

DETEKSI SINYAL TARGET DENGAN MENGGUNAKAN METODE ALGORITMA CONSTANT FALSE ALARM RATE (CFAR) PADA SEARCH RADAR

DETECTION OF SIGNAL TARGET WITH CONSTANT FALSE ALARM RATE (CFAR) ALGORITHM METHOD ON SEARCH RADAR

Farhan Ramadhan¹, Dharu Arseno², Fiky Yosef Suratman³

^{1,2}Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom

³Prodi S1 Teknik Elektro, Fakultas Teknik Elektro, Universitas Telkom

¹farhanramadhan@student.telkomuniversity.ac.id, ²dharuarseno@telkomuniversity.ac.id, ³fysuratman@telkomuniversity.ac.id

³fysuratman@telkomuniversity.ac.id

Abstrak

Penelitian di bidang teknologi radar saat ini sedang banyak dilakukan, termasuk di Indonesia. Akan tetapi banyak dari hasil penelitian tersebut hanya fokus pada subsistem transmisi, sedangkan pada subsistem *signal processing* jarang dilakukan. Subsistem *signal processing* sangat penting dalam hal melakukan proses deteksi, yang merupakan fungsi utama dari teknologi radar. Banyak permasalahan deteksi yang dapat mengganggu kinerja dari radar. Salah satunya adalah *false alarm*. Permasalahan *false alarm* dapat diatasi dengan algoritma *Constant False Alarm Rate* (CFAR). Algoritma CFAR mengacu pada bentuk umum dari algoritma adaptif yang digunakan pada sistem radar untuk mendeteksi dengan latar belakang *noise* yang tidak bisa dihindari. Sehingga *false alarm* dapat diatasi menjadi konstan dan *resource* pada radar tidak akan terambil cukup banyak. Pada simulasi kali ini digunakan dua skenario simulasi, dengan masing-masing skenario menggunakan nilai parameter $P_{FA} = 10^{-1}$ sampai 10^{-4} , *Training Cell* = 100, *Guard Cell* = 20, dan SNR = 10.2 dB & 12.8 dB. Dari dua skenario simulasi yang telah dilakukan didapatkan bahwa detektor dengan algoritma CFAR, yaitu CA-CFAR dan GOCA-CFAR, dapat bekerja dengan baik saat dalam kondisi *noise* homogen dan heterogen. Algoritma CFAR dapat mendeteksi target dengan baik dan menghasilkan tingkat false alarm yang sesuai dengan nilai parameter yang diinginkan.

Kata Kunci : CFAR, Radar, False Alarm, Noise, Interferensi, Threshold

Abstract

*Research in the field of radar technology is currently being carried out, including in Indonesia. However, many of these research results only focus on the transmission subsystem, while the signal processing subsystem is rarely done. The signal processing subsystem is very important in terms of carrying out the detection process, which is the main function of radar technology. Many detection problems can interfere with the performance of the radar. One of them is a false alarm. False alarm problems can be solved with the Constant False Alarm Rate (CFAR) algorithm. The CFAR algorithm refers to a common form of adaptive algorithm used on a radar system to detect against an inevitable noise background. So that false alarms can be overcome to be constant and the resources on the radar will not be taken up enough. In this simulation, two simulation scenarios are used, with each scenario using the P_{FA} parameter value = 10^{-1} until 10^{-4} , *Training Cell* = 100, *Guard Cell* = 20, and SNR = 10.2 dB & 12.8 dB. From the two simulation scenarios that have been done, it is found that the detector with CFAR, CA-CFAR and GOCA-CFAR algorithms, can work well when in homogeneous and heterogeneous noise conditions. The CFAR algorithm can detect targets well and produce a false alarm level that matches the desired parameter value*

Keywords : CFAR, Radar, False Alarm, Noise, Interference Threshold

1. Pendahuluan

Perkembangan yang sangat pesat dari bidang teknologi, informasi, dan komunikasi, terutama dalam bidang teknologi radar, membuat para peneliti di Indonesia akhir - akhir ini sedang giat dalam mengembangkan teknologi radar. seperti radar untuk keperluan militer, cuaca, transportasi, maupun keperluan dibidang lainnya. Radar memiliki banyak tipe yang digunakan saat ini, tergantung dari segi fungsi serta tujuannya. Fungsi dari berbagai macam radar yang digunakan saat ini dapat dibedakan menjadi 3 fungsi dasar, yaitu (1) *Search/Detect Radar*; (2) *Track Radar*; dan (3) *Imaging Radar* [1]. Masing – masing mempunyai keahlian dan fungsi yang berbeda. Untuk *Search Radar* berfungsi hanya untuk mendeteksi dan mengetahui posisi target yang ada disekitar. *Tracking Radar* berfungsi untuk melacak sekaligus mengetahui karakteristik dari target. Dan *Imaging Radar* berfungsi untuk membuat gambar atau citra 2 Dimensi, biasanya untuk citra daratan salah satu contohnya adalah *Ground Penetrating Radar* (GPR).

