A Counter Rocket, Artillery and Mortar System with Laser Simulation Software

Maria Epp, Hendrik Rothe
Institute of Automation Technology,
Measurement and Information Technology
Helmut-Schmidt-University
Hamburg, Germany
maria.epp, rothe@hsu-hh.de

***Abstract* — Several countries develop laser weapons to use them for protecting critical infrastructures. Before the weapon is used for protection, the decision must be made, whether it is beneficial to use a laser instead of existing weapons. In order to make this decision, one must run several tests under varying conditions. These tests are not only very expensive, but also difficult to organize. This paper describes the Counter-RAM with laser simulation software. It simulates different attacks of rockets, artillery or mortar against the protected area. The simulation can simulate attack on the protected territory, the detection and tracking of missiles. It can classify the projectile as danger and simulate the intercepting of this projectile. This paper describes the development of the simulation.**

Keywords-C-RAM; simulation; RAM intercept; Laser weapon system

I. INTRODUCTION

Mortars and rockets are common weapons of insurgents. The inexpensive projectiles are fired at the defended area, where they can damage the infrastructure and kill or injure numerous people [1].

A Counter Rocket, Artillery and mortar system (C-RAM) is a defense system for providing warnings to vulnerable assets and for intercepting RAM threats in the air [2]. The system considered in this paper is equipped with a radar system and high energy laser weapons. Fig. 1 demonstrates an attack and engagement scenario: A launched mortar is detected by an acquisition radar. During the tracking, the trajectory of the projectile is predicted. If the projectile is classified as a target, a laser weapon is assigned to it. Once the laser is aligned to the target, optical tracking of the projectile is started. The laser weapon is activated and the target is destroyed in flight [3].

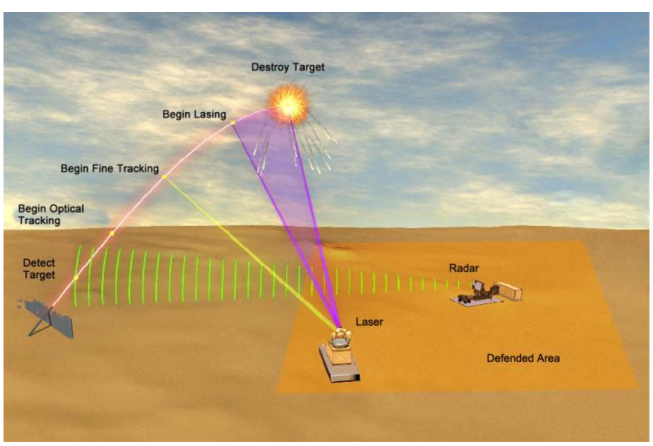

Figure 1: Typical engagement scenario (adapted from [3])

Only projectiles are modeled in the simulation, which are unguided and do not have sensors, that direct the flight path. The average flight time of such projectile is 25-35 seconds.

In the simulation, the projectile mass is concentrated at one point and is affected only by the force of gravity. Fig. 2 shows the simulated trajectory of a projectile: $\vec{v}_0$ is the muzzle velocity of the projectile, ϑ_0 its elevation, $\vec{F}_G$ is the force of gravity, $\vec{v}$ is the velocity, ω is the angle of sight.

The concept of the simulation is presented in [4]. Knapp and Rothe [4] describes the basic idea of a simulation for CRAM with laser weapon. This idea has been adapted and for this simulation and developed.

Section II describes the program flow of the simulation. The listing of the selected tools for programming can be found in Section III. Section IV will talk about possible advancements and extensions of the program.

II. SIMULATION DESCRIPTION

The simulation is split into two separate parts. The first part simulates an attack of a protected asset and the second part performs the defense of this area.

The software is divided into several individual modules.

