33rd Annual INCOSE
international symposium
hybrid event
Honolulu, HI, USA
July 15 - 20, 2023

Using the Unified Architecture Framework to Perform Hazard Analysis for Systems of Systems

Lars-Olof Kihlström
CAG Syntell
P.O.Box 10022, SE-10055 Stockholm,
Sweden
+46 706661978
lars.olof.kihlstrom@cag.se

Matthew Hause
Systems Solutions Inc (SSI)
3208 Misty Oaks Way, Round Rock, Texas
USA
+1 917 514 7581
mhause@systemxi.com

Joakim Fröberg
Safety Integrity AB
Hemdalsvägen 17, SE-72335, Västerås, Sweden
+46 705587055
joakim.froberg@safetyintegrity.se

Abstract. Hazard analysis for individual systems is a task that is required as part of the production of most complex systems to identify hazards and to take appropriate measures to alleviate such hazards. It is a laborious and expensive task that requires key specialists and lengthy analysis. This is not just for systems on their own. Since collaboration between systems is becoming increasingly important, this complicates hazard analysis even further. It requires a systemic process looking at the individual systems as well as the system of systems and the environment in which they operate. The example looks at the concept of platooning i.e., where trucks transporting goods can operate as a platoon, traveling very closely together under the control of a platoon leader. Platooning has the benefit of reducing traffic and harmful emissions but introduces new hazards that must be examined and mitigated. The paper describes an approach to hazard analysis for a **system** of systems (SoS) that uses a model created using the Unified Architecture Framework (UAF) as an aid in identifying and analyzing hazards.

Introduction

Hazard analysis for products where safety criticality is a factor is a requirement and needs to be carried out to identify hazards as well as to mitigate them. Performing hazard analysis involves the creation of scenarios that illustrate the hazards and makes it possible to define ways to mitigate and protect against them. Performing this kind of analysis is both difficult and time consuming. Attempting to perform hazard analysis for systems of systems is even more difficult. During 7 months in 2021 a project named Model-based Risk Assessment and Safety Analysis (MBRASA) was conducted to look at hazard analysis for system of systems. It involved a couple of companies as well as academia and was supported by government agencies. (TECOSA, 2021)

The main task was to look how modelling could be used to extrapolate on the single vehicle/machine scope of current safety standards, such that extended system/item definitions can be modelled to incorporate multiple systems and edge resources. The model was intended to aid in the definition of hazards for defined combinations of systems into a system of system. The model can be used to address a specific system of system issue namely that: A system of systems will exhibit a much larger hazard space (Functions, modes, failure modes, situations) than a single controlled system. The model therefore needs to address the system of systems specific issues. Based on the results achieved in this work effort further work was initiated to look in more detail at hazards resulting solely from the system of systems approach. The aim being to ensure that the hazards solely depending on the systems of systems could be looked at in isolation.

The modelling makes use of the Object Management Group (OMG) Unified Architecture Framework (UAF) as a modelling language with some additions defined for hazard definition. (OMG, 2022)

What Characterizes a System of System?

M.W. Maier described principles for architecting systems of systems (Maier, 1996). ISO/IEC/IEEE 21839 (ISO, 2019) also provides a definition of SoS: System of Systems (SoS) — Set of systems or system elements that interact to provide a unique capability that none of the constituent systems can accomplish on its own. Note: Systems elements can be necessary to facilitate the interaction of the constituent systems in the system of systems. The INCOSE Guide to the Systems Engineering Body of Knowledge (SEBoK*)* (INCOSE, 2022) created a set of characteristics that differ in between systems and system of systems (Svenson and Axelsson, 2021):

- A system usually has a well-defined set of stakeholders, whereas a system of systems usually has several levels of different stakeholders with mixed and sometimes contradictory and/or competing interests.
- A system usually has clear goals and purpose, whereas a system of systems usually has several, possibly contradictory goals and purposes.
- A system usually has clear operational priorities and mechanisms to manage these priorities, whereas a system of systems usually has several and sometimes different operational priorities and with no defined way of escalating any issues.
- A system usually has a single life cycle, whereas a system of systems usually has several lifecycles with elements that are implemented asynchronously.
- A system usually has a clearly defined ownership with the ability to transfer resources in between elements, whereas a system of systems usually has several owners that make decisions independently of one another.

The possibility of emergent behavior is also of importance here. Emergent behavior implies that interactions between the systems in an SoS reveals unanticipated behavior. In an SoS consisting of safety critical systems unanticipated emergent behavior can represent a serious hazard and such behavior therefore needs to be managed. The focus here is on the conditions that are solely due to the connection of a set of systems into a whole and the hazards that this creates.

Platooning.

Platooning implies that several trucks combine in a convoy under the control of a platoon leader where the distance in between trucks in the convoy can be much shorter than normal. They are even closer to one another than the distance that would be used based on adaptive cruise control (ACC). The advantages would be that congestion on roads would be alleviated. Given that the trucks in the platoon are shielded in front will also help reduce fuel consumption which would have an environmental benefit.

The most advanced form of platooning would be one where trucks could be allowed to form platoons dynamically on roads and broadcast their willingness either to lead a platoon or join a platoon. This is also the form of platooning that will be used here as an example of system of system hazard analysis. For this to be possible a set of different capabilities needs to exist in the trucks that make up a platoon as shown in figure 2.

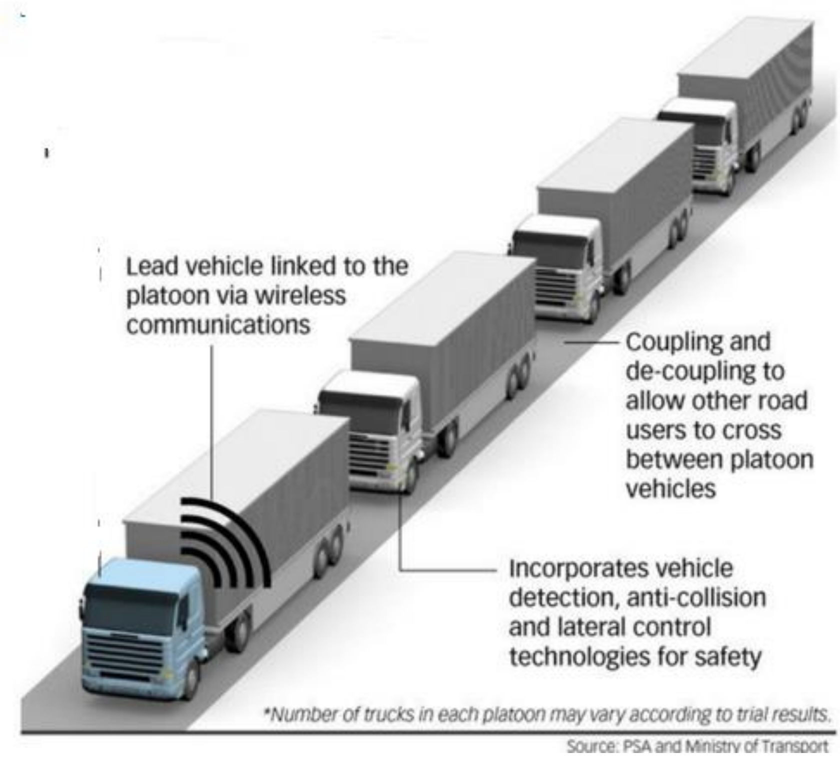

Lead vehicle linked to the platoon via wireless communications
Coupling and de-coupling to allow other road users to cross between platoon vehicles
Incorporates vehicle detection, anti-collision and lateral control technologies for safety
*Number of trucks in each platoon may vary according to trial results.
Source: PSA and Ministry of Transport

Figure 1. Truck Platooning Concept

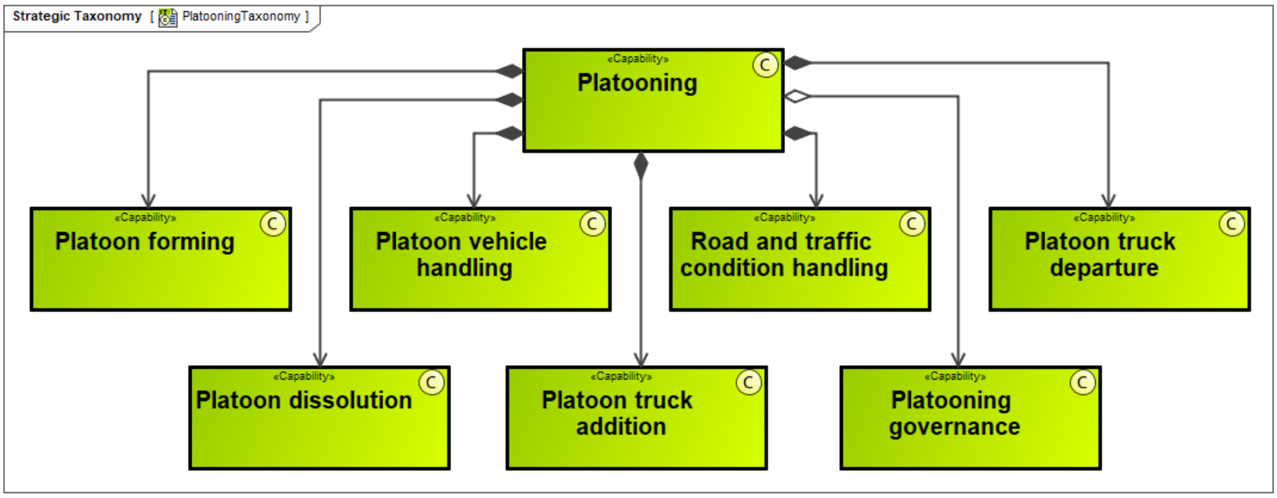

Strategic Taxonomy [PlatooningTaxonomy]
«Capability» Platooning
«Capability» Platoon forming
«Capability» Platoon vehicle handling
«Capability» Road and traffic condition handling
«Capability» Platoon truck departure
«Capability» Platoon dissolution
«Capability» Platoon truck addition
«Capability» Platooning governance

Figure 2. Truck Platooning Concept Capabilities

The figure shows a set of capabilities that platooning needs to exhibit. These are capabilities that will need to be exhibited by individual trucks, the platoon, external systems (governance) and the SoS as a whole.

Do trucks formed into a platoon represent a system of systems (Baumgart et al, 2017)?
The kind of platoon that can form dynamically on a road by cooperating trucks matches the criteria for an SoS. Trucks can have different manufacturers. They can have different usages defined by their owners such as logistics companies, contractors as well as different authorities. By using the general term trucks to accommodate all these possibilities the different SoS criteria's can be analyzed.

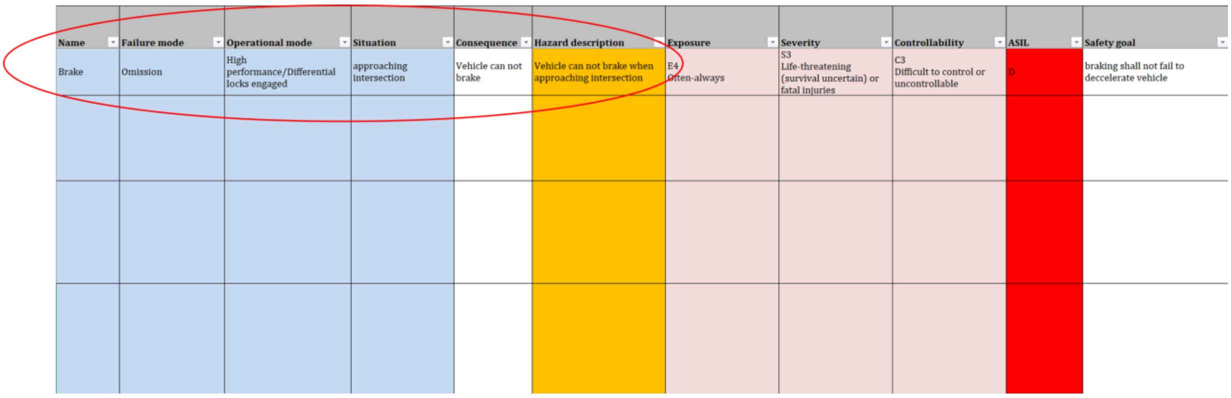
- A system of systems usually has several levels of different stakeholders with mixed and sometimes contradictory and/or competing interests.
    - General trucks would meet this criterion.
- A system of systems usually has several, possibly contradictory goals and purposes.
    - General trucks would also meet this criterion.

- A system of systems usually has several and sometimes different operational priorities and with no defined way escalating any issues.
    - General trucks would meet this criterion as well.
- A system of systems usually has several lifecycles with elements that are implemented asynchronously.
    - The life cycle of general trucks will be different.
- A system of systems usually has several owners and drivers that make decisions independently of one another.
    - The owners and drivers of general trucks would also take independent decisions.

