Junho Tak

Dr. John Stewart

PHYS 2074H – Section H1

April 19, 2010

Contribution of Physics in Computer Technology: Physics Engine

A physics engine is a piece of computer software that is designed to simulate some of the principles of physics. Depending on the how complex and detail the engine is programmed, the engine can represent variety of physical system, including Newton's law of motions, different types of dynamics, and even the relativity of matters (if available). When first hearing of this concept, one may wonder why such a thing would be necessary or even attractive to physicists. One reason is that the physics engine provides a way to find a solution for very complicated problems such as a Computational fluid dynamics (CFD), "one of the branches of fluid mechanics[i] that uses numerical methods and algorithms to solve and analyze problems that involve fluid flows." ("Computational fluid dynamics.") CFD deals with such a complex theories and calculations that even "Super Computers," can only show approximate value. In this case, the physics engine that is very accurate and precise to reality can provide more reliable results. A Physics engines also can be used in the field of film industry as a form of computer graphics (GC), also play a large role in interactive entertainment, as known as video game industry, typically as a middleware[ii]. Before details of the physics engines are actually explained, it would

be prudent to provide the glimpse of the computer and computer software that have brought this concept of physics engine to its current formulation.

A Glimpse of Computer and Computer Software

When thinking of a computer, many will imagine as a machine that doest pretty much everything. These days, people can watch their favorite television shows, create a variety of documents, even able to make a phone call with Skype®, chat with friends on Facebook® or Instant messengers (IM). However, do people truly know about these "does-everything-machines?" The answer is probably no. More clear-cut definition of computer would be, a machine that received inputs and spits out outputs based on the "software" that is determined by the user. When computer was first invented, it was as big as a classroom and was only able to perform very simple calculation. (One of them was called, ENIAC) In fact, computer can be summarized as a simple word, "a really fast calculator." Computer can be broken down to two categories, hardware and software. Simply, hardware is something that people can touch (tangible), and software isn't something can be touched (data, protocol). In other words, hardware can be compared to a human body, and software is the materials that reside in brain, such as a though process, or memories. In early years, computers not only lacked the cutting edge hardware, but also the programming language, which is a language that computer utilizes, weren't complicated enough to perform variety of activities like playing music or show a picture on the screen. Over the several decades, many of technological advances allowed computers to intercept faster processors to do more composite work at a short time, and able to understand

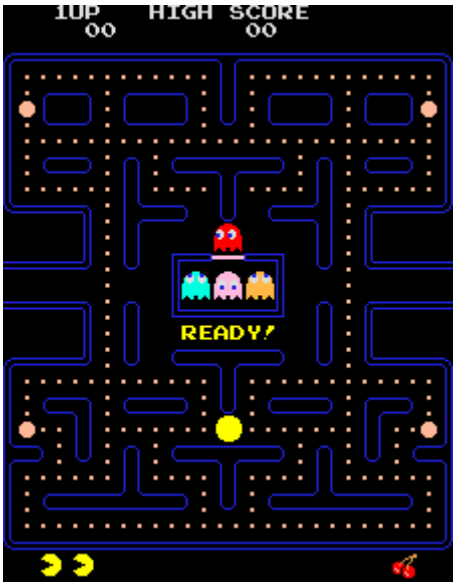
more complex programming language. Researches during this time made computers smarter that it could be used in the field of forecasting, economy and high-tech science.

One of the resulting materials was an "Operating System (OS)." In short, OS is software that connects hardware and connects other software. To make a comparison to a human body (once again) would be the brain, which controls physical component of the body and handle information and memories. From 1980's, Microsoft® has been a big player of OS market. Latest OS included so-called GUI, which is Graphic User Interface, provides the way for users to interact with computer real-time, such as showing what user is typing or pointing with mouse. Windows OS was released around this time, and this is why people can do all the activities which mentioned above with computers these days. Because of programmers from the early ages who created the basics, people no longer have to learn about binary codes or C^{++} languages to put favorite songs in their iPod®.

