*International Journal of Information and Communication Sciences*
2026, Vol. 10, No. 3, pp. 57-71

Research Article

# Compatibility of Adaptive Filtering and Pulse Compression, and Knowledge-aided Multi-technology MTI Processing

**Xubao Zhang**[1, 2, *]

[1]Department of EE, Xi'an University of Electronic Science and Technology, Xi'an, China
[2]Research and Development, Unitron Sonova, Kitchener, Canada

## Abstract

This paper researches adaptive stagger-block-lattice-filtering (SBLF), phase-code pulse compression (PC), and stagger PRI (pulse repetition interval) technologies, which can be used in an MTI radar. First, we deduce the compatibility between the MTI filtering and PC, especially the placement order of both. When the MTI filter is a transversal filter, the overall performance is independent of the placement order; however, when the MTI filtering is adaptive filtering, e.g., real-time weight calculating and filtering, the overall performance is much dependent on their placement order. The deduction and simulation verify that a much better signal-clutter-ratio improvement is reached when the PC is placed behind the adaptive filtering. The high speed and small radar-cross-section of a target compel an MTI radar to be upgraded with stagger PRI, PC transmission, and adaptive filtering. In such a case, only arithmetic computation largely restricts clutter suppression performance in complex environments. Our experiments of years prove the necessity of incorporating AI into MTI processor, i.e., utilizing abundant prior knowledge. Then, we describe the basic knowledge for the multi-technology processor: adaptive filtering reasoning center, performance behaviors of two filtering modes, detection threshold set-up, target-terrace identification, and utilization of stagger PRI. Thereupon, five heuristic strategies for the intelligent operation are proposed: 1) non-clutter block decision and threshold set-up, 2) PC target-terrace identifications, 3) SBLF coefficient estimation, 4) Establishment of clutter-map, 5) Utilization of multiple stagger PRIs. In order to verify the effectiveness of the intelligent operations, we make many experiments using computer and the basic architecture parameters are based on a ship-borne radar; the generated clutters reflect land, sea, and weather returns, and the selected targets reflect weak, high-speed aircraft returns. In the strong homogeneous and severe heterogeneous clutters, this processor demonstrates maximized performance to acquire the weak targets.

## Keywords

Adaptive Filtering, Artificial Intelligence, Lattice Filter, Pulse Compression, Stagger PRI, MTI

## 1. Introduction

Today's MTI radars always incorporate multiple technologies to meet requirements of measurement and national defense, such as pulse compression (PC), frequency agility, stagger PRI (pulse repetition interval), and adaptive filtering [1-3]. The utilization of these technologies faces a big challenge of their compatibility. Thereinto, the compatibility of PC and adaptive filtering leads to much concern. Due to technological requirements, in conventional MTI processing,

*Corresponding author: xbzwdl@yahoo.com (Xubao Zhang)

**Received:** 10 October 2025; **Accepted:** 22 October 2025; **Published:** 28 April 2026

the PC is placed ahead of the MTI filter/canceller [2, 4, 5] and the overall performance is achieved well, independent of their placement order. However, when an MTI processor involves adaptive filtering, the weight coefficients are to be calculated in real-time rather than designed in advance. The adaptive filtering has to adopt the delay-filtering mode and employs the target-free data to calculate the weight coefficients only representing clutter. If the PC processing performs ahead of the filtering, the PC output contains target sidelobes in a wide-range region, close to the target main lobe; thus, it is hardly possible to find training data to make valid weight estimate. So, in this paper, we first research compatibility of the PC and MTI filtering, particularly, the placement order of both technologies.

Based on the stationary process assumption, many literatures studying adaptive filters describe the optimization theories, which are restricted within numerical computations [3, 6, 7]. In the early 1980s, references [8-10] introduced spectrum estimation and practical adaptive radar processors, and had incorporated some nonnumerical operations due to reasoning requirements; actually, those were just a class of primary intelligent operations. Because of the technology limitations and low-level devices, the adaptive track speeds of those radars were not fast and the reasoning strategies were not complete. By the 21st century, AI in signal processing has rapidly developed, associating with speech recognition, navigation, radar systems, etc. [11-13]. The training data cannot contain the target return and it is a crucial step to use them to estimate the adaptive weights. Reference [12] employed the measured data from radar programs to illustrate realistic effects limiting radar performance. Then, they examine intelligent training methods to enhance space-time adaptive detection performance. They described this issue in detail.

Speeds of modern aircraft and missiles are extremely high, up to Mach 3 or larger, so the Doppler frequency may also reach the corresponding range. No matter a uniform PRI (UPRI) is selected to be low, medium, or high, it always causes speed ambiguity or/and range ambiguity. Stagger-PRI pulse transmission is commonly used to solve the speed ambiguity [2]. Large time-width PC is also involved in MTI radar for low pulse transmission power, high range resolution, and electronic counter-countermeasures [14, 15]. Thus, the adaptive MTI processor always incorporates the both technologies. An adaptive filtering usually produces its coefficients in two ways: one is to design them in advance with the matched or eigenvector algorithm [7] and to store them in a weight base; the other is to calculate them in real-time with the lattice algorithm [16]. Based on many years' studies on adaptive filtering for a stagger-PRI MTI system, we propose a block lattice filtering algorithm, which calculates the coefficients with range x azimuth block data of the stagger-PRI returns and performs optimal improvement factor in the stationary clutter as the lattice filter with UPRI does. The filtering algorithm is called the stagger-block-lattice-filtering (SBLF). In addition, this structure has a low computation load and low sensitivity of operation word length in comparison with the transversal filter structure, so it is suitable to implement in a real-time filtering processor.

This paper describes prior knowledge for introducing intelligent operations into the multi-technology MTI processor, and the effects of incorporating heuristic strategies based on the prior knowledge. Then, we demonstrate data and graphs' results to illustrate the maximized performances of the processor in homogeneous and heterogeneous clutters. In simulation experiments, the conditions of strong, severe environment clutters and weak, high-speed flight targets are applied.

## 2. Compatibility Between MTI Filtering and Pulse Compression

### 2.1. Conventional MTI Processing

In conventional MTI processing, the filter is a canceller or transversal filter [7], which is designed in advance and works over coherent azimuth data; the phase-code or frequency-modulated PC performs matched-filtering along range bins. When both technologies work together, their compatibility or placement order can be verified. Given a radar return input (envelope) $x_r(t_n, l)$, $\{t_n\}$ coherent processing interval (CPI) times, $n \in \{0, 1, \cdots, N_a - 1\}$, $N_a$ number of azimuth data and $l \in \{1, 2, \cdots, L_r\}$, $L_r$ range bins' length, it contains both PC and Doppler frequency information.

When we place the PC ahead of the MTI filter, the PC output $y_{pc}(t_n, l)$ is represented as

$$y_{pc}(t_n, l) = \sum_{j=1}^{L_{pc}} x_r(t_n, l + j - 1)\, h_{pc}(j), \qquad (1)$$

where $h_{pc}(j), j \in \{1, 2, \cdots, L_{pc}\}$, is the impulse response of the compressor and $L_{pc}$ is the PC length. For simplification, we select a phase-code compressor, then $h_{pc}(j)$ is its code. Equation (1) is a computation along range bins at $t_n$. When $y_{pc}(t_n, l)$, $n \in \{0, 1, \cdots, N_a - 1\}$, goes through a transversal MTI filter, the filter output $y_{tf}(t_n, l)$ at the range gate bin $l$ is represented as

$$y_{tf}(t_n, l) = \sum_{i=0}^{N_a - 1} y_{pc}(t_{n+i}, l) h_{tn}(\tau_i)$$

$$= \sum_{i=0}^{N_a - 1} \sum_{j=1}^{L_{pc}} x_r(t_{n+i}, l + j - 1)\, h_{pc}(j)\, h_{tn}(\tau_i), \qquad (2)$$

where $h_{tn}(\tau_i)$ is impulse response of the transversal filter, whose time interval can be uniform or stagger, i.e., $t_n - t_{n+1} = t_{n+1} - t_{n+2}$ or $t_n - t_{n+1} \neq t_{n+1} - t_{n+2}$.

When we place the MTI filter ahead of the pulse compressor, the filter output $y_{tf}(t_n, l)$ at range gate bin $l$ is represented as

$$y_{tf}(t_n, l) = \sum_{i=0}^{N_a-1} x_r(t_{n+i}, l)\, h_{tn}(\tau_i). \quad (3)$$

Equation (3) is a computation over azimuth data. Following the filter, the compressor accepts its output and the PC output $y_{pc}(t_n, l)$ in the range bin $l$ is represented as

$$y_{pc}(t_n, l) = \sum_{j=1}^{L_{pc}} y_{tf}(t_n, l+j-1) h_{pc}(j)$$

$$= \sum_{j=1}^{L_{pc}} \sum_{i=0}^{N_a-1} x_r(t_{n+i}, l+j-1) h_{tn}(\tau_i) h_{pc}(j). \quad (4)$$

Comparing (4) with (2), we conclude that the overall performances of both orders of inverted placements are the same.

## 2.2. Utilization of Adaptive MTI Filtering

In today's MTI processing, the filtering may be adaptive filtering of real-time structure, rather than non-real-time one. Then both outputs of (2) and (4) may not be equal because the filter impulse response $\{h_{tn}(\tau_i)\}$ is variable in the inverted placements. An adaptive filter needs to calculate its weight coefficients in real-time and the coefficients cannot be corrupted by the target in radar return to suppress the clutter only. For this requirement, a delay-filtering mode is always selected. That is, the data for estimating weights are ahead of the data to be filtered. There are two schemes: 1) When the PC is carried out ahead of the filtering, the PC target output forms a high target main lobe in the middle of both wide-range sidelobes, which cause a large problem of filtering weight estimation since the clutter output data are mixed with the target sidelobes, except when the target is quite weak. 2) When the filtering is carried out ahead of the compression, only clutter data is present before the PC target. Thus, the data in front of the target can be employed as the training data for the weight estimation, even if the PC target is non weak. Thus, the outputs of (2) and (4) do not represent the case of utilizing the adaptive filtering. To further verify the deduction, we make the following simulation experiments.