The criteria seem to be met. It is however clear that a platoon of trucks needs to have specified controlling mechanisms to achieve the capability to perform platooning safely. The behavior that would result without any such ability would be catastrophic. Standardization of platooning control is essential. In the same way an ability to test the implementation of this standard for trucks allowed to act as a platoon leader or platoon participants is also a requirement. Despite this overall control a platoon that can be created dynamically by a set of general trucks can still be considered as a system of systems.

Identifying and dealing with hazards

Generally, hazard analysis and risk assessment (HARA) is performed by exploring all possible failure in all possible usage situations, and then estimate the criticality of each dangerous consequence (hazard). Below is an example of using the ISO 26262 method for HARA where parameters are put in different columns in a large table that can be dealt with to assess hazards and possible actions as well as severity. (ISO, 2018), (Axelsson, 2017)



Name	Failure mode	Operational mode	Situation	Consequence	Hazard description	Exposure	Severity	Controllability	ASIL	Safety goal
Brake	Omission	High performance/Differential locks engaged	approaching intersection	Vehicle can not brake	Vehicle can not brake when approaching intersection	E4 Often-always	S3 Life-threatening (survival uncertain) or fatal injuries	C3 Difficult to control or uncontrollable	D	braking shall not fail to deccelerate vehicle

Figure 3. Hazard analysis and risk assessment

The first column in this table defines the function concerned, the second deals with failure modes. The third considers operational modes, the fourth situations and the fifth discusses consequence. A situation crossed with a consequence (taking a failure mode and an operational mode into account) yields a hazard. Each hazard is assessed for severity, exposure, controllability and an ASIL level is determined. Finally, a safety goal is a requirement that if it is upheld keeps the system from exhibiting the hazard. Obviously, the cross-combination of all the blue columns can yield a very large hazard space even for a single system. The example of a brake-based hazard above can be viewed as a hazard affecting a single vehicle. If this is expanded to a system of systems, the hazard space becomes much larger.

Using a model to analyze hazards for SoS

As described above a controlled platoon of trucks can be considered as a system of systems but where a platoon leader control has been added to manage the interactions between the trucks such that a platoon can be handled safely. The intent here is to consider hazards and safety goals associated with the platoon and leave the hazard handling of individual trucks to the hazard handling associated with the individual trucks themselves. Once the platoon specific hazards have been considered it will be possible to look at the individual hazard analysis for the trucks to define how the individual hazard handling for trucks impacts on the hazards for the platoon. Of specific interest however för an SoS are hazards that appear solely because of the SoS and where there is no failure in the individual trucks but where the hazards are directly connected platooning. Using a model of a platoon such hazards can be analyzed.

The relatively simple model in Figure 3 describes the influence of external elements as well as the influence that the trucks in the platoon have on one another.

- The weather influences the trucks directly as well as the road on which they travel (snow, rain, ice, fog, heat, cold).
- The road with its changing number of lanes, gradients, speed restrictions and road works will impact of the platoon.
- The traffic that is not part of the platoon will need to be dealt with. The kind of vehicle that interacts with the platoon may well need different handling (police, ambulance, fire brigade, military vehicles, other trucks, civilian vehicles etc.).
- An overall platoon governance entity has been added since there may well be a need for an overall platooning control for a region. It can provide governance for the platoons in the region and provide data regarding conditions beyond immediate line of sight for a given platoon leader.

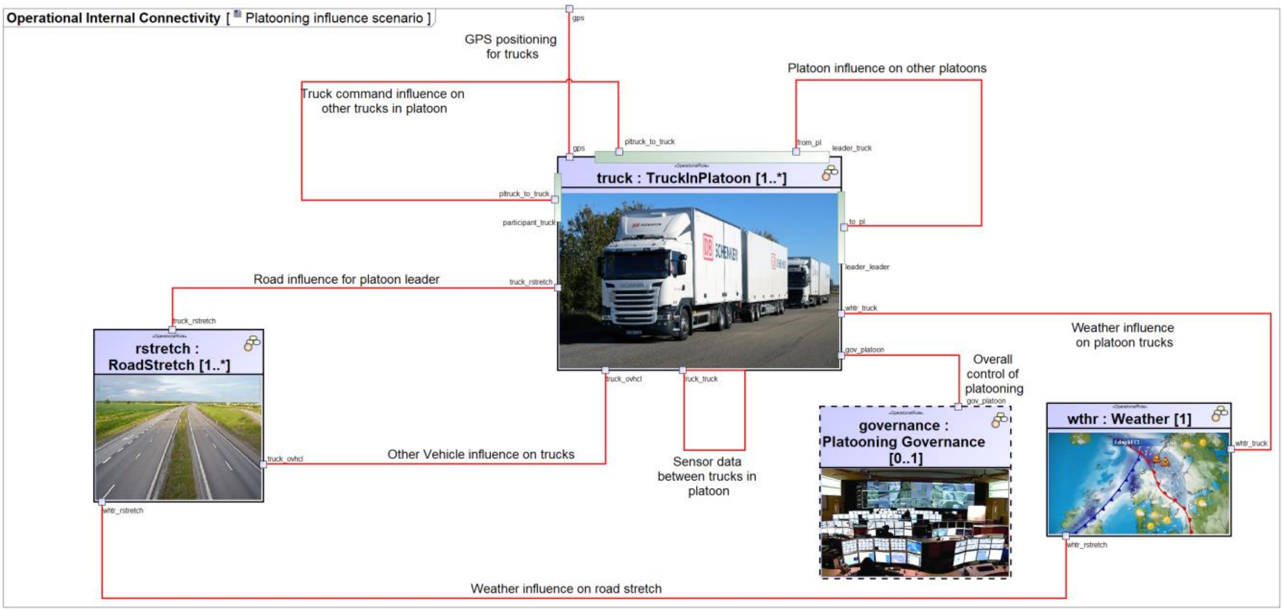

Operational Internal Connectivity [Platooning influence scenario]
gps
GPS positioning for trucks
Truck command influence on other trucks in platoon
Platoon influence on other platoons
gps
pltruck_to_truck
from_pl
leader_truck
truck : TruckInPlatoon [1..*]
pltruck_to_truck
participant_truck
to_pl
leader_leader
Road influence for platoon leader
truck_rstretch
truck_rstretch
whtr_truck
rstretch : RoadStretch [1..*]
Weather influence on platoon trucks
gov_platoon
Overall control of platooning
gov_platoon
truck_ovhcl
truck_truck
governance : Platooning Governance [0..1]
wthr : Weather [1]
whtr_truck
truck_ovhcl
Other Vehicle influence on trucks
Sensor data between trucks in platoon
whtr_rstretch
whtr_rstretch
Weather influence on road stretch

Figure 3. Platooning Influence Scenario

Truck constraints that need to be considered within a platoon.

The following rules and regulations act as constraints on trucks within a platoon:

- Trucks have a regulated maximum length. The length differs between different countries, but 25 meters is a reasonable assumption concerning a truck maximum length.
- As the number of trucks in the platoon increase a leader follower approach as regards steering needs to be employed such that steering follows both lanes within the road as well as what the truck directly in front is doing.
- Trucks also have a maximum weight. Also, this can vary between countries and is furthermore subject to regulatory changes. A maximum weight of 60 tons is a reasonable assumption.
- If trucks in a platoon have different maximum power ability, gaps within the platoon may appear as the incline is negotiated. As an example, a 200-meter incline can be negotiated in 8 seconds by a truck capable of maintaining the speed 25 m/s (90 km/h). A truck that is only capable of 20 m/s (72 km/h) will only travel 160 meters in 8 seconds which would yield a gap

of 40 meters in between the trucks. A platoon with such gaps appearing will be very difficult to control.

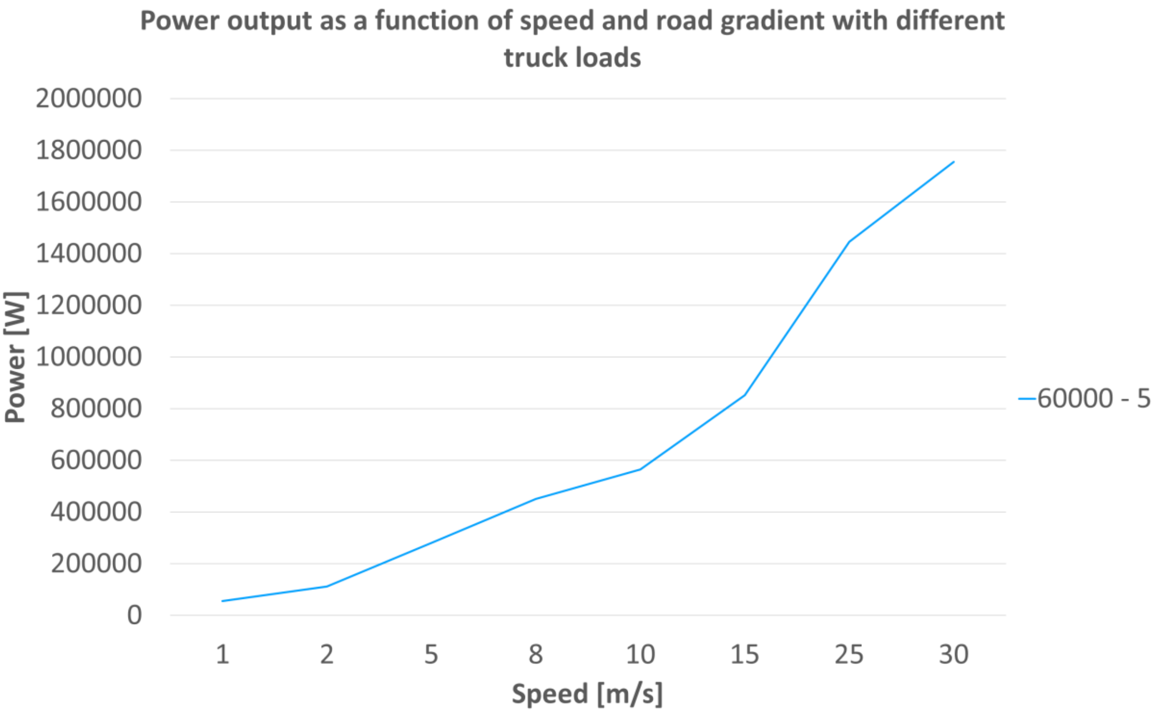


Figure 4. Power required by a 60-ton truck to maintain speeds on a 5-degree incline.

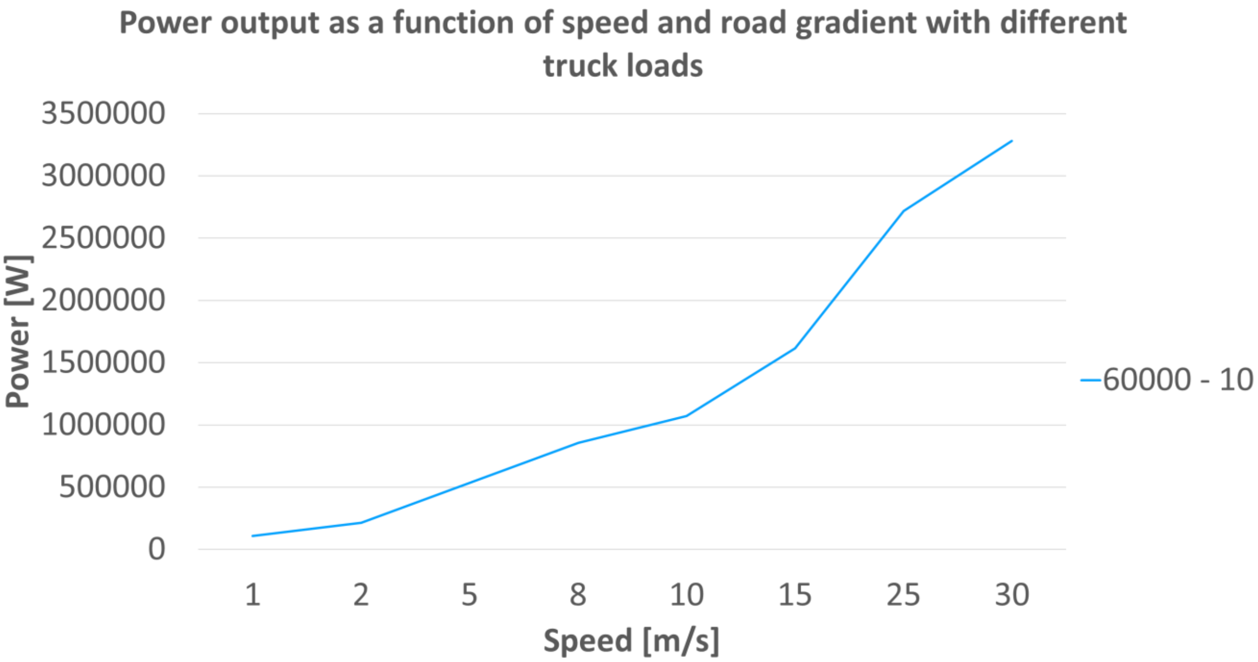


Figure 5. Power required by a 60-ton truck to maintain speeds on a 10-degree incline.

