Analysis of Genetic Algorithm on Bearings-Only Target Motion Analysis

Erdem Köse
Department of Electronics Engineering, Gebze Technical University
Gebze-Kocaeli, TR-41400, Turkey
Email: ekose@gtu.edu.tr

Abstract—Target motion analysis using only bearing angles is an important study for tracking targets in water. Several methods including Kalman-like filters and evolutionary strategies are used to get a good predictor. Kalman-like filters couldn’t get the expected results thus evolutionary strategies have been using in this area for a long time. Target Motion Analysis with Genetic Algorithm is the most successful method for Bearings-Only Target Motion Analysis and we investigated it. We found that Covariance Matrix Adaptation Evolutionary Strategies does the similar work with Target Motion Analysis with Genetic Algorithm and tried it; but it has statistical feedback mechanism and converges faster than other methods. In this study, we compared and criticize the methods.

I. INTRODUCTION

Bearings-Only Target Motion Analysis (BO-TMA) [1] is an analysis method for submarines using sonar bearings received from the hydrophone array. In this study we have only bearings information to track the target; this makes finding an optimal method for the study very hard.

Kalman filters [2] are very popular in Target Motion Analysis (TMA), but due to initial value and linearization problems, evolutionary strategies [3] are more feasible than Kalman like optimum filters. Genetic Algorithms [3], [4], Big Bang-Big Crunch Algorithm (BB-BC) [5]–[7] and Covariance Matrix Adaptation Evolutionary Strategies (CMA-ES) [8]–[10] are the evolutionary strategies which have been used for BO-TMA.

Cost function is simply the Euclidean distance between observed bearing angle and calculated bearing angle [4]. This cost function has complex calculations when getting calculated bearing angle. Because of that a new cost function [11] developed by formulating problem as to fit target courses such that the bearing lines divide the course candidates to equal distances and assuming bearings were measured at same constant time intervals. Equidistant line distance difference cost function is faster but not immune to noise.

In Section II we’ll introduce the BO-TMA and it’s important functions, in Section III we’ll introduce Target Motion Analysis with Genetic Algorithm (TMAGA) and investigate it, in Section IV we’ll introduce CMA-ES and it’s algorithm, in Section V using TMAGA trial dataset we’ll show the performance of CMA-ES, then in Section VI we’ll summarize the paper.

II. BEARINGS-ONLY TARGET MOTION ANALYSIS (BO-TMA)

In this section, we will introduce BO-TMA and essential equations to use evolutionary strategies for tracking target. As we mentioned before, BO-TMA is an analysis method for submarines using sonar bearings received from the hydrophone array. In Fig. 1 we see actual target with dashes, observer with straight bold line. Observer must make a leg to get only one possible linear track candidate for target’s motion, otherwise there will be infinite possible linear track candidates.

A. Position Equations

If we know our initial position $X_{obs}(0)$ and $Y_{obs}(0)$, target’s initial bearing $B_{obsd}(0)$, initial range from observer $R_{tar}(0)$; then we can extract target’s initial position $X_{tar}(0)$ and $Y_{tar}(0)$ from Eq. (1).

\[
\begin{pmatrix}
X_{tar}(0) \\
Y_{tar}(0)
\end{pmatrix}
= \begin{pmatrix}
X_{obs}(0) \\
Y_{obs}(0)
\end{pmatrix} + \begin{pmatrix}
sin(B_{obsd}(0)) \\
cos(B_{obsd}(0))
\end{pmatrix} * R_{tar}(0) \quad (1)
\]

If we know our target’s course angle $C_{tar}$ and velocity $S_{tar}$; then we can extract target’s following positions $X_{tar}(t + \Delta t)$ and $Y_{tar}(t + \Delta t)$, which is changing in $\Delta t$ equal time, from Eq. (2).
\[
\begin{align*}
(X_{\text{tar}}(t + \Delta t) & = X_{\text{tar}}(t) + S_{\text{tar}} \times \sin(C_{\text{tar}}) \times \Delta t \\
Y_{\text{tar}}(t + \Delta t) & = Y_{\text{tar}}(t) \times \cos(C_{\text{tar}}) \times \Delta t
\end{align*}
\] (2)