### 2.2.1. Comparison in Homogeneous Clutter

In the experiment with homogeneous clutter, the phase-code PC is of a minimum sidelobe 63-bit code, with peak sidelobe of 4 (-23.9 dB) [15], see Section 5.1.2. The adaptive filtering is real-time SBLF, which is equivalent to a backward prediction transversal filter [16], see Section 5.1.1. We generate the land and cloud clutters [17], which are of $P_g$ 30 dB, $F_g$ 12 Hz, $D_g$ 7.2 Hz[1] and $P_w$ 10 dB, $F_w$ 240 Hz, $D_w$ 58 Hz, respectively, and the target signals, of $P_{t1}$ 5 dB, representing a weak target and $P_{t2}$ 30 dB, non-weak target, over range bins from 61 to 123 and 171 to 233, respectively. We select $N_a$=7, the stagger PRIs 1.29, 1.25, 1.49, 1.39, 1.44, 1.34, and 1.53 ms, the corresponding code 27 26 31 29 30 28 32. The blind-speed extension is 29, the stagger ratio is 1.23, and the average PRI is 1/720 s.

Figure 1 shows the performances of two schemes of the MTI processor in the homogeneous clutter. Figure 1 (a) results from the scheme of the PC ahead of the SBLF; the black, blue, and red curves represent the PC input, PC output, and SBLF output, respectively. We observe that the input clutter submerges the 5 dB target signal and reveals the 30 dB non-weak target terrace, which is irregular due to the clutter corruption. The PC output shows only one target impulse at bin 171, 66 dB high; the SBLF output shows two target impulses: one is at bin 61, 46.5 dB high, with about 30 dB lobes at its two sides, and the other is at bin 171, 44.3 dB high, with about 28 dB lobes at its two sides. The result indicates that this scheme acquires 16.5 dB signal-to-clutter ratio (SCR) output with the weak target and 16.3 dB SCR output with the non-weak target. This scheme has good clutter-suppression with the weak target but not with the non-weak target. Why? When a target is present and the PC filter matches it, its output forms two wide 62-bin target sidelobes. The sidelobes are much lower than its main lobe but they may not be much lower than the clutter level. When this happens, the coefficient estimation will be controlled by the front sidelobe to a certain degree. Roughly calculating, the sidelobes of the weak target are about 17 (5+12) dB and the clutter neighboring the main lobe is about 48 (30+18) dB, so the SBLF coefficient estimate is not dominated by the target sidelobe; however, the sidelobes of the non-weak target are about 42 (30+12) dB and the clutter neighboring the main lobe is about 48 (30+18) dB, so the SBLF coefficient estimates are much controlled by the target sidelobe; the SBLF filters out the target to a large degree.

Figure 1 (b) results from the other scheme of the pulse compression behind the SBLF; the black, blue, and red curves represent SBLF input, SBLF output, and PC output powers, respectively. The input signal submerges the 5 dB target signal and reveals the 30 dB non-weak target terrace, which is irregular due to clutter corruption. The SBLF output shows a regular target terrace in the region between bins 171 and 233, about 38 dB high; the PC output reveals two target impulses: one is at bin 61, 49.3 dB high, with about 30 dB lobes at its two sides, and the other is at bin 171, 74.1 dB high, with about 49 dB lobes at its two sides. The result indicates this scheme acquires 19.3 dB SCR output with the weak target and 27.1 dB SCR output with the non-weak target; this is a maximized, much better clutter suppression than (a). Why? In this scheme, when a target is present, the range bins in front of the target contain the clutter only. So, the SBLF coefficient estimates in the range bins are dominated by the clutter, and the SBLF filters out only the clutter; the PC output impulse of the non-weak target is 25 dB higher than that of the weak target.

[1] In this paper, all the clutters are modeled as a Gaussian spectrum. $P_g$, $P_w$, and $P_s$ denote the powers of land, weather (clouds or rain), and sea clutters, respectively; $F_g$, $F_w$, and $F_s$ denote the center frequencies of the three-class clutters, respectively; $D_g$, $D_w$, and $D_s$ denote the standard variance bandwidths of the three-class clutters, respectively. The target is modeled as a point-frequency spectrum. $P_t$ and $F_t$ denote its power and frequency, respectively.

### 2.2.2. Comparison in Heterogeneous Clutter

For the experiment in heterogeneous clutter, we generate the sea clutter, which is of $P_s$ 20 dB, $F_s$ 12 Hz, $D_s$ 48 Hz, the rain clutter, varying $P_w$ from 10 dB to 25 dB, $F_w$ 240 Hz, $D_w$ 48 Hz, and the weak target, $P_{t1}$ 5 dB, non-weak target, 15 dB, in regions from bins 61 to 123 and bins 171 to 233 individually. Stagger PRIs are 1.44, 1.29, 1.25, 1.34, 1.39, 1.49, 1.53 ms, the corresponding code 30 27 26 28 29 31 32. $N_a$, the average PRI, blind-speed extension, and stagger ratio are the same as those in the homogeneous case. The phase-code PC and adaptive filtering are also the same as those.

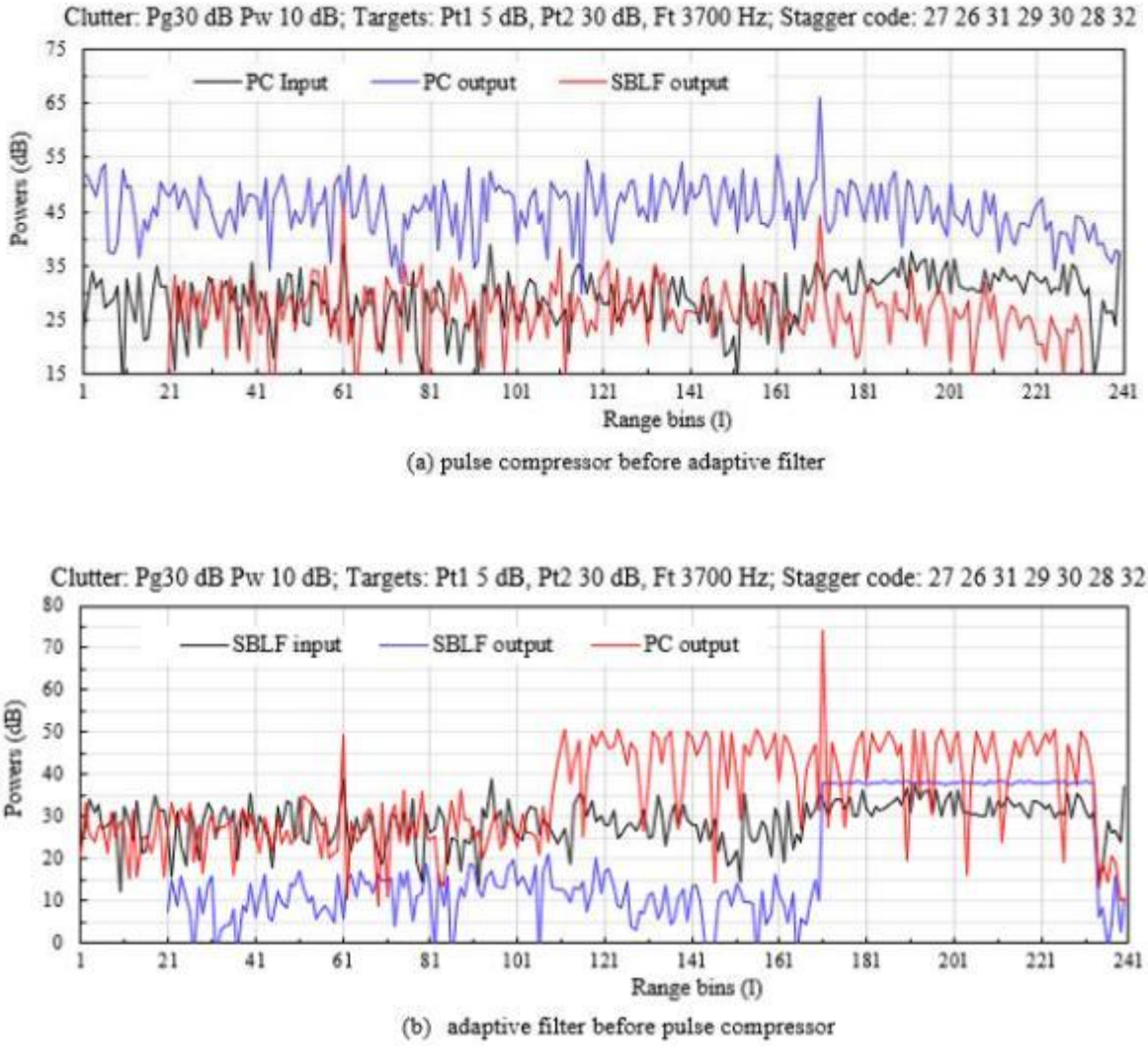


***Figure 1.*** *Performances of two schemes of the MTIprocessor in a homogeneous clutter.*

Figure 2 shows the performances of the MTI processor with the two schemes in the heterogeneous clutter. Figure 2 (a) results from the scheme of the PC ahead of the SBLF; the black, blue, and red curves represent PC input, PC output, and SBLF output, respectively. The input clutter submerges both weak and non-weak targets because the clutter power is much higher than the targets' powers. The PC output shows only one target impulse at bin 171, 51.5 dB high; the SBLF output reveals one target impulse at bin 61, 42.2 dB high, with about 38.9 dB lobes at its two sides, and the other target impulse at bin 171, 54.2 dB high, with about 38 dB lobes at its two sides. The result indicates that this scheme acquires 3.3 dB and 16.2 dB SCR output with the weak and non-weak targets, respectively. In fact, the weak target impulse is almost submerged in the nearby output. The result further indicates that this scheme does not perform good clutter suppression. Why? The answer is the same as that of Figure 1 (a).

