Linguistic Radar Modeling for Electronic Support: Pulse Processing for Stochastic Regular Grammars

Fred A. Dilkes
Defence Research and Development Canada
3701 Carling Avenue, Ottawa ON, K1A 0Z4, CANADA
Email: Fred.Dilkes@drdc-rddc.gc.ca

***Abstract*— Some aspects of grammatically based signal processing are presented in the context of electronic surveillance for radar signals. The theory of formal languages and context-free phrase-structured grammars is reviewed. A theoretical framework for pulse signal processing is presented for agile radar systems whose pulse-to-pulse dynamics is described by a stochastic regular grammar. A recursive formula is developed for the approximate computation of probability density functions in the presence of stationary noise.**

I. Introduction

Electronic surveillance (ES) systems are passive sensors that detect, recognize and localize radar and radio emitters. Their ability to identify and radar types and behaviours makes them invaluable for tactical support including threat assessment, sensor cueing, counter-targeting, and situation awareness. These functions rely heavily on threat libraries that contain *a priori* signal information, known as electronic intelligence (ELINT).

Virtually every modern military radar system responds dynamically to one or more input stimuli. A stimulus can be either human intervention (such as the operator turning a control knob, or manually adjusting a range gate), or it can be an automatic response to target returns (such as the radar software switching amongst search, track and engagement modes, or adjusting the pulse pattern in order to mitigate blind ranges and speeds). Most of these are invisible to the ES system whose only means of accumulating information is through the front end of its receiver.

Given sufficient *a priori* intelligence, one may know that certain signal behaviours are associated with particular stimuli, while other sequences cannot occur at all. In some cases, the interpretation of radar behaviour may not as simple as making a direct observation of its current signal parameters since those parameters may be used for several functions. One may need to interpret the current behaviour of the system in the context of its recent history. This information may help to distinguish one radar system among others that use similar signal parameters or to estimate the immediacy of a potential threat.

The result is that increasing amounts of ELINT defy the standard representation in terms of stable classification patterns and parameterizations. Much of the known intelligence cannot be programmed into ES tactical systems, resulting in suboptimal performance. A modernized framework is required to represent of the dynamics of complex radar systems and perform the corresponding signal processing functions.

This contribution proposes to formulate the ES problem mathematically using a construction known as a "stochastic grammar". This is expected to accommodate nearly all of the required information associated with complex radar agility, and provide a natural mechanism for assigning probabilities to the dynamics. Moreover, a technique is presented here to account for the effects of signal degradation.

The techniques discussed in this paper fall under the umbrella of mathematical methods that study syntactic patterns [1]. The methods have been applied to civilian fields such as computer language parsing [2], human language recognition [3], speech recognition [4], and genetic sequencing [5]. As a result, the formalism has the virtue of being associated with a large quantity of mathematical literature.

This paper is organized as follows. Section II provides a brief review of deterministic and stochastic grammatical languages and establishes notation. Section III presents a mathematical framework for radar pulses based on stochastic grammars. In Section IV, the framework is used to develop a dynamic approach to ES signal processing.

II. Languages and Grammars

This section contains a review of some elements of the theory of stochastic languages and grammars. A more complete and rigorous treatment of deterministic grammars can be found in [2] while stochastic grammars are discussed in [1], [5]–[7].

A. Deterministic formal languages

The framework begins with a finite set of symbols known as the "vocabulary", denoted by $\mathcal{T}$. A "string" is any finite sequence of symbols in the vocabulary. The set of all strings over a set of symbols $\mathcal{T}$ is denoted by $\mathcal{T}^*$. The number of symbols contained in the string α is called the "length" of the string, and is denoted by $|\alpha|$. For example, a sequence $\alpha = (a_1 a_2 \cdots a_L)$ is a string in $\mathcal{T}^*$ of length $|\alpha| = L$, as long as each component a_m, $m = 1 \ldots L$ is an element of $\mathcal{T}$. There is a unique "empty string" denoted by $\varepsilon \in \mathcal{T}^*$ whose length vanishes, $|\varepsilon| = 0$.

