NPS-53Ru73091A

~~NPS-53Ru73061A~~

NAVAL POSTGRADUATE SCHOOL
Monterey, California

PRELIMINARY RESULTS CONCERNING THE IMPROVEMENTS REALIZABLE THROUGH THE USE OF VARIABLE THRUST TOGETHER WITH ENGINE GIMBALING FOR A PARTICULAR INTERCEPTOR MISSILE

by

I.ra BERT RUSSAK

SEPTEMBER 1973

Approved for public release; distribution unlimited.

NAVAL POSTGRADUATE SCHOOL
Monterey, California

Rear Admiral Mason B. Freeman
Superintendent

M. U. Clauser
Provost

ABSTRACT:

Some interceptor missiles as presently formulated possess a programmed thrust magnitude history with a gimbaled engine to provide steering. We examine one such missile to determine whether performance can be improved if we allow a variable thrust magnitude together with engine gimbaling to provide control.

Two trajectory optimization programs were written to provide an initial answer to this problem. Preliminary results indicate reductions in the time to intercept by as much as thirty per-cent over that obtained by the presently used guidance scheme. With tuning of the programs it seems reasonable to expect even greater improvements and further investigation seems warranted.

This task was supported by: Job Order No. NOV017K

I. Bert Russak
I. BERT RUSSAK
Associate Professor of Mathematics

Approved by:

W. M. Woods
W. M. WOODS
Chairman
Department of Mathematics

John M Wozencraft
JOHN M. WOZENCRAFT
Dean of Research

NPS-53Ru730~~8~~91A
September 1973

Introduction

Some interceptor type missiles as presently formulated possess programmed thrust magnitude history with a gimbaled engine to provide steering. The present guidance scheme used on these missiles determines the steering control and hence the direction of the thrust vector. We examine one such missile and answer the question as to whether performance can be improved if we allow a variable thrust magnitude together with thrust direction to be controlled by some guidance scheme.

In order to take the first step in answering this question, two trajectory optimization programs were written. These were designed to determine optimal histories of thrust magnitude and direction in order to obtain minimum time to interception for our missile under given scenarios. While the programs are not in a finely tuned state, nevertheless, preliminary results indicate reductions in the time to intercept by as much as thirty per cent from that obtained by the present scheme. With tuning of the programs it seems reasonable to expect even greater improvements and further investigation seems warranted.

Model

The missile model used was two dimensional since all test trajectories were flown in a horizontal plane.

Letting the indicated terms have the meaning specified in the nomenclature, then the picture of the model is:

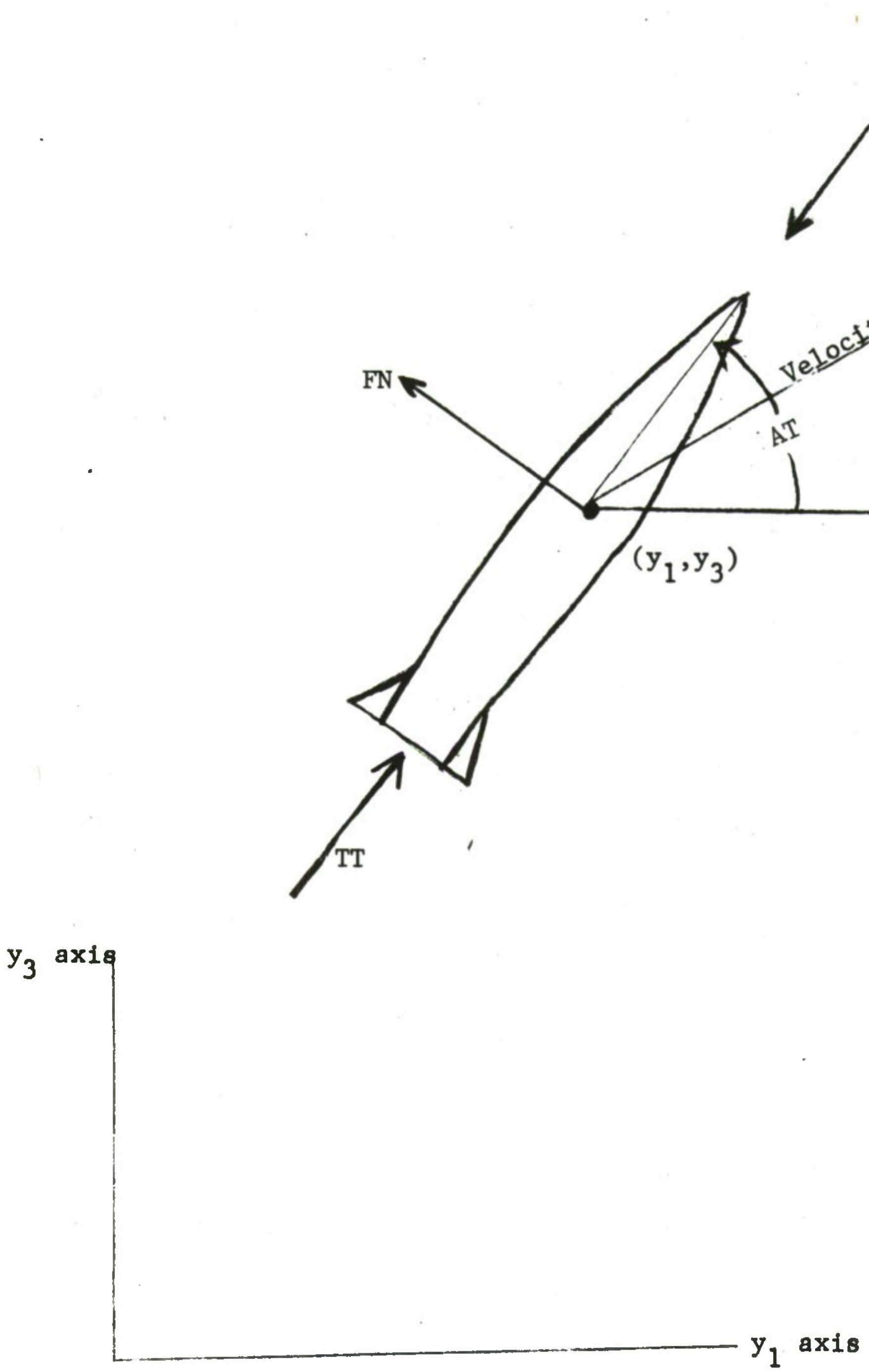


Figure 1

Missile Model

The differential equations for this model are(1) :

1a) $\dot{y}_1 = y_2$

1b) $\dot{y}_2 = \frac{TT-FA}{y_5} \cos AT - \frac{FN}{y_5} E_1$

1c) $\dot{y}_3 = y_4$

1d) $\dot{y}_4 = \frac{TT-FA}{y_5} \sin AT - \frac{FN}{y_5} E_2$

1e) $\dot{y}_5 = - \frac{TT}{8050}$

where i) $y_1, \ldots, y_5$ are called state variables since they define the state of the missile and TT,AT are called control variables since they control the state through the equations 1); ii) FA,FN are functions of the velocity vector and the control angle AT.

The constraints for this problem are

2a) $0 \leq TT \leq 14400.0$

2b) $\int_0^{TF} TTdt \leq 38,500$

in which 2a) is a thrust level constraint which says that our thrust must be non-negative and is bounded above by 14400 lbs. and 2b) is a condition on the amount of fuel used.

Our task is, given the initial conditions

3a) $y_{1_0}, y_{2_0}, y_{3_0}, y_{4_0}, y_{5_0}$

for the missile and

$y_{1T_0}, \dot{y}_{1T_0}, y_{3T_0}, \dot{y}_{3T_0}$

for the target, then determine a history of TT, AT in time which

---

(1) Detailed equations are presented in the Appendix

yields a minimum for the time of intercept TF. Using the penalty method to include the constraint of target impact in the cost function, our cost function is then

4) $$c = TF + UN[(y_1 - y_{1T})^2 + (y_3 - y_{3T})^2]$$

Method of Solution

A. General Techniques Available

There are many ways to attack a problem of the type specified above. For example;

a) the classical calculus of variations technique

b) gradient technique

c) conjugate gradient technique

Of these a) is an indirect method, which seeks a trajectory which satisfies certain necessary conditions rather than seeking to reduce the cost function directly. This method depends upon the choice of the initial values of a set of multipliers called adjoint variables which satisfy a certain system of differential equations. This choice is often a highly sensitive one and instability in attempting to converge to a solution trajectory can result.

Methods b) and c) are direct methods in that they directly seek to minimize the cost function by seeking new trajectories with lower values of cost function. All of these methods are based on generating a sequence of trajectories which converges to the minimizing one. The gradient technique works by linearizing the cost function at each trajectory of the sequence developed and iterates to the next trajectory of the sequence by changing the controls in the direction opposite to the gradient. The conjugate gradient technique is a step more sophisticated than the gradient technique in that it generates new trajectories in its sequence by effectively expanding the cost function in a Taylor Series up through the second order, thus obtaining a more accurate representation of this function.

All of these methods together with a number of others were considered for the problem at hand and because of greater sureness of convergence the conjugate gradient method was selected.

B. Brief Description of the Conjugate Gradient Technique

This method is most easily described when discussing the problem of minimizing a cost function which is a quadratic function of the N variables $x_1,\ldots,x_n$. Thus assume that we are given the problem selecting values of $x_1,\ldots,x_n$ in order to obtain a minimum of the quadratic function

5) $$c(X) = d + BX + 1/2\ X^T GX$$

where: i) X denotes the vector $(x_1,\ldots,x_n)$; ii) d denotes a constant and B denotes a constant vector; iii) G denotes the matrix of second partial derivatives of c. Given a starting point X_0 the conjugate gradient method computes a sequence of vectors H_0, $H_1,\ldots,$ along which the function c is minimized. Thus starting at X_0 the method computes a direction H_0 which depends on the cost function c and the point X_0 and determines a value X_1 which is a minimum of c in that direction. Next, a direction H_1 is computed at X_1 and c is minimized along that direction to produce the point X_2. The sequence continues in this manner and it can be shown that in the absence of round-off, the method will converge to the minimum point in at most N iterations (where N is the dimension of the vector X).

In general, as in our case, the cost function is not quadratic. The procedure then is to approximate the cost function by the first three terms if its Taylor Series at each iteration point so that it has the form of a quadratic and to develop the directions H_i from those approximations as outlined above for the quadratic case. Details of the conjugate gradient method as originally developed by Hestenes for linear systems, are in [1] and its application to general functions is explained in [2]. Furthermore, the technique of conjugate gradients works on more general functions than functions of a finite number of variables and one may apply it with some modification to functions of an infinite number of variables (see [3]). Thus for a cost function which depends upon an infinite number of variables as our cost function which depends upon the value of TT and AT at each time point, one may use this technique to seek out those values which minimize it.

C. Application of the Conjugate Gradient Technique to Our Problem

In order to apply the conjugate gradient technique to our problem, two computer programs were written.

The first of these programs was written using the conjugate gradient technique for an infinite number of variables as referred to above. This program is listed in the Appendix B and was never fully checked out due to lack of time.

The second program was written using the conjugate gradient method for functions of a finite number of variables as outlined above. Now as previously stated, the cost function for our problem depends upon infinite dimensional controls, namely the magnitude TT and direction AT of the thrust vector at each time point. However in any computing machine procedure for integrating the differential equations for our problem, only values of the controls TT and AT at a finite number of time points are used. For example, in the simplest type of integration scheme, if the time interval is denoted by DT and t_o, t_1, $t_2, \ldots t_j \ldots$ are the time points of the integration scheme then

$$
\begin{aligned}
& y(t_1) = y(t_0) + \dot{y}(t_0)\cdot DT \\
6)\qquad & y(t_2) = y(t_1) + \dot{y}(t_1)\cdot DT \\
& \vdots \\
& y(t_{j+1}) = y(t_j) + \dot{y}(t_j)\cdot DT \\
& \vdots \\
& y(TF) = y(TF-DT) + \dot{y}(TF-DT)\cdot DT
\end{aligned}
$$

where $y, \dot{y}$ denote the state variable to be integrated and its derivative and TF denotes the final time. In this scheme only the values of TT and AT at the time points t_i affect the trajectory. Thus our cost function which depends upon y at the final time in turn also depends on the values of TT and AT only at these time points.