Figure 2 (b) results from the other scheme of the PC behind the SBLF; the black, blue, and red curves represent the SBLF input, SBLF output, and PC output, respectively. The input signal submerges both weak and non-weak targets. The SBLF output reveals only one irregular target terrace in the region of bins 171 to 233, about 28 dB high; the PC output

reveals two target impulses: one is at bin 61, 51.1 dB high, with about 38 dB lobes at its two sides; the other is at bin 171, 61.8 dB high, with about 41 dB lobes at its two sides. The result indicates that this scheme acquires 13.1 dB SCR output with the weak target and 20.8 dB SCR output with the non-weak target. This result further indicates that this scheme provides a maximized, much better clutter suppression than (a). Why? The answer is the same as that of Figure 1 (b).

We summarize that 1) when an MTI filter is designed well in advance, such as the traditional canceller or weight-base (ROM) filter, the overall performance is independent of the placement order of the MTI filter and pulse compression; 2) when an MTI filtering is adaptive, specifically, its weights are calculated in real-time, the overall performance depends on the placement order of the MTI filtering and pulse compression, and a much better SCR improvement is achieved if the pulse compression is placed behind the adaptive filtering. Thus, in the later experiments, the adaptive filtering is placed ahead of the pulse compression.

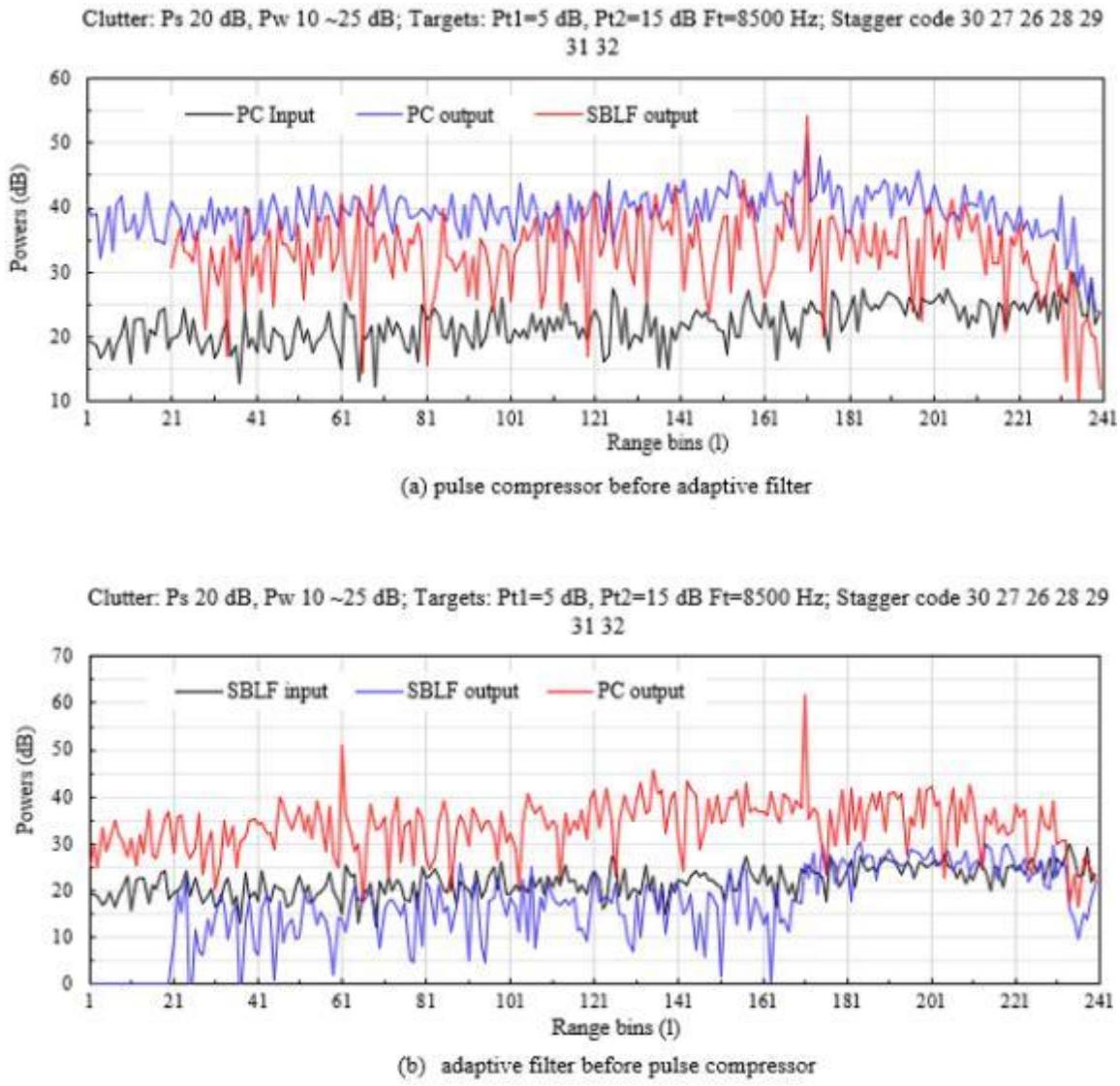


***Figure 2.*** *Performances of two schemes of the MTIprocessor in a heterogeneous clutter.*

## 3. Prior Knowledge for the Intelligent MTI Processor

The so-called knowledge means radar professionals' knowledge. The following knowledge is basic for the intelligent MTI Processor.

### 3.1. Reasoning Construction of the Adaptive SBLF

Aircraft and missile returns (targets) or land, sea, and weather returns (clutters) from echoed coherent pulses emit-

ted by an MTI radar are autocorrelated over adjacent azimuth data, but the Doppler frequencies of the two-class returns are different. This builds the foundation of an MTI processing. Advent of high-speed missile and stealth aircraft for electronic countermeasures leads to the radar technology upgrade. Generally, a target's radar-cross-section (RCS) is small and its return power is weak while a clutter return has a very large RCS and submerges the target. Thus, the MTI processing focuses on strong clutter suppression and weak target acquisition, and incorporates the adaptive filtering, stagger PRI, and large time-width PC technologies. Theoretically, the optimal filter algorithm can reach a maximized improvement factor of 50 dB or larger [6, 16]. In practice, such a performance is compromised due to clutter heterogeneity and multi-technology involution. For example, to prevent the target from being filtered out, the adaptive filtering has to estimate the coefficients in data of a block where no target is present, referred to as training data [18]. The crucial problem is how we can realize them. Thus, the first operation for this processor to perform is not numerical computation like "y=F(x)" but rather a reasoning operation like "if-then", specifically, learning, analyzing, identification, examination, decision, and memorizing. As soon as the training block is recognized, the adaptive filter coefficients are calculated with no target presence, then are used to filter the next block data. This way is the common delaying-and-filtering or delay-filtering.

In the experiments below, one data array is segmented into multiple blocks. One block is composed of data $L_b$ x $N_a$, $L_b$ length of the block bins and $N_a$ number of the azimuth data, and $N_a$=7 in this whole paper. When a burst of blocks enters the processor, several blocks carry out the SBLF operation together with relevant intelligent operations and are called the reasoning center. Figure 3 shows the center composed of five blocks, which are called as Estimate, Test, Test I, Test II, and Test III blocks in turn. Estimate block before Test block is used to estimate SBLF coefficients when the training data are recognized inside; Test I, Test II, and Test III blocks after Test block are used for filtering-self for setting-up the detection threshold, and delay-filtering for identifying the PC target terrace. The Test block is used to filter, identify, and declare the resulting information data, and send out them into the clutter- map. This processing is repeated as the sequential block data enter.

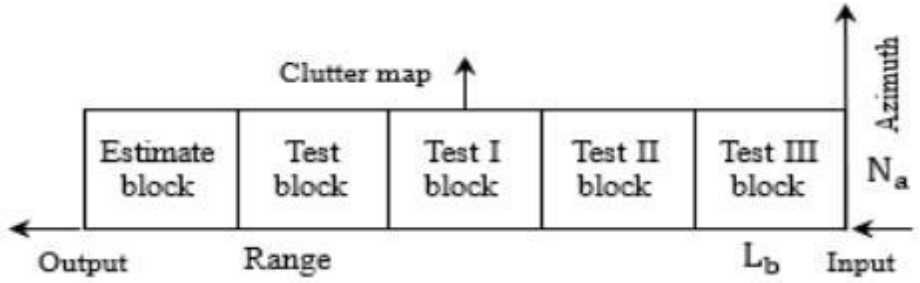


***Figure 3.*** *Reasoning center of the adaptive SBLF processor.*

## 3.2. Performance Behaviors of Two Filtering Modes

Now we make an experiment with two different filtering modes. A two-clutter plus two-target return array of 240 range bins x 7 azimuth data is segmented into 12 blocks of length 20 bins; 5 dB and 30 dB PC targets are placed in bins 41 and 151, respectively. When adjacent blocks' data enter the reasoning center of Figure 3, 1) data of each block of Estimate, Test, and Test I are used to estimate SBLF coefficients, which are used to filter their individual block data; this mode is called estimating-and filtering-self or filtering-self for short; 2) the SBLF coefficients are calculated with Estimate block data and are used to filter Test block data; this mode is called block-delay filtering. For PC target return, its data are present in several blocks, then the data of Test, Test I, Test II, and potential Test III blocks are filtered with the same SBLF coefficients of Estimate block; this mode is called terrace-delay filtering. Figure 4 shows the test results from the SBLF processing, and the two modes perform their individual behaviors. The parameters of the land plus cloud clutters are: $P_g$ 60 dB, $F_g$ 12 Hz, $D_g$ 7.2 Hz plus $P_w$ 10 dB, $F_w$ 240 Hz, $D_w$ 58 Hz, respectively, and the other parameters are listed at the top of Figure 4. The white noise power is normalized to 0 dB. The black curve represents the power of SBLF input, fluctuating around 60 dB. The blue curve represents the power of SBLF output in filtering-self mode, fluctuating around 13 dB, which does not reveal any target cue. The red curve represents the output power of SBLF output in terrace-delay filtering mode, two PC target terraces appear in place and each length is 63 bins; one is the irregular terrace, about 22 dB high, from bins 41 to 103, and the other is the regular terrace, about 45 dB high from bins 151 to 213, the clutter at both sides of the terraces is fluctuating around 17 dB. There is no data from bins 1 to 20 due to no filtering. These test results indicate that for a PC target, the terrace-delay filtering mode provides remarkable cues. This is one key prior knowledge of the in-

telligent processing.