A "deterministic formal language" $\mathcal{L} \subseteq \mathcal{T}^*$ is some particular set of strings over a specified vocabulary $\mathcal{T}$. This notion is very broad, and potentially very complex as the language may contain an infinite number of different strings. Nevertheless, it is often possible to describe languages of practical interest in terms of a finite set of rules.

As a simple example, consider the language over the vocabulary $\mathcal{T} = \{a, b, c\}$ consisting of all finite repetitions of the sequence $bcac$, including the null string:

$$\begin{aligned} \mathcal{L} &= \{\varepsilon, bcac, bcacbcac, \ldots\} \\ &= \{(bcac)^n | n = 0, 1, 2, 3, \ldots\} . \end{aligned} \quad (1)$$

This language is quite repetitive and may represent a signal with a degree of stability. The individual strings do not contain much information and it is trivial to determine whether or not a string is in this language.

B. Deterministic context-free grammars

For the ES application, the language would represent all possible combination of sequences that a radar could ever execute, from power-up to shutdown. This set is effectively infinite even for the simplest radar system. For electronically agile radar systems, the language can be quite sophisticated and does not have a straightforward description like that shown in (1). A general mechanism is required to define the language associated with complex radar systems.

Many formal languages of interest can be represented using a constructive mathematical formalism originally developed by Chomsky [3] known as a "phase-structured grammar". A complete treatment of this construction can be found in [2]. Here, it suffices to restrict attention to a particular subclass known as a "context-free grammar" (CFG), defined to be a 4-tuple $\mathcal{G} = (\mathcal{T}, \mathcal{N}, \mathcal{R}, S)$ with the following components. The vocabulary $\mathcal{T}$ is the finite set of symbols described in Subsection II-A. The set $\mathcal{N}$ is an additional collection of symbols. Symbols in $\mathcal{T}$ are called "terminals" and symbols in $\mathcal{N}$ are known as "nonterminals"; the two sets are disjoint $\mathcal{T} \cap \mathcal{N} = \emptyset$ so that a symbol can be a terminal or a nonterminal, but not both. One of the nonterminals, called the "starting symbol", has a distinguished status and is denoted by $S \in \mathcal{N}$.

The final component of the grammar is the set of "replacement rules" $\mathcal{R}$. Each element of $\mathcal{R}$ has the form $A \to \Omega$ where $A \in \mathcal{N}$ and $\Omega \in (\mathcal{T} \cup \mathcal{N})^*$. The set $(\mathcal{T} \cup \mathcal{N})^*$ consists of all finite sequences of symbols drawn from both $\mathcal{T}$ and $\mathcal{N}$.

To illustrate the connection between grammars and formal languages, consider an example grammar $\mathcal{G}$ prescribed by

$$\mathcal{G} = \left(\begin{array}{ll} \mathcal{T} = \{a, b, c\} \,, & \mathcal{N} = \{S, D\} \,, \\ \mathcal{R} = \{S \to DS \,, S \to \varepsilon, D \to bcac\} \,, & S \end{array} \right) . \quad (2)$$

This grammar is a generative model for the language (1). It is possible, for example, to derive the example string $bcacbcacbcac$ from the starting symbol S by invoking a sequence of replacements represented by the graph in Figure 1. Such a graph is known as a "parse tree". At each branch, one of the nonterminals is replaced using one of the rules in $\mathcal{R}$. The figure shows three instances of the replacement $S \to DS$, three of $D \to bcac$ and one of $S \to \varepsilon$.

If a specified CFG admits a parse tree whose root node is $A \in \mathcal{N}$ and whose yield is $\Omega \in (\mathcal{T} \cup \mathcal{N})^*$ then one says that "Ω can be derived from A" or simply $A \stackrel{*}{\Rightarrow} \Omega$. In this notation, Figure 1 implies that

$$S \stackrel{*}{\Rightarrow} bcacbcacbcac \,.$$

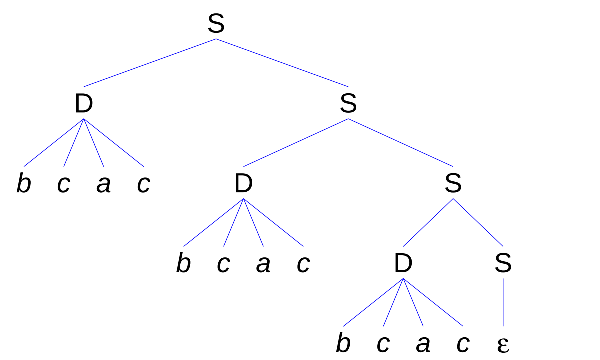


Fig. 1. A parse tree for the derivation $S \stackrel{*}{\Rightarrow} bcacbcacbcac$.