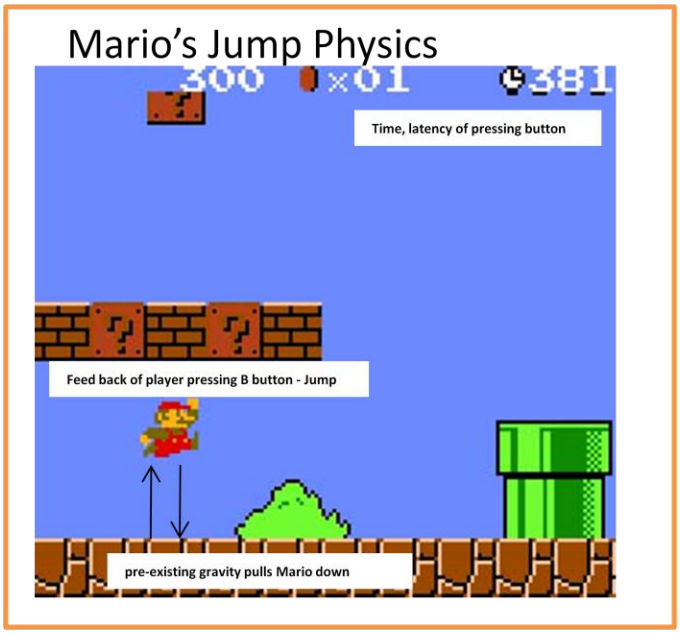
A Rise of Physics Engines

Originally each effect was programmed for its own sake, creating a game with only the physics needed for that title. (Millington 2-3) If a game needed a moving car, then the direction and the vector of the car could be programmed into the game. Because the purpose of the game was only showing a moving car, if a programmer included anything else would be a waste.

In other words, users only had to put something they wanted to use. In the original Pac-Man®, only physics that was used are; changing a direction and a physical contact with enemies that led to game-over. Because of the fact that there has to be code for each movement (in this case) when the program gets complicated, some of the codes get mixed up by not connecting different codes, resulted in glitch or lags in the games and make it difficult for programmers to make their software more physically realistic. In earlier stage, programmers had to spend lots of times to either create the whole new code for dedicated software or provide a code to connect different "little-programs."



At this stage, software that has a reusable technology was needed, which programmers do not have to write a code for every effect, but allows program itself to continue the phenomenon from preexisting effect. In other words, if the system was programmed with simple physics as collision, programmers didn't have to write a code for two cars bouncing off after crash, but program will recognize the situation and move the two cars either stop or change the direction to opposite. In the original Mario game, for example, you can stomp on enemies to eliminate them; enemies didn't just disappeared from the screen, but show the "feedback" such as getting smashed or hide in the shell. "The physics engine is basically a big calculator: it does the mathematics needed to



simulate physics. But it doesn't know what needs to be simulated. In addition to the engine we also need game-specific data that represents the game level (Millington 3).” Therefore, pressing the B button on the controller represent the input, which is the “jump” and the game engine calculate the rate of the time, the latency of pressing button, the integrate with the pre-programmed gravity formula that is given to make Mario to jump certain amount.

Types of Physics Engine: 1. Game Engine

Game Engine is software that is made to create and develop video games. Co-founder of and lead programmer at Orbus Gameworks answered in his article, “You would expect that the answer would be as simple as being shown a car's engine: "Yeup, thar she is." After all, the game engine, much like a car's engine, is what makes the game go. Unfortunately, sometimes there's a fuzzy line between where a game's engine ends and where the content of a game begins, as if there were a fuzzy line between whether a car's air conditioner is part of its engine (Ward).”

Game engine is a collection of parts that is used in the game, such as rendering (visual art), sound, animation, artificial intelligence, networking and other features that can be found in the games. “the concept of a game engine is fairly simple: it exists to abstract the (sometime platform-dependent) details of doing common game-related tasks, like rendering, physics, and input, so that developers (artists, designers, scripter and, yes, even other programmers) can focus on the details that make their games unique (Ward).” Even though Super Mario Bros.® and Pac-Man® is 8-bit era third generation video game, the game engine is what makes those two game different from one another, actually made Mario Bros. more entertaining and appealing game.

Technically, Pac-Man only provided the changing in direction where Mario implemented the side-scrolling which required much more work then Pac-Man. In general, complexity of the engine represents more realistic features that can get.

Types of Physics Engine: 2. Scientific Engines

One of the first computers ENIAC was originally designed for creating ballistics tables to help U.S military, "estimate where artillery shells of various mass would land when fired at varying angles and gunpowder charges, also accounting for drift caused by wind (Physics Engine)."

Even though calculation was a single time only, this opened up the gates for other scientific methods to come through. Physics engine is also used to predict the weather patterns. Specialized engine, Computational Fluid Dynamics (CFD) as mentioned above, calculate how overall weather will circulate by assigning a force vector to particles of air and combining many other forces that could be involved in real situation (perhaps, gravity, humidity, etc) (Physics Engine).