- If a platoon is to alleviate road congestion to any extent it must operate with a distance in between trucks that is less than that provided by an adaptive cruise control. If this is not the case, there is little benefit for the truck in participating in a platoon. If a truck can drive with a smaller distance to the truck in front of it in a platoon, this implies that any ACC would have to be disabled while platooning is in progress.

- The maximum power that a truck can deliver is of primary importance as far as platooning is concerned. If this differs in between trucks the speed that a truck can have going uphill can differ significantly in between the trucks in a platoon as is shown in figures 4 and 5.

Using a logical model of a Platoon to Determine Hazards

The requirements concerning the handling of the platoon can be formalized as a logical need to exchange different information as well as commands. Taking account of the capabilities defined in Figure 2 makes it possible to formalize these exchanges in a logical model. (Axelsson, 2017)

Figure 6 describes the individual truck in the platoon, its interactions with other trucks in the platoon as well as the road and road conditions and external elements such at the weather and GPS satellite systems. Messages are exchanged between the different truck as well as other context elements.

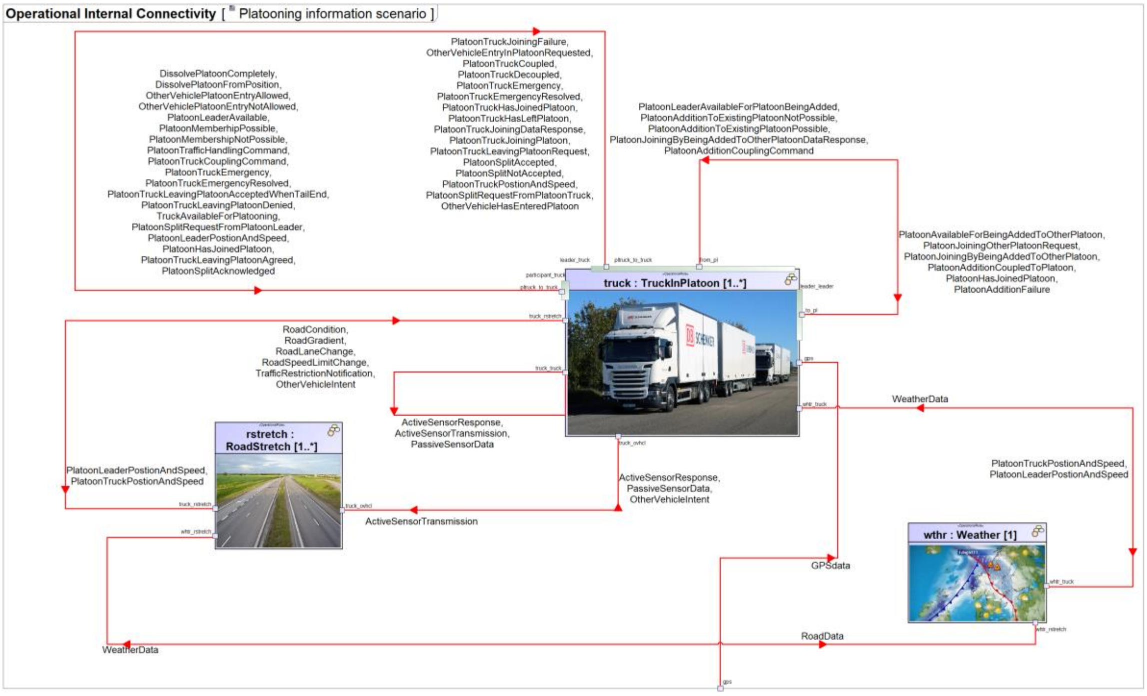

Operational Internal Connectivity [Platooning information scenario]
DissolvePlatoonCompletely, DissolvePlatoonFromPosition, OtherVehiclePlatoonEntryAllowed, OtherVehiclePlatoonEntryNotAllowed, PlatoonLeaderAvailable, PlatoonMemberhipPossible, PlatoonMembershipNotPossible, PlatoonTrafficHandlingCommand, PlatoonTruckCouplingCommand, PlatoonTruckEmergency, PlatoonTruckEmergencyResolved, PlatoonTruckLeavingPlatoonAcceptedWhenTailEnd, PlatoonTruckLeavingPlatoonDenied, TruckAvailableForPlatooning, PlatoonSplitRequestFromPlatoonLeader, PlatoonLeaderPostionAndSpeed, PlatoonHasJoinedPlatoon, PlatoonTruckLeavingPlatoonAgreed, PlatoonSplitAcknowledged
PlatoonTruckJoiningFailure, OtherVehicleEntryInPlatoonRequested, PlatoonTruckCoupled, PlatoonTruckDecoupled, PlatoonTruckEmergency, PlatoonTruckEmergencyResolved, PlatoonTruckHasJoinedPlatoon, PlatoonTruckHasLeftPlatoon, PlatoonTruckJoiningDataResponse, PlatoonTruckJoiningPlatoon, PlatoonTruckLeavingPlatoonRequest, PlatoonSplitAccepted, PlatoonSplitNotAccepted, PlatoonTruckPostionAndSpeed, PlatoonSplitRequestFromPlatoonTruck, OtherVehicleHasEnteredPlatoon
PlatoonLeaderAvailableForPlatoonBeingAdded, PlatoonAdditionToExistingPlatoonNotPossible, PlatoonAdditionToExistingPlatoonPossible, PlatoonJoiningByBeingAddedToOtherPlatoonDataResponse, PlatoonAdditionCouplingCommand
PlatoonAvailableForBeingAddedToOtherPlatoon, PlatoonJoiningOtherPlatoonRequest, PlatoonJoiningByBeingAddedToOtherPlatoon, PlatoonAdditionCoupledToPlatoon, PlatoonHasJoinedPlatoon, PlatoonAdditionFailure
leader_truck
pltruck_to_truck
from_pl
participant_truck
pltruck_to_truck
leader_leader
truck : TruckInPlatoon [1..*]
truck_rstretch
to_pl
RoadCondition, RoadGradient, RoadLaneChange, RoadSpeedLimitChange, TrafficRestrictionNotification, OtherVehicleIntent
truck_truck
gps
wthr_truck
WeatherData
ActiveSensorResponse, ActiveSensorTransmission, PassiveSensorData
truck_ovhcl
rstretch : RoadStretch [1..*]
PlatoonLeaderPostionAndSpeed, PlatoonTruckPostionAndSpeed
ActiveSensorResponse, PassiveSensorData, OtherVehicleIntent
PlatoonTruckPostionAndSpeed, PlatoonLeaderPostionAndSpeed
truck_rstretch
truck_ovhcl
ActiveSensorTransmission
wthr_rstretch
wthr : Weather [1]
GPSdata
wthr_truck
wthr_rstretch
RoadData
WeatherData
gps

Figure 6. Formalized Commands and Information Exchanges Based on Requirements.

It is important to realize that this logic needs to allow a truck to be either a platoon leader or a platoon participant truck and that a participant may change to a leader and that a leader may change to be a participant. The most compact way to define the logic involved as well as to identify the hazards involved is to look at the state machine for the truck as part of the platoon as shown in Figure 8. Figure 7 shows the concentrated platooning logic for the truck.

It is important to realize that this logic needs to allow a truck to be either a platoon leader or a platoon participant truck and that a participant may change to a leader and that a leader may change to be a participant. The most compact way to define the logic involved as well as to identify the hazards involved is to look at the state machine for the truck as part of the platoon. The possible governance is not included here since this will need to be considered further. The state machine for the truck, shown in figure 8, enables a detailed reasoning about the logic as well as the hazards involved. This is a compact state machine that contains several different concurrent regions that each cover various aspects of truck being a part of a platoon or leading a platoon.

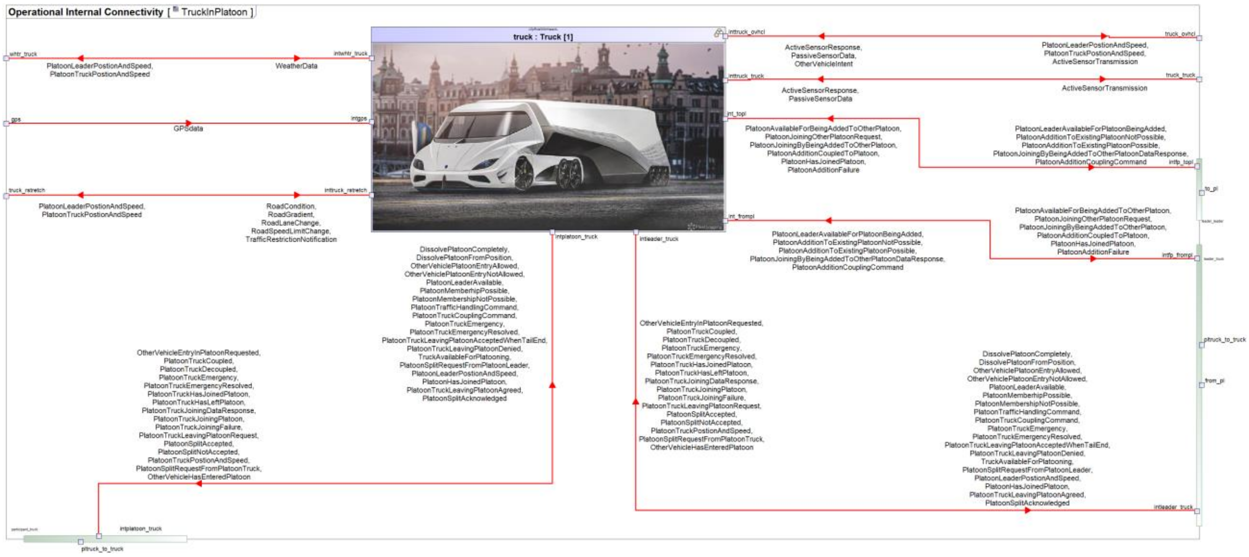

Operational Internal Connectivity [TruckInPlatoon]
truck : Truck [1]
whtr_truck
intwhtr_truck
PlatoonLeaderPostionAndSpeed, PlatoonTruckPostionAndSpeed
WeatherData
gps
intgps
GPSdata
truck_rstretch
inttruck_rstretch
PlatoonLeaderPostionAndSpeed, PlatoonTruckPostionAndSpeed
RoadCondition, RoadGradient, RoadLaneChange, RoadSpeedLimitChange, TrafficRestrictionNotification
inttruck_ovhcl
truck_ovhcl
ActiveSensorResponse, PassiveSensorData, OtherVehicleIntent
PlatoonLeaderPostionAndSpeed, PlatoonTruckPostionAndSpeed, ActiveSensorTransmission
inttruck_truck
truck_truck
ActiveSensorResponse, PassiveSensorData
ActiveSensorTransmission
int_topl
PlatoonAvailableForBeingAddedToOtherPlatoon, PlatoonJoiningOtherPlatoonRequest, PlatoonJoiningByBeingAddedToOtherPlatoon, PlatoonAdditionCoupledToPlatoon, PlatoonHasJoinedPlatoon, PlatoonAdditionFailure
PlatoonLeaderAvailableForPlatoonBeingAdded, PlatoonAdditionToExistingPlatoonNotPossible, PlatoonAdditionToExistingPlatoonPossible, PlatoonJoiningByBeingAddedToOtherPlatoonDataResponse, PlatoonAdditionCouplingCommand
intfp_topl
to_pl
int_frompl
PlatoonLeaderAvailableForPlatoonBeingAdded, PlatoonAdditionToExistingPlatoonNotPossible, PlatoonAdditionToExistingPlatoonPossible, PlatoonJoiningByBeingAddedToOtherPlatoonDataResponse, PlatoonAdditionCouplingCommand
PlatoonAvailableForBeingAddedToOtherPlatoon, PlatoonJoiningOtherPlatoonRequest, PlatoonJoiningByBeingAddedToOtherPlatoon, PlatoonAdditionCoupledToPlatoon, PlatoonHasJoinedPlatoon, PlatoonAdditionFailure
intfp_frompl
intplatoon_truck
intleader_truck
DissolvePlatoonCompletely, DissolvePlatoonFromPosition, OtherVehiclePlatoonEntryAllowed, OtherVehiclePlatoonEntryNotAllowed, PlatoonLeaderAvailable, PlatoonMemberhipPossible, PlatoonMembershipNotPossible, PlatoonTrafficHandlingCommand, PlatoonTruckCouplingCommand, PlatoonTruckEmergency, PlatoonTruckEmergencyResolved, PlatoonTruckLeavingPlatoonAcceptedWhenTailEnd, PlatoonTruckLeavingPlatoonDenied, TruckAvailableForPlatooning, PlatoonSplitRequestFromPlatoonLeader, PlatoonLeaderPostionAndSpeed, PlatoonHasJoinedPlatoon, PlatoonTruckLeavingPlatoonAgreed, PlatoonSplitAcknowledged
OtherVehicleEntryInPlatoonRequested, PlatoonTruckCoupled, PlatoonTruckDecoupled, PlatoonTruckEmergency, PlatoonTruckEmergencyResolved, PlatoonTruckHasJoinedPlatoon, PlatoonTruckHasLeftPlatoon, PlatoonTruckJoiningDataResponse, PlatoonTruckJoiningPlatoon, PlatoonTruckJoiningFailure, PlatoonTruckLeavingPlatoonRequest, PlatoonSplitAccepted, PlatoonSplitNotAccepted, PlatoonTruckPostionAndSpeed, PlatoonSplitRequestFromPlatoonTruck, OtherVehicleHasEnteredPlatoon
OtherVehicleEntryInPlatoonRequested, PlatoonTruckCoupled, PlatoonTruckDecoupled, PlatoonTruckEmergency, PlatoonTruckEmergencyResolved, PlatoonTruckHasJoinedPlatoon, PlatoonTruckHasLeftPlatoon, PlatoonTruckJoiningDataResponse, PlatoonTruckJoiningPlatoon, PlatoonTruckJoiningFailure, PlatoonTruckLeavingPlatoonRequest, PlatoonSplitAccepted, PlatoonSplitNotAccepted, PlatoonTruckPostionAndSpeed, PlatoonSplitRequestFromPlatoonTruck, OtherVehicleHasEnteredPlatoon
DissolvePlatoonCompletely, DissolvePlatoonFromPosition, OtherVehiclePlatoonEntryAllowed, OtherVehiclePlatoonEntryNotAllowed, PlatoonLeaderAvailable, PlatoonMemberhipPossible, PlatoonMembershipNotPossible, PlatoonTrafficHandlingCommand, PlatoonTruckCouplingCommand, PlatoonTruckEmergency, PlatoonTruckEmergencyResolved, PlatoonTruckLeavingPlatoonAcceptedWhenTailEnd, PlatoonTruckLeavingPlatoonDenied, TruckAvailableForPlatooning, PlatoonSplitRequestFromPlatoonLeader, PlatoonLeaderPostionAndSpeed, PlatoonHasJoinedPlatoon, PlatoonTruckLeavingPlatoonAgreed, PlatoonSplitAcknowledged
pltruck_to_truck
from_pl
intleader_truck
intplatoon_truck
pltruck_to_truck

