THREAT ESTIMATION OF MULTIFUNCTION RADARS: MODELING AND STATISTICAL SIGNAL PROCESSING OF STOCHASTIC CONTEXT FREE GRAMMARS

Alex Wang, Vikram Krishnamurthy

University of British Columbia
Department of Electrical Engineering
2332 Main Mall, Vancouver, BC Canada V6T 1Z4

ABSTRACT

Multi-Function Radars (MFRs) are sophisticated sensors with complex dynamical modes that are widely used in surveillance and tracking systems. It is shown in this paper that the stochastic context free grammar (SCFG) is an adequate model for capturing the essential features of the MFR dynamics. We model MFRs as systems that "speak" according to a SCFG, and the grammar is modulated by a Markov chain representing MFRs' policies of operation. We then deal with the statistical signal processing problems of the MFR signal, especially the problem of threat evaluation (electronic support). Maximum likelihood estimator is derived to estimate the threat of the MFR and Bayesian estimator to infer the system parameter values.

Index Terms— electronic warfare, formal languages, maximum likelihood estimation, radar signal processing

1. INTRODUCTION

Multi-function radars (MFRs) are radio-frequency sensors that are widely used in modern surveillance and tracking systems. They have the capability to perform multitude of different tasks such as search, acquisition and target tracking. MFRs use electronic beam-steering antennas to perform multiple tasks simultaneously by multiplexing them over short time scales. Finite Markov models have been widely used to model the dynamics of conventional radars [1], however, due to the MFRs' sophisticated nature, traditional radar signal processing algorithms are not suited to the structure and the complexity of MFRs' signal. The main idea of the paper is to extend the finite Markov model with a more general dynamical structure called *stochastic context free grammar* (SCFG). We model MFRs as a Markov modulated SCFG, and then present a statistical signal processing algorithm for evaluating the threat MFR poses on its target.

Traditionally, MFRs' signal modes are represented by volumes of parameterized data records known as Electronic Intelligence (ELINT) [2]; the data records are annotated by lines of text explaining why, when and how a signal may switch modes. When it comes to rapid radar mode estimation and threat evaluation, ELINT is not suitable. Based on syntactic modeling [3], SCFG is the new methodology to model MFRs' signal and it has several potential advantages: i) SCFG is a compact representation for modeling complex system dynamics, and it allows model designers to more naturally express MFRs' control rules, and thus allows more convenient modeling of the human computer interface. ii) Compared to stochastic regular grammar, or equivalently hidden Markov model, if the same number of parameters are used, SCFG is more efficient in modeling hidden branching process; the predictive power of a SCFG measured in entropy is greater than that of the stochastic regular grammar [4], and iii) the recursive embedding structure of MFRs' control rules is more naturally modeled in SCFG; the Markovian type model has dependency of variable length, and the maximum range dependency must be considered.

In summary, the main results of the paper are: 1) A detailed model of MFRs' dynamics using SCFG. A MFR can be viewed as a discrete event system that "speaks" some known, or partially known, formal languages. 2) Novel use of Markov modulated SCFG to model MFRs with changing policies of operation. 3) The threat evaluation problem is reduced to a state estimation problem and maximum likelihood estimator is derived based on a hybrid of the forward-backward and the inside-outside algorithm. The rest of the paper is organized as follows. Sec. 2 summarizes the electronic support problem and MFRs' system architecture. Sec. 3 models MFRs' signal based on a careful study of its distinguishing features. Sec. 4 presents the threat estimation algorithm and Sec. 5 concludes the paper.

2. ELECTRONIC SUPPORT AND MFR

Electronic Warfare (EW) can be broadly defined as any military action with the objective of controlling the electromagnetic spectrum [2]. In this paper, a sub-division of EW called electronic support is considered where the goal is to protect

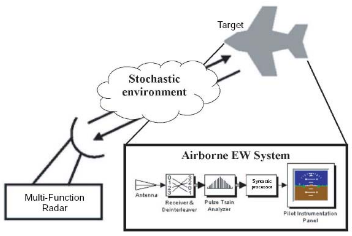

Fig. 1. The Electronic Support framework.

targets from a radar-equipped threat by collecting radar emissions and evaluating threat in real time. The specific threat considered is multifunction radars.

