High Fidelity Physiological Model for Immersive Simulation and Training

Dr. Teresita M. Sotomayor	Benjamin Quintero	Angela M. Salva	James Sherrill
U.S. Army Research	ECS	SIMETRI, Inc.	ECS
Orlando, Florida	Orlando, Florida	Orlando, Florida	Orlando, Florida
teresita.sotomayor@us.army.m	bquintero@ecsorl.co	angela@simetri.us	jsherrill@ecsorl.co

ABSTRACT

The U.S. Army began evaluating the current state of computer models for physiological processes to determine the feasibility of their use in a range of medical simulation training systems that rely on physiology models. After researching potential models, the Quantitative Human Physiology (QHP)/HumMod model of human physiology seemed to be the most robust and compatible. This open-source physiology model simulated many organ systems of the body and the interplay among the organ systems. The model's structure was based on documented physiological responses within the peer-reviewed literature, but the framework communication protocol was not compatible with real time simulation systems due to performance limitations. Further, QHP incorporated over 5000 variables into its model and not all of this information is needed by a particular simulation. As a result, the Physiology Abstraction Model (PAM) was developed using simple physiological data to create a standalone "plugin" model that offered a standard interface and communication protocol between different components of a physiology model and any medical simulation using the model. The intent of this research was to reduce the complexity of incorporating these high fidelity models into real time systems. This paper documents the initial efforts of integrating HumMod with PAM on hemorrhage control scenarios. It will provide a detailed understanding of the complexities and issues associated with doing this and will provide significant data to demonstrate the computational benefits of utilizing physiology models in this fashion. Finally, this paper will also discuss how the resulting lessons learned have been applied to future research and design considerations.

ABOUT THE AUTHORS

Dr. Teresita M. Sotomayor is a Science and Technology Manager for Medical Simulation Technologies at the U.S. Army Research Laboratory (ARL), Human Dimension, Simulation and Training Directorate (HRED), Simulation and Training Technology Center (STTC). She leads multiple efforts resulting from the Severe Trauma Simulations Army Technology Objective Program and various Small Business Innovation Research projects. Dr. Sotomayor has over 20 years of experience in military modeling and simulation to include constructive and virtual simulations, synthetic natural environments, user-centered design, severe trauma medical simulations, game-based simulations, virtual agents and training effectiveness evaluation. She received a B.S. in Industrial Engineering from the University of Puerto Rico and a M.S. in Operations Research from the George Washington University. She received her Ph.D. in Modeling and Simulation from the University of Central Florida in 2008.

Mr. Benjamin Quintero is Senior Software Engineer at Engineering and Computer Simulations (ECS), based in North Carolina. As a key member and technical pioneer for ECS, Benjamin has over 16 years of experience in Software Engineering. He has spent over 5 years working with ECS to improve their Medical division as well as their internal technologies. Benjamin has a B.S. degree in Computer Science from the College of Engineering at University of Central Florida. Most recently, he has developed SimTech, the next-generation game engine used across many of the top-tier products at ECS, as well as designed and co-developed PAM.

Ms. Angela M. Salva is President and CEO of SIMETRI, Inc., based in Orlando, Florida. She has extensive experience in a wide range of disciplines (research, development, production, business development, and management), enabling her to oversee all SIMETRI initiatives, including medical training and treatment by merging medical science with modeling and simulation technology. Ms. Salva has a B.S. in Mathematics and Computer Science from Emory University and a M.S. in Computer Engineering from the University of Central Florida. She completed the Defense Systems Management College Advanced Program Manager's Course to parallel 17 years of experience in the Simulation and Training Industry.

Mr. James Sherrill is a Software Engineer at Engineering and Computer Simulations (ECS), based in Orlando, Florida, working primarily on medical simulations. Mr. Sherrill has a B.S. in Game Development from Full Sail University and has been working in the simulation industry for three years and is a co-developer for PAM.

High Fidelity Physiological Model for Immersive Simulation and Training

Dr. Teresita M. Sotomayor	Benjamin Quintero	Angela M. Salva	James Sherrill
U.S. Army Research Laboratory	ECS	SIMETRI, Inc.	ECS
Orlando, Florida	Orlando, Florida	Orlando,	Orlando, Florida
teresita.sotomayor@us.army.mil	bquintero@ecsorl.com	angela@simetri-	jsherrill@ecsorl.com

INTRODUCTION

The U.S. Army Research Laboratory Simulation Training and Technology Center (ARL-STTC) has forged a partnership with the U.S. Army Medical Department (AMEDD) Center and School, Department of Combat Medic Training (DCMT) at Fort Sam Houston, TX, to develop and implement new technologies to support simulation-based training environments for Army combat medics.

