**Scientific Journals**
**Maritime University of Szczecin**

**Zeszyty Naukowe**
**Akademia Morska w Szczecinie**

**2012, 32(104) z. 2 pp. 88–93**

**2012, 32(104) z. 2 s. 88–93**

# Software implementation of multiple model neural filter for radar target tracking

**Witold Kazimierski, Natalia Wawrzyniak**

Maritime University of Szczecin, Faculty of Navigation, Chair of Geoinformatics
70-500 Szczecin, ul. Wały Chrobrego 1–2, e-mail: {w.kazimierski; n.wawrzyniak}@am.szczecin.pl

**Key words:** radar target tracking, multiple model filters, neural networks

**Abstract**

The paper presents a software implementation of multiple model neural filter for radar target tracking. Such a filter may be proposed as an interesting alternative for numerical filters. The main purpose of software implementation is to provide a tool for complex research of the filter possibilities and adjusting options. A concept of a filter is briefly mentioned, however the main body of paper is focused on user-approach detailed description of application with UML use-case diagrams. Examples of detailed presentation of use-cases are given and the general use-case diagram for application is included. The application itself is to be an advanced tool for researchers interested in analyzing target tracking process, providing different tracking methods and the possibility of adjusting their parameters. The possibility of simulating any scenario, as well as working with real data (also on-line) was ensured. The research was financed by Polish National Centre of Science under the research project "Development of radar target tracking methods of floating targets with the use of multiple model neural filtering".

## Introduction

Marine radar is one of the key devices on the bridge of modern vessels. According to Resolution of Maritime Safety Committee, the radar equipment should assist in safe navigation and in avoiding collision by providing an indication, in relation to own ship, of the position of other surface craft, obstructions and hazards, navigation objects and shorelines. For this purpose, radar should provide the integration and display of radar video, target tracking information, positional data derived from own ship's position (EPFS) and geo referenced data [1].

Radar target tracking seems to be of vital importance in the aspect of collision avoidance. It can be defined as estimating target's state vector on the basis of measured relative positions and own ship movement. Based on the information from tracking system the OOW may judge the situation and make a decision about the suitable manoeuvre to solve collision situation. Thus, the quality of target tracking has a direct impact on safety of navigation. The errors and delays of estimation can cause serious misunderstandings and wrong decisions [2].

The quality of tracking depends on the tracking filter used. Commonly used methods are different modifications of Kalman filter. An interesting example of these might be a group of so called multiple model methods. The main concept is to provide a few elementary filters for tracking targets with different movement dynamics (different filter parameters are required) and join their output for one final estimation. The method is well described for numerical filters. However, it may be applied to other filters.

Multiple model neural filter for radar target tracking is a concept developed in the research project, which has been carried out in Maritime University of Szczecin, entitled "Development of methods of marine radar target tracking with the use of multiple model neural filtering". The research carried out so far has shown that it is legitimate to use such a method for radar target tracking. The research results have shown also that, different neural network parameters should be used for the tracking targets with different dynamics of movement. Thus, the concept of multiple model filter arose. The filter consists of a few independent neural networks with different parameters connected with interaction module, which is responsible for proper joining (mixing) of outputs of these networks. Final verification of the quality of filtering requires suitable implementation of filter in dedicated software. It allows to perform both kind of verification research – simulated performance and performance based on real data. In order to analyze simulated data, a method of Monte-Carlo simulation has been implemented. The data are simulated directly in the application with the use of radar measurements errors generator. In order to analyze real situation the possibility of decoding NMEA 0183 data is ensured. The data may be read from text file placed on the disk or the data can be also read on-line from one of the serial port of the PC on which the application is installed. The paper presents the specification of the application including verbal description and UML diagrams. This allows precise and synthetic presentation of functions and possibilities in the application, as well as the goal and requirements and graphical user interface. The application is opened for future amendments. This shall allow complex analysis of radar target tracking.

## Multiple model neural tracking filter

There are many tracking algorithms for estimating movement vector of radar targets. Most of them are the modifications of Kalman filter. The main idea of the modification is to include non-linearity into tracking process. One of the possible approaches is the concept of so called multiple model filters. A few elementary filters are working simultaneously and the final estimation is computed as the combination of elementary estimations. The state and measurements equations are as follows [3, 4]:

$$x_{i+1} = F\left[M_{i+1}\right]x_i + v\left[i, M_{i+1}\right] \quad (1)$$

$$z_{i+1} = G\left[M_{i+1}\right]x_{i+1} + w\left[i+1, M_{i+1}\right] \quad (2)$$

where $M_{i+1}$ is the model, which has been used at the moment of measurement $i + 1$. The model is chosen or combined for the actual state from the bank of $r$ modes:

$$M_{i+1} \in \left\{M^j\right\}_{j=r}^{r} \quad (3)$$

There are many multiple model filters varied with the methods of interaction between elementary filters. Many algorithms for numerical filtration can be found in [5, 6].

