

# Robot Teknolojilerine Giriş

# Keşif Algoritmaları

Mehmet Fatih Amasyalı



# Robot Gezinimi

- Bilinen bir ortamda hareket
  - Lokalizasyon (neredeyim)
  - Navigasyon (şuraya nasıl giderim)
- Bilinmeyen bir ortamda hareket
  - Keşif (nereye gideyim)
  - Haritalama (nasıl bir ortamdayım)
  - Navigasyon (şuraya nasıl giderim)



# Neden Keşif

- Arama kurtarma
- Hız
- Güvenlik
- Daha keşfedecek çok yer var
  - Uzay, okyanuslar



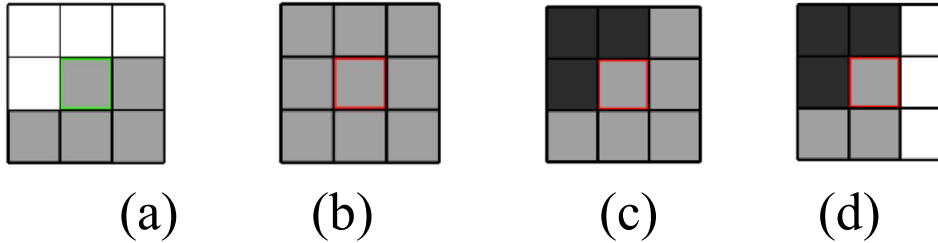
# Basit keşif algoritmaları

- Duvar takibi
- En uzağa git
- Haritalama gerektirmezler
- Ama ortamın tamamını gezemezler



# Her yeri gezmek için

- Gezilen ve gezilmeyen yerleri bilmek gerekir (Harita)
- Grid/hücre tabanlı yaklaşım:



Örnek hücreler. Beyaz hücreler keşfedilmemiş, gri hücreler keşfedilmiş ve boş, siyah hücreler ise keşfedilmiş ve doludur. Ortadaki hücre sadece (a) durumunda Frontier (öncül) olarak belirlenmektedir.

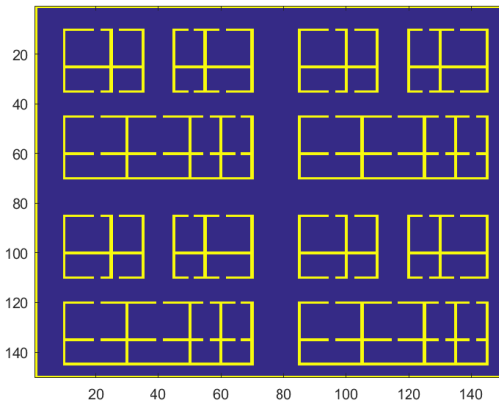
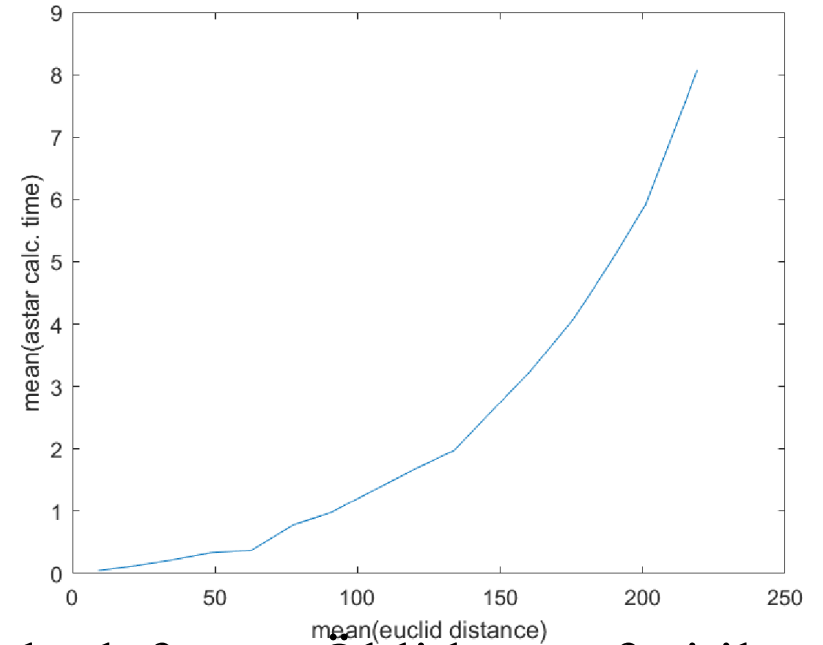
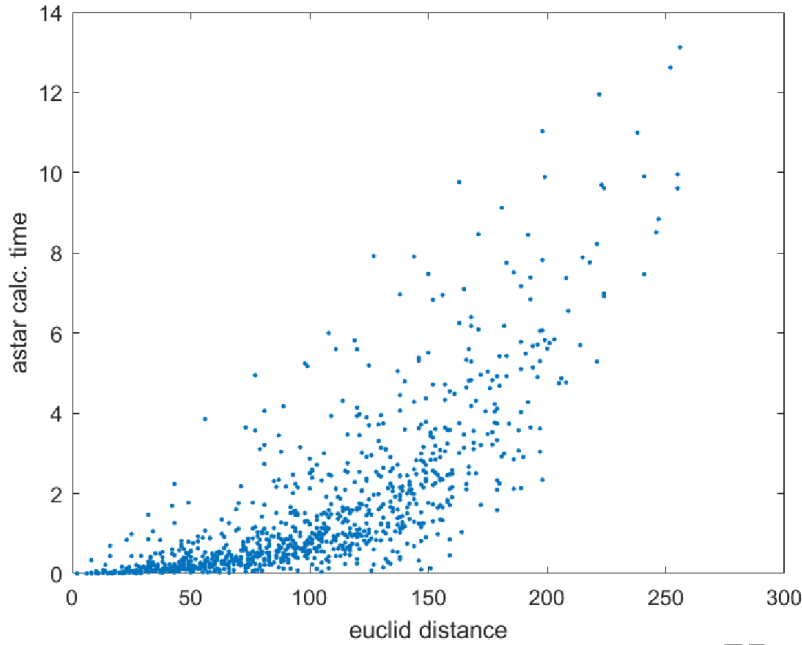
Öncül: kendisi keşfedilmiş ve boş ve en az bir keşfedilmemiş komşusu olan hücre