Saat melakukan deteksi target, proses pengolahan sinyal pada sistem radar sangatlah penting. Karena hal itu merupakan fungsi utama dari teknologi radar. Banyak permasalahan deteksi yang dapat mengganggu

kinerja dari radar. Salah satunya adalah *false alarm*. *False alarm* dapat terjadi jika saat proses deteksi tidak terdapat sinyal *echo* tetapi hanya terdapat sinyal interferensi dengan daya cukup sehingga menghasilkan harga tes statistik lebih besar dari *threshold* yang sudah ditentukan [1]. Kesalahan ini sering terjadi pada radar, terutama pada sistem *search* radar yang berfungsi untuk mendeteksi target dan mengetahui posisi dari target.

Tugas akhir ini dilakukan untuk mengatasi permasalahan yang sering terjadi seperti kasus diatas dengan menggunakan metode algoritma deteksi *Constant False Alarm Rate* (CFAR). Metode CFAR ini bekerja dengan cara memperkirakan statistik interferensi dari pengukuran radar serta melacak perubahan pada interferensi untuk menyesuaikan *threshold* yang nantinya bertujuan untuk mempertahankan probabilitas *false alarm* yang konstan agar kinerja pada radar semakin optimal dalam mendeteksi target dan *resource* radar tidak terambil cukup besar karena tingkat *false alarm* yang tinggi.

2. Dasar Teori

2.1 Search Radar

Setiap tipe radar pada dasarnya memiliki tugas untuk mendeteksi target yang dicari, termasuk untuk radar dengan tipe *search* radar. Tugas utama *search* radar adalah untuk mencari dan mengetahui lokasi target yang ada disekitar dengan cara mengarahkan antena *transmitter* nya ke segala arah dan mengirimkan gelombang pulsa secara berulang.

2.2 Konsep Deteksi Threshold

Pada sebagian besar sistem radar modern, proses deteksi dilakukan secara otomatis pada prosesor sinyal. Hal ini dilakukan dengan menetapkan tingkatan *threshold* berdasarkan suatu gangguan, seperti *noise* dan *clutter*. Kemudian dari *threshold* tersebut akan dapat menentukan adanya target dengan membandingkan tingkat sinyal di setiap sel.

Deteksi sinyal pada sistem radar, memiliki 2 hipotesis dimana hipotesis ini berguna dan berpengaruh dalam melakukan tes statistik dari radar. Hipotesis ini dikenal dengan nama *Hypothesis Testing*. Berikut penjelasan dari 2 *Hypothesis Testing* [1]:

H0 : Tidak ada sinyal target, hanya terdapat interferensi atau *noise* saja

H1 : Ada sinyal target

Tapi perlu diperhatikan juga bahwa jika interferensi yang sangat kuat dapat melewati *threshold* dan bisa mengakibatkan *false alarm*.

2.3 Fluktuasi Target

Selain dari fakta bahwa sinyal *noise* yang berfluktuasi, sinyal target pun kenyataannya juga berfluktuasi. Peter Swerling mengembangkan seperangkat empat model statistik yang menjelaskan tentang empat kondisi fluktuasi target yang berbeda [2]. Keempat kasus tersebut meliputi dua model *Probability Density Function* (PDF) yaitu *Rayleigh* dan *4-th degree chi-square* dan dua tingkat fluktuasi (*scan-to-scan* dan *pulse-to-pulse*). Masing-masing diberi label Swerling 1, 2, 3, dan 4 (SW1, SW2, SW3, dan SW4) sedangkan untuk target yang tidak berfluktuasi disebut Swerling 0 (SW0) atau model target Marcum [1].

Berikut tabel penjelasan dari perbedaan masing – masing bentuk model Swerling berdasarkan model dekorelasi dan juga ada atau tidaknya penyebar yang dominan.

Tabel 1. Perbedaan Masing - masing Model Swerling [3]

Model Swerling	Deskripsi
Sweling 0 atau Swerling 5	Sinyal pulsa yang tidak berfluktuasi
Swerling 1	<ul style="list-style-type: none"> ➤ Dekorelasi <i>scan-to-scan</i> ➤ Model PDF <i>Reyleigh</i> / Eksponensial ($m = 1$) ➤ Menggunakan sejumlah penyebaran yang terdistribusi secara acak tanpa penyebar yang dominan
Swerling 2	<ul style="list-style-type: none"> ➤ Dekorelasi <i>pulse-to-pulse</i> ➤ Model PDF <i>Reyleigh</i> / Eksponensial ($m = 1$) ➤ Menggunakan sejumlah penyebaran yang terdistribusi secara acak tanpa penyebar yang dominan
Swerling 3	<ul style="list-style-type: none"> ➤ Dekorelasi <i>scan-to-scan</i> ➤ Model PDF <i>Chi-square</i> dengan 4 derajat kebebasan ($m = 2$) ➤ Menggunakan sejumlah penyebaran dengan 1 penyebar yang dominan
Swerling 4	<ul style="list-style-type: none"> ➤ Dekorelasi <i>pulse-to-pulse</i> ➤ Model PDF <i>Chi-square</i> dengan 4 derajat kebebasan ($m = 2$)

	➤ Menggunakan sejumlah penyebaran dengan 1 penyebar yang dominan
--	--

2.4 Detektor Neyman-Pearson

Dalam aplikasi radar dan sonar, Neyman-Pearson banyak digunakan untuk menentukan probabilitas false alarm (P_{FA}) berada pada level tertentu dan berguna untuk memaksimalkan probabilitas deteksi (P_D) [1]. Kriteria Neyman-Pearson diterapkan agar masalah deteksi *threshold* radar dengan menggunakan berbagai macam statistik data dari deteksi radar.