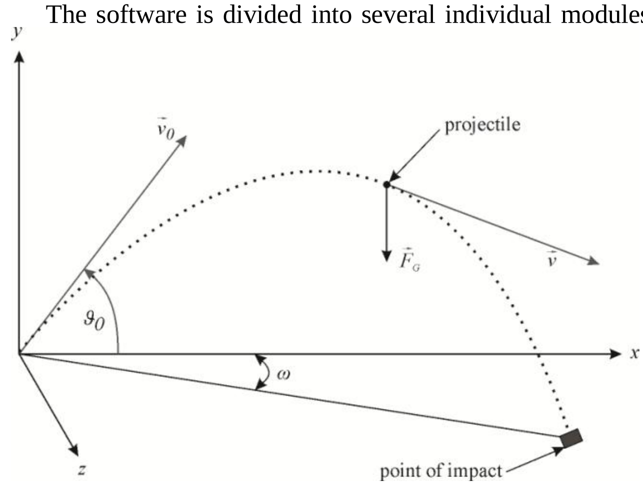

Figure 2: Trajectory of an unguided projectile

 ISBN: 978-1-61208-304-9

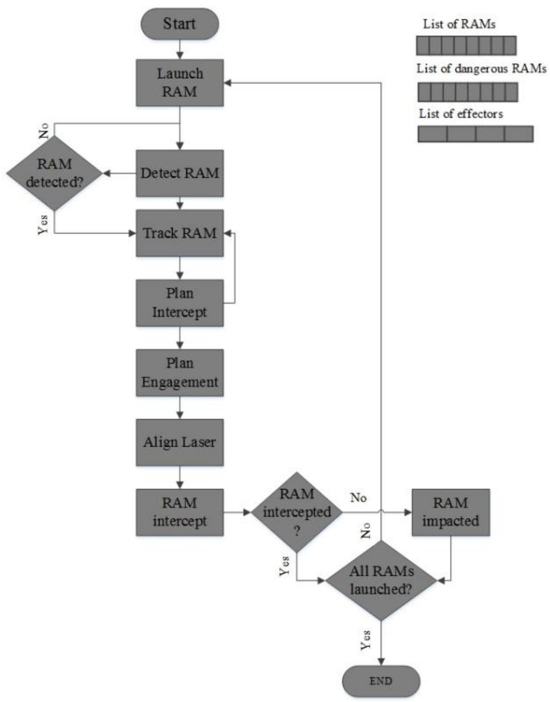

Figure 3: Workflow of the simulation

There is an internal database and a Graphic User Interface (GUI). A user has two options to start the simulation:

1) Enter all new relevant parameters for simulation via the GUI. Before the simulation starts, the parameters will be saved in the database.

2) Load existing simulation parameters. The parameters can be changed and saved as new simulation data or the old data set can be overwritten.

The workflow of the simulation is illustrated in Fig. 3. The simulation starts by firing of first projectile.

A. RAM Launch

This module simulates an attack on the protected area.

All RAMs are saved in a list and are sorted by firing time. At the beginning of the simulation, the trajectory for each projectile in the list is calculated. As of now, only vacuum trajectories are taken into account in the simulation.

The (x, y, z) –coordinates of the projectile, with respect to time, are calculated using equations (1), (2), and (3) [5].

$$x = \vec{v}_0 t \cos \vartheta_0 \quad (1)$$

$$y = \vec{v}_0 t \sin \vartheta_0 - {}^1\!/_2\, g t^2 \quad (2)$$

$$z = x \sin \omega \quad (3)$$

Once the first RAM in the list is launched, the simulation is started.

