Developing Firing Table Software for Artillery Projectiles using Iterative Search and 6-DOF Trajectory Model

Pawat Chusilp*, Weerawut Charubhun, and Artit Ridluan

Defence Technology Institute (Public Organization)
47/433 Chang Wattana Road, Pakkred, Nonthaburi, Thailand 11120
*Corresponding Author: pawat.c@dti.or.th, Tel: 66 2 980 6612 to 15, Fax: 66 2 980 6688 ext 300

Abstract

This paper presents the development of the firing table software for computing firing solutions for an artillery projectile. Unlike traditional tabular firing tables, in which the firing solution is determined by interpolating standard condition data then applying correction factors, the iterative search approach employed in this study uses a trajectory model to predict the point of impact in the non-standard conditions and uses search algorithms to converge to the firing solution. The concept and method used in the firing table software are explained and applied to the 155-mm HE M107 projectile. The demonstration program was tested with test cases in non-standard conditions. The results are presented and discussed.

***Keywords**:* Firing Tables, Artillery, Trajectory Simulation, Projectiles.

1. Background

Developing a tabular firing table for an artillery system consumes considerable amount of effort that up to 200,000 trajectories may need to be computed [1]. In the past, the calculation was done by human. Then the tasks began to be handed over to the electronic computers since the birth of ENIAC [2,3]. A tabular firing table normally consists of a collection of smaller tables, of which the format can be standardized [4], or the tabular data could be represented by a family of functions estimated by regression models [5].

To use a tabular firing table to determine a firing solution for a given target, the tabular data are interpolated to obtain a solution for standard conditions, i.e. standard atmosphere, no wind, no earth's rotation, firing at sea level, assuming rigidity of trajectory, etc. Then the correction factors, which accounts for non-standard conditions, are applied to the solution [6,7]. In the battlefield, these calculation steps must be done quickly. So the graphical firing table was created to simplify the calculation but the accuracy may be compromised [8].

Today advancing computer technology enables the modern firing table software that not only can perform all tasks mentioned above, but also has much more capabilities. The most notable firing table software is probably the NATO Armaments Ballistic Kernel (NABK) [9,10], which is part of NATO's sharable software suite or S4 [11,12]. NABK employs the modified point mass and five degrees of freedom models [13] for trajectory simulation. The firing solution algorithm is used for converging the calculation to the solution through iterations of trajectory simulation [10,14]. Although significant progress has been made, much information on this software is still classified and not many detail publications are publically available.

This paper presents the effort at Defence Technology Institute (Public Organization), Thailand, to develop a firing table software to support an artillery fire control system. The 155-mm HE M107 artillery projectile, as shown in Fig. 1, is chosen for a demonstration case. The selected M107 projectile is almost 0.7 m long and weights 43 kg approximately. With propelling charge M4A2 zone 5, the maximum range reaches 10 km in standard conditions.

From the following section, Section 2 describes the method and structure of the software. In Section 3, the demonstration program was presented and tested with test cases. Finally, Section 4 concludes the paper.

Fig. 1 M107 projectile with lifting plug

2. Firing Table Software Description

2.1 Concept

The presented firing table software employs the iterative method that the firing solution is determined through iterations of trajectory simulation. The structure of the software is illustrated in Fig. 2. The software consists of Firing Table Main, Search Engine, Trajectory Model, and several supporting components.

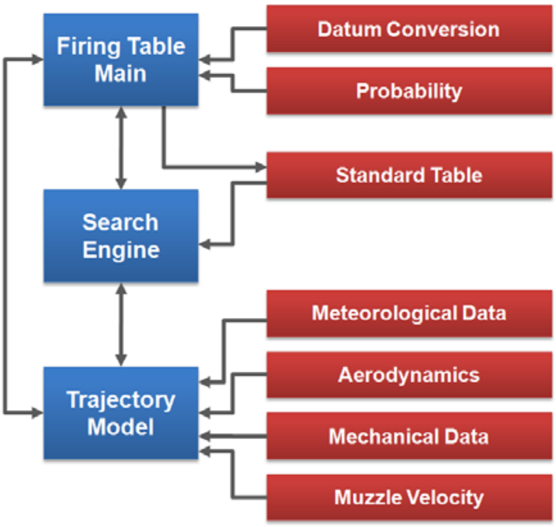


Fig. 2 Software structure model

The Firing Table Main controls the operation of the software. The Search Engine is composed of algorithms for calculating the firing solution for a given target. The Trajectory Model is used by the Search Engine for simulating the projectile trajectory and the point of impact. In addition, the Trajectory Model is also used by the Firing Table Main to generate the firing table data in stand conditions, which is stored in the Standard Table.

