

Physics Based Modeling of a Warhead Detonation using Graphics Processors

Mr. Craig Eidman, Mr. Garry Boyle
Air Force Research Laboratory, 711th Human
Performance Wing, Warfighter
Readiness Research Division, Mesa, AZ
craig.eidman@mesa.afmc.af.mil,
garry.boyle@mesa.afmc.af.mil

Mr. Merrill Lay, Mr. Eytan Pollak
L-3 Communications, Link Simulation &
Training, Orlando FL
Merrill.Lay@l-3com.com,
Eytan.pollak@l-3com.com

ABSTRACT

Modeling highly complex entity interactions with high precision is challenging in distributed training simulations. When using existing protocols and standards, most weapons effects are resolved by a probability of a kill or a roll of the dice, reducing accuracy and fair fight. This paper presents the results of a study by the Air Force Research Lab (AFRL) to determine if running physics algorithms using the NVIDIA PhysX SDK with a constructive simulation could more accurately model the damage associated with an aircraft/missile engagement than can be done with the currently used statistics based approach. A statistical approach assumes damage based only on missile proximity at the time of detonation along with a pre-determined probability of kill. This approach typically does not factor the exact aspect, geometry, aircraft material strength, and missile warhead capability at the time of detonation. The study leveraged a Physics-Based Environment Generator (PBEG) running with the Expert Common Interactive Training Environment (XCITE) constructive simulation. The PBEG was programmed to take entity control of the aircraft and missiles from XCITE to determine the level of damage, provide a realistic visualization of the impact and return damage assessment to XCITE. A target aircraft model was constructed rendering multiple sections and facets, each capable of individually sustaining damage. A missile warhead model with a simplified TNT explosive and fragmentation was integrated. The PBEG computed the force of the initial blast shockwave and impact from each high velocity shrapnel piece striking the aircraft on each specific facet. A numerical damage value is assigned for each aircraft part, and damage results are returned to XCITE for training feedback. This approach was shown to increase accuracy in damage assessments over the traditional approach. This paper summarizes the study results and provides recommendations for further investigations using graphics processors for modeling weapons effects.

ABOUT THE AUTHORS

Mr. Craig Eidman is a Senior Electrical Engineer with the Air Force Research Laboratory, 711th Human Performance Wing, Warfighter Readiness Research Division, Mesa Research Site. He serves as the Immersive Environments Branch Technical Advisor and is the lead technologist for all laboratory synthetic environments research, Electronic Warfare Training, and advanced Modeling and Simulation efforts. Mr. Eidman attended the United States Air Force Academy where he earned a Bachelor of Science in Electrical Engineering in 1983. He served in the Air Force as a Command Pilot flying various fighter and attack aircraft in multiple theaters. Additionally, he has a Masters of Aeronautical Science, Embry-Riddle Aeronautical University, with Distinction and is an Outstanding Graduate of the United States Air War College.

Mr. Garry Boyle is a senior electronic engineer in the Immersive Environments Branch of the Air Force Research Laboratory, 711th Human Performance Wing Warfighter Readiness Research Division. Mr. Boyle has extensive experience in simulator systems engineering, synthetic environment development, and M&S software development. He is currently the project lead for the eXpert Common Immersive Environment R&D, F-22 CDS R&D, F-16 CCIP flight sim architecture, and technical support for coalition training and international standards development. He is the TTCP AER TP-1 working group lead for Improved CGFs. He led the AFRES Multitask Trainer (MTT) program, which was an R&D effort that provided a squadron-based trainer that could also be deployed with the unit.

Mr. Eytan Pollak received his B.Sc. and M.Sc. from Technion-I.I.T (Haifa, Israel) and his Ph.D. from Purdue University, West Lafayette, IN. He has over thirty years experience in managing research and development

programs, he holds several patents and has published papers in Control Systems, Robotics, Distributed Flight and Ground Simulations/Simulators, Embedded Systems, and Cyber Physical Systems. He served as an M&S Technology Manager, IRAD Technical Director and Principal Investigator at Lockheed Martin. Since 2004, he has been Director of Strategic Technologies at L-3 Communications Link Simulation & Training. He is a research professor at the UCF School of Electrical and Computer Science and Editorial Consultant for the Journal of Robotic Surgery.

Mr. Merrill Lay is the Manager of Advanced Technology for Link Simulation and Training, where he is currently developing and prototyping next generation training concepts. He has over 33 years of collective experience in Flight & Ground Simulation domain, specializing in computer image generation. He was a key contributor in the development and production of the Compu-Scene series product line and held the position of Lead Product Application Engineer was a Principal Investigator in the Independent Research & Development group at Lockheed Martin Simulation, Training and Support. His career has included assignments supporting customers in the advancement of visualization for aircraft such as the B-2, F-22, and RAH-66. Merrill holds a bachelor's degree in Electronic Engineering Technology.