There is however another approach possible for radar target tracking, namely the use of neural networks. The research on this has been carried out in Maritime University of Szczecin for a few years. The results have shown that the use of General Regression Neural Network for the needs of target tracking may be an interesting alternative against numerical filters. Different variations of such a filter have been presented in [7, 8]. It has been observed that different network parameters shall be used for different movement dynamics. Thus the concepts of multiple model filtering and neural filter can be combined into one multiple model neural filter. It consists of a few neural networks working in parallel and their outputs are combined together as one final estimation. Proper determination of filter's parameters is here the key issue.

General regression neural network has been proposed in [9] for solving regression problems. It is in fact the network structure for so called kernel regression. In a nutshell, the operation of the GRNN for target tracking, gives the output value for those new points is estimated by the network using the weighted mean of inputs for the teaching examples, where weights depend on the distances of particular training points from the point for which the estimation is performed.

The structure of GRNN itself may be adjusted with one of the three parameters:

- the length of teaching sequence;
- the smoothing factor;
- activation function.

Apart of these also filter parameters may be adjusted like the number of models, manoeuvre detection methods or interaction rules.

There is a need for dedicated software which could allow thorough research and examination of multiple model neural filter and its parameters in different tracking scenarios. The example algorithm of two model neural filter may be found in [10].

## General system architecture

Software implementation of multiple model tracking filter is the main task of the designed application. However, it should be perceived also in the wider aspect as a part of the entire system suitable for testing tracking methods in simulated and real marine environment. In this concept the application is an extension for a radar set.

During the research project a proper radar set has been proposed. It can be mounted either on a ship or as a mobile land station independent from vessel's sensors. The scheme of the radar set in ships' version is presented in the figure 1. The system can integrate many external information for other than radar sensors. However, the figure includes only the sensors important form the tracking

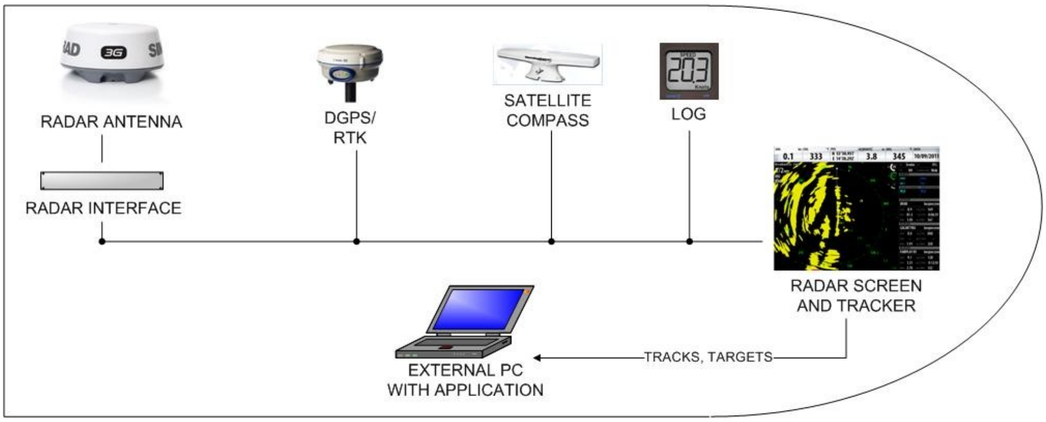


Fig. 1. Radar set for radar target tracking research

point of view, namely: positioning system, heading and speed devices. The power supply for elements of the system comes from the ship's supply.

In the variant of shore station the power supply comes from the external power generator. There is also a problem of heading, while the radar works as stationary station. Suitable NMEA sentences are simulated on PC and transmitted to tracking filters. The information about targets is transmitted to PC via NMEA TTM sentence. Received information is used as a measurement for tracking filters (bearing and distance) and as reference information for filters comparison (course and speed). In general the application's functionality is to provide target tracking for radar targets with different filters and with the possibility of adjusting of parameters of filters.

## Functional concept of the application

User-centered approach was used to present functional concept of the implementation. UML use-case diagrams allows to represent operations required to execute a task assigned by its user [11]. They show the main functions performed by the system and the resources from outside of system needed for the application to perform its main tasks. Use-case diagrams are commonly used for software documentation [12]. This kind of specification consists of an UML diagram, actors description document and possible scenarios tables.