Thus, the computer really reduces the infinite dimensional problem to a finite dimensional one. Furthermore if we take this into account in formulating our model then our numerical optimization scheme which must abide by such shortcomings of the computer, will be surer of success.

This then is the technique used to adapt the finite dimensional conjugate gradient method to our problem. The integration scheme selected is the one used on already existing trajectory computer programs for the missile under consideration and is as follows:

$$
\begin{aligned}
&y_1(t_{j+1}) = y_1(t_j) + f_1(t_j)\cdot DT + f_2(t_j)\cdot\frac{DT^2}{2} \\
&y_2(t_{j+1}) = y_2(t_j) + f_2(t_j)\cdot DT \\
7)\quad &y_3(t_{j+1}) = y_3(t_j) + f_3(t_j)\cdot DT + f_4(t_j)\frac{DT^2}{2} \\
&y_4(t_{j+1}) = y_4(t_j) + f_4(t_j)\cdot DT \\
&y_5(t_{j+1}) = y_5(t_j) + f_5(t_j)\cdot DT
\end{aligned}
$$

where we have denoted by f_i $i = 1,\ldots,5$ the right hand sides of 1). This integration scheme essentially integrates the position components y_1 and y_3 by using the first two derivatives of position, while integrating the velocity components y_2, y_4 and the mass y_5 by using only the first derivatives of these quantities.

Besides computation of the cost function at each iteration point, the conjugate gradient method requires us also to compute the derivative of the cost function with respect to the control variables $TT(t_i)$, $AT(t_i)$. By the chain rule for differentiation, this requires that we first differentiate the cost with respect to the state variables at TF and then differentiate the state variables at TF with respect to the controls at the times t_j. The former derivatives are easily formed, however the latter derivatives are formed sequentially as follows: According to the integration scheme 7) forming the derivative of $y_i(i = 1,\ldots,5)$ at t_0 with respect to $AT(t_0)$ and $TT(t_0)$ yields

$$
8)\qquad \frac{\partial y_i(t_0)}{\partial AT(t_0)} = 0 \qquad \frac{\partial y_i(t_0)}{\partial TT(t_0)} = 0 \qquad i = 1,\ldots,5
$$

Forming the derivative of y_i at t_1 with respect to $AT(t_1)$ and $TT(t_1)$ yields

9a) $$\frac{\partial y_i(t_1)}{\partial AT(t_1)} = 0 \qquad \frac{\partial y_i(t_1)}{\partial TT(t_1)} = 0 \qquad i = 1,\ldots,5$$

and next, forming derivatives with respect to $AT(t_o)$ yields

$$\frac{\partial y_i(t_1)}{\partial AT(t_0)} = \frac{\partial y_i(t_0)}{\partial AT(t_0)} + \left[\sum_{j=1}^{5} \frac{\partial f_i(t_0)}{\partial y_j(t_0)} \frac{\partial y_j(t_0)}{\partial AT(t_o)} + \frac{\partial f_i(t_0)}{\partial AT(t_0)}\right] \cdot DT$$

$$= \frac{\partial f_i(t_0)}{\partial AT(t_0)} \cdot DT \quad i = 2,4,5$$

9b) $$\frac{\partial y_i(t_1)}{\partial AT(t_0)} = \frac{\partial y_i(t_0)}{\partial AT(t_0)} + \left[\sum_{j=1}^{5} \frac{\partial f_i(t_0)}{\partial y_j(t_0)} \frac{\partial y_j(t_0)}{\partial AT(t_0)} + \frac{\partial f_i(t_0)}{\partial AT(t_0)}\right] \cdot DT$$

$$+ \left[\sum_{j=1}^{5} \frac{\partial f_{i+1}(t_0)}{\partial y_j(t_0)} \frac{\partial y_j(t_0)}{\partial AT(t_0)} + \frac{\partial f_{i+1}(t_0)}{\partial AT(t_0)}\right] \cdot \frac{DT^2}{2}$$

$$= \frac{\partial f_i(t_0)}{\partial AT(t_0)} \cdot DT + \frac{\partial f_{i+1}(t_0)}{\partial AT(t_0)} \cdot \frac{DT^2}{2} \qquad i = 1,3$$

where the last equalities in 9b) result from 9a) and where $f_i(t_k)$ means the function f_i evaluated with arguments $y(t_k)$, $AT(t_k)$, $TT(t_k)$. Similar equations hold for the derivatives with respect to $TT(t_1)$ and $TT(t_0)$. Continuing in this fashion, then at time t_k we form the derivatives of $y_i(t_k)$ with respect to TT and AT at all time points up through t_k. Forming the derivatives with respect to AT at all such times, first we set, (as in 9) the derivative with respect to $AT(t_k)$

10a) $$\frac{\partial y_i(t_k)}{\partial AT(t_k)} = 0 \qquad i = 1,\ldots,5$$

while for the derivative with respect to AT at the immediately preceeding time point t_{k-1}

10b) $$\frac{\partial y_i(t_k)}{\partial AT(t_{k-1})} = \frac{\partial f_i(t_{k-1})}{\partial AT(t_{k-1})} \cdot DT \qquad i = 2,4,5$$

$$\frac{\partial y_i(t_k)}{\partial AT(t_{k-1})} = \frac{\partial f_i(t_{k-1})}{\partial AT(t_{k-1})} \cdot DT + \frac{\partial f_{i+1}(t_{k-1})}{\partial AT(t_{k-1})} \cdot \frac{DT^2}{2} \qquad i = 1,3$$

and finally, for the derivative with respect to AT at all other preceeding time points $t_s, s = 0,1,\ldots k-2$

$$\frac{\partial y_i(t_k)}{\partial AT(t_s)} = \frac{\partial y_i(t_{k-1})}{\partial AT(t_s)} + \sum_{j=1}^{5} \frac{\partial f_i(t_{k-1})}{\partial y_j(t_{k-1})} \frac{\partial y_j(t_{k-1})}{\partial AT(t_s)} DT \qquad i = 2,4,5$$

10c) $$\frac{\partial y_i(t_k)}{\partial AT(t_s)} = \frac{\partial y_i(t_{k-1})}{\partial AT(t_s)} + \sum_{j=1}^{5} \frac{\partial f_i(t_{k-1})}{\partial y_j(t_{k-1})} \frac{\partial y_j(t_{k-1})}{\partial AT(t_s)} \cdot DT$$

$$+ \sum_{j=1}^{5} \frac{\partial f_{i+1}(t_{k-1})}{\partial y_j(t_{k-1})} \frac{\partial y_j(t_{k-1})}{\partial AT(t_s)} \cdot \frac{DT^2}{2}$$

with similar equations holding for the derivative with respect to TT at all time points. It is recognized that all derivatives of the state y required on the right hand side of 10) have already been formed at previous steps in the process.

This procedure continues until we reach TF and thus obtain the required derivatives of final state.

Since the cost function also depends upon TF, we are also required to form the derivative of the cost with respect to TF, however this presents no difficulty.

Results

In stating the results of using the above described computer program, it is to be noted that there were severe time limitations on this initial phase of the project so that only a minimal amount of time was left after formulation, development and checkout of the basic computer program. Consequently, the results presented herein are preliminary in the sense that no "tuning" (such as problem scaling) of the computer program to this problem was done. Such tuning will produce better and often very significantly better results than the basic program. Nevertheless, the results that were obtained indicate significant savings in time over those obtained from the presently used guidance scheme.

The basic missile target scenario that was used had the target at 20,000 feet initial range. Both missile and target had initial velocity of 800 feet per second. The missile heading and target aspect were varied as depicted by dashed lines in the figure below.