B. Cost Function

When we get the positions of target \(X_{\text{target,EA}}\) and \(Y_{\text{target,EA}}\) using evolutionary strategies, we can get estimated bearing angles using Eq. (3).

\[
B_{\text{estd}} = \arctan \left( \frac{X_{\text{target,EA}} - X_{\text{obsd}}}{Y_{\text{target,EA}} - Y_{\text{obsd}}} \right)
\] (3)

Cost = \[\sum_{i=0}^{N-1} (B_{\text{estd}}(i) - B_{\text{obsd}}(i))^2\] (4)

The cost function for target motion analysis depends on Euclidean distance between estimated bearing angles and observed ones. This cost function has more calculation complexity but more feasible than the equidistant line distance difference cost function [11].

III. TARGET MOTION ANALYSIS WITH GENETIC ALGORITHM (TMAGA)

We may know all parameters except target’s initial range from observer \(R_{\text{tar}}(0)\), course of target \(C_{\text{tar}}\) and velocity of target \(S_{\text{tar}}\). Evolutionary strategies and genetic algorithms have been helping us for a long time to find these parameters in a short time and using a cost function. Exhaustive search can take so much time for large search areas; but evolutionary strategies gets help from statistics and cost functions to find true path to get true solution.

TMAGA [4] uses genetic algorithm to track target and has holding the best results in TMA studies. TMAGA has three principal structure: space narrowing and tuning chromosome lengths, N generation genetic algorithm and M times Monte Carlo runs. In each Monte Carlo run, 200 generation genetic algorithm runs for narrowing space then chromosomes are tuned using space limits; 500 generation genetic algorithm runs again to find best candidate. After 20 Monte Carlo runs system has 20 offspring to solve the problem. Then we take fitness weighted averages of these 20 offspring to get one solution.

In sections above, we will explain TMAGA in details and discuss the possible problems.

A. Target Motion Representation – Chromosome Structure

A solution individual is a chromosome bit array represents target motion. Solution individual’s structure is given in Eq. (5). \(X_0\) and \(Y_0\) are initial positions of target on \(X\) and \(Y\) axes respectively. \(C\) is the course angle and \(S\) is the velocity of target.

\[
\ldots X_0 \ldots Y_0 \ldots C \ldots S \ldots 2
\] (5)

If we set a parameter bit-length for one of the four parameters above, we must know maximum \(\text{Max}\) and minimum \(\text{Min}\) value that parameter can get at limits. Then we use Eq. (6) with the degree of precision \(p\), and the number of bits \(m\). IEEE Standard for Floating-Point Arithmetic [12] says that for single precision floating point parameter must be \(\log_{10}(2^p) \geq 7.225\). If we set precision \(7\) and \(\text{Max} = 2000\), \(\text{Min} = 0\) meters for \(X_0\) and \(Y_0\) parameters of chromosome, due to Eq. (6) our bit width will be \(38\). For \(C\) and \(S\) bit-widths are \(32\) and \(28\) due to limits \([360\ 0]\) degrees and \([25\ 0]\) meters/seconds respectively. Total \(136\) bits were used; but if we decide to use single precision floating point, we would use only \(128\) bits for representation.

\[
2^{m-1} < (\text{Max} - \text{Min})10^p < 2^m - 1
\] (6)

Instead of representing position by known bearing \(B_{\text{obsd}}\) and range \(R\); the authors chose to use two parameters \(X_0\) and \(Y_0\). These values are correlated by the corresponding bearing. Then using two parameter; creates correlation problems, increases search dimension so enlarges search space causing more time to obtain correct solution. So ideal chromosome must look like Eq. (7), this form uses \(96\) bits single precision floating point numbers.

\[
(R_0|C|S)_{32\text{bits} \ FP \ Real \ Value}
\] (7)

In summary, genetic algorithm is a memory exhaustive and slow method to use in 2009, real valued evolutionary strategies uses real values instead of bit strings, which is more suitable for high level computing, and uses less memory.