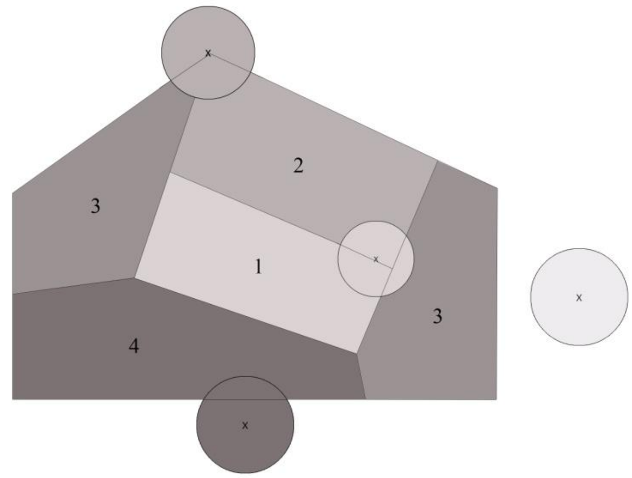

Figure 4: Defended area and impact points of the projectile with bursting radius: 1-4 are priorities of districts. 1is highest, 4 is lowest priority.

B. RAM Detection

Hence, the simulation of the defense of the asset begins.

The detection radar is located in the middle of the area, which is to be defended. The radar has the angle of sight [°], the radar range [m], and the detection rate [s]. An arbitrary number of RAMs can be detected by the radar. To determine whether the projectile is inside the radar range, the distance between the radar position (x_r, y_r, z_r) and the current position of the projectile (x_p, y_p, z_p) is calculated by using equation (4). The calculated distance is compared to the radar range. If the distance is less than the radar range, the projectile is classified as detected.

$$d = \sqrt{(x_r + x_p)^2 + (y_r + y_p)^2 + (z_r + z_p)^2} \quad (4)$$

C. RAM Tracking

As soon as a projectile is detected, the tracking of its trajectory starts. The tracking rate is defined by the user. An arbitrary number of projectiles can be tracked simultaneously.

The radar data is used in the trajectory prediction model [6, 7]. During tracking, the time and coordinates of the impact point are calculated [4], recurrently.

The prediction model needs several tracking datasets to calculate the trajectory. Therefore, only the projectile is tracked at first. Once enough data is available the first prediction coordinates and prediction times are determined. The more tracking data is available, the more accurate the predicted impact point will be calculated.

D. Interception Planning

Fig. 4 shows a defended area, which is split into several districts. Each district has a different priority. As soon as the first predicted impact point has been calculated, it is decided whether the projectile is a threat for the protected area. It is determined, whether the calculated point of impact, with its bursting radius taken into account, is inside the defended zone. If the bursting radius is located outside of the area, it is discarded. If the radius intersects with districts of several priorities, the highest priority will be assigned to the threat level of the projectile.

If more than one projectile is classified as a threat, these will be saved in a list and sorted by threat level.

E. Engagement Planning

If more than one projectile is identified as a threat, it must be decided in which order the projectiles should be fought. In the simulation software, the user has the choice between three engagement principles:

1) *Interception by first in first out (FIFO) principle:* The first detected projectile will be intercepted first.

2) *Interception by intercept time:* The projectile with the shortest intercept time will be intercepted first.

3) *Interception by priority of projectile:* The projectile with the highest priority in the list will be intercepted first.

F. Laser Assignment

After the order of engagement is determined, a laser weapon must be allocated to each projectile.

The weapon assignment problem is a fundamental problem of battle management. The problem is to assign weapons to RAM-threats in an optimal way. The expected damage to the protected area has to be minimized [8, 9].

At that moment, the weapon, that can be directed to the target the fastest, will be chosen. During the engagement analysis, the laser weapon is assigned to a threat. The laser can be assigned to another threat as long as it is not activated. Once the laser is activated, it is blocked for other threats. The projectile is destroyed within 3-8 seconds of engagement time, depending on distance to the threat. Because the more is the projectile to the laser weapon, the longer laser takes to heat the threat, due to the scattering of laser light.

G. Program End

The program terminates, once all projectiles from the list of RAMs have been processed. The number and the fighting time of the intercepted projectiles are displayed and recorded. For the impacted projectiles in the protected area, the fraction of damage is calculated. For each of the laser weapons, the consumed energy is recorded.