# Haritada birçok öncül hücre olabilir. Hangisine gitmeliyim?

- $(x_i, x_j)$  koordinatındaki bir hücrenin iyiliği nasıl ölçülür:
- $(r_i, r_j)$  koordinatındaki robota en yakın en iyi:
- $U(x_i, x_j) = C((r_i, r_j), (x_i, x_j))$
- $C$ ? öklid?  $A^*$ ?
- min  $U$  değerine sahip hücre hedef



# A\* maliyet



Kaynak-hedef arası Öklid mesafesi ile  
path hesaplama süresi arasındaki ilişki

a) 1000 denemenin tüm sonuçları

b) b) ortalama değerler

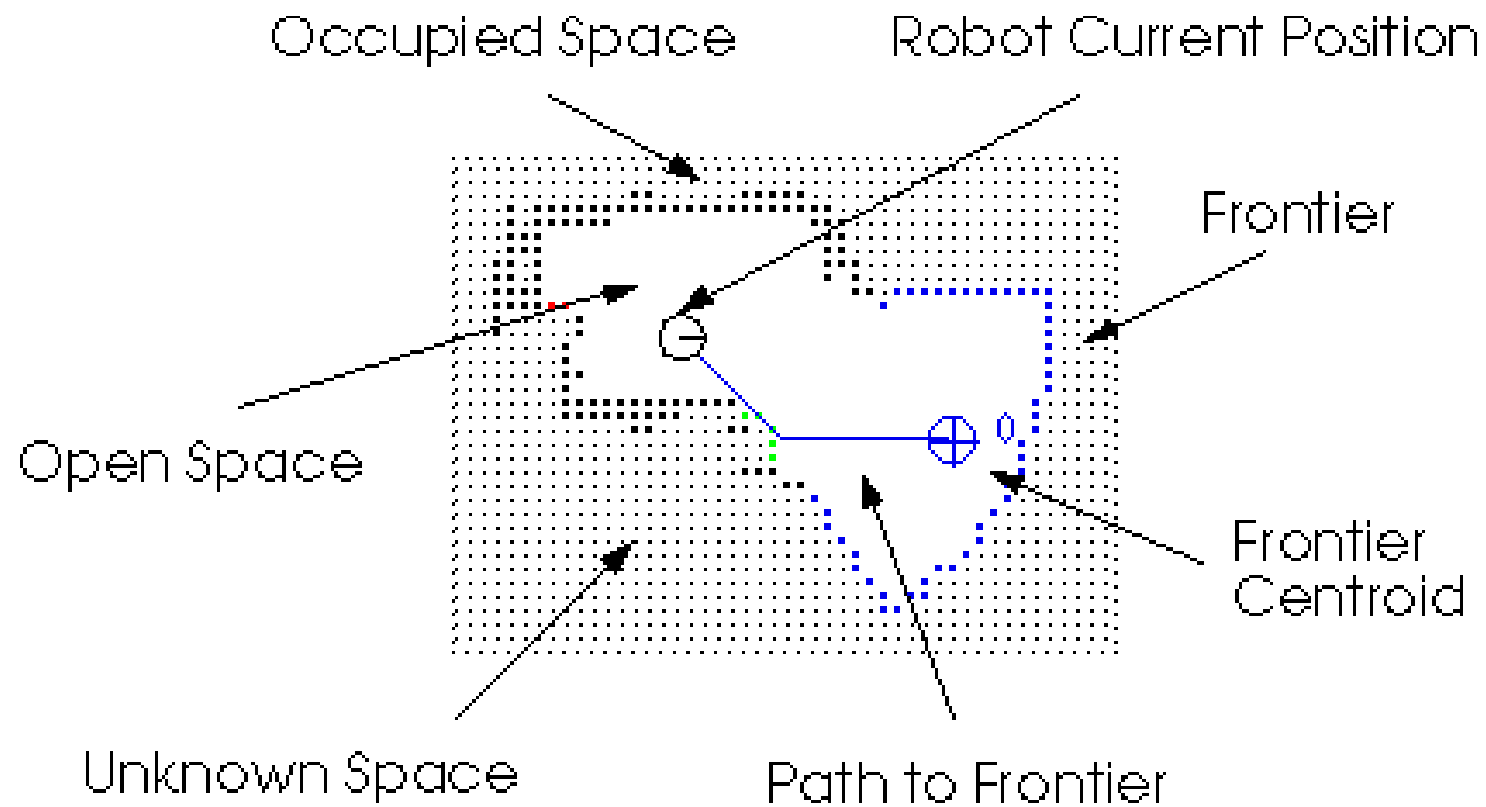
Boş bir harita için ilişki nasıl olur?