It has been said that there are some difficulties for predicting weather pattern based on CFD. One is that the current simulator doesn't have enough details to fully emulate the real atmosphere, which can be lead to not only the failure of the forecasting but also slowing in evacuation in case of natural disaster. In a research paper, "Reservoir Design Improvement Using CFD." done by Glynn and Shilton said that "A jet impinging on a plane will spread radially outwards on the plane, and this spread may be expected to be axisymmetric. It proves, however, to be difficult to attain axisymmetry when the model is based on a Cartesian grid …

For such grids, the velocity of the radial outflow is generally predicted to be different for flow along the grid diagonals and for flow in the Cartesian directions. This phenomenon has sometimes been referred to as the “butterfly effect” (Glynn and Shilton)." This shows the importance of the accuracy (not to mention the precision) in the field of forecasting which must required using very complicated, yet accurate model, because even a single digit error can cause irreversible situation such as plane crash, or slow response to crisis (Earthquake, Tsunami, or Tornado).

Physics Engines can also be used in tire manufacture industry; it is known that analyzing the air condition, humidity, wetness of the road condition from variety of software allows them to use new materials that can be resistance to hydroplaning, or preventing tire from blowing up easily.

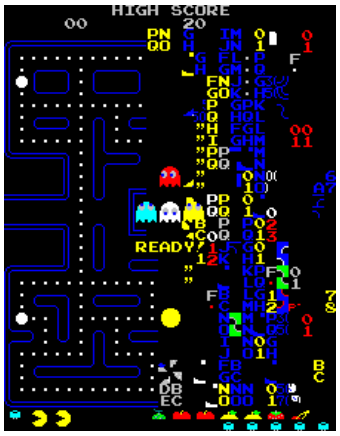
Advance in Computer Technology and Physics Engines

Over the last decades, there has been a great deal of technological advance in computer industry. Not only the higher possessing hardware became more accessible to general consumers but also the progress of the software allowed physics engine to evolve.

To gain a better understanding of advance, one should understand why such an improvement is deemed necessary by people (physicists, computer programmers, even hackers) who support it.

As a game physics perspectives, enhancement of the hardware gave a chance for game developers to create much more realistic atmosphere which they can now try many of different things that they could have not done that in past. Looking back in 1980’s and early 90’s, developers were only able to write a program in 4-bit or 8-bit, and 256 was the highest possible

value (255 is maximum representable) in those system. Because of the lack of the power in hardware, there were several technological difficulties including integer overflow, "which occurs when an arithmetic operation attempts to create a numeric value that is larger than can be represented within the available storage space (Integer Overflow)." The kill screen from several early video games shows the good examples of this concept. Among several games, arcade version of Pac-Man had this bug. Pac-Mac's level count which was a single 8-bit byte only was able to store value from 0 to 255. When player complete level 255 and reach the 256th level, game screen starts crashes with "draws the fruit, the right side of the 256th level becomes a garbled mess of text and symbols, rendering the level impossible to pass by legitimate means (Pac-Man)." Bugs that was found in original Pac-Man was eventually fixed, however this was a good implication of hardware difficulties. After 30 years from the release of Pac-Man, there has been number of innovations taken place in video gaming industry. Release of the Playstation 3® (PS3) by Sony showed the most realistic video gaming atmosphere. Powered with IBM's Cell processor, PS3 allows users to enjoy the maximum experience of virtual reality. Advancement of the physics engine didn't reside on game engines only. Since the Sony's PS3 came to the consumer market, several sectors of society started looking at the powerful machine in order to implement to their purposes. The US Air Force already has purchased thousands of PS3 as a part of their supercomputing project running on in-house Linux system. Also, other filed of science already looked into this powerful hardware to help very complicated



calculations; there was an article about an Astrophysicist replacing his supercomputer with eight Playstation3 because the console was able to perform well enough so it was cheaper to use them.

Pros and Cons of Physics Engines

As mentioned above, physics engine were used in many different field, not only for the research in the laboratory, but also the arcade machine that is collection of piece of hardware and appropriate software for the purpose. One of the main goals of the physics engine is to "save the time." "If you intend to use physics effects in more than one game (and you'll probably be using them in most of your games from now on), then putting the effort into creating a physics engine now pays off when you can simply import it into each new project (Millington 3)." In other words, it is like a writing a music. Once the entire key are on the notes, (creating an engine) then the musician can improvise it with the original songs (using the engine in different project). Since original is kept as a hard copy, it is always available to make a copy, fix or substitute with some other ideas. (Optimizing an engine for other project). Another perspective of advantages is the quality. As mentioned earlier, "the complicated the physics engine gets the realistic the results gets." Not only programming the few other laws of physics, but also adding stuff like, a color of the car, intensity simulator for relationship between windshield and the wind blowing from multiple directions can make the software much more realistic.