Figure 7. The concentrated platooning logic.

The state machine uses a set of concurrent regions to describe the platooning logic. The following set of basic regions are present:

- **Common (see figure 8)**:
  This region deals with interactions applicable irrespective of whether the truck is a platoon leader, platoon truck or a truck hoping to be part of a platoon.
- **PlatoonLeader (see figure 8)**:
  This region contains a set of interactions that a platoon leader needs to be able to deal with and is in turn subdivided into a set of concurrent regions:
  - HandlingTrucksJoining (see figure 9)
  - HandlingTrucksLeaving (see figure 13)
  - HandlingEntryOfOtherVehicles (see figure 13)
  - PlatoonTruckHandling (see figure 12)
  - PlatoonInterestedInJoiningOtherPlatoon (see figure 14)
  - PlatoonCapableOfAddingPlatoon (see figure 14)
- **WouldBePlatoonTruck (see figure 9)**:
  This region deals with interactions performed by a truck that wishes to become a member of a platoon.
- **PlatoonTruck (see figure 11)**:
  This region deals with interactions performed by a tuck that has become a member of a platoon.

The regions are coordinated by the value properties owned by the truck block that all of them access and manipulate. The details as well as the hazards can be identified by looking at the detailed combinations of interactions supported by a platoon truck and a platoon leader. The interactions required to manage a platoon composed of trucks from different possible manufacturers and users will require detailed standardization as well as regular inspections by authorities. Within the Figure 8 state machine, references to figures, that analyze and look at hazards associated with a given part of the total state machine, are included.