.

The language $\mathcal{L}(\mathcal{G})$ described by a grammar $\mathcal{G}$ is the set of strings of terminals that can be derived from S.

A very special subclass of context-free grammars is the case of "regular grammars" (RGs). For present purposes,[1] a regular grammar is defined to be a CFG in which each production rule in $\mathcal{R}$ is either of the form $A \to aB$ or $A \to a$ where $A, B \in \mathcal{N}$ and $a \in \mathcal{T}$. Every RG is a CFG but the converse is not true.

C. Stochastic languages and grammars

A language is not expected to be a perfect or deterministic model for tactical observations of a radar signal since the formalism does not account for unknown input stimuli or uncertainties associated with the observation process. Sources of signal degradation include traditional problems such as interference, de-interleaving errors or simple failures to make some observations. The management and quantification of uncertainties is crucial for the proper functioning of an ES system. This can be done by introducing an element of stochasticity to the formulation.

A "stochastic language" over a vocabulary $\mathcal{T}$ is an assignment of a probability distribution to the set of strings $\mathcal{T}^*$. Each string $\alpha \in \mathcal{T}^*$ carries a non-negative probability $P(\alpha) \geq 0$ so that $\sum_{\alpha \in \mathcal{T}^*} P(\alpha) = 1$.

In the case of a language generated by a CFG (or RG), the natural way to assign the probabilities is to replace the production rules $\mathcal{R}$ with an assignment of "production probabilities" $\mathcal{P}$. To every possible rule of the form $A \to \Omega$, in which $A \in \mathcal{N}$ and $\Omega \in (\mathcal{T} \cup \mathcal{N})^*$, a production probability $\mathcal{P}(A \to \Omega) \in [0, 1]$ is assigned. These must satisfy

$$\sum_{\Omega \in (\mathcal{T} \cup \mathcal{N})^*} \mathcal{P}(A \to \Omega) = 1 \quad \text{for} \quad \forall A \in \mathcal{N} \,.$$

The result $\mathcal{G}_{\mathcal{P}} = \{\mathcal{T}, \mathcal{N}, \mathcal{P}, S\}$ is known as a "stochastic context-free grammar" (SCFG). It is assumed that $\mathcal{P}$ assigns positive probabilities only to a finite set of rules $\mathcal{P}^{-1}((0, 1])$. If $\mathcal{P}^{-1}((0, 1]) \subseteq \mathcal{R}$ where $\mathcal{R}$ is the set of production rules of some deterministic grammar $\mathcal{G} = \{\mathcal{T}, \mathcal{N}, \mathcal{R}, S\}$, then $\mathcal{G}$ is known as a "characteristic grammar" for $\mathcal{G}_{\mathcal{P}}$.

[1]This is actually a specialized definition of a RG. Such a grammar might more properly be described as a "right-linear RG in normal form".

In the example where $\mathcal{G}$ given by (2), we may construct a stochastic grammar by prescribing that the non-zero production probabilities are $\mathcal{P}(S \to DS) = 0.6$, $\mathcal{P}(S \to \varepsilon) = 0.4$ and $\mathcal{P}(D \to bcac) = 1$. Consequently, $\mathcal{G}$ is a characteristic grammar for $\mathcal{G}_{\mathcal{P}} = \{\mathcal{T}, \mathcal{N}, \mathcal{P}, S\}$. Since Figure 1 is the only parse tree to derive $bcacbcacbcac$ from S, a "derivation probability" for this string can be computed by multiplying the relevant production probabilities:

$$\mathcal{P}(S \overset{*}{\Rightarrow} bcacbcacbcac) = (0.6)^3(1)^3(0.4)^1 = 0.0864\,.$$

If $\mathcal{G}_{\mathcal{P}}$ has a characteristic grammar that is regular, then $\mathcal{G}_{\mathcal{P}}$ is known as a "stochastic regular grammar" SRG and is closely related to a "hidden Markov model" [4]. The latter has also been suggested for exploitation in radar ES [8].