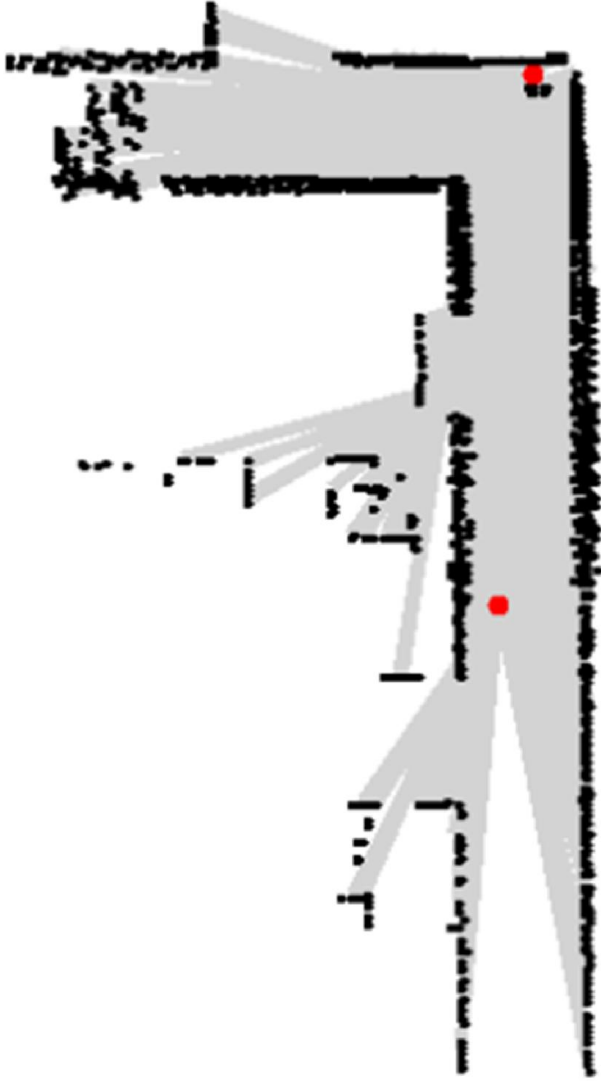
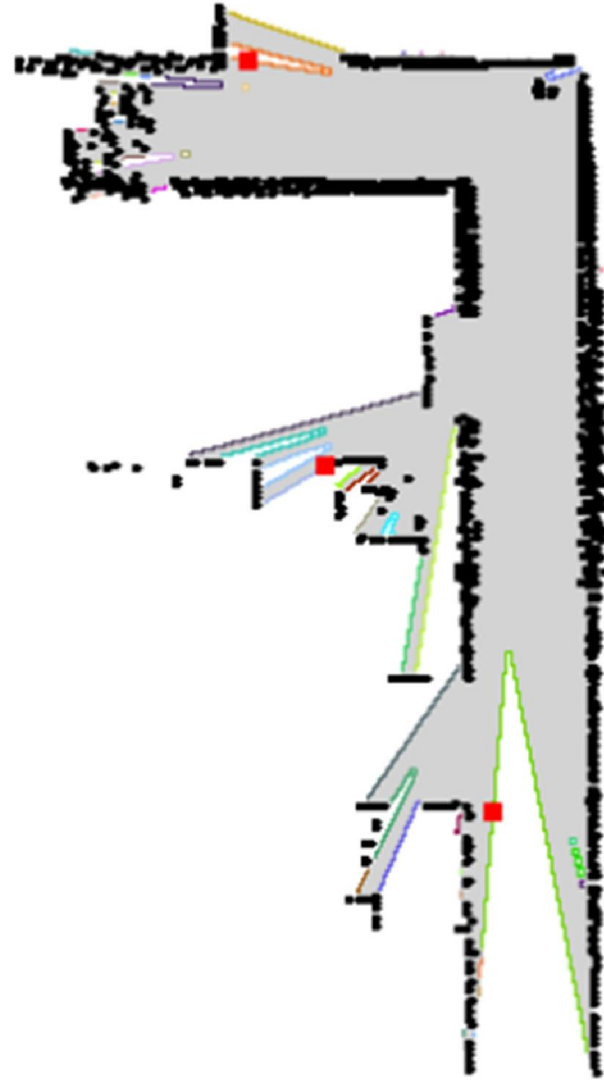
# Öncülleri kümelemek