Physics Based Modeling of a Warhead Detonation using Graphics Processors

Mr. Craig Eidman, Mr. Garry Boyle
Air Force Research Laboratory, 711th Human
Performance Wing, Warfighter
Readiness Research Division, Mesa, AZ
craig.eidman@mesa.afmc.af.mil,
garry.boyle@mesa.afmc.af.mil

Mr. Merrill Lay, Mr. Eytan Pollak
L-3 Communications, Link Simulation &
Training, Orlando FL
Merrill.Lay@l-3com.com,
Eytan.pollak@l-3com.com

INTRODUCTION

Modeling complex entity interactions with high precision is challenging in distributed training simulations. The effect of modeling a high speed missile and its interaction with an aircraft target is particularly difficult. Missiles and aircraft can close together hundreds of feet in a fraction of a second. Even with rapid processing frame times and manageable network latencies, an error in position between a missile detonation and the aircraft target is likely to occur in distributed simulations. Even when these distance errors are in the tens of feet, the missile's small warhead and an inexact target aircraft location can produce significant inaccuracies when calculating detonation effects. The Joint Advanced Distributed Simulation Joint Test Force running a series of detailed distributed simulation tests for air-to-air missile engagements determined that even very slight positional errors introduced by latency and jitter could cause missile models to fail (Wright, 1999). Inability to accurately determine a missile's effectiveness against a target may produce undesired effects in training simulations. When using typical distributed protocols and standards for real-time training simulations, most weapons effects are resolved by a probability of a kill or a roll of the dice, reducing accuracy and fair fight. Being on the "wrong" side of the roll of the dice often leaves the pilot who was "hit" by the missile with the questions: Why did it kill me? How close was it? How could I have better survived?

PPU Approach for Weapons Effects Modeling

Some of the technical limitations in distributed simulation missile model performance may be compensated for by utilizing advanced algorithms and hardware. One approach that may be useful is downloading the missile engagement to a parallel process for more accurate computation. Previous studies have demonstrated success in moving complex constructive interactions for computer generated forces from CPUs onto PC physics processing units (PPU)

and graphics processing units (GPU) to compensate for the limitations of high entity counts and line-of-sight calculations (Verdesca, Munro, Hoffman, Bauer, & Manocha, 2005). This approach may be used to move similarly calculation intensive interactive models onto comparable architectures for improved results. Ideally, physics processors should be "optimized" by design for calculating physical interactions of the blast, warhead fragmentation, and impact on the target. Additionally, the blast pattern from a warhead at close ranges can approximate to a ray tracing problem, similar to the algorithms that drive highly detailed scene generation in graphic processors. Studies have shown that ray-tracing algorithms run faster in GPUs than on CPU based architectures. (Purcell, Buck, Mark, & Hanrahan, 2002). Therefore, GPUs should be efficient at modeling the flyout of fragments and the supporting ray tracing problems such as line-of-site, target planform and exposure.

Study Description

This paper discusses the results of a joint study conducted by the United States Air Force Research Laboratory and L-3 Communications, Link Simulation & Training Division, to determine if running physics algorithms on a GPUs could more accurately model the damage associated with an aircraft/missile engagement than can be done with the currently used statistical based approach. The study used the NVIDIA PhysX Software Development Kit and algorithms from an extremely high fidelity, physics based, constructive simulation. The resultant Physics-Based Environment Generator (PBEG) ran code derived from the Expert Common Interactive Training Environment (XCITE) constructive simulation. This approach was shown to increase accuracy in damage assessments over the traditional approach. This paper summarizes the study results and provides recommendations for further investigations using graphics processors for modeling weapons effects, specifically those supporting distributed training applications.

WARHEAD MODELING SHORTFALLS

The Statistical Approach

The statistical approach is the primary method used to analyze warhead/target interactions in distributed simulations and can be summed up in the phrase “a roll of the dice”. This approach assumes damage based only on missile proximity at the time of detonation (assuming it can be accurately interpreted by the simulation) along with a pre-determined probability of kill (P_k). This approach typically does not factor in all critical information that influence the results of a warhead detonation, such as exact aspect, geometry, aircraft material strength, and missile warhead capability at detonation. One significant advantage of this approach is computationally efficiency. Lower fidelity models may use a P_k that does not include detonation distance. High fidelity statistical models may use the distance between the detonation and the target to adjust the P_k , but latency between simulations and jitter may degrade these models.

The “Brute Force” Approach

When low fidelity models interact, or when a distributed simulation encounters lost data packets and network delays, another approach used is for the missile to send any detonation as a direct impact – a “zero” value miss distance. The targeted simulator host can accept or reject the “kill” as it sees fit. Most host systems in the targeted aircraft will attempt to calculate the perceived miss distance independently to use a localized statistical approach. Others may just accept the “kill” and kill the aircraft. This mismatch between systems often leads to confusion as to whether a missile was truly effective or not. Other systems increase the kill distance of the warhead to compensate for any latency or network issues, sometimes up to thousands of feet.

