Hierarchical classification of dynamically varying radar pulse repetition interval modulation patterns

Jukka-Pekka Kauppi a,*, Kalle Martikainen b, Ulla Ruotsalainen a

a Tampere University of Technology, Department of Signal Processing, P.O. Box 553, FI-33101 Tampere, Finland
b Patria Aviation Oy, Naulakatu 3, FI-33100 Tampere, Finland

ARTICLE INFO

Article history:
Received 1 October 2008
Received in revised form 13 April 2010
Accepted 24 June 2010

Keywords:
Radar emitter classification
Pulse train
Scheduled PRI modulation
Multifunction radar
Feature extraction
Multi-layer perceptron
Frequency agility

ABSTRACT

The central purpose of passive signal intercept receivers is to perform automatic categorization of unknown radar signals. Currently, there is an urgent need to develop intelligent classification algorithms for these devices due to emerging complexity of radar waveforms. Especially multifunction radars (MFRs) capable of performing several simultaneous tasks by utilizing complex, dynamically varying scheduled waveforms are a major challenge for automatic pattern classification systems. To assist recognition of complex radar emissions in modern intercept receivers, we have developed a novel method to recognize dynamically varying pulse repetition interval (PRI) modulation patterns emitted by MFRs. We use robust feature extraction and classifier design techniques to assist recognition in unpredictable real-world signal environments. We classify received pulse trains hierarchically which allows unambiguous detection of the subpatterns using a sliding window. Accuracy, robustness and reliability of the technique are demonstrated with extensive simulations using both static and dynamically varying PRI modulation patterns.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction

Modern passive signal intercept receiver devices are highly sophisticated digital sensors that are designed to intercept and classify surrounding radar emissions over a wide radio frequency (RF) spectrum. Potential applications requiring categorization of unknown radar signals include maritime barrier operations designed to prevent illegal immigration, weapon and drug smuggling, illegal fishing and piracy (Menhinick, 2005). During military crisis, classification is needed, for example, to alert defensive systems in the presence of threat emissions (Schleher, 1999). Fig. 1 presents the main stages of radar emitter categorization in a passive signal intercept receiver. After interception, incoming radar pulses are parameterized in a pulse parameter encoder, which produces pulse descriptor words (PDWs) describing parameters of single radar pulses, such as RF, pulse amplitude (PA), pulse width (PW), time-of-arrival (TOA), direction-of-arrival (DOA), pulse rise/fall times, and intrapulse modulation type. Because there can exist several simultaneously active emitters in the complex signal environment, an interleaved sequence of pulses originating from several emitters is typically received. Interleaved pulses are sorted in a deinterleaver, which clusters together those PDWs originating from the same emitters (Anderson, Gately, Penz, & Collins, 1990; Ata’a & Abdullah, 2007; Granger, Rubin, Grossberg, & Lavoie, 2001; Liu, Lee, Li, Luo, & Wong, 2005; Mardia, 1989). A sequence of PDWs of a single emitter is then processed in an interpulse modulation encoder, which recognizes and estimates parameters of different pulse-to-pulse parameter variation patterns of the sequence. Especially important information can be obtained by recognizing a pulse repetition interval (PRI) modulation pattern of a signal based on the difference of successive TOA values of a sequence. Also RF and PW agility patterns as well as antenna beam pattern may be recognized in order to gain additional knowledge of the emitter type or its functional purpose. The interpulse modulation encoder produces files called emitter descriptor words (EDWs), describing the most relevant intra- and interpulse radar signal parameters as accurately as possible. Finally, EDWs are matched against actively collected threat libraries to identify the emitters. If the emitter cannot be identified, its functional purpose and threat level is assessed solely based on the extracted EDW parameters.

To enable reliable functioning in complex signal environments with multiple emitters, modern intercept receivers must be capable of processing unknown, corrupted and ambiguous measurements in a robust and reliable manner (Roe, Cussons, & Feltham, 1990). In recent years, the classification problem has become even more challenging due to advent of multifunction radars (MFRs; also called electronically scanned radars). These radars perform several simultaneous tasks such as search and multi-target tracking by emitting highly diverse and unpredictable scan patterns that
are simultaneously agile in RF, PW and PRI (Arasaratnam, Haykin, Kirubarajan, & Dilkes, 2006; Visnevski, Krishnamurthy, Wang, & Haykin, 2007; Wirth, 2001). Because emission characteristics of radars are not known a priori (from the standpoint of passive signal interception) in many cases, the only possibility of revealing intentions of radars is to automatically analyze their waveform characteristics. Especially recognition of radar interpulse modulation (e.g. PRI, RF and PW modulation) has become the central issue of intercept receiver signal processing, because it provides invaluable information about the functional purpose of the radar. Since many existing methods have been designed to cope only with traditional radars having static emission characteristics, there is an increasing need to develop new interpulse recognition algorithms in modern intercept receivers. We improve the situation by presenting a novel technique that can be applied to reveal complex PRI modulation patterns emitted by MFRs. The core of our technique is an invariant feature set that is capable of discriminating all basic PRI modulation types in a feature space (see Kauppi and Martikainen (2007)).

In this paper, we present a classification scheme that utilizes the features to recognize PRI modulation patterns in hierarchical manner: PRI modulation substructures are first detected in an entire sequence based on coarse category information using a sliding window, and the extracted subpatterns are then classified more accurately using second-level classifiers. The hierarchical classification scheme is required to unambiguously capture different PRI modulation substructures in an entire sequence. Within this framework, we implement an exemplary classifier that can automatically handle corrupted or other unexpected sequences. It is extremely important to avoid misclassifying such sequences in practice—consequences of erroneous emitter classification can be especially severe if an emission intended for hostile activity is misclassified as harmless. A much better option in this situation would be to warn a defensive platform about the unknown emission.