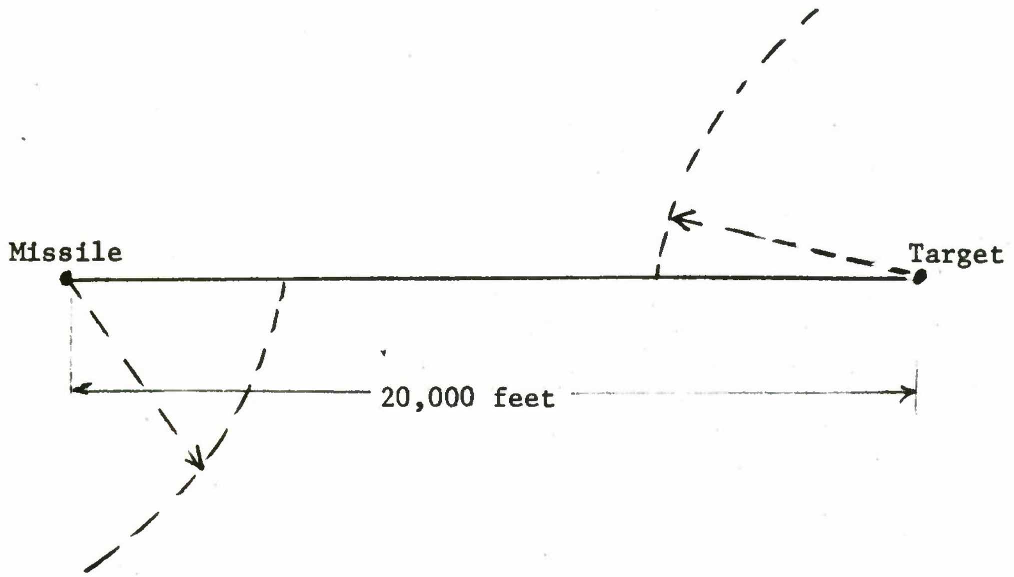


Figure 2

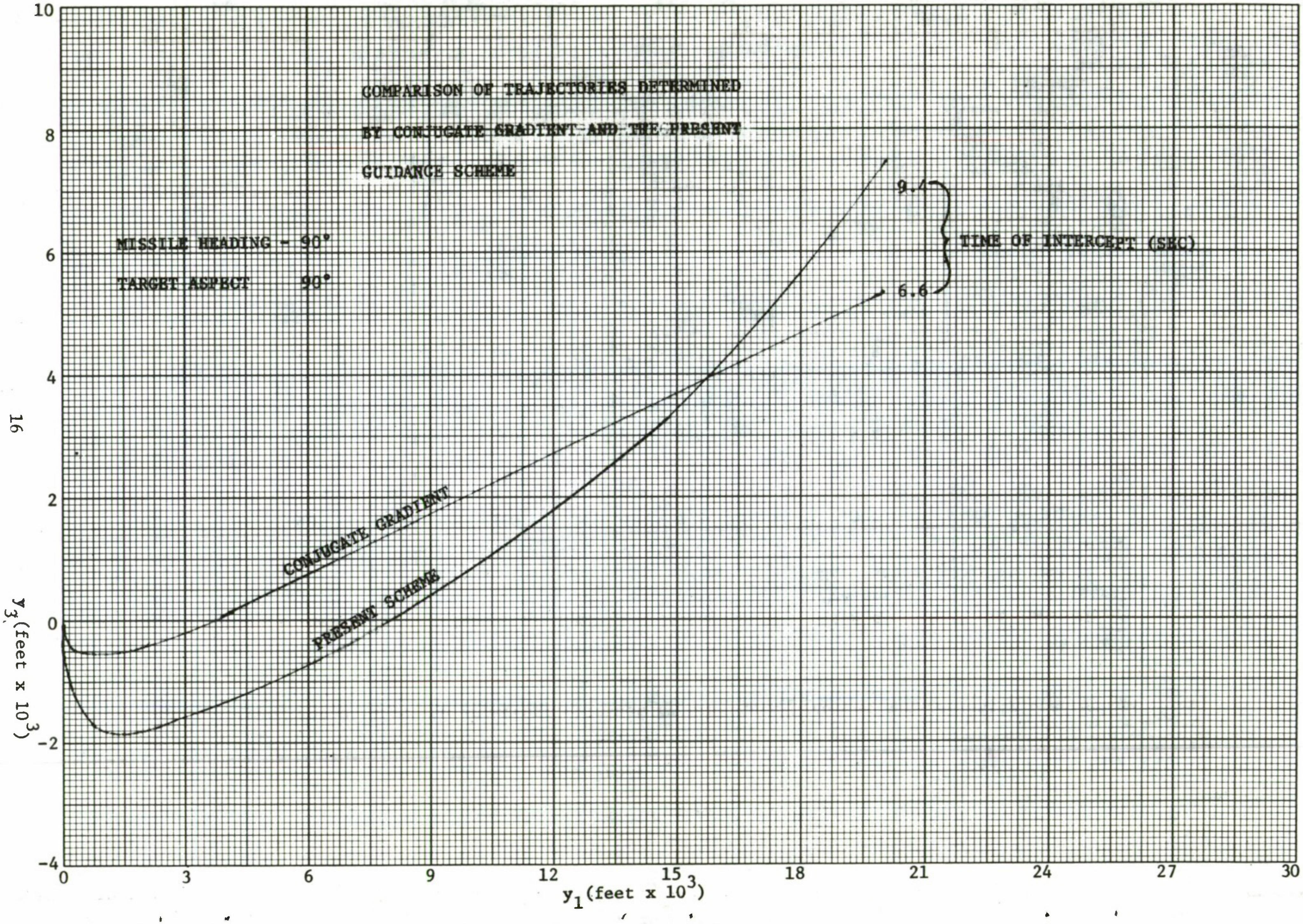
Basic Missile-Target Scenario

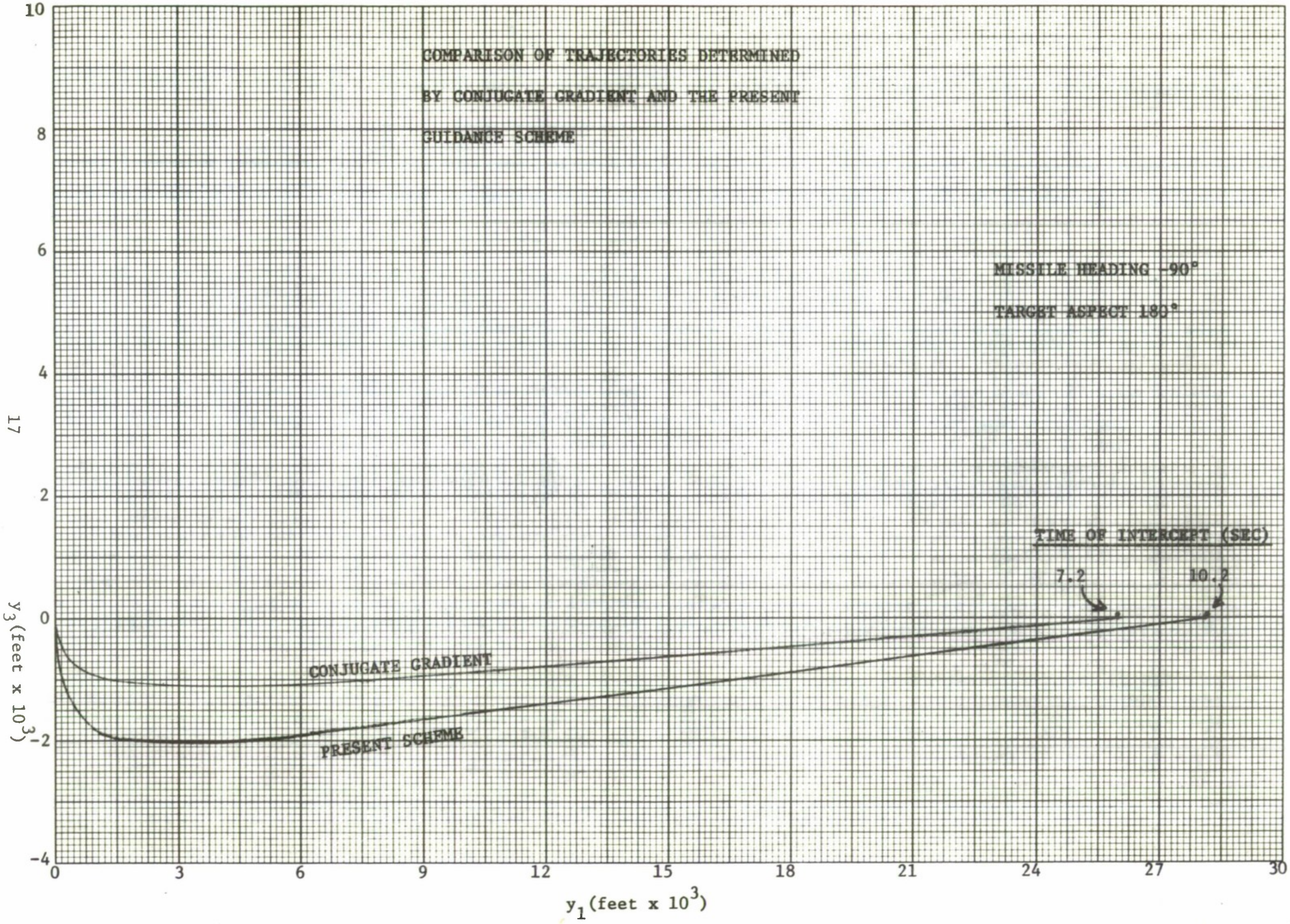
The results of the conjugate gradient runs together with a comparison to the results of the presently used guidance scheme are presented in the table which follows. In addition, plots of some of these comparison trajectories are also presented. In each plot is indicated the time of intercept with the target. Finally, the values of the control variables TT and AT at each time point t_j are listed for each plot. The number of such time points or equivalently the number of intervals in the integration process is arbitrary and was generally selected to give roughly an interval of .25 sec for the initial trajectory and time of flight which were used to start the program for each case.

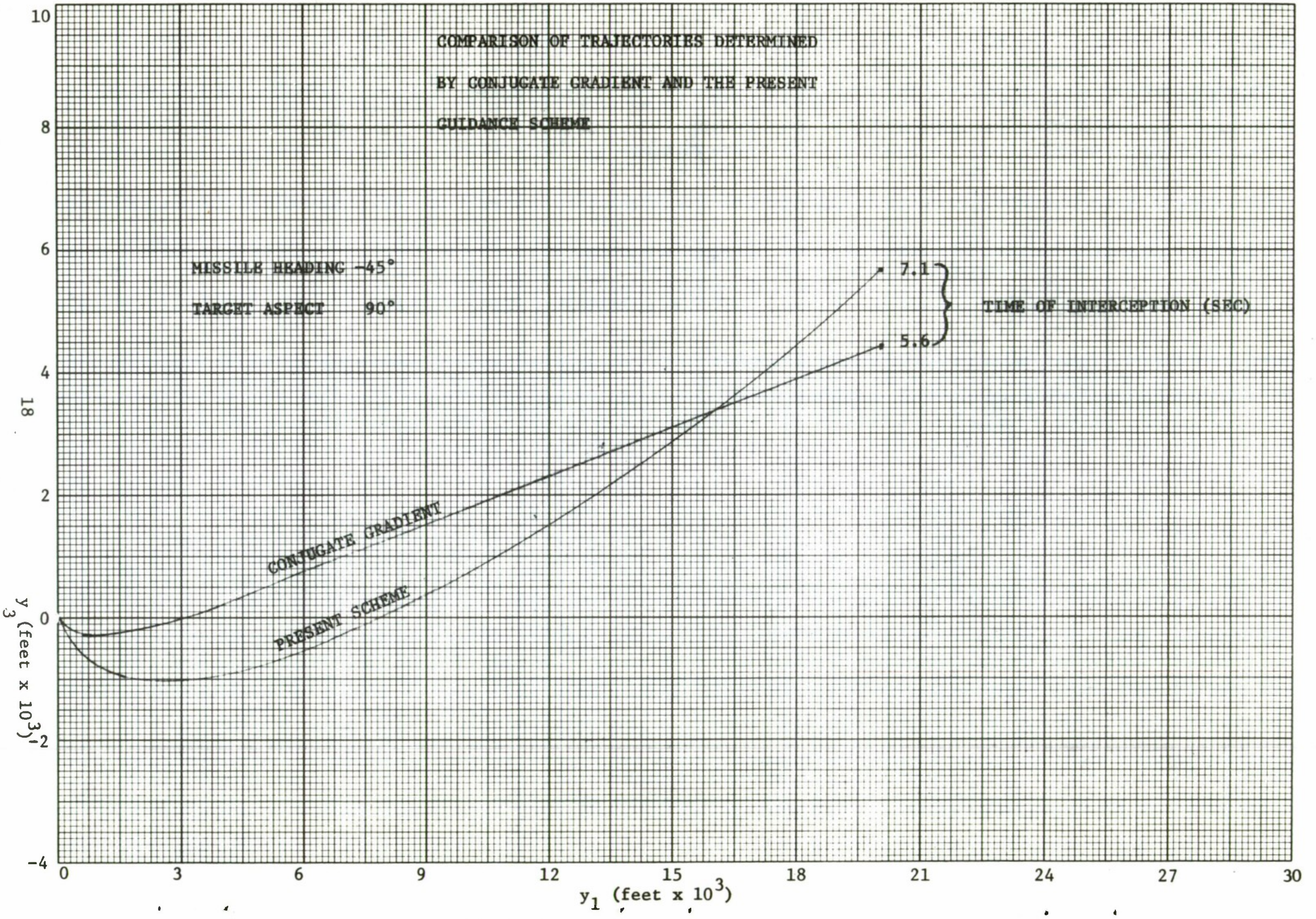
Table 1

Comparison of Times to Intercept Obtained By Conjugate Gradient and Presently Used Scheme

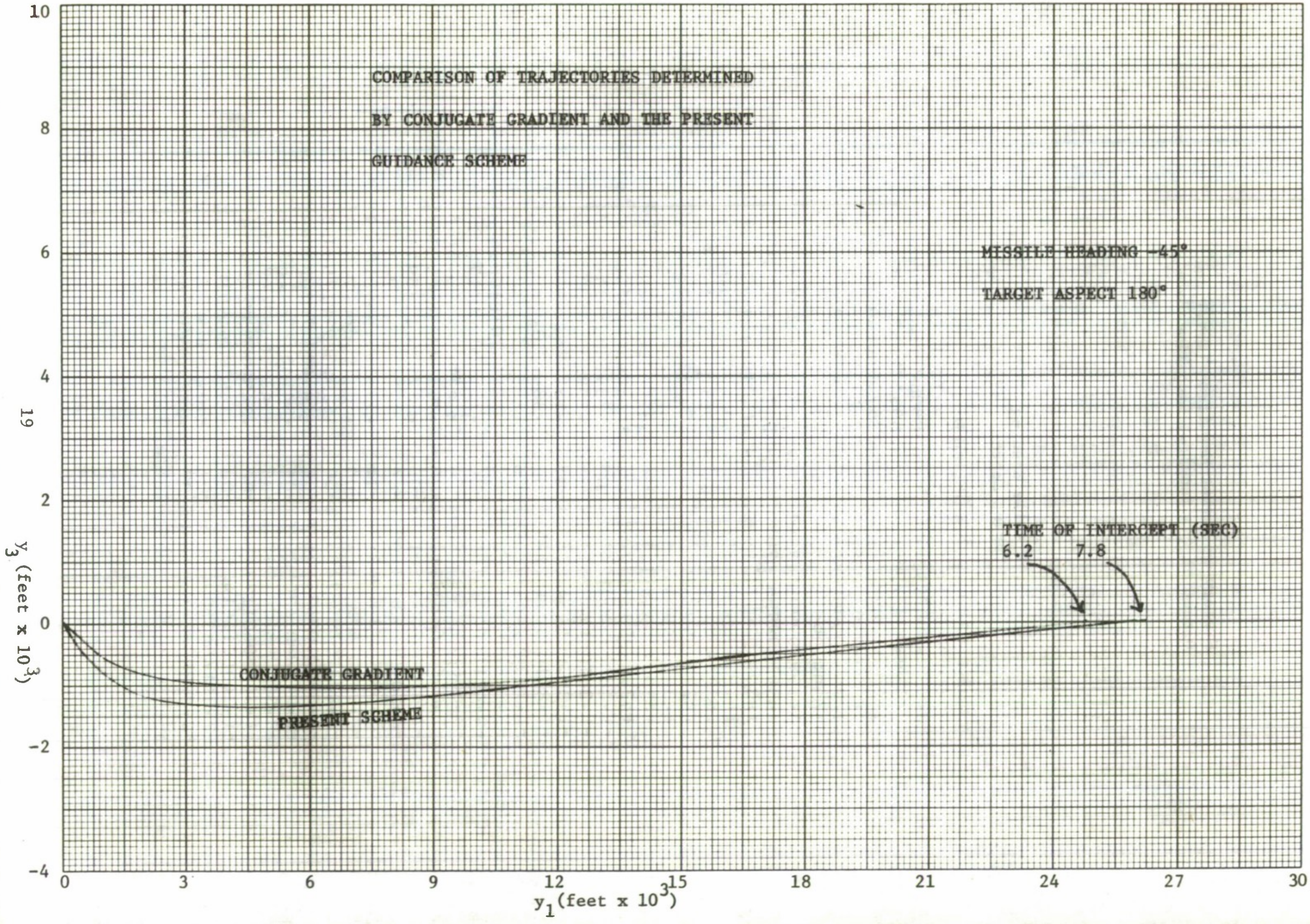
Missile Heading	Target Aspect	Conjugate Gradient Time	Time of Presently Used Scheme	% Improvement Over Present Scheme
-90°	180°	7.2	10.2	30%
-90°	90°	6.6	9.4	30%
-45°	180°	6.2	7.8	21%
-45°	90°	5.6	7.1	21%
-45°	0°	4.7	5.5	15%