There are five determined actors that interact with the system (Fig. 2). The obvious, main actor is an application user. He can start, end and operate the application as well as change systems parameters to obtain better / different tracking results. Inferior actors of the systems are: a radar as real-time tracking system, a file with previously recorded radar data, an error generator used for simulated data and a file as the applications output data storage.

As seen in the figure 2 there are two sets of use-cases that can be performed. One is strictly related to the main purpose of the software – tracking by acquiring filtered ship tracks and movement parameters as its course and speed. The other group of functions is related to building new filters or adjusting parameters for multimodal neural filtering.

The main flow of the diagram seems more complicated while it uses more independent functions for performing real-time or simulated tracking. In the software itself two kinds of filters were implemented – the numerical Kalman filter and the multiple mode neural GRNN filter. If the application runs on radar equipped unit then tracks of targets are filtered on-the-fly and the results (movement vector) are visualized and written to a proper text file. If the software is used for testing and adjusting filters parameters then the errors generator is used to provide noise for simulated targets scenario. Application in offline mode also can replay tracks from a file containing NMEA protocol as it was real-time data and use the same built-in filters to obtain and output the result into a file. This most commonly used scenario is presented in a table 1.

Worth mentioning is the Monte Carlo algorithm implemented for management of errors being generated. It allows to run the simulated scenario defined number of times (initially set to 100) with different errors. Their weighted mean is calculated together with standard deviation and the output is presented on-screen.

The other group of use-cases is related to either creating and adding new filters or adjusting parameters of GRNN elementary filters and multiple model filter to improve tracking results.

As application is dedicated for research and use of multi-modal neural methods for filtering, it allows many options to adjust designed GRNN network for tracking enhancement. It also allows to add any type of numeric filter with any parameters