Konsep dari kriteria Neyman-Pearson sederhananya adalah dengan menetapkan *threshold* secara konstan, yang kemudian akan dibandingkan dengan hasil statistik data dari deteksi radar. Jika sampel sinyal berada dibawah *threshold*, maka sampel sinyal dinyatakan hanya interferensi saja. Tetapi jika sampel sinyal berada diatas *threshold*, maka sampel sinyal dinyatakan target terdeteksi yang terdiri dari sinyal intereferensi dan sinyal *echo* target.

Aturan keputusan Neyman-Pearson seperti diatas berasal dari bentuk umum *Likelihood Ratio Test* (LRT) seperti berikut [1].

$$\frac{P_y(y|H_1)}{P_y(y|H_0)} \underset{H_0}{\overset{H_1}{>}} T_y \quad (1)$$

Dimana T_y merupakan nilai *threshold*, H_0 dan H_1 merupakan Hypothesis Testing. Persamaan diatas menyatakan bahwa rasio dari dua PDF tersebut dievaluasi dengan data y yang diamati untk menghasilkan satu nilai numerik. Sehingga nilai yang didapat akan dibandingkan dengan nilai *threshold*, T_y .

2.5 Constant False Alarm Rate (CFAR)

2.5.1 False Alarm

False alarm sangat menguras sumber daya pada radar dengan muncul sebagai pendeteksian target yang valid yang memerlukan tindakan radar berikutnya. Jika tingkat *false alarm* mengalami peningkatan yang signifikan maka akan mengalami penurunan kinerja sistem yang substansial.

Tingkat false alarm didefinisikan sebagai jumlah *false alarm* yang terjadi dalam interval waktu tertentu. Tingkat false alarm, dihitung dengan menggunakan rumus berikut [1]:

$$FAR = \frac{P_{FAM}}{T_M} = N_D P_{FA} \quad (2)$$

Dimana untuk M menunjukkan jumlah resolusi sel yang dikumpulkan selama interval waktu tertentu yaitu T_M . kemudian N_D merupakan jumlah ketetapan deteksi per satuan waktu yang bisa dicari dengan menggunakan rumus $N_D = M / T_M$.

Suatu probabilitas dari *false alarm* dapat dihitung dengan persamaan (2), jika adanya interferensi yang telah didistribusikan secara eksponensial [1].

$$P_{FA} = \int_T^\infty \frac{1}{\sigma_i^2} \exp\left(-\frac{z}{\sigma_i^2}\right) dz \quad (3)$$

Dimana z adalah output dari detektor dengan inputnya berupa sinyal interferensi y yang kompleks dan dapat dicari menggunakan rumus $z = |y|^2$, dan untuk σ_i^2 adalah rata-rata atau nilai yang diharapkan dari z [1].

$$P_{FA} = \exp\left(-\frac{T}{\sigma_i^2}\right) \quad (4)$$

Untuk *threshold* atau T yang tetap, peningkatan daya interferensi menyebabkan peningkatan P_{FA} seperti yang terlihat pada persamaan (4). Misalkan daya interferensi rata-rata dinaikkan dengan faktor k kemudian *threshold* tidak disesuaikan, maka dengan menggunakan persamaan (4) probabilitas *false alarm* akibat kenaikan daya interferensi adalah [1].

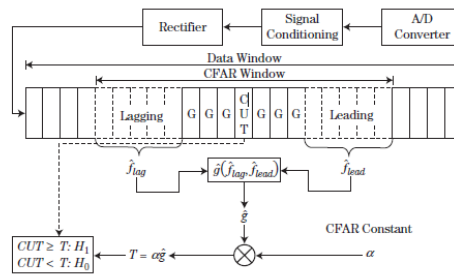
$$P_{FA_{final}} = (P_{FA_{initial}})^{\frac{1}{k}} \quad (5)$$

Dimana untuk $P_{FA_{initial}}$ adalah probabilitas awal dari *false alarm*, dan $P_{FA_{final}}$ adalah probabilitas *false alarm* yang dihasilkan setelah daya interferensi naik dengan faktor k .

