Proje Raporu

Proje Geliştirme ve Mantıksal Arkaplanı

Proje isterleri raporuna bağlı kalmaya çalışılarak geliştirilen projede apache kafka ve kestirim gibi terimsel ifadeler üzerine tecrübe eksikliği nedeniyle ilgili konularda araştırma ve bilgi edinme projenin büyük bir kısmında rol aldı.

Kafka broker, topic , zookeeper ve server yapıları ile ilgili bilgiler edinildi. Projedeki yapıya yönelik bir tasarım yapıldı. 2 producer 1 consumer yapısı üzerine bir algoritma tasarlandı. Bu noktada producer görevini projedeki sensorler üstlenirken consumer merkezi birim oldu. Bu noktada server üzerinden haberleşme ile bilgilerin aktarımı sağlanıp merkezi birim tarafında matematiksel hesaplamaların yapılması kararlaştırıldı.

Sensörlerden gelecek lokasyon ve kestirim açılarından bir hedef lokasyonu belirlemek için gerekli matematiksel hazırlık yapıldı. Kestirim açılarından bir eğim çıkarılıp sensör lokasyonundan bir doğruya ulaşılabileceği saptandı.

Bir noktası ve eğimi verilen 2 doğrunun kesiştiği nokta hedefin bulunduğu yerin haritadaki koordinatlarını verecektir. Bu noktada dikkat edilmesi gereken matematiksel durumlar mevcuttu:

Aynı düzlemde bulunan iki doğrunun kesişimi üç farklı şekilde olabilir.

1. Doğruların eğimleri farklı ise tek bir noktada kesişirler, dolayısıyla iki doğrunun da denklemini sağlayan tek bir (x,y) ikilisi vardır.
2. Doğrular birbirine paralel ise hiçbir noktada kesişmezler, dolayısıyla iki doğrunun da denklemini sağlayan (x,y) ikilisi yoktur.
3. Doğrular çakışık ise sonsuz sayıda ortak noktaları vardır, dolayısıyla iki doğrunun da denklemini sağlayan sonsuz sayıda (x,y) ikilisi vardır.

Bu üç durum için algoritmamızda kontroller eklememiz gerekiyordu. Bazı kontroller eklendi.

Sensörlerin gönderdiği açıların tanjant değerleriyle eğim bulnup aşağıdaki formülle iki sensörden çıkan doğruların denklemleri kuruldu ve kesişim noktaları hedefin lokasyonunu verdi.

$A(x_1, y_1)$ noktasından geçen ve eğimi m olan doğrunun denklemi:

$$y - y_1 = m(x - x_1)$$

Kullanılan Alyapı ve Çalıştırma Aşamaları

Java programlama dili ile Linux Ubuntu üzerinde Apache Maven yapısına sahip bir proje üzerinden ilerlendi. Eclipse IDE kullanıldı. Apache Kafka kütüphanelerinden haberleşme konfigürasyonları sağlandı. İki farklı program yapısı için Threadler kullanıldı. Proje daha uzun bir zaman aralığında çeşitli eklemeler ile görselleştirmeye ve detaylandırmaya da uygundur.

Çalıştırma gereksinimleri:

Sistemde apache kafka kurulu olup , program öncesinde zookeeper ve server yapıları aşağıdaki komutlarla çalıştırılır:

- bin/zookeeper-server-start.sh config/zookeeper.properties
- kafka-server-start.sh config/server.properties

İletişim için bir topic yapısı oluşturulur

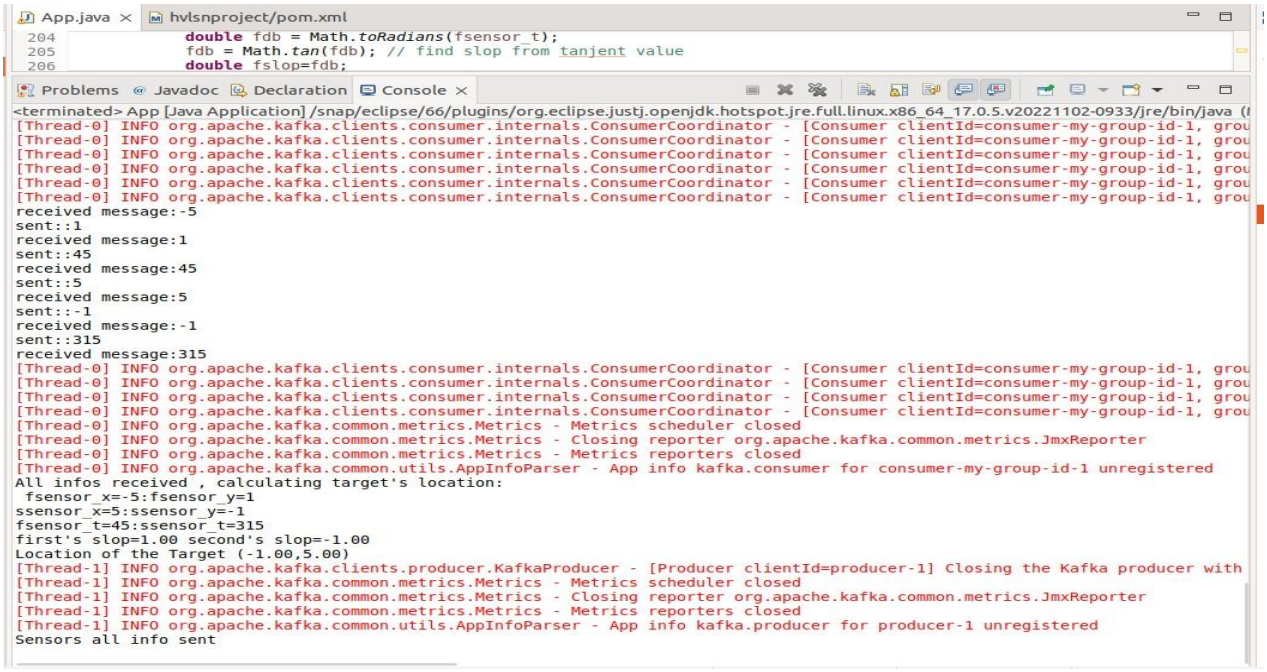
- kafka-topics.sh --bootstrap-server localhost:9092 --partitions 2 --replication-factor 1 --topic ftopic –create

