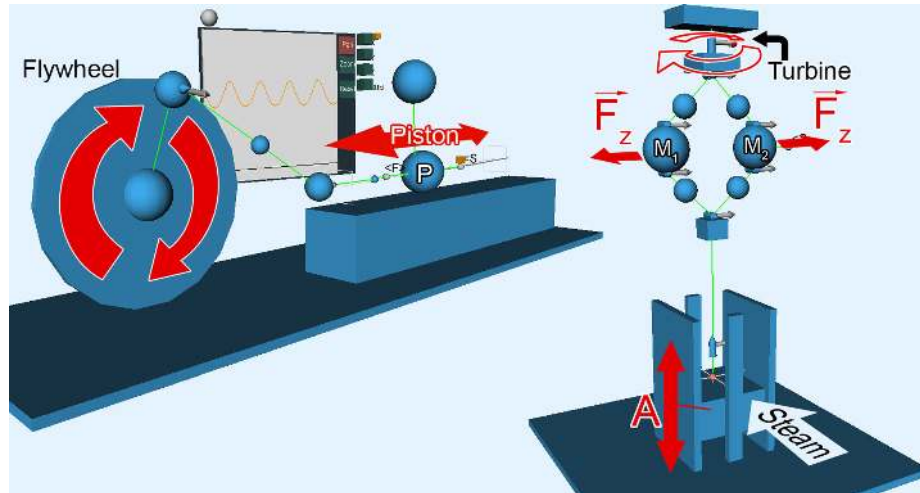


# Simulating Educational Physical Experiments in Augmented Reality

Hannes Kaufmann\*

Bernd Meyer†

Interactive Media Systems Group  
Institute of Software Technology and Interactive Systems  
Vienna University of Technology



**Figure 1:** Left: An experiment simulating the motion of a car crankshaft. Right: A centrifugal regulator in PhysicsPlayground.

## Abstract

We present an augmented reality application for mechanics education. It utilizes a recent physics engine developed for the PC gaming market to simulate physical experiments in the domain of mechanics in real time. Students are enabled to actively build own experiments and study them in a three-dimensional virtual world. A variety of tools are provided to analyze forces, mass, paths and other properties of objects before, during and after experiments. Innovative teaching content is presented that exploits the strengths of our immersive virtual environment. PhysicsPlayground serves as an example of how current technologies can be combined to deliver a new quality in physics education.

**CR Categories:** K.3.1 [Computers and Education]: Computer Uses in Education—Collaborative learning; H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Artificial, augmented, and virtual realities

**Keywords:** physics education, mechanics, augmented reality, virtual reality

## 1 Motivation

Classical mechanics [Goldstein et al. 2001; Lifshitz and Landau 1982] is the oldest discipline in the field of physics. It describes the common motion of objects that humans perceive in everyday life. The three fundamental laws of motion which were formulated by Isaac Newton (1642 - 1727) are still of high importance and concepts such as force, velocity and acceleration are traditionally

taught in schools. Our knowledge about physics changed throughout centuries but most books on classical physics begin with a chapter on mechanics [Gerthsen and Vogel 1993; Körner et al. 1988; Pohl 1931].

Sometimes students have problems understanding the physical concepts of mechanics. It might be that physics in the traditional sense is sometimes taught in an abstract, jejune way and is therefore not very motivating for students. The result is that theoretical models behind physical phenomena are often misunderstood. It is not necessary to stress that conveying these concepts correctly is of utmost importance since they are fundamental to physics. Many theoretical models are based on Newton's laws of motion.

Therefore the authors developed an educational augmented reality (AR) application called PhysicsPlayground [Meyer 2007] that is supposed to support students in studying and finally understanding the concepts of mechanics (Figure 2). In this three-dimensional virtual environment learners and educators are able to freely create physical experiments that can be simulated in real time. The hardware setup is described in section 3. Features of the application are explained in section 4. We consider the analyzing functionality (section 4.3) an important strength of a virtual laboratory like PhysicsPlayground. It offers possibilities that are far superior to what can be done in a real physics lab. A direct connection between simulated reality and physical data is supposed to help students grasp the theoretical basics of mechanics. To establish a direct link to students pre-knowledge, physical data - that can be acquired through the application - is presented in a way so that it closely relates to formulas and equations of school mechanics. In section 5 we present teaching content that demonstrates the power and added benefit of our educational AR/VR software. We will also elaborate on correctness and robustness of physical simulations for educational purposes.

\*e-mail: kaufmann@ims.tuwien.ac.at

†e-mail: meyer@ims.tuwien.ac.at



**Figure 2:** Left: A student working with *PhysicsPlayground*. In the right hand he holds a wireless pen (used as input device), in the left hand the PIP. Right: Collaborative work in *PhysicsPlayground*.

## 2 Related Work

In the following a few immersive virtual environments for mathematics and science education will be presented briefly. They all demonstrate unique advantages of using AR/VR for education and give insights to what the technology can offer.