- $A^*$  pahalı ama lazım,
- Çok fazla hesaplamamak lazım
- Her öncül hücre için hesaplamak yerine
- Hücreleri bağımlı bileşenler ile kümeleyelim. Her küme için hesaplayalım.









# Öncülleri kümelemek

- $(x_i, x_j)$  koordinatındaki bir hücre kümesinin iyiliği nasıl ölçülür:
- $U(x_i, x_j) = P(x_i, x_j) + C((r_i, r_j), (x_i, x_j))$
- $P$ ? Kümenin büyüklüğü? Küçüklüğü?
- Kümenin koordinatı?
  - İçerdiği öncüllerin ortası (öncül olmayabilir)
  - İçerdiği öncüllerden robota en yakını (pahalı)
  - Kümenin ortasına en yakın öncül (az pahalı)

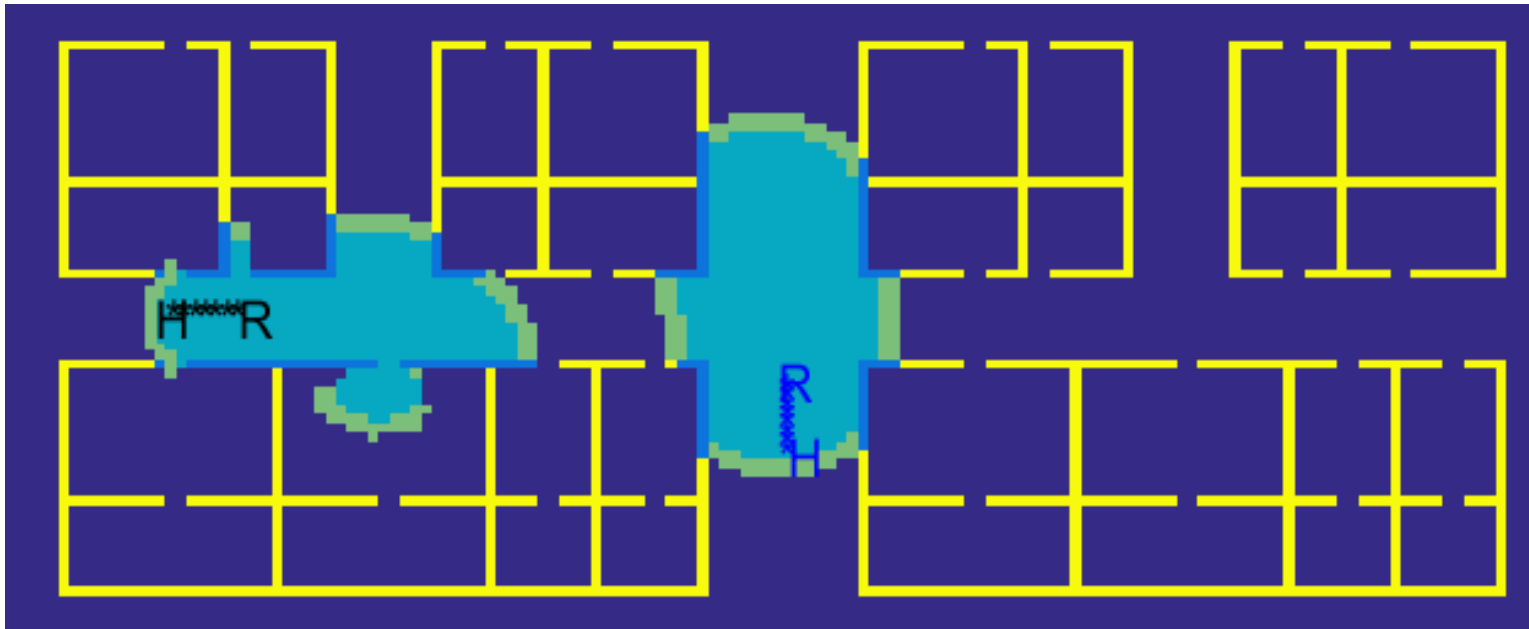


# Beyond frontiers

- Hedef seçmek için bir başka yol
- Lazer range in belirli bir aralığının içinde kalan ve boş olan hücreler beyond frontier hücreler olarak belirlenir. Bu hücrelere CCL uygulanarak kümelenirler. Her kümenin merkezine en yakın hücreler hedeftir.