For the supporting components, the Datum Conversion module transforms the target and origin locations from the datum input by the user to the datum used by the Trajectory Model. The datum transformation can be done using Helmert 7-parameters transformation [15]. The Probability module estimates the probability of hit, number of rounds required to fire, and expected target damage. The calculation is based on the probable errors of the weapon, the effective radius of the warhead, target area, and shape of target area. The Meteorological Data stores the atmospheric data obtained by a weather balloon or other means. The Aerodynamics stores the aerodynamic characteristics of the projectile. The Mechanical Data keeps the information on mass, center of gravity, moment of inertia, etc., of the projectile. The Muzzle Velocity module calculates the muzzle velocity based on zone number and propellant temperature.

2.2 Iterative Search Method

The iterative search method is different from the traditionally table lookup scheme that it simulates the trajectory in the non-standard conditions of the battlefield, i.e. wind, temperature, latitude, non-rigidity of the trajectory, etc. In the traditional table lookup scheme, correction factors are applied to the solution of the standard condition to compensate the non-standard conditions. The advantage of the iterative search method over the traditional table lookup scheme is that it does not need to construct the whole tabular firing tables, which is a time consuming task. Moreover, it provides more flexibility to apply the meteorological data and other non-standard conditions to the trajectory simulation.

For indirect fire howitzers, there are only two aiming parameters to control besides propellant selection. These two parameters are the quadrant elevation (QE) and the azimuth (AZ). The Search Engine needs to find a pair of QE and AZ that produces the point of impact at the target or very close to the target.

The top-level flowchart of the iterative search method used in this study is illustrated in Fig. 3. The notation i represents the loop number.

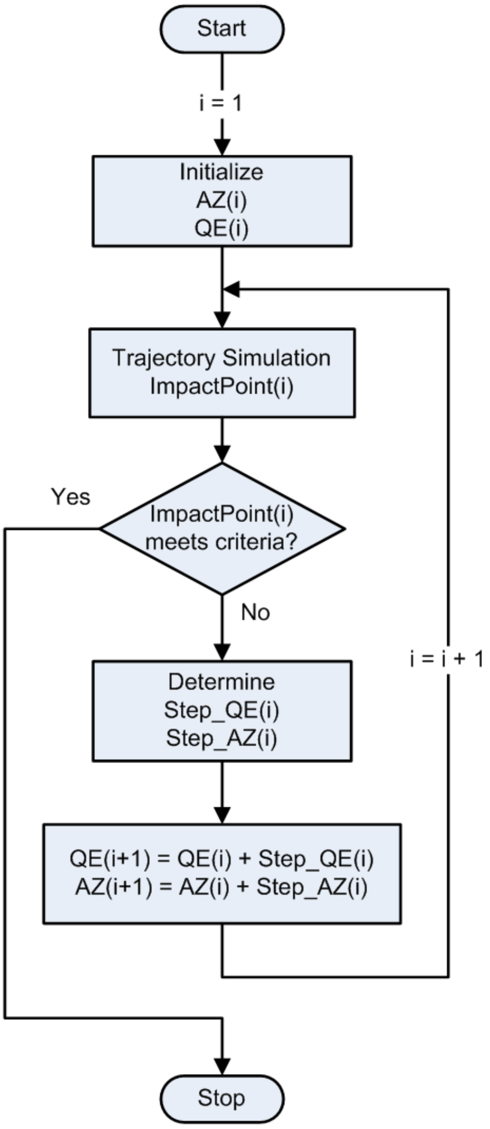


Fig. 3 Flowchart of iterative search method

In Fig. 3, the calculation begins with the initial AZ and QE, which is determined by interpolating the firing solution in the Standard Table. Alternatively, the initial AZ and QE can simply be defined by initial guess but the Search Engine may take longer to converge to the solution. Next a trajectory and the point of impact are simulated using the Trajectory Model. The miss distance, which is the distance between the point of impact and the target, is calculated. If the miss distance is greater than the given convergence criterion, the calculation is continued to the next iteration. AZ and QE for are adjusted by StepSize_AZ and StepSize_QE, which are determined by the search algorithm. If the miss distance is lower than the convergence criterion, the firing solution is found and the calculation ends.