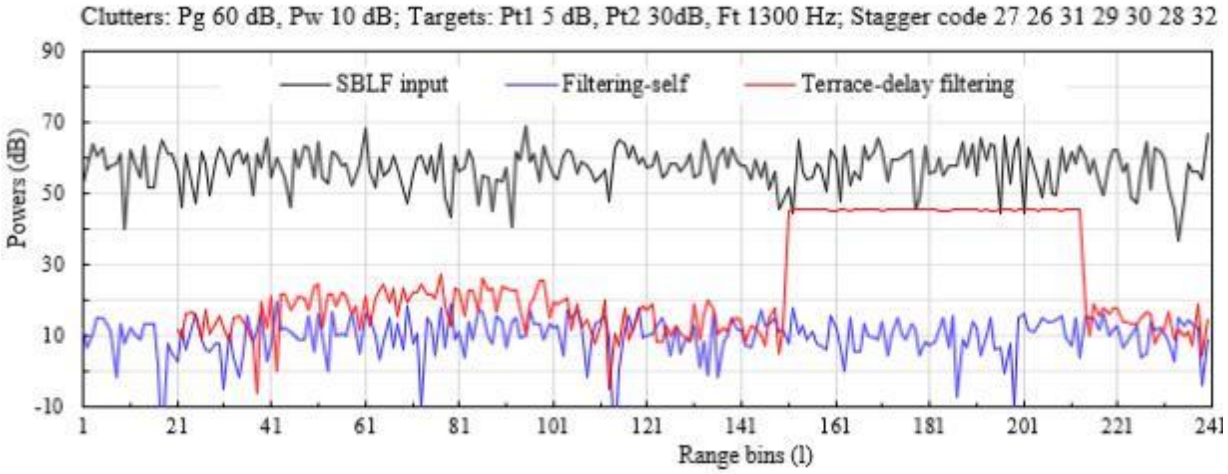


***Figure 4.*** *Performance behaviors of two filtering modes of the MTI processor in a homogeneous clutter.*

## 3.3. Adaptive Detection Threshold and Target-terrace Identification

From the blue curve in Figure 4, no matter the targets are 5 dB weak or 30 dB non-weak, the filtering-self output of the SBLF has no target cue and is relatively stable; theoretically, the output is noise plus the residual clutter/target, relatively stable. Utilizing this filtering behavior, we can use the filtering-self output to set up target detection thresholds in homogeneous and heterogeneous clutters. When comparing the blue curve with the red SBLF output, the processor easily determines whether a target is present in Test block. In the case of a PC target, if the average output of each of the Test, Test I, Test II, and potential Test III blocks exceeds a threshold determined by the filtering-self outputs, a PC target in the blocks is acquired; this is a simple reasoning in adaptive clutter suppression but its effectiveness is quite remarkable. So, it is a great finding of setting up an adaptive threshold with the filtering-self outputs, and the adaptivity also increases detection sensitivity, i.e., beneficial to weak target acquisition. Specifically, we take the average of filtering-self output magnitudes $O_E$, $O_T$, and $O_{TI}$ of the Estimate, Test, and Test I blocks as

$$T_D = (O_E + O_T + O_{TI})/3, \quad (5)$$

and call the average value $T_D$ the output threshold to detect a target. Additionally, a threshold factor $F_T$, from 1.5 to 2.5, is also selected to control the detection sensitivity. The product of $T_D$ multiplied by $F_T$, $T_{Da} = F_T T_D$, is called the adaptive threshold. Later, we use $T_{Da}$ to detect targets in the homogeneous clutter. When a PC target is present, usually the target terrace on SBLF output appears in several adjacent blocks, e.g., Test, Test I, Test II, and potential Test III; however, when the target signal is corrupted much or in a heterogeneous clutter, the terrace may be irregular, even hidden. Additionally, as long as the target is declared in Test block, the block data cannot be training data, and the SBLF coefficients from Estimate block are maintained until the target-terrace declaration ends.

In the case of heterogeneous clutter, since statistics of the clutter are unstable, formulas (5) of the output threshold may be modified and the block length (average number of estimates) needs to be increased to obtain stabler SBLF coefficients.

## 3.4. Multiple Stagger PRIs

This SBLF is compatible with the stagger PRI, which makes the frequency/speed response period extended by many times longer than that frequency period of the low UPRI, then the MTI overcomes the speed-ambiguity disadvantage. For example, for an S-band radar of 3 GHz, assume that an aircraft speed is Mach 1, then its Doppler frequency is at highest 6.86 kHz. If a uniform PRF (UPRF) is 720 Hz, there are 9.53 ambiguous speeds within Mach 1. However, when the stagger PRI is designed to be of average PRI, 1/720 s, and its stagger code is from seven integers, 26, 27, 28, 29, 30, 31, and 32, the frequency period extension is 29, and the corresponding stagger PRIs are 1.25, 1.29, 1.34, 1.39, 1.44, 1.49, and 1.53 ms. In order to get the shallowest notches, the seven integers need to be rearranged by the mathematical programming [ 19]. As a result, the rearranged integers 27 26 31 29 30 28 32 are called the stagger code, which has the stagger PRIs 1.29, 1.25, 1.49, 1.39, 1.44, 1.34, 1.53 ms, and the deepest notch is 0.97 dB within 5 times the UPRF. The SCR improvement of the SBLF is still impacted when the Doppler frequency drops into a notch. If the average SCR improvement factor of the SBLF is

53 dB and the notch depth is −5 dB, the improvement with a target dropping into the notch will be 48 dB only; this is impossible to acquire a weak target in a strong clutter. So, it is necessary to check the notches of the real-time SBLF after the coefficients are calculated, although the real-time frequency response is close to the analytic one when an enough large average number of estimating the coefficients is selected.

In non-real-time MTI processing, the given frequency response notches are determined by analytic calculation, e.g., the conventional canceller notches, and they are not optimized for application. In state-of-the-art adaptive processing, the coefficients and notches are both calculated in real-time instead of analytically. For example, we utilize two-class clutters and their parameters are: one, $P_g$ 55 dB, $F_g$ 12 Hz, $D_g$ 7.2 Hz, plus $P_w$ 20 dB, $F_w$ 240 Hz, $D_w$ 58 Hz; the other, $P_s$ 30 dB, $F_s$ 24 Hz, $D_s$ 48 Hz plus $P_w$ 40 dB, $F_w$ 240 Hz, $D_w$ 48 Hz. When the stagger PRI periods are designed, we use the different blind-frequency response criteria. 1) The UPRI, 1/720 s, is for comparison and its period code is 29 29 29 29 29 29 29; 2) one stagger PRI has the shallowest notch within 5 times the UPRF and the stagger code is 27 26 31 29 30 28 32; 3) the other stagger PRI has the shallowest notch from 5 to 12 times the UPRF and the code is 30 27 26 28 29 31 32; 4) the stagger PRI of the real-time filtering is the same as that in 2) and a large average number, 30 bins, of SBLF coefficients estimates is used. Figure 5 shows their frequency responses within a frequency range up to 8.64 kHz, equivalent to a flight of Mach 1.26; they are denoted by black, green, blue, and red curves, respectively. We observe that the UPRI performs badly with twelve deep notches, lower than −30 dB at multiples of the UPRF; we cannot acquire any target dropping into such a notch, even if it is non-weak, e.g., 30 dB; the first stagger PRI has the deepest notch of 0.97 dB at 4.9 kHz, much shallower than the notches behind 5 UPRF; the second stagger PRI has the deepest notch of 0.6 dB at 8.1 kHz, much shallower than notches in front of 5 UPRF; the real-time stagger PRI has the deepest notch of 1.57 dB at 1.3 kHz, with much shallower notches behind 5 UPRF. The three stagger PRIs perform with much shallower notches than the UPRI does, so they can really alleviate improvement factor loss. The notches of real-time coefficients are similar to those of the analytic coefficients. Thus, in practical operation, providing multiple different conditionally-optimized stagger PRIs and checking the notches can improve the acquisition of weak target.

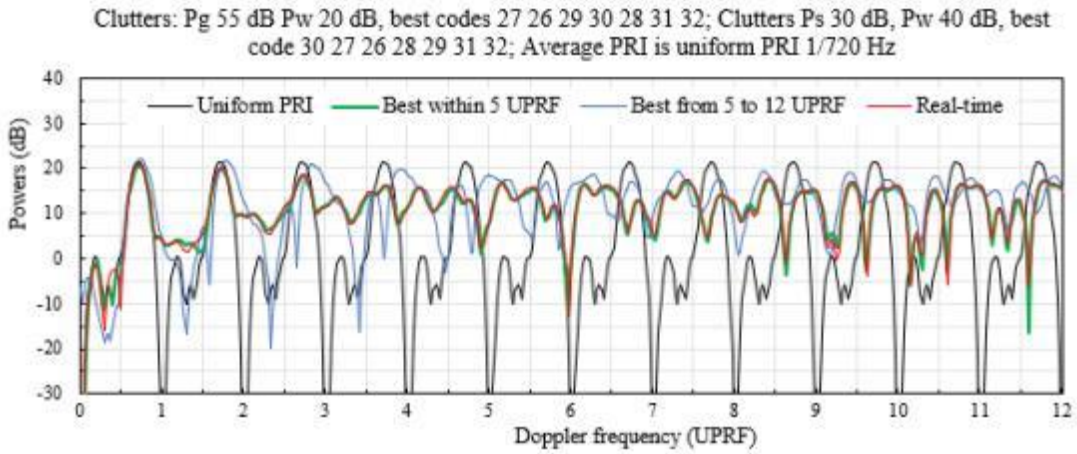


*Figure 5. Blind Doppler-frequency responses of the SBLF processor with uniform and stagger PRIs.*

## 4. Strategies of Multi-technology MTI Processor

In the multi-technology MTI processor, the adaptive filtering, pulse compression, and stagger PRI operate together; furthermore, the clutter returns may be heterogeneous because the environment is actually a combination of discrete, varying territories. So, how all the collected prior knowledge is applied to the processor refers to specific reasoning operations. For example, in the entire processing, when the SBLF needs to be enabled, how the processor realizes the training block, how the target block is identified, how the PC target terrace is declared, etc. are related to specific strategies. Under the prerequisites: 1) involvement with the SBLF and phase-coded PC, 2) the two-dimensional return arrays of range x azimuth, 3) the clutter backgrounds of mountain, clouds, sea, and rain, 4) the SBLF placed ahead of PC, we propose five heuristic strategies as follows.