# Beyond frontiers



# Tek robotla keşif algoritması

- Hedefleri belirle
- Hedeflerin iyiliğini hesapla
- Hedefi seç
- Robot-hedef için yol bul
- Robotu hedefine doğru ilerlet
- Robot hedefine varınca başa dön



# Çok robotla keşif algoritması

- Hedefleri belirle
- Hedeflerin her bir robot için iyiliklerini hesapla
- Robot-hedef eşlemesi yap
- Eşlenmiş robot-hedef ikilileri için yolları bul
- Robotları hedeflerine doğru ilerlet
- Robotlardan biri hedefine varınca başa dön



# Robot hedef eşlemesi

- R robot, H hedef olsun
- U:  $R \times H$  boyutlu bir matris
- Her bir hücre için  $A^*$  hesapla (çok pahalı)
- R kez
  - U matrisinin en küçük elemanını bul  $U(i,j)$
  - i. robota j. hedefi ata
  - i. satır ve j. sütunu U matrisinden sil
- $R \times H$  kez  $A^*$  çağrılıyor





# Hızlandırılma

- U:  $R \times H$  boyutlu bir matris
- Her bir hücre için Öklid hesaplama (çok ucuz)
- R kez
  - U matrisinin en küçük elemanını bul  $U(i,j)$
  - i. satırdaki tüm elemanlar için  $A^*$  hesaplama. En küçüğünü (j) bul. i. robota j. hedefi ata
  - i. satırı ve j. sütunu U matrisinden sil
- Faydası? her bir adımda hesaplanan  $A^*$  path sayısı birer birer azalmakta



# Daha da hızlandır Salih Marangoz ☺

- U:  $R \times H$  boyutlu bir matris
- Her bir hücre için Öklid hesapla (çok ucuz)
- R kez
  - 1- U matrisinin en küçük elemanını bul  $U(i,j)$
  - 2-  $U(i,j)$  için  $A^*$  hesapla. Bu değerden büyük tüm U elemanlarını sil
  - İşlenmemiş eleman kalmayıncaya kadar 1 ve 2 yi tekrar et
  - U'nun en küçüğünü bul  $U(k,t)$ . k. robota t. hedefi ata
  - k. satırı ve t. sütunu U matrisinden sil



# Daha da hızlandır Salih Marangoz 😊

Step	U matrix					
1.step						
	T1	T2	T3	T4	T5	T6
R1	7	3	8	9	12	20
R2	8	9	4	15	20	23
2.step						
	T1	T2	T3	T4	T5	T6
R1	7	7	inf	inf	inf	inf
R2	inf	inf	4	inf	inf	inf
3.step						
	T1	T2	T3	T4	T5	T6
R1	inf	inf	inf	inf	inf	inf
R2	inf	inf	6	inf	inf	inf
4.step						
	T1	T2	T4	T5	T6	
R1	7	3	9	12	20	
5.step						
	T1	T2	T4	T5	T6	
R1	7	7	inf	inf	inf	
6.step						
	T1	T2	T4	T5	T6	
R1	8	7	inf	inf	inf	

## Process

[r,h]=min(U), r=1, h=2

Calculate Path\_size(Rs(1,:),Ts(2,:))

Assume path\_size=7

[r,h]=min(U), r=2, h=3

Calculate Path\_size(Rs(2,:),Ts(3,:))

Assume path\_size=6

All cells of U is processed

Assign T3 to R2

Exclude assigned robot and target from U

[r,h]=min(U), r=1, h=2

Path\_size(Rs(1,:),Ts(2,:)) is already  
calculated. path\_size=7

[r,h]=min(U), r=1, h=1

Calculate Path\_size(Rs(1,:),Ts(1,:))