The rest of the paper is organized as follows. In Section 2, we first describe elementary PRI modulation categories (Section 2.1) and then review and discuss the limitations of existing PRI modulation recognition techniques (Section 2.2). In Section 3, we introduce a new method for PRI modulation recognition which consists of two major entities: feature extraction (Section 3.1) and hierarchical classification (Section 3.2). Thereafter, we present an exemplary classifier design by utilizing an extensive Monte Carlo simulated signal environment (Section 4), provide both visual and quantitative recognition results of the diverse simulations (Section 5), and finally summarize our study (Section 6).

2. PRI modulation recognition

2.1. PRI modulation categories

In general, PRI modulation can be described as follows:

\[ F(k) = t_{k+1} - t_k = x_k, \quad k = 1, 2, \ldots, N - 1, \]

where \( t_k \) is TOA of the \( k \)th pulse, \( x_k \) denotes \( k \)th pulse interval, also called TOA difference (dTOA), and \( N \) is a total number of pulses in a sequence. Depending on an emitted sequence, the function \( F \) can have different forms. We use terms pulse train and PRI sequence interchangeably to denote the sequence of pulses originating from a single emitter (although spurious pulses originating from other emitters can exist in the sequence).

Depending on the task and intelligence of the radar, one pulse train may contain several successive PRI modulation subpatterns with unique characteristics. In the following, we describe the most common PRI modulation categories according to Wiley (2006, 2007).

Basically, different PRI modulation types can be divided in six categories named Constant, Stagger, Jittered, Sliding, Dwell and Switch (we use a shorthand notation Dwell & Switch), and Periodic PRI. Although detailed discussion is beyond the scope of this paper, it is important to note that, depending on parameter settings, each PRI modulation category can be associated with one or more radar functions. If pulse interval variation is less than 1% of the mean PRI, it can be categorized as Constant PRI because such variation hardly serves any useful purpose. If radar switches pulse intervals on a pulse-to-pulse basis in periodic manner, resulting variation is called PRI Stagger. The number of positions of Stagger means the total number of pulse intervals that make up one period. It may be possible that the same pulse interval value appears more than once in a period. For example, periodic repetition of two short equal pulse intervals followed by one long pulse interval corresponds to PRI Stagger with three positions (i.e., the period length is three), but the sequence contains only two different pulse interval values. The number of Stagger positions of modern radars can vary from two even up to 64.

Jittered PRI means intentional random pulse interval variation up to about 30% of the mean PRI (Wiley, 2006). Variation is typically Gaussian, uniform, non-uniform or discrete. For discrete Jittered PRI, approximately up to 64 distinct pulse interval values can be expected. Sliding PRI means either monotonic increase or decrease of successive pulse intervals. Sliding is typically periodic, when there is a rapid switch to another extreme limit when another is reached. Very large variations may be expected in some applications, but also more limited variations are possible. Dwell & Switch PRI consists of two or more stages of extremely stable constant PRI bursts which may or may not be repeated periodically. The number of pulse intervals in a single burst can vary remarkably between radars and radar modes, and the number of bursts in one period may vary from 2 to 16 or even more. Periodic PRI means sinusoidal or possibly triangular pulse interval variation. Slight variations less than 5% of the mean PRI with a frequency of about 50 Hz are expected in missile guidance applications, but larger variations may be possible if Periodic PRI is utilized for some other purposes.

As conventional radars typically transmit a long sequence of pulses associated with one of the presented categories, modern MFR can transmit a sequence that consists of numerous sequential modulation patterns called Scheduled PRIs (Wiley, 2006). Because MFRs schedule different tasks, such as search, acquisition, multiple target tracking, and missile guidance in adaptive manner (Wirth, 2001), the exact number and length of different modulation

---

Fig. 1. The main stages of radar emitter categorization in a passive signal intercept receiver.
patterns is not known in advance. For example, the number of transmitted patterns varies depending on the number of targets being tracked and their locations. During one beam time, however, transmitted PRI sequence is usually the same or very similar to one of the basic six PRI modulation types.

2.2. Existing techniques

Broadly speaking, existing PRI modulation-type recognition techniques can be divided into two categories: those that utilize PRI histograms and those that exploit sequential pulse interval information to discriminate different PRI modulation types from each other. PRI histogram analysis is a common technique to support traditional human-operated PRI modulation-type recognition. Unfortunately, histograms are unsuitable for the automatic recognition of some PRI modulation types such as Periodic and Sliding PRI, because for these modulation types, the shape of a histogram varies depending on the number of available pulses, the choice of histogram bin width, and the amount of imperfections present in an intercepted pulse train. However, histograms serve more useful purpose for the recognition of those PRI modulation types involving distinct pulse interval values, such as Stagger and Constant PRI. A histogram-based technique proposed in Mardia (1989) is mainly designed for pulse deinterleaving, but is also used for the recognition of Constant and Stagger PRIs. The technique is based on histogram peak detection in so-called cumulative difference (CDIF) histograms which present frequencies of the occurrence of selected pulse pairs of a sequence. If any peak in the CDIF-histogram exceeds a predetermined detection threshold, a pulse train corresponding to that peak is extracted from the interleaved sequence of pulses. The original technique has been modified in Milojecic and Popovic (1992), where the CDIF-histogram is replaced by simpler and computationally more efficient histogram called the sequential difference (SDIF) histogram. The first-order SDIF-histogram also called the PRI histogram) shows the approximate distribution of the TOAs of a sequence. The second-order SDIF-histogram considers those time intervals of sequence that are calculated between each pulse and the next pulse but one appearing in a sequence, and so on. Thus, $d$th order SDIF-histogram presents frequencies of the occurrence of the following time intervals:

