has an anchor. A pin is a joint that fixes one point of the body, the body can still move around the pin. A stick is a joint that fixes the distance between two points on two bodies.



Tutorial 5 describes a double pendulum.

Add a  **Particle** to the scene then join this particle to Particle2 with a stick. Press the  **Stick** button on the **Palette** panel. You then need to select the first object attached to the stick (particle2) with the left mouse button then drag the mouse to the second object (particle3) and release the mouse button on particle3. You now have a triple pendulum!

Chapter 4

Step examples

Step package contains several instructive examples to help you understand the principles of the application work:

File → Examples

Opens a submenu with different action items.

Open Example...

Opens an example from the default set

Open Downloaded Example...

Opens the downloaded examples.

Download New Experiments...

Download examples shared by other users.

Share Current Experiment...

You can share your own examples.

You can find the descriptions of the default example files below.

`brownian.step`

Plots trajectory of the rigid disk interacting with 40 particles that randomly drifting in a box. This example simulates Brownian motion of ideal gas particles.

`doublependulum.step`

This example simulates double pendulum motion using 2 massive particles and two sticks.

`eightpendulum.step`

This example is a simple demonstration of the famous Newton's cradle. It is done in Step using sticks, 8 discs and a box. The six balls in the middle are not moving because they just transfer momentum and energy, not a motion.

`first.step`: First example

This example has two parts. The first part contains two particles connected by a spring and the second part contains two charged particles.

Two particles connected by a spring

In this example two particles are added to the scene and spring is connected between them. The properties of both the particles such as velocity, momentum, position etc. has been set in the properties browser. The properties of the spring such as stiffness, restLength, damping etc. also has been set in the properties browser.

Explanation of the simulation:

This is good example of a simple harmonic motion. Here the acceleration of the one particle is set in direction of positive x-axis and the acceleration of the other particle is set along negative x-axis. As a result both the particles pulls the spring in opposite directions, where as spring tries to bring the two particles back to their original positions. Thus the system executes simple harmonic motion. The simulation of the particles and spring under these conditions can be seen on the scene.

Two charged particles

Velocity of the each charged particle is set in some direction so, the charged particles moves in respective direction of their velocity but each particle has been given a equal and opposite charge so the particles try to attract each other. As a result the simulation of the charged particles under these conditions can be seen on the scene.

`fourpendula.step`

This example is a correct demonstration of the Newton's cradle. As the system is imperfect two disks in the middle get visual movement with time.

`gas.step`

This example simulates ideal gas pressure caused by Brownian motion.

`graph.step`

Plots velocity vs. position graph for particle1 in the system of two particles connected with a spring.

`liquid.step`

This example simulates monoatomic liquid.

`lissajous.step`

This example simulates Lissajous curve using two-particle model. The parameters on the model can be changed using the controller at the center of the world.

`motor1.step`

Simulates triangular rigid body under the loading of the three linear motors.

`motor.step`

Simulates interaction of the linear motor with a rigid rectangular body on a spring.

`note.step`

Example with LaTeX formula (divergence theorem) and embedded image.

`resonance.step`

This example simulates resonance in the system with angular motor.

`softbody.step`

This example simulates interaction of two rigid bodies with a soft body between them.

`solar.step`

This example simulates the motion of Solar system major bodies (Sun and the planets).

`springs.step`

This example simulates the motion of the planar system of five particles connected with four springs.

`wave.step`

The graph on the scene shows oscillations of the green particle. When you start simulation the wave starts to travel from the red particle. The blue particle will reflect the wave and it will travel in reverse direction until the red particle reflects in again. After some time the wave will vanish because springs have damping.

Chapter 5

Credits and License

Step

Contributors:

- Author: Vladimir Kuznetsov ks.vladimir@gmail.com
- Contributor: Carsten Niehaus cniehaus@kde.org