### 4.1. Non-clutter Input Decision and Adaptive Threshold Set-up

This strategy works in Test block first. When the input power is lower than 10 dB, the input is regarded as non-clutter input. The SBLF is disabled in Test block and the input data directly enters the pulse compressor. It is possible that a target

is present in the low-level input. However, 1) if the SBLF is enabled, it may lead to an SCR loss due to frequency response notches; 2) generally, the target-viewable factor is SCR 8 dB at the screen display and the value can be reached when the PC is enabled.

Secondly, given a complex input $x_c(t_n, l)$, $n \in \{0, 1, \cdots, N_a - 1\}$, $N_a$ number of azimuth data and $l \in \{1, 2, \cdots, L_b\}$, $L_b$ length of a block (number of bins); $\{t_n\}$, the CPI time of a range gate, are stagger. After filtering-self in Estimate, Test, and Test I blocks, the outputs $y_c(t_n, l)$ of the blocks are used to set up a detection threshold. The output magnitude average of Test block can be calculated by

$$O_T=\sum_{l=1}^{L_b}\sum_{n=0}^{N_a-1}[y_c(t_n, l)\, y_c^*(t_n, l)]^{1/2}/N_a/L_b. \quad (6)$$

After three-block averaged magnitudes $O_E$, $O_T$, and $O_{TI}$ are obtained from (6), we average them again in terms of (5). Then the output threshold $T_D$ and adaptive threshold $T_{Da}$ are obtained. In heterogeneous clutter, using (5) to set up the detection threshold is still effective because the environment can be locally homogeneous; plus, a larger average number of the SBLF coefficient estimate may be required, e.g., $N_a$=30. The threshold factor, $F_T$, is selected to be alternative and the detection sensitivity may drop, but the false alarm rate can be controlled. In fact, in an entire antenna scan, the $F_T$ has to be assigned as multiple different values, depending on the clutter distribution of the radar site.

## 4.2. PC Target-terrace Identification and Declaration

This strategy works in five blocks: Estimate, Test, Test I, Test II, and potential Test III of Figure 1. When the SBLF suppresses a clutter and detects a target, the SBLF coefficients have to characterize the clutter only; otherwise, the SBLF can filter out both clutter and target. At the SBLF input end, the weak target is often submerged and is no way recognized; however, after delay-filtering, the powers of the target and clutter are changed inversely and the target is easily identified. From the experiment in Figure 4, we know that the block-delay filtering output reveals the target cues; when we use the coefficients from Estimate block with training data to filter the data of Test block, if a step-like waveform appears at its output, the block is called the target block and its index is set as 1; inversely, the block without a step-like output is called the clutter block and its index is set as 0. When the adjacent blocks, Test, Test I, Test II, and potential Test III are the target blocks, the combined step-like waveforms look like a terrace, called the target terrace. In machine learning, the processor compares the output averages of three or four blocks with the adaptive threshold and records the indices. From Figure 4, we also observe that when the weak target of5 dB is present, the terrace waveform is irregular due to clutter corruption, but when the non-weak target of 30 dB is present, the target terrace is regular. For example of identifying the target-terrace, in the case of a 63 bits phase-code PC, when the sum, $N_{ai}$, of the adjacent blocks' indices is equal to or close to Mod ($L_{pc}/L_b$) +1, $L_{pc}$ PC length, the PC target presence is a large probability and the target terrace is declared, its index is set as 1. When $N_{ai}$ is one or two, the PC target presence is a small probability and the index is set as 0. After the target terrace is declared, the block following the terrace block is regarded as a Guard block (where a target is probably present) and its data do not enter Estimate block. This strategy is reliable in a homogeneous clutter due to stable statistics.

In a heterogeneous clutter, a strong clutter in a block may severely corrupt a target inside, or there may be a much higher level than clutters in neighboring blocks. Then, the target may be missing, or the clutter may be declared as a target.

## 4.3. Estimation of SBLF Coefficients

The SBLF is a class of real-time adaptive filtering, including estimating and filtering; plus, it has a low computation load and low sensitivity of operation word-length. For the details of SBLF structure, design, and optimization, see reference [16] and Section 5.1.1 below. With Estimate block, we calculate the SBLF coefficients to reflect the clutter characteristics of Test block. When Test block has been declared as target index 0 or a block following the assigned Guard block, its data are recognized as the training data and moved into Estimate block. As long as the target index is 1 in Test block, the block data are not the target-free data, and the current SBLF coefficients from Estimate block are maintained until the terrace declaration ends, i.e., $N_{ai}$ is equal to 4 or 3. The $N_{ai}$ value is also regarded as the adaptive dwelling time since they do not need calculating the SBLF coefficients.

In the case of homogeneous clutter, statistics of blocks' data are even. In this paper, the block length is also the average number to estimate SBLF coefficients. The longer the block length is, the higher the accuracy of the coefficient estimates is, and the shallower the blind-frequency notches are. In the case of heterogeneous clutter, the boundary transition from non-clutter to clutter or from strong to weak levels, and the inverse transition, has varying clutter statistics. The coefficient estimates of a longer block are relatively stable but may not reflect the statistics of its partial data or its neighboring blocks. According to our experiments, the block length is selected as 20 to 30 bins.

## 4.4. Establishment of Clutter-map

The clutter-map is a database that stores all information collected in the reasoning enter into the corresponding map cell to be invoked for reasoning, specifically the following five-class information. A point $Z_m(l, t_n)$ corresponding to a range bin x an azimuth bin of the return array forms a cell of the clutter-map. All the stored data are updated by the reasoning center in the next scan.

1) The clutter levels, which are average magnitudes of the

inputs of Test block and the outputs of filtering-self with Estimate, Test, and Test I blocks, no matter whether a target is present inside.

2) The detection thresholds, which contain the output threshold and adaptive threshold from Test block.
3) The SBLF coefficients, which are calculated in Estimate block and used in Test, Test I, Test II, and potential Test III blocks. The parameters of the SBLF blind-frequency notch are also stored in the corresponding map cell.
4) The target-block and -terrace indices, which are set in Test, Test I, Test II, and potential Test III blocks after block-delay and terrace-delay filtering, and stored in the cells corresponding to the blocks.
5) The PC output impulse level, which is recorded when the SCR is higher than 8 dB, and stored in the map cell.

### 4.5. Utilization of Multiple Stagger PRI

This strategy requires designing multiple PRIs, whose shallowest blind-frequency responses cover the individual regions and which all achieve the maximized improvement factors. For example, one stagger PRI suits detecting targets lower than Mach 0.5, and the other stagger PRI suits targets higher than Mach 0.5. The SBLF is compatible with the stagger PRI technology [16]. As soon as the SBLF coefficients are calculated, the blind-frequency notch parameters are also determined, see Figure 5. Even if the best stagger PRI is designed, there are still multiple shallow notches, which also degrade weak target acquisition. When the machine learns and tracks, if a target is dropping into a notch of −15 dB depth, it can make an alert and agilely assign the other stagger PRI that has a different blind-frequency response. Then, the Doppler frequency of the target may drop into a shallower notch or be outside of notches, depending on the current clutter situation. If the new notch depth is −5 dB or shallower, the improvement factor will increase by 10 dB or larger.

In modern war, electronic countermeasures are unavoidable. Hostile air reconnaissance always aims at a measuring radar to steal parameters [20], including PRI; if the parameters are intercepted, the radar is easily attacked and then destroyed. Agile stagger PRIs can decrease the risk of being measured and attacked [22-46].

Except for the above strategies, the processor also contains a managing strategy, which is a manager to control the order of carrying out the strategies.

## 5. Simulation and Performances of the Multi-technology Processor

### 5.1. Simulation Method

In order to examine the effectiveness of the multi-technology MTI processor, we select several clutter models reflecting S-band ship-borne radar environments [2, 7, 21] and make simulation experiments [4]. The generated clutters have two models: one is forested mountain plus windblown clouds, representing homogeneous strong clutter, and the other is surging sea plus heavy rain, representing heterogeneous severe clutter [2]. Two weak targets of high speeds are selected and their Doppler parameters are listed at the tops of Figures 8 and 9. Two codes for stagger PRIs are optimized under the individual blind-frequency conditions. The stagger code 27 26 31 29 30 28 32 for the homogeneous clutter has the best blind-frequency response within 5 times the UPRF (720 Hz). The other stagger code 30 27 26 28 29 31 32 for the heterogeneous clutter has the best blind-frequency response from 5 to 12 times the UPRF. Their stagger ratio is 1.23 and positive/negative blind-speed extension is 14.5. For the method of searching the best stagger code, see [19]. The generated return arrays have 240 range bins x 7 azimuth data. The adaptive threshold is calculated in terms of (5). White noise power is normalized to one, i.e., 0 dB.