$$X_k^{(d)} = f_k \pm d - f_k, \quad k = 1, 2, \ldots, N - d.$$  

Fig. 2. shows the first two SDIF-histograms of an ideal PRI Stagger sequence with two positions. In the first-order SDIF-histogram (Fig. 2(a)), neither of the peaks exceed the exponential detection threshold. However, the second-order time intervals are constant, introducing a strong peak in the final histogram (Fig. 2(b)). The peak is called a stable sum, and it exists for any periodic pulse pattern as long as enough pulses are available. In general, if the period of a pulse train contains $d$ pulse intervals and no imperfections are present, a single peak indicating the stable sum is observed in a $d$th order SDIF-histogram. If pulse trains are corrupted, the detection of a stable sum becomes considerably more difficult due to false peaks in the histogram. A major limitation of the presented peak detection technique is that it is mainly designed to deinterleave and recognize Constant and Stagger PRIs from each other and does not take into account more complex PRI modulation types. Another weakness is the requirement for threshold selection, which critically depends on the amount of imperfections present in pulse trains.

A remarkable disadvantage of statistical techniques is that they lose all sequential information of pulse trains leading to ambiguities between different modulation types. For example, the PRI histograms of Stagger, Dwell & Switch and discrete Jittered PRI are all characterized by distinct peaks. Also higher level SDIF-histogram can be ambiguous, because stable sums exist not only for Stagger PRIs, but also for any other periodic modulation patterns. To overcome problems associated with histograms, some methods based on sequential information of pulse intervals have been proposed. In Ryoo, Song, and Kim (2007), the autocorrelation-based feature extraction technique has been presented to recognize four different PRI modulation types (Sliding, Jittered, Dwell & Switch, and Periodic). However, due to sensitivity of the features against real-world imperfections, compensation of missing pulses and removal of spurious pulses must be performed before feature extraction. In practice, the use of preprocessing can be infeasible because restrictive assumptions about the settings of PRI modulation parameters are made. In Noone (1999), an $N$-dimensional feature vector is constructed by using the difference information of the TOAs. A major drawback of the technique is that the proposed features are not invariant against common parameter variations, such as length and phase variations of regular PRI modulation patterns. To allow more flexible recognition, dimension reduction of the feature vector from $N$ to 2 is suggested in Rong, Jin, and Zhang (2006). This technique does not require $N$ to be fixed and it provides much better recognition accuracy in complex recognition environment compared to Noone (1999), but the features are still very limited in terms of separating capability.

3. Methods

In this section, we first present the robust feature set to distinguish the six PRI modulation types (Section 3.1), and then present a hierarchical classification concept that can be used to recognize sequential combinations of these PRI modulation patterns (Section 3.2).

3.1. Feature extraction

Feature extraction has a major role in PRI modulation recognition, because carefully chosen features can remarkably simplify the actual classification task, improving accuracy and reliability of the recognition. Because the unique characteristics of modulation patterns can be lost if preprocessing is applied to pulse trains before recognition, it is important to find features that are efficient even when preprocessing is not used. Thus, features should tolerate high amount of imperfections in pulse trains including missing and spurious pulses as well as inaccuracy in successive TOA measurements. A feature set should also be robust against variations in parameter...
settings including magnitude, deviation, and phase variations between patterns. Similarly, variations in the lengths of modulation periods between pulse trains should not affect the feature values. Previously, we have shown that five features can separate the six PRI modulation types in the feature space (Kauppi & Martikainen, 2007). These features describe the following properties of PRI sequences: (1) single histogram peak, (2) stable sum, (3) pulse interval changes, (4) unidirectional pulse interval changes, and (5) local extrema of pulse intervals. Next, we review the presented features.

### 3.1.1. Histogram-based features

The first two features are obtained from histograms. In practice, a histogram peak originating from a constant pulse interval sequence may spread over two or more histogram bins due to TOA uncertainty present in pulse trains. To ensure robust detection of histogram peaks, we calculate histograms by using the histogram stabilization algorithm presented in Algorithm 1. Pulse intervals are first sorted into ascending order, and they are then assigned in a current bin one by one as long as the first pulse interval does not fit to that bin. Based on the value of this pulse interval, a new bin is created and the construction of the histogram is continued.

#### Notations:

- \((x(k))\) \(k\)th ordered time interval value
- \(N\) total number of pulses
- \(\epsilon\) relative tolerance defining constant time interval
- \(X(m)\) starting value of the \(m\)th bin
- \(n(m)\) number of occurrences in the \(m\)th bin

Sort pulse interval values into ascending order

| init \(e\), \(k \leftarrow 2\), \(m \leftarrow 2\), \(n(1) \leftarrow 1\), \(X(1) \leftarrow x(1), X(2) \leftarrow [(1 + e_{\omega})/(1 - e_{\omega})]x(1)\) |
| while \((k < N)\) |
| if \((x(k) < X(m)) \) increase occurrences in current bin \(n(m) \leftarrow n(m) + 1\) |
| else if \((x(k) \geq X(m)) \) create adjacent bin \(n(m+1) \leftarrow 1, X(m+1) \leftarrow [(1 + e_{\omega})/(1 - e_{\omega})]X(m)\) |
| end |
| create distant bin \(X(m+2) \leftarrow [(1 + e_{\omega})/(1 - e_{\omega})]x(k)\) |
| \(n(m) \leftarrow 0, n(m+1) \leftarrow 1\) |
| \(m \leftarrow m + 2\) |
| end |
| \(k \leftarrow k + 1\) |

Algorithm 1. Histogram peak stabilization. We use the algorithm as a part of feature extraction to form PRI histograms as well as SDIF-histograms in several levels up to \(D_{\text{max}}\). Relative jitter tolerance \(\epsilon\) is a user parameter that has to be determined beforehand.