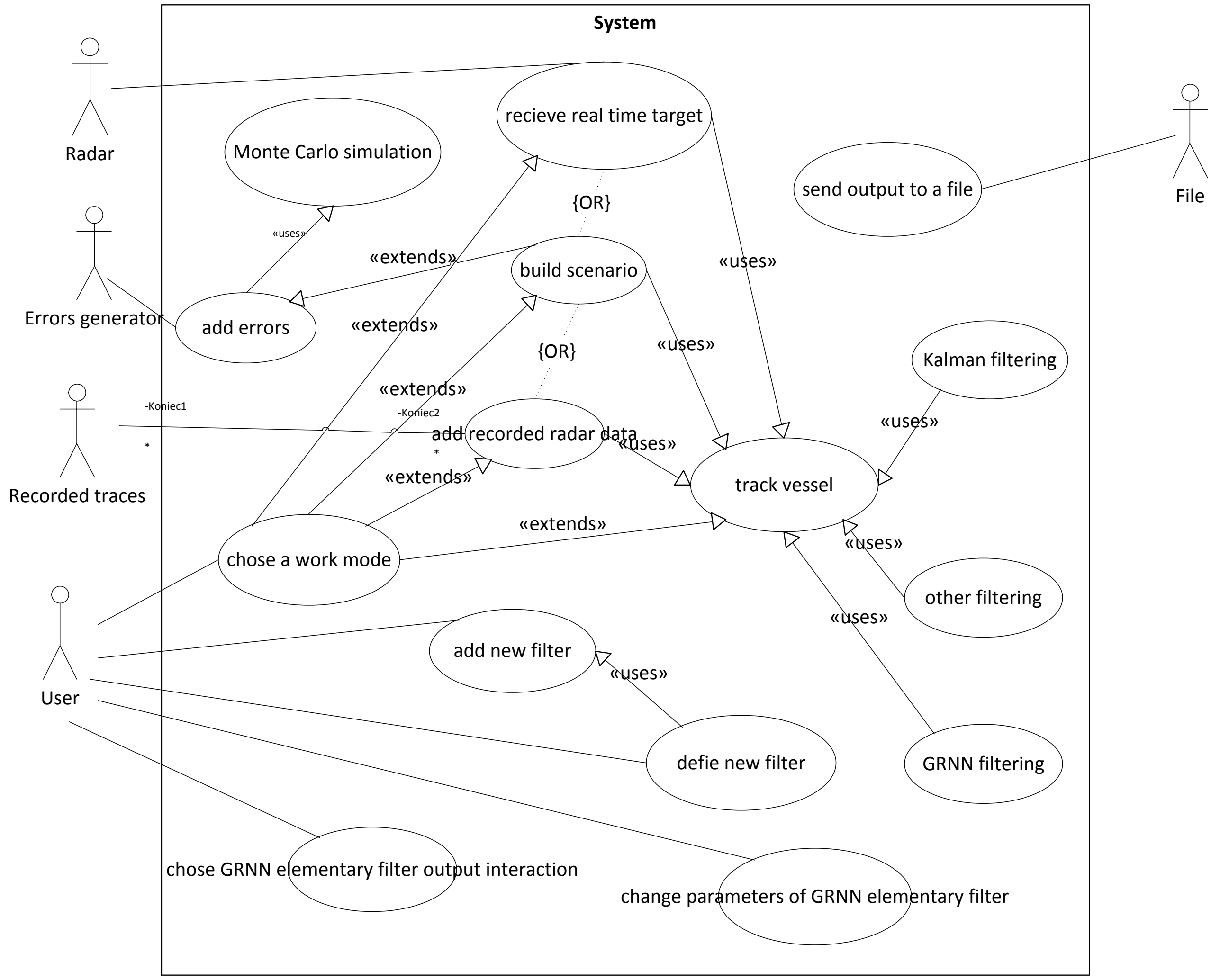


Fig. 2. UML use-case diagram of the application

Table 1. Example of use-case scenario – track vessel

| Use-case | Description | | |
|---|---|---|---|
| track vessel | Choose a work mode (A, B,C) | | |
| | 2A. Wait for NMEA from the radar | 2B. Open file with recorded NMEA | 2C.<br>a) Plan a scenario<br>b) (optional) Start MonteCarlo simulation<br>c) Start errors generator<br>d) Add noise to tracks |
| | Heading and bearing calculation<br>Simultaneously filter with chosen filters (Kalman, multimodal GRNN, other created)<br>Visualize estimated position and vector<br>Write each filtered output (course and speed) to a separate file<br>Write calculation for each error to a separate file | | |
| | Initial conditions: properly configured port for NMEA transmission<br>Existing file with proper NMEA content<br>Properly planned scenario<br>End conditions: filtered course and speed | | |
| | Remarks: User can stop tracking any time while the function is running.<br>User can start tracking (in any work mode) only if the process is not running already.<br>Filtration runs simultaneously for all given filters | | |

defined together with it, as long as it recognizes bearing and distance as input values and course and speed as output. However, adjusting its individual parameters from the level of the software isn't possible, as this is not a task of neural filtration and numerical filters are used only for comparison.

Table 2. Example of use-case scenario – add filter

<table>
<tr><td>Use-case</td><td colspan="2">Description</td></tr>
<tr><td>Add filter</td><td colspan="2">Define type of filter (numeric A, neural multi-modal GRNN B).</td></tr>
<tr><td></td><td>2A. Add filter to library catalog.</td><td>2B<br>a) Define number of filters.<br>b) Define parameters for each of elementary neural GRNN filters.<br>c) Define interaction parameters for elementary filters output.<br>d) Add filter to library catalog.</td></tr>
<tr><td></td><td colspan="2">Confirm created filter.<br>Chose new filters files path for tracking output.<br>Save changes.</td></tr>
<tr><td></td><td colspan="2">Initial conditions: User can perform this action only if tracking procedure is not already running. User can quit this action without saving data any time<br>End conditions: new filter created</td></tr>
</table>

Input data for tracking is text data embedded in NMEA 1083 protocol. One of the NMEA sentences $RATTM covers bearing and heading readout of a target. System awaits and reads the data when it is provided with every turn of the radar antenna. An example of NMEA 0183 registration of radar track is presented in figure 3. If the data is replayed, the application uses files with recorded NMEA messages. In the case of simulation the message is generated exactly every 3 seconds.

```
$RATTM,01,3.82,115.5,T,7.9,216.0,T,2.7,10.1,N,,T,,164152.00,M*08
$RATTM,02,5.62,92.5,T,13.3,53.2,T,5.6,0.2,N,,T,,164152.00,M*07
$RATTM,03,0.35,122.7,T,11.6,110.3,T,0.3,-4.5,N,,L,,164152.00,M*3F
$RATTM,05,0.40,9.4,T,19.6,168.4,T,0.1,1.3,N,,T,,164152.00,M*04
$RATLL,103,5432.158,N,01837.442,E,STENA BA,164152.00,T,*70
$RATLL,104,5423.945,N,01842.360,E,,164152.00,T,*18
$RATLL,105,5431.708,N,01831.780,E,,164152.00,T,*13
$RATLL,106,5432.750,N,01830.370,E,DENEB,164152.00,T,*5C
$RATLL,107,5432.094,N,01832.181,E,KAPITAN ,164152.00,T,*7C
```

Fig. 3. An example of data recorded in NMEA 0183 format, used as an input for an application

Output is automatically displayed on a screen. Besides that a two text files are generated for each kind of data stream: unfiltered track, track after numeric filtration, track after GRNN filtration and track after any additional type of filtration added to the application. First file saves course and speed of the vessel. In the other errors for each position are saved.