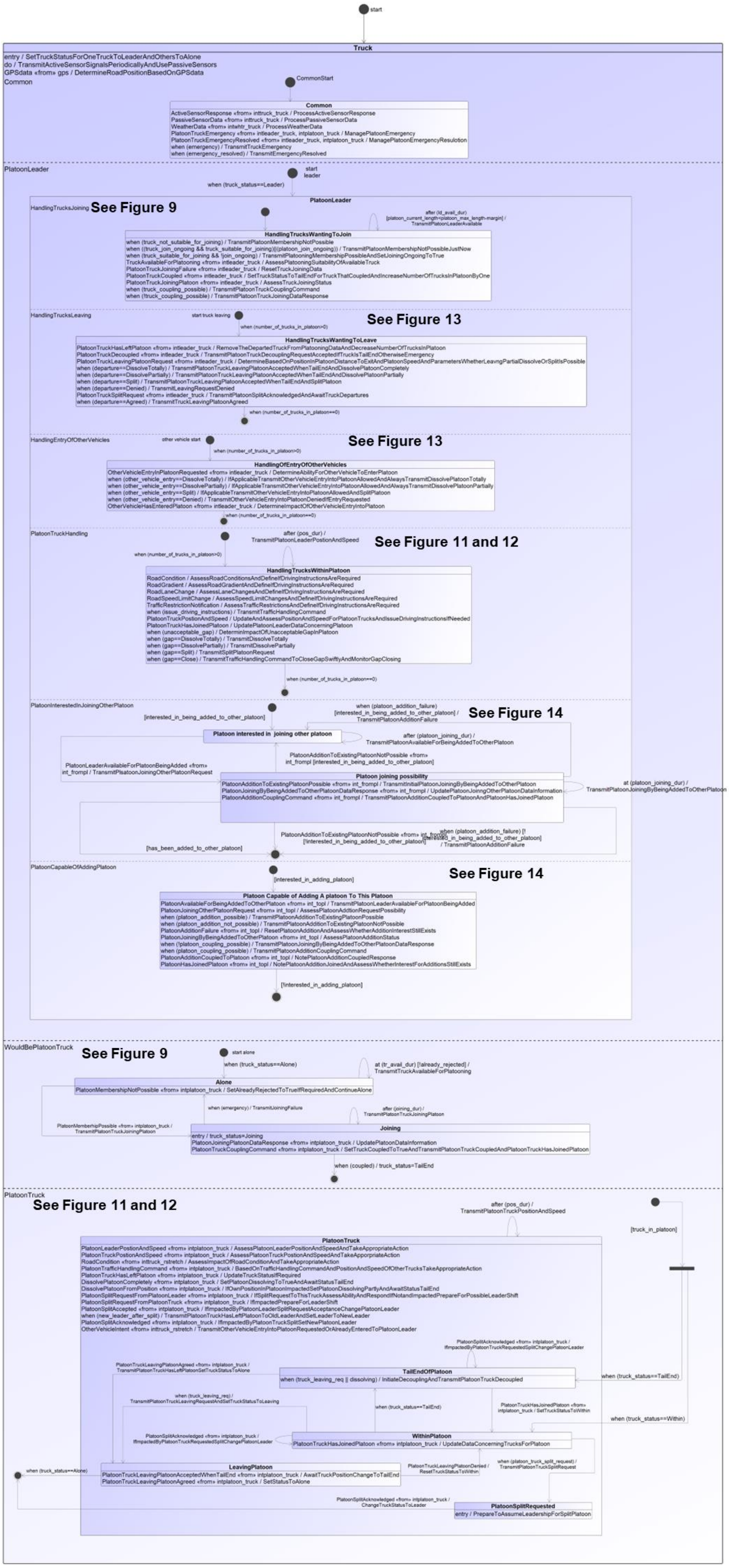

start
Truck
entry / SetTruckStatusForOneTruckToLeaderAndOthersToAlone
do / TransmitActiveSensorSignalsPeriodicallyAndUsePassiveSensors
GPSdata «from» gps / DetermineRoadPositionBasedOnGPSdata
Common
CommonStart
Common
ActiveSensorResponse «from» inttruck_truck / ProcessActiveSensorResponse
PassiveSensorData «from» inttruck_truck / ProcessPassiveSensorData
WeatherData «from» intwhtr_truck / ProcessWeatherData
PlatoonTruckEmergency «from» intleader_truck, intplatoon_truck / ManagePlatoonEmergency
PlatoonTruckEmergencyResolved «from» intleader_truck, intplatoon_truck / ManagePlatoonEmergencyResulotion
when (emergency) / TransmitTruckEmergency
when (emergency_resolved) / TransmitEmergencyResolved
PlatoonLeader
start leader
when (truck_status==Leader)
PlatoonLeader
HandlingTrucksJoining
See Figure 9
after (ld_avail_dur) [platoon_current_length<platoon_max_length-margin] / TransmitPlatoonLeaderAvailable
HandlingTrucksWantingToJoin
when (truck_not_sutaible_for_joining) / TransmitPlatoonMembershipNotPossible
when ((truck_join_ongoing && truck_suitable_for_joining)||(platoon_join_ongoing)) / TransmitPlatoonMembershipNotPossibleJustNow
when (truck_suitable_for_joining && !join_ongoing) / TransmitPlatooningMembershipPossibleAndSetJoiningOngoingToTrue
TruckAvailableForPlatooning «from» intleader_truck / AssessPlatooningSuitabilityOfAvailableTruck
PlatoonTruckJoiningFailure «from» intleader_truck / ResetTruckJoiningData
PlatoonTruckCoupled «from» intleader_truck / SetTruckStatusToTailEndForTruckThatCoupledAndIncreaseNumberOfTrucksInPlatoonByOne
PlatoonTruckJoiningPlatoon «from» intleader_truck / AssessTruckJoiningStatus
when (truck_coupling_possible) / TransmitPlatoonTruckCouplingCommand
when (!truck_coupling_possible) / TransmitPlatoonTruckJoiningDataResponse
HandlingTrucksLeaving
start truck leaving
See Figure 13
when (number_of_trucks_in_platoon>0)
HandlingTrucksWantingToLeave
PlatoonTruckHasLeftPlatoon «from» intleader_truck / RemoveTheDepartedTruckFromPlatooningDataAndDecreaseNumberOfTrucksInPlatoon
PlatoonTruckDecoupled «from» intleader_truck / TransmitPlatoonTruckDecouplingRequestAcceptedIfTruckIsTailEndOtherwiseEmergency
PlatoonTruckLeavingPlatoonRequest «from» intleader_truck / DetermineBasedOnPositionInPlatoonDistanceToExitAndPlatoonSpeedAndParametersWhetherLeavngPartialDissolveOrSplitIsPossible
when (departure==DissolveTotally) / TransmitPlatoonTruckLeavingPlatoonAcceptedWhenTailEndAndDissolvePlatoonCompletely
when (departure==DissolvePartially) / TransmitPlatoonTruckLeavingPlatoonAcceptedWhenTailEndAndDissolvePlatoonPartially
when (departure==Split) / TransmitPlatoonTruckLeavingPlatoonAcceptedWhenTailEndAndSplitPlatoon
when (departure==Denied) / TransmitLeavingRequestDenied
PlatoonTruckSplitRequest «from» intleader_truck / TransmitPlatoonSplitAcknowledgedAndAwaitTruckDepartures
when (departure==Agreed) / TransmitTruckLeavingPlatoonAgreed
when (number_of_trucks_in_platoon==0)
HandlingEntryOfOtherVehicles
other vehicle start
See Figure 13
when (number_of_trucks_in_platoon>0)
HandlingOfEntryOfOtherVehicles
OtherVehicleEntryInPlatoonRequested «from» intleader_truck / DetermineAbilityForOtherVehicleToEnterPlatoon
when (other_vehicle_entry==DissolveTotally) / IfApplicableTransmitOtherVehicleEntryIntoPlatoonAllowedAndAlwaysTransmitDissolvePlatoonTotally
when (other_vehicle_entry==DissolvePartially) / IfApplicableTransmitOtherVehicleEntryIntoPlatoonAllowedAndAlwaysTransmitDissolvePlatoonPartially
when (other_vehicle_entry==Split) / IfApplicableTransmitOtherVehicleEntryIntoPlatoonAllowedAndSplitPlatoon
when (other_vehicle_entry==Denied) / TransmitOtherVehicleEntryIntoPlatoonDeniedIfEntryRequested
OtherVehicleHasEnteredPlatoon «from» intleader_truck / DetermineImpactOfOtherVehicleEntryIntoPlatoon
when (number_of_trucks_in_platoon==0)
PlatoonTruckHandling
after (pos_dur) / TransmitPlatoonLeaderPostionAndSpeed
See Figure 11 and 12
when (number_of_trucks_in_platoon>0)
HandlingTrucksWithinPlatoon
RoadCondition / AssessRoadConditionsAndDefineIfDrivingInstructionsAreRequired
RoadGradient / AssessRoadGradientAndDefineIfDrivingInstructionsAreRequired
RoadLaneChange / AssessLaneChangesAndDefineIfDrivingInstructionsAreRequired
RoadSpeedLimitChange / AssessSpeedLimitChangesAndDefineIfDrivingInstructionsAreRequired
TrafficRestrictionNotification / AssessTrafficRestrictionsAndDefineIfDrivingInstructionsAreRequired
when (issue_driving_instructions) / TransmitTrafficHandlingCommand
PlatoonTruckPostionAndSpeed / UpdateAndAssessPositionAndSpeedForPlatoonTrucksAndIssueDrivingInstructionsIfNeeded
PlatoonTruckHasJoinedPlatoon / UpdatePlatoonLeaderDataConcerningPlatoon
when (unacceptable_gap) / DeterminImpactOfUnacceptableGapInPlatoon
when (gap==DissolveTotally) / TransmitDissolveTotally
when (gap==DissolvePartially) / TransmitDissolvePartially
when (gap==Split) / TransmitSplitPlatoonRequest
when (gap==Close) / TransmitTrafficHandlingCommandToCloseGapSwiftlyAndMonitorGapClosing
when (number_of_trucks_in_platoon==0)
PlatoonInterestedInJoiningOtherPlatoon
[interested_in_being_added_to_other_platoon]
when (platoon_addition_failure) [interested_in_being_added_to_other_platoon] / TransmitPlatoonAdditionFailure
See Figure 14
Platoon interested in joining other platoon
after (platoon_joining_dur) / TransmitPlatoonAvailableForBeingAddedToOtherPlatoon
PlatoonAdditionToExistingPlatoonNotPossible «from» int_frompl [interested_in_being_added_to_other_platoon]
PlatoonLeaderAvailableForPlatoonBeingAdded «from» int_frompl / TransmitPlsatoonJoiningOtherPlatoonRequest
Platoon joining possibility
PlatoonAdditionToExistingPlatoonPossible «from» int_frompl / TransmitInitialPlatoonJoiningByBeingAddedToOtherPlatoon
PlatoonJoiningByBeingAddedToOtherPlatoonDataResponse «from» int_frompl / UpdatePlatoonJoinngOtherPlatoonDataInformation
PlatoonAdditionCouplingCommand «from» int_frompl / TransmitPlatoonAdditionCoupledToPlatoonAndPlatoonHasJoinedPlatoon
at (platoon_joining_dur) / TransmitPlatoonJoiningByBeingAddedToOtherPlatoon
PlatoonAdditionToExistingPlatoonNotPossible «from» int_frompl [!interested_in_being_added_to_other_platoon]
when (platoon_addition_failure) [!interested_in_being_added_to_other_platoon] / TransmitPlatoonAdditionFailure
[has_been_added_to_other_platoon]
PlatoonCapableOfAddingPlatoon
[interested_in_adding_platoon]
See Figure 14
Platoon Capable of Adding A platoon To This Platoon
PlatoonAvailableForBeingAddedToOtherPlatoon «from» int_topl / TransmitPlatoonLeaderAvailableForPlatoonBeingAdded
PlatoonJoiningOtherPlatoonRequest «from» int_topl / AssessPlatoonAddtionRequestPossibility
when (platoon_addition_possible) / TransmitPlatoonAdditionToExistingPlatoonPossible
when (platoon_addition_not_possible) / TransmitPlatoonAdditionToExistingPlatoonNotPossible
PlatoonAdditionFailure «from» int_topl / ResetPlatoonAdditionAndAssessWhetherAdditionInterestStillExists
PlatoonJoiningByBeingAddedToOtherPlatoon «from» int_topl / AssessPlatoonAdditionStatus
when (!platoon_coupling_possible) / TransmitPlatoonJoiningByBeingAddedToOtherPlatoonDataResponse
when (platoon_coupling_possible) / TransmitPlatoonAdditionCouplingCommand
PlatoonAdditionCoupledToPlatoon «from» int_topl / NotePlatoonAdditionCoupledResponse
PlatoonHasJoinedPlatoon «from» int_topl / NotePlatoonAdditionJoinedAndAssessWhetherInterestForAdditionsStillExists
[!interested_in_adding_platoon]
WouldBePlatoonTruck
See Figure 9
start alone
when (truck_status==Alone)
at (tr_avail_dur) [!already_rejected] / TransmitTruckAvailableForPlatooning
Alone
PlatoonMembershipNotPossible «from» intplatoon_truck / SetAlreadyRejectedToTrueIfRequiredAndContinueAlone
when (emergency) / TransmitJoiningFailure
after (joining_dur) / TransmitPlatoonTruckJoiningPlatoon
PlatoonMemberhipPossible «from» intplatoon_truck / TransmitPlatoonTruckJoiningPlatoon
Joining
entry / truck_status=Joining
PlatoonJoiningPlatoonDataResponse «from» intplatoon_truck / UpdatePlatoonDataInformation
PlatoonTruckCouplingCommand «from» intplatoon_truck / SetTruckCoupledToTrueAndTransmitPlatoonTruckCoupledAndPlatoonTruckHasJoinedPlatoon
when (coupled) / truck_status=TailEnd
PlatoonTruck
See Figure 11 and 12
after (pos_dur) / TransmitPlatoonTruckPositionAndSpeed
[truck_in_platoon]
PlatoonTruck
PlatoonLeaderPostionAndSpeed «from» intplatoon_truck / AssessPlatoonLeaderPositionAndSpeedAndTakeAppropriateAction
PlatoonTruckPostionAndSpeed «from» intplatoon_truck / AssessPlatoonTruckPostionAndSpeedAndTakeApporpriateAction
RoadCondition «from» inttruck_rstretch / AssessImpactOfRoadConditionAndTakeAppropriateAction
PlatoonTrafficHandlingCommand «from» intplatoon_truck / BasedOnTrafficHandlingCommandAndPositionAndSpeedOfOtherTrucksTakeAppropriateAction
PlatoonTruckHasLeftPlatoon «from» intplatoon_truck / UpdateTruckStatusIfRequired
DissolvePlatoonCompletely «from» intplatoon_truck / SetPlatoonDissolvingToTrueAndAwaitStatusTailEnd
DissolvePlatoonFromPosition «from» intplatoon_truck / IfOwnPositionInPlatoonImpactedSetPlatoonDissolvingPartlyAndAwaitStatusTailEnd
PlatoonSplitRequestFromPlatoonLeader «from» intplatoon_truck / IfSplitRequestToThisTruckAssessAbilityAndRespondIfNotandImpactedPrepareForPossibleLeaderShift
PlatoonSplitRequestFromPlatoonTruck «from» intplatoon_truck / IfImpactedPrepareForLeaderShift
PlatoonSplitAccepted «from» intplatoon_truck / IfImpactedByPlatoonLeaderSplitRequestAcceptanceChangePlatoonLeader
when (new_leader_after_split) / TransmitPlatoonTruckHasLeftPlatoonToOldLeaderAndSetLeaderToNewLeader
PlatoonSplitAcknowledged «from» intplatoon_truck / IfImpactedByPlatoonTruckSplitSetNewPlatoonLeader
OtherVehicleIntent «from» inttruck_rstretch / TransmitOtherVehicleEntryIntoPlatoonRequestedOrAlreadyEnteredToPlatoonLeader
PlatoonSplitAcknowledged «from» intplatoon_truck / IfImpactedByPlatoonTruckRequestedSplitChangePlatoonLeader
PlatoonTruckLeavingPlatoonAgreed «from» intplatoon_truck / TransmitPlatoonTruckHasLeftPlatoonSetTruckStatusToAlone
TailEndOfPlatoon
when (truck_leaving_req || dissolving) / InitiateDecouplingAndTransmitPlatoonTruckDecoupled
when (truck_status==TailEnd)
when (truck_leaving_req) / TransmitPlatoonTruckLeavingRequestAndSetTruckStatusToLeaving
when (truck_status==TailEnd)
PlatoonTruckHasJoinedPlatoon «from» intplatoon_truck / SetTruckStatusToWithin
when (truck_status==Within)
PlatoonSplitAcknowledged «from» intplatoon_truck / IfImpactedByPlatoonTruckRequestedSplitChangePlatoonLeader
WithinPlatoon
PlatoonTruckHasJoinedPlatoon «from» intplatoon_truck / UpdateDataConcerningTrucksForPlatoon
when (truck_status==Alone)
LeavingPlatoon
PlatoonTruckLeavingPlatoonAcceptedWhenTailEnd «from» intplatoon_truck / AwaitTruckPositionChangeToTailEnd
PlatoonTruckLeavingPlatoonAgreed «from» intplatoon_truck / SetStatusToAlone
PlatoonTruckLeavingPlatoonDenied / ResetTruckStatusToWithin
when (platoon_truck_split_request) / TransmitPlatoonTruckSplitRequest
PlatoonSplitAcknowledged «from» intplatoon_truck / ChangeTruckStatusToLeader
PlatoonSplitRequested
entry / PrepareToAssumeLeadershipForSplitPlatoon

Figure 8. The complete truck state machine for platooning logic.

Truck joining a platoon

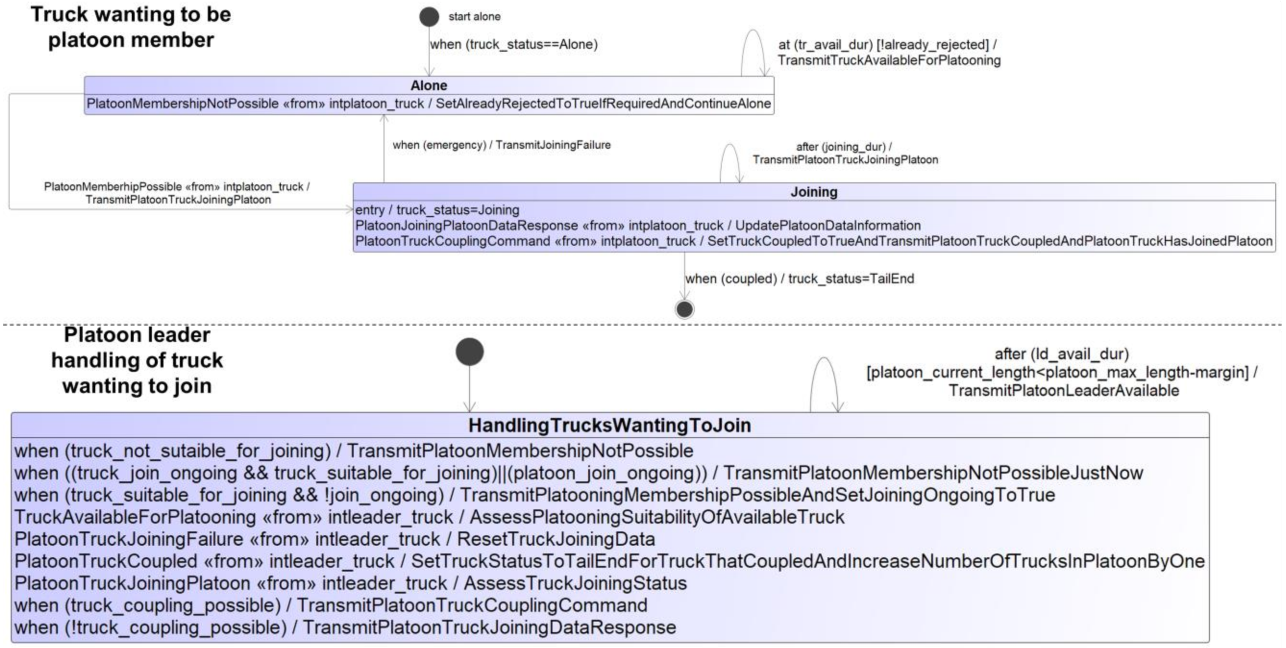

Truck wanting to be platoon member
start alone
when (truck_status==Alone)
at (tr_avail_dur) [!already_rejected] / TransmitTruckAvailableForPlatooning
Alone
PlatoonMembershipNotPossible «from» intplatoon_truck / SetAlreadyRejectedToTrueIfRequiredAndContinueAlone
when (emergency) / TransmitJoiningFailure
after (joining_dur) / TransmitPlatoonTruckJoiningPlatoon
PlatoonMemberhipPossible «from» intplatoon_truck / TransmitPlatoonTruckJoiningPlatoon
Joining
entry / truck_status=Joining
PlatoonJoiningPlatoonDataResponse «from» intplatoon_truck / UpdatePlatoonDataInformation
PlatoonTruckCouplingCommand «from» intplatoon_truck / SetTruckCoupledToTrueAndTransmitPlatoonTruckCoupledAndPlatoonTruckHasJoinedPlatoon
when (coupled) / truck_status=TailEnd
Platoon leader handling of truck wanting to join
after (ld_avail_dur) [platoon_current_length<platoon_max_length-margin] / TransmitPlatoonLeaderAvailable
HandlingTrucksWantingToJoin
when (truck_not_sutaible_for_joining) / TransmitPlatoonMembershipNotPossible
when ((truck_join_ongoing && truck_suitable_for_joining)||(platoon_join_ongoing)) / TransmitPlatoonMembershipNotPossibleJustNow
when (truck_suitable_for_joining && !join_ongoing) / TransmitPlatooningMembershipPossibleAndSetJoiningOngoingToTrue
TruckAvailableForPlatooning «from» intleader_truck / AssessPlatooningSuitabilityOfAvailableTruck
PlatoonTruckJoiningFailure «from» intleader_truck / ResetTruckJoiningData
PlatoonTruckCoupled «from» intleader_truck / SetTruckStatusToTailEndForTruckThatCoupledAndIncreaseNumberOfTrucksInPlatoonByOne
PlatoonTruckJoiningPlatoon «from» intleader_truck / AssessTruckJoiningStatus
when (truck_coupling_possible) / TransmitPlatoonTruckCouplingCommand
when (!truck_coupling_possible) / TransmitPlatoonTruckJoiningDataResponse

Figure 9. Truck Joining Platoon Handling.