The first feature is obtained as a fraction of two highest histogram peaks in the PRI histogram:

\[
f_1 = \frac{n_{\text{max}}}{n_{\text{max}} - 1},
\]

where \(n_{\text{max}}\) indicates the highest, and \(n_{\text{max}} - 1\) the second highest histogram peak. Clearly, low feature value indicates that no intentional modulation is present (Constant PRI). The second feature is obtained from higher order SDIF-histograms and it describes the relative strength of a stable sum. Because a stable sum emphasizes periodicities but is not characteristic of random sequences, the feature is especially well suited to distinguish jittered from Stagger PRIs. We define the relative strength of a stable sum in the \(d\)th order SDIF-histogram as follows:

\[
u_d = \frac{n_d^{\text{max}}}{N - d - 1},
\]

where \(n_d^{\text{max}}\) is the highest peak in the histogram. Because the number of positions in Stagger PRI is generally unknown, we calculate the values of \(u_d\) for several \(d\) and take the maximum as a final feature value:

\[
f_2 = \max(u_d), \quad d = 2, 3, \ldots, D_{\text{max}}.
\]

where \(D_{\text{max}}\) is chosen according to the highest expected number of positions in Stagger PRI.

#### 3.1.2. Sequential information-based features

The other three features describe sequential information of pulse trains. We first calculate the vector of the second TOA differences (\(d^2\)TOAs) of a pulse train from (1) according to Noone (1999):

\[
z_k = x_{k+1} - x_k, \quad \text{for} \quad k = 1, 2, \ldots, N - 2.
\]

Note the distinction between this equation and (2) that is used to form SDIF-histograms. We retain only the directional information using the following transformation (Noone, 1999):

\[
s = \text{sgn}_s(z),
\]

where \(z\) is the vector of all \(d^2\)TOAs in a sequence, and a signum-function is defined as

\[
\text{sgn}_s(z_k) = \begin{cases} -1, & \text{when } z_k < -\epsilon \\ 0, & \text{when } |z_k| \leq \epsilon \\ 1, & \text{when } z_k > \epsilon. \end{cases}
\]

Here, \(\epsilon\) is a jitter tolerance to distinguish slight unintentional PRI variations from intentional one. The first feature which is based on sequential information describes the relative amount of changes in successive pulse intervals:

\[
f_3 = \frac{\sum_{k=1}^{N-2} |s_k|}{N - 2},
\]

where \(s_k\) is the \(k\)th element of the vector given in (5). Clearly, this feature is good in separating stable PRI sequences from non-stable ones. Next feature describes the relative amount of unidirectional pulse interval changes in a pulse train:

\[
f_4 = \frac{\sum_{k=1}^{N-2} |s_k|}{N - 2}.
\]

This feature is especially good in separating sliding PRIs from the other modulation types. Our last feature describes how often the evolution of pulse intervals changes its direction:

\[
f_5 = \frac{\sum_{k=1}^{N-3} |\text{sgn}(r_k)|}{N - 3},
\]

where \(\text{sgn}(x)\) is a standard signum-function, and \(r_k = s_{k+1} - s_k\) for \(k = 1, 2, \ldots, N - 3\). This feature is particularly good in separating jittered from Periodic PRIs.

### 3.2. Hierarchical classification

In order to recognize different PRI modulation subpatterns of Scheduled PRIs, we propose that pulse trains are not recognized as entire sequences as in previous studies, but rather analyzed temporally using a sliding window. However, feature extraction and classification of the short segments is problematic because the optimal window size depends on PRI modulation parameter settings that can vary arbitrarily from pulse train to another. Fig. 3 illustrates the problem that arises when using direct recognition of pulse train segments. A window \(M_2\) is suitable for segmenting PRI sequence shown on the top, because segments would capture the relevant information considering the specific PRI modulation type (Dwell & Switch PRI). However, the same window is not
suitable for the recognition of a sequence shown on the bottom because segments would contain information from two distinct modulation patterns (Dwell & Switch and Stagger PRI). Although a smaller window ($M_1$) would be applicable in this case, it is not, on the other hand, suitable for segmenting the upper sequence. This is because segments would contain insufficient information to recognize the specific modulation types (Dwell & Switch PRI sequence would be confused with Constant PRI).

The previous example demonstrates that direct recognition of the segments is not reliable in practice (fixed window size may easily lead to ambiguities especially between Constant and Dwell & Switch PRIs, Stagger and Jittered PRIs, or Sliding and Periodic PRIs). To make recognition results less dependent on window size selection, we suggest coarse detection of the subpatterns before actual categorization. Subpattern detection is performed using a sliding window, but the detection is not sensitive to window size selection because only coarse information is utilized. In Fig. 3, the benefit of such detection is evident when considering Dwell & Switch and Constant PRI as a single supercategory. In this case, the $M_1$ would be a meaningful window for both sequences.