The framework of the electronic support considered in this paper consists of three layers: Receiver/ Deinterleaver, Pulse train analyzer and Syntactic processor [3]. The layers are depicted in Fig.1 and a brief description is given here: The receiver processes the radar pulses intercepted by the antenna, and outputs a sequence of pulse descriptor words, which is a data structure containing parameters such as carrier frequency, pulse amplitude or pulse width. The deinterleaver processes the pulse descriptor words, groups them according to their possible originating radar emitters and stores them in their corresponding track files. The pulse train analyzer processes the track file, and further groups the pulse descriptor words into radar words. (See below for definitions.) Finally, the syntactic processor analyzes the syntactic structure of the radar words, estimates the state of the radar system and its threat level, and outputs the results on a pilot instrumentation panel. Because the receiver, deinterleaver and pulse train analyzer have been well studied, the syntactic processor is the focus of this paper.

The main purpose of the syntactic processor is to capture the structural patterns in the MFRs' signal and evaluates its threat. We will start by studying the basic assumptions of the MFRs' signal. The building blocks making up MFRs' signal are defined as follows: i) *Radar word*: A fixed arrangement of finite number of pulses that is optimized for extracting a particular target information. For example pulses with a fixed pulse repetition frequency. ii) *Radar phrase (radar task)*: Catenation of finite number of radar words. Each phrase may be implemented by more than one catenation of radar words. Examples are search and target acquisition. iii) *Radar policy*: Pre-optimized schemes that allocate different amount of resources to different radar phrases. An example is rules of engagement or policies of operation. The generation process of radar words is governed by the MFRs' system architecture[1], and which is illustrated in Fig. 2. A MFR consists of three main components: Situation assessment, System manager and Phrase scheduler/Radar controller. The situation assessment module provides feedback of the tactic environment, and the system manager, based on the feedback, selects a radar policy. Each radar policy is a resource allocation scheme that represents trade-offs between different performance measures, and it dictates how the phrase scheduler/radar controller will operate.

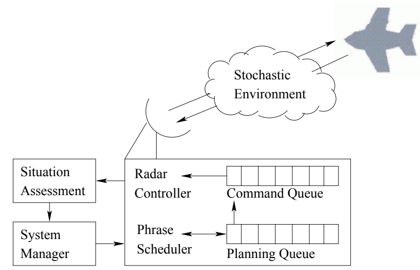

Fig. 2. MFR consists of a situation assessment module, a system manager, a radar controller and a phrase scheduler.

MFRs are modeled with two queues because of their need to be adaptive and fast [5]. Phrase scheduler pro-actively monitors the feasibility of the radar phrases in the planning queue sequentially [1]. It processes different types of radar phrases by their corresponding control rules; the rule takes the radar phrase being processed as input, and responds by appending appropriate radar phrases into the command queue and/or the planning queue. The selection of the control rules is a function of radar policies, and which are expressed by how probable each rule would be selected. Radar controller, on the other hand, allows the MFR to have finite response time. It processes the radar phrases in the command queue sequentially and maps them to a multitude of different radar words according to a set of control rules. Such an arrangement follows the macro/micro architecture as described in Blackman and Popoli [1].

3. A SYNTACTIC APPROACH TO MFR

For a description of the formal language theory, please see [6]. As an illustrative example showing the correspondence between the grammar and the MFR, consider the grammatical production rules of the form i) $A \rightarrow a\ A$ and ii) $A \rightarrow B\ A$, where A and B are radar phrases in the planning queue and a is a radar phrase in the command queue. $A \rightarrow a\ A$ is interpreted as a control rule that append a to the command queue, and A to the planning queue. Similarly, $A \rightarrow B\ A$ is interpreted as preempting A in the planning queue and inserting B in front of A. Suppose the planning queue contains a radar phrase A, a possible realization of the radar word generation process is illustrated in Fig.3. It can be seen that as long as the command queue phrases appear only to the left of planning queue phrases in the rule, the command queue and the planning queue are well represented.