2.5.2 Cell Averaging CFAR (CA-CFAR)

Metode Deteksi CFAR berfungsi untuk memperkirakan statistik interferensi dari pengukuran radar dan melakukan penyesuaian terhadap *threshold* untuk mempertahankan tingkat *false alarm* yang konstan [1]. Metode CFAR ini banyak bentuknya, seperti *Cell Averaging* CFAR (CA-CFAR), *Greatest-of* CA-CFAR (GOCA-CFAR), *Smallest-of* CA-CFAR (SOCA-CFAR), *Censored* (CS-CFAR) atau *Trimmed Mean* (TM-CFAR), dan *Order Statistic* (OS-CFAR) [1].

Kinerja dari semua bentuk metode CFAR mengacu pada arsitektur dasar CFAR seperti yang digambarkan pada Gambar 1.



Gambar 1. Bentuk umum dari dasar arsitektur detektor CFAR [1]

CFAR *window* berada didalam *window data* dan CFAR *window* terdiri dari *leading window* dan *lagging window*, sel penjaga atau *Guard Cell* (GC), dan juga sel yang sedang diuji atau *Cell Under Test* (CUT). Tujuan dari *Guard Cell* adalah untuk menghindari bocornya suatu komponen sinyal ke dalam sel pelatihan atau *training cells*, yang nantinya dapat mempengaruhi estimasi dari suatu *noise*. Untuk *leading window* maupun *lagging window* pada jendela referensi, pengukurannya digunakan untuk memperkirakan statistik interferensi.

CA-CFAR dirancang dan diperkenalkan pertama kali oleh Finn dan Johnson pada tahun 1968 [4]. Algoritma ini bisa dibidang relatif sederhana karena menghitung *threshold* berdasarkan perkiraan kekuatan interferensi rata-rata pada jendela referensi, yaitu pada *leading window* dan *lagging window*. Untuk deteksi *threshold* atau T, bisa di dapatkan menggunakan rumus [3].

$$T = \alpha P_n \tag{6}$$

Dimana untuk P_n adalah estimasi dari daya *noise*, dan α adalah faktor *threshold* atau bisa juga disebut sebagai konstanta *threshold*.

Pada CA-CFAR, sampel dari *noise* diekstrak dari sel referensi, yaitu *leading cells* dan *lagging cells* yang berada disisi CUT. Untuk mendapatkan estimasi dari daya *noise* atau P_n , dapat dihitung dengan rumus [3].

$$P_n = \frac{1}{N} \sum_{m=1}^N (x_m)^2 \tag{7}$$

Dimana untuk N merupakan jumlah pada *training cells* atau sel referensi dan x_m adalah suatu sampel *square magnitude* dari setiap sel referensi. Untuk konstanta atau faktor *threshold* yaitu α , bisa juga didapatkan dengan [3].

$$\alpha = N(P_{FA})^{\frac{-1}{N}} - 1 \tag{8}$$

Dimana untuk P_{FA} adalah probabilitas dari *false alarm*

2.5.3 Greatest-of Cell Averaging CFAR (GOCA-CFAR)

Hansen dan Sawyers telah mengembangkan metode CA-CFAR yang disebut *Greatest-Of Cell Averaging* (GOCA-CFAR) untuk mengurangi gangguan *false alarm*. Gangguan *false alarm* dapat ditekan atau dikurangi dengan cara menghitung daya interferensi rata-rata pada *leading window* dan *lagging window* secara terpisah dan memilih yang paling besar diantara hasil perhitungan tersebut sebagai statistik CFAR [5].

3. Pembahasan

3.1 Model Simulasi

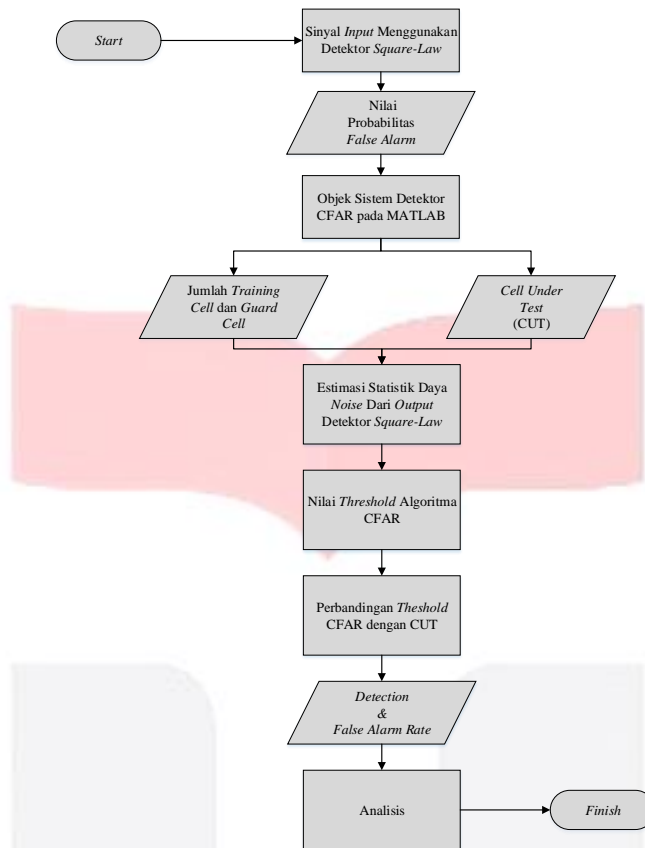
Pada simulasi ini akan dilakukan 2 skenario simulasi dengan kondisi yang homogen dan heterogen. Untuk kedua skenario akan menggunakan sinyal yang sama-sama berasal dari detektor *square-law*. Untuk skenario pertama akan menggunakan algoritma CA-CFAR dengan dasar teori deteksi Neyman-Pearson sebagai pembanding. Lalu untuk skenario kedua akan menggunakan algoritma GOCA-CFAR dengan CA-CFAR sebagai pembanding. Simulasi ini menggunakan nilai asumsi untuk beberapa parameter seperti pada tabel berikut.

