The following is a Mobile Computing project. After going through the information answer the questions accordingly

In Project 1 you learnt about developing a monitoring application using on-board sensors in a smart phone. In this project, we will design a context aware adaptation strategy for the development of autonomous braking system for Level 3 autonomous cars which assumes an underlying distributed monitoring and actuation cyber-physical system specifically human-in-the-loop and human-in-the plant.

Before we delve into the details let us understand some basic definitions.

**Level 3 autonomy**: The vehicle operates autonomously, however, requires the driver to be attentive at all times for potential switch to manual mode.

**Controller**: It is a software that takes sensor data from the vehicle such as speed, distance between cars and acceleration or deceleration and outputs the braking pressure to apply deceleration to stop the car.

**Controller gain:** It is a property of the controller that determines how aggressively a controller will apply brakes and stop the car.

**Deceleration Limit**: This is the maximum limit on the deceleration applied by the controller. High deceleration can lead to unwanted harm to the driver behind the wheel. The controller can never apply deceleration that is higher than the maximum limit.

**Controller frequency**: It is the number of times the controller needs to compute a new braking input per second.

**Reaction time**: It is the time taken for the human to decide on an action after a switch to manual mode

**Action time**: Time taken to execute the action decided by the human and stop the car.

Autonomous vehicle braking system example

Figure 1: Autonomous Driving example

Consider a user driving a Level 3 autonomous car (Fig. 1). We are specifically focusing on the autonomous braking system. As a part of the autonomous braking system, there is an advisory control

that:

a) processes a driving scenario, by sensing environmental conditions

b) Predicts if the autonomous braking system will be able to stop the car

c) Determines if the control should be transferred to the human behind the wheel

Autonomous braking system has a feed-back loop controller that senses the distance between the autonomous car 𝐴 and the car in front 𝐼, and computes a braking force as a function of the 𝐴’s initial speed 𝑣!, and distance between 𝐴 and 𝐼. The braking force applied is also a function of the controller gain 𝐺, which is a design parameter that you will have to tune.

The braking force is constrained by the maximum deceleration limit 𝑎"#$, which is dependent on the road condition. For example, on dry road you can apply more deceleration than on wet road in rainy season. The braking force is thus given by the following equation:

𝐵𝑟𝑎𝑘𝑖𝑛𝑔 𝑓𝑜𝑟𝑐𝑒 𝑏 = min (𝑓(𝑣!, 𝑑!%, 𝐺), 𝑎"#$).

Once the braking force 𝑏 is applied, the vehicle kinematics show how the vehicle behaves in the real world. Given the initial speed 𝑣! and initial distance between 𝐴 and 𝐼, 𝑑!%, the vehicle kinematics gives the stopping distance 𝑑&'(), and stopping time 𝑡&'() for the vehicle. This you can obtain using the vehicle kinematics Simulink model (𝑀\*+,-./+) provided to you. If 𝑑&'() ≥ 𝑑!% then the autonomous vehicle fails, and the cars collide, else there is no collision. If cars collide, then stopping time 𝑡&'() is same as collision time 𝑡.

In this example, we will fix the distance between cars 𝐴 and 𝐼, 𝑑!%. However, changing initial velocity 𝑣!, controller gain 𝐺, and deceleration limit 𝑎"#$, can change the stopping distance and stopping time resulting in potential collisions.

**Human driver model**

The human driver is assumed to be alert all the time to take over control. However, we have to account for the reaction time 𝑡0 of the human driver after a switching signal is sent through the dashboard controls. In addition to the reaction time the human action also takes some time to execute. In this example, we assume that the human provides a deceleration that is 10% higher than the deceleration limit 𝑎"#$, i.e. ℎ# = 1.1𝑎"#$. The action time 𝑡# can be obtained by removing the braking controller from the Simulink model and applying a static deceleration of ℎ# into the vehicle kinematics model. After simulation the action time can be obtained as the stopping time reported in the Simulink model. The total time taken by the human to completely stop the car is ℎ&'() = 𝑡0 + 𝑡#. We are now in a position to define a basic logic for the advisory control. The advisory control uses the following algorithm to decide if it wants to switch:

SWITCH = ADVISORY CONTROL (𝑣!,𝐺,𝑎"#$, 𝑡0)

Step 1: Predict if autonomous braking control collides or not using 𝑣!, 𝐺, 𝑎𝑛𝑑 𝑎"#$.

Step 2: If no collision then

Step 3: Do not switch to manual control

Step 4: Else

Step 5: Collision time 𝑡.= predicted stopping time from Step 1

Step 6: Predict action time 𝑡#, using 𝑣!, remove controller, and constant

deceleration 1.1𝑎"#$.