#### 5.1.1. Stagger-PRI Adaptive Filtering

The adaptive filtering for the MTI processor is required to operate in real-time and be compatible with stagger PRI, so, we select the SBLF algorithm [16], which calculates the coefficients using block data of range x azimuth from the stagger-PRI returns and achieves the optimal improvement factor as the lattice predictor of UPRI does. Figure 6 shows an M-order SBLF structure. $R^f_m(t_n)$ and $R^b_m(t_n)$, $n \in \{1, 2, \cdots, M\}$, are the forward and backward reflection coefficients, respectively, of the mth (1 to M) stage, at $t_n$, observation time; $T_m(t_n)$ is delay time of the mth stage, and $e^b_m(t_n)$ and $e^f_m(t_n)$ are the backward and forward prediction errors, respectively, of the mth stage. In order to be compatible with stagger-PRI, the reflection coefficients of the lattice predictor must be time-variant with $t_n$. A staggered lattice predictor of order M has ∑M forward and backward reflection coefficients, ∑M sum of the integers 1, 2, ⋯, M. In comparison, the coefficients of a UPRI lattice predictor are time-invariant; the UPRI lattice predictor of order M has M forward and backward reflection coefficients.

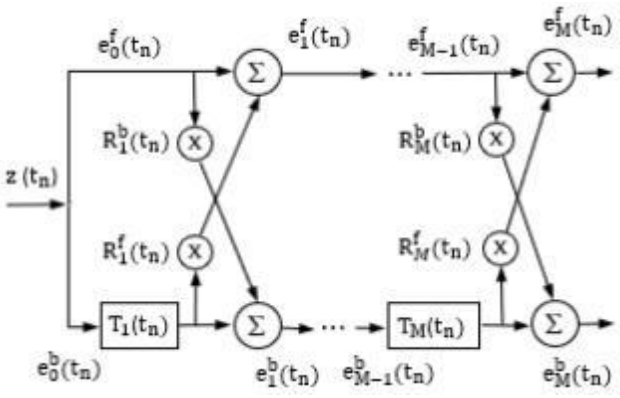


***Figure 6.*** *A stagger-period lattice predicter of order M.*

Given that block data of a return array are denoted by $x_c(t_n, l)$, $n \in \{0, 1, \cdots, N_a - 1\}$, $N_a$ number of azimuth data, $l \in \{1, 2, \cdots, L_b\}$, $L_b$ the block length, the mth stage prediction errors at time $t_n$ are computed as

$$e_m^f(t_n, l) = e_{m-1}^f(t_n, l) + R_m^f(t_n) e_{m-1}^b(t_{n-1}, l),\ m \in \{1, 2, \cdots, n\},\ n \in \{1, 2, \cdots, N_a - 1\}, \quad (7)$$

$$e_m^b(t_n, l) = e_{m-1}^b(t_{n-1}, l) + R_m^b(t_n) e_{m-1}^f(t_n, l),\ m \in \{1, 2, \cdots, n\},\ n \in \{1, 2, \cdots, N_a - 1\}, \quad (8)$$

$$e_0^f(t_n, l) = e_0^b(t_n, l) = x_c(t_n, l),\ n \in \{0, 1, 2, \cdots, N_a - 1\},\ l \in \{1, 2, \cdots, L_b\}. \quad (9)$$

In practical computation, we use time average estimation of the reflection coefficients to substitute for the ensemble mean, i.e., $R_m^b(t_n)$ and $R_m^f(t_n)$ of the staggered block lattice are estimated by

$$\hat{R}_m^f(t_n) = \frac{-\sum_{l=1}^{L_b} e_{m-1}^f(t_n; l)\, e_{m-1}^{b*}(t_{n-1}; l)}{\sum_{l=1}^{L_b} [(1-\alpha)|e_{m-1}^f(t_n; l)|^2 + \alpha |e_{m-1}^b(t_{n-1}; l)|^2]},\ m \in \{1, 2, \cdots, n\},\ n \in \{1, 2, \cdots, N_a - 1\}, \quad (10)$$

$$\hat{R}_m^b(t_n) = \frac{-\sum_{l=1}^{L_b} e_{m-1}^{f*}(t_n; l)\, e_{m-1}^b(t_{n-1}; l)}{\sum_{l=1}^{L_b} [\alpha |e_{m-1}^f(t_n; l)|^2 + (1-\alpha)|e_{m-1}^b(t_{n-1}; l)|^2]},\ m \in \{1, 2, \cdots, n\},\ n \in \{1, 2, \cdots, N_a - 1\}, \quad (11)$$

where α is the balance factor, taking a value from 0.5 to 1. The larger the α is, the faster the convergence of estimates (10) and (11) is; we take 0.99 in all the experiments.

### 5.1.2. Phase-code Pulse Compression

The large time-width PC technology is involved; its compatibility with the SBLF is not an issue when it is placed behind the SBLF, see Section 2. Although the entire PC pulse has a large time width, each sub-pulse of it has a narrow time-width, called an element. The complex envelope of the phase-code waveform is expressed as

$$x_p(\tau) = A_c \sum_{i=1}^{L_{pc}} C_p(i)\, \mathrm{Rect}\left[\frac{\tau - (i-0.5)\Delta}{\Delta}\right],\ 0 \le \tau \le L_{pc}\Delta, \quad (12)$$

where $A_c$ is a complex constant; $C_p(i)$, $i \in \{1, 2, \cdots, L_{pc}\}$, is the code element, $L_{pc}$ the code length; Δ is the sub-pulse width, which determines range resolution, and Rect [] is a rectangle function. The selected phase-code waveform is a minimum sidelobe bi-phase code, whose hexadecimal format is 04CF5A2471657C6F, with a peak sidelobe of 4 (-23.9 dB) [15]. Figure 7 shows the autocorrelation function of the phase code. When matched-filtering, the compression filter, i.e., this code, matches the target return and does not match the clutter return, and outputs a main lobe of 63. The target enhancement is 36 dB and the clutter is magnified by about 18 dB.

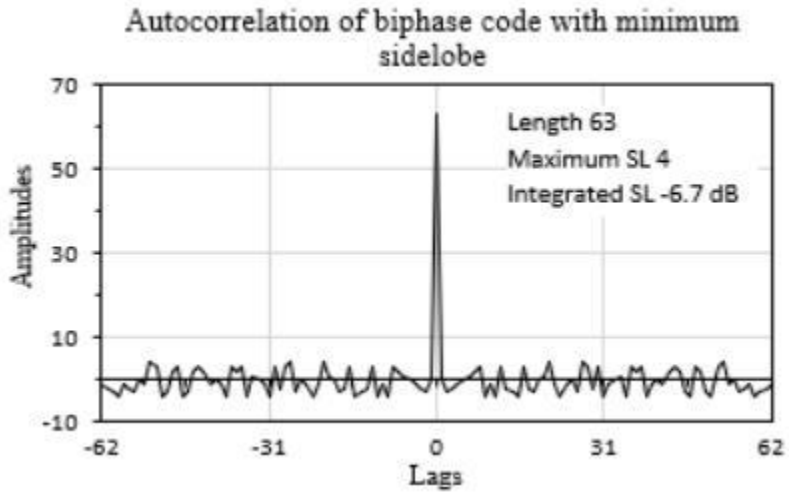


***Figure 7.*** *Autocorrelation of a minimum sidelobe phase code of length 63.*

## 5.2. Experiment in Homogeneous Clutter

The simulation experiment examines weak target acquisition in the homogeneous strong clutter. The forested mountain clutter is of $P_g$ 60 dB, $F_g$ 12 Hz, $D_g$ 7.2 Hz, plus the moving cloud clutter, $P_w$ 10 dB, $F_w$ 240 Hz, $D_w$ 58 Hz. Two weak targets have SCRs −55 dB and −50 dB individually, and the radial speed, Mach 0.54. The return array is segmented into 12 blocks of length 20 bins; the target-terrace declaration is with $N_{ai}$=3 or 4. The adaptive threshold factor is $F_T$=1.85. Figure 8 shows target acquisition of the MTI processor. The input clutter curve, black, completely submerges the targets. The SBLF output curve, blue, reveals irregular and regular PC target-terraces, indicating that the clutter has been filtered out, which are of length 63 bins, from 61 to 123, about 20 dB height, and from 171 to 233, about 24 dB height, respectively; the clutter at both sides of the terraces is fluctuating around 15 dB, then the clutter is suppressed by about 45 dB. On the PC output curve, red, two target impulses appear in place, about 52.9 dB and 58.7 dB high, respectively. The residual clutter at both sides of the impulses is fluctuating around 35 dB, thus, the target-to-clutter ratios of the PC output are about 17.9 dB and 23.7 dB individually, and such target dots will be clearly visible on the radar screen display. The excellent performance of the MTI processor in the homogeneous clutter results from three aspects: the real-time SBLF reaches optimal improvement factor, the target Doppler frequency does not drop into a blind-frequency notch, and the heuristic strategies effectively control the reasoning operations.

## 5.3. Experiment in Heterogeneous Clutter

For the simulation experiment in heterogeneous clutter, we generate alternative clutter return: the surging sea clutter is of $P_s$ 35 dB, $F_s$ 24 Hz, $D_s$ 48 Hz, and the heavy rain clutter is of varying $P_w$ from 20 dB to 45 dB, $F_w$ 240 Hz, $D_w$ 48 Hz. Varying power is a common nature of heterogeneous clutter. Two weak targets have about SCRs −30 dB and −35 dB individually and the radial speed, Mach 1.24. The return array is segmented into 8 blocks of length 30 bins; the target-terrace declaration is with $N_{ai}$=2 or 3. The adaptive threshold factor $F_T$=1.6. Figure 9 shows the target acquisition of the MTI processor. We observe that the input power curve, black, is fluctuating around from 35 dB to 45 dB due to the mix of two sub-clutters, and submerges the two targets. The SBLF output curve, blue, hides two irregular target-terraces, one from bins 61 to 123, fluctuating around 28 dB, and the other from bins 171 to 233, fluctuating around 35 dB; the clutter at both sides of the terraces is fluctuating around the same level, and the input clutter is suppressed by about only 10 dB due to the severe wide-band clutter. The PC output curve, red, reveals two target impulses in place, about 58 dB and 60.7 dB high individually; the residual clutter at both sides of the impulses is fluctuating around 49 dB and 53 dB, respectively. Thus, the target-to-clutter ratios are about 9 dB and 7.7 dB, and such target dots can also be visible on the radar screen display. Note: the two PC terraces are much corrupted, but the processor still identifies the hidden target terrace with the $N_{ai}$=3. In the severe heterogeneous clutter plus the weak targets, the processor still acquires the weak targets, although the result is not quite satisfactory; through examining, the SBLF coefficients achieve optimal improvement factor and the target Doppler frequency does not drop into the blind-frequency notch; the severe rain clutter has varying power and wide frequency band, but its max power is 45 dB, not very strong.