## Graphical user interface

Graphical user interface in many applications is designed to be most suited and friendly for future user, which are usually only using the functions and not interacting with them. This time the idea of the interface is to allow the researcher adjusting filter's parameters and scenarios without the need of working with applications code. Thus, the main impact is laid on the functionality and possibilities of the interface rather than on its design and lookout.

The interface consists of the main window and three forms for adjusting different parameters, namely:

– filter's parameters;
– simulation parameters;
– interacting parameters.

Main window layout is showed in the figure 4.


Fig. 4. GUI – layout of the main window of application

The central panel presents the "bird's view" with the movement of target and of own ship. This may be therefore considered as position plan indicator, like in real radars. On the right side two panels presenting vessel's data can be found. Above all the panels menu can be found. By selecting suitable function from the menu, user can open the additional forms.

The form *Filter Parameters* allows to choose what kind of filter shall be used for tracking – numerical filter or neural filter and the parameters of neural filter – how many models and the parameters of each model in details like length of teaching sequence and smoothing factor.

The form *Simulation parameters* offers the possibility of designing any situation scenario for simulation or selection of working on-line. Different movement of targets can be chosen – in-line uniform motion, course manoeuvring or speed manoeuvring. Various dynamics of selected manoeuvres can also be chosen by adjusting rate of turn or speed rate.

The form *Interacting parameters* allows to select the method of mixing of outputs of elementary filters, as well as method of manoeuvre detection in case if detection-based approach is selected. Different parameters of selected method like transition matrix values can be here set.

Presented GUI makes easier working with application and allows quite fast adjusting of the filter and situation for the research.

## Conclusions

Multiple model neural filter seems to be a promising alternative for numerical tracking filters commonly used nowadays in marine radars at sea. Initial research conducted so far have given interesting results. More sophisticated research however were needed to examine the method and its parameters thoroughly. This could have only been done with software implementation of an algorithm. Such an application has been prepared and presented in this paper.

The application includes not only software implementation of multiple model neural tracking filter but also a set of methods and functions for designing research on target tracking. From this point of view an application is a kind of PC-based simulator with the possibility of working with real data on-line and off-line. The filter itself is implemented in such a way that all possible parameters can be easily and freely adjusted. The simulation scenario can also be changed as per user request.

The application contains also other filter, like Kalman Filter and IMM is also planned. Furthermore there is a possibility of receiving data from tracking filter in real radar. This allows to compare results obtained with neural filter to other solutions.

## References

1. IMO Resolution MSC.192(79), Adoption of the revised performance standards for radar equipment, 2005.
2. BOLE A.G., DINELEY W.O., WALL A.: Radar and ARPA Manual. Elsevier Science & Technology Book, 2005.
3. BLACKMAN S., POPOLI R.: Design and Analysis of Modern Tracking Systems. Artech House, Norwood USA, 1999.
4. BAR SHALOM Y., LI X.R.: Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software. John Wiley & Sons, Inc., NY USA, 2001.
5. LI X.R, JILKOV V.P.: A Survey of Manoeuvring Target Tracking. Part V: Multiple-Model Methods. IEEE Transactions on Aerospace and Electronic Eystems, Vol. 41, 2005.
6. BAR SHALOM Y., LI X.R.: Estimation and tracking: principles, techniques, and software. YBS, Norwood 1998.
7. STATECZNY A. (ed.): Radar navigation. GTN, Gdańsk 2011 (in Polish).
8. KAZIMIERSKI W.: Two-stage General Regression Neural Network for radar target tracking. Polish Journal of Environmental Studies, Vol. 17, No. 3B.
9. SPECHT D.F.: A General Regression Neural Network. IEEE Transactions on Neural Network, Vol. 2, No. 6, 1991.
10. STATECZNY A., KAZIMIERSKI W.: The Process of Radar Tracking by Means of GRNN Artificial Neural Network with Dynamically Adapted Teaching Sequence Length in Algorithmic Depiction. Proceedeings of 7th International Symposium of Navigation TrnasNav2007, Gdynia 2007.
11. STEMPOSZ E., PŁODZIEŃ J.: Analiza i projektowanie systemów informatycznych. Wydawnictwo PJWSTK, Warszawa 2003.
12. MISER H.J., QUADE E.S.: Handbook of System Analysis. Wiley 1985.