III. SIMULATION DESIGN

1) Time continuous simulation: The program is based on time continuous simulation to make the simulation realistic. Thereby, the variables of the program are varied at defined points of time. At the beginning, the user can define the time step of the simulation via the GUI module. Each module is started at defined time steps.

2) Material:

a) C#: This is an object-oriented programming language, which has been developed by Microsoft. It is a widespread programming language which is why there are many internet communities to help with the programming, and also there is good support from Microsoft. Visual Studio 2012 [10] has been chosen as the programming environment.

b) NUnit: This is an open source unit testing framework from Microsoft. NUnit tests the modules of computer programs for correct functionality.

c) Windows Presentation Foundation (WPF): The GUI is developed with WPF [11]. It is a graphic framework for Windows-based applications. WPF uses DirectX [11]. WPF is based on the Extensible Application Markup Language (XAML).

IV. CONCLUSION

Each module is tested with NUnit individually. Once the modules are fully functional, the next step is to combine all modules and to test common systems with several different scenarios.

The existing simulators combat the approaching targets with artillery [12, 13]. The David's Sling System is the state of the art by C-RAM systems, based on interception by artillery [14]. The in this paper described system simulates the fighting of RAMs with laser weapon.

After a number of tests have been performed, the statement must be made whether or not it would be beneficial to adopt the laser weapon as a defense weapon.

REFERENCES

[1] M. Libeau, "Laser Counter Rocket, Artillery, and Mortar (C-RAM) Efforts," NAVAL SURFACE WARFARE CENTER DAHLGREN DIV VA, January 2012, pp. 82–85.

[2] C. Corbett, B. Beigh, and S. S. Thompson, "Counter-rocket, artillery, and mortar (C-RAM) joint intercept capability: shaping the future joint force," Fires, March-April 2012, pp. 46–54.

[3] J. Schwartz, G. T. Wilson, and J. Avidor, "Tactical High Energy Laser," SPIE Proceedings on Laser and Beam Control Technologies, vol. 4632, January 21, 2002.

[4] M. Knapp and H. Rothe, "Concept for Simulating Engagement Strategies for C-RAM Systems using Laser Weapons," Conference on Defense and Military Modeling & Simulation. San Diego, CA, 2012.

[5] D. E. Carlucci and S. S. Jacobson, "Ballistics: Theory ans Design of guns and ammunition," CRC Press, 2008, pp. 195-202.

[6] M. Graswald, I. Shaydurov, and H. Rothe, "Analysis of weapon systems protecting military camps against mortar fire," Computational ballistics III, Southampton: WIT Press, 2007, pp. 21-30.

[7] I.Shaydurov and H. Rothe, "Hitting with the first shot: miniaturized fire control computer with digital signal processors" Duesseldorf: Wiedemeier & Martin, vol. 17, 2007, pp. 32-36.

[8] R. K. Ahuja, A. Kumar, K. J. James, and J. B. Orlin, "Exact and Heuristic Methods for the Weapon Target Assignment Problem," MIT Sloan School of Management, Working Paper 4464-03, July 2003.

[9] A. Toet and H. de Waard, "The Wepon-Target Assignment Problem," TNO Human Factors Research Insitut, February, 1994.

[10] Microsoft, http://www.microsoft.com/visualstudio/eng/products/visual-studio-ultimate-2012, 2013

[11] Microsoft, http://msdn.microsoft.com/library/vstudio/ms754130, 2013

[12] M. Knapp, M. Graswald, and H. Rothe "Simulation for Detrmining Engagement Strategies for C-RAM systems," Proceeding of the 2011 Summer Computer Simulation Conference, 2011, pp. 168-174

[13] M. Graswald, I. Shaydurov, and H. Rothe. "Analysis of weapon systems protecting military camps against mortar fire," Computational Ballistics III, Ashurst (2007).

[14] "David's Sling System – First Successful Interception Test," Israel Defense, http://www.israeldefense.com/?CategoryID=483&ArticleID=1784 , 2012.