It must be possible for a would-be platoon leader to assess a truck wanting to join to assess the distance that the platoon needs to maintain in between the truck joining and the one in front. This requires assessment of length, weight, engine power as well as braking distances. Several hazards can be associated with truck joining a platoon.

Table 1: Joining hazards

Hazard name	**Failure/operational mode**	**Situation**	**Hazard description**
Platoon truck assessment hazard	Incorrect data or assessment of parameters for truck wanting to join.	Truck wanting to join platoon	Platoon access allowed with unsafe safety distance to the truck in front.
Platoon length hazard	Truck joining leading to increase in platoon length	Platoon size increase	The length of the platoon is too long for safe control by platoon leader.

Table 2: Joining hazard safety goals

Hazard name	**Safety goal**
Platoon truck assessment hazard	Inspection handling is required to ensure that the trucks that want to join a platoon deliver correct information to the platoon leader.
Platoon truck assessment hazard	Inspection handling is required to ensure that a platoon leader assessment of truck joining suitability is correct.
Platoon length hazard	The assessment as to maximum length of platoon needs to be made external traffic flow conditions as well as road conditions into account.

The truck joining scenario can also be described in further detail as a sequence chart. There are several aspects to the hazards defined above. As an example, changing conditions due to traffic or road condition external to the platoon may imply that while originally safe, individual safety distances for trucks may no longer be correct and need to be revisited. The length of the platoon may also become unsafe due to traffic or road condition changes. It is highly likely that the application governing platooning will need to contain machine learning to ensure that condition changes are handled in a safe manner. The driver of the platoon leader will have to be aided by AI in order to handle the necessary decisions when allowing or disallowing a truck to join the platoon.

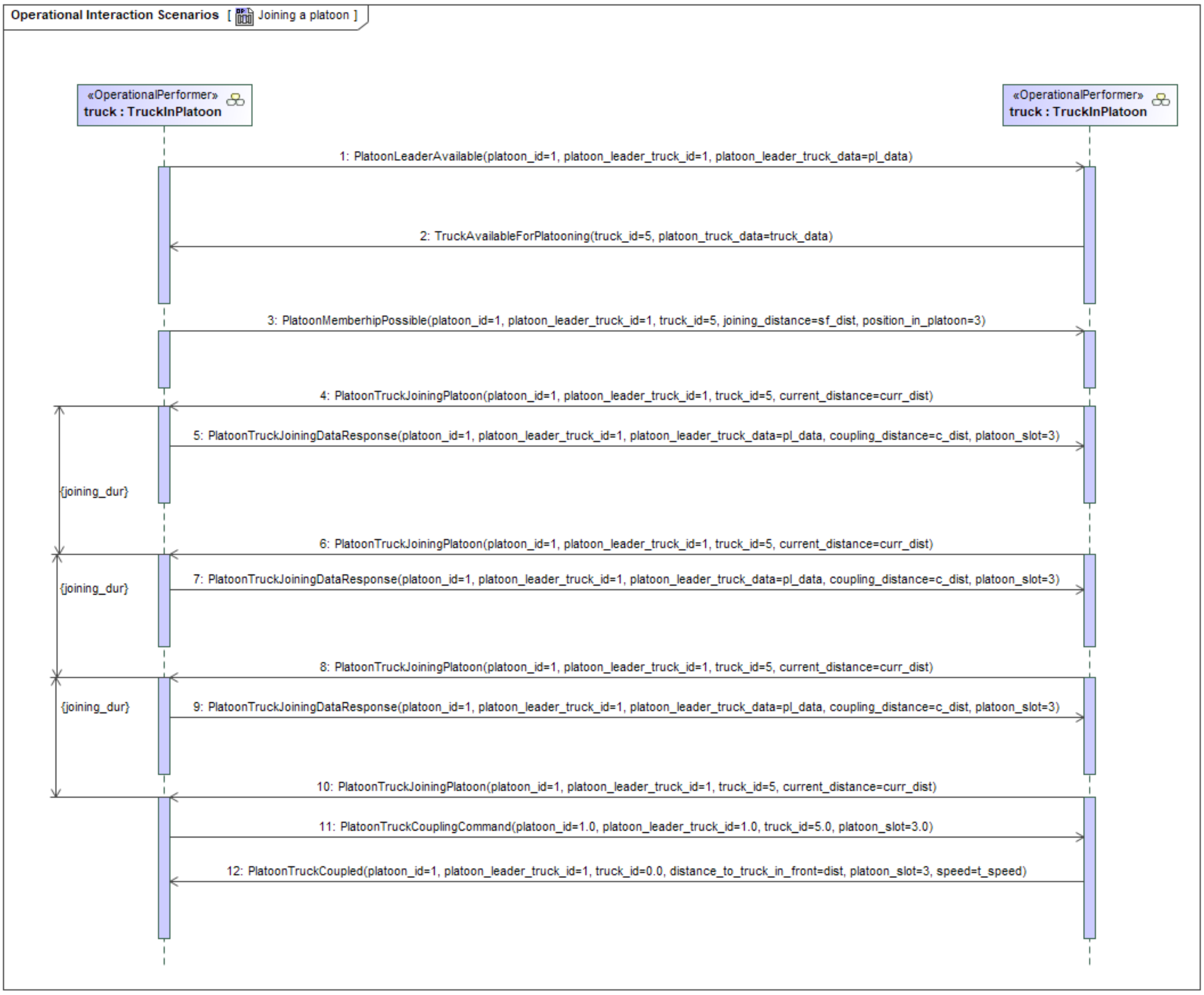

Operational Interaction Scenarios [Joining a platoon]
«OperationalPerformer»
truck : TruckInPlatoon
«OperationalPerformer»
truck : TruckInPlatoon
1: PlatoonLeaderAvailable(platoon_id=1, platoon_leader_truck_id=1, platoon_leader_truck_data=pl_data)
2: TruckAvailableForPlatooning(truck_id=5, platoon_truck_data=truck_data)
3: PlatoonMemberhipPossible(platoon_id=1, platoon_leader_truck_id=1, truck_id=5, joining_distance=sf_dist, position_in_platoon=3)
4: PlatoonTruckJoiningPlatoon(platoon_id=1, platoon_leader_truck_id=1, truck_id=5, current_distance=curr_dist)
5: PlatoonTruckJoiningDataResponse(platoon_id=1, platoon_leader_truck_id=1, platoon_leader_truck_data=pl_data, coupling_distance=c_dist, platoon_slot=3)
{joining_dur}
6: PlatoonTruckJoiningPlatoon(platoon_id=1, platoon_leader_truck_id=1, truck_id=5, current_distance=curr_dist)
7: PlatoonTruckJoiningDataResponse(platoon_id=1, platoon_leader_truck_id=1, platoon_leader_truck_data=pl_data, coupling_distance=c_dist, platoon_slot=3)
{joining_dur}
8: PlatoonTruckJoiningPlatoon(platoon_id=1, platoon_leader_truck_id=1, truck_id=5, current_distance=curr_dist)
9: PlatoonTruckJoiningDataResponse(platoon_id=1, platoon_leader_truck_id=1, platoon_leader_truck_data=pl_data, coupling_distance=c_dist, platoon_slot=3)
{joining_dur}
10: PlatoonTruckJoiningPlatoon(platoon_id=1, platoon_leader_truck_id=1, truck_id=5, current_distance=curr_dist)
11: PlatoonTruckCouplingCommand(platoon_id=1.0, platoon_leader_truck_id=1.0, truck_id=5.0, platoon_slot=3.0)
12: PlatoonTruckCoupled(platoon_id=1, platoon_leader_truck_id=1, truck_id=0.0, distance_to_truck_in_front=dist, platoon_slot=3, speed=t_speed)

Figure 10. Truck joining platoon as a sequence chart.