In a practical MTI radar, involvement of the new technologies will create a larger challenge than what we face here; this requires us to dig out more prior knowledge to improve them.

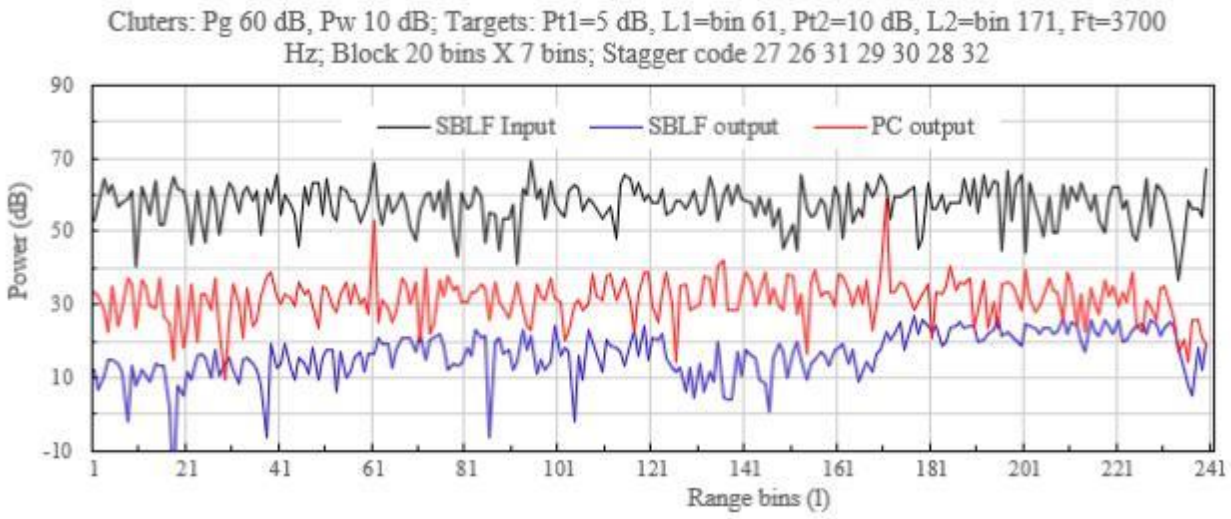


***Figure 8.*** *Weak target acquisition of the MTI processor in strong mountain clutter plus moving cloud clutter.*

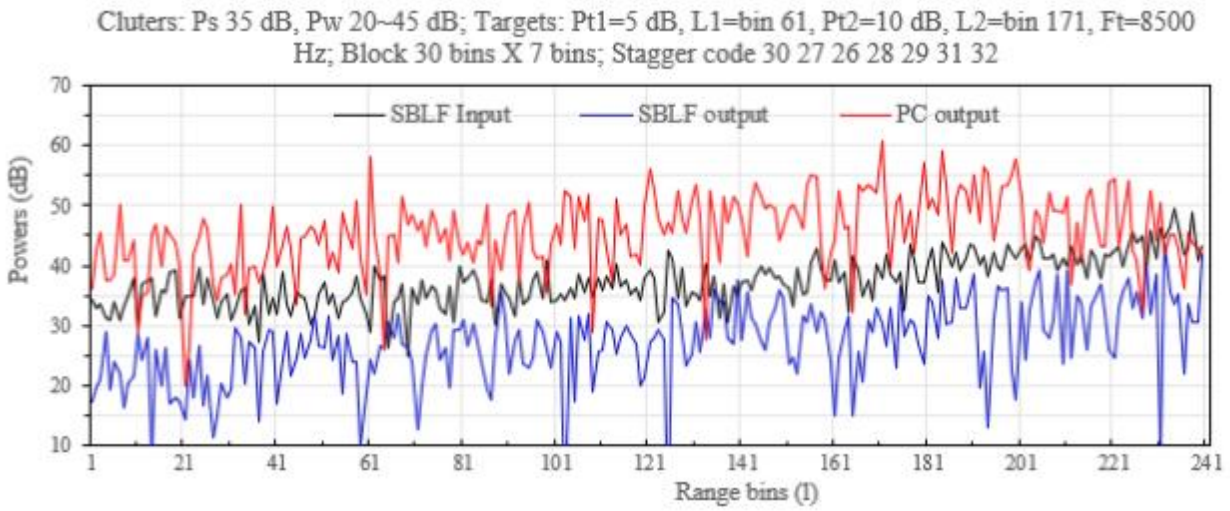


***Figure 9.*** *Weak target acquisition of the MTI processor in sea clutter plus varying rain clutter.*

## 6. Conclusion

The real-time SBLF is the core of the clutter suppression algorithm and the reasoning center is the core of the intelligent operation; the combination of both comprises a foundation of the intelligent multi-technology MTI processor. Through many experiments on target acquisition in the homogeneous and heterogeneous clutters, the processor without artificial intelligence is verified to not meet the performance requirements of a today's MTI radar, and the approach of using only optimized algorithm has become outdated.

1) In a conventional MTI processor, the PC is always placed ahead of MTI filtering. Since the MTI canceller/filter is a linear operation over azimuth data and the PC is a linear operation along range bins, the overall performance of both technologies is independent of their placement order. We give a mathematical deduction to prove this point. However, in a state-of-the-art MTI radar, the MTI filtering can be adaptive and its weight coefficients are calculated in real-time. In this case, we reinvestigate the placement order. For adaptive filtering, the training data neighboring the target block need to be ascertained. If the PC is placed ahead, its output forms a target main lobe and quite wide range sidelobes around it; we can hardly find the target-free data to calculate the coefficients. Conversely, if the adaptive filtering is placed ahead, the data neighboring are not corrupted by a target and the estimated coefficients will filter out the clutter only. Our deduction and experiments affirm that in the case of adaptive filtering, the overall performance of both technologies is related to the placement order; it is much better when the adaptive filtering is placed ahead of the PC.
2) An early MTI processor only utilized an arithmetic computation but a today's MTI processor involves multiple advanced technologies, such as adaptive filtering, large time-wide PC, etc. In order to combine the delay-filtering mode of adaptive filtering with large time-width PC, the reasoning center of the intelligent operation of five data blocks is proposed. The filtering-self outputs of data blocks are suitable to set up a detection threshold. Since the valid placement order is the adaptive filtering ahead of the PC and the SBLF is employed, a novel reasoning rule to identify PC target terrace is proposed and refers to the PC time-width; when three or four adjacent target blocks are declared, the PC target-terrace presence is a large probability. For large blind-speed expanding, e.g., 30, it is impossible for the blind-frequency response to have very shallow notches. We propose partial-region-optimized blind-frequency responses, which have the shallowest notches in low-speed or middle-speed or high-speed flight regions, and by the cooperation of multiple stagger PRIs, the entire extended frequency response performs the shallowest notch.
3) Through many simulation experiments, we propose five heuristic strategies for the MTI processor. 1) Non-clutter input decision and adaptive threshold set-up, which decides if Test block input is weak enough to disable SBLF and to enter PC directly; the filtering-self outputs of several blocks are used to set up the detection threshold. 2) PC target-terrace identification and declaration, which creates the rule for identifying a target block, declaring PC target-terrace, and setting PC target-terrace index. 3) SBLF coefficient estimation, which calculates the SBLF coefficients in real-time in Estimate block and loads them to Test block. 4) Establishment of clutter-map, which stores the adaptive thresholds, target block indices, target-terrace indices, PC output impulse levels, and parameters of blind-frequency notches into the corresponding clutter-map cells. 5) Utilization of multiple stagger PRIs, which employs multiple stag-

ger-PRIs and examines their responses to prevent the target from dropping into a deep notch. The max positive/negative blind-speed extension of the stagger PRIs is 14.5.

4) In our virification experiments, the strong and severe clutters are selected to reflect forested mountain plus windblown clouds and surging sea plus varying heavy rain, respectively. The stagger PRIs for the two-class clutters are the best for individual blind-speed responses. The SBLF and minimum sidelobe 63-bit phase-code PC are involved. The return arrays of 240 range bins x 7 azimuth data are segmented into 12 and 8 blocks for the individual experiments. Five heuristic strategies are applied. In the strong clutter plus weak targets of SCRs −50 dB, −55 dB, the SBLF output illustrates regular target-terraces, and two target impulses appear at the PC output, having SCRs 17.9 dB and 23.7 dB, respectively. Such excellent performance cannot be reached without the intelligent operations. In the severe clutter plus weak targets of SCRs −30 dB, −35 dB, the SBLF output hides irregular target-terraces but the PC output exhibits the target impulses in place, having SCRs 9 dB and 7.7 dB, respectively. The target acquisitions of such SCRs are maximized but not satisfactory yet. Thus, it is necessary for us to continue researching the weak target acquisition under the severe heterogeneous condition.

## Abbreviations

| | |
|---|---|
| CPI | Coherent Processing Interval |
| MTI | Moving Target Indication |
| PC | Pulse Compression |
| PRF | Pulse Repetition Frequency |
| PRI | Pulse Repetition Interval |
| RCS | Radar-cross-section |
| SBLF | Stagger-block-lattice-filtering |
| SCR | Signal-to-clutter ratio |
| UPRF | Uniform PRF |
| UPRI | Uniform PRI |

## Acknowledgments

I would like to thank Prof. Zeng Bao for his proposing this project and directing this study at Xi'an University of Electronic Science and Technology.

## Author Contributions

Xubao Zhang is the sole author. The author read and approved the final manuscript.