COMPARISON OF TRAJECTORIES DETERMINED
BY CONJUGATE GRADIENT AND THE PRESENT
GUIDANCE SCHEME
MISSILE HEADING - 90°
TARGET ASPECT 90°
9.4
6.6
TIME OF INTERCEPT (SEC)
CONJUGATE GRADIENT
PRESENT SCHEME
10
8
6
4
2
0
-2
-4
0
3
6
9
12
15
18
21
24
27
30
y_3 (feet x 10^3)
y_1 (feet x 10^3)


COMPARISON OF TRAJECTORIES DETERMINED
BY CONJUGATE GRADIENT AND THE PRESENT
GUIDANCE SCHEME
MISSILE HEADING -90°
TARGET ASPECT 180°
TIME OF INTERCEPT (SEC)
7.2
10.2
CONJUGATE GRADIENT
PRESENT SCHEME
10
8
6
4
2
0
-2
-4
0
3
6
9
12
15
18
21
24
27
30
y_1 (feet x 10^3)
y_3 (feet x 10^3)



COMPARISON OF TRAJECTORIES DETERMINED
BY CONJUGATE GRADIENT AND THE PRESENT
GUIDANCE SCHEME


COMPARISON OF TRAJECTORIES DETERMINED
BY CONJUGATE GRADIENT AND THE PRESENT
GUIDANCE SCHEME
MISSILE HEADING -45°
TARGET ASPECT 180°
TIME OF INTERCEPT (SEC)
6.2
7.8
CONJUGATE GRADIENT
PRESENT SCHEME
10
8
6
4
2
0
-2
-4
y_3 (feet x 10^3)
0
3
6
9
12
15
18
21
24
27
30
y_1 (feet x 10^3)

History of Thrust Magnitude (lbs.) and Direction (Radians) at Each Time Point

Missile Heading -90°
Target Aspect 90°

```
THRUST USED      ANGLE USED
-4.605009591450599E-06    4.723121598960707
 14399.99997933899       .3859842903584423
 14399.99997969224       .3803681117161034
 14399.99998002296       .3767565214444642
 14399.99998032638       .3748835541679647
 14399.99998060886       .3742062958686325
 14399.999980874     .3739137777922477
 14399.99998117321       .3716230858999721
 14399.99998147072       .3662140979628795
 14399.99998178906       .3611019836883072
 14399.99998213868       .3562308881011089
 14399.99998253284       .3516144227031929
 14399.99998298858       .3472233046931528
 14399.99998352771       .3429874251066726
 14399.99998417795       .3387945142194222
 14399.9999849742        .3344809309495753
-1.484135400174198E-05    .3395337541740675
-1.367361742331171E-05    .3383031815255845
-1.293201954702595E-05    .3374743729385739
-1.201623031575876E-05    .336939188427191
-1.112597441374629E-05    .3366259984016299
-1.026101913346218E-05    .336457010192663
-9.421166493039155E-06    .3364904484178823
-8.606247083233235E-06    .3366156214075165
-7.816116653991806E-06    .3368497515614064
-7.050653059250109E-06    .337185911259483
-6.309754339144073E-06    .3376216675022208
-5.593337347902797E-06    .3381581899218649
-4.901336836832315E-06    .3387996661726958
-4.233704898745038E-06    .3395529221752953
-3.590410713869574E-06    .3404271763094834
-2.971440567082996E-06    .341433874412035
-2.376798128100541E-06    .3425865604473136
-1.806505011774549E-06    .3439007378455418
-1.260601661405314E-06    .345393669010457
-7.391486353900189E-07    .3470840440900813
-2.422864241523421E-07    .3489914219494018
 0       .3500000000003638
```

History of Thrust Magnitude (lbs.) and Direction (Radians) at Each Time Point

Missile Heading -90°
Target Aspect 180°

```
THRUST USED     ANGLE USED
-1.164666815255909E-05     4.723242003744533
 14399.99998027039      6.479262294937597E-02
 14399.99998047564      6.335285438950643E-02
 14399.99998062214      6.437690961887 19E-02
 14399.9999807254       6.797136202113957E-02
 14399.99998080467      7.415169752049955E-02
 14399.99998095616      7.622122700236574E-02
 14399.99998112485      7.700876366212161E-02
 14399.99998132152      7.828509488296193E-02
 14399.99998155864      8.014004523567484E-02
 14399.99998185297      8.273575855923544E-02
 14399.99998222689      8.632825473188248E-02
 14399.99998270991      9.131522395380891E-02
 14399.99998334048      9.832592525874304E-02
 14399.99998416803      .1083958274495448
-1.566493161176776E-05     7.969501831334 19E-02
-1.460274935475232E-05     7.595037419676 36E-02
-1.357041566137291E-05     7.301455145550678E-02
-1.256871182486963E-05     7.067956974863 07E-02
-1.159822829443353E-05     6.878189976333292E-02
-1.065942187043155E-05     6.719148077868842E-02
-9.752657448003086E-06     6.580201445930516E-02
-8.878238563914252E-06     6.452315951794057E-02
-8.036430022016277E-06     6.327434468444718E-02
-7.227474929841107E-06     6.197975874175827E-02
-6.451607773021927E-06     6.056412099550624E-02
-5.709064629795342E-06     5.894892171735704E-02
-5.000091267843396E-06     5.704890571965851E-02
-4.324949577890757E-06     5.476865053780268E-02
-3.683922571868313E-06     5.199917073219558E-02
-3.077043609868248E-06     4.869794646464365E-02
-2.504968690771787E-06     4.444098055129932E-02
-1.968061221118917E-06     3.92564571099527E-02
-1.470174244710638E-06     3.272654142216548E-02
-1.007127876053239E-06     2.621487103644177E-02
-5.79326339606816E-07      1.768514541362738E-02
-1.871986448563937E-07     6.513313859402577E-03
 0      0
```

31

History of Thrust Magnitude (lbs.) and Direction (Raidans) at Each Time Point

Missile Heading -45°
Target Aspect 90°

```
THRUST USED        ANGLE USED
-1.63353040870099E-05     4.686448382899316
 14399.99995182016       .3876516777088849
 14399.99995216494       .3832390471804268
 14399.99995247651       .3801851728985935
 14399.99995275883       .3771757452566404
 14399.99995308508       .3712756293081592
 14399.99995338534       .3581180177472189
 14399.99995369842       .3462880488682431
 14399.99995403657       .3348542026327179
 14399.99995441427       .3239179702043936
 14399.99995484988       .3135202020455658
 14399.99995536674       .3036432105026189
 14399.99995599447       .2942186466090938
 14399.99995677048       .2851357111224346
 14399.99995774172       .2762476770389699
 14399.99995896665       .2673758108020489
 14399.99996051725       .2583097067560104
 14399.99996248091       .2488022056680618
 14399.99996496153       .2384328227279375
-3.360690567644473E-05      .2701896461683151
-3.055014234939948E-05      .2705728921977258
-2.761881665880012E-05      .2712376903123615
-2.481337271253712E-05      .2721725650935123
-2.213474462303725E-05      .2733844764567251
-1.958432548230329E-05      .2748939275923527
-1.716396195193501E-05      .2767326749345042
-1.487597041217111E-05      .2789430921864116
-1.272317324197033E-05      .2815787121560107
-1.070895599476627E-05      .2847057554868285
-8.837348332594896E-06      .2884056528652088
-7.113133978187216E-06      .2927787357020200
-5.541998031254628E-06      .297949450815962
-4.130724330628648E-06      .304073678922899
-2.887462000867227E-06      .3113490206533355
-1.822090320944155E-06      .3200292054864544
-9.46727132607903E-07       .3304437549857585
-2.764530322192844E-07      .3430223498022051
 0        .3500000000003638 -
```

History and Thrust Magnitude (lbs.) and Direction (Radians) at Each Time Point

Missile Heading -45°
Target Aspect 180°

```
THRUST USED       ANGLE USED
 1.367321582005183E-05    4.83717008429501 4
 14400.00005733645    -1.397415958209399E-02
 14400.00005555382    -1.281514725891007E-02
 14400.00005378982    -1.385445635108004E-02
 14400.00005200976    -1.909021660226769E-02
 14400.00005010682    -9.936584067779497E-03
 14400.0000481275      7.221619234648867E-04
 14400.0000460542      1.034171363879729E-02
 14400.00004386944     1.878848443514379E-02
 14400.00004155421     2.593911288554153E-02
 14400.00003908793     3.165518097112531E-02
 14400.0000364487      3.574040945795267E-02
 3.504169887886131E-05    7.42664862136405E-02
 3.215485467067173E-05    8.1668059898296E-02
 2.928264034391217E-05    8.757594125583021E-02
 2.642542579467247E-05    9.238318277518468E-02
 2.358333612686819E-05    9.634692870081228E-02
 2.075631875359627E-05    9.962887758792461E-02
 1.794418815165447E-05    .102316828785256
 1.514665623229595E-05    .1044341137683111
 1.236335325271146E-05    .1059400256996459
 9.593842453132146E-06    .1067222295889411
 6.837630698889432E-06    .106580729974342
 4.094177052457139E-06    .1052018906808591
 1.362901359636352E-06    .1021200558159281
 0       9.999999999990905E-02
```

History of Thrust Magnitude (lbs.) and Direction

Missile Heading -45°
Target Aspect 0°

```
THRUST USED     ANGLE USED
-8.63609128032059 8E-06    5.497045771010888
 14399.99999224112      4.874623557450994E-02
 14399.99999235387      5.320166791474908E-02
 14399.99999246009      6.068224114497995E-02
 14399.99999263668      6.58399346371108E-02
 14399.9999928314       7.008778557880289E-02
 14399.99999305048      7.454326872394856E-02
 14399.99999330091      7.929779042401483E-02
 14399.99999359178      8.452702936649252E-02
 14399.99999393462      9.05150492605648E-02
 14399.99999434359      9.771095047974996E-02
 14399.99999483552      .106847337975765 9
 14399.99999542933      .119195073060362 7
-4.167844072779003E-06     8.506952862118209E-02
-3.46731233393475E-06      7.810431814077696E-02
-2.809366742004528E-06     7.048833264681219E-02
-2.195648663283228E-06     6.185014095925982E-02
-1.628276506234843E-06     5.24025385183512E-02
-1.10631018930917E-06      4.115799875021087E-02
-6.306300426630698E-07     2.712890277855009E-02
-2.031446021637576E-07     9.850414457511216E-03
 0       0
```

Conclusions and Recommendations

From the table, the general pattern is that the conjugate gradient trajectories have significantly shorter times to intercept for all cases with the greatest improvement occurring for the longer duration trajectories and the average improvement being around 25%. The general nature of the conjugate gradient trajectory is to burn at full throttle for as long as possible. It should be noted here that these results represent local minimums of the cost function 4) and not global minimums. There are other local minimums which may be significantly better than the ones obtained. "Tuning" of the computer program and more experimentation with our cost function, to determine its "hills and valleys," as a function of thrust magnitude and direction history will enable us to achieve these.