III. Grammatical Radar-ES Pulse Model

For this presentation, a "radar-ES model" refers to any stochastic description of the arrival times $T^{(\mathrm{Rx})}$ of pulses received from a radar system by an ES system. This simplification of the real-world situation does not include such important parameters as pulse-duration, radio frequency, and pulse modulation. The rationale behind emphasizing the arrival times stems from the fact that the resulting signal processing problem involves synchronization and detection issues that are not as inherently associated with the other parameters. It is hoped that a robust processing methodology associated exclusively with these times will lead to naturally to extensions in other parametric domains.

Although the application of stochastic grammars is well motivated, it remains to formulate the practical aspects of ES in such a way as to invoke this formalism. The approach advocated here is not unique and other potential formulations are also possible [8].

For the class of models described here, pulse trains shall be described by associating the intervals between sequential pulses, known as "pulse-to-pulse intervals" (PPIs), with symbols in a vocabulary. In turn, a grammar will describe the arrangements of these symbols, and consequently arrangements of the intervals themselves. Superposed on this mechanism is a stationary random noise process describing signal degradation.

Specifically, the ES-radar model under consideration $\mathcal{M} = (\mathcal{G}_{\mathcal{P}}, \mathcal{F}, P_{\mathrm{det}}, \rho_{\mathrm{spur}})$ consists of four components:

- a stochastic regular grammar $\mathcal{G}_{\mathcal{P}} = (\mathcal{T}, \mathcal{N}, \mathcal{P}, S)$,
- a set of probability density functions $\mathcal{F} = \{f_a(t) | t \geq 0, a \in \mathcal{T}\}$, that specify distributions of any PPI associated with each terminal symbol,
- a pulse detection probability $P_{\mathrm{det}} \in [0, 1]$, and
- a spurious pulse Poisson rate $\rho_{\mathrm{spur}} \geq 0$.

The dynamics for this model are prescribed as follows:

Suppose that there is a discrete random string $\alpha = (a_1 a_2 \cdots a_L) \in \mathcal{T}^*$ whose marginal distribution $P(\alpha)$ is determined by the stochastic grammar $\mathcal{G}_{\mathcal{P}}$. There is no restriction on the length $L = |\alpha|$.

Next, suppose that there is a random set of $L+1$ "transmitted" pulses represented by their leading edge arrival times $T^{(\mathrm{Tx})} = \{\tilde{t}_0, \tilde{t}_1, \ldots, \tilde{t}_L\}$. The PPI between each sequential pair of transmitted pulses $\tilde{\delta}_l \equiv \tilde{t}_l - \tilde{t}_{l-1}$ is positive for $l = 1, \ldots, L$. The statistics of $\tilde{\delta}_l$ are completely specified by the corresponding symbol a_l through the conditional probability density function $p(\tilde{\delta}_l | a_l) = f_{a_l}(\tilde{\delta}_l)$, and are otherwise independent of every other random variable. In this way $\mathcal{G}_{\mathcal{P}}$ and $\mathcal{F}$ are sufficient to define the statistics for the equivalence class of $T^{(\mathrm{Tx})}$, modulo an uninteresting overall time translation.

The transmitted pulse set $T^{(\mathrm{Tx})}$ is randomly mapped to a "detected" pulse set $T^{(\mathrm{det})}$ and a "missing" pulse set $T^{(\mathrm{miss})}$ by a stochastic binary map. Each pulse in $T^{(\mathrm{Tx})}$ independently has a probability P_{det} of appearing in the set $T^{(\mathrm{det})}$; otherwise, it appears in the set $T^{(\mathrm{miss})}$. As a result, we have $T^{(\mathrm{Tx})} = T^{(\mathrm{det})} \cup T^{(\mathrm{miss})}$ and $T^{(\mathrm{det})} \cap T^{(\mathrm{miss})} = \emptyset$.