[1] The system architecture does not include multiple target tracking functionalities such as data association. The paper focuses on a single target's self protection and threat estimation, and thus models only the radar signal that a single target can observe.

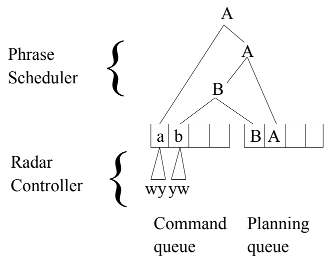

Fig. 3. A possible realization of the scheduling process represented by a grammatical derivation process. A and B are radar phrases in the planning queue, a and b are radar phrases in the command queue, and w and y are radar words.

Because the syntactic modeling is application dependent, for illustrative purpose, the discussion is based on a particular MFR called Mercury (the declassified version of the radar's textual intelligence reports can be found in [7]). The set of radar words, $\{w_1, \ldots, w_9\}$, consists of nine distinct elements. The set of radar phrases is {3-Word search, 4-Word search, Acquisition, Non-adaptive track, three stages of Range resolution, Track maintenance, Fine track maintenance}, and it is written in shorthand as $\{3WS_t, 4WS_t, A_t, NAT_t, RR1_t, RR2_t, RR3_t, TM_t, FTM_t\}$, where t = p or c denoting planning queue phrases or command queue phrases respectively.

Radar Controller The main purpose of the radar controller is the mapping of the command queue phrases to radar words. The production rules associated with the mapping are listed below, and they are constructed based on the syntactic pattern of the radar words [7].

$$
\begin{array}{llll}
4WS_c & \to W_1W_2W_4W_5| & NAT_c & \to S_1T_6|Q_6 \\
 & W_2W_4W_5W_1| & RR1_c & \to W_7T_6 \\
 & W_4W_5W_1W_2| & RR2_c & \to W_8T_6 \\
 & W_5W_1W_2W_4 & RR3_c & \to W_9T_6 \\
3WS_c & \to W_1W_3W_5W_1| & A_c & \to Q_1|Q_2|Q_3|Q_4|Q_5|Q_6 \\
 & W_3W_5W_1W_3| & TM_c & \to S_1T_6|S_2T_7|S_2T_8|S_2T_9 \\
 & W_5W_1W_3W_5 & FTM_c & \to Q_6|Q_7|Q_8|Q_9 \\
S_2 & \to S_1|W_6 & T_6 & \to W_6W_6W_6 \\
S_1 & \to W_1|W_2|W_3|W_4|W_5 & T_8 & \to W_8W_8W_8 \\
Q_i & \to W_iW_iW_iW_i & T_9 & \to W_9W_9W_9 \\
W_i & \xrightarrow{p_i} w_i \text{ for} i = 1, \ldots, 9 & &
\end{array}
$$

Phrase Scheduler The phrase scheduler models the MFRs' ability to plan and to preempt radar phrases. To simplify the discussion, suppose the planning queue phrases are $\{A, B, C\}$ and the command queue phrases are $\{a, b, c\}$, the basic control rules that are available to the phrase scheduler are i) Markov $B \to bA|bB|bC$, ii) Adaptive $B \to AB|BC$ and iii) Terminating $B \to b$. The interpretation of the rules follows the example given at the beginning of the section.

The significance of the Markov rule is obvious. The adaptive rules, on the other hand, model MFRs' ability to reschedule radar phrases when the system loading or the tactic environment changes. The two adaptive rules model the MFRs' ability to i) Preempt and ii) Plan the radar phrases. The preempt ability is demonstrated in the rule $B \to A\ B$ where the radar phrase B is preempted when a higher priority task A enters the queue. The ability to plan is captured in the rule $B \to B\ C$ where the phrase C is scheduled ahead of time if its predicted performance exceeds a threshold. Furthermore, the terminating rule reflects the fact that the queues have finite length, and the grammatical derivation process must terminate and yield a terminal string of finite length. The specification of the grammar is complete if the production rules' probabilities were assigned, and which will be discussed next.