B. Narrowing the Parameter Space

Due to TMAGA, search space between estimation and limit values could be reduced \(20\%\) using 200 generations with 50 population size and 20 runs. If we think that \(\sigma = 0.3\) in case of degrees, which is a quite high noise, generation and population size are quite low to converge to the ideal solution. Even if TMAGA says the parameter space’s being narrowed into an infeasible area is a very unlikely case, it seems like very likely case, because genetic algorithm does not care the numerical value of the individual chromosome and does not use a feedback mechanism to narrow the area except the cost function. Because of this genetic algorithm results are expected to converge a different point in every different 20 runs.

If route is perpendicular to observer, noise becomes more destructive because determining the breaking point becomes more critical and it changes whole track even with a minimal change. In this case it is impossible to narrow the area to get correct result, because even with clean bearing it is hard to track target.

C. Cost Function

TMAGA uses a non-metric cost function called Total Deviation and shown in Eq. (8) where \(\Theta_{G_i}\) and \(\Theta_{M_j}\) are ground truth and calculated bearing angles of target at \(i^{th}\) sample point. So this could cause divergence instead of convergence.
The ideal cost function is must be like in Eq. (9) to obtain Euclidean distance properties and become a metric distance measure.

\[
\text{Total Deviation} = \sum_{i=1}^{N} (\Theta_{Mi} - \Theta_{Gi})^2
\]  

(9)

**D. Selection, Mutation and Crossover Operators**

Genetic algorithm’s mutation and crossover operators are somewhat weaker than real valued evolutionary strategy’s ones. Because bitwise crossover and mutation options are limited and binary numbers’ digit value is not meaningful by itself to these operators.

TMAGA uses the simplest selection method; individuals are weighted by their fitnesses obtained from cost function, then randomly picked from population using these weights. The highest weighted individual has the highest pick chance, but the lowest weighted one has being picked chance too. But this straight method is weak; because the highest two individual’s offspring won’t create a good individual every time, sometimes two bad individual could create the best individual; so stochastic universal sampling and elitism could help converge faster than used method.

TMAGA uses two piece crossover and randomly picked bitwise mutation methods. These methods are old fashioned for 2009 and their performances are not satisfying anymore. In real value measure non-uniform mutation and n-point real valued weighted averaging crossover methods could give much more better performance and easy to implement.

**E. Function Evaluations**

TMAGA’s main flow include two layered structure. Outer layer creates population, narrows search space and keeps track of best solution. Inner layer controls genetic algorithm operations. The flow can be seen in Fig. 2.

Outer and inner layers have Monte Carlo runs separately; so Monte Carlo run increases exponentially. TMAGA uses 20 inner, 20 outer Monte Carlo runs. Inner Monte Carlo is used for narrowing space and genetic algorithm run, outer one is used for statistical analysis.

At the end of process, outer layer calculates weighted average of the best results from Monte Carlo runs and labels it as the best result.

TMAGA uses 50 population size, 200 generations for space narrowing, 500 generations for genetic algorithm, 20 statistical, 20 in-system Monte Carlo counts. If we calculate function evaluation count with these informations; \(50 \times 700 \times 20 = 700000\) function evaluation will be done in a single Monte Carlo of TMAGA, if 20 statistical Monte Carlo runs completed it will be 14000000. 14 million function evaluation is too much for \(\text{limits}(X_0) = [20000, 0]\) meters, \(\text{limits}(C) = [360, 0]\) degrees and \(\text{limits}(S) = [25, 0]\) meters/seconds.