In addition, a finite random set of "spurious" pulse times $T^{(\mathrm{spur})}$ is independently selected from a Poisson process whose average rate is ρ_{spur} pulses per time unit over the finite time interval $[\tilde{t}_0, \tilde{t}_L]$.

A pulse is defined to be "received" if it is either detected or spurious. The set of all such pulses is denoted by $T^{(\mathrm{Rx})} \equiv T^{(\mathrm{det})} \cup T^{(\mathrm{spur})}$.

IV. Probabilistic Signal Processing

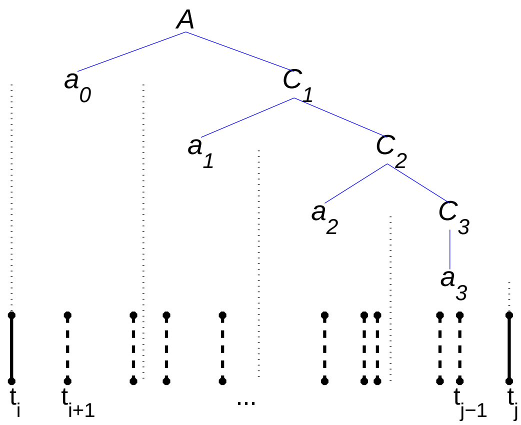
The ES system is required to make probabilistic estimations and decisions based entirely on the received pulse set $T^{(\mathrm{Rx})}$ and an *a priori* ES-radar model $\mathcal{M}$. It has no additional information about missing pulses no direct mechanism for distinguishing detected pulses from spurious pulses.

The usual parsing techniques for SRGs are the forward algorithm [4], for computing an overall probability of generating a particular string, and the Viterbi algorithm [4], [9], for determining the most likely parse tree for the string. Generalizations to other SCFGs can be found in [6], [7]. All of these algorithms are dynamic programming techniques based on recursive principles and apply naturally to discrete time sequences.

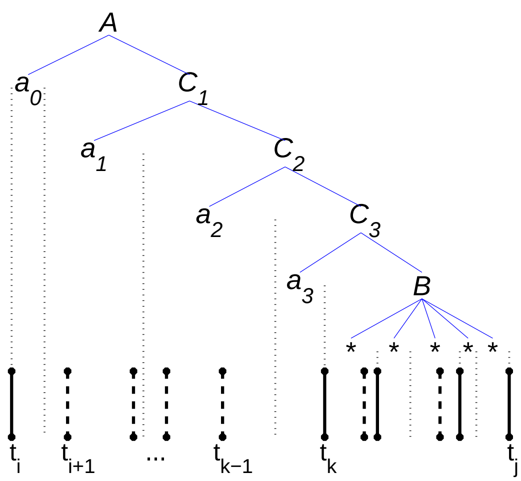
On the other hand, very different techniques, such as the one presented in [10], have been developed for analyzing noisy agile sequences that are not governed by stochastic dynamical models.

In the present case, dynamically generated pulse sequences are observed after a degree of degradation. Spurious pulses have the effect of splitting a single pulse-to-pulse interval into two or more intervals while missing pulses have the effect of combining two adjacent intervals into one larger interval. In either cases, the result is a sequence of time intervals whose connection with the original model is hidden. As a result, the identification of the symbols from the pulse train requires an additional level of processing.

In the remainder of this section, a signal processing formalism is presented that integrates features of dynamic programming with the pulse-level signal processing. The set of received pulses $T^{(\mathrm{Rx})} = \{t_0, t_1, \ldots, t_M\}$ is assumed to have been generated by an ES-radar model $\mathcal{M}$ and arranged in monotonically increasing order so that $t_0 < t_1 < \cdots < t_M$.



(a) Non-recursive pulse parse tree.



(b) Recursive pulse parse tree.

Fig. 2. Parse trees for pulses $\{t_i, \dots t_j\}$. Transmitted pulses are indicated by dotted lines. Each interval between a pair of successive transmitted pulses is associated with a terminal symbol. Detected pulses are indicated by solid vertical line segments. Spurious pulses are indicated by dashed vertical line segments. Illustration (a) shows $L = 3$ missing pulses and $j - i - 1 = 9$ spurious pulses in the interval between t_i and t_j. By contrast, (b) shows a detected pulse at t_k; there are $k - i - 1 = 4$ spurious pulses and $L = 3$ missing pulses in the interval between t_i and t_k.