Step 7: Predict reaction time 𝑡0

Step 8: Compute ℎ&'() = 𝑡0 + 𝑡#

Step 9: if 𝒉𝒔𝒕𝒐𝒑 < 𝒕𝒄

Step 10: Switch to human

Step 11: Else

Step 12: Do not switch.

Driving Context Changes

Road conditions depend on several factors including:

a) User mobility patterns

b) Traffic conditions

c) Weather patterns

Each context can be modeled using stochastic modeling techniques such as Markov chains. In this assignment you are given a composite model of driving context that has two states: a) normal with deceleration limit -200 " &+.! and b) poor road conditions with deceleration limit -150 " 678! .

The prior probabilities of each state 𝑝(𝑛𝑜𝑟𝑚𝑎𝑙) = 0.6, 𝑎𝑛𝑑 𝑝(𝑝𝑜𝑜𝑟) = 0.4. The transition probabilities are given by the matrix:

𝑞 =

𝑁𝑜𝑟𝑚𝑎𝑙 𝑃𝑜𝑜𝑟

𝑁𝑜𝑟𝑚𝑎𝑙 0.6 0.4

𝑃𝑜𝑜𝑟 0.85 0.15

**Human Reaction time model**

The cognitive workload of a human changes based on driving context. We will assume that normal driving conditions have low workload, and poor driving conditions require high workload. The heart rate and respiratory rate is provided for single task under low cognitive workload (LCW) and high cognitive workload (HCW) tasks. Use the mean and standard deviation to create a gaussian model of heart rate and respiratory rate. For each driving condition, the heart rate 𝐻𝑟 and respiratory rate 𝑅𝑟 is sampled from the gaussian model. The respiratory quotient can be computed as 𝑅𝑞 = 90. The human reaction time 𝑡0 can be computed using the following equation:

𝑡0 = 0.01 ∗ 𝑅𝑞

Q1) If the tight coupling between road conditions and cognitive workload is relaxed, what challenges might arise in predicting human reaction time (t\_r), and why? What additional data and modelling techniques might be necessary if road conditions no longer solely determine cognitive workload?

Q2) How might the shift from deterministic to probabilistic decision-making affect the advisory control system's performance in autonomous vehicles?

Q3) What are some potential implications of relying more on sensors and biometrics to assess driver readiness and cognitive load?

Q4) In terms of safety, what risks are associated with a system that cannot accurately gauge a driver's cognitive workload and make appropriate control handover decisions?

Q5) How might these changes impact the overall trust and usability of autonomous driving systems?

Q6) Based on the paper (attached in the project document), consider example 1- ECG sensor lifetime analysis. Discuss how you will achieve customization, and personalization in that example.

Customization and personalization in ECG sensor lifetime analysis can be achieved through the following methods:

**Individualized Monitoring Plans:**

Tailor monitoring plans based on an individual's medical history, current health conditions, and specific cardiac needs. Consider factors like age, existing heart conditions, medications, and other health data.

**Adjustable Monitoring Intervals:**

Allow for adjustable monitoring intervals based on the individual's health status. For instance, individuals with chronic heart conditions might require more frequent monitoring than those without any heart issues.

**Alert Threshold Customization:**

Enable customization of alert thresholds for abnormal ECG patterns. Individuals can set personalized thresholds based on their historical data and in consultation with their healthcare providers.

**User-Friendly Interfaces:**

Develop easy-to-use interfaces that allow individuals to view and interpret their ECG data easily. Graphs, trends, and summaries should be presented in a clear and understandable manner.

**Integration with Health Records:**

Integrate ECG sensor data with an individual's electronic health record (EHR) to provide a holistic view of their health. This integration can aid healthcare professionals in personalizing recommendations and treatments.

**Machine Learning and AI:**

Utilize machine learning algorithms to analyze ECG data and provide personalized insights. AI can identify unique patterns, predict anomalies, and suggest individualized recommendations.

**Behavioral and Lifestyle Integration:**

Incorporate data about an individual's lifestyle factors like diet, exercise, stress levels, and sleep patterns. Correlate this data with ECG readings to offer personalized advice for a heart-healthy lifestyle.

**Patient Involvement and Feedback:**

Encourage individuals to actively participate in the monitoring process and provide feedback. Their insights can be used to fine-tune the monitoring system and enhance personalization.

**Remote Monitoring Preferences:**

Allow individuals to choose their preferred method of remote monitoring, whether through a wearable device, smartphone app, or other monitoring systems that suit their lifestyle.

By incorporating these customization features, ECG sensor lifetime analysis can be personalized to cater to an individual's unique health profile, leading to more effective monitoring and better healthcare outcomes.