If we chose \((R_0|C|S)\) form Eq. (7) and \(\text{limits}(R) = [28000, 0]\) meters, \(\text{limits}(C) = [360, 0]\) degrees and \(\text{limits}(S) = [25, 0]\) meters/seconds; then search the space with 100 unit precision for \(R\), 0.5 unit precision for \(C\) and 0.1 unit precision for \(S\) using brute force; \(280 \times 720 \times 250 = 50400000\) function evaluation needed. Results for brute force searching is given in Table II with ground truth values in Table I. Even with 50400000 function evaluations, we cannot get correct results.

**TABLE I: Trials’ ground truth values.**

| Trial | \(R(m)\) | \(C(°)\) | \(S(m/s)\) | \(\sigma_{\text{noise}}(°)\) |
|-------|---------|----------|-----------|-----------------|
| 1     | 7994    | 90       | 10        | 0.3             |
| 2     | 7071    | 90       | 6         | 0.3             |
| 3     | 4006    | 90       | 6         | 0.3             |
| 4     | 8000    | 175      | 10        | 0.3             |
| 5     | 8000    | 150      | 10        | 0.3             |
| 6     | 8000    | 120      | 10        | 0.3             |
| 7     | 4006    | 90       | 10        | 0               |
| 8     | 4006    | 90       | 10        | 0.5             |
| 9     | 4006    | 90       | 10        | 1               |
| 10    | 4006    | 90       | 10        | 2               |
| 11    | 4006    | 90       | 10        | 5               |
| 12    | 4006    | 90       | 10        | 10              |

From Table II we can see that even brute force cannot find a clear solution for that high noise levels. Thus we can say noise is disrupting useful information in bearing data.
TABLE II: Brute force search results.

| Trial | R(m) | C(°) | S(m/s) | Trial | R(m) | C(°) | S(m/s) |
|-------|------|------|--------|-------|------|------|--------|
| 1     | 7940 | 93   | 10     | 7     | 4010 | 90   | 10     |
| 2     | 6490 | 60   | 6      | 8     | 8820 | 106  | 25     |
| 3     | 4110 | 84   | 0      | 9     | 5900 | 96   | 14     |
| 4     | 4440 | 177  | 20     | 10    | 6920 | 119  | 24     |
| 5     | 7810 | 105  | 4      | 11    | 3020 | 137  | 25     |
| 6     | 10010| 125  | 13     | 12    | 11570| 19   | 25     |

F. Statistical Results

We said that TMAGA has not guaranteed results before; because in the paper [4] there is no information about variance of Monte Carlo runs and statistics are only given for means or the best results. Thus we cannot determine the real performance of the TMAGA. Higher performances on higher error standard deviations like $\sigma_{\text{noise}} = 0.5$ and $\sigma_{\text{noise}} = 1.0$ in degree measure are suspicious. Additionally, perpendicular movement of target to observer makes tracking difficult; but TMAGA says it is tracking target very accurate with these error variances. Even we look at Table [I] brute force couldn’t get the precise results.

If we look at Table 3 which is given by [4], we could see Monte Carlo is done two times and mean of the second Monte Carlo is given as the average result. There is no information about variance or standard deviation; so this type of reporting is not acceptable. Solutions are obtained by weighted averages of the first Monte Carlo’s results. There is no statistical relation between best solution and fitness in extra noisy data condition.

IV. COVARIANCE MATRIX ADAPTATION EVOLUTIONARY STRATEGIES (CMA-ES)

In this section, we will explain how the Covariance Adaptation Evolutionary Strategies [9] works and model a new TMA system by CMA-ES.

A. CMA-ES

CMA-ES is a kind of evolutionary strategy that uses statistical feedback and covariance matrix for capturing search space shape. CMA-ES uses cost weighted recombination instead of crossover; random number generation using Gaussian distribution instead of mutation.