A. Parse trees for pulses

The techniques proposed in this paper are based on parse trees defined over sets of pulses. A parse tree over some subset of received pulses $T_{ij}^{(\mathrm{Rx})} \equiv \{t_i, t_{i+1}, \dots, t_j\}$ is characterized by three defining properties:

1) a partitioning of the pulse set $T_{ij}^{(\mathrm{Rx})}$ into the union of two disjoint sets $\hat{T}_{ij}^{(\mathrm{det})}$ and $\hat{T}_{ij}^{(\mathrm{spur})}$ representing estimates of $T^{(\mathrm{det})} \cap T_{ij}^{(\mathrm{Rx})}$ and $T^{(\mathrm{spur})} \cap T_{ij}^{(\mathrm{Rx})}$ respectively,
2) an assignment of finite terminal symbol strings to the pulse-to-pulse intervals in $\hat{T}^{(\mathrm{det})}$, and
3) a standard grammatical parse tree for the terminal symbols in terms of nonterminals.

Two such parse trees are illustrated in Figure 2.

In Figure 2(a), t_i and t_j are hypothesized to be the only detected pulses in $T_{ij}^{(\mathrm{Rx})}$ so that $\hat{T}_{ij}^{(\mathrm{det})} = \{t_i, t_j\}$. The remaining $j-i-1$ internal pulses are interpreted as spurious so that $\hat{T}_{ij}^{(\mathrm{spur})} = \{t_{i+1}, \dots, t_{j-1}\}$. In addition, the time interval $[t_i, t_j]$ may contain a number of "missing" pulses from $T^{(\mathrm{miss})}$. The interval in $\hat{T}_{ij}^{(\mathrm{det})}$ is associated with the terminal string $(a_0 a_1 \cdots a_L)$ whose parse tree is represented by the sequence of nonterminal nodes $(AC_1C_2 \cdots C_L)$ where L is the number of missing pulses.

The parse tree in Figure 2(b) illustrates a scenario in which t_i, t_j and at least one of the internal pulses $\{t_{i+1}, \dots, t_{j-1}\}$ are assumed to be detected. The first detected pulse appearing after t_i is denoted by t_k for some $k = i+1, \dots, j-1$. If t_k is not the first received pulse after t_i then the intervening pulses must be spurious $\{t_{i+1}, \dots, t_{k-1}\} \subseteq \hat{T}_{ij}^{(\mathrm{spur})}$. The interval from t_i to t_k is associated with some terminal string $(a_0 a_1 \cdots a_L)$ generated by some sequence of nonterminal nodes $(AC_1C_2 \cdots C_L)$ where L is the number of missing pulses between t_i and t_k. The intervals amongst the final subset of pulses $\{t_k, \dots, t_j\}$ are abstractly shown to have been generated by some nonterminal node B.

B. Probability computation

A theoretical method is presented here to compute the probability that the model presented in Section III will generate a particular sequence of measured arrival times $T^{(\mathrm{Rx})}$.

Probability measures can be defined more rigorously in terms of sequences of pulse-to-pulse intervals, rather than in terms of the sets of arrival times. The set of all non-empty sequences of positive pulse-to-pulse intervals is denoted by $\bigcup_{N=1}^{\infty} \mathbb{R}^{(+)N}$. Elements of this set are generically denoted by $\Delta = (\delta_1, \delta_2, \dots \delta_N)$ for any positive integer $N > 0$.