Remark: The set of production rules presented above is a self-embedding context free grammar and thus its language is not regular, and cannot be represented by a Markov chain [6]. A context-free grammar is self-embedding if there exists a non-terminal A such that $A \overset{*}{\Rightarrow} \eta A \beta$ with $\eta, \beta \in (N \cup T)^+$. For the rules presented, self-embedding property can be shown by a simple derivation $B \to A\ B \to A\ B\ C$.

System Manager The phrase scheduler and the radar controller form the context free backbone of the MFRs' grammar. The system manager, for each time period, assigns probabilities to the backbone based on the policies of operation, its internal system state. The evolution of the system manager is modeled as a Markov chain, and in essence, MFR is a Markov modulated SCFG.

Let $k = 0, 1, \ldots$ denotes discrete time. The state, x_k, is a M-state Markov chain. Define the transition probability matrix as $A = [a_{ji}]_{M \times M}$, where $a_{ji} = P(x_k = e_i | x_{k-1} = e_j)$ for $i, j \in \{1, 2, \ldots, M\}$. In each state, specific probability values are assigned to the production rules of the MFR's context-free backbone, and thus the MFR is modeled to "speak" a different "language" in each state according to its state grammar. One practical issue is that the signal generated by radar systems has finite length, and this finiteness constraint is discussed by first defining the stochastic mean matrix.

Definition *Let $A, B \in N$, the stochastic mean matrix M_N is a $|N| \times |N|$ square matrix with its (A, B)th entry being the expected number of variables B resulting from rewriting A:*

$$M_N(A, B) = \sum_{\eta \in (N \cup T)^* s.t. (A \to \eta) \in P} P(A \to \eta) n(B; \eta)$$

where $P(A \to \eta)$ is the probability of applying the production rule $A \to \eta$, and $n(B; \eta)$ is the number of instances of B in η [8].

The finiteness constraint is satisfied if the grammar in each state satisfies the following theorem.

Theorem *If the spectral radius of M_N is less than one, the generation process of the stochastic context free grammar will terminate, and the derived sentence is finite.*

Proof The proof can be found in [8].

4. THREAT AND PARAMETER ESTIMATION

Threat evaluation is reduced to the estimation of MFR's policies of operation. Let $x_{0:n} = (x_0, \ldots, x_n)$ be the (unknown) state sequence, and $\gamma_{1:n} = (\gamma_1, \gamma_2, \ldots, \gamma_n)$ be the corresponding intercepted radar signal stored in the track file. Each $\gamma_k = (w_1, w_2, \ldots, w_{m_k})$ is a string of concatenated terminal symbols (radar words), and m_k is the length of γ_k.

Assuming complete knowledge of system parameters, i.e., the Markov chain's transition matrix and the SCFG's production rules, the MFR's state estimation could be trivially implemented using the Viterbi algorithm and the Inside algorithm [2]. In reality, such parameters are often unknown and inferrence of the parameter values is essential. In this section, EM algorithm is applied to a batch of noisy radar signal, and parameters are estimated iteratively. In EM's terminology, the radar words, $\gamma_{1:n}$, is the incomplete observation sequence, and it is made complete if augmented with $\{x_{0:n}, C_{1:n}\}$. $C_{1:n} = (C_1, \ldots, C_n)$ is the number of counts each production rule is used to derive $\gamma_{1:n}$, and in particular, $C_k = (C^1(A \to \eta; \gamma_k), C^2(A \to \eta; \gamma_k), \ldots, C^M(A \to \eta; \gamma_k))$ and $C^i(A \to \eta; \gamma_k)$ is the number of counts grammar G_i applies the production rule $A \to \eta$ in deriving γ_k.