The Tactical Combat Casualty Care Simulation (TC3Sim) was a prototype application developed through this partnership to support the training of Army Combat Medics, also known as 68Ws. Army Medics are medical personnel responsible for providing the initial medical care to casualties on the battlefield. They are the first responders and must quickly assess a situation, and decide on an appropriate course of action to save lives under combat conditions. This work was leveraged to develop a Navy/Marine Corps variant entitled the Computer Based Corpsman Training System (CBCTS). The TC3Sim / CBCTS games have recently been updated by ARL STTC and renamed Virtual Medic (vMedic). The vMedic uses game-engine based simulations and state-of-the-art instructional development strategies to support a student's need to master a variety of competencies and to apply them in unique situations. The simulation immerses students into scenario driven events to teach and evaluate a student's knowledge regarding the essential tactics, techniques and procedures required to successfully perform as a Combat Medic. Those key tasks include the ability to assess casualties, perform triage, provide initial treatment, and prepare a casualty for evacuation under battlefield conditions. The casualties' physiology is simulated using the Physiology Abstraction Model (PAM), a stand-alone physiological model. PAM provided enough fidelity to simulate the casualties' physiology, but research was needed to determine if higher fidelity, open source physiological models could be integrated and provide more realistic simulation scenarios.

The primary objective for this research was to increase the fidelity of the physiology model as well as to add the serialization of model data. Specifically, the proposed effort focused on enhancing the existing PAM and demonstrating how this model might be used in conjunction with the Army's current medical simulation training systems, such as vMedic.

BACKGROUND

The vMedic simulation trains military personnel to successfully perform as an Army Combat Medic and Combat Lifesaver. The simulation was developed to support the training of Tactical Combat Casualty Care (TC3) principles. TC3 is the pre-hospital care rendered to a casualty at the point of injury. The application of TC3 principles during a tactical combat environment has proven highly effective and is a major reason why the Army has experienced an increase in survivability in the battlefield. TC3 addresses the casualties who will die if they do not receive timely and appropriate medical intervention. TC3 is structured so that the correct intervention is performed at the correct time in order to meet the three important goals of field care: treat the casualty, prevent additional casualties and complete the mission. The simulation focuses on the critical tasks required to assess casualties, perform triage, provide initial treatment, and prepare a casualty for evacuation under battlefield conditions.

To simulate these casualties, vMedic incorporates a physiology model for each casualty that must interoperate with other systems within the simulation. Each casualty incorporates behaviors, animations and sounds that are driven by user interactions with the physiology model. For example, each synthetic casualty will exhibit signs and symptoms based on which interventions the user applies to the casualty. As these signs and symptoms change, messages are passed between the physiology model to other subsystems within the vMedic architecture to tell the casualty how to behave within the simulation, e.g.,

how to act, what to say, and which animations should be played at a particular time.
The intent of the PAM effort was to decouple the physiology model from medical simulation applications to enable a centrally managed physiology engine that can be used to power multiple applications. This effort continued research and development of a high fidelity physiological model for immersive simulation and training that could be used with vMedic by integrating an Open Source physiology engine, such as HumMod, developed by a third party.

Simulating a human body is a complex problem that involves the cooperation of many systems. During our initial research we determined that the Quantitative Human Physiology model (QHP), later renamed HumMod, was the most comprehensive physiology model at the time. HumMod provided an independent model solver based on a collection of validated sets of values. These models described internal values such as time-based rate functions that exhibited nonlinear behaviors. Throughout the research, HumMod was continually compared to fast-approximation methods such as the native physiology model found in vMedic.