For each nonterminal symbol $A \in \mathcal{N}$, we define a probability measure for the sequences of intervals generated by A. For each sequence length $N > 0$, the measure over a differential region of $\mathbb{R}^{(+)N}$ is defined by

$$\begin{aligned} \mathrm{d}\mu_\mathrm{A} &= \mu_A(\delta_1, \delta_2, \dots \delta_N)\, \mathrm{d}\delta_1\, \mathrm{d}\delta_2 \dots \mathrm{d}\delta_\mathrm{N} \\ &= P\big(A \Rrightarrow [\delta_1, \delta_1 + \mathrm{d}\delta_1] \times [\delta_2, \delta_2 + \mathrm{d}\delta_2] \times \cdots \\ &\qquad \times [\delta_N, \delta_N + \mathrm{d}\delta_\mathrm{N}]\big)\,. \end{aligned}$$

Here, $P(A \Rrightarrow \dots)$ indicates a conditional probability that A generates some string of terminals that, along with the stationary noise process, leads to the given sequence of PPIs. This probability is conditioned on the requirements that the first and last pulses of the transmitted sequence associated with A are detected, and correspond with the first and last pulses of the received pulse subset.

In general, the probability density functions $\mu_A(\Delta)$ are non-trivial to evaluate. We propose to propagate probabilities in a recursive fashion, similar to the forward and backward variables of hidden Markov model processing [4] or the inside variables [6] for SCFGs.

Let Δ_{ij} be the sequence of PPIs corresponding to the subset $\{t_i, t_{i+1}, \dots, t_j\}$. A probabilistic construction for the variables $\mu_A(\Delta_{ij})$ can be understood by referring to the two generic types of parse trees shown in Figure 2. The exact expression for $\mu_A(\Delta_{ij})$ is accomplished by summing over every tree that leads to a sequence of the appropriate length. The result is

$$\mu_A(\Delta_{ij}) = \lambda_A(t_j - t_i) e^{-\rho_{\text{spur}}(t_j - t_i)} \rho_{\text{spur}}^{j-i-1} + \sum_{k=i+1}^{j-1} \sum_{B \in \mathcal{N}} \nu_{AB}(t_k - t_i) e^{-\rho_{\text{spur}}(t_k - t_i)} \rho_{\text{spur}}^{k-i-1} \times P_{\text{det}} \mu_B(\Delta_{kj}) \tag{3}$$

where $\lambda_A(t)$ and $\nu_{AB}(t)$ are functions described below.

The first term of (3) accounts for Figure 2(a). The expression $\lambda_A(t_j - t_i)$ results from enumerating all parse trees in which a nonterminal $A \in \mathcal{N}$ generates a string $\alpha \in \mathcal{T}^*$; this string, in turn, is associated with the duration of the entire interval $t_j - t_i$. Moreover, it accounts for the $|\alpha| - 1$ internal pulses that go undetected. The result function λ_A is defined by

$$\lambda_A(t) \equiv \sum_{\alpha \in \mathcal{T}^*} (1 - P_{\text{det}})^{|\alpha|-1} \mathcal{P}(A \stackrel{*}{\Rightarrow} \alpha) f_\alpha^*(t) \tag{4a}$$

where

$$f^*_{(a_1 a_2 \cdots a_L)} \equiv f_{a_1} * f_{a_2} * \ldots * f_{a_L}$$

is a convolution defined for any $(a_1 a_2 \cdots a_L) \in \mathcal{T}^*$.

The second term of (3) accounts for Figure 2(b). The function $\nu_{AB}(t)$ represents all parse trees in which a nonterminal $A \in \mathcal{N}$ generates a string of the form αB where $\alpha \in \mathcal{T}^*$ and $B \in \mathcal{N}$, resulting in

$$\nu_{AB}(t) \equiv \sum_{\alpha \in \mathcal{T}^*} (1 - P_{\text{det}})^{|\alpha|-1} \mathcal{P}(A \stackrel{*}{\Rightarrow} \alpha B) f_\alpha^*(t) \, . \tag{4b}$$

The nonterminal symbol B in Figure 2(b) leads to the recursive appearance of $\mu_B(\Delta_{kj})$ in (3).