The entire classification scheme for recognizing Scheduled PRI modulation patterns is depicted in Fig. 4(a). In the first stage, a subpattern detector uses a sliding window to find and extract subpatterns with static properties from the entire pulse train (the size $M$ of a window is determined by the number of DTOAs captured within each frame). The subpattern detector classifies each segment according to the following coarse-level information: (1) Stable PRIs meaning that no or only occasional changes occur in the consecutive pulse interval values, (2) Directional PRIs indicating increasing and/or decreasing pulse interval ordering, or (3) Non-directional PRIs meaning that the frame contains unstable pulse-to-pulse variation which does not follow increasing/decreasing ordering. We denote the associated category labels of these patterns as $\theta_i$, $i \in \{1, 2, 3\}$. If the subpattern detector finds successive frames with class labels $\theta_i$ and $\theta_j$ such that $k \neq l$, it can be concluded that properties of a PRI sequence have been changed. In this case, the subpattern detector extracts a subpattern from the overall sequence and passes it to the final classification stage. An appropriate component classifier then assigns the actual class labels $c_i$, $i \in \{1, 2, \ldots, 6\}$ for the subpattern, or optionally discards it as uncertain. Because a supercategory label is known after extraction, the recognition problem reduces to three two-category problems in the final stage. This is due to the inherent hierarchical structure of PRI modulation (see Fig. 4(b)): Constant and Dwell & Switch PRI are subcategories of Stable PRI, Periodic and Sliding PRI are subcategories of Directional PRI, and Jittered and Stagger PRI are subcategories of Non-Directional PRI.

4. Simulations and implementation

We next present an exemplary classifier design together with a simulation example. The selection of a realistic simulation model is an extremely important part of the classifier design—only realistic simulations can guarantee reliable and accurate recognition results in real-world unpredictable signal environment. After simulation model is determined, there are many possibilities to select component classifiers. We decided to use multi-layer perceptron (MLP) as a subpattern detector and three individual Kernel density estimators (KDEs) as final classifiers. The major reasons for selecting these classifiers were that (1) they provide posterior probability estimate of the category memberships that helps to monitor uncertainty of the classification, (2) they can be implemented efficiently (for KDE, this is true in the one-dimensional case), (3) they require no any distributional assumptions, and (4) they are straightforward to re-implement if the simulation model is modified later.

4.1. Synthetic data generation

We generated feature distributions describing a diverse signal environment using an extensive Monte Carlo simulation. The parameter limits for the simulation are presented in Table 1. Equal amounts of pulse trains from each six PRI modulation category were generated such that pulse train parameters were drawn uniformly from the shown ranges. Altogether $n = 36,000$ pulse trains were created, meaning that there were 6000 representatives in each category. After generating the pulse trains, feature extraction was performed according to Eqs. (3), (4) with $D_{\text{max}} = 64$, (7)–(9).

In the proposed model, very high variability in modulation parameters was allowed for different PRI modulation types. Jittered PRIs were modeled according to three different distributions (uniform, Gaussian, and discrete). It should be pointed out that although remarkable variability was permitted for parameters, the
Hierarchical classification of Scheduled PRIs. (a) The proposed classification scheme. At first, a subpattern detector searches and extracts three types of patterns from the overall sequence by classifying segments of dTOAs using a sliding window (mth subpattern is denoted by \( p_m \) and labeled by its associated supercategory label \( \theta_i \)). When a subpattern is found, one of the three final classifiers assigns it in one of the six elementary PRI modulation categories (labeled by \( \omega_j \)). (b) Classification hierarchy using the proposed method. The categories determined by the subpattern detector are named Stable, Non-directional and Directional PRIs. The final PRI modulation categories fall into these supercategories.

Table 1

<table>
<thead>
<tr>
<th>Proposed model to assist classification.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deviation of the average PRI</td>
</tr>
<tr>
<td>Jittered (uniform, Gaussian(^a))</td>
</tr>
<tr>
<td>Periodic</td>
</tr>
<tr>
<td>Sliding(^b)</td>
</tr>
<tr>
<td>Number of positions/bursts</td>
</tr>
<tr>
<td>Stagger(^c)</td>
</tr>
<tr>
<td>Dwell &amp; Switch(^d)</td>
</tr>
<tr>
<td>Jittered (discrete)</td>
</tr>
<tr>
<td>Length of the burst in pulses</td>
</tr>
<tr>
<td>Dwell &amp; Switch(^e)</td>
</tr>
<tr>
<td>Number of periods</td>
</tr>
<tr>
<td>Periodic, Sliding</td>
</tr>
<tr>
<td>Stagger</td>
</tr>
<tr>
<td>Missing pulses</td>
</tr>
<tr>
<td>Spurious pulses</td>
</tr>
<tr>
<td>TOA uncertainty</td>
</tr>
</tbody>
</table>

\(^a\) 3\( \sigma \)-limits determine the deviation of the Gaussian model.
\(^b\) Deviation is defined as a ratio of minimum to maximum PRI.
\(^c\) For a Stagger PRI with \( p \) positions, the number of distinct pulse interval values was varied from \( \max[2, (p - 2)] \) to \( p \).
\(^d\) Both periodic and non-periodic patterns were modeled.
\(^e\) For a given pulse train, the maximum ratio of longest to shortest burst was limited to 3:1.

The amount of imperfections in pulse trains depends on many factors such as SNR, radar timing circuitry, receiver measurement accuracy as well as the design of pulse detector, pulse parameter extractor and pulse deinterleaver. We assumed that in normal operation conditions, 15\% of pulses can be lost or 10\% of random spurious pulses may occur in sequences. We also assumed that slight TOA measurement uncertainty (including unintentional PRI variation) is present in normal operation conditions and therefore we added zero-mean, independent Gaussian noise into successive pulse intervals for the sequences (we set the upper limit of the noise standard deviation equal to \( \epsilon/2 \), meaning that approximately 5\% of originally stable pulse interval values exceeded the tolerance \( \epsilon \)). As a special case, larger unintentional variation was allowed for Constant PRIs than for the other PRI modulation types (Wiley, 2006). To ensure that feature distributions were sampled appropriately, we modeled imperfections for the 36,000 pulse trains as follows: 9000 randomly selected sequences were corrupted by increasing uncertainty in TOA measurements, another 9000 sequences were corrupted by dropping random pulses from them, another 9000 sequences were mixed with spurious pulses, and the remaining 9000 sequences were left free of errors.