The purpose of the initial phase of this project has been accomplished in establishing the desirability of considering variable thrust engines in conjunction with engine gimbling to provide trajectories with significantly improved characteristics. Specifically, from these results the time to intercept has been improved, but improvement in other characteristics such as fuel used, can also be obtained. Furthermore, numerical results indicate that an engine capable only of restarting in flight rather than a continuously variable one achieves these improvements. (1)

It is noted here that this work establishes the presence of improved trajectories over the ones presently being used. Such items as mechanization of these trajectories into an actual missile have not been considered.

---

(1) However, this type of control may not provide the global minimum

Suggestions

The following extensions of this work are suggested:

- a) Tuning of the computer program (problem scaling)
- b) Experimentation with additional cases and with the weighting factor UN of the cost to determine the best value for reducing the time to intercept
- c) Modifying the program to consider minimizing the fuel used till intercept or other trajectory parameters of interest
- d) Modifying the computer program to include three dimensional trajectories.

Appendix A

Conjugate Gradient Program
In Finite Dimensional Space

```
2 DOUBLE PRECISION ARG(101),G(101),HH(202),F
2.5 DIMENSION YO(5),AT(50),TT(50)
2.6 EXTERNAL FUNCT
3             COMMON/FMAN/IFL,LF,H,NI,DY1T,DY3T,Y1TO,Y3TO,UN,YO
4 DATA(AT(I),I=1,38)/
5 DATA(TT(I),I=1,38)/
6 DATA N,EST,LIMIT,IFL,H,NI,DY1T,DY3T,Y1TO,Y3TO,UN,TF,LF/

7  DATA(YO(I),I=1,5)/
8     EPS=.0001
9             DO 10 I=1,(N-2),2
10             IN=(I+1)/2
11             ARG(I)=TT(IN)
12             ARG(I+1)=AT(IN)
13    10      CONTINUE
14             ARG(N)=TF
15 CALL DFMCG(FUNCT,N,ARG,F,G,EST,EPS,LIMIT,IER,HH)
16 WRITE (1,111) F
17 111 FORMAT (1X,12HMIN COST IS=,F19.8//1X,11HMIN PTS ARE/)
17.1 DO 1415 IX=1,N
17.11 WRITE (1,1414) ARG(IX)
17.12 1414 FORMAT(1X,F19.8)
17.12 1415 CONTINUE
18    END
26  SUBROUTINE FUNCT(N,ARG,VAL,GRAD)
27 DIMENSION Y(5),YO(5),FY1(5),YJ1(5),FYJ1(5,5),FUJ1(5,2),YU(5,2,50),
YJ1U(5,2,50)
27.5 DOUBLE PRECISION ARG(N)
27.6 DOUBLE PRECISION GRAD(N)
27.7 DOUBLE PRECISION VAL
27.75 DOUBLE PRECISION AMBDA,TPI,TTMIN,[illegible]
27.9 DOUBLE PRECISION HH(202)
27.91 COMMON/PRINT/HH,AMBDA,IERR
30    C: GETTING CLOSED FORM PARTIALS OF STATE INTEGRALS
31 COMMON TAT,CAT,SAT,MN,E1,E2,CS,OSC
32 COMMON/FMAN/ IFL,LF,H,NI,DY1T,DY3T,Y1TO,Y3TO,UN,YO
43 REAL MN
43.01 DANGABS=0.0
43.02 DO 1010 I=2,N-1,2
43.03 DANGMAG=DABS(ARG(I)-HH(N+I))
43.04 IF(DANGMAG.LE.DANGABS) GO TO 1010
43.05 DANGABS=DANGMAG
43.06 1010 CONTINUE
43.07 DTTABS=0.0
43.08 DO 1020 I=1,N-2,2
43.09 DTTMAG=DABS(ARG(I)-HH(N+I))
43.1 IF(DTTMAG.LE.DTTABS)GO TO 1020
43.11 DTTABS=DTTMAG
43.12 1020 CONTINUE
43.13 DISPLAY"DANGABS=",DANGABS,"DTTABS=",DTTABS,"AMBDA=",AMBDA,"IERR=",
IERR,"TF=",ARG(N)
44             INDT=0
45 TPI=6.28318D0
46             MNI=NI+1
47             IND1=0
48             INDG=0
49             CS=1117.77-40.92*H
50 OSC=.1734*.00243* EXP(-.334*H)
51             DT=ARG(N)/NI
52             J=2
53             DO 10 I=1,5
54             Y(I)=YO(I)
55    10      CONTINUE
57    C:
58    C:
59    C: BIG LOOP FOR INTEG. & DIFF. IF INTEG.
```

```
              DO 15 I=1,5
              DO 15 J=1,2
              YU(I,J,1)=0.0
    15        CONTINUE
              IF (IFL.EQ.0) GO TO 500
    C:
    C:
    C: SIMPLE INTEGRATION
    500       DO 1000 J=2,MNI
              DO 520 L=1,(J-1)
              DO 520 I=1,5
              DO520 K=1,2
              YJ1U(I,K,L)=YU(I,K,L)
    520       CONTINUE
              DO 530 L=1,5
              YJ1(L)=Y(L)
    530       CONTINUE
              IAT=2*(J-1)
              ITT=IAT-1
IF(ARG(IAT).GE.0.0D0) GO TO 535
              ATJ1=ARG(IAT)+TPI
              GOTO 538
    535       IF(ARG(IAT).LE.TPI) GO TO 537
              ATJ1=ARG(IAT)-TPI
              GOTO 538
    537       ATJ1=ARG(IAT)
    538       TTJ1=ARG(ITT)
              MTJ1=J-1
CALL FU(YJ1,ATJ1,TTJ1,MTJ1,INB1,[illegible],FK1)
DO 540 I=1,5
Y(I)=YJ1(I)+FK1(I)*DT
540 CONTINUE
Y(1)=Y(1)+LF*FK1(2)*DT*DT/2.0
Y(3)=Y(3)+LF*FK1(4)*DT*DT/2.0
CALL GRADIENT(YJ1,FK1,FYJ1,FUJ1,TTJ1)
              DO 550 I=1,5
              DO 550 K=1,5
FYJ1(I,K)=FYJ1(I,K)*DT
    550       CONTINUE
              DO 553 I=1,5
              DO 553 K=1,2
              FUJ1(I,K)=FUJ1(I,K)*DT
    553        CONTINUE
    C:
    C:
    C: LOOP FOR [illegible] SIMPLE INTEG. PARTIALS
               DO 600 K=1,J
    555        IF(K.NE.J) GOTO 570
               DO 560 I=1,5
               DO560 L=1,2
YU(I,L,J)=0.0
    560        CONTINUE
               GO TO 600
    570        IF(K.NE.(J-1)) GO TO 590
               DO 580 I=1,5
               DO 580 L=1,2
               YU(I,L,(J-1))=FUJ1(I,L)
    580        CONTINUE
DO 585 L=1,2
YU(1,L,(J-1))=YU(1,L,(J-1))+LF*FUJ1(2,L)*DT/2.0
YU(3,L,(J-1))=YU(3,L,(J-1))+LF*FUJ1(4,L)*DT/2.0
585 CONTINUE
               GOTO 600
    590        DO 595 I=1,5
               DO 595 L=1,2
               YU(I,L,K)=YJ1U(I,L,K)
```

```
          DO 595 IJ=1,5
          YU(I,L,K)=YU(I,L,K)+FYJ1(I,IJ)*YJ1U(IJ,L,K)
  595     CONTINUE
DO 597 L=1,2
DO 597 I=1,5
YU(1,L,K)=YU(1,L,K)+LF*FYJ1(2,I)*YJ1U(I,L,K)*DT/2.0
YU(3,L,K)=YU(3,L,K)+LF*FYJ1(4,I)*YJ1U(I,L,K)*DT/2.0
597 CONTINUE
  600     CONTINUE
  1000    CONTINUE
  C: SETTING UP TARGET COORD.
          Y1T=Y1T0+DY1T*ARG(N)
          Y3T=Y3T0+DY3T*ARG(N)
          VAL=ARG(N)+UN*((Y(1)-Y1T)**2+(Y(3)-Y3T)**2)
DISTAN=(Y(1)-Y1T)**2+(Y(3)-Y3T)**2
DISPLAY"DISTAN=",DISTAN
  C:
  C:
  C: COMPUTE PARTIAL OF COST W.R.T. TF
GRAD(N)=1.0+2.0*UN*((Y(1)-Y1T)*(FK1(1)-DY1T)+(Y(3)-Y3T)*(FK1(3)-
DY3T))
  C: FORMING PARTIALS OF COSR W.R.T. U
            CTY1=2.0*UN*(Y(1)-Y1T)
          CTY3=2.0*UN*(Y(3)-Y3T)
          DO 620 K=1,(N-2),2
KK=(K+1)/2
          GRAD(K)=CTY1*YU(1,1,KK)+CTY3*YU(3,1,KK)
           GRAD(K+1)=CTY1*YU(1,2,KK)+CTY3*YU(3,2,KK)
  620     CONTINUE
  C:
  C:
  C: PRINT COST, G VIOLATIONS, BAD TAT VALUES
WRITE (1,777) VAL,INDG,IND1
  777     FORMAT(1X,4HVAL=,F19.8,5X,5HINDG=,I8,5X,5HIND1=,I8)
  C:
  C:
  C: COMPUTE FUEL USED
          FS=0.0
          DO 630 I=1,(N-2),2
          FS=FS+ARG(I)*DT
  630     CONTINUE
WRITE (1,888) FS
  888     FORMAT(1X,10HFUEL USED=,F19.8)
  C:
          C:
  C: COMPUTE THRUST VIOLATIONS AND MAX,MIN VALUES
TTMAX=14400.0D0
TTMIN=0.0D0
DO 680 I=1,(N-2),2
IF(ARG(I).GT.TTMIN) GO TO 660
TTMIN=ARG(I)
INDT=INDT+1
GO TO 680
660 IF(ARG(I).LT.TTMAX) GO TO 680
TTMAX=ARG(I)
INDT=INDT+1
680 CONTINUE
C: PRINT NUMBER OF THRUST VIOLATIONS AND MAX,MIN VALUES
WRITE (1,9999) INDT,TTMAX,TTMIN
9999 FORMAT(1X,28HNUMBER OF THRUST VIOLATIONS=,I8,/,1X,
6HTTMAX=,F19.8,/,1X,6HTTMIN=,F19.8)
1212 CONTINUE
2222 FORMAT (1X,4HVAL=,F19.8,3X//1X,10HYU(I,1,1)=,