Because each trajectory simulation take considerable amount of time, the search algorithm must be efficient that minimal iterations are required to converge to the solution. In the battlefield, time is of the essence so waiting too long for the calculation is not acceptable.

2.3 Trajectory Simulation

The six degrees of freedom (6-DOF) rigid body model was employed for trajectory simulation. The model is developed based on Newton's equations of motion. The model includes Earth's rotation and ellipsoidal shape, Magnus effect, wind, and non-standard atmosphere. The derivation of a 6-DOF model can be found in many textbooks. Khalil et al. [16] also provides good explanation of the 6-DOF model of the M107 155mm projectile.

The computation is performed in time-marching scheme. It was found that the time step 0.0001 sec is appropriate for the selected projectile. The fourth-order Runge-Kutta method was employed for integration.

Aerodynamic forces and moments were calculated using aerodynamic coefficients of the M107 projectile predicted by Khalil et al [16]. In their works, it was described that the coefficients were calculated using Prodas [17], which a well-known aerodynamics prediction code developed by Arrow Tech Inc. Fig. 4 shows an example of the trajectory plot.

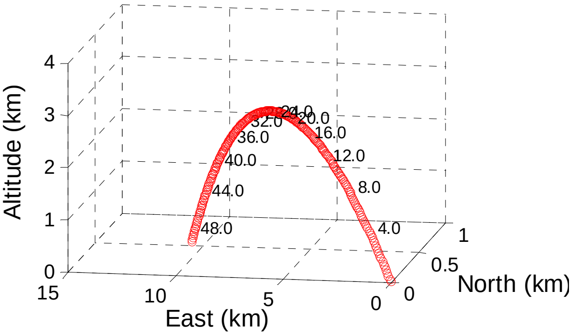


Fig. 4 Trajectory plot

3. Concept Demonstration

3.1 Demo Program

The first version of the demonstration program was developed in MATLAB environment for convenience. An example of the graphical user interface is shown in Fig. 5. Note that the input parameters in the figure are for illustration only.

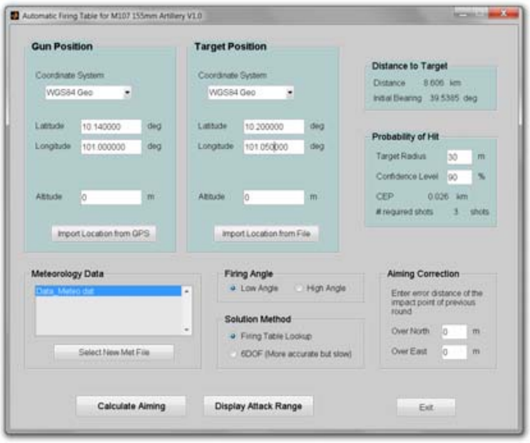
The demo program can search for either low or high angle fire. Locations of the weapon and target can be input in both UTM (northing, easting, and height) and geodetic (latitude, longitude, height) systems. The program accepts

WGS84 and Indian1975 datums, which are frequently used in the region of Thailand.

The atmospheric model used in the program is one-dimensional that the atmospheric data varies with the altitude only. A more sophisticated model, such as the four-dimensional atmospheric model [18,19], may be implemented in our future works if necessary.

The Probability and Muzzle Velocity modules have not been fully implemented in the demo program shown in this paper.

In addition, before the automatic firing program can be used, it is necessary to create the firing table data for the standard conditions, i.e. no wind, standard atmosphere [20], no Earth's rotation, assuming both target and origin at sea level. The preparation can take days but it is one-time effort. The calculated data is stored in the Standard Table and will be used for initializing QE and AZ in the search.