Training Implications

Significant training issues can arise from these two approaches to missile warhead modeling. The pilot in a high fidelity flight simulator could have maneuvered or reacted to a missile effectively, but be killed by a bad roll of the dice or a poorly implemented kill decision. It becomes impossible to tell the pilot how to improve his counter missile defensive tactics if the feedback from the results of the engagement appear random. Replacing these P_k methods with an improved representation of the interaction of the missile and the

target may vastly improve counter-threat training in these types of flight simulators.

PHYSICS-BASED ENVIRONMENT GENERATOR (PBEG)

AFRL and Link Training designed, integrated, and tested the Physics-Based Environment Generator (PBEG) to determine if a commercial off the shelf physics engine could be used to more accurately model the damage associated with an aircraft/missile engagement than the often used statistical approach. The approach used Link’s Physics-Based Environment Generator (PBEG) running the NVIDIA PhysX physics computational engine to perform damage calculations. This computational engine was chosen due to extensive experience with this commercial product. A physics based methodology was applied to the XCITE computer generated forces (CGF) system in place of the normal statistically based damage assessment (Figure 1). As a test of the physics based approach, an engagement between a specific aircraft and missile was chosen. The physics based approach considered missile proximity, explosive force, and the structural strength aspects of the various aircraft components in determining aircraft damage. Unclassified data was used throughout development.



Figure 1. PBEG for XCITE Damage Assessment

Methodology

The PBEG uses several steps to calculate the results of an air-to-air missile engaging a target:

1. Handoff of the XCITE CGF missile and target entity states to the PBEG for off board calculation of the impact.
2. Parallel processing of aero models for missile and target aircraft so entity state can be preserved within the distributed simulation.

3. Generate a multi-surface, 3-D model of the missile and target airframe to represent each objects individual parts for interaction and later visualization.
4. Calculate structural strength for each part of aircraft.
5. Model the explosive impact on the dynamic articulating 3-D target by calculating apply forces from missile detonation (blast and fragmentation impact) using physics calculations and PhysX .
6. Accurately and rapidly transmit physics based results and state information from missile impact back into the XCITE CGF simulation.
7. After the engagement is completed, generate a 3-D visualization of the engagement.

AIRCRAFT, MISSILE, AND WARHEAD ASSUMPTIONS

For this study, an AIM-120 advanced medium-range air-to-air missile (AMRAAM) and MiG-29 aircraft were selected. These models were chosen because satisfactory open-source data was readily available to allow reasonable model development. The system was designed to be capable of ingesting higher fidelity proprietary or classified data if available. Data from other missiles and aircraft could be easily interchanged.

AAMRAM A-to-A Missile and Warhead

The AAMRAM missile selected for this study is a commonly deployed air-to-air missile. Because the AMRAAM’s specific warhead size, explosive formulation, and fragmentation are unavailable in open sources, a generic TNT model was developed to model the explosive. The other specifications for the missile including missile weight, Gurney Constant, Charge-to-Metal Ratio, and K constant were derived from a variety of open sources. The fusing model was taken from XCITE. These values, shown in Table 1, are

Table 1. Open Source AMRAAM Parameters

Missile Part (Piece)	Value
Total Weight of Missile	157 (Kg)
Weight of Warhead	22 (Kg)
Number of Fragment Pieces	198 (Kg)
Weight of Each Fragment Piece	1.5 (Kg)
Gurney Constant	2700
Charge-to-Metal Ratio	163
K constant	0.5

adjustable in the PBEG for any type of explosive or fusing model and did not cause error in the implementation.

A physical 3-dimensional wireframe representation of the missile and warhead were developed so each part separating during blast could be modeled for impact. These models also supported later visualization of the event.

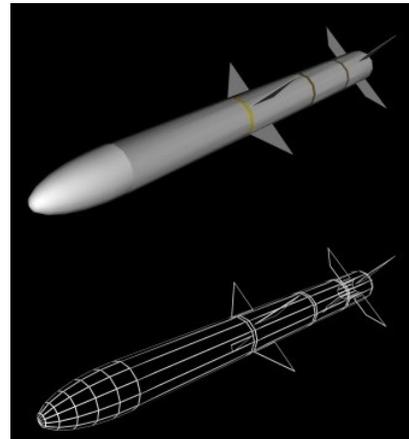


Figure 2. PBEG AMRAAM Models

MiG-29 Fighter Aircraft

The aircraft modeled for this study was a Mikoyan MiG-29. This aircraft was used because sufficient information to develop a model was available in open sources. The specifications used in the PBEG test are shown in Table 2. Like the missile, the aircraft model

Table 2. Mass of Main Aircraft Parts

Aircraft Part (Pieces)	Mass (Kg)
Engines(2)	1,000 (each)
Wings(2)	1,000 (each)
Fuselage	3,000
Vertical Stabilizers(2)	500 (each)
Horizontal Stabilizers(2)	500 (each)
Cockpit	2,000
Total Weight	11,000

was constructed as a wireframe representation. Part locations and orientations are a physically accurate representation of the plane. Each major component of the plane is mapped to a corresponding PhysX part. To determine the maximum force for aircraft structural



Figure 3. MIG_29 and PBEG Wireframe Model

mass of the aircraft piece effected was used. For design purposes a safety factor of 2 was included.