MILITARY RELEVANCE

DCMT at the AMEDD Center and School serves as the proponent for the 68W Health Care Specialist and provides the Army with highly motivated and disciplined Combat Medics who are National Registry Emergency Medical Technician-Basic certified. Pre-hospital care continues to be a critical aspect of battlefield medicine. During the 16-week long 68W10 Healthcare Specialist Course at AMEDD, instructors use a variety of instructional development strategies to effectively prepare Soldier medics for administering care and saving lives on the battlefield. Soldier medics prepare for these missions by participating in field exercises with an emphasis on mass casualties and patient evacuation. Classroom instruction provides Soldiers with the basic knowledge and skills while scenario-based training provides Soldiers with an opportunity to test their skills in multiple and relevant operating environments. vMedic was developed to support critical thinking skills development. Other simulation capabilities, to include mannequins and part task trainers, provide opportunities to conduct hands-on training for skill and confidence development. Most of the systems have proprietary physiological models. In addition, the fidelity of the model may vary from system to system. In scenario based training, where different systems are introduced to create an immersive environment, it would be beneficial to be able to tailor the fidelity of the physiology model to support the training of medical personnel through the continuum of care. The idea behind this research was to investigate the feasibility of decoupling the physiology model from the vMedic system and to provide an interface that would allow the system to integrate with an external open source physiology engine as a proof of principle.

TECHNICAL APPROACH

The primary objective for this research was to increase the fidelity of the physiology model while enabling the serialization of model data. Specifically, the effort focused on enhancing the existing PAM and exploring how this model might be used with vMedic.

HumMod is open-source software developed through the University of Mississippi Medical Center. The model includes many organ systems of the body and the interplay among systems. In addition to this physiological framework, a variety of diseases and interventions can be simulated to add more variability for a student as they work to treat simulated casualties. To date, the HumMod model is one of the most advanced open source physiological model available, but it is not incorporated into real time medical simulation training systems developed for combat medics.

Ongoing research efforts have focused on implementing an interface that will allow ARL STTC medical simulation training systems to communicate with an external physiological model. Current systems being modified to support this interface include vMedic and the Virtual Medical Simulation Training Center, a persistent virtual world capability that implements a Dynamic Learning Management System (DLMS). The DLMS is an adaptive, instructorless system that continually tailors instruction based on the student's learning style and level of competency.

This new research effort aimed to improve and enhance the realism and fidelity of the casualty models used in different medical simulation training systems. The outcome of this research was intended to be a high fidelity physiology model that was interoperable with systems, including serious games, virtual worlds, part task training systems, etc. The goal was to enhance the physiology of a casualty so that the simulated training exercises would be more realistic and immersive to help prepare the

Warfighter to make medical decisions within an operational environment more effectively.

To better explain the steps taken in our research, the following is a brief explanation of how HumMod functions. HumMod is a collection of libraries that is compiled into several executables. The two main executables are titled "HumMod" and "Model Solver". HumMod represents the front end visualization; a 2D graph generator that shows the simulation over a fixed span of time. Generally these spans of time are very large and allow HumMod to show the status of a casualty over the span of many hours. The model solver performs the parsing and computation, the actual simulation of the virtual patient. The two executables communicate with each other using a process known as piping. Named pipes are treated as a file that resides on the local computer's file system, and they serve as a method of communicating between two independent processes. One of the many drawbacks of pipes however, is that they only provide a single direction for the flow of information (half-duplex). In order to maintain a full-duplex communication protocol, two separate pipes need to be created and destroyed per event or message. One pipe represents the data that is flowing into the model solver, and the other pipe represents the data that is flowing out of the model solver and into the visualization tool.

For the purposes of this research, the model solver was integrated into vMedic, a real-time 3D visualization simulation. Leveraging the 3D environment and characters allowed true physical reactions to be associated with a virtual patient's health, instead of a bar chart, though much of that information was also accessible at run-time. The reasoning for this visual association was for the purpose of training combat medics for Care-Under-Fire scenarios. It was important to give players only the tools that would be available to them on the field, not all of the luxuries afforded in a well equipped facility such as a hospital. In order to achieve this result, most of the effort concentrated in two specific areas. The first area focused in the communication between the 3D patient and the model solver. The second area focused on performance enhancements to HumMod while avoiding architectural changes to the model solver.

The objective of integrating HumMod was superseded by the greater objective of integrating any and all model solvers. The intent was to design a system that allowed a single form of communication for any visualization tool or simulation. This single protocol would become a standardized framework that served as the glue between the simulation and any physiology model that followed the standard. The benefit to this system was that it allowed for fast comparison of results between multiple physiology models by simply replacing a Dynamic Link Library (DLL), where the physiology model resides. This abstraction is what would later be known as the PAM. With PAM, any physiology model only needed to implement a small interface that allowed it to communicate with compatible visualization tools. Physiology models were then free to generate results with varying degrees of fidelity. For example, vMedic used only a small subset of factors when determining the visual state of a virtual patient. The vMedic fast-approximation model only simulated these exact states, but HumMod simulated an entire human model and only returned the minimal states that were requested. This resulted in models that appeared very different when placed side-by-side. One model used an approach that was tuned for a player's training experience and the other model was tuned to be more physically accurate. Still, both models were used interchangeably, without any need to modify vMedic after exchanging the physiology library. The tangible benefit of this approach is that disparate systems could potentially interoperate with the same physiology model at different fidelity levels.