Assume path\_size=8

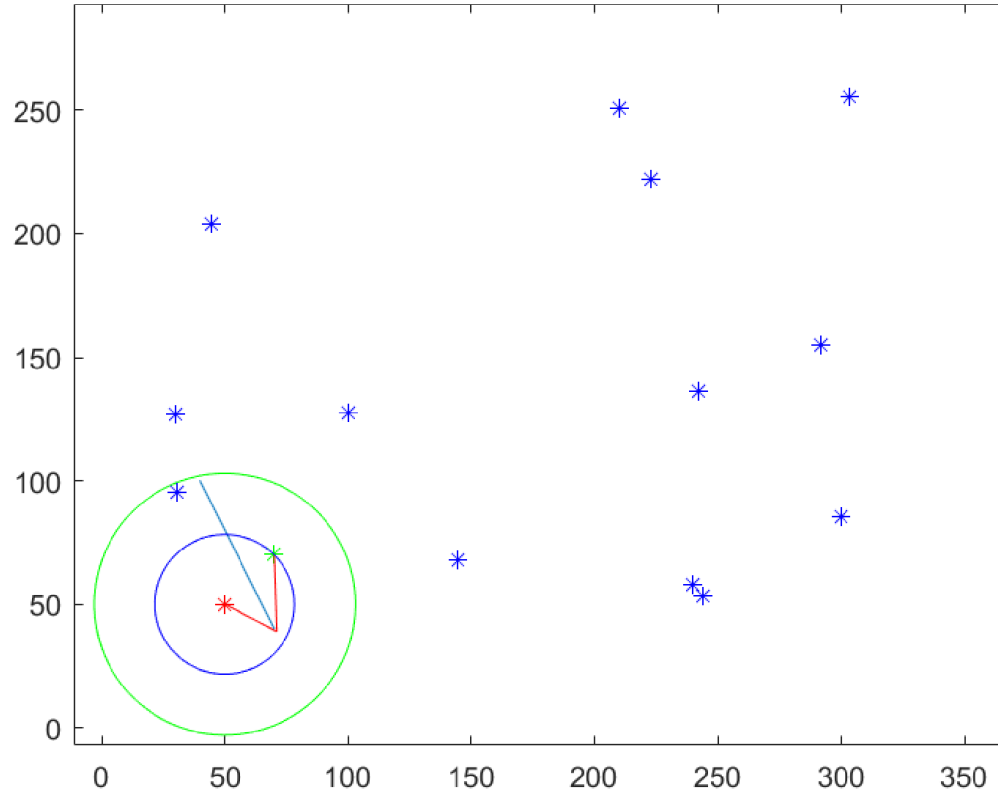
All cells of U is processed

Assign T2 to R1

Ü



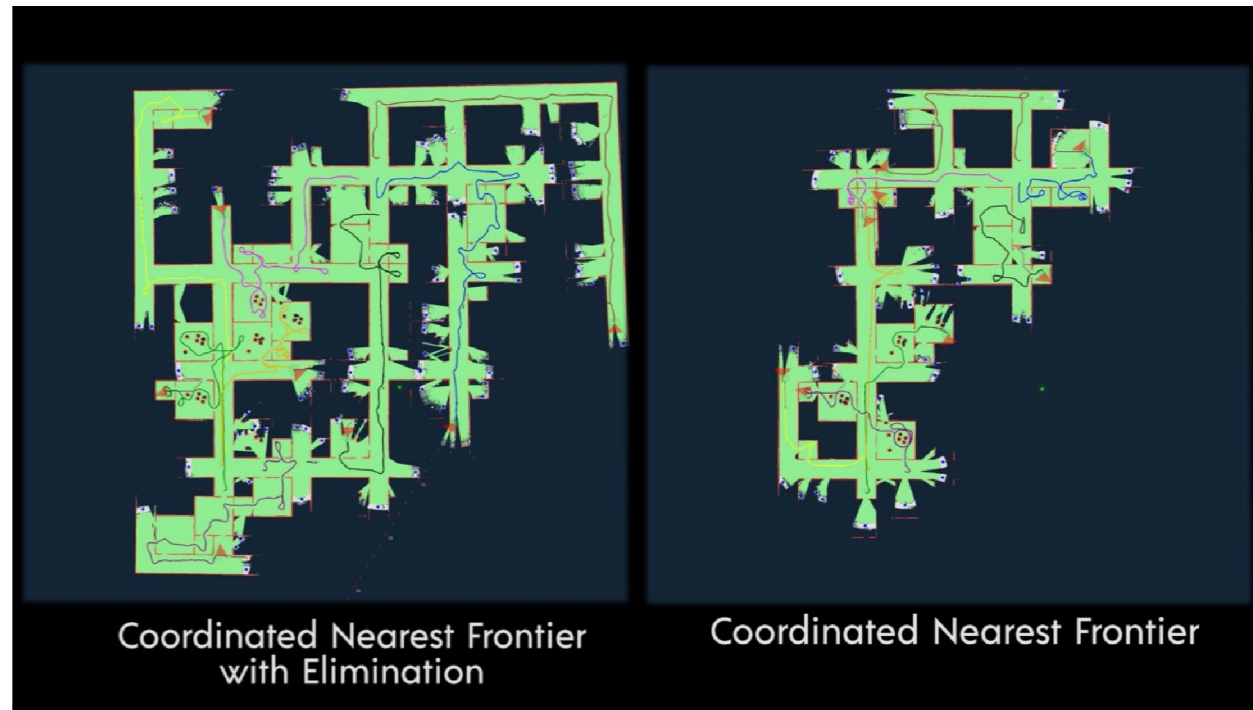
# Daha da hızlandır Salih Marangoz ☺



# Sonuçları görelim

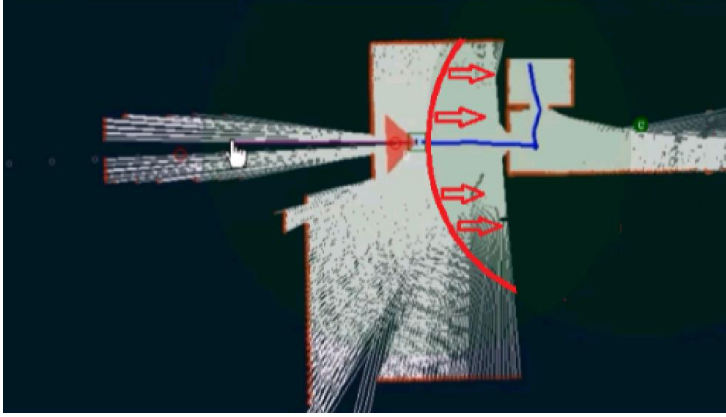
- Matlab uygulaması (multi\_kesif\_v7.m)
- Önce VI
- Sonra SA