Tabel 2. Parameter Simulasi

No	Parameter	Nilai
1	Jumlah <i>Training Cell</i>	100 cell
2	Jumlah <i>Guard Cell</i>	20 cell
3	Jumlah Pengujian Monte Carlo	1000
4	Probabilitas <i>False Alarm</i> (P_{FA})	10^{-1} 10^{-2}

		10^{-3}
		10^{-4}
5	Probabilitas Deteksi (P_D)	90 %

dan simulasi ini akan mengacu pada *flowchart* seperti gambar berikut.

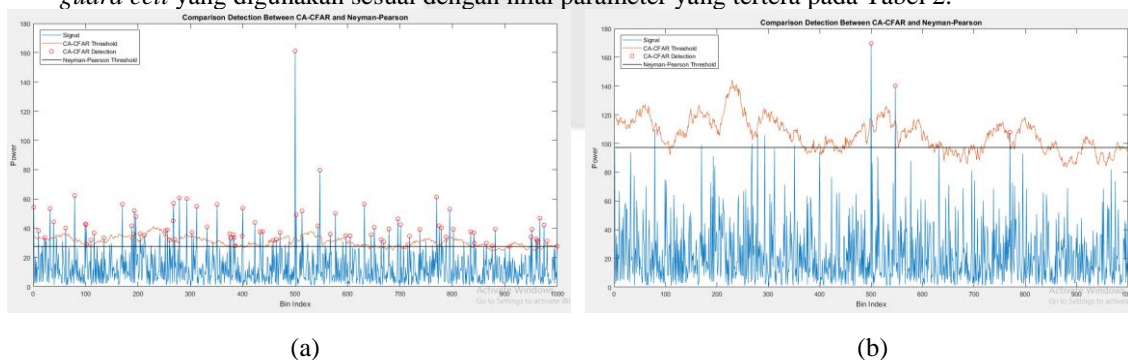


Gambar 2. *Flowchart* Percancangan Algoritma CFAR

3.2 Hasil Simulasi

3.2.1 Skenario Satu (CA-CFAR)

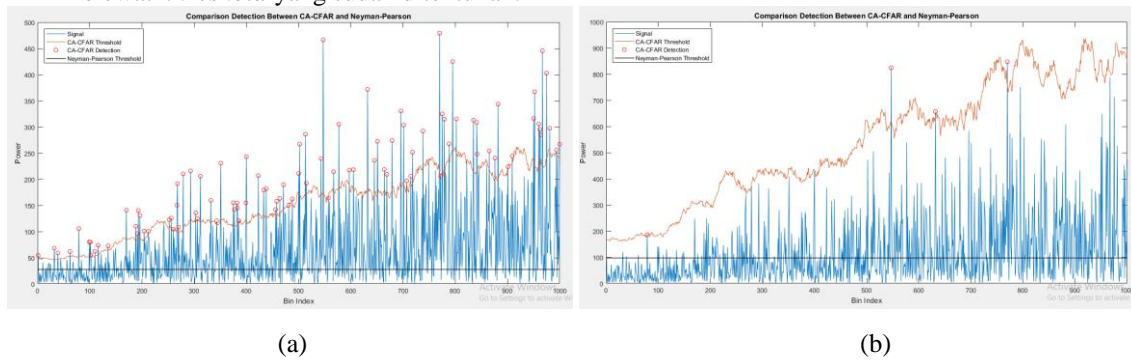
Simulasi skenario satu untuk kondisi *noise* yang homogen menggunakan nilai PFA sebesar 10^{-1} dan 10^{-2} agar analisis dapat dilakukan dengan mudah. Nilai SNR untuk mendapatkan *threshold* yang diinginkan adalah masing-masing sebesar 10.2 dB dan 12.8 dB. Kemudian jumlah *training cell* dan *guard cell* yang digunakan sesuai dengan nilai parameter yang tertera pada Tabel 2.