Denote the model parameters as $\Phi = \{\, a_{ji}, P^1(A \to \eta), P^2(A \to \eta), \ldots, P^M(A \to \eta)\}$, where $P^i(A \to \eta)$ is set of production rule probabilities of grammar i, for any $\phi \in \Phi$, the complete-data likelihood is

$$\mathcal{L}_n(\phi) = \prod_{k=1}^{n} P(\gamma_k, C_k | x_k, \phi) P(x_k | x_{k-1}, \phi) P(x_o | \phi).$$

The Expectation step of the EM algorithm yields:

$$\begin{aligned}
&E_{\tilde{\phi}}(\log \mathcal{L}_n(\phi)) = \\
&\sum_{k=1}^{n} \sum_{x_k} \sum_{A^{x_k}} \sum_{T^{x_k}} E_{\tilde{\phi}}(C^{x_k}(A \to \eta; \gamma_k)) \log P^{x_k}(A \to \eta) \chi_{x_k}(k) \\
&+ \sum_{k=1}^{n} \sum_{x_k} \sum_{x_{k-1}} \log(a_{x_k|x_{k-1}}) \xi_{x_{k-1}x_k}(k-1) \\
&+ \sum_{k=1}^{n} \sum_{x_0} \log P(x_0) \chi_{x_0}(k)
\end{aligned}$$

where $E_{\tilde{\phi}}(C^{x_k}(A \to \eta; \gamma_k))$ can be computed using inside and outside algorithms [4], and $\chi_i(k) = P(x_k = e_i | \gamma_{1:n})$ and $\xi_{ji}(k) = P(x_k = e_j, x_{k+1} = e_i | \gamma_{1:n})$ are variables as defined in [9] that can be computed readily with forward and backward algorithms.

The Maximization step of the EM algorithm could be computed by applying Lagrange Multiplier. Since the parameters we wish to optimize are independently separated into three terms in the sum, we can optimize the parameter term by term. The updating equation of the production rule probabilities is

$$P^{x_k}(A \to \eta) = \frac{\sum_{k=1}^{n} E_{\tilde{\phi}}(C^{x_k}(A \to \eta; \gamma_k)) \chi_{x_k}(k)}{\sum_{\eta} \sum_{k=1}^{n} E_{\tilde{\phi}}(C^{x_k}(A \to \eta; \gamma_k)) \chi_{x_k}(k)}.$$

[2] The estimator of MFR's state at time k is $\hat{x}_k = \arg\max_i P(x_k = e_i | \gamma_{1:n})$,, and the output probabililty of γ_k could be computed efficiently by the Inside algorithm.

and the updating equation of the transition matrix is $a_{ji} = \frac{\sum_{k=1}^{n-1} \xi_{ji}(k)}{\sum_{k=1}^{n-1} \chi_j(k)}$. Iterative computations of the expectation and maximization steps above will produce a sequence of parameter estimates with monotonically nondecreasing likelihood. Due to space limitation, detailed numerical studies are not presented, however, in summary, the estimated state sequence has average error rate of 6.67%, and the estimated parameter values are sufficiently close to the true values.

5. CONCLUSION

In this paper, the multifunction radar is carefully studied and modeled as a Markov modulated SCFG. The threat evaluation is reduced to a state estimation problem, and a maximum likelihood estimator is derived to evaluate the threat. In addition, a Bayesian algorithm with EM is derived to infer the unknown model parameters.

6. REFERENCES

[1] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Artech House, 1999.

[2] R. G. Wiley, *Electronic Intelligence: The analysis of radar signals*, Artech House, 1993.

[3] N. A. Visnevski, F. A. Dilkes, S. Haykin, and V. Krishnamurthy, "Non-self-embedding context-free grammars for multi-function radar modeling - electronic warfare application," *International Radar Conference*, pp. 669–674, 2005.

[4] K. Lari and S. J. Young, "The estimation of stochastic context free grammars using the Inside-Outside algorithm," *Computer Speech and Language*, vol. 4, pp. 35–56, 1990.

[5] P. L. Bogler, *Radar Principles with Applications to Tracking Systems*, John Wiley & Sons, 1990.

[6] N. Chomskey, "On certain formal properties of grammars," *Information and Control*, vol. 2, pp. 137–167, 1959.

[7] A. Wang, V. Krishnamurthy, F. A. Dilkes, and N. A. Visnevski, "Threat estimation by electronic surveillance of multifunction radars: a stochastic context free grammar approach," *Conference on Decision and Control*, 2006.

[8] Z. Chi, "Statistical properties of probabilistic context-free grammars," *Computational Linguistics*, vol. 25, pp. 131–160, 1999.

[9] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of IEEE*, vol. 77, pp. 257–286, 1989.