Automatic Firing Table for M107 155mm Artillery V1.0

Gun Position
Coordinate System: WGS84 Geo
Latitude: 10.140000 deg
Longitude: 101.000000 deg
Altitude: 0 m

Target Position
Coordinate System: WGS84 Geo
Latitude: 10.200000 deg
Longitude: 101.050000 deg
Altitude: 0 m

Distance to Target
Distance 8.606 km
Initial Bearing 39.5385 deg

Probability of Hit
Target Radius: 30 m
Confidence Level: 90 %
CEP 0.026 km
required shots 3 shots

Meteorology Data
Data_Meteo.dat

Firing Angle
● Low Angle ○ High Angle

Solution Method
● Firing Table Lookup
○ 6DOF (More accurate but slow)

Aiming Correction
Enter error distance of the impact point of previous round
Over North: 0 m
Over East: 0 m

Fig. 5 User interface of the demo program

3.2 Test Problems

The demonstration program was tested with 1,000 test cases. The origin was at randomized locations within Thailand region. The targets were evenly randomized were at range between 6 km and 10 km, directions between -180° and 180°, and altitude between -500 m and 500 m relative to the origin. Fig. 6 - 8 show the distribution of the range, direction, altitude of the test cases. Non-standard atmospheric data was used. In this test, one muzzle velocity was assumed and only low angle fire was calculated.

The convergence criterion was specified that the miss distance is less than 10 m. This criterion is reasonable considering the probable error of the unguided 155mm artillery system.

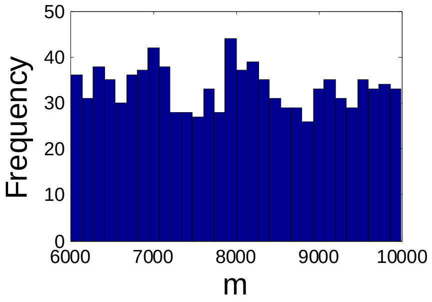


Fig. 6 Range of targets

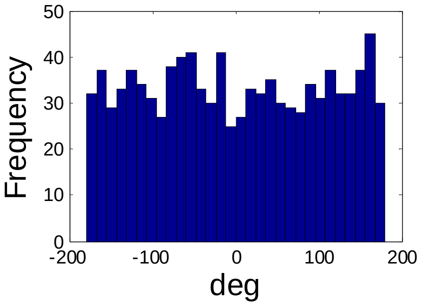


Fig. 7 Direction of targets

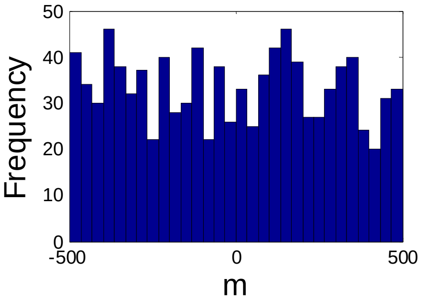


Fig. 8 Relative height of targets

3.3 Results

The results show that the demo program could find the firing solution in all cases. The average calculation time on a laptop computer (Intel Core2 Duo P8400 CPU 2.27 GHz, 4GB RAM, Window 7) was 2 min approximately. Fig. 9 shows the histogram of the calculation time. There were a few cases in which the calculation time is greater than 300 sec. So some improvement needs to be made to the Search Engine to reduce calculation time of these cases.

In addition, it was found that the calculation can be 3-4 times faster if the program is implemented and run in C# environment instead of MATLAB [21]. Therefore, the average computation time in C# should be less than 1 min. This computation speed should be acceptable for real applications.

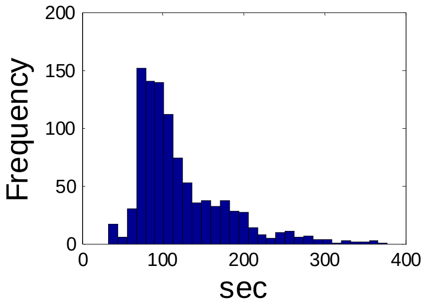


Fig. 9 Calculation time

4. Summary

The development of firing table software at Defence Technology Institute (Public Organization) was described in this paper. The concept and method were discussed. Then the demonstration program for the 155-mm HE M107 projectile was presented and tested with test cases. Overall, the preliminary results were satisfactory but several improvements must be made. Furthermore, the software is subjected to extensive testing to ensure the robustness of the methods used in the software. Our future reports will address these works.