MISSILE DAMAGE

A fixed wing aircraft struck by a missile receives two types of damage. The first type of damage is the due to the blast or increase in pressure by the force of the missile explosion. The blast typically damages the aircraft through structural failure and weakening. The second type of damage is the piercing of the aircraft skin from high velocity fragments projected by the missile explosion. Fragmentation damage destroys specific aircraft systems such as engines or fuel tanks.

Pressure Calculation

To find the pressure caused by the XCITE model detonation, the exact distance from the target to detonation was calculated. The AMRAAM model assumed a 22 kg warhead size. Source data driving the blast model assumed a 1 kg TNT warhead size. To compensate, a distance from explosion to warhead size ratio was integrated into the algorithm. Since the ratio of the warhead weight and distance from the explosion was established, that value could be used to interpolate the effects from a 1 kg explosion to a 22 kg explosion. The resultant value was compared using a look up table derived from open TNT data sources, shown in Figure 4. The pressure from the blast could then be calculated.

Blast Damage Effects Model

As shown in Figure 4, two distinct pressures are caused from the blast, peak overpressure and dynamic pressure. For this model, these pressures were summed to derive the total blast pressure.

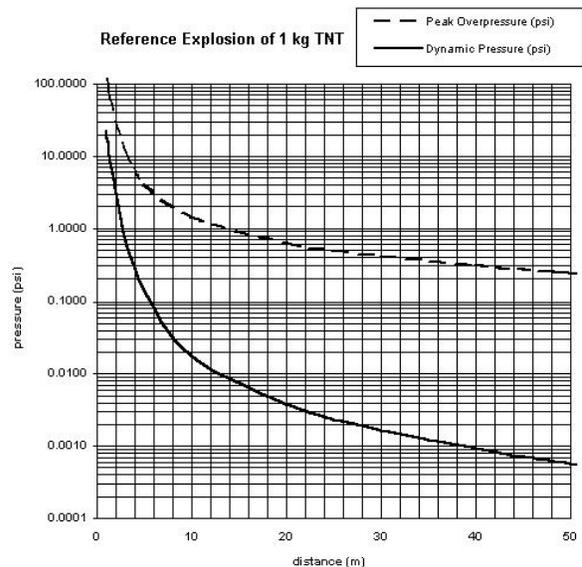


Figure 4. Reference Explosion of 1kg TNT

The next step is to calculate the force on the aircraft caused by the total pressure. This is derived by multiplying the pressure times the area affected. The exact aircraft aspect and the actual blast wave are rarely perpendicular to the target surface. In order to account for this, the damage calculation process analyzed angles-of-incident and applied damage accordingly. The MiG-29 was broken down into ten large sub-systems (called pieces in this study), which are furthermore comprised of many small parts (for the sake of this study a part was defined as a single square or triangular facet used to approximate the shape of the aircraft). Upon receiving a damage calculation request, PBEG/PhysX iterated the effects on every part of the wireframe model. The damage on each part was applied in sum to the piece that the part belonged to. For example, after an explosion, each of the facets of the left engine were run through the algorithm. The damage returned was applied to the total left engine damage and a running total is kept. To adjust damage

for angle-of-incident, the center of each part was compared to the edges and the angle between normal vector and warhead vector calculated. This was included in the overall algorithm to determine the projected blast area.

Once the projected area was found it was multiplied by the pressure derived from Figure 4 to calculate the force caused from the blast.

$$\text{Stress} = \sigma = F/A \quad (1)$$

Next, the maximum yield force for certain parts of the aircraft were calculated to determine structural failure points. To find the maximum force that each part can sustain to failure, the mass values in Table 2 were multiplied by 9 g's (9 x 9.81m/s/s). The formula was derived from Newton's second law which is force is equal to the mass multiplied by the acceleration. The table adjusted to maximum force is shown in Table 3.

Table 3. Breaking Forces (on the MiG-29 Aircraft)

Aircraft Component (Piece)	Failure Force (N)
Engines(2)	176580
Wings(2)	176580
Fuselage	529740
Vertical Stabilizers(2)	88308
Horizontal Stabilizers(2)	88308
Cockpit	353160

The values from Table 3 were compared to the forces calculated from the explosion. This comparison was used to give a numerical value to the damage.

Kinetic Energy and Fragmentation

Fragmentation damage is inflicted from the piercing of the aircraft skin by extremely high velocity blast fragments which are small pieces of metal ejected by the explosion. From analyzing open source data, it was determined that for a typical fighter aircraft approximately 4000 Joules of energy are required to pierce the skin. From this number, since we know the maximum amount of energy aircraft skin can withstand, we can model the damage caused from the fragments.