However, there isn't only good stuff about physics engines. One of the problem raise from the development of physics engine is the matter of specificity. Since physics engine is software trying to create a real-time simulation of whatever the field is. Because of the attempts

to try to create such a high quality complex model of engine, oftentimes results in situation where just writing a code for that matter is much easier work. For example, "In a game
I worked on recently; we needed no physics other than flags waving in the wind. We could have used a commercial physics engine, but the developer would have had to calculate the properties of each flag, its mass, springiness, and so on. This data would then need to be fed into the physics engine to get it to simulate the flags. … So instead we created a special bit of code just for flag simulation; the characteristics of flags were hard-coded in the software …We avoided using a physics engine because special-case code was more convenient (Millington 4)."

Conclusion

A physics engines has the possibility to overturn the way that the universe of the software is viewed in such a simple way that it could even be compared to the Mario games. However, after discussing fundamentals of physics engine, it was discovered that the assumption wasn't quite true. A physics engines mainly deals with study of dynamics for approximate simulation of physical systems, which is used in as a form of computer graphics in the movies, as a game engine in video game development, and also can be used in multiple fields such as Computational Fluid Dynamics for weather prediction, air plane navigation and also can be used in different sectors of industries. A physics engines provide a way to save time and software development more efficient and convenient, also allow developers to create more realistic atmosphere of simulation. Problems like speed and specificity sometimes take place; however, these can be resolved by optimizing the engine (setting the specificity) and better hardware.

Work Cited

Bourg, David. *Physics for Game Developers*. 1st ed. Sebastopol: O'Reilly Media, Inc., 2001. Print.

Eberly, David H.. *Game Physics (Interactive 3d Technology Series)*. San Francisco: Morgan Kaufmann, 2003. Print.

Glynn, D R and Shilton, A. "Reservoir Design Improvement Using CFD." *Institute of Technology and Engineering, Massey University*

Millington, Ian. *Game Physics Engine Development (The Morgan Kaufmann Series in Interactive 3D Technology)*. 1 ed. San Francisco: Morgan Kaufmann, 2007. Print.

"Physics In Automobiles and Trucks." *Public Services - Social, Religious, Scientific, Products, Environment*. N.p., n.d. Web. 17 Apr. 2010. <http://mb-soft.com/public2/car.html>.

Saad, Shamin. "Advancement in Computer Hardware." *Article Alley*. N.p., 16 Aug. 2005. Web. 15 Apr. 2010. < www.articlealley.com/article_5184_10.html>

Donoghue, Andrew. "US Air Force Upgrading Playstation 3 Linux Cluster." *Eweekeurope,* <http://www.eweekeurope.co.uk/news/news-it-infrastructure/us-air-force-upgrading-playstation-3-linux-cluster-2613>

Ward, Jeff. "What is a Game Engine?." *Main - GameCareerGuide.com* . N.p., 29 Apr. 2008. Web. 17 Apr. 2010. <http://www.gamecareerguide.com/features/529/what_is_a_game_.php>

Wikipedia contributors. "Computational fluid dynamics." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 16 Apr. 2010. Web. 18 Apr. 2010.

Wikipedia contributors. "Computer." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 15 Apr. 2010. Web. 17 Apr. 2010.

Wikipedia contributors. "Fluid mechanics." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 15 Apr. 2010. Web. 17 Apr. 2010.

Wikipedia contributors. "Graphical user interface." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 12 Apr. 2010. Web. 17 Apr. 2010.

Wikipedia contributors. "Hydroplaning (tires)." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 16 Apr. 2010. Web. 17 Apr. 2010.

Wikipedia contributors. "Integer overflow." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 16 Apr. 2010. Web. 17 Apr. 2010.

Wikipedia contributors. "Pac-Man." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 13 Apr. 2010. Web. 17 Apr. 2010.

Wikipedia contributors. "Physics engine." *Wikipedia, The Free Encyclopedia*. Wikipedia, The Free Encyclopedia, 8 Apr. 2010. Web. 17 Apr. 2010.

---

[i] **Fluid mechanics** is the study of fluids and the forces on them. (Fluids include liquids, gases, and plasmas.) ("Fluid mechanics")

[ii] **Middleware** is computer software that connects software components or applications.