$m_i$ is the favorite solution at the start of each generation and could be obtained using Eq. (10). $w_{1:\lambda}$ is the weight obtained from the cost of the parent $i$ for the $\lambda^{th}$ offspring cluster, $x_i$ is the $i^{th}$ parent.

$$w_{1:\lambda} >= w_{2:\lambda} >= ... >= w_{\mu:\lambda} \Rightarrow \sum_{i=1}^{\mu} w_{i:\lambda} = 1 \Rightarrow m_i = \sum_{i=1}^{\mu} x_i w_{i:\lambda} \tag{10}$$

Covariance matrix $C$ determines the shape of the random number generation and step size $\sigma$ determines the width of probability density function. Initially we use $\sigma$ predefined around like a number 0.3 and $C$ unit diagonal matrix. We learn them each generation via the methods described by Hansen[9].

Then we use Eq. (11) to get new population individuals $x_i$ around the $m_i$ favorite individual.

$$x_i = m_i + \sigma_{\text{CMA}} N_i(0, C) \tag{11}$$

CMA-ES narrows the search space using statistical information learned by updating $C$ and $\sigma$. This narrowing helps CMA-ES converge with less calculations than classical evolutionary strategies. Fig. 3 shows how CMA works and converges to the result by the help of $C$ and $\sigma$.

![Fig. 3: Concept work of the CMA-ES.](image)

V. TRIALS

We’re using CMA [9] with cost function given by Eq. (4) to solve the target motion analysis problem. CMA is a powerful method as it converges quickly to the solution without stucking to local minima. Our parameters for CMA is given in Table [III]

| Parameter          | Value       |
|--------------------|-------------|
| No of Parameters   | 3           |
| Parent Size        | 100         |
| Offspring Size     | 100         |
| No of Generations  | 50000       |
| Function Evaluation Budget | 50000      |
| Range Limits       | [0, 28000] meters |
| Course Angle Limits| [0, 360] degrees |
| Speed Limits       | [0, 25] meters/second |
| No of Monte Carlo Runs | 1000      |

TABLE III: CMA parameters.

Results for CMA-ES searching is given in Table [IV] with ground truth values in Table [I] $\mu$ represents Monte Carlo mean, $\sigma$ represents Monte Carlo standart deviation and $|\text{Dev.}|$ represents the absolute deviation between estimation and ground truth. As we can see from Table [IV] author’s error standart deviation is too much for tracking and result variates in wide ranges. If we reduce error standart deviation by 0.1x, we can see that results are becoming much more reliable. We can say that maximum error standart deviation is $\sigma_{\text{noise}} = 0.2^\circ$.

CMA-ES solves problem with 50000 function evaluation which is 14 times smaller than 700000 function evaluation of brute force and results’ standart deviation is smaller enough to trust.
TABLE IV: CMA-ES search results.

| Trial | R(m) | C(°) | S(m/s) |
|-------|------|------|--------|
| 1     | 9732.594 | 89.608 | 13.075 |
|       | 4619.759 | 19.677 | 6.662  |
|       | 1738.371 | 0.391  | 3.075  |
| 2     | 8501.276 | 88.662 | 9.710  |
|       | 4105.756 | 36.325 | 6.609  |
|       | 1432.208 | 1.337  | 3.710  |
| 3     | 4471.893 | 90.753 | 7.272  |
|       | 1648.322 | 18.434 | 3.652  |
|       | 465.098  | 0.753  | 1.272  |
| 4     | 5064.392 | 83.3033| 8.4811 |
|       | 1504.822 | 48.075 | 6.732  |
| 5     | 8273.251 | 123.914| 13.074 |
|       | 2308.739 | 49.989 | 8.5072 |
|       | 273.244  | 26.086 | 3.074  |
| 6     | 9017.057 | 110.775| 13.375 |
|       | 3679.193 | 28.754 | 8.153  |
|       | 1017.051 | 9.224  | 3.375  |
| 7     | 4006.794 | 90     | 10     |
|       | 0        | 0      | 0      |
| 8     | 5119.677 | 90.475 | 13.389 |
|       | 2436.746 | 11.958 | 7.017  |
|       | 1122.883 | 0.475  | 3.389  |
| 9     | 5368.986 | 87.240 | 14.419 |
|       | 2994.409 | 18.946 | 8.552  |
|       | 1362.192 | 2.759  | 4.419  |
| 10    | 5360.204 | 83.377 | 14.792 |
|       | 3365.029 | 25.509 | 9.349  |
|       | 6632.410 | 6.622  | 4.792  |
| 11    | 5517.113 | 77.005 | 15.927 |
|       | 4140.429 | 37.484 | 9.844  |
|       | 1510.319 | 12.994 | 5.927  |
| 12    | 3877.869 | 90.121 | 16.445 |
|       | 5939.399 | 77.892 | 10.674 |
|       | 128.9249 | 0.121  | 6.445  |