The formula for kinetic energy is a function of mass and velocity. The mass of each fragment piece can be provided from warhead specifications. Velocity

calculation is more complicated since a fragment may eject from the blast with an extremely high velocity but slow very rapidly due to aerodynamic drag. To determine the velocity of each fragment the algorithm determined two factors: the initial velocity and the deceleration as a factor of distance. The initial velocity was calculated from the Gurney Constant, Charge-to-Metal Ratio, and K constant for cylinders. The formula for initial velocity was:

$$V = \sqrt{2\Delta E} \sqrt{\frac{C/M}{1 + K(C/M)}}, \quad (2)$$

where (C/M) = Charge-to-Metal Ratio

To find the velocity as a function of distance, the formula used included the initial velocity and the distance traveled. The distance in the formula was calculated from the XCITE fusing model. Since this distance was from the detonation point to the center of the target model, the distance used in the calculation was interpolated for each part.

$$V(S) = V_0 e^{-(\rho C_d A S / 2M)}, \quad (3)$$

Where S = distance from warhead to part

This formula includes values that change depending on certain warhead and atmospheric parameters. The coefficient of drag used is dependent on the shape of the fragments. The fragments ejected from an AMRAAM resemble small bricks which have a coefficient of drag of approximately 2.1. This value can be edited depending on the shape of fragments. The cross sectional area of the fragments changes depending on their size and shape. The value used for this test was .2 meters, which is an approximation of the size of the AMRAAM fragments. The mass used in the formula is the mass of each individual fragment piece and was calculated to be 0.1 kg for the AMRAAM. This was obtained by dividing the warhead mass into 198 individual pieces. The density of air used in the test was 1.2 kg/meter-cubed, which could be obtained from standardized atmospheric models for any altitude. The energy transferred by each fragment could therefore be obtained from the following equation:

$$\text{KE} = 0.5MV^2 \quad (4)$$

If the value produced from this formula exceeds the maximum 4000 Joule threshold then it was assumed aircraft skin penetration occurs.

Damage Quantification

Damage assessment was classified into 4 categories: no damage, light damage, medium damage, or heavy damage. Comparison of the calculated total force applied compared to the maximum force for each piece is determined the damage category. This breakdown is shown in Table 4.

Table 4. Damage Assessment Criteria

Damage Level	Damage Force
None	0% to 30% of Force Max Applied
Light	30% to 50% of Force Max Applied
Medium	50% to 100% of Force Max Applied
Heavy	Gtr than 100% of Force Max Applied

These damage force categories were refined into more detail for each aircraft piece as shown in table 5.

The damage assessment is cumulative meaning that if both the cockpit and wing receive light damage the number values will be added together to show the total damage. For this study, if that number exceeded 100 the aircraft was destroyed. This algorithm was created to allow for the user to change the set values depending on the aircraft to be used.

SOFTWARE DESIGN

Information Flow

The components of the PBEB and the XCITE Computer Generated Forces software were designed to interact seamlessly for the damage calculation

Table 5. Aircraft Damage Assessment

Aircraft Damage Level	Fuselage	Cockpit	Wing	Engine	Stabilizers
Light	10	15	15	5	5
Medium	25	40	25	25	10
Heavy	100	100	100	50	15

interface. The controlling module within the PBEB system is SAFDriver. SAFDriver coordinated the data flow between all components including transfer of data from XCITE to the PBEB, PhysX based damage calculations, visualization, and transfer of damage calculations back to XCITE.

PhysX and XCITE were the primary information sources sending entity state data to PBEB. The entity state for the target aircraft was mirrored into the PBEB and was processed in parallel with the XCITE entity state allowing realistic behavior to be modeled. Typically, SAFDriver used the entity state data from XCITE as the de facto mode of operation, especially if the aircraft was undamaged. Upon damage or a catastrophic entity kill, control of the plane was shifted to PhysX, which managed the damaged aircraft entity state. The DIS environment still maintained the original XCITE entry state until the aircraft was deleted after kill, but the engagement visualization model in the PBEB followed PhysX.

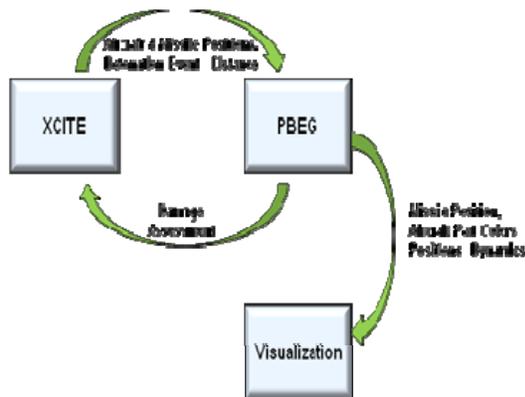


Figure 5. XCITE-PBEG Interaction and Visualization

XCITE Software Modifications

A more complicated case arises when a piece is blown off, but the plane survives. To accomplish this aircraft state, the XCITE computer generated forces software itself was modified to bypass the standard damage calculations and instead, use the result provided to it from PBEB. Also included in the XCITE modifications was code to push entity state data to PBEB in a non-DIS manner.