Platoon leader handling of a platoon truck

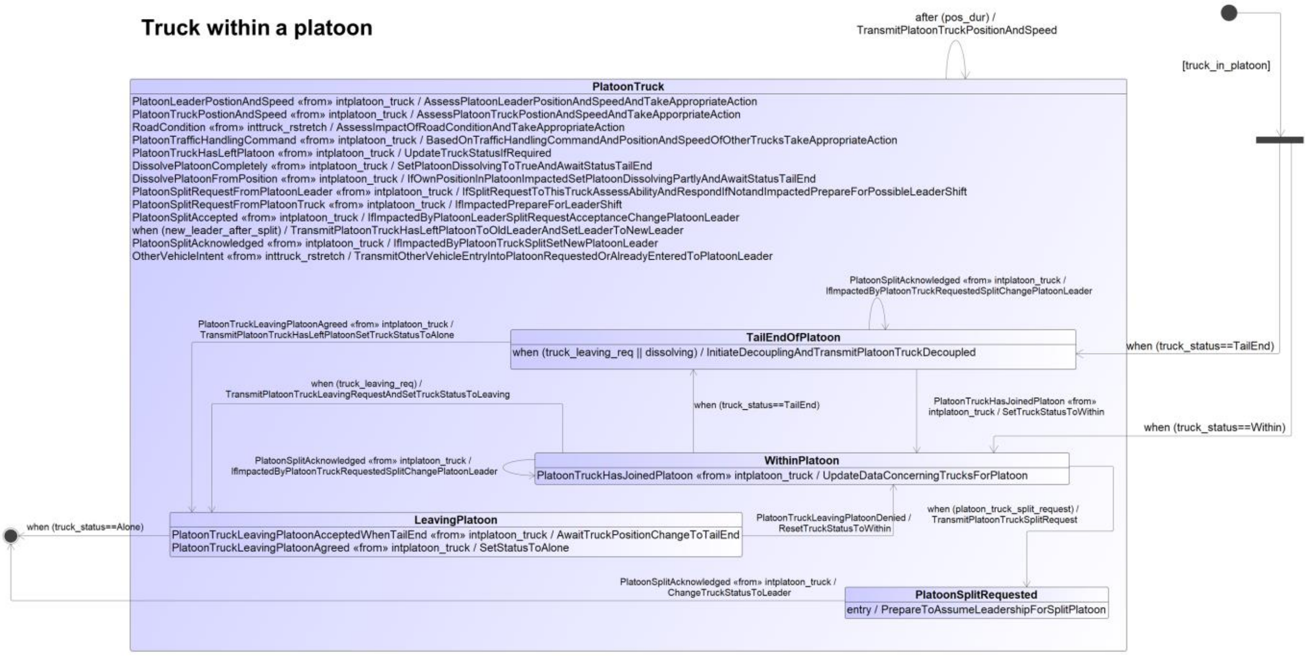

Truck within a platoon
after (pos_dur) /
TransmitPlatoonTruckPositionAndSpeed
[truck_in_platoon]
PlatoonTruck
PlatoonLeaderPostionAndSpeed «from» intplatoon_truck / AssessPlatoonLeaderPositionAndSpeedAndTakeAppropriateAction
PlatoonTruckPostionAndSpeed «from» intplatoon_truck / AssessPlatoonTruckPostionAndSpeedAndTakeApporpriateAction
RoadCondition «from» inttruck_rstretch / AssessImpactOfRoadConditionAndTakeAppropriateAction
PlatoonTrafficHandlingCommand «from» intplatoon_truck / BasedOnTrafficHandlingCommandAndPositionAndSpeedOfOtherTrucksTakeAppropriateAction
PlatoonTruckHasLeftPlatoon «from» intplatoon_truck / UpdateTruckStatusIfRequired
DissolvePlatoonCompletely «from» intplatoon_truck / SetPlatoonDissolvingToTrueAndAwaitStatusTailEnd
DissolvePlatoonFromPosition «from» intplatoon_truck / IfOwnPositionInPlatoonImpactedSetPlatoonDissolvingPartlyAndAwaitStatusTailEnd
PlatoonSplitRequestFromPlatoonLeader «from» intplatoon_truck / IfSplitRequestToThisTruckAssessAbilityAndRespondIfNotandImpactedPrepareForPossibleLeaderShift
PlatoonSplitRequestFromPlatoonTruck «from» intplatoon_truck / IfImpactedPrepareForLeaderShift
PlatoonSplitAccepted «from» intplatoon_truck / IfImpactedByPlatoonLeaderSplitRequestAcceptanceChangePlatoonLeader
when (new_leader_after_split) / TransmitPlatoonTruckHasLeftPlatoonToOldLeaderAndSetLeaderToNewLeader
PlatoonSplitAcknowledged «from» intplatoon_truck / IfImpactedByPlatoonTruckSplitSetNewPlatoonLeader
OtherVehicleIntent «from» inttruck_rstretch / TransmitOtherVehicleEntryIntoPlatoonRequestedOrAlreadyEnteredToPlatoonLeader
PlatoonSplitAcknowledged «from» intplatoon_truck /
IfImpactedByPlatoonTruckRequestedSplitChangePlatoonLeader
PlatoonTruckLeavingPlatoonAgreed «from» intplatoon_truck /
TransmitPlatoonTruckHasLeftPlatoonSetTruckStatusToAlone
TailEndOfPlatoon
when (truck_leaving_req || dissolving) / InitiateDecouplingAndTransmitPlatoonTruckDecoupled
when (truck_status==TailEnd)
when (truck_leaving_req) /
TransmitPlatoonTruckLeavingRequestAndSetTruckStatusToLeaving
when (truck_status==TailEnd)
PlatoonTruckHasJoinedPlatoon «from»
intplatoon_truck / SetTruckStatusToWithin
when (truck_status==Within)
PlatoonSplitAcknowledged «from» intplatoon_truck /
IfImpactedByPlatoonTruckRequestedSplitChangePlatoonLeader
WithinPlatoon
PlatoonTruckHasJoinedPlatoon «from» intplatoon_truck / UpdateDataConcerningTrucksForPlatoon
when (truck_status==Alone)
LeavingPlatoon
PlatoonTruckLeavingPlatoonAcceptedWhenTailEnd «from» intplatoon_truck / AwaitTruckPositionChangeToTailEnd
PlatoonTruckLeavingPlatoonAgreed «from» intplatoon_truck / SetStatusToAlone
PlatoonTruckLeavingPlatoonDenied /
ResetTruckStatusToWithin
when (platoon_truck_split_request) /
TransmitPlatoonTruckSplitRequest
PlatoonSplitAcknowledged «from» intplatoon_truck /
ChangeTruckStatusToLeader
PlatoonSplitRequested
entry / PrepareToAssumeLeadershipForSplitPlatoon

Figure 11. Platoon truck handling.

A truck joining a platoon will normally always join at the tail end. The only exception to this rule is if an entire platoon joins another platoon in which case the truck that changes from platoon leader to platoon truck will be given a slot number within the new platoon and appear as being within the platoon from start. In case a platoon truck wishes to leave the platoon, this needs to be notified to the platoon leader and the departure is also associated with hazards.

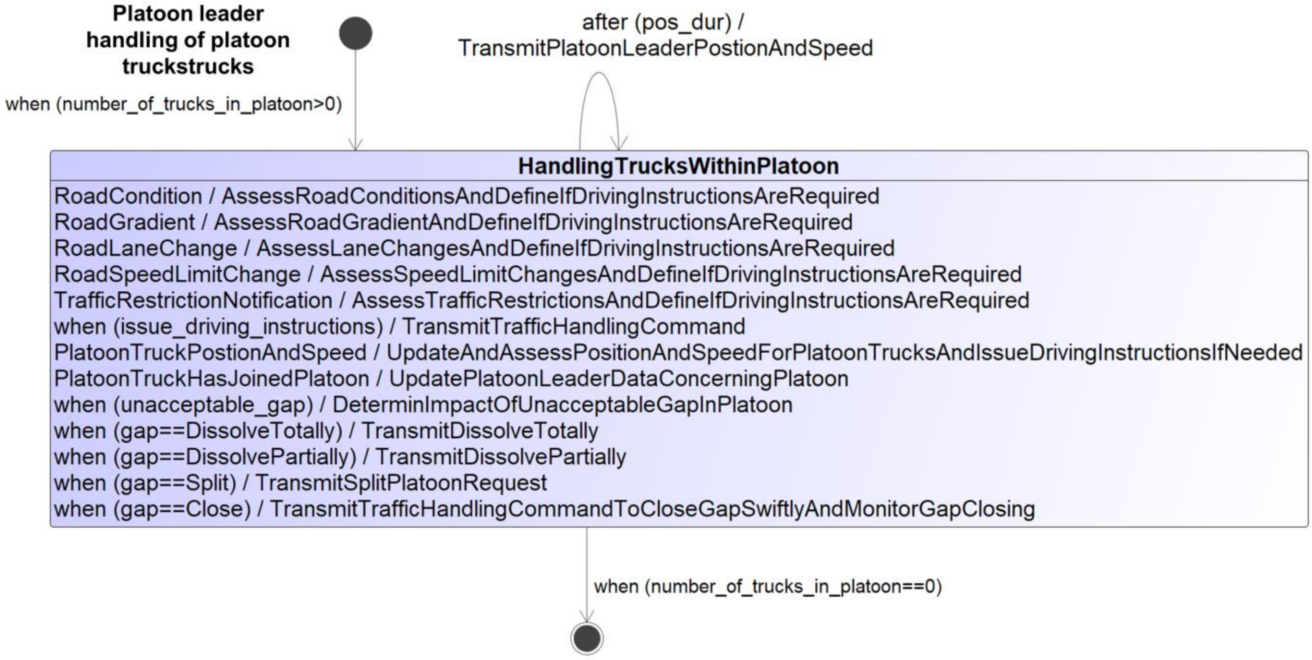

Platoon leader handling of platoon truckstrucks
when (number_of_trucks_in_platoon>0)
after (pos_dur) / TransmitPlatoonLeaderPostionAndSpeed
HandlingTrucksWithinPlatoon
RoadCondition / AssessRoadConditionsAndDefineIfDrivingInstructionsAreRequired
RoadGradient / AssessRoadGradientAndDefineIfDrivingInstructionsAreRequired
RoadLaneChange / AssessLaneChangesAndDefineIfDrivingInstructionsAreRequired
RoadSpeedLimitChange / AssessSpeedLimitChangesAndDefineIfDrivingInstructionsAreRequired
TrafficRestrictionNotification / AssessTrafficRestrictionsAndDefineIfDrivingInstructionsAreRequired
when (issue_driving_instructions) / TransmitTrafficHandlingCommand
PlatoonTruckPostionAndSpeed / UpdateAndAssessPositionAndSpeedForPlatoonTrucksAndIssueDrivingInstructionsIfNeeded
PlatoonTruckHasJoinedPlatoon / UpdatePlatoonLeaderDataConcerningPlatoon
when (unacceptable_gap) / DeterminImpactOfUnacceptableGapInPlatoon
when (gap==DissolveTotally) / TransmitDissolveTotally
when (gap==DissolvePartially) / TransmitDissolvePartially
when (gap==Split) / TransmitSplitPlatoonRequest
when (gap==Close) / TransmitTrafficHandlingCommandToCloseGapSwiftlyAndMonitorGapClosing
when (number_of_trucks_in_platoon==0)

Figure 12. Platoon leader handling of trucks within the platoon.

Table 3: Platoon handling hazards

Hazard name	**Failure/operational mode**	**Situation**	**Hazard description**
Road condition handling hazard	Road condition changes (lanes, speed restrictions, traffic lights, traffic flow, gradients)	Platoon moving normally	Changes in road conditions cannot be handled safely by members of the platoon.
Platoon gap hazard handling	Gaps appear inside of the platoon where individual truck members cannot follow the instructions originating from the platoon leader.	Platoon contains gaps that result for inability to manage road conditions	Uncontrolled changes in distance between trucks within a platoon leading to gaps that can be used by other non-platoon vehicles leading to uncontrollability.

Table 4: Platoon handling hazard safety goals goals

Hazard name	**Safety goal**
Road condition handling hazard	Platoon leader shall monitor status of member with a frequency that ensures that road condition changes can be dealt with.
Platoon gap handling hazard	If gaps appear within the platoon the platoon leader shall be able to act to either close the gap, dissolve the platoon, dissolve the platoon partially or split the platoon into two platoons, making the truck with the gap just in front of it the platoon leader for the trucks behind it.

Truck leaving and entry of other vehicles

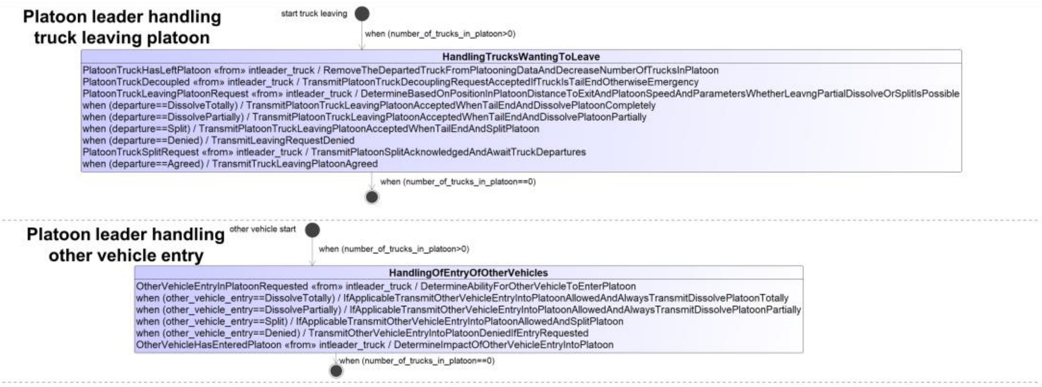

Platoon leader handling truck leaving platoon
start truck leaving
when (number_of_trucks_in_platoon>0)
HandlingTrucksWantingToLeave
PlatoonTruckHasLeftPlatoon «from» intleader_truck / RemoveTheDepartedTruckFromPlatooningDataAndDecreaseNumberOfTrucksInPlatoon
PlatoonTruckDecoupled «from» intleader_truck / TransmitPlatoonTruckDecouplingRequestAcceptedIfTruckIsTailEndOtherwiseEmergency
PlatoonTruckLeavingPlatoonRequest «from» intleader_truck / DetermineBasedOnPositionInPlatoonDistanceToExitAndPlatoonSpeedAndParametersWhetherLeavngPartialDissolveOrSplitIsPossible
when (departure==DissolveTotally) / TransmitPlatoonTruckLeavingPlatoonAcceptedWhenTailEndAndDissolvePlatoonCompletely
when (departure==DissolvePartially) / TransmitPlatoonTruckLeavingPlatoonAcceptedWhenTailEndAndDissolvePlatoonPartially
when (departure==Split) / TransmitPlatoonTruckLeavingPlatoonAcceptedWhenTailEndAndSplitPlatoon
when (departure==Denied) / TransmitLeavingRequestDenied
PlatoonTruckSplitRequest «from» intleader_truck / TransmitPlatoonSplitAcknowledgedAndAwaitTruckDepartures
when (departure==Agreed) / TransmitTruckLeavingPlatoonAgreed
when (number_of_trucks_in_platoon==0)
Platoon leader handling other vehicle entry
other vehicle start
when (number_of_trucks_in_platoon>0)
HandlingOfEntryOfOtherVehicles
OtherVehicleEntryInPlatoonRequested «from» intleader_truck / DetermineAbilityForOtherVehicleToEnterPlatoon
when (other_vehicle_entry==DissolveTotally) / IfApplicableTransmitOtherVehicleEntryIntoPlatoonAllowedAndAlwaysTransmitDissolvePlatoonTotally
when (other_vehicle_entry==DissolvePartially) / IfApplicableTransmitOtherVehicleEntryIntoPlatoonAllowedAndAlwaysTransmitDissolvePlatoonPartially
when (other_vehicle_entry==Split) / IfApplicableTransmitOtherVehicleEntryIntoPlatoonAllowedAndSplitPlatoon
when (other_vehicle_entry==Denied) / TransmitOtherVehicleEntryIntoPlatoonDeniedIfEntryRequested
OtherVehicleHasEnteredPlatoon «from» intleader_truck / DetermineImpactOfOtherVehicleEntryIntoPlatoon
when (number_of_trucks_in_platoon==0)

Figure 13. Departure of truck and entry of other vehicles.