App.java programı çalıştırılır ve çıktılar ekrana gelir.

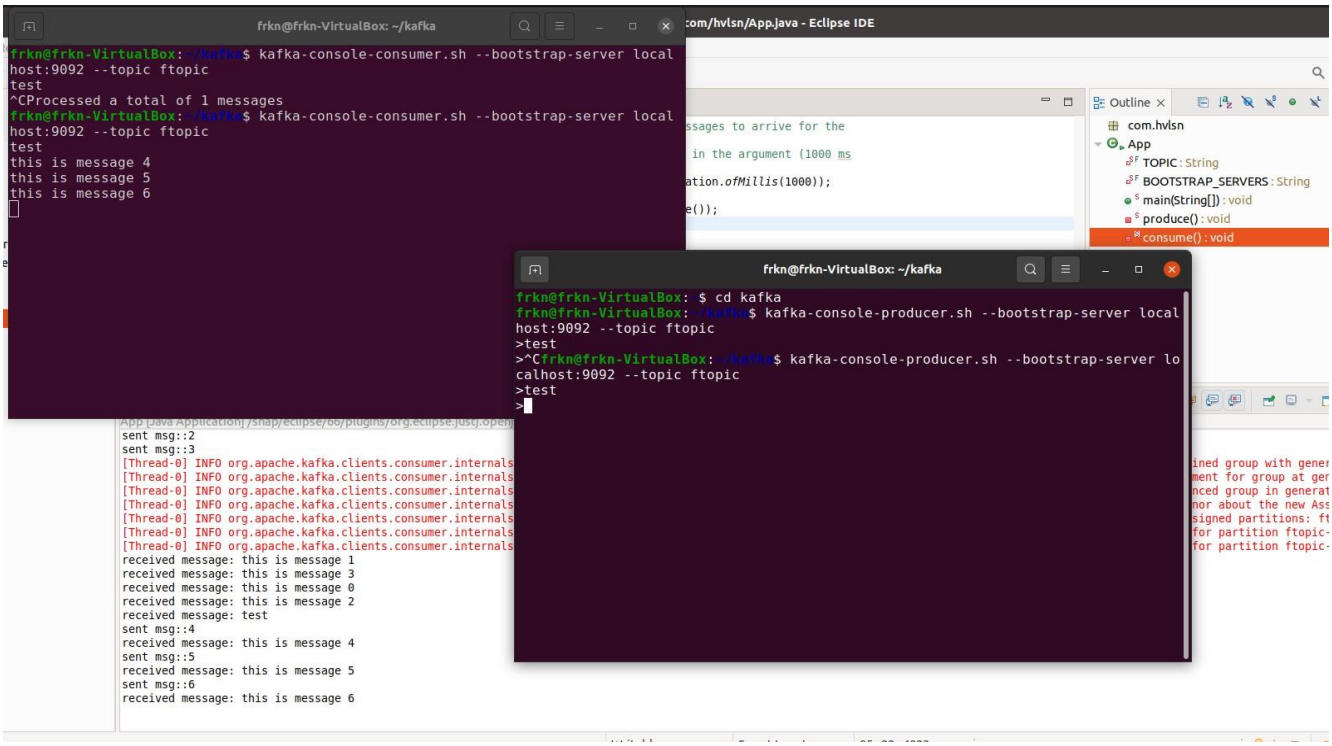
Sensör konfigürasyonları için alttaki lokasyondan manuel değişiklik yapılabilir veya random atama için comment yapılabilir // ile.

```
int fsensor_x = random.nextInt(max - min) + min; // sensor's x coordinate
int fsensor_y = random.nextInt(max - min) + min; // sensor's y coordinate
int fsensor_t = 68; // sensor's estimation direction degree to target


// second sensor
int ssensor_x = random.nextInt(max - min) + min;
int ssensor_y = random.nextInt(max - min) + min;
int ssensor_t = 423; // sensor's estimation direction degree to target

// Manually set for testing purposes , can be comment with // for random locations
fsensor_x=-5;fsensor_y=1;fsensor_t=45;
ssensor_x=5;ssensor_y=-1;ssensor_t=315;
```

- Aşağıdaki gibi başarılı çıktılar görülebilir:



Program terminal üzerinden başlatılabilen aynı topic e bağlı consumer programdan da takip edilebilir.



Kaynakça

https://www.derspresso.com.tr/matematik/dogrunun-analitigi/iki-dogrunun-kesisimi

https://docs.confluent.io/platform/current/clients/producer.html#learn-more

https://docs.confluent.io/platform/current/clients/consumer.html

https://kafka.apache.org/21/javadoc/org/apache/kafka/common/serialization/package-frame.html

Furkan Çelen

03/2023