Shown in Figure 6 are the functional changes made to XCITE to support PBEG damage calculations. The original XCITE damage code was replaced by a function call to the PBEG interface, a delay, and then follow-up calls to the PBEG interface. No changes were necessary to the XCITE Instructor Operator System.

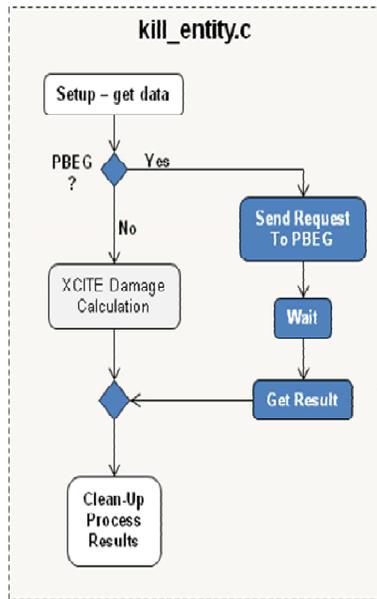


Figure 6. Modification to XCITE Damage Calculation

ENGAGEMENT VISUALIZATION

The visualization component within the system is optional but is unique in that it leverages the GPU's graphics capability directly for parallel visualization of the engagement. The system directly renders the wireframe representations of the aircraft and missile throughout the engagement. After the engagement occurs, each part of the plane is colored to reflect the resultant damage level:

- Green: No damage
- Yellow: Slight damage
- Orange: Moderate damage
- Red: Part failed or detached from main airframe

The visualization used its own network library that it shared with SAF Driver. Visualization leveraged the hardware's OpenGL capabilities. The visualization process is demonstrated in the following Figures x-x.

FINDINGS

PBEG Assumptions and Limitations

Certain potentially important calculations were not performed by the system which may affect an actual missile engagement. Additional damage caused by aerodynamic forces due to structural bending resulting from a blast were not included. The open source data used may also limit these results. All flow was assumed to be subsonic and the linear drag model used may not be accurate at high airspeeds. Partial damage to aircraft skin was not modeled, only complete penetration.

NVIDIA PhysX limitations

Due to limitations in the implementation of the code for this study, the calculations inside the PBEG were uncoupled after PBEG took control of the engagement. Any aircraft reaction that takes place after PBEG assumes engagement control would not be reflected in the final output, for example, an aggressively flown counter-threat tactic. Future efforts may attempt to precisely tune PBEG to the target aircraft model for direct coupling at a faster frame rate. The actual forces to break joints and seams in the aircraft structure were not capable of being modeled in PhysX. Finally, the method for calculating impact or fusing relied on the host XCITE simulation. While XCITE runs at 60 HZ frame rate, fusing errors in the magnitude of single digit feet are possible. For improved accuracy, these models will later be moved into the PBEG.

System Latency

The damage calculations in this study took approximately 3 frames, with a fourth frame used by XCITE to manage the simulation results, bringing the total time cost of a frame with damage calculations to approximately 60 milliseconds. While this does represent an XCITE frame overrun, it was mandated by the structure of the XCITE software's kill entity functions. The 4 frames constitute a slight delay that becomes insignificant in the course of a normal operation due to the extremely low likelihood that any one specific frame requires a damage assessment.

Because the default state is for XCITE to assume an aircraft was alive (this was done to prevent deadlocking in the case that PBEG fails or does not start properly), it is possible that on slower computers the wait period may expire before PBEG has a chance to reply. This can be recognized by PBEG indicating the plane was killed while the plane remains alive in