5. Acknowledgement

The authors would like to thank Lt. Col. Prayuth Kongtunnikool and Lt. Col. Rashen Yungchum at Royal Thai Army (RTA) Artillery Center for their invaluable advice on the tabular firing table calculation procedure.

6. References

[1] Dickinson, E.R. (1967). The production of firing tables for cannon artillery, BRL Report No. 1371, Ballistic Research Laboratory, Aberdeen Proving Ground, MD, USA.

[2] Reed, H.L. (1952). Firing table computations on the ENIAC, *Proceedings of the 1952 ACM National Meeting*, Pittsburgh, PA, USA.

[3] Polachek, H. (1997). Before the Eniac, *IEEE Annals of the History of Computing*, vol. 19(2).

[4] STANAG 4119, Adoption of a standard cannon artillery firing table format, 2nd edition, NATO Standardization Agency, 2007.

[5] Sherif, Y.S., Liou, E.M.K., Chang, T.H., and Yao, S.F. (1985). Modelling the firing tables of filed artillery (cannon 105mm howitzer), *Microelectronic Reliability*, vol. 25(1), pp 41-53.

[6] Field Artillery Vol. 6, Ballistics and ammunition, B-GL-306-006/FP-001, DND, Canada, 1992.

[7] STANAG 4144, Procedures to determine the fire control inputs for use in indirect fire control

systems, 2nd edition, NATO Standardization Agency, 2005.

[8] Matts, J.A. and McCoy, D.H. (1970). A graphical firing table model and a comparison of the accuracy of three utilization schmemes, BRL Report No. 2035, Ballistic Research Laboratory, Aberdeen Proving Ground, MD, USA.

[9] Sowa, A.J. (2002). The NATO Armaments Ballistic Kernel – the 1st item of an emerging suite of software component items for fire control systems, the 7th International Artillery and Indirect Fire Symposium and Exhibition, Parsipany, NJ, USA.

[10] NATO Armament Ballistic Kernel, Brochure, AOP-37, URL: http://www.aop-37.org, accessed on 06/06/2011.

[11] Sowa, A.J. (2008). NATO shareable software developing into true suite supporting national operational, fire control systems, the 24th International Symposium on Ballistics, New Orleans, LA, USA.

[12] STANAG 4537, Sub-group 2 sharable (fire control) software suit (S4), 3rd edition, NATO Standardization Agency, 2010.

[13] STANAG 4355, The modified point mass and five degrees of freedom trajectory models, 4th edition, NATO Standardization Agency, 2006.

[14] Ortac, S.A., Durak, U., Kutluay, U., Kucuk, K., and Candanm C. (2007). NABK based next generation ballistic table toolkit, the 23rd International Symposium on Ballistics, Tarragona, Spain.

[15] STANAG 2211, Geodetic datums, projections, grids and grid references, NATO Standardization Agency, 2001.

[16] Khalil, M., Abdalla, H., and Kamal, O. (2009). Dispersion analysis for spinning artillery projectile, *the 13th International Conference on Aerospace Sciences and Aviation Technology*, Cairo, Egypt.

[17] Projectile Rocket Ordanance Design & Analysis System, Arrow Tech Associates, Inc., VT, USA, URL: http://www.prodas.com, accessed on 06/06/2011.

[18] STANAG 6022, Adoption of a standard gridded data meteorological message, 1st edition, NATO Standardization Agency, 2005.

[19] NATO Armament Meteorological Kernel, Brochure, AOP-37, URL: http://www.aop-37.org, accessed on 06/06/2011.

[20] Manual of the ICAO Standard Atmosphere (extended to 80 kilometres (262 500 feet)), 3rd edition, Doc 7488-CD, ICAO, 1993.

[21] Teotrakool, K., Mingkwan, E., and Chadil, N. (2011). Computation speed comparison of 6-DOF simulation in MATLAB and C#, *Internal Report*, Defence Technology Institute (Public Organization), Nonthaburi, Thailand.