### 2.1 Mathematics and Science Education in AR/VR

Water on Tap [Byrne 1996] is one of the earliest immersive virtual environments for chemistry education. It is a chemistry world which allows to build molecules. Therefore electrons have to be placed in orbits around the kernel of an atom. The spin of the electrons and other properties can be selected. ScienceSpace [Dede et al. 1996] is a collection of immersive virtual worlds consisting of NewtonWorld, MaxwellWorld and PaulingWorld. They have been developed to study the strengths and limits of virtual reality for science education. NewtonWorld provides an environment for investigating kinematics and dynamics of one-dimensional motion. MaxwellWorld supports the exploration of electrostatics, up to the concept of Gauss' Law, and PaulingWorld enables the study of molecular structures via a variety of representations. Formative evaluation studies of these virtual worlds have been conducted with respect to usability and learn-ability. These studies report on learners' engagement, surprise and understanding. Limitations and discomfort caused by the head-mounted displays hindered usability and learning.

A technically advanced project for mathematics education is CyberMath [Taxen and Naeve 2001]. CyberMath is an avatar-based shared virtual environment aimed at improving mathematics education. It is suitable for exploring and teaching mathematics in situations where both teacher and students are co-present or physically separated. CyberMath is built like a museum with a virtual lecture hall in its center. Special care has been taken to design the environment as inviting as possible. Virtual mathematical objects can be manipulated and discussed in a realistic way. CyberMath has been tested for distributed learning in CAVEs but is also running as a desktop VR application with no support of immersive displays. Two usability studies of the DIVE version of CyberMath have been performed with 15 participants in total. Teacher and students worked in two separate locations. The studies provided useful feedback for further improvement of the application and for increasing robustness of the distributed environment. The developers

believe that CyberMath in a networked CAVE environment holds the potential to provide a high-tech front end which is interesting enough to create public interest and contribute to a more positive attitude towards mathematics - especially among young people. It could also provide a useful platform for developing various forms of interactive problem solving games with an emphasis on cooperative problem solving skills.

Construct3D [Kaufmann and Schmalstieg 2003] is a three-dimensional dynamic geometry construction tool that can be used in high school and university education. It uses augmented reality to provide a natural setting for face-to-face collaboration of teachers and students. The main advantage of using VR and AR is that students actually see three dimensional objects which they until now had to calculate and construct with traditional (mostly pen and paper) methods. By working directly in 3D space, complex spatial problems and spatial relationships may be comprehended better and faster than with traditional methods. Three usability studies with more than 100 students have been conducted since 2000 [Kaufmann and Dünser 2007] and guidelines have been formulated regarding how to design AR applications for (geometry) education [Kaufmann and Schmalstieg 2006]. Although usability of Construct3D is high and teachers as well as students are highly motivated to use the application, practical usage in schools is hindered by hardware costs, support of a low number of users and technical complexity of the whole setup (requiring dedicated personnel for maintenance).

The SMILE project [Adamo-Villani et al. 2006; Adamo-Villani and Wright 2007] is an immersive learning game that employs a fantasy 3D virtual environment to engage deaf and hearing children in math and science-based educational tasks. SMILE is one of the first bilingual immersive virtual learning environments for deaf and hearing students combining key elements of successful computer games, emotionally appealing graphics, and realistic real-time 3D signing, with goal-oriented, standards-based learning activities that are grounded in research on effective pedagogy.

### 2.2 Pedagogic Background

Constructivist theory provides a valid and reliable basis for a theory of learning in virtual environments [Winn 1993]. Constructionism is based on constructivism and promotes that learning takes place when students can construct things. As Mantovani [2003] points out, the basic assumption that the learning process will take place naturally through the simple exploration and discovery of the virtual environment should be reviewed. Despite the value of exploratory learning, when the knowledge context is too unstructured, the learning process can become difficult. The learning process should support building of conceptual models that are both consistent with what students already understand and with new content. In order to ensure successful adaptation of old knowledge to new experience, flexible learning environments should be provided. One possibility is to integrate known types of information other than a 3D representation (such as audio and text annotations, images etc.). In our case we included the analyzer (section 4.3) as a tool similar to an oscilloscope. In addition our environment supports collaboration and therefore learning as an active, social process. Finally, VR environments can be tailored to individual learning and performance styles. Our examples in section 5 allow experimentation and support constructionism.

## 3 Working Environment

The implementation of *PhysicsPlayground* is based on the Studierstube AR framework [Schmalstieg et al. 2002]. The standard hardware setup consist of an head-mounted display (HMD), a wireless pen and the so called personal interaction panel (PIP) [Szalavari

and Gervautz 1997]. Pen and PIP are used to fully control the application in 3D space (Figure 3). The overall hardware setup is equivalent to the one used in Construct3D [Kaufmann and Schmalstieg 2003], an augmented reality application for geometry education, since both applications were developed in the same lab. The setup supports two users, allows direct manipulation, free roaming around virtual objects and is favored by students. Since we're using Sony Glasstron see-through head mounted displays students can see each other and their interaction devices. Each of the hardware components is tracked by an iotracker [Pintaric and Kaufmann 2007] infrared-optical tracking system in six degrees of freedom. This gives the user freedom in motion some students lie down on the floor to look at objects from below or step on a chair - and simultaneously enhances the feeling of immersion and fun. PhysicsPlayground runs on a standard desktop setup as well but is more intuitive and more impressive to use in an immersive environment.