4.2. Component classifier design

4.2.1. Subpattern detector

We trained a three-layer MLP to perform coarse-level classification in the subpattern detector. One motivation for using the MLP was that once appropriately learned, it can efficiently utilize information contained in the simulation data because the decision boundaries of an MLP can be nearly arbitrarily complex. This is not the case, for example, with a simple tree classifier, that...
can only produce decision boundaries perpendicular to the feature axis. Another major advantage of an MLP is its ability to estimate uncertainty of the classification outcomes (Richard & Lippmann, 1991). This property is important because it enables detection of unexpected pulse trains not explicitly represented in training data. Moreover, MLP can evaluate classification outcomes very quickly which is a critical requirement of the presented application. A network design and classifier training was performed using the neural networks toolbox in Matlab. Altogether 18,000 samples (samples were picked randomly from the simulation data consisting of 36,000 samples) were used to train our network. During training, the standard backpropagation algorithm was used to evaluate derivatives of the mean squared error (MSE) cost function with respect to network weights and biases. Minimization of the MSE criterion was performed using the quasi-Newton BFGS (Broyden–Fletcher–Goldfarb–Shanno) update algorithm, because it extensively provided fast and consistent convergence in our simulations. The network was trained in batch mode using binary target vectors, and training was stopped after 500 epochs because there was no noticeable change in a total training error after this number.

The size of the network output layer (three) was equal to the total number of supercategories. The size of the network input layer (three) was selected based on detailed analysis of the simulation data which revealed that the three features ($f_1$, $f_2$, and $f_3$) were required to classify coarse PRI modulation patterns (see Section 5.1). The size of a hidden layer (eight) was determined using the following model selection procedure. At first, the performance of several trained candidate MLPs with different numbers of hidden nodes was assessed with a separate test data set (the test set consisted of the remaining 18,000 samples of the simulation data that were not used during classifier training) against the following model selection criterion:

$$\max |z_{ij} - t_{ij}| \leq 0.1 \quad \text{for } i = 1, 2, 3,$$

where $z_{ij}$ is the $i$th output activation of the $j$th feature vector in the test data set, and $t_{ij}$ is the corresponding ideal target output. More specifically, if we denote the $j$th test sample as $f_j^0 = [f_1^0, f_2^0, f_3^0]^T$, the corresponding target output is $t_{ij} = 1$, if $f_j^0 \in \theta_i$, and $t_{ij} = 0$ otherwise. Importantly, the presented criterion takes into account whether the output of the MLP can approximate both category labels and posterior probabilities of the category memberships. The reason for this is that the criterion is satisfied only if the activation of every output node differs less than 10$\%$ from the ideal 0–1 target probability. The trained MLP candidates were evaluated against the criterion in the order of increasing complexity (we started evaluation with the MLP consisting of eight hidden nodes (99.1$\%$ of the test samples satisfied Eq. (10)). The limit 99$\%$ was used because there was no consistent improvement in the results beyond this point although the complexity of the network was increased. As suggested by extremely high classification accuracy with the test set, the selected MLP did not suffer from the problem of overlearning. The explanation for this was that the three-dimensional feature distributions were sampled comprehensively (18,000 samples were drawn for both training and testing).

After training and selecting the MLP, actual detection and classification of the subpatterns was performed using a sliding window. We improved robustness of the detector against abrupt anomalous changes in segments by accepting the subpattern detection only after it was observed at least in five successive frames. We empirically set a rejection threshold for the MLP for 0.95. If the maximum output activation of the MLP for some segment was below this threshold, the segment was tagged as uncertain and was not further classified.

### 4.2.2. Final classifiers

The inspection of the feature distributions revealed that the single feature (either $f_1$, $f_2$, or $f_3$) was sufficient to perform the final categorization of subpatterns (see Section 5.1). Therefore, we constructed three one-dimensional component classifiers to perform binary classification in the final recognition stage. To enable uncertainty estimation of classification outcomes, we estimated probability density functions (PDFs) of the training distributions using the Parzen method (Parzen, 1962, also known as kernel density estimation) in each three case. In practice, classification rules based on Parzen estimates can be determined using two alternative methods: either using visual inspection or by applying the Bayes rule, which assigns a sample to the category with the highest posterior probability (Duda, Hart, & Stork, 2001). In the latter case, a probabilistic threshold can be set to monitor recognition uncertainty and to discard unexpected patterns. Because the feature distributions of classifiers 1 and 3 were not overlapping in our simulations (see Section 5.1), we selected classification rules for them based on visual inspection. For classifier 1, we decided Constant PRI if $f_1 \leq 0.25$, Dwell & Switch PRI if $f_1 \geq 0.50$, and rejected a pattern otherwise. For classifier 2, we decided Sliding PRI if $f_2 \leq 0.25$, Periodic PRI if $f_2 \geq 0.50$, and rejected a pattern otherwise. For classifier 3, we used the Bayes rule to calculate category memberships (we used prior probability 0.50 for both categories). We set a rejection threshold for 0.95, meaning that at least 95$\%$ confidence was required to assign a category label to a pattern.