3F19.8/1X,2F19.8/1X,10HYU(I,2,1)=,3F19.8/1X,2F19.8/1X,10HYU(I,1,2)=,
3F19.8/1X,2F19.8)
```

```
2323 FORMAT (1X,10HYU(I,1,3)=,3F19.8/1X,2F19.8/1X,10HYU(I,2,3)=,
3F19.8/1X,2F19.8/1X,10HYU(I,2,2)=,3F19.8/1X,2F19.8//1X,11HGRAD VALUES/)
WRITE(1,4444) CTY1,CTY3,(Y(I),I=1,5),Y1T,Y3T
4444 FORMAT(1X,5HCTY1=,F19.8,5X,5HCTY3=,F19.8/1X,5HY(I)=,
3F19.8/1X,2F19.8/1X,4HY1T=,F19.8,5X,4HY3T=,F19.8)
5000 RETURN
END
SUBROUTINE FU(Y,AT,TT,MT,IND1,INDG,DY)
DIMENSION Y(5),B(5),DY(5)
COMMON TAT,CAT,SAT,MN,E1,E2,CS,QSC
DATA (B(I),I=1,5)/0.0,3.14159,-3.14159,0.8318,-0.8318/
REAL MN
PI=3.14159
VMS=Y(2)**2 +Y(4)**2
VM=SQRT(VMS)
QS=QSC*VMS
MN=VM/CS
TAU=ATAN2(Y(4),Y(2))
IF(TAU.GE.0.0) GO TO 2110
TAU=TAU+2.0*PI
2110 CONTINUE
TAT=TAU-AT
AB=ABS(TAT)
IF(AB.GT.PI) GOTO 3
ALP=AB
GOTO 5
3 ALP = 2.0 * PI - AB
5 DO 8 I=1,5
IF(TAT.EQ.B(I)) GOTO 10
8 CONTINUE
GO TO 12
10 IND1=IND1+1
12 IF(TAT.GE.0.0.AND.TAT.LE.PI) GOTO 15
IF(TAT.LE.-PI) GOTO 15
E1=SIN(AT)
E2=-COS(AT)
GOTO 20
15 E1=-SIN(AT)
E2=COS(AT)


C: FORMING CA AND CN FUNCTIONS
20     CCA=0.9283*COS(2.5714*ALP)
CA=-0.5565+CCA*AMIN1(0.85+0.15*MN,1.157-0.17*MN)
CCN=-5.2387E-02*(1.0-23.2199*ALP*(1.0+5.9235*ALP*(1.0-.6658*ALP*(
1.0-.1580*ALP))))
CN=CCN*AMIN1(.30+.55*MN,1.035-.074*MN)
FN=CN*QS
FA=CA*QS
IF(FN/Y(5).LE.1353.0) GOTO 23
INDG=INDG+1
23 CAT=COS(AT)
SAT=SIN(AT)
DY(1)=Y(2)
DY(2)=((TT-FA)*CAT-FN*E1)/Y(5)
DY(3)=Y(4)
DY(4)=((TT-FA)*SAT-FN*E2)/Y(5)
DY(5)=-TT/8050.0
RETURN
END

SUBROUTINE GRADIENT(Y,DY,DFY,DFU,[illegible])
DIMENSION Y(5),DFY(5,5),DFU(5,2),DY(5)
COMMON TAT,CAT,SAT,MN,E1,E2,CS,QSC
DATA PI/3.14159/
REAL MNY2,MNY4,MN
```

```
IF (TAT .GE. 0.0 .AND. TAT .LE. PI) GOTO 35
IF (TAT .LE. -PI .OR. TAT .EQ. 2.0*PI) GOTO 35
E1AT=CAT
E2AT=SAT
ALPTAU=-1.0
GOTO 40
35 ALPTAU=1.0
E1AT=-CAT
E2AT=-SAT
40 ALPAT=-ALPTAU
QSY2=QSC*2.0*Y(2)
QSY4=QSC*2.0*Y(4)
VMS=(Y(2)**2  +Y(4)**2)
VM=SQRT(VMS)
QS=QSC*VMS
CTAU=Y(2)/VM
STAU=Y(4)/VM
ALPY2=-ALPTAU*Y(4)/VMS
ALPY4=ALPTAU*Y(2)/VMS
MNY2=CTAU/CS
MNY4=STAU/CS

C  FORMING DERVATIVES OF DY W.R.T. STATE
AS=ABS(TAT)
IF (AS.GT.PI) GO TO 43
ALP=AS
GO TO 44
43 ALP=2.0*PI-AS
44 CCA=0.9283*COS(2.5714*ALP)
CA=-0.5565+CCA*AMIN1(0.85+0.13*MN,1.197-0.17*MN)
CCN=-(1.0-23.8199*ALP*(1.0+5.9834*ALP*(1.0-.6654*ALP*(1.0-.1580*ALP
))))*5.2387E-2
CN=CCN*AMIN1(.30+0.55*MN,1.095-0.074*MN)
CAALP=-2.387031*SIN(2.5714*ALP)*AMIN1(0.85+0.13*MN,1.197-0.17*MN)
IF (MN.LE.1.155) GO TO 45
CAMN=-.170*CCA
GO TO 47
45 CAMN=.13*CCA
47 CCNALP=-5.2387E-2*(-23.8199-283.0[illegible]*ALP+951.77654*ALP*ALP-
208.932836*ALP*ALP*ALP)
CNALP=CCNALP*AMIN1(.3+.55*MN,1.095-0.074*MN)
IF(MN.LE.1.275) GO TO 48
CNMN=-0.074*CCN
GO TO 49
48 CNMN=.55 *CCN
49 DFY(2,2)=((-(CAALP*CAT+CNALP*E1)*ALPY2-(CAMN*CAT+CNMN*E1)*
MNY2)*QS-(CA*CAT+CN*E1)*QSY2)/Y(5)
DFY(2,4)=((-(CAALP*CAT+CNALP*E1)*ALPY4-(CAMN*CAT+CNMN*E1
)*MNY4)*QS-(CA*CAT+CN*E1)*QSY4)/Y(5)
DFY(2,5)=-DY(2)/Y(5)
DFY(4,2)=((-(CAALP*SAT+CNALP*E2)*ALPY2-(CAMN*SAT+CNMN*E2
)*MNY2)*QS-(CA*SAT+CN*E2)*QSY2)/Y(5)
DFY(4,4)=((-(CAALP*SAT+CNALP*E2)*ALPY4-(CAMN*SAT+CNMN*E2
)*MNY4)*QS-(CA*SAT+CN*E2)*QSY4)/Y(5)
DFY(4,5)=-DY(4)/Y(5)


DFY(1,1)=0.0
DFY(1,2)=1.0
DFY(1,3)=0.0
DFY(1,4)=0.0
DFY(1,5)=0.0
DFY(2,1)=0.0
DFY(2,3)=0.0
DFY(3,1)=0.0
DFY(3,2)=0.0
```

```
353 DFY(3,3)=0.0
354 DFY(3,4)=1.0
355 DFY(3,5)=0.0
356 DFY(4,1)=0.0
357 DFY(4,3)=0.0
358 DFY(5,1)=0.0
359 DFY(5,2)=0.0
360 DFY(5,3)=0.0
361 DFY(5,4)=0.0
362 DFY(5,5)=0.0
363 DFU(2,2)=(-(CAALP*CAT +CNALP*SAT)*ALPAT-CN*CAT+CA*
SAT)*QS/Y(5)
364 DFU(4,2)=(-(CAALP*SAT+CNALP*CAT)*ALPAT-CN*SAT-CA*CAT)*QS/Y(5)
364.5 DFU(2,2)=DFU(2,2)-TT*SAT/Y(5)
364.6 DFU(4,2)=DFU(4,2)+TT*CAT/Y(5)
365 DFU(1,1)=0.0
366 DFU(1,2)=0.0
367 DFU(3,1)=0.0
368 DFU(5,2)=0.0
369 DFU (3,2)=0.0
370 DFU(2,1)=1.0/Y(5)*CAT
371 DFU(4,1)=1.0/Y(5)*SAT
372 DFU(5,1)=-1.0/[illegible]
373 RETURN
374 END
400          SUBROUTINE DFMCG(FUNCT,N,X,F,G,EST,EPS,LIMIT,IER,H)
401       DIMENSION X(N)
401.05 DOUBLE PRECISION HH(202)
401.1 DIMENSION G(N)
401.2 DIMENSION H(N)
402          DOUBLE PRECISION X,G,GNRM,H,HNRM,F,FX,FY,OLDF,OLDG,SNRM,AMBDA,
ALFA,DALFS,T,Z,W,DX,DY
402.1 COMMON/PRINT/HH,AMBDA,IERR
402.5 DISPLAY  "OLDF=",OLDF,"NUMBER=1"
403          CALL FUNCT(N,X,F,G)
403.11 IERR=IER
404          KOUNT=0
405          IER=0
406          N1=N+1
407        1 DO 43 II=1,N1
408          KOUNT=KOUNT+1
409          OLDF=F
410          GNRM=0.D0
411          DO 2 J=1,N
412        2 GNRM=GNRM+G(J)*G(J)
413          IF(GNRM)46,46,3
414        3 IF(II-1)4,4,6
415        4 DO 5 J=1,N
416        5 H(J)=-G(J)
417          GO TO 8
418        6 AMBDA=GNRM/OLDG
419          DO 7 J=1,N
420        7 H(J)=AMBDA*H(J)-G(J)
421        8 DY=0.D0
422          HNRM=0.D0
423          DO 9 J=1,N
424          K=J+N
425          H(K)=X(J)
426          HNRM=HNRM+DABS(H(J))
426.2 9 DY=DY+H(J)*G(J)
427 DO 734 I=1,[illegible]
427.01 HH(I)=H(I)
427.02 734 CONTINUE
428          IF(DY)10,42,42
429       10 SNRM=1.D0/HNRM
430          FY=F
```

```
      ALFA=2.D0*(EST-F)/DY
      AMBDA=1.D0
      IF(ALFA)13,13,11
   11 IF(ALFA-AMBDA)12,13,13
   12 AMBDA=ALFA
   13 ALFA=0.D0
   14 FX=FY
      DX=DY
      DO 15 I=1,N
   15 X(I)=X(I)+AMBDA*H(I)
   DISPLAY "OLDF=",OLDF,"NUMBER=2"
   DISPLAY "THRUST USED",    "ANGLE USED"
   DO 800 III=1,N-2,2
   DISPLAY X(III),X(III+1)
   800 CONTINUE
      CALL FUNCT(N,X,F,G)
   IERR=IER
      FY=F
      DY=0.D0
      DO 16 I=1,N
   16 DY=DY+G(I)*H(I)
      IF(DY)17,38,20
   17 IF(FY-FX)18,20,20
   18 AMBDA=AMBDA+ALFA
      ALFA=AMBDA
      IF(HNRM*AMBDA-1.D10)14,14,19
   19 IER=2
      RETURN
   20 T=0.
   21 IF(AMBDA)22,38,22
   22 Z=3.D0*(FX-FY)/AMBDA+DX+DY
      ALFA=DMAX1(DABS(Z),DABS(DX),DABS(DY))
      DALFA=Z/ALFA
      DALFA=DALFA*DALFA-DX/ALFA*DY/ALFA
      IF(DALFA)23,27,27
   23 DO 24 J=1,N
      K=N+J
   24 X(J)=H(J)
   DISPLAY "OLDF=",OLDF,"NUMBER=3"
      CALL FUNCT(N,X,F,G)
   IERR=IER
   25 IF(IER)47,26,47
   26 IER=-1
      GOTO 1
   27 W=ALFA*DSQRT(DALFA)
      ALFA=(DY+W-Z)*AMBDA/(DY+2.D0*W-DX)
      DO 28 I=1,N
   28 X(I)=X(I)+(T-ALFA)*H(I)
   DISPLAY "OLDF=",OLDF,"NUMBER=4"
      CALL FUNCT(N,X,F,G)
   IERR=IER
      IF(F-FX)29,29,30
   29 IF(F-FY)38,38,30
   30 DALFA=0.D0
      DO 31 I=1,N
   31 DALFA=DALFA+G(I)*H(I)
      IF(DALFA)32,35,35
   32 IF(F-FX)34,33,35
   33 IF(DX-DALFA)34,38,34
   34 FX=F
      DX=DALFA
      T=ALFA
      AMBDA=ALFA
      GO TO 21
   35 IF(FY-F)37,36,37
   36 IF(DY-DALFA)37,38,37
```

```
487      37 FY=F
488         DY=DALFA
489         AMBDA=AMBDA-ALFA
490         GO TO 20
491      38 T=0.D0
492         DO 39 J=1,N
493         K=J+N
494         H(K)=X(J)-H(K)
495      39 T=T+DABS(H(K))
495.01 DO 735 I=1,2*N
495.02 HF(I)=H(I)
495.03 735 CONTINUE
496         IF(KOUNT-N1)41,40,40
497      40 IF(T-EPS)45,45,41
498      41 IF(OLDF-F+EPS)19,25,42
499      42 OLDG=GNRM
500         IF(KOUNT-LIMIT)43,44,44
501  43 IER=0
502         GO TO 1
503      44 IER=1
504         IF(GNRM-EPS)46,46,47
505      45 IF(GNRM-EPS)46,46,25
506      46 IER=0
507      47 RETURN
508         END
```