**Figure 3:** Each user is equipped with input devices pen and PIP and a head mounted display when interacting with PhysicsPlayground.

A wireless pen with one clickable button is used to select and drag objects and to control the application. Most of the application functionality can be invoked through the PIP, a touchable plexiglass sheet providing haptic feedback. When looking through the head mounted display control elements are displayed on the PIP which the user can click in order to trigger different actions (Figure 4). We intentionally designed it to be similar to a 2D GUI in order to provide students with a familiar type of interface. Studying usability of educational AR applications using Construct3D, the main author showed [Kaufmann and Dünser 2007] that a 2D type of menu interface on the PIP is perceived as highly usable by students, provided that some basic guidelines are followed [Kaufmann and Schmalstieg 2006].

In Figure 4 several menu widgets are shown. Many of them have 3D icons placed on top which are animated when moving the pen over them. This is self-explanatory and clarifies their specific functionality. The GUI is a standalone component which can flexibly be used by other VR applications. An additional layout manager system like it is standard for 2D GUIs is not available at present.

## 4 Features and Design

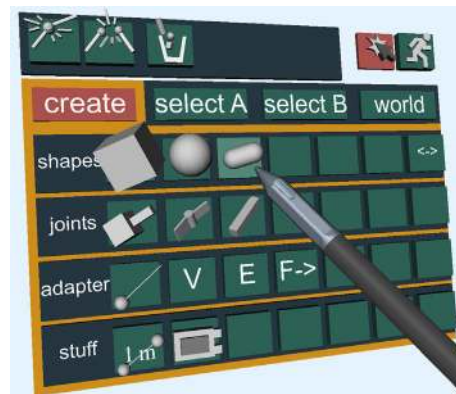
In [Seugling and Rölin 2006] a number of widely used physics engines have been evaluated in detail. Amongst them was the

commercial engine PhysX formerly known as NovodeX/AGEIA, now NVIDIA [AGEIA 2008] and the open source engines Newton Game Dynamics and Open Dynamics Engine. In summary the evaluation showed that PhysX is the most stable, feature rich, precise and fastest engine available at the moment. Therefore the PhysX API was chosen as the base of PhysicsPlayground. PhysX (now owned by NVIDIA) is available for free for commercial and non-commercial purposes, supports Windows and Linux and the PS3 platform and is written in C++. Objects which can be used by PhysX in simulations are rigid bodies, soft bodies, deformable objects, fluids and cloth. Related to PhysicsPlayground robust simulation of rigid bodies is of highest importance. A rigid body in its original definition is an object with fixed geometrical characteristics [Hecker 1996]. At each point in time points within the rigid body stay fixed to each other. Additional dynamics provide the rigid body with the ability of translational and rotational motion. In PhysX rigid bodies and rigid body dynamics are referred to as shapes and actors. In the following the features of PhysicsPlayground will be described briefly.

### 4.1 3D Shapes

Because the application is intended to simulate physical school experiments it must be possible to integrate virtual models of real life objects. Therefore PhysicsPlayground allows users to create, destroy, modify and interact with different kinds of shapes. Each shape can be either static or dynamic and represents a solid object, enabled for collision detection during simulation. Static shapes stay in place while the physical simulation is running. They have an infinite mass. Dynamic shapes behave like real world objects. They have an adjustable mass, a center of mass, a surface friction and are affected by force during simulation.

The appearance of shapes can have various forms. Shapes can be simple objects like boxes, spheres, cylinders or more complex ones such as a looping (Figure 2 left) or a car. A number of primitive shapes are integrated into PhysicsPlayground by default. More complex objects can be defined by the user or loaded on demand. Position, appearance and parameters of all shapes, for example the width of a box, can be configured by the user after object creation. Modification takes place through the PIP or by direct manipulation of the shapes with the pen. Finally shapes can be grouped into larger shapes making it possible to create advanced objects.



**Figure 4:** An example menu of PhysicsPlayground on the PIP. The virtual representation of the pen is shown as well.

## 4.2 Joints

Rather than simply grouping shapes our application allows the definition of mechanical linkages between different shapes (Figure 4). Currently implemented are a revolute joint, a prismatic joint and a stiff joint connection. In case of a revolute joint a motor can be added. It causes its attached shape to spin with constant radial velocity. With such connections at hand it is possible to create more complex physical scenes. A centrifugal regulator for example can be built and simulated in PhysicsPlayground this way (e.g. Figure 1 right). Modification and positioning of each joint is either done utilizing the PIP or directly by selecting a joint in 3D space.