VI. CONCLUSION

As conclusion we can say that genetic algorithm is a non-modern weak method to solve target motion analysis. Even using two different position parameters enlarges search space and creates correlation problems. Distance measure is wrong; because it don’t justify metric rules. Search space narrowing is useless because algorithm may converge to local minima with genetic algorithm, there is no feedback to avoid it. Statistical results are given in wrong way, and there is too much calculation to solve problem.

CMA-ES narrows space via statistical feedback and converges quickly without being stuck to local minima. Error standart deviation is too large to converge, with experiments we can say 0.2 is the maximum value for error standart deviation. As result we can get solution with 14 times smaller function evaluation count than brute force via CMA-ES.

REFERENCES

[1] A. Aytun and S. Bulkan, “Bearing-Only Target Motion Analysis,” in Operations Research for Military Organizations. IGI Global, 2019, pp. 330–346.

[2] T. Kronhamn, “Bearings-only target motion analysis based on a multiphypothesis Kalman filter and adaptive ownship motion control,” IEEE Proceedings-Radar, Sonar and Navigation, vol. 145, no. 4, pp. 247–252, 1998.

[3] H.-G. Beyer and H.-P. Schwefel, “Evolution strategies–A comprehensive introduction,” Natural computing, vol. 1, no. 1, pp. 3–52, 2002.

[4] L. Ince, B. Sezen, E. Saridogan, and H. Ince, “An evolutionary computing approach for the target motion analysis (TMA) problem for underwater tracks,” Expert Systems with Applications, vol. 36, no. 2, pp. 3866–3879, 2009.

[5] O. K. Erol and I. Eksin, “A new optimization method: big bang–big crunch,” Advances in Engineering Software, vol. 37, no. 2, pp. 106–111, 2006.

[6] H. Genc and A. Hocaoğlu, “Bearing-only target tracking based on big bang–big crunch algorithm,” in Computing in the Global Information Technology, 2008. ICCGI’08. The Third International Multi-Conference on. IEEE, 2008, pp. 229–233.

[7] A. Tokta, A. K. Hocaoğlu, and H. M. Genc, “Target motion analysis with fitness adaptive big bang-big crunch optimization algorithm,” in Signal Processing and Communications Applications Conference (SIU), 2017 25th. IEEE, 2017, pp. 1–4.

[8] N. Hansen, S. D. Müller, and P. Koumoutsakos, “Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES),” Evolutionary computation, vol. 11, no. 1, pp. 1–18, 2003.

[9] N. Hansen, “The CMA evolution strategy: A tutorial,” arXiv preprint arXiv:1604.00772, 2016.

[10] H. H. Sönmez, A. K. Hocaoğlu, and H. M. Genc, “A new solution method for the passive target motion analysis problem using evolutionary strategies,” in Signal Processing and Communications Applications Conference (SIU), 2017 25th. IEEE, 2017, pp. 1–4.

[11] H. Genc, “A new solution approach for the bearing only target tracking problem,” in Soft Computing Applications (SOFA), 2010 4th International Workshop on. IEEE, 2010, pp. 95–100.

[12] D. Zuras, M. Cowlishaw, A. Aiken, M. Applegate, D. Bailey, S. Bass, D. Bhandarkar, M. Bhat, D. Bindel, S. Boldo, and others, “IEEE standard for floating-point arithmetic,” IEEE Std 754-2008, pp. 1–70, 2008.