Appendix B

Conjugate Gradient Program
In Infinite Dimensional Space

```
   1 C: THIS PROGRAM DOES CONJ. GRAD. FOR MISSLE TRAJ.
2
3
4
5 DIMENSION TT(81),AT(81),E1(81),E2(81),B(5),Y(5),LAM(5),TAT(81),
Y2(81),Y4(81),CA(81),CN(81),CAT(81),SAT(81),Y5(81),F2(81),F4(81),
DLAM(5),DY(5),HAT(81),HT(81),S1(81),S2(81),TT1(81),AT1(81),Y12(81)
6 DIMENSION Y14(81),Y1(5),YO(5),TAT1(81),E11(81),E12(81),CN1(81),
CA1(81),CAT1(81),SAT1(81),Y15(81),DY1(5),F12(81),F14(81),ATL(81),
TTL(81),EL1(81),EL2(81),TATL(81),YL2(81),YL4(81),CAL(81)
7  DIMENSION CNL(81),CATL(81),SATL(81),YL5(81),FL2(81),FL4(81)
8 DATA ITMAX,ITGMAX,STEPM/1
9 DATA NC0,NC1,NC2,NC3,NC4,NC5,NC6,NC7,NC8,NC9,NC10,NC11/.23069881,
3.2485197,29.609739,-20.952979,4.1362894,-.12274384,.50764409,
-.12286171,1.3576835,-1.1542471,.3547580[illegible],-.03[illegible]493/
10 DATA (TT(I),I=1,81)/


11 DATA (AT(I),I=1,14)/


11.5 DATA (AT(I),I=15,25)/


12 DATA (YO(I), I=1,5)/
13 DATA H,TF,[illegible]/

14 DATA (B(I),I=1,5)/0.0,3.14159,-3.14159,6.28318,-6.28318/
15 DATA CC0,CC1,CC2,CC3,CC4,CC5,CC6,CC7,CC8,CC9,CC10,CC11/.30574065,
2.537371,-11.984872,11.098411,-3.755[illegible]63,.4885[illegible],.758700[illegible],
.35336886,-.25822295,.071176152,-.01490616,.0017203865/
16 PI=3.14159
17 2 ITG=1
18 IND1=0
19 INDG=0
20 ITL=0
21
22
23 C: COMPUTE INITIAL GRADIENT TRAJECTORY
24 DO 4 I=1,5
25 Y(I)=YO(I)
26 4 CONTINUE
27 CS=1117.77-40.92*H
28 QSC=.1734*.00243*EXP(-.334/H)
29 DT=TF/80.0
30 DO 30 MT=1,81
31 Y2(MT)=Y(2)
32 Y4(MT)=Y(4)
32.5 VMS=Y(2)**2.0+Y(4)**2.0
33 VM=SQRT(VMS)
34 QS=QSC*VMS
35 MN=VM/CS
36 TAU=ATAN2(Y(4),Y(2))
37 TAT(MT)=TAU-AT(MT)
38 AB=ABS(TAT(MT))
39 IF(AB.GT.PI) GOTO 3
40 ALP=AB
41 GOTO 5
42 3 ALP = 2.0 * PI - AB
43 5 DO 8 I=1,5
44 IF(TAT(MT).EQ.B(I)) GOTO 10
```

```
8 CONTINUE
10 MT1=MT1+1
~~[illegible]~~
IF(TT(MT).LE.-PI) GOTO 15
E1(MT)=SIN(AT(MT))
E2(MT)=-COS(AT(MT))
GOTO 20
15 E1(MT)=-SIN(AT(MT))
E2(MT)=COS(AT(MT))


C: FORMING CA AND CN FUNCTIONS
20 ALP2=ALP*ALP
ALP3=ALP2*ALP
ALP4=ALP3*ALP
ALP5=ALP4*ALP
MN2=MN*MN
MN3=MN2*MN
MN4=MN3*MN
MN5=MN4*MN
C1=CC0+CC1*ALP+CC2*ALP2+CC3*ALP3+CC4*ALP4+CC5*ALP5
C2=CC6+CC7*MN+CC8*MN2+CC9*MN3+CC10*MN4+CC11*MN5
CA(MT)=C1*C2
N1=NC0+NC1*ALP+NC2*ALP2+NC3*ALP3+NC4*ALP4+NC5*ALP5
N2=NC6+NC7*MN+NC8*MN2+NC9*MN3+NC10*MN4+NC11*MN5
CN(MT)=N1*N2
FN=CN(MT)*QS
FA=CA(MT)*QS
IF(FA/Y(5).LE.1353.0) GOTO 23
INDG=INDG+1
23 CAT(MT)=COS(AT(MT))
SAT(MT)=SIN(AT(MT))
Y5(MT)=Y(5)
DY(1)=Y(2)
DY(2)=((TT(MT)-FA)*CAT(MT)-FN*E1(MT))/Y(5)
F2(MT)=DY(2)
DY(3)=Y(4)
DY(4)=((TT(MT)-FA)*SAT(MT)-FN*E2(MT))/Y(5)
F4(MT)=DY(4)
DY(5)=-TT(MT)/8050.0
IF (MT.EQ.81) GOTO 30
C: SIMPLE INTEGRATION
Y(1)=Y(1)+(DY(1)+DY(2)*DT/2.0)*DT
Y(2)=Y(2)+DY(2)*DT
Y(3)=Y(3)+(DY(3)+DY(4)*DT/2.0)*DT
Y(4)=Y(4)+DY(4)*DT
Y(5)=Y(5)+DY(5)*DT
30 CONTINUE


C: SETTING TARGET COORD.
Y1T=Y1T0+DY1T*TF
Y3T=Y3T0+DY3T*TF


C: COST EXPRESSION
CT=TF+UN*((Y(1)-Y1T)**2.0+(Y(3)-Y3T)**2.0)


C: SET Y AND DY VALUES FOR CTTF COMPUTATION
Y1F=Y(1)
Y3F=Y(3)
DY1F=DY(1)
DY3F=DY(3)
C: WRITE VIOLATIONS [illegible]
32 WRITE (1,501) ITG,I001,INDG,CTL,CT
```

```
501 FORMAT(4HITG=,I8,5HIND1=,I8,5HINDG=,I8,//,4HITL=,I8,6HCOST=,F19


C:FORMING LAMBDA,HT,HAT,CTTF
C: DERIVATIVE OF COST W.R.T. FINAL TIME
CTTF= 1.0 + 2.0 *UN*((Y1F-Y1T) * (DY1F-DY1T) + (Y3F-Y3T)*(DY3F-
DY3T))


C: SETTING FINAL VALUES OF LAMBDA
LAM(1)= 2.0 *UN* (Y1F-Y1T)
LAM(2)=0.0
LAM(3)=2.0*UN*(Y3F-Y3T)
LAM(4)=0.0
LAM(5)=0.0


C: LOOP FOR GETTING GRADIENT AND LAMBDA
DT=TF/80.0
DO 60 MT=81,1
C: EQS. FOR GETTING DERIVATIVES W.R.T. STATE
IF (TAT(MT) .GE. 0.0 .AND. TAT(MT) .LE. PI) GOTO 35
IF (TAT(MT) .LE. -PI .OR. TAT(MT) .EQ. 2.0*PI) GOTO 35
E1AT=CAT(MT)
E2AT=SAT(MT)
ALPTAU=-1.0
GOTO 40
35 ALPTAU=1.0
E1AT=-CAT(MT)
E2AT=-SAT(MT)
40 ALPAT=-ALPTAU
QSY2=QSC*2.0*Y2(MT)
QSY4=QSC*2.0*Y4(MT)
VMS=(Y2(MT)**2.0+Y4(MT)**2.0)
VM=SQRT(VMS)
QS=QSC*VMS
CTAU=Y2(MT)/VM
STAU=Y4(MT)/VM
ALPY2=-ALPTAU*CTAU*CTAU*Y4(MT) / (Y2(MT)*Y2(MT))
ALPY4=ALPTAU*CTAU**2/Y2(MT)
MNY2=CTAU/CS
MNY4= STAU/CS


C: FORMING DERIVATIVES OF DY W.R.T. STATE
AB=(ABS(TAT(MT)))
IF (AB.GT.PI) GO TO 43
ALP=AB
GO TO 44
43 ALP=2.0*PI-AB
44 ALP2=ALP*ALP
ALP3=ALP2*ALP
ALP4=ALP3*ALP
ALP5=ALP4*ALP
MN2=MN*MN
MN3=MN2*MN
MN4=MN3*MN
MN5=MN4*MN
C1ALP=CC1+2.0*CC2*ALP+3.0*CC3*ALP2+4.0*CC4*ALP3+5.0*CC5*ALP4
C2MN=CC7+2.0*CC8*MN+3.0*CC9*MN2+4.0*CC10*MN3+5.0*CC11*MN4
N1ALP=NC1+2.0*NC2*ALP+3.0*NC3*ALP2+4.0*NC4*ALP3+5.0*NC5*ALP4
N2MN=NC7+2.0*NC8*MN+3.0*NC9*MN2+4.0*NC10*MN3+5.0*NC11*MN4
C1=CC0+CC1*ALP+CC2*ALP2+CC3*ALP3+CC4*ALP4+CC5*ALP5
C2=CC6+CC7*MN+CC8*MN2+CC9*MN3+CC10*MN4+CC11*MN5
N1=NC0+NC1*ALP+NC2*ALP2+NC3*ALP3+NC4*ALP4+NC5*ALP5
N2=NC6+NC7*MN+NC8*MN2+NC9*MN3+NC10*MN4+NC11*MN5
```

```
CAALP=C2*C1ALP
CAMN=C1*C2MN
CNALP=N2*N1ALP
CNMN=N1*N2MN
F2Y2=((-(CAALP*CAT(MT)+CNALP*E1(MT))*ALPY2-(CAMN*CAT(MT)+CNMN*EI(MT)
)*MNY2)*QS-(CA(MT)*CAT(MT)+CN(MT)*E1(MT))*QSY2)/Y5(MT)
F2Y4=((-(CAALP*CAT(MT)+CNALP*E1(MT))*ALPY4-(CAMN*CAT(MT)+CNMN*E1(MT)
)*MNY4)*QS-(CA(MT)*CAT(MT)+CN(MT)*E1(MT))*QSY4)/Y5(MT)
F2Y5=-F2(MT)/Y5(MT)
F4Y2=((-(CAALP*SAT(MT)+CNALP*E2(MT))*ALPY2-(CAMN*SAT(MT)+CNMN*E2(MT)
)*MNY2)*QS-(CA(MT)*SAT(MT)+CN(MT)*E2(MT))*QSY2)/Y5(MT)
F4Y4=((-(CAALP*SAT(MT)+CNALP*E2(MT))*ALPY4-(CAMN*SAT(MT)+CNMN*E2(MT)
)*MNY4)*QS-(CA(MT)*SAT(MT)+CN(MT)*E2(MT))*QSY4)/Y5(MT)
F4Y5=-F4(MT)/Y5(MT)


C: FORMING D.E. FOR LAMBDA
DLAM(1)=0.0
DLAM(2)=LAM(1)+LAM(2)*F2Y2+LAM(4)*F4Y2
DLAM(3)=0.0
DLAM(4)=LAM(2)*F2Y4+LAM(3)+LAM(4)*F4Y4
DLAM(5)= LAM(2) * F2Y5 + LAM(4) * F4Y5


C: DERIVATIVES OF DY W.R.T. THETA
F2T=1.0/Y5(MT)
F4T=1.0/Y5(MT)
F5T=-1.0/8050.0
F2AT=(-(CAALP*CAT(MT)+CNALP*E1(MT))*ALPT-CN(MT)*E1AT+CA(MT)*SAT(MT)
)*QS/Y5(MT)
F4AT=(-(CAALP*SAT(MT)+CNALP*E2(MT))*ALPT-CN(MT)*E2AT-CA(MT)*CAT(MT)
)*QS/Y5(MT)


C: GETTING INSTANTANEOUS GRADIENT
HT(MT)= LAM(2) * F2T +LAM(4) * F4T + LAM(5) * F5T
HAT(MT)= LAM(2) *F2AT + LAM(4) * F4AT
IF (MT.EQ.1) GO TO 60


C: SIMPLE INTEGRATION
LAM(2) = LAM(2) + DLAM(2) * DT
LAM(4) = LAM(4) + DLAM(4) * DT
LAM(5) = LAM(5) + DLAM(5) * DT
60 CONTINUE


C: SETTING UP INITIAL ITERATION ALONG SEARCH DIRECTION
IF (ITG.NE.1) GO TO 70
S3=0.0
BETA=0.0
BD=0.0
DO 65 MT=1,81
S1(MT)=0.0
S2(MT)=0.0
65 CONTINUE


C: SIMPLE INTEGRATION FOR BN
70 BN=0.0
DO 72 J= 1,80
BN= BN + (HAT(J )*HAT(J )+HT(J )*HT(J ))*DT
72 CONTINUE
BN= BN + CTTF * CTTF
IF(ITG.EQ.1) GO TO 74
BETA=BN/BD
```

```
GOTO 76
74 DO 75 J=1,81
TTL(J)=TT(J)
ATL(J)=AT(J)
FL1(J)=F1(J)
FL2(J)=F2(J)
TATL(J)=TAT(J)
YL2(J)=Y2(J)
YL4(J)=Y4(J)
CAL(J)=CA(J)
CNL(J)=CN(J)
CATL(J)=CAT(J)
SATL(J)=SAT(J)
YL5(J)=Y5(J)
FL2(J)=F2(J)
FL4(J)=F4(J)
75 CONTINUE
DYL1F=DY1F
DYL3F=DY3F
IND1L=IND1
INDGL=INDG
TFL=TF
CTL=CT
YL1F=Y1F
YL3F=Y3F
76 IT=1
77 IF(IT.NE.1) GOTO 80
STEP=STEPO
S3=-CTTF + BETA *S3
ITL=0
STEPL=0.0
DO 79 J=1,81
S1(J)=-HT(J)+BETA*S1(J)
S2(J)=-HAT(J)+BETA*S2(J)
79 CONTINUE
GOTO 85
80 IF (IT.NE.ITMAX) GO TO 85
WRITE(1,600) IT,ITMAX,ITG,STEPL,CTL
600 FORMAT (4HIT= ,I8,6HITMAX=,I8,4HITG=,I8,6HSTEPL=,F19.8,4HCTL=
,F19.8)
GOTO 147
85 INDT=0
FS=0.0
TF1=TFL+STEP*S3
DO 95 J=1,81
TT1(J)=TTL(J)+STEP*S1(J)
AT1(J)=ATL(J)+STEP*S2(J)
IF(AT1(J).GE.0.0)GOTO 86
AT1(J)=AT1(J)+2.0*PI
GOTO 87
86 IF(AT1(J).LE.2.0*PI)GO TO 87
AT1(J)=AT1(J)-2.0*PI
87 IF(TT1(J).LE.14400.0)GOTO 89
TT1(J)=14400.0
INDT=INDT+1
GOTO 91
89 IF(TT1(J).GE.0.0)GOTO 91
TT1(J)=0.0
INDT=INDT+1
91 FS=FS+TT1(J)*DT
95 CONTINUE
IF (FS.LE.38500.0)GO TO 98
DO 97 J1=1,81
TT1(J1)=38500.0/FS*TT1(J1)
97 CONTINUE
WRITE(1,700) ITG,IT
```

```
700 FORMAT(6HTOO MU,6HCH FUE,6HL USED,4HITG=,I8,3HIT=,I8)


C: INTEGRATE STEPPED TRAJECTORY
98 DO 105 J=1,5
Y1(J)=Y0(J)
105 CONTINUE
CS=1117.77-40.92*H
QSC=0.1734*.00243*EXP(-.334/H)
DT1=TF1/80.0
INDG=0
IND1=0
DO 115 J=1,81
Y12(J)=Y1(2)
Y14(J)=Y1(4)
VMS=Y1(2)**2.0+Y1(4)**2.0
VM=SQRT(VMS)
QS=QSC*VMS
MN=VM/CS
TAU=ATAN2(Y1(4),Y1(2))
TAT1(J)=TAU-AT1(J)
AB=ABS(TAT1(J))
IF(AB.GT.PI)GOTO 117
ALP=AB
GOTO 120
117 ALP=2.0*PI-AB
120 DO 125 I=1,5
IF(TAT1(J).EQ.B(I))GOTO 130
125 CONTINUE
130 IND1=IND1+1
IF(TAT1(J).GE.0.0.AND.TAT1(J).LE.PI) GOTO 135
IF(TAT1(J).LE.-PI) GOTO 135
E11(J)=SIN(AT1(J))
E12(J)=-COS(AT1(J))
GOTO 140
135 E11(J)=-SIN(AT1(J))
E12(J)=COS(AT1(J))
C: FORMING CA1 AND CN1 FUNCTIONS
140 ALP2=ALP*ALP
ALP3=ALP2*ALP
ALP4=ALP3*ALP
ALP5=ALP4*ALP
MN2=MN*MN
MN3=MN2*MN
MN4=MN3*MN
MN5=MN4*MN
C11=CC0+CC1*ALP+CC2*ALP2+CC3*ALP3+CC4*ALP4+CC5*ALP5
C12=CC6+CC7*MN+CC8*MN2+CC9*MN3+CC10*MN4+CC11*MN5
CA1(J)=C11*C12
N11=NC0+NC1*ALP+NC2*ALP2+NC3*ALP3+NC4*ALP4+NC5*ALP5
N12=NC6+NC7*MN+NC8*MN2+NC9*MN3+MN4*NC10+NC11*MN5
CN1(J)=N11*N12
FN=CN1(J)*QS
FA=CA1(J)*QS
IF(FN/Y1(5) .LE. 1353.0) GOTO 113
INDG=INDG +1
113CAT1(J)=COS(AT1(J))
SAT1(J)=SIN(AT1(J))
Y15(J)=Y1(5)
DY1(1)=Y1(2)
DY1(2)=((TT1(J)-FA)*CAT1(J)-FN*E11(J))/Y1(5)
F12(J)=DY1(2)
DY1(3)=Y1(4)
DY1(4)=((TT1(J) - FA) * SAT1(J) - FN * E12(J)) / Y1(5)
F14(J)=DY1(4)
```

```
DY1(5)=-TT1(J) / 8050.0
IF(J .EQ. 81) GOTO 115


C: SIMPLE INTEGRATION
Y1(1)=Y1(1)+(DY1(1)+DY1(2)*DT1/2.0)*DT1
Y1(2)=Y1(2)+DY1(2)*DT1
Y1(3)=Y1(3)+(DY1(3)+DY1(4)*DT1/2.0)*DT1
Y1(4)=Y1(4)+DY1(4)*DT1
Y1(5)= Y1(5)+DY1(5)*DT1
115 CONTINUE

C: SETTING TARGET COORDINATES
Y1T1=Y1T0+DY1T*TF1
Y3T1=Y3T0+DY3T*TF1
C:
C:
C: COMPUTE [illegible] COST
CT1= TF1 + [illegible] * ((Y1(1) - Y1T1)**2.0 + (Y1(3)-Y3T1)**2.0)
WRITE(1,800) ITG,IT,INDT, IND1,FS,INDG,CT1
800 FORMAT (4HITG=,I8,3HIT=,I8,5HINDT=,I8,5HIND1=,I8,//,3HFS=,
F19.8,5HINDG=,I8,4HCT1=,F19.8)
IF(CT1 .GE. CTL) GOTO 130
DO 125 J1=1,81
TTL(J1)=TT1(J1)
ATL(J1)=AT1(J1)
EL1(J1)=E11(J1)
FL2(J1)=E12(J1)
TATL(J1)=TAT1(J1)
YL2(J1)=Y12(J1)
YL4(J1)=Y14(J1)
CAL(J1)=CA1(J1)
CNL(J1)=CN1(J1)
CATL(J1)=CAT1(J1)
SATL(J1)=SAT1(J1)
YL5(J1)=Y15(J1)
FL2(J1)=F12(J1)
FL4(J1)=F14(J1)
125 CONTINUE
IND1L=IND1
INDGL=INDG
TFL=TF1
CTL=CT1
STEPL=STEP
YLIF=Y1(1)
YL3F=Y1(3)
DYL1F=DY1(1)
DYL3F=DY1(3)
ITL=IT
GOTO 140
130 STEP=STEP/2.0
IF(STEP .LT. STEPM) GOTO 145
140 IT=IT+1
GOTO 77
145 WRITE(1,900)STEP,STEPM,ITL,CTL,ITG,STEPL
900 FORMAT(5HSTEP=,F19.8,6HSTEPM=,F19.8,4HITL=,I8,//,4HCTL=,F19.8,
4HITG=,I8,5HSTEPL=,F19.8)
147 IF(CTL.EQ.CT) GOTO 2000
DO 150 J1=1,81
TT(J1)=TTL(J1)
AT(J1)=ATL(J1)
Y2(J1)=YL2(J1)
Y4(J1)=YL4(J1)
E1(J1)=EL1(J1)
E2(J1)=EL2(J1)
TAT(J1)=TATL(J1)
```

```
CA(J1)=CAL(J1)
CN(J1)=CNL(J1)
CAT(J1)=CATL(J1)
SAT(J1)=SATL(J1)
Y5(J1)=YL5(J1)
F2(J1)=FL2(J1)
F4(J1)=FL4(J1)
150 CONTINUE
TF=TFL
BD=BN
CT=CTL
IND1=IND1L
INDG=INDGL
Y1F=YL1F
Y3F=YL3F
DY1F=DYL1F
DY3F=DYL3F
IF(ITG.GT.ITMAX)GOTO 2100
ITG=ITG+1
Y1T=Y1T0+DY1T*TFL
Y3T=Y3T0+DY3T*TFL
GOTO 32
2000 WRITE(1,902)ITG
902 FORMAT(6HNO IMP,6HROVEME,6HNT POS,6HSIBLE ,6HFROM G,
6HRADIEN,6HT IN T,6HHIS DI,6HRECTIO,6HN,     ,6HITG=  ,I8)
C: (NO IMPROVEMENT POSSIBLE FROM GRADIENT IN THIS DIRECTION)
GOTO 5000
2100 WRITE(1,903)
903 FORMAT(6HITG=IT,6HGMAX  )
5000 STOP
END
```