## 4.3 The Analyzer

During every physical experiment certain magnitudes and properties of the simulated objects might be of interest. To be able to compare a virtual simulation with a real one it is important to offer possibilities to extract physical properties of objects before, during and after run time of the simulation. For educational purposes these data can be used to learn about the theoretical background or to confirm results in a traditional way for instance by using appropriate formulas. Physical magnitudes with a high interest of analyzability are speed, acceleration, force, friction, energy and path. Two mechanisms are provided to output such data:

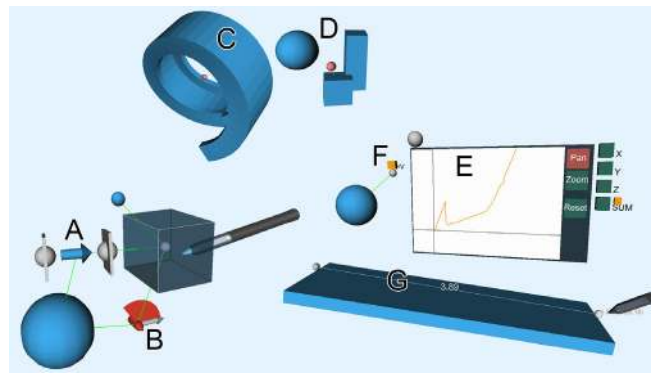
- A simple measurement tool to measure distances and
- A powerful analyzer which can extract and visualize all interesting, relevant physical magnitudes from shapes (Figure 5 right).

As an archetype for the analyzer we used an oscilloscope. An oscilloscope has two axes, a time axis and a voltage axis. Both axes are equivalently used in our application. On the value axis we can visualize components of 3D vectors in addition to absolute values, like voltage for example. This is needed because magnitudes like velocity or path have three dimensions plus a value. Therefore our analyzer has four input fields - x, y, z and an absolute value.

The input fields can be connected to different shape attributes, like speed or kinetic energy. The attributes are represented through so called *adapters*. Velocity, energy and a path adapter are supported. Adapters have to be added to each shape manually. An added adapter interfaces a specific physical magnitude and via the adapter data can be plugged into the analyzer. During simulation the adapter will continually send its values to the analyzer and the accordant time-value function is drawn in real time. This allows students to study all data during the running experiment. Moreover adapters do not only act as interfaces to the analyzer but can visualize their values. For example a path-adapter (which is typically attached to an object) records and subsequently draws its trajectory whereas a velocity-adapter splits and visualizes the absolute velocity vector in its x, y and z direction.

## 4.4 Force Adapter

Besides all the adapters with analyzing capabilities we added a force adapter. With the force adapter a directed and dynamically changeable force can be put onto a shape to affect its motion. This enables many new physical experiments: For example imagine a simulation of precisely accelerating and slowing down a car. Configuration of the force adapter is done through the PIP. When selecting the adapter a force-time function can be defined using a built-in key frame editor. The function is finally processed during simulation. This functionality is demonstrated in the next section in the experiment Teaching Lesson: Speed and Velocity.



**Figure 5:** The PhysicsPlayground elements and their visualizations in VR.

In Figure 5 the above mentioned building blocks of PhysicsPlayground are shown. *A* and *B* represent mechanical linkages. Both connect the same shapes, a box and a sphere. The revolute joint *B* restricts the rotational motion between both bodies along the axis in which the red arrow indicates. The prismatic joint *A* restricts the translational motion between both bodies along the axis of the blue arrow. *C* is a more complex shape, a loop-the-loop. *D* visualizes grouped shapes which act together during simulation. *E* denotes the analyzer which is connected to the velocity adapter *F* of a nearby sphere. During simulation the velocity value of *F* is recorded. The orange curve on the analyzer results from an earlier run of the experiment where the sphere bounced onto the underlying plane and continued to roll down. The recorded time-velocity values can be read exactly by just moving the pen over the orange line. They are output in textual form next to the tip of the virtual pen. *G* demonstrates the measurement tool. Currently the length of a box is measured.

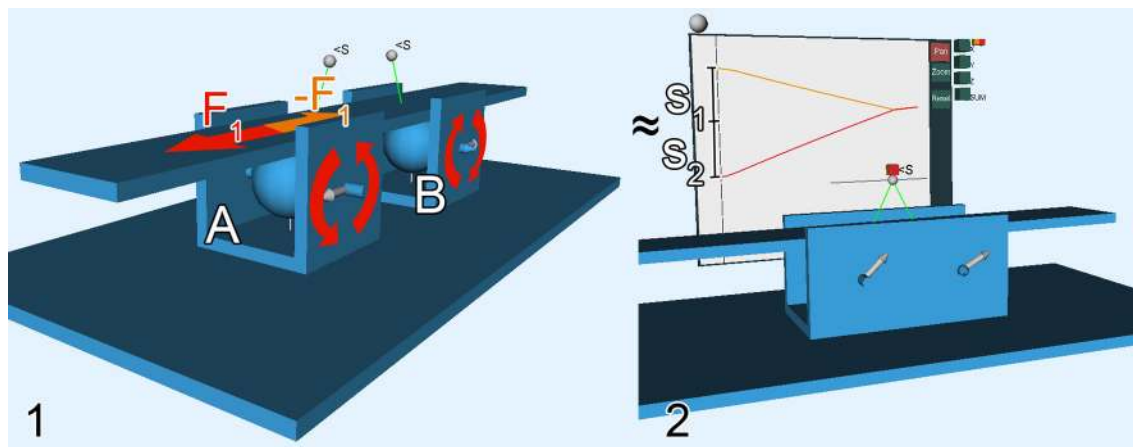
## 5 Educational Use Cases

In the following the applicability of PhysicsPlayground for school experiments and school exercises will be demonstrated by two selected examples that exploit the strengths of our immersive virtual simulation tool. These examples are targeted to high school students aged 12-18. Depending on the curriculum of the specific physics course Newton's laws are taught at different grade levels. We think that PhysicsPlayground can also be utilized for students at a younger age as well as for basic university/college courses.