### 4.3. Test sequence

We investigated classification accuracy and rejection capability of our method using a diverse test sequence consisting of 10 distinct PRI modulation subpatterns. Table 2 shows the categories and key parameters of each subpattern in the sequence. The number of pulses in subpatterns was varied to see how the method can handle patterns with different lengths. To examine robustness of the classifier against large modulation parameter variations, PRI modulation patterns with very different parameters were included in the sequence. For example, PRI Stagger with both short (3 positions) and long (32 positions) periods were generated. Also Dwell & Switch PRI patterns were highly different: another consisted of several short PRI bursts whereas the other included only three long bursts. For Periodic PRI, both long- and short-term variation with different amplitudes were generated. Jittered PRI was modeled according to both discrete and Gaussian distribution.

### 5. Results

#### 5.1. Feature set separating capability

Visual inspection of the feature distributions was important part of the performance evaluation of the proposed methodology, because the discriminative feature set is the fundamental requirement to perform reliable real-world recognition. Fig. 5 presents feature distributions of an entire simulation data ($n = 36,000$) in three-dimensional feature space. These features were used with the MLP to detect the subpatterns. Clearly, the three supercategories were very well separable using the features $f_1$, $f_2$, and $f_3$. Specifically, Directional PRIs were distinguished from Non-directional PRIs using the feature $f_3$ (local extrema), and the feature $f_1$ (pulse interval changes) was completely able to separate Stable PRIs from Directional PRIs. Less intuitively, there was partial overlap between Stable and Non-directional PRI categories in the ($f_3$, $f_2$) plane. The reason for this was that our simulation model allowed more unintentional PRI variation for the Constant PRI than for the other modulation types. However, this overlap was prevented using the additional feature ($f_1$, single histogram peak).
Table 2

Description of the test sequence.

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Number of pulses</th>
<th>Positions</th>
<th>Number of pulses</th>
<th>Number of periods</th>
<th>Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pattern 1: Stagger</td>
<td>120</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pattern 2: Sliding</td>
<td>180</td>
<td>5</td>
<td></td>
<td></td>
<td>1:2 (min:max)</td>
</tr>
<tr>
<td>Pattern 3: Dwell &amp; Switch</td>
<td>Periodic</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pattern 4: Jittered</td>
<td>Discrete</td>
<td>220</td>
<td></td>
<td>64</td>
<td></td>
</tr>
<tr>
<td>Pattern 5: Constant</td>
<td></td>
<td>140</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pattern 6: Stagger</td>
<td>320</td>
<td>32</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pattern 7: Periodic</td>
<td>Non-periodic</td>
<td>330</td>
<td></td>
<td>5%</td>
<td>110</td>
</tr>
<tr>
<td>Pattern 8: Dwell &amp; Switch</td>
<td>Gaussian</td>
<td>300</td>
<td></td>
<td>3</td>
<td>85, 90, or 125</td>
</tr>
<tr>
<td>Pattern 9: Jittered</td>
<td></td>
<td>240</td>
<td>3%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pattern 10: Periodic</td>
<td></td>
<td>170</td>
<td></td>
<td>25%</td>
<td>10</td>
</tr>
</tbody>
</table>

Parzen estimates of simulation data utilized by the final component classifiers are presented in Fig. 6 (Gaussian kernel with a smoothing parameter $h = 2/n^{1/2}$ was used). Distributions of Constant and Dwell & Switch PRI (Fig. 6(a)) as well as Periodic and Sliding PRI (Fig. 6(c)) were extremely well separable with the features $f_1$ and $f_4$, respectively. The Stagger and Jittered PRI distributions (Fig. 6(b)) were partially overlapping. This overlapping was inherently impossible to prevent with any feature, because Stagger PRIs started to approximate discrete Jittered PRIs when the amount of random imperfections in the sequences was high.

Fig. 7 presents the same data using the features proposed by Rong et al. (2006). In this case, most distributions were heavily overlapping, indicating that the two features are not sufficient to achieve reliable real-world classification. Although the "frequency feature" was able to distinguish Sliding PRIs from the other PRI modulation categories well, we achieved similar separation using the feature $f_5$ (unidirectional pulse interval changes) without the requirement for transforming sequences to the frequency domain. The proposed “shape feature” is essentially similar to $f_5$.

5.2. Test sequence classification results

5.2.1. The effect of the window size

Fig. 8 shows the number of correctly classified subpatterns (true detections) as well as the number of spurious/misclassified subpatterns (false detections) in the test sequence using several window sizes. Both ideal and corrupted situations were investigated with and without the rejection option. In the corrupted case, we first randomly removed 7.5% of the TOAs from the original
between distinct subpatterns. Fig. 8(b) shows the corresponding results when using the rejection option. In this case, all the true patterns were found using several window sizes \((30 \leq M \leq 130)\), and all false detections were efficiently eliminated. Fig. 8(c) and (d) show the corresponding results for the corrupted test sequence. Even in this case, all false detections were efficiently eliminated (Fig. 8(d)). Some of the true subpatterns were detected as unknown due to uncertainty involved in the classification outcome. To conclude the above results, the recognition of the test sequence was very satisfying when using the proposed hierarchical classifier together with the rejection option: results were not critically dependent on the window size selection and false detections were efficiently eliminated.

5.2.2. Sequence-wise results

Fig. 9 shows sequence-wise classification results for the ideal test sequence using window size \(M = 60\). This window size was selected according to the results shown in Fig. 8. Fig. 9(a) shows the DTOAs of the test sequence, and Fig. 9(b) shows the output activations of the MLP for every segment (rejection threshold 0.95 is shown with a horizontal dashed line). Clearly, MLP approximated coarse class memberships of the 10 subpatterns with very high confidence. Transitions between category memberships were sharp, indicating that each subpattern was localized accurately (the extraction points are denoted by colored circles). Fig. 9(c) presents the final classification results of the extracted subpatterns together with the target categories. All the 10 patterns were correctly classified with high localization accuracy.