In working with HumMod, many strengths as well as weaknesses of a large physiology model became apparent. As part of the research, the intent was to use HumMod as-is. Therefore, there was a need to evaluate the usability of the model solver without modifications. One of the great strengths of HumMod is its understanding of such a large number of parameters. With over 5000 variables being simulated, HumMod is able to produce results for a wide array of conditions and treatments. The data that is currently being used by HumMod is an open-sourced global effort that is comprised of many physiologists who contribute. This database is still active which means that new and better formulas are being entered consistently. With that, it can be inferred that the fidelity of HumMod will also improve over time.

CHALLENGES

One of the technical hurdles encountered using a physiology model as complex as HumMod was in obtaining real-time performance. At first, HumMod seemed to only be able to generate results at about 1-3 updates per second while most real-time simulations are required to run at 30-60 frames per second; a factor of 20x or more. To circumvent the considerable gap in performance between vMedic physiology requirements and the HumMod physiology model, a multi-core threading solution was required. The multi-core solution used to achieve run-time performance with HumMod involved the decoupling of vMedic and HumMod. While vMedic was executing on the primary thread, a secondary thread was created to manage communications with HumMod. Special care was taken to avoid situations where vMedic would be left waiting for HumMod to finish its simulation loop. A command buffer was used so that commands could be queued by vMedic and processed as soon as possible by HumMod. This decoupling ensured that vMedic would maintain its responsiveness while ensuring that HumMod had the time it needed to process complex models.

One of the concerns that were raised by using this approach of decoupling is that HumMod may begin to fall behind as the simulation raced ahead. This would have resulted in displaying data that is likely far behind the actual real-time state of the simulation. An example of what might happen can be described with a scenario of two vehicles traveling from point A to point B, with one of the cars traveling at 10mph faster. After an hour of driving, the faster car (the visual simulation) would be 10 miles ahead of the other car (HumMod) and the things that they see at that time (the Physiology Data) will be completely different. In the end we had to reduce the frequency of calls to HumMod, allowing it to process larger time-steps, and minimize the use of the named pipes. Since the speed of pipe communication is often limited to the performance of a local machines' hard drive, reducing the number of request messages resulted in a reduction of input/output latency.

As part of integrating HumMod with PAM, a capability was developed to capture and display data generated by the physiology model during a simulated scenario. Developing this capability allowed users to closely analyze how the same scenarios performed when using vMedic with each of the two separate physiological models. The data captured was exported as a ".csv" file, which was depicted graphically using Microsoft Excel.
A "Chest Wound" scenario was run using each of the models which resulted in different outcomes. Treatments were not administered so that the models were tested in an unaltered state. As expected, the PAM/vMedic model changes over time were graphed (depicted in Figure 1), illustrating that the physiological equations within the model are linear.

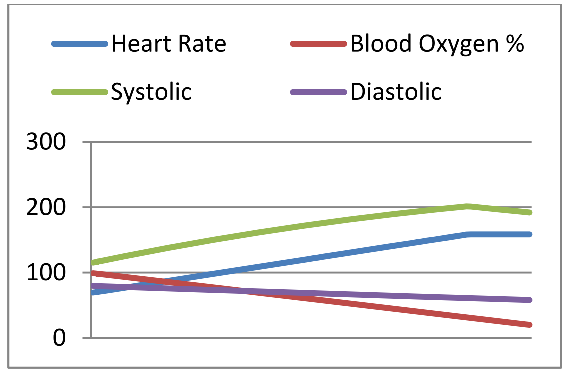

Figure 1. PAM/vMedic Chest Wound Scenario Physiological Model Data

In contrast, when the same "Chest Wound" scenario was run using the PAM/HumMod physiological model, the graph depicted in Figure 2 illustrates that these physiological equations are more complex than and not as linear as the PAM/vMedic model.