Gazebo  
uygulaması



# Kısıtlı iletişim durumu

## Teşekkürler Attila Akıncı ☺



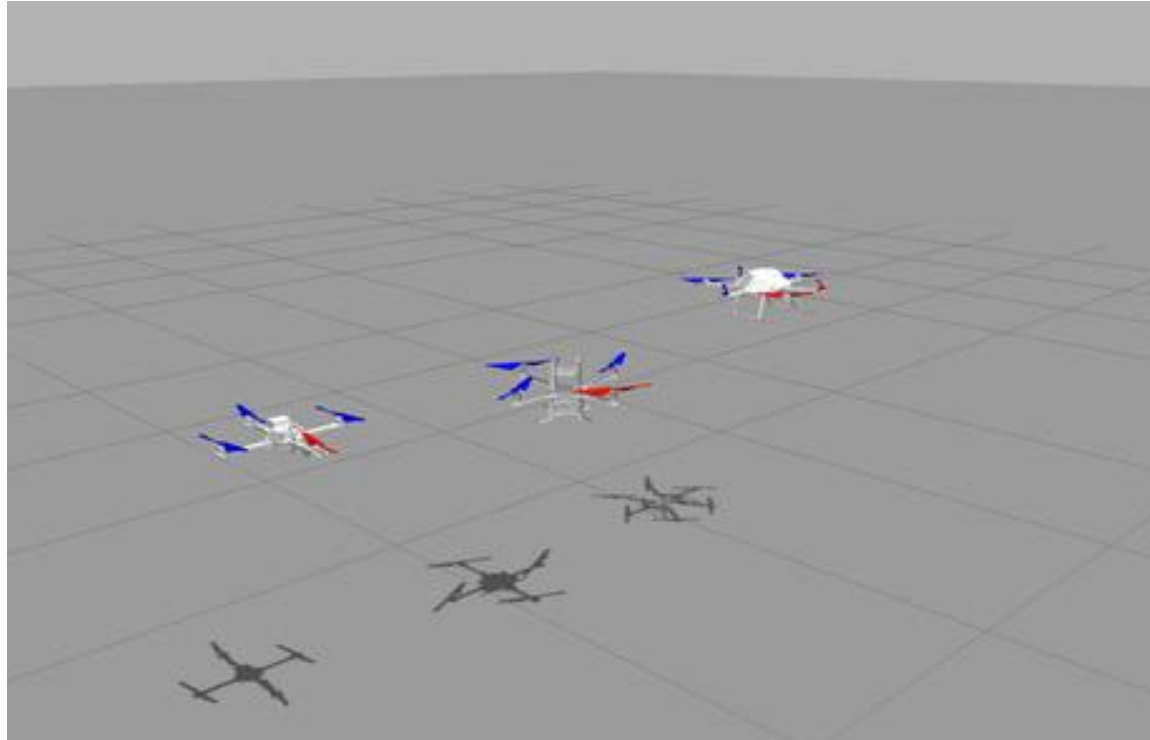
Robotun kapsama alanını terk edişi

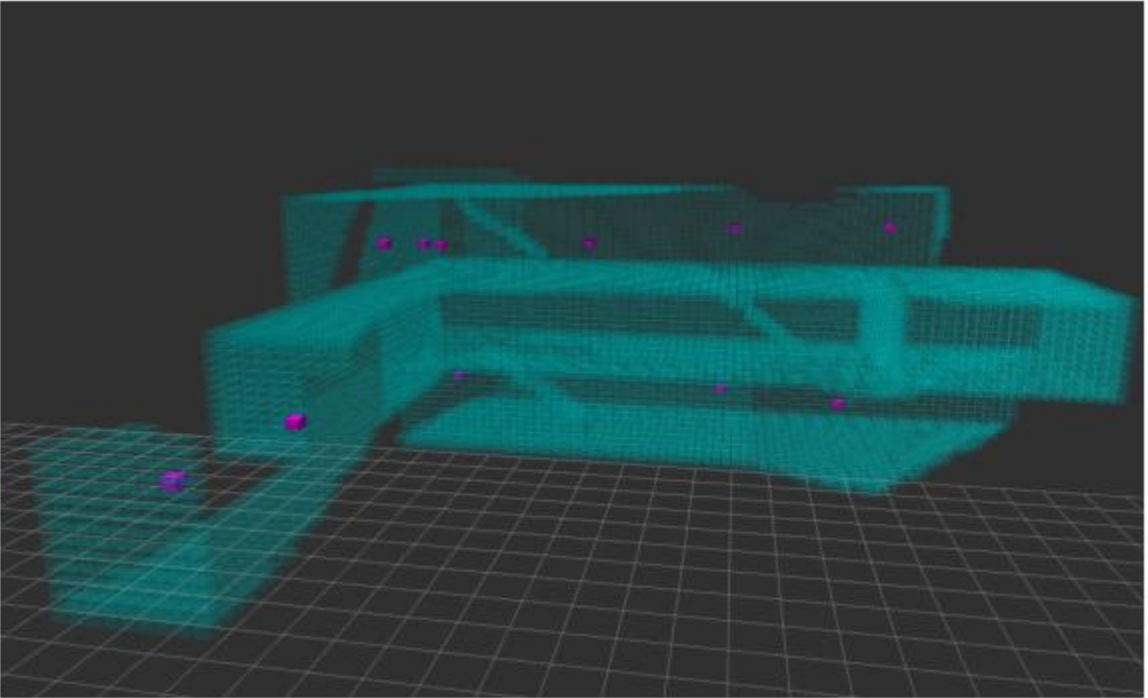
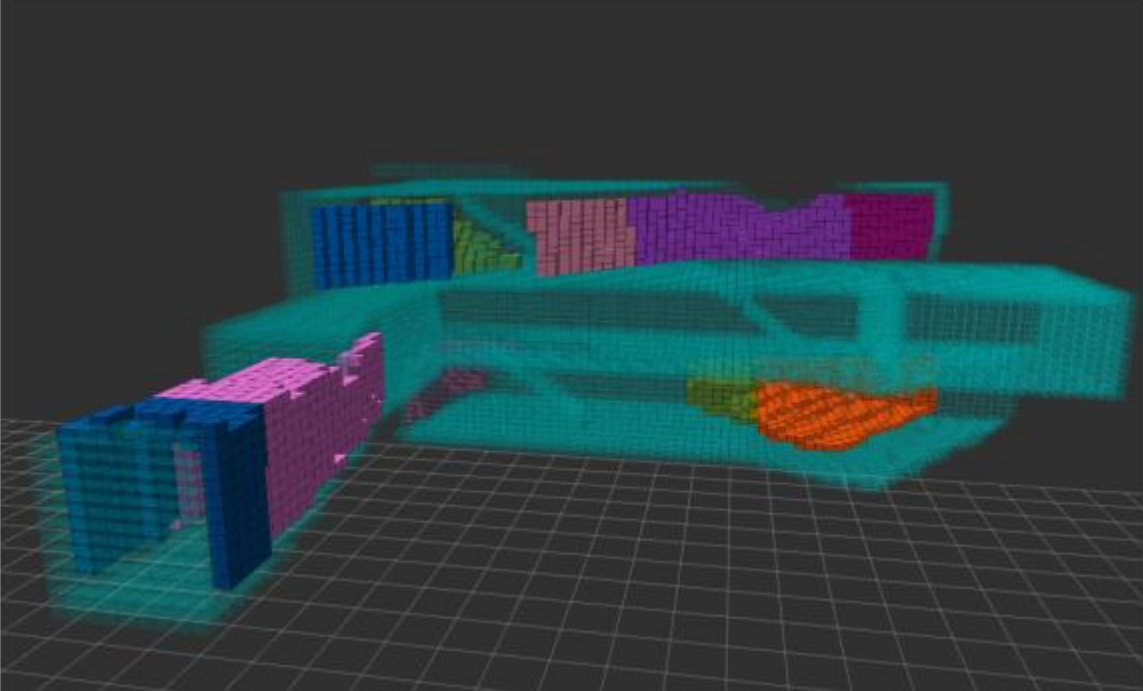


Robotun kapsama alanına topladığı bilgilerle dönüşü

# 3 boyutlu keşif

- 3 boyutlu hareket, drone
- 3 boyutlu hedef belirleme
- 3 boyutlu A\*
- Teşekkürler Salih Marangoz ve Ezgi Ekin Ergün ☺







# Kaynaklar

- <http://robotfrontier.com>
- Visser, Arnoud, and Bayu A. Slamet. "Including communication success in the estimation of information gain for multi-robot exploration.", WiOPT 2008.
- <http://www.kostasalexis.com/rotors-simulator.html>
- [https://www.youtube.com/channel/UCd\\_EolndKWCPKJ1KQprSb3g](https://www.youtube.com/channel/UCd_EolndKWCPKJ1KQprSb3g)