Fig. 10 shows corresponding results for the corrupted test sequence. Fig. 10(a) presents the DTOAs of the corrupted test sequence, where double intervals correspond to randomly removed pulses, and split intervals correspond to randomly inserted pulses. Results of the subpattern detection are shown in Fig. 10(b). MLP was able to recognize Stable and Non-directional PRIs with high confidence. Although there was more uncertainty associated with Directional PRIs (subpatterns #2, #7, and #10), they could still be detected. The method was robust against temporary loss in the activation of a certain output node (see MLP outputs for subpattern #7) as well as against very short activation peaks (see outputs between subpatterns #2 and #3). The final classification sequence, and then inserted random TOAs (7.5%) in it. Fig. 8(a) presents the results for the ideal test sequence when rejection option was not used. All 10 true subpatterns were successfully found using a wide range of window sizes \((30 \leq M \leq 160)\). Because no rejection option was used, also unwanted subpatterns were present when using large window sizes. Further investigation revealed that the spurious patterns were located in the border areas.

![Fig. 8](image)

Fig. 8. Test sequence classification results using various window sizes (the size of a window \(M\) is given as the number of pulse intervals captured by the window). True detections denote the number of correctly found subpatterns, and false detections denote the total number of misclassified and extra subpatterns found. Results of (a) the ideal sequence without using the rejection, (b) the ideal sequence when using the rejection, (c) the corrupted sequence (7.5% of removed + 7.5% of inserted pulses) without using the rejection, and (d) the corrupted sequence when using the rejection option.
Fig. 9. Sequence-wise classification results for the ideal test sequence using the window $M = 60$: (a) the test sequence, (b) subpattern detector outputs, and (c) estimated final PRI modulation categories together with the corresponding target categories. The final categories are: 1 = Constant, 2 = Stagger, 3 = Jittered, 4 = Sliding, 5 = Dwell & Switch, and 6 = Periodic PRI.

results are shown in Fig. 10(c). It can be seen that 9 out of 10 patterns were correctly recognized and a 32-position PRI Stagger (subpattern #6) was detected as unknown. The rejection was not surprising in this case, because the distinction between Stagger and discrete Jittered PRI sequence was not obvious even after manual inspection.
5.2.3. Robustness against imperfections

We tested robustness of the classifier against real-world imperfections also more systematically. Table 3 summarizes the recognition results ($M = 50$) in several conditions when different types of imperfections (missing pulses, spurious pulses, Gaussian noise) were incorporated in the test sequence. A total number of found subpatterns (true detections) and a total number of misclassified or spurious patterns (false detections) are reported, as well as which subpatterns were misclassified or detected as unknown. All the true patterns were recognized when 5% of pulses were missing in the original sequence. In the case of 10–15% of missing pulses, a single pattern (#6, 32-position Stagger PRI) was reported as unknown, and two additional subpatterns were rejected in the cases of 20% or 25% of missing pulses. Remarkably, no false detections were observed in the case of missing pulses for up to 25%. This is an important result, because the test condition was more challenging than that suggested by our simulation model, where at most 15% of missing pulses was assumed to be present in the sequences. Classification results with spurious pulses were similar to those with missing pulses. The only misclassification was observed in the presence of 20% of randomly added pulses, when Sliding PRI pattern (#2) was misclassified as a Periodic PRI. Results after inserting Gaussian noise in the test sequence are also presented in Table 3. After insertion, the test sequence contained high amount unintentional PRI variation (two to five times larger variation compared to the fixed jitter tolerance $\varepsilon$), distorting especially stable parts of the sequence. Although we did not assume such conditions during classifier training, no misclassified or extra subpatterns were present in the results.

5.3. Discussion

To conclude, our classifier provided consistently desirable results in various challenging recognition conditions emulating real-world signal environment. The recognition results were very accurate as long as unique characteristics of different modulation patterns were preserved, and misclassifications were consistently avoided when unexpected or ambiguous patterns were present. In addition, the results were not critically dependent on the window size selection. Although several recognition conditions were tested in our simulations, there are at least two important situations that should be further investigated. Firstly, recognition of very short subpatterns (say, less than 50 pulse intervals) should be analyzed thoroughly with various window sizes. Secondly, subpattern detection should be investigated in situations where the ordering of subpatterns is arbitrarily varied. For example, sequential combinations of Stagger and Jittered PRI patterns are difficult to extract in practice, because both patterns belong to a same coarse-level category. However, extraction and detection of such combinations might be possible by using an additional specialized detector before the final categorization.

6. Conclusions

We developed a novel technique for the automatic recognition of radar PRI modulation patterns to support real-world radar emitter classification in passive radar intercept receivers. The main focus was to address the recognition of scheduled (dynamically varying) PRI modulation patterns emitted by MFRs, but the presented ideas are applicable for the recognition of also other types of interpulse modulation patterns (e.g. both static and scheduled PW and RF agility patterns). We utilized two-stage hierarchical classification scheme, where different subpatterns with consistent properties were first detected and extracted from the entire pulse train based on coarse-level information of the pulse interval ordering, and the extracted subpatterns were then classified in more detail. The presented method was accurate even when considerable parameter variation and high amount of imperfections in modulation patterns was allowed. Importantly, pulse trains possessing a high level of uncertainty were automatically detected. Future work involves the estimation of modulation parameters based on PRI modulation recognition results, real-time implementation of the method, and more structured hierarchical classifier design.

Acknowledgements

The authors wish to thank Dr. Richard G. Wiley for his valuable comments and suggestions considering realistic simulations. This work was partly supported by the Academy of Finland (application number 213462, Finnish Programme for Centres of Excellence in Research 2006–2011), and partly by TISE (Tampere Graduate School in Information Science and Engineering).

References