### 5.1 Force and Counterforce

The effect of force and counterforce is described by the third law of motion. It states that every force which is invoked by a body *A* on a body *B* leads to a counterforce into the opposite direction with the same absolute value. An experiment that we use to demonstrate force and counterforce is described in [Pohl 1931]. Thereby two persons stand face to face each on their own frictionless moving carriage (Figure 5). At the same time both persons hold a rope which connects them. With this setup a series of tests can be conducted:

1. Both participants pull the rope at the same time.
2. Only the left person pulls the rope whereas the right person holds the rope.
3. The same as before but now only the right person pulls on the rope.



**Figure 6:** Demonstration of force and counterforce by using two frictionless slidable boxes.

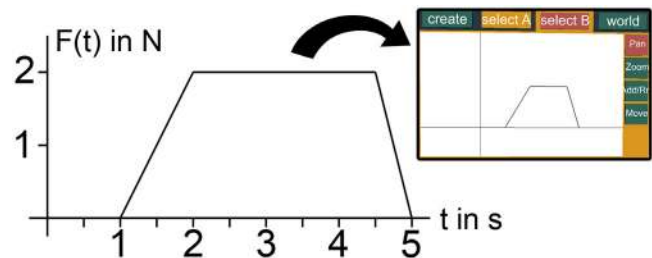
Independent of which participant pulls the rope both wagons will move the same distance from start to the point where both collide in the middle. This is a result of force and counterforce.

In our simulation of the experiment the wagons (where the persons stand on) are replaced by two identical, frictionless slidable box frames *A* and *B*. Each of the boxes contains a rotatable sphere in the middle which is mounted on the box frame via two revolute joints. These spheres represent the two persons. The rope is then realized through a bar which lies on top of both spheres. Additionally bar and sphere must have a maximized friction coefficient of 1 so that everything works out right. Next we want the participants to pull the rope during simulation. Therefore a motor has to be attached to one revolute joint of each box frame. On body *A* this motor has to spin counter-clockwise whereas on body *B* it has to turn clockwise. When one of the motors is activated during simulation, it will put a predefined torque along the rotational axis of its attached sphere. This will put the sphere into a rotational motion. The sphere again will try to transfer this motion to the bar at the point where the bar touches the sphere. This leads to a force  $F_1$  which affects the sphere tangentially opposed to its rotational direction. In the other direction it invokes a counterforce  $-F_1$  on the bar which has the same absolute value as force  $F_1$ . No matter if one or both motors are activated, the person who watches the simulation will notice that both box frames will always meet in the middle (the mid position in between their starting positions). Additionally this can be proven in PhysicsPlayground by analyzing the path of each box frame through a trajectory adapter. During simulation the analyzer will record a similar curve like shown in Figure 6 right. Students can read the distances  $s_1$  and  $s_2$  from the analyzer. Regardless of which person pulls the rope both carriages always move equivalently.

## 5.2 Teaching Lesson: Speed and Velocity

This example demonstrates how PhysicsPlayground can be integrated into a traditional physics lesson. An appropriate physical exercise from [Stark 2002] was chosen. It was part of an actual high school final physics exam. In this exercise two bodies *A* and *B* slide on top of a plane along the global  $x$ -axis with different speeds. At some point in time both bodies collide. A number of physical scenarios can be studied with this setup. We will recreate the scenario which relates most to the school task [Stark 2002] and extend it in the following. The starting setup in PhysicsPlayground contains a plane and two bodies *A* and *B* as shown in step 1 of Figure 7.