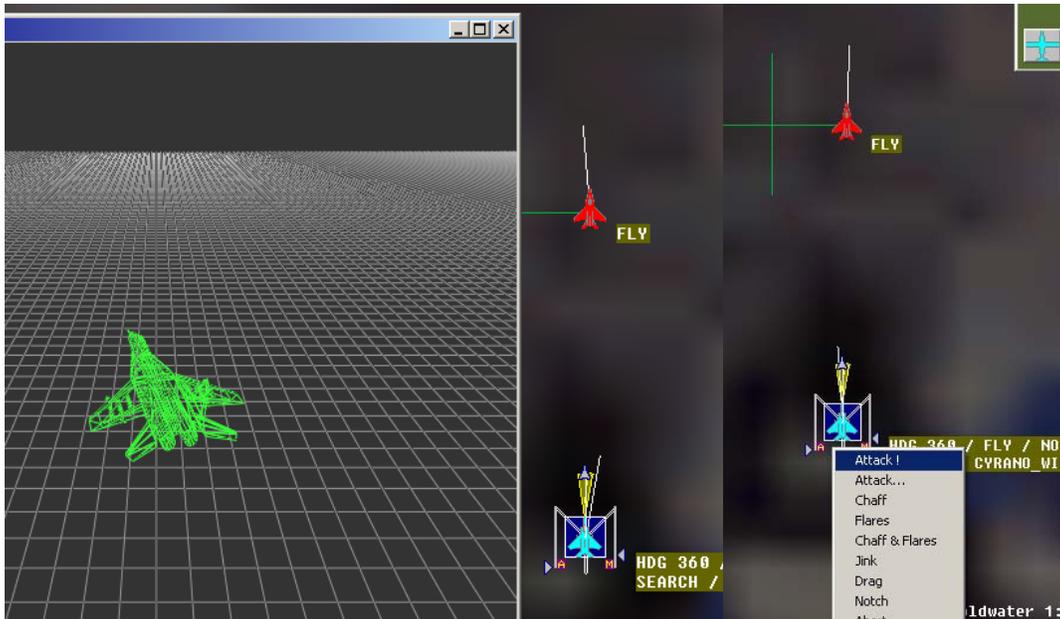


Figure 7. PBEG Model (left), XCITE Engagement Initialized (right)

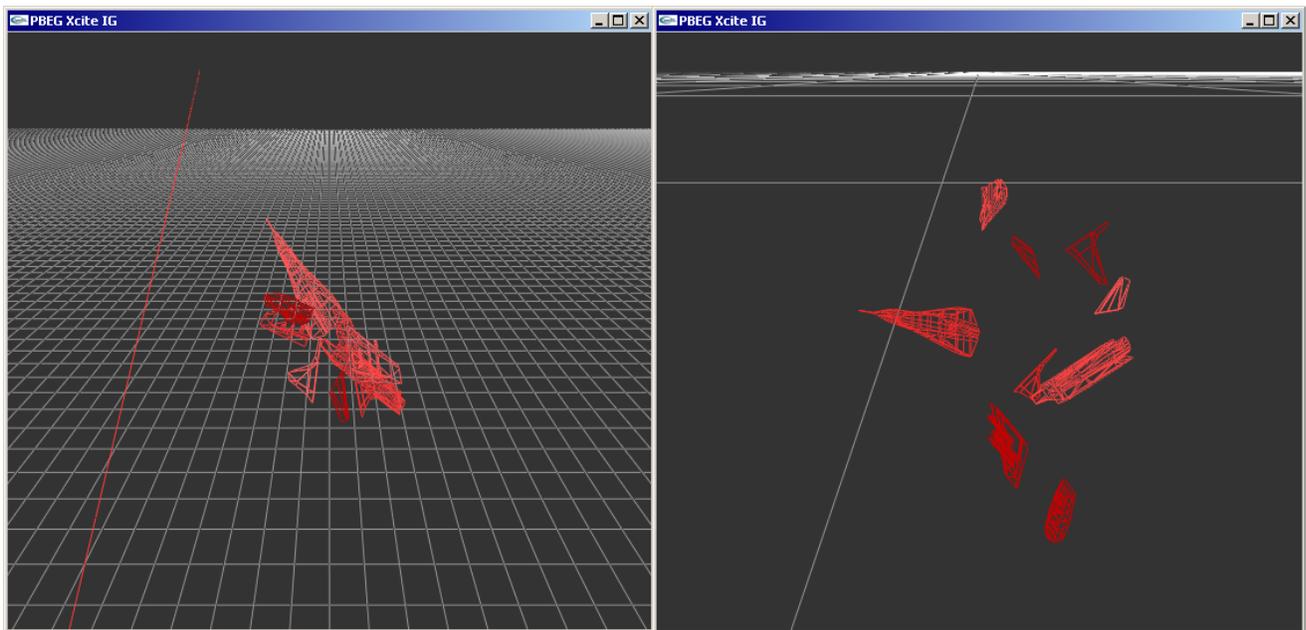


Figure 8. Missile Impact and Target Breakup

XCITE. The situation can be corrected by either using a more powerful computer to host PBEG or, more practically, by increasing the delay value.

STUDY CONCLUSIONS

From this study it was shown that approximate real-time damage calculations can be attained using a physics based approach. Physics calculations were used in conjunction with NVIDIA PhysX to capture a more realistic explosion. Because of the time

constraints placed on the calculations, some loss of accuracy is expected from the algorithm derived, especially when all aerodynamic forces are not included. This approach does allow pilots and trainers to receive near-instant feedback of the effects of ordnance, with measured performance improvements over other approaches for warhead modeling. Additionally, this approach can generate a detailed visualization of the engagement for immediate feedback as well as pilot debriefing or technical troubleshooting. This near real time simulation and representation of the damage is capable of providing improved feedback for better training of pilots.