Gambar 3. Hasil Simulasi CA-CFAR Dengan (a) $P_{FA} = 10^{-1}$ dan SNR=10.2 dB, (b) $P_{FA} = 10^{-2}$ dan SNR=12.8 dB, Kondisi *Noise* Homogen

Dari Gambar 3. memperlihatkan bahwa perbedaan hasil deteksi dengan menggunakan nilai P_{FA} yang berbeda mengakibatkan tingkat *false alarm* yang berbeda. Semakin besar nilai P_{FA} yang dipilih, maka semakin besar juga tingkat *false alarm* yang dihasilkan. Hal ini dikarenakan *threshold* dari detektor diatur dengan menggunakan nilai P_{FA} . Jika nilai P_{FA} -nya besar, maka jangkauan *threshold*

dengan sinyal akan sangat dekat. Sehingga banyak sinyal, baik itu sinyal target atau sinyal *noise*, melewati *threshold* yang sudah ditentukan.

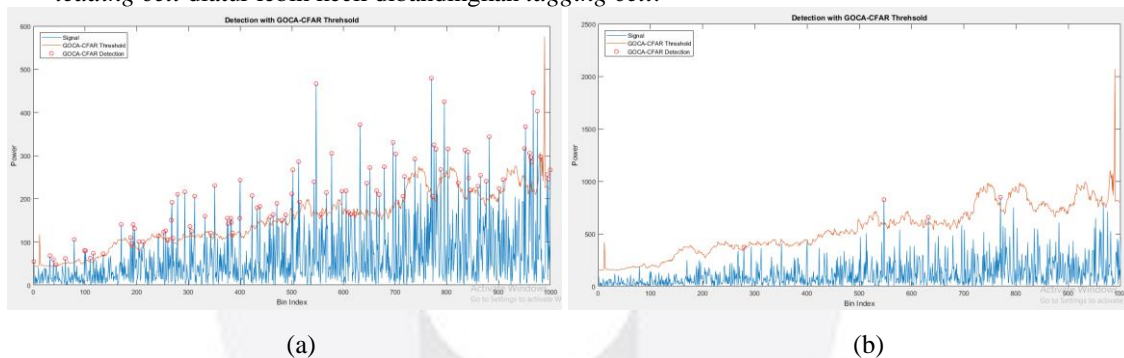


Gambar 4. Hasil Simulasi CA-CFAR Dengan (a) $P_{FA} = 10^{-1}$ dan SNR=10.2 dB, (b) $P_{FA} = 10^{-2}$ dan SNR=12.8 dB, Kondisi *Noise* Heterogen

Berdasarkan Gambar 3 dan Gambar 4, dapat dikatakan bahwa detektor CA-CFAR dapat melakukan deteksi dengan baik ketika berada dalam kondisi *noise* homogen dan heterogen. Berbeda dengan detektor Neyman-Pearson, pada saat kondisi *noise* homogen, kinerja detektor masih cukup bagus dengan menghasilkan tingkat *false alarm* yang sama dengan detektor CA-CFAR. Tetapi ketika kondisi *noise* heterogen, detektor tidak dapat mengatasi kenaikan daya *noise* yang mengakibatkan tingkat *false alarm*-nya sangat tinggi.

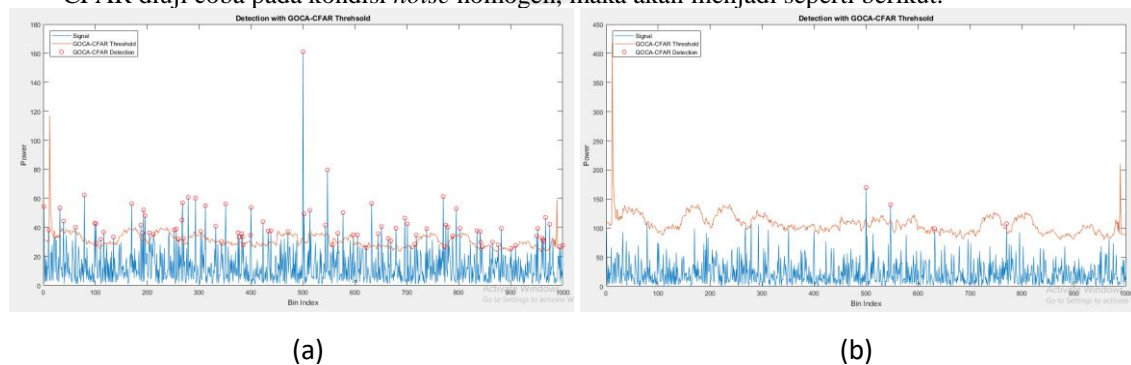
3.2.2 Skenario Dua (GOCA-CFAR)

Metode GOCA-CFAR sejatinya digunakan untuk kondisi *noise* yang heterogen. Jadi untuk skenario dua menggunakan nilai parameter P_{FA} yang sama dengan skenario satu. Tetapi jumlah pada *leading cell* diatur lebih kecil dibandingkan *lagging cell*.