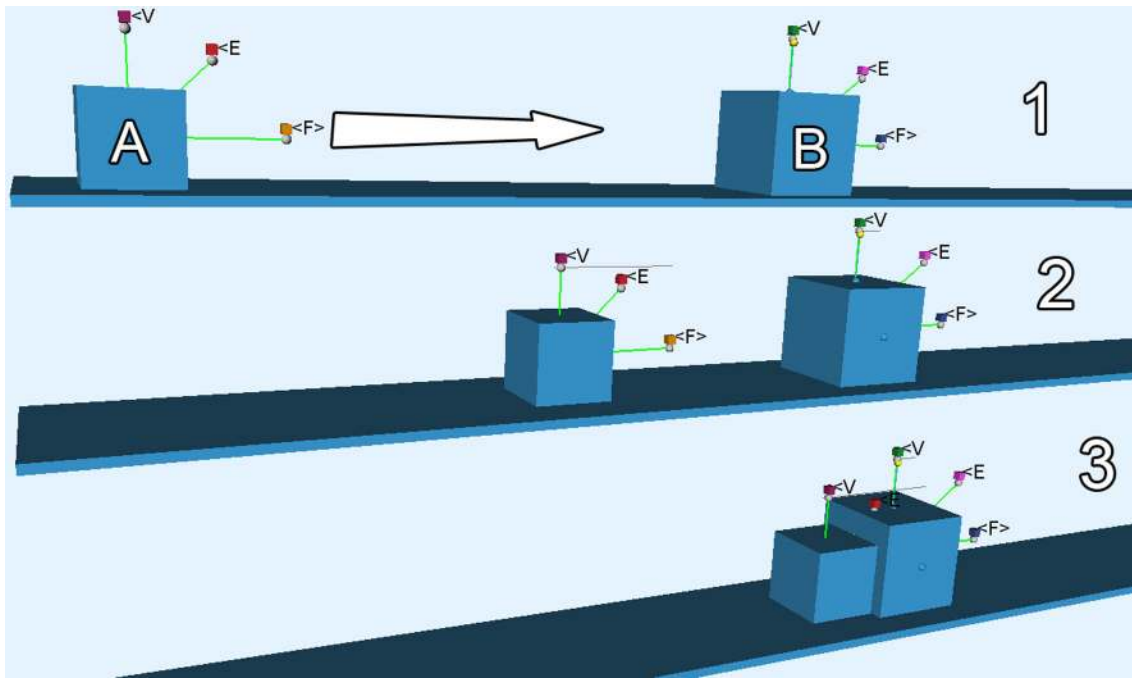
The task description for students is the following: A toy car which is represented by body *A* is accelerated along the positive  $x$ -axis. Starting at its standing position it is accelerated by a time dependent force  $F_A(t)$  which is given in Figure 8. The friction of all shapes is zero. Use PhysicsPlayground to acquire the final speed  $v_{end}$  of the toy car *A* and check your result by calculating it manually.



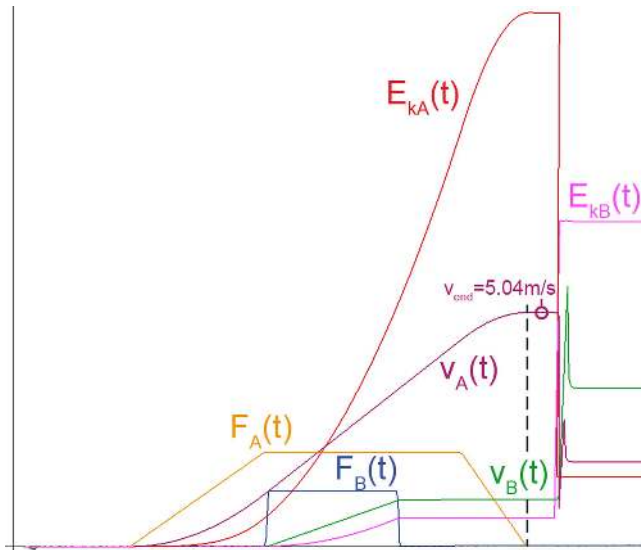
**Figure 8:** The given time force-function (left) and force adapter conversation (right).

In order to solve this task with the help of PhysicsPlayground, students have to transfer the time-force function to PhysicsPlayground first. Basically this is done by adding a force adapter which points towards the positive  $x$ -axis to the toy car. The adapter is displayed in Figure 7, attached to object *A* (orange dot denoted  $\langle F \rangle$ ). In the next step the key-frames of the adapter have to be adjusted so that they represent the time-force function as can be seen in Figure 8. If done correctly the body should start to move along the  $x$ -axis after starting the simulation. In order to check the movement students may create an analyzer and connect the force adapter to the sum-input of the analyzer. If the recording of the analyzer is equal to the time-force function in Figure 8 the previous steps were performed correctly. The next step is to investigate the velocity of the toy car. This is straight forward as we only have to attach a velocity adapter to body *A*. Afterwards the velocity adapter has to be connected with the  $x$ -input of the analyzer to get a recording of the final speed of the toy car on the analyzer. The print out of the analyzer is finally shown in Figure 9.

The only relevant graphs are  $F_A(t)$  and  $v_A(t)$ . In figure 9 there are additional printouts of body *A* and *B* just to demonstrate what can be done with the analyzer (they are not relevant for this task). The graphs labelled *E* stand for kinetic energy. The graph of  $F_A(t)$  represents the time force function which is applied to the toy car during simulation. It is obvious that its analyzer graph is pretty



**Figure 7:** Complete setup for a PhysicsPlayground based school exercise. Steps one to three visualize the simulation progress in PhysicsPlayground.



**Figure 9:** Analyzer output for bodies A and B. The single curves were labeled afterwards.

much the same as in Figure 8. This is exactly how it is supposed to be. The velocity graph  $v_A(t)$  of the velocity adapter derives from  $F_A(t)$ . Final velocity is reached when the force  $F_A$  stops to affect body A. From this point on the black dotted line perpendicular to the abscissa (where  $F_A(t)$  stops affecting A) intersects  $v_A(t)$  at a point from which on  $v_{end}$  is reached. According to the analyzer  $v_{end} = 5.04 \text{ m/s}$ . Since the solution, namely  $v_{end}$  has been found using PhysicsPlayground students can try to calculate it next.