## Conflicts of Interest

This author declares no any conflicts of interest.

## References

[1] Wikipedia. Advanced Technology Demonstrator. Available from: https://en.wikipedia.org/wiki/Advanced_Technology_Demonstrator (accessed 19 July 2025).

[2] Robert M. O'Donnell. Introduction to Radar Systems- Lecture 7- Radar Clutter and Chaff; Part 1. Available from: https://www.youtube.com/- watch?v=XFapyIIzX_8 (accessed 19 July 2025).

[3] Wenxu Zhang, Shudi Ma, Qiuying Du. Optimization of Adaptive MTI Filter. Int. J. Communications, Network and System Sciences. 2017, 10, 206-217.

[4] Dingqing Lu. Simulation, Verification of Pulse Doppler Radar Systems Application Note. USA. 2010. Available from: www.agilent.com/find/radar (accessed 19 July 2025).

[5] Jie Song, Wei Xiong, Xiaolong Chen, Yuan Lu. Experimental Study of Maritime Moving Target Detection Using Hitchhiking Bistatic Radar Note. Remote Sensing. 2022, 14, 3611. https://doi.org/10.3390/rs14153611

[6] Simon Haykin, Adaptive filter theory. Fifth Edition. London, UK: Pearson Education Limited, Edinburgh Gate, 2014, pp. 150-208.

[7] Robert E. Thurber. Advanced Signal Processing Techniques for the Detection of Surface Targets. Johns Hopkins University Applied Physics Laboratory. 1983; 4(4), 285-295. https://secwww.jhuapl.edu

[8] James H. Gaby, Monson H. Hayes. Artificial Intelligence Applied to Spectrum Estimation. The 1984 International Conference on Acoustics, Speech, and Signal Processing, San Diego, USA, 1984; pp. 13.5.1-13.5.4.

[9] Wanda S. Gass, Richard T. Tarrant, B. I. Pawate, Michele Gammel, P. K. Rajasekaran, Richard H. Wiggins, C. David Covington. Multiple Digital Signal Processor Environment for Intelligent Signal Processing. Proceedings of the IEEE. 1987, 75(9), pp. 1246-1259.

[10] Wicks, M. C., Melvin, W. L., Chen, P. An Efficient Architecture for Non-Homogeneity Detection in Space-Time Adaptive Processing Airborne Early Warning Radar. Proceedings of the 1997 International Radar Conference, Edinburgh, UK, 1997; pp. 295-299.

[11] William l. Melvin, Joseph R. Guerci. Knowledge-Aided Signal Processing: A New Paradigm for Radar and Other Advanced Sensors. IEEE Transactions on Aerospace and Electronic Systems. 2006, 42(3), 938-996.

[12] Michael C. Wicks, Muralidhar Rangaswamy, Raviraj Adve, Todd B. Hale. Space-Time Adaptive Processing. [A knowledge-based perspective for airborne radar]. IEEE Signal Processing Magazine. 2006, Jan., 51-65.

[13] Raviraj S. Adve, Michael C. Wicks, Todd B. Hale, Paul Antonik. Ground Moving Target Indication Using Knowledge Based Space Time Adaptive Processing. Record of the IEEE 2000 International Radar Conference, VA, USA, 2000; pp. 735-740.

[14] Chris Allen. Radar Pulse Compression 2004. Available from: https://www.mathworks.com›help›signal (accessed 19 July 2025).

[15] Greg C, Jon R. Efficient Exhaustive Search for Optimal-Peak-Sidelobe Binary Codes. IEEE TRANS. on AES. 2005, 41(1), 302-308.

[16] Xubao Zhang. Lattice Predictors for Stagger-Period Sequence: their Theory and Application. Science PG Journal of Electrical and Electronic Engineering. 2022, 10(5), 184-198.

[17] Xubao Zhang. New Methods of Simulating Radar Clutter Return Arrays. EJECE, European Journal of Engineering and Computer Science. 2023, 7(6), 46-57.

[18] Jun Liu, Filippo Biondi, Danilo Orlando, Alfonso Farina. Training Data Classification Algorithms for Radar Applications. IEEE Signal Processing Letters. 2019, 26 (10), 1446-1450.

[19] Li, Qingyang. "Dynamic Adaptive Attention and Supervised Contrastive Learning: A Novel Hybrid Framework for Text Sentiment Classification." arXiv preprint arXiv:2604.10459 (2026).

[20] Mo, Zengxian. "Development of an Intelligent Retrieval and Recommendation System for Chinese Language Educational Resources Based on DNN Technology." 2025 2nd International Conference on Intelligent Computing and Robotics (ICICR). IEEE, 2025.

[21] Pei J, Frascolla V, Al-Dulaimi A, et al. Distributed large models training optimization with real-time wireless channel feedback[J]. IEEE Journal on Selected Areas in Communications, 2025.

[22] Xiong, Zhexiao, et al. "Groundingbooth: Grounding text-to-image customization." arXiv preprint arXiv:2409.08520 (2024).

[23] Jia, Zexi, et al. "Event-based semantic segmentation with posterior attention." IEEE Transactions on Image Processing 32 (2023): 1829-1842.

[24] Wang, Haozhe, et al. "Reverse-engineered reasoning for open-ended generation." arXiv preprint arXiv:2509.06160 (2025).

[25] Huang, Chuanwei, et al. "ArtFRD: A Fisher-Rao Mixture Metric for Generative Model Aesthetic Evaluation." Proceedings of the 33rd ACM International Conference on Multimedia. 2025.

[26] Pu, Zhichen, et al. "Dual roles of IL-18 in colitis through regulation of the function and quantity of goblet cells." International journal of molecular medicine 43.6 (2019): 2291-2302.

[27] Liu, Lijun. “Research on Intelligent Generation Algorithm of Interface Icon Based on Diffusion Model ”. International Scientific Technical and Economic Research , vol. 4, no. 1, Mar. 2026, pp. 149-67, https://doi.org/10.71451/ISTAER2607.

[28] Bart, Steven, et al. "Frontier ai safety confidence evaluate." (2025).

[29] Jia, Zexi, et al. "Secret lies in color: Enhancing ai-generated images detection with color distribution analysis." Proceedings of the Computer Vision and Pattern Recognition Conference. 2025.

[30] Su, Junhao, et al. "Failure makes the agent stronger: Enhancing accuracy through structured reflection for reliable tool interactions." arXiv preprint arXiv:2509.18847 (2025).

[31] Shen, Yichen, et al. "A rank-based sampling framework for offline reinforcement learning." 2021 IEEE International Conference on Computer Science, Electronic Information Engineering and Intelligent Control Technology (CEI). IEEE, 2021.

[32] Cheng L, Gu Y, Liu Q, et al. Advancements in accelerating deep neural network inference on AIoT devices: A survey[J]. IEEE Transactions on Sustainable Computing, 2024, 9(6): 830 -847.

[33] Hu, Xing, and René Caldentey. "Trust and reciprocity in firms' capacity sharing." Manufacturing & Service Operations Management 25.4 (2023): 1436-1450.

[34] Li, Yu, and Xinzi Ma. “Intelligent Delineation Algorithm of Urban Development Boundary Based on Graph Neural Network ”. International Scientific Technical and Economic Research , vol. 4, no. 1, Mar. 2026, pp. 123-48, https://doi.org/10.71451/ISTAER2606.

[35] Zhu, Taozheng, and Qiqi Lei. "PEMM-REC: A Prompt Engineering-Based Semantic Alignment Framework for Large Multimodal Model Recommender System." 2025 3rd International Conference on Artificial Intelligence and Automation Control (AIAC). IEEE, 2025.

[36] Jia, Zexi, et al. "A visual leap in clip compositionality reasoning through generation of counterfactual sets." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2025.

[37] Pei J, Dai M, Prasad R R V, et al. FL Meets LLM: A Hybrid Security Framework for the Internet of Energy[J]. IEEE Network, 2025.

[38] Hu, Yibing, and Peng Jiang. “ Multi-Stage Network Optimization Model and Decomposition Algorithm for Improving the Efficiency of Enterprise Production System under the Background of Digital Economy ”. International Scientific Technical and Economic Research , vol. 4, no. 1, Mar. 2026, pp. 168-87, https://doi.org/10.71451/ISTAER2608.

[39] Jin, Haopeng. "FreqFormer: Hierarchical Frequency-Domain Attention with Adaptive Spectral Routing for Long-Sequence Video Diffusion Transformers." arXiv preprint arXiv:2604.22808 (2026).

[40] Duan L, Yang H, Qi Q, Wu Q, Shao C, Jiang L. Seismic Fragility of Urban Rail Transport RC Solid Piers Considering Multiparameter Effects. Buildings. 2026; 16(12):2327. https://doi.org/10.3390/buildings16122327

[41] Chen, Guona. “ Research on Algorithm Improvement of ARIMA-LSTM Hybrid Model in Time Series Prediction of

Inflation Rate ” . International Scientific Technical and Economic Research , vol. 4, no. 1, Mar. 2026, pp. 90-122, https://doi.org/10.71451/ISTAER2605.

[42] Pei J, Liu W, Li J, et al. A review of federated learning methods in heterogeneous scenarios[J]. IEEE Transactions on Consumer Electronics, 2024, 70(3): 5983-5999.

[43] Jia, Zexi, et al. "Finger recovery transformer: Toward better incomplete fingerprint identification." IEEE Transactions on Information Forensics and Security 19 (2024): 8860-8874.

[44] Xubao Zhang. Finite Impulse Response Filters for Stagger-Period Signals, their Designs and Applications. European Journal of Information Technologies and Computer Science. 2021, 1(1), 1-11.

[45] International Defense Security & Technology, Inc. Real-Time Digital Signal Processing for Electronic Warfare Systems. Available from: https://idstch.com/technology/ict/real-time-digital-signal-processing-for-electronic-warfare-systems (accessed 19 July 2025).

[46] Capt. H. Subramaniam, Shipborne radar and ARPA. Third edition. Mumbai India: VIJAYA PUBLICATIONS; 2001, pp. 1-78.