FOLLOW ON STUDIES

Improving PBEG

Follow-on studies are planned to improve the system's capabilities and transition the system into actual flight simulators and distributed training networks for evaluation. Comparison of the PBEG's effects with detailed offline calculations of weapons detonations will be conducted to validate and improve PBEG model accuracy. A large number of DoD weapons

engagement and survivability models exist that can be used for validation or to derive improved source data (SURVIAC, 2010). Incorporating some of these engineering level models into a PPU or GPU based framework may yield results usable for a real-time training system application like the PBEG. Additional aeronautical models for both the missile and aircraft will be developed allowing a more coupled simulation between XCITE and the PBEG, as well as a capability to model aerodynamic forces after a detonation. PBEG may be used to develop a more robust network damage server capable of servicing requests from multiple XCITES and simulator hosts simultaneously.

Modeling Directed Energy Attack

The ray tracing and graphics processing capabilities in the GPU make it an ideal platform to model directed energy attacks against dynamically articulating targets in distributed simulations. AFRL and Link Training are developing a directed energy weapons server based upon the PBEG architecture. This system will implement a realistic, real-time simulation of the interactions between a directed energy (DE) beam and a target entity (maintained by XCITE). The system under development includes a framework to execute a

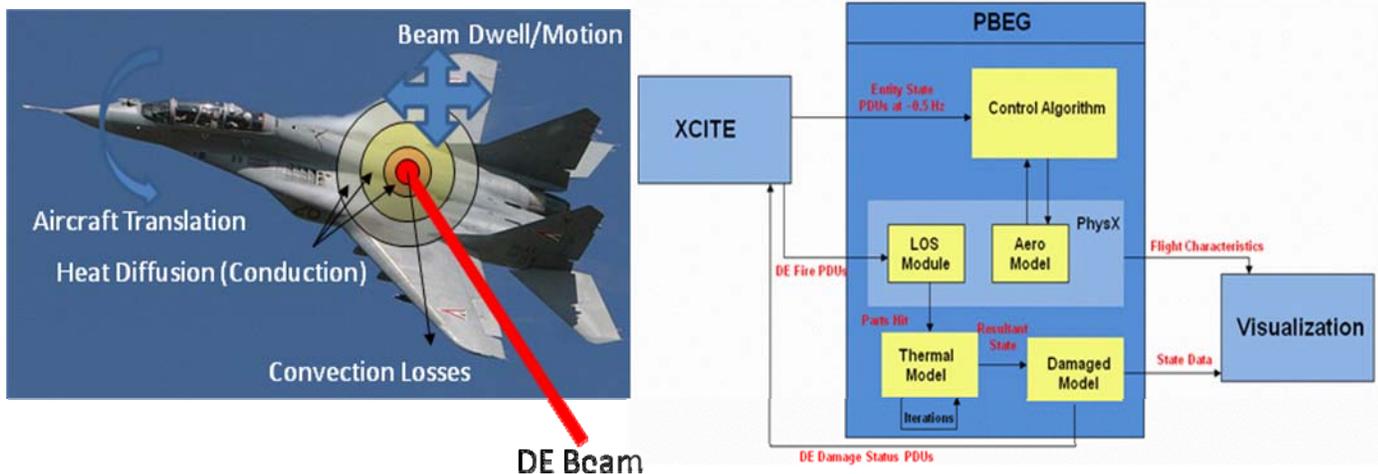


Figure 9. PBEG Directed Energy Attack System Model

thermal damage model and supports drop-in thermal, atmospheric, target, and DE beam models. A real-time interface with XCITE will provide accurate DE engagement feedback and the results will be available in a visualization.

ACKNOWLEDGEMENTS

The authors would like to thank Mr. Timothy Kochert and Mr. Dean Lewandowski, Link Simulation & Training, Warfighter Training Engineering Support, Mesa Research Site, for their invaluable assistance in coordinating efforts between Link Simulation & Training Division, Orlando, FL and Air Force Research Laboratory, Mesa, AZ, as well as their

outstanding technical support for XCITE software modifications required for the project.

REFERENCES

Federation of American Scientists, (2000). *AIM-120 AMRAAM Slammer*. Retrieved from <http://www.fas.org/man/dod-101/sys/missile/aim-120.htm>.

Purcell, T., Buck, I., Mark, W. R., & Hanrahan, P. (2002). Ray Tracing on Programmable Graphics Hardware, *ACM Transactions on Graphics*. 21 (3), pp. 703-712, 2002.

Verdesca, M., Munro, J., Hoffman, M., Bauer, M., & Manocha, D., (2005). Using Graphics Processor Units to Accelerate OneSAF: A Case Study in Technology Transition, *Interservice/Industry Training, Simulation, and Education Conference (IITSEC) 2005*.

United States Department of Defense, Joint Aircraft Survivability Program (JASP) and the Joint Technical Coordinating Group for Munitions Effectiveness (JTTCG/ME), (2010). *SURVIAC Model Guide*.

United States Navy, (1998). *ES310: Introduction to Naval Weapons Engineering, Warheads* (pp 382-387).

Wright, D.L. & United States Department of Defense, Joint Advanced Distributed Simulation Joint Test Force (1999). *The Utility of Advanced Distributed Simulation for Electronic Warfare Systems Testing*.