In the solution of Stark [2002] the area below the force function

$F_a = \sum F * \Delta t$  is calculated first. To calculate  $v_{end}$  the area  $F_a$  is used as a parameter in a formula for constant accelerated movement  $v_{end} = F_a/m$  whereby  $m$  is the mass of body A. After determining its mass within PhysicsPlayground we calculate the result  $v_{end}$  and get a final result of  $5.00 \text{ m/s}$ . This is very close to the result of the simulation.

### 5.2.1 Accuracy and Robustness

At this point we also want to discuss the matter of accuracy, precision and robustness of the simulation and the physics engine that is being used. In order to study physical experiments live in an interactive environment all simulations in PhysicsPlayground have to be performed in real time. Therefore a physics engine was chosen that is optimized for the gaming market and is able to perform all required calculations in real time. For optimization purposes this means that some physical properties cannot be calculated exactly but have to be approximated by the physics engine. In contrast to that results of experiments do not only have to look believable and realistic but must also be correct in order to be usable in education. A tool that teaches wrong contents cannot be used in classrooms even if it only produces wrong results in a small percentage of cases. In addition educational tools must be very robust. If crashes or chaotic behavior of the simulation occur frequently students loose motivation quickly.

In [Seugling and Rölin 2006] multiple physics engines have been evaluated and the AGEIA/NVIDIA PhysX engine proved to be the most accurate, most precise and most robust of all 'real-time' engines targeted to the gaming market. Accuracy was compared using physical formulas to calculate properties such as friction, gyroscopic forces, stability of constrains and others. The theoretical values were then compared to the actual behavior of the physics engine. In this comparison AGEIA performed best but for further detail we refer to [Seugling and Rölin 2006]. In our example (in 5.2) the practical result of the experiment with  $v_{end} = 5.04 \text{ m/s}$

is very close to the theoretical value of  $5.00 \text{ m/s}$  and is considered sufficiently accurate for educational purposes. PhysicsPlayground has been designed with the assumption that experiments are explained and guided by a teacher in a classroom setting in any case. In such a setting teachers are supposed to discuss numerical errors of simulated experiments with students. Nowadays students are using a number of educational tools that produce numerical errors – such as numerical calculators, dynamic geometry or CAD packages – and therefore that topic is of general importance.

Apart from the above mentioned exercises and experiments PhysicsPlayground can be used to demonstrate a wide variety of mechanical devices as shown in Figure 1.

### 5.3 Motion Paths

Last but not least the application can also be used to visualize mathematical and geometrical contents such as kinematics (Figure 10). Through an appropriate linkage and rotational forces between three shapes  $A$ ,  $B$  and  $C$ , we can cause shape  $B$  to rotate around the static shape  $A$ , and  $C$  rotate around  $B$ . When the trajectory is visualized through a path adapter on  $C$ , mathematical curves like epicycloids and hypocycloids are generated.

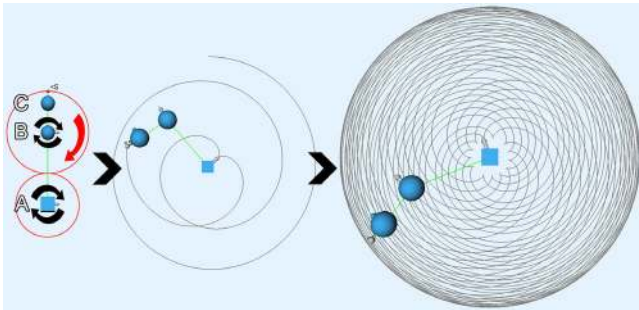


Figure 10: An epicycloid in PhysicsPlayground.

## 6 Conclusion

With PhysicsPlayground an AR application was developed that aids teaching and learning mechanics. The physics engine PhysX on top of the Studierstube framework provides the technical basis and manages all physical calculations. The accuracy of the simulations is good and considered sufficient for educational purposes compared to exact calculations which we used to verify our results.

The educational use cases demonstrate how versatile PhysicsPlayground can be integrated into physics lessons. The potential applications are manifold. Students can build virtual mock-up models of experiments to study physical properties, verify formulas, develop theories and actively participate in physics education in general. It fosters experimentation and understanding.

Up to now PhysicsPlayground was only evaluated by staff members of the Interactive Media Systems Group at Vienna University of Technology. No evaluation about its effectiveness for learning has been conducted yet. It would also be interesting to gather feedback on usability of the application as it uses its own GUI and interaction scheme. Therefore an expert-based usability evaluation including physics teachers and students from different grades would be a meaningful next step.

At trials with PhysicsPlayground we realized that there are two major strengths of the presented educational tool:

- Nearly haptic interaction when building and running physical experiments. Students are able to walk around objects and can view the experiments from different perspectives.
- The possibility to simulate experiments in real time enables quick variation of parameters and reconfiguration of an experiment. It encourages modifications.