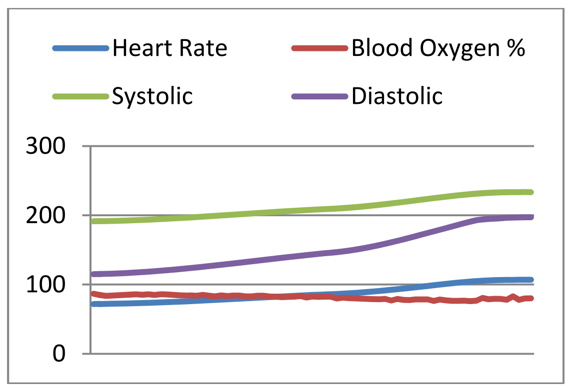

Figure 2. PAM/HumMod Chest Wound Scenario Physiological Model Data

The collected data is being used as a part of the integration effort to confirm whether scenarios are behaving as expected. The "Chest Wound" scenario showed that many of the datapoints generated by the PAM/HumMod model unexpectedly increased over time. Despite an injury to the chest, the casualty's blood oxygen and blood volume levels increased by only fractions of a percent but still resulted in 21 units of blood being added into the equation over a period of 8 – 10 minutes. When investigating how this could occur, it was assumed that some of the measures taken to increase HumMod's performance, or the real-time intervals of time given to HumMod, may have caused HumMod to behave erratically. After more investigation, it was found that the values could not decrease over time.

The focus switched to improving the file format used to import data output from PAM into Microsoft Excel for improved visualization. The logs from each casualty are graphed to display differences in the simulations. This data is being used to identify differences between TC3Sim's low fidelity (high frequency) model, and HumMods' slower, but higher fidelity model.

At the completion of the task, it was concluded that more accurate physiological data for vMedic is being logged when running PAM with HumMod. Data logs continue to be collected for the various hemorrhage scenarios with vMedic running PAM with HumMod. A single casualty - upper arm amputation scenario was run using both of the models and different results were generated. Treatments were not administered so that the models were tested in an unaltered state. As expected, the PAM/vMedic model changes over time (depicted in Figure 3), illustrated that the physiological equations within the model are linear.

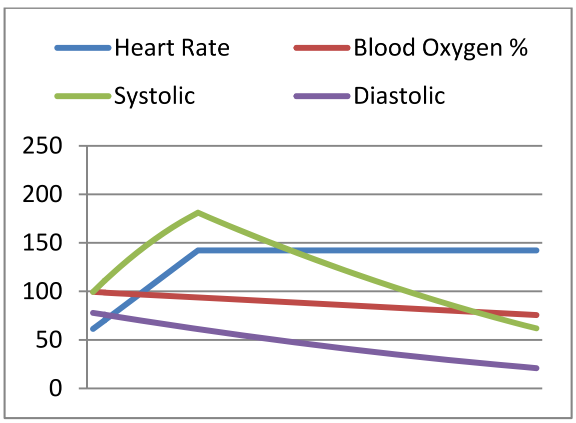

Figure 3. PAM/vMedic Upper Arm Amputation Scenario Physiological Model Data

In contrast, the same "Amputation" scenario was run using the PAM/HumMod physiological model (depicted in Figure 4), illustrating that these physiological equations are more complex than and not as linear as the PAM/vMedic model.

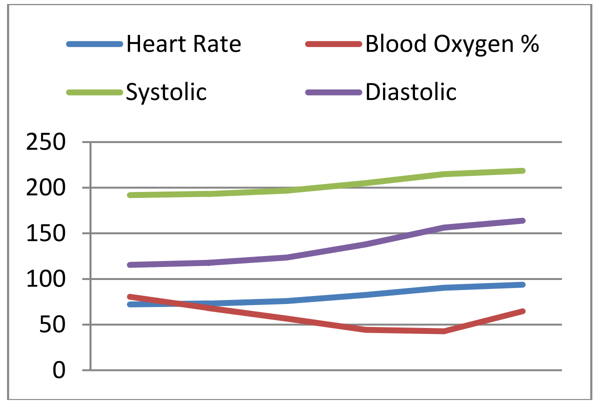

Figure 4. PAM/HumMod Upper Arm Amputation Scenario Physiological Model Data

The data serialization task has proven to be useful in finding issues during the integration efforts, but it has also proven the original hypothesis to be correct: HumMod is a higher fidelity physiological model that more realistically simulates injuries than the vMedic physiology model that was originally developed for TC3Sim.

LESSONS LEARNED

Results obtained regarding HumMod performance apply for a single casualty. Though real-time performance was achieved with HumMod, its effectiveness for large scale simulations is yet to be determined. Currently HumMod requires a single instance of the model solver to execute per virtual patient. This would imply that mass casualty scenarios would require a large amount of memory and a significant number of Input/Output operations. System resources would quickly erode using this approach. While low fidelity models such as the TC3Sim fast-approximation model are capable of simulating mass casualty scenarios, HumMod is still not proven to achieve this level of performance in a graceful manner.