The examples of parse trees for pulses described in Figure 2 and the corresponding definition of $\mathrm{d}\mu_{\mathrm{A}}$ excludes the possibility that the first and last transmitted pulses associated with the intervals generated by A may be missing. It is possible to remove the condition that the last pulse associated with A be detected by replacing (4a) with

$$\lambda_A(t) \equiv \sum_{\alpha, \beta \in \mathcal{T}^*} (1 - P_{\text{det}})^{|\alpha|-1+|\beta|} \mathcal{P}(A \stackrel{*}{\Rightarrow} \alpha\beta) f_\alpha^*(t) \, .$$

Here, the string β is associated with a sequence of intervals that occur after the final detected pulse. Similarly, the condition that the first pulse associated with A be detected can be removed by defining a new measure $\mathrm{d}\mu'_{\mathrm{A}}$ whose probability density functions are given by

$$\mu'_A(\Delta_{ij}) \equiv \sum_{\beta \in \mathcal{T}^*} (1 - P_{\text{det}})^{|\beta|} \times \sum_{B \in \mathcal{N}} \mathcal{P}(A \stackrel{*}{\Rightarrow} \beta B) \mu_B(\Delta_{ij}) \, .$$

In this case, β represents a set of intervals that occur before the first detected pulse. Note that the term in the summation corresponding to $\beta = \varepsilon$ involves a "unit" derivation whose probability is defined to be $\mathcal{P}(A \stackrel{*}{\Rightarrow} B) \equiv \delta_{AB} \equiv \{1 \text{ if } A = B; 0 \text{ if } A \neq B\}$.

An algorithm that exploits this formalism, and its complexity is under investigation. The most significant barrier to the implementation of these formulas is the appearance of infinite summations over $\alpha \in \mathcal{T}^*$ in (4a) and (4b). Fortunately, this problem is tempered by factors such as $f_\alpha^*(t)$ and $(1 - P_{\text{det}})^{|\alpha|-1}$ so that only a few strings α contribute significantly. The forward algorithm provides an efficient mechanism for evaluating the probabilities $\mathcal{P}(A \stackrel{*}{\Rightarrow} \alpha B)$ and $\mathcal{P}(A \stackrel{*}{\Rightarrow} \alpha)$; wherever possible, these should be computed offline.

This section is concluded by noting that $\mu_A(\Delta_{ij})$ (or $\mu'_A(\Delta_{ij})$) represents only an overall probability measure of generating the given sequence. A second problem of interest is the computation of the most likely pulse parse tree for Δ_{ij} and its individual probability. These two problems are related in the same way that the forward algorithm is related to Viterbi: one can simply replace summations with maxima and argument maxima [4].

V. Conclusion

The parsing of pulse sequences using stochastic grammatical models is a potentially powerful but unsolved problem for electronic support.

This paper contributes a recursive framework for computation of the overall probability for the reception of a pulse set from an ES-radar model in which pulse-to-pulse agility is described by a regular stochastic grammar and the noise is stationary. A sub-optimal implementation of this framework, and possible generalizations to stochastic context-free grammars, are areas for future exploration.

Acknowledgment

The author would like to offer thanks to Nikita Visnevski who suggested the application of stochastic grammars to ES.

References

[1] K. S. Fu, *Syntactic Methods in Pattern Recognition*. New York, NY: Academic Press, 1974.

[2] J. E. Hopcroft and J. D. Ullman, *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, 1979.

[3] N. Chomsky, "Three models for the description of language," *IRE Transactions on Information Theory*, vol. IT-2, no. 3, pp. 113–124, 1956.

[4] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*. New Jersey: Prentice Hall, 1993.

[5] R. Durbin, S. Eddy, A. Krogh, and G. Mitchisen, *Biological sequence analysis: Probabilistic models of proteins and nucleic acids*. Cambridge, UK: Cambridge University Press, 1998.

[6] 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.

[7] F. Jelinek, J. D. Lafferty, and R. L. Mercer, "Basic methods of probabilistic context free grammars," in *Speech Recognition and Understanding*, P. Laface and R. DeMori, Eds. Berlin: Springer, 1992, pp. 345–360.

[8] P. Lavoie, "Hidden markov modeling for radar electronic warfare," U.S. patent application 20 030 085 831, May 8, 2003.

[9] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactios on Information Theory*, vol. IT-32, no. 2, pp. 260–269, 1967.

[10] S. D. Elton, "A cross-correlation technique for radar emitter recognition using pulse time of arrival information," in *Proceedings of 2001 Workshop on Defence Applications of Signal Processing*, Borossa Valley, South Australia, 2001, pp. 19–23.