In summary PhysicsPlayground is best suited for simulating and solving inherent three-dimensional physical problems that are hard to do in real life.

## Acknowledgments

The authors would like to thank Mathis Csisinko for his support during the development of PhysicsPlayground.

## References

- ADAMO-VILLANI, N., AND WRIGHT, K. 2007. Smile: an immersive learning game for deaf and hearing children. In *ACM Proceedings of SIGGRAPH 2007 Educators Program*, ACM Publications, San Diego, USA.
- ADAMO-VILLANI, N., CARPENTER, E., AND ARNS, L. 2006. An immersive virtual environment for learning sign language mathematics. In *ACM Proceedings of SIGGRAPH 2006 Educators Program*, ACM Publications, Boston, MA, USA.
- AGEIA. 2008. *AGEIA PhysX Documentation*. NVIDIA, <http://developer.nvidia.com/object/physx.htm>.
- BYRNE, C. 1996. Water on tap: The use of virtual reality as an educational tool. Tech. rep., University of Washington, College of Engineering, Washington.
- DEDE, C., SALZMAN, M. C., AND LOFTIN, R. B. 1996. Sciencospace: Virtual realities for learning complex and abstract scientific concepts. *Proceedings of IEEE VRAIS '96*, 246–252.
- GERTHSEN, C., AND VOGEL, H. 1993. *Physik - Ein Lehrbuch zum Gebrauch neben Vorlesungen, 17th edition*. Springer publishing, Berlin, Germany.
- GOLDSTEIN, H., POOLE, C. P., AND SAFKO, J. L. 2001. *Classical Mechanics, 3rd Edition*. Addison Wesley, Reading, Massachusetts, USA.
- HECKER, C. 1996. Physics, the next frontier. *Game Developer Magazine*, 12–20.
- KAUFMANN, H., AND DÜNSER, A. 2007. Summary of usability evaluations of an educational augmented reality application. R. Shumaker (Ed.): *Virtual Reality, HCI International Conference (HCII 2007)*, LNCS 4563, Springer-Verlag Berlin Heidelberg, 660669.
- KAUFMANN, H., AND SCHMALSTIEG, D. 2003. Mathematics and geometry education with collaborative augmented reality. *Computers & Graphics* 27, 3 (Jun), 339–345.
- KAUFMANN, H., AND SCHMALSTIEG, D. 2006. Designing immersive virtual reality for geometry education. In *Proceedings of IEEE Virtual Reality Conference 2006*, 51–58.
- KÖRNER, W., HAUSMANN, E., KIELING, G., MENDE, D., AND SPRETKE, H. 1988. *Physik - Fundament der Technik, 10th edition*. VEB publishing company, Lipsk, Germany.
- LIFSHITZ, E. M., AND LANDAU, L. D. 1982. *Course of Theoretical Physics : Mechanics, 3rd ed.* Butterworth-Heinemann.

- MANTOVANI, F. 2003. Vr learning: potential and challenges for the use of 3d environments in education and training. In *Towards CyberPsychology: Mind, Cognitions and Society in the Internet Age*, G. Riva and C. Galamberti, Eds. IOS Press, Amsterdam, 207–225.
- MEYER, B. 2007. *Physics Education in the Field of Mechanics with Virtual Reality*. Master's thesis, University of Applied Sciences Deggendorf and Vienna University of Technology.
- PINTARIC, T., AND KAUFMANN, H. 2007. Affordable infrared-optical pose-tracking for virtual and augmented reality. *Proceedings of Trends and Issues in Tracking for Virtual Environments Workshop, IEEE VR 2007, Charlotte, NC, USA*, 44–51.
- POHL, R. W. 1931. *Mechanik und Akustik - Einföhrung in die Physik, 2th edition*. Springer publishing company, Berlin, Germany.
- SCHMALSTIEG, D., FUHRMANN, A., HESINA, G., SZALAVARI, Z. S., ENCARNACAO, L. M., GERVAUTZ, M., AND PURGATHOFER, W. 2002. The studierstube augmented reality project. *Presence - Teleoperators and Virtual Environments 11*, 1 (Feb), 33–54.
- SEUGLING, A., AND RÖLIN, M. 2006. *Evaluation of Physics Engines and Implementation of a Physics Module in a 3d-Authoring Tool*. Master's thesis, Umea University, Department of Computing Science, Sweden, Umea.
- STARK. 2002. *FOS, Abschluss-Prüfungsaufgaben mit Lösungen, Fachoberschule*. Stark publishing company.
- SZALAVARI, Z. S., AND GERVAUTZ, M. 1997. The personal interaction panel - a two-handed interface for augmented reality. *Computer Graphics Forum 16*, 3, 335–346.
- TAXEN, G., AND NAEVE, A. 2001. Cybermath: Exploring open issues in vr-based learning. *SIGGRAPH 2001 Educators Program SIGGRAPH 2001 Conference Abstracts and Applications*, 49–51.
- WINN, W. 1993. A conceptual basis for educational applications of virtual reality. *Technical Report TR 93-9*.