Table 5: Truck departure and vehicle entry hazard

Hazard name	Failure/operational mode	Situation	Hazard description
Platoon truck departure hazard	Platoon truck leaves platoon	Platoon driving	Platoon truck departure initiated in an uncontrolled manner.
Platoon split request hazard	Platoon leader split request	Platoon should be divided into two platoons	Platoon split into two platoons attempted in an uncontrolled manner.
Other vehicle interested in entry into platoon.	A vehicle attempts to gain entry to platoon	Parts of the platoon has different end destination than other parts.	Other non-platoon vehicle attempting or succeeding in gaining entry into the platoon making the platoon uncontrollable.

Table 6: Departure and other vehicle entry hazard safety goals

Hazard name	Safety goal
Platoon truck departure hazard	The platoon leader shall have the ability to respond to a departure request either by agreeing (tail-end truck can easily leave) or by dissolving the platoon completely, partially or by splitting it based on the conditions at the time of the departure request.
Platoon split request hazard	The platoon leader shall be able to manage a split of the platoon either because of a request it generates or after having a platoon truck requesting a split.
Other vehicle interested in entry into platoon.	The platoon leader shall be able to manage a request entry or the fact that another vehicle has already succeeded in entering the platoon by either dissolving it totally or partially or by splitting it making the truck behind the other vehicle platoon leader for the new platoon.

Platoon joining other platoon

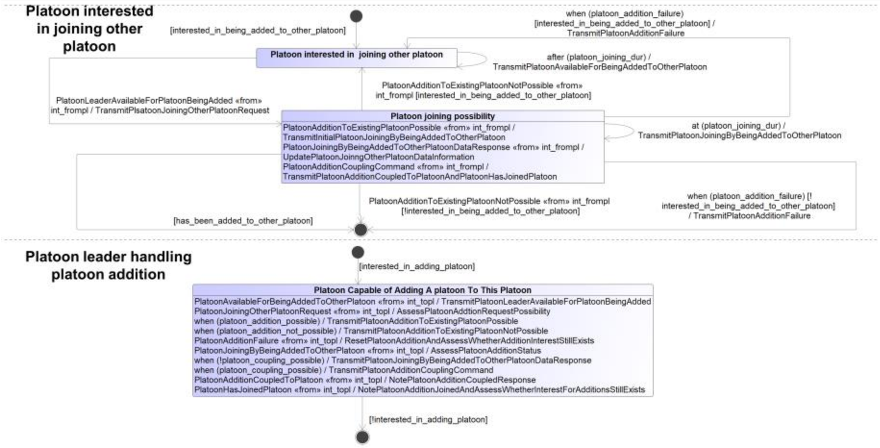

Platoon interested in joining other platoon
[interested_in_being_added_to_other_platoon]
when (platoon_addition_failure) [interested_in_being_added_to_other_platoon] / TransmitPlatoonAdditionFailure
Platoon interested in joining other platoon
after (platoon_joining_dur) / TransmitPlatoonAvailableForBeingAddedToOtherPlatoon
PlatoonAdditionToExistingPlatoonNotPossible «from» int_frompl [interested_in_being_added_to_other_platoon]
PlatoonLeaderAvailableForPlatoonBeingAdded «from» int_frompl / TransmitPlsatoonJoiningOtherPlatoonRequest
Platoon joining possibility
PlatoonAdditionToExistingPlatoonPossible «from» int_frompl / TransmitInitialPlatoonJoiningByBeingAddedToOtherPlatoon
PlatoonJoiningByBeingAddedToOtherPlatoonDataResponse «from» int_frompl / UpdatePlatoonJoinngOtherPlatoonDataInformation
PlatoonAdditionCouplingCommand «from» int_frompl / TransmitPlatoonAdditionCoupledToPlatoonAndPlatoonHasJoinedPlatoon
at (platoon_joining_dur) / TransmitPlatoonJoiningByBeingAddedToOtherPlatoon
PlatoonAdditionToExistingPlatoonNotPossible «from» int_frompl [!interested_in_being_added_to_other_platoon]
when (platoon_addition_failure) [! interested_in_being_added_to_other_platoon] / TransmitPlatoonAdditionFailure
[has_been_added_to_other_platoon]
Platoon leader handling platoon addition
[interested_in_adding_platoon]
Platoon Capable of Adding A platoon To This Platoon
PlatoonAvailableForBeingAddedToOtherPlatoon «from» int_topl / TransmitPlatoonLeaderAvailableForPlatoonBeingAdded
PlatoonJoiningOtherPlatoonRequest «from» int_topl / AssessPlatoonAddtionRequestPossibility
when (platoon_addition_possible) / TransmitPlatoonAdditionToExistingPlatoonPossible
when (platoon_addition_not_possible) / TransmitPlatoonAdditionToExistingPlatoonNotPossible
PlatoonAdditionFailure «from» int_topl / ResetPlatoonAdditionAndAssessWhetherAdditionInterestStillExists
PlatoonJoiningByBeingAddedToOtherPlatoon «from» int_topl / AssessPlatoonAdditionStatus
when (!platoon_coupling_possible) / TransmitPlatoonJoiningByBeingAddedToOtherPlatoonDataResponse
when (platoon_coupling_possible) / TransmitPlatoonAdditionCouplingCommand
PlatoonAdditionCoupledToPlatoon «from» int_topl / NotePlatoonAdditionCoupledResponse
PlatoonHasJoinedPlatoon «from» int_topl / NotePlatoonAdditionJoinedAndAssessWhetherInterestForAdditionsStillExists
[!interested_in_adding_platoon]

Figure 14. Platoon joining another platoon.

Table 7: Platoon joining hazards

Hazard name	Failure/operational mode	Situation	Hazard description
Platoon joining platoon assessment hazard	Incorrect data or assessment of parameters for platoon wanting to join.	Platoon wanting to be added to another platoon	Platoon addition allowed with unsafe characteristics.
Platoon joining platoon length hazard	Platoon joining leading to increase in platoon length	Platoon size increase	The length of the platoon is too long for safe control by platoon leader.

Table 2: Joining hazard safety goals

Hazard name	Safety goal
Platoon joining platoon assessment hazard	Allowing an existing platoon to join another platoon requires assessment of the parameters of all trucks within the platoon wanting to join based on a standardized approach.
Platoon joining platoon length hazard	Adding an entire platoon to an existing platoon shall only be possible if the total length falls within the length safety margin given the external conditions.

The entire table described in Figure 3 can be placed within the model and maintained there. This means that the resulting table of hazards and its analysis can be maintained within the model used to describe the system of system hazards. This also ensures that data in the model and the table are kept in synch since data items will not have to be duplicated to appear in the model and in the tables making the handling more efficient as well as less prone to errors.

From the perspective of the platooning example and its hazards one conclusion that can be drawn is that the handling will require a significant amount of machine learning to make a platoon adapt to changing external circumstances. The use of the Unified Architecture Framework (UAF) can be used further to describe the example. The service domain of UAF could be used to define services that delivers detailed road condition and traffic information to the platoon leader for the leader (both human driver and the platoon leader application) to determine suitable platoon actions in advance of line of sight. The communication in between trucks could also benefit from the security domain within UAF to determine and manage cyber as well as human threats. Any driver that can act as a platoon leader also needs competence and training and this can be defined using the personnel views within UAF.

Conclusions

In this paper, we have modelled a platooning system using the Unified Architecture Framework, UAF, to analyze how hazard analysis can be performed for such a system-of-systems. We have elaborated on some of the steps and how those could be made effectively and how hurdles of complexity can be avoided. Analyzing hazards for systems of systems presents a large hazard space to analyze, and to manage such a laborious task it needs to be performed with the system of system perspective in focus so as not to be bogged down by constituent system details. The system hazard analysis also needs to be performed on a system-by-system basis and any results pertinent to a system of system hazard analysis fed into the conditions that the system of systems must deal with. Based on the work performed, the use of a logical model to characterize the needed behavior of the system of systems is very useful in determining the hazards that a given system of system will need to deal with. The use of a state machine is a very compact way to do this and within it details a very large amount of possible sequence charts that would look at one hazard at the time. If only the sequence charts are used, then the chance of maintaining overall consistency would be much less making the hazard analysis less secure.

UAF is eminently suitable for performing an analysis of this kind since it already contains the domains and the relationships that are of interest in performing such an analysis. It is a framework standardized by the Object Management group and is maintained by the group to keep it up to date. It is also implemented by several different tool manufacturers. Having elaborated on modelling a system-of-systems with the intent of aiding in hazard analysis, one important conclusion is that modelling seems to be an necessary key to manage a very large combinatory space of situations, fault modes, and system states. Reusing hazard analysis models from constituent system analysis enables an abstraction of details and helps to focus the modelling effort. Based on the decisions that the driver of the platoon leader truck will have to make for trucks joining or exiting a platoon it seems clear that it has to be supported by artificial intelligence (AI) applications to perform all of the analysis needed in order to present the lead driver with clear cut decisions that will not impact on the driver driving the truck he/ she is driving.

References

Axelsson J., "Safety in Vehicle Platooning: A Systematic Literature Review," in IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 5, pp. 1033-1045, May 2017, doi: 10.1109/TITS.2016.2598873.

Baumgart, S., Fröberg, J., & Punnekkat, S. (2017). Analyzing Hazards in System-of-Systems : Described in a Quarry Site Automation Context. 11th Annual IEEE International Systems Conference SysCon, 544–551. https://doi.org/10.1109/SYSCON.2017.7934783

DoD CIO, 2012, DoD Architecture Framework Version 2.02, DoD Deputy Chief Information Officer, Available online at http://dodcio.defense.gov/dodaf20/dodaf20_pes.aspx, accessed June, 2014.

ISO 26262, International Standard, "Road Vehicles - Functional Safety", Second edition 2018-12

INCOSE, 2022, INCOSE Guide to the Systems Engineering Body of Knowledge (SEBoK).

M. W. Maier, “Architecting Principles for Systems-of-Systems,” INCOSE Int. Symp., pp. 565–573, Jul. 1996.

MOD Architectural Framework, Version 1.2, 2020, Office of Public Sector Information, https://www.gov.uk/guidance/mod-architecture-framework/

NATO Architecture Framework Version 4, January 2018, Architecture Capability Team Consultation, Command & Control Board

OMG 2013. OMG2013-08-04:2013. Unified Profile for DoDAF/MODAF (UPDM) V2.1, http://www.omg.org/spec/UPDM/2.1/PDF

OMG, 2017, OMG2012-06-01.OMG Systems Modeling Language (OMG SysML™), V1.7, http://www.omg.org/spec/SysML/1.7/PDF/.

OMG, 2022, The Unified Architecture Framework, (UAF) Available from https ://www.omg.org/spec/UAF

Powel Douglass B., Real-Time UML Workshop for Embedded Systems (Second Edition), 2014

Svenson P. and Axelsson J., "Should I Stay or Should I Go? How Constituent Systems Decide to Join or Leave Constellations in Collaborative SoS," 2021 16th International Conference of System of Systems Engineering (SoSE), 2021, pp. 179-184, doi: 10.1109/ SOSE 52739.2021.9497474.

Stevens, Richard, et al, 1998, Systems Engineer Coping with Complexity, published by Prentice Hall

TECOSA 2021, MBRASA: Model-Based Risk Assessment and Safety Analysis – exploring new methods for ensuring safe autonomous transport system for the future!, online available at https://www.tecosa.center.kth.se/2021/09/16/mbrasa-model-based-risk-assessment-and-safety-analysis-exploring-new-methods-for-ensuring-safe-autonomous-transport-system-for-the-future/

Biography

Lars-Olof Kihlström. Lars-Olof Kihlström is a principal consultant at CAG Syntell where he has worked since 2013, primarily in the area of MBSE. He has been a core member of the UAF group within the OMG since its start as the UPDM group. He was involved in the development of NAF as well as MODAF. He has worked with modelling in a variety of domains since the middle of the 1980:ies such as telecommunications, automotive, defence as well as financial systems. He is specifically interested in models that can be used to analyze the behavior of system of systems.

Matthew Hause. Matthew Hause is a principal consultant at SSI, a member of the UAF group, and a member of the OMG SysML specification team. He has been developing multi-national complex systems for almost 40 years as a systems and software engineer. He worked in the power systems industry, command and control systems, process control, SCADA, military systems, and many other areas. His role at SSI includes consulting, mentoring, standards development, specification of the UAF profile and training.

Joakim Fröberg. Joakim Fröberg is a Safety assessor and researcher at Safety Integrity AB. Joakim holds a PhD in Computer Science since 2007 and has 20+ years of industry experience in developing software-intensive and safety-critical embedded systems. Joakim's research interests include safety analysis and safety engineering, systems engineering, and evaluation of system architecture. Joakim has participated in many research and development projects including hybrid-electric automotive drive systems, autonomous and remote-controlled construction equipment and heavy vehicles, and vehicle platooning systems. Joakim has worked with safety analysis and architecture analysis in many application areas including construction, automotive, transport, agriculture, mining.