Through all of the challenges encountered in achieving real-time performance, PAM was consistently a pillar in the development process. Toggling between PAM and HumMod proved to be an invaluable tool in testing the differences of models that resided on polar ends of the fidelity scale. The decision to create an abstraction layer did present an additional challenge to the development process, but a solution that is ready to accept varied levels of fidelity with low impact to the visual simulation was developed.

FUTURE RESEARCH

Moving forward, the hope is to continue to expand on what was achieved with the implementation of the PAM. The PAM is a framework that has been proven to work with low and high fidelity models. The simplistic design allows for a lightweight mediator to standardize the language between all physiology models. There is however, still much work to be done.

PAM currently has a very small language specification. For the purposes of visualizing a virtual patient using current graphics technology, only a small number of parameters were required. Patients could react expressively to actions such as screaming in pain or groaning, but subtle wincing or pursing of the lips are difficult to achieve at this time. In addition, vMedic was a Combat Medic trainer, meaning that most of the generic events passed into the physiology model were related to hemorrhage. As virtual human visualization becomes more advanced, and more nuanced expressions are able to be shown, a more complicated array of parameters will need to be defined. More potential injuries will need to be defined, as well as treatments, to ensure that PAM can be used for dynamic simulations such as vMedic in addition to controlled environments such as a hospital trainer.

HumMod is a powerful physiology simulator, but it lacks the capability to execute for large scale scenarios. Looking to the future, the best solution may be to find or instigate an open-source model solver that incorporates the open-source data used by HumMod. This approach would allow for a community of interested parties to contribute to a common goal. With a distributed approach like open-source, real-time performance could be achieved for more than a single virtual patient at a time. New and better communication methods could be utilized instead of using named pipes, and also by extending the hard drive. Pipes are restricted to the performance of a machine's hard drive, the slowest component of most personal computers today. HumMod also uses threaded polling to access pipes along with fixed sleep intervals. Creating a well encapsulated dynamic or static library that allows for more immediate response from the physiology simulation would help to reduce these inherent latencies. If a library is not possible, then using local sockets or some form of Transmission Control Protocol (TCP) connection would prove to be more efficient than relying on system files.

AKNOWLEDGEMENTS

The team would like to recognize the contribution of Dr. Robert Hester from the University of Mississippi. As one of the principal developers of the model, he supported the team in the research of the HumMod.

REFERENCES

1. "Quantitative Circulatory Physiology: an integrative mathematical model of human physiology for medical education," Sean R. Abram, Benjamin L. Hodnett, Richard L. Summers, Thomas G. Coleman, and Robert L. Hester, Advances in Physiology Education, 31:202-210, 2007

2. Farmer E, van Rooij J, Riemersma J, Joma P, Morall J. Handbook of Simulator Based Training. Aldershot, Hampshire, UK: Ashgate; 1999.

3. Miller RB. Psychological considerations in the design of training equipment. Report no. WADC-TR-54-563, AD 71202. Wright Patterson Air Force Base, OH; Wright Air Development Center; 1953.

4. "Quantitative Circulatory Physiology: an integrative mathematical model of human physiology for medical education," Sean R. Abram, Benjamin L. Hodnett, Richard L. Summers, Thomas G. Coleman, and Robert L. Hester, Advances in Physiology Education, 31:202-210, 2007

5. "GOLEM – Multimedia Simulator for Medical Education," Jiri Kofranek, Luu Danh Anh Vu, Hana Snaselova, roman Kerekes, Tomas Velan, MedInfo 2001

6. "Applied Mathematical Models in Human Physiology", J.T. Ottesen, M.S. Olufsen, and J.K. Larsen, Department of Mathematics and Physics, Roskilde University, October 2006

7. "Avatars Alive!, The Integration of Physiology Models and Computer Generated Avatars in a Multiplayer Online Simulation," Laura Kusumoto, Wm. LeRoy Heinrichs, Parvati Dev, and Patricia Youngblood, Stanford University Medical Media and Information Technologies (SUMMIT), TATRC White Paper, 2007

8. Bevington, Philip R., and D. Keith Robinson. Data Reduction and Error Analysis for the Physical Sciences. New York: McGraw-Hill, 1992.