Gambar 5. Hasil Simulasi GOCA-CFAR Dengan (a) $P_{FA} = 10^{-1}$ dan SNR=10.2 dB, (b) $P_{FA} = 10^{-2}$ dan SNR=12.8 dB, Kondisi *Noise* Heterogen

Gambar 5. Menjelaskan bahwa detektor GOCA-CFAR dapat melakukan deteksi sinyal dengan baik saat berada dalam kondisi *noise* heterogen. Meskipun terjadi kenaikan daya *noise*, tingkat *false alarm* yang dihasilkan tetap stabil dan sesuai dengan nilai parameter yang diinginkan. Apabila GOCA-CFAR diuji coba pada kondisi *noise* homogen, maka akan menjadi seperti berikut.



Gambar 6. Hasil GOCA-CFAR Dengan (a) $P_{FA} = 10^{-1}$ dan SNR=10.2 dB, (b) $P_{FA} = 10^{-2}$ dan SNR=12.8 dB, Kondisi *Noise* Homogen

Jika dibandingkan antara Gambar 5. dan Gambar 6, maka kinerja dari detektor GOCA-CFAR hampir sama dengan CA-CFAR. Karena dari hasil deteksi yang dilakukan, GOCA-CFAR menghasilkan tingkat *false alarm* yang sama dan sinyal target pun tetap dapat terdeteksi. Tetapi ada sedikit perbedaan dari segi jumlah deteksi ketika melakukan 1000 kali percobaan Monte-Carlo dengan nilai $P_{FA} = 10^{-2}$, yaitu pada CA-CFAR mendapatkan 3 hasil deteksi sedangkan GOCA-CFAR mendapatkan 4 hasil deteksi. Meskipun berbeda 1 angka saja, jika dalam realita aslinya hal ini dapat membuat kinerja radar sedikit menurun. Karena sejatinya GOCA-CFAR diaplikasikan ketika sedang berada dalam kondisi *noise* yang heterogen.

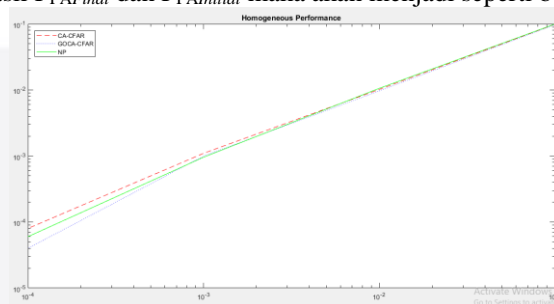
3.2.3 Hasil Analisis Dari Dua Skenario Simulasi

Dari hasil antara Skenario Satu dan Skenario Dua, didapatkan nilai probabilitas *false alarm* akhir ($P_{FAFinal}$) dengan menggunakan nilai P_{FA} awal ($P_{FAInitial}$), dimana nilai $P_{FAInitial}$ merupakan nilai parameter pada tabel 2. Hasil nilai $P_{FAFinal}$ dengan $P_{FAInitial}$ sebesar 10^{-1} sampai 10^{-4} adalah sebagai berikut.

Tabel 3. Hasil Nilai $P_{FAFinal}$ Saat Kondisi *Noise* Homogen

$P_{FAInitial}$	$P_{FAFinal}$		
	CA-CFAR	GOCA-CFAR	Neyman-Pearson
10^{-1}	0.0992	0.0975	0.0997
10^{-2}	0.0102	0.0097	0.0106
10^{-3}	0.0011	1.0000e-03	9.6000e-04
10^{-4}	8.0000e-05	4.0000e-05	6.0000e-05

Hasil dari Tabel 3. didapatkan dengan melakukan 50.000 kali percobaan Monte Carlo dan jika dibuatkan grafik dari hasil $P_{FAFinal}$ dan $P_{FAInitial}$ maka akan menjadi seperti berikut.



Gambar 7 Grafik antara Hasil $P_{FAFinal}$ dan $P_{FAInitial}$ Dalam Kondisi *Noise* Homogen

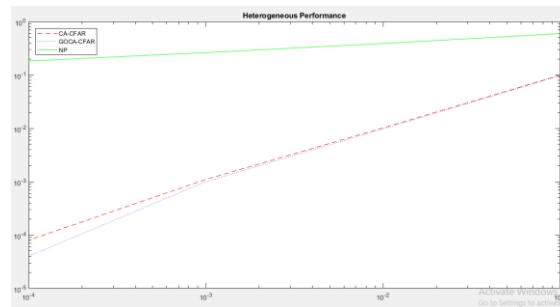
Berdasarkan Gambar 7 bisa dikatakan bahwa kinerja dari CA-CFAR, GOCA-CFAR, dan Neyman-Pearson menghasilkan hasil $P_{FAFinal}$ yang hampir sama saat berada dalam kondisi *noise* homogen. Hal ini dikarenakan CA-CFAR, GOCA-CFAR, maupun Neyman-Pearson dapat menjaga tingkat *false alarm* sesuai dengan nilai $P_{FAInitial}$ sebagai nilai acuannya seperti yang sudah diperlihatkan pada Gambar 3 dan Gambar 6.