References

1. M. R. Hestenes and E. Stiefel, "Methods of Conjugate Gradients for Solving Linear Systems" J. Res. N.B.S. vol. 49, p. 409, 1952.

2. R. Fletcher and C. M. Reeves, "Function Minimization by Conjugate Gradients," Computer Journal, vol. 7, no. 2, 1964, pp. 149-154.

3. Lasdon Mitter, and Warren, "The Conjugate Gradient Method for Optimal Control Problems," IEEE Transactions on Automatic Control," 1967.

DISTRIBUTION LIST

	No. Copies
I. B. Russak, Code 53Ru Department of Mathematics Naval Postgraduate School Monterey, California 93940	5
Professor Frank Faulkner, Code 53Fa Department of Mathematics Naval Postgraduate School Monterey, California 93940	1
Professor W. M. Woods, Code 53Wo Chairman, Department of Mathematics Naval Postgraduate School Monterey, California 93940	1
E. Q. Smith, Code 5100 Naval Missile Center Point Mugu, California 93042	5
Defense Documentation Center (DDC) Cameron Station Alexandria, Virginia 22314	20
Library Naval Postgraduate School Monterey, California 93940	2
Dean of Research Naval Postgraduate School Monterey, California 93940	1

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS
BEFORE COMPLETING FORM

1. REPORT NUMBER
NPS53Ru73091A

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (*and Subtitle*)
Preliminary Results Concerning the Improvements Realizable Through the use of Variable Thrust Together with Engine Gimbaling for a Particular Interceptor Missile

5. TYPE OF REPORT & PERIOD COVERED
Technical Report, 1973

6. PERFORMING ORG. REPORT NUMBER
~~NPS-53Ru53091A~~

7. AUTHOR(s)
I. Bert Russak

8. CONTRACT OR GRANT NUMBER(s)
NPS-53Ru73091A
PO 3-0014

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Naval Postgraduate School
Monterey, California 93940

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
Element - 62756N
Project - ZF61-512-001

11. CONTROLLING OFFICE NAME AND ADDRESS
Director of Laboratory Programs
Department of the Navy
Washington, D. C. 20360

12. REPORT DATE
September 1973

13. NUMBER OF PAGES
48

14. MONITORING AGENCY NAME & ADDRESS (*if different from Controlling Office*)

15. SECURITY CLASS. (*of this report*)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (*of this Report*)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (*of the abstract entered in Block 20, if different from Report*)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (*Continue on reverse side if necessary and identify by block number*)
Trajectory Optimization
Conjugate Gradient
Calculus of Variations
Variable Thrust

20. ABSTRACT (*Continue on reverse side if necessary and identify by block number*)

Some interceptor missiles as presently formulated possess a programmed thrust magnitude history with a gimbaled engine to provide steering. We examine one such missile to determine whether performance can be improved if we allow a variable thrust magnitude together with engine gimbaling to provide control.

Two trajectory optimization programs were written to provide an initial answer to this problem. Preliminary results indicate reductions in the time to intercept by as much as thirty per-cent over that obtained by the presently

DD Form 1473 (BACK)
1 Jan 73
S/N 0102-014-6601