Berbeda halnya dengan hasil $P_{FAFinal}$ dari $P_{FAInitial}$ saat berada dalam kondisi *noise* heterogen seperti pada tabel berikut.

Tabel 4 Hasil Nilai $P_{FAFinal}$ Saat Kondisi *Noise* Heterogen

$P_{FAInitial}$	$P_{FAFinal}$		
	CA-CFAR	GOCA-CFAR	Neyman-Pearson
10^{-1}	0.0992	0.0975	0.5972
10^{-2}	0.0102	0.0097	0.3889
10^{-3}	0.0011	1.0000e-03	0.2625
10^{-4}	8.0000e-05	4.0000e-05	0.1826

Tabel 4 memperlihatkan hasil CA-CFAR dan GOCA-CFAR saat dalam kondisi *noise* heterogen mendapatkan nilai $P_{FAFinal}$ yang sama ketika dalam kondisi *noise* homogen. Tetapi untuk Neyman-Pearson mendapatkan hasil berbeda dan sangat jauh dari nilai $P_{FAInitial}$ sebagai nilai acuannya. Jika dibuatkan grafik, maka hasil $P_{FAFinal}$ akan terlihat seperti berikut.



Gambar 8 Grafik antara Hasil $P_{FAFinal}$ dan $P_{FAInitial}$ Dalam Kondisi *Noise* Heterogen

Dari Gambar 8 terlihat jelas bahwa tingkat *false alarm* antara CA-CFAR dan GOCA-CFAR dengan Neyman-Pearson sangatlah jauh dan berbeda. Hal ini dikarenakan ketika melakukan proses deteksi, *threshold* Neyman-Pearson ditetapkan secara konstan pada level tertentu. Sehingga ketika terjadi kenaikan daya dari *noise*, *threshold* dari Neyman-Pearson banyak dilewati oleh sinyal yang dihasilkan dari detektor *square-law* dan menghasilkan tingkat *false alarm* yang sangat tinggi dan tidak terkontrol seperti yang sudah diperlihatkan pada Gambar 4 dan Gambar 5.

4. Kesimpulan

Berdasarkan hasil simulasi dan analisis data yang dilakukan, maka dapat disimpulkan menjadi beberapa poin seperti berikut :

1. Semakin kecil nilai probabilitas *false alarm* yang digunakan, maka jangkauan antara *threshold* dan sinyal akan semakin jauh. Sehingga tingkat *false alarm* dapat ditekan seminimal mungkin. Apabila nilai P_{FA} diatur semakin kecil tanpa memperhitungkan tes statistik dari sinyal keluaran detektor, maka dapat menimbulkan masalah lain yaitu terjadi *miss detection*.
2. Dalam kondisi *noise* yang homogen, hasil deteksi CA-CFAR dan GOCA-CFAR serta Neyman-Pearson tidak ditemukan perbedaan yang cukup signifikan. Ketiga detektor tersebut dapat mendeteksi target dengan baik dan menghasilkan tingkat *false alarm* yang sesuai dan tidak terlalu jauh dengan nilai parameter $P_{FAInitial}$.
3. Dalam kondisi *noise* yang heterogen, detektor dengan algoritma CFAR sangat baik dalam melakukan proses deteksi jika dibandingkan dengan detektor dari Neyman-Pearson. Hal ini dikarenakan algoritma CFAR dapat beradaptasi dengan sinyal keluaran dari detektor, salah satunya detektor *square-law*. Jadi meskipun ada kenaikan daya *noise* saat melakukan deteksi, algoritma CFAR tetap dapat mendeteksi target dengan baik dan menghasilkan tingkat *false alarm* yang sesuai dengan nilai parameter $P_{FAInitial}$. Berbeda dengan detektor Neyman-Pearson yang menunjukkan kinerja cukup buruk. Banyak sinyal yang melewati *threshold* dari Neyman-Pearson sehingga tingkat *false alarm* yang dihasilkan sangat tinggi, baik menggunakan nilai $P_{FAInitial}$ sebesar 10^{-1} sampai 10^{-4} .

Daftar Pustaka

- [1] J. A. Scheer, W. A. Holm and M. A. Richards, Principles of Modern Radar, Vol. 1: Basic Principle, vol. I, D. R. Kay, Ed., Atlanta, Georgia: SciTech Publishing, 2010.
- [2] M. Skolnik, "Introduction to Radar Systems," 3 ed., New York, McGraw Hill, 2001, p. 66.
- [3] M. Richard, Fundamental of Radar Signal Processing, New York: McGraw Hill, 2005.
- [4] H. Finn and R. Johnson, "Adaptive Detection Mode with Threshold Control as a Function of Spatially Sampled Clutter-Level Estimates," RCA Review, 1968, pp. 414-464.
- [5] V. G. Hansen and J. H. Sawyers, "Detectability Loss Due to "Greatest of" Selection in a Cell-Averaging CFAR," Detectability Loss Due to "Greatest of" Selection in a Cell-Averaging CFAR, Vols. AES-16, no. 1, pp. 115-118, 1980.