applied sciences

MDPI

Article

An Optimization Framework for Information Management in Adaptive Automotive Human–Machine Interfaces

Francesco Tufano [1], Sushant Waman Bahadure [2], Manuela Tufo [2,3], Luigi Novella [2,3], Giovanni Fiengo [2,3] and Stefania Santini [4,*]

1 Department of Industrial Engineering, University of Naples Federico II, Via Claudio, 21, 80125 Naples, Italy; francesco.tufano@unina.it
2 Kineton R&D, Kineton S.r.l., 80146 Naples, Italy; sushant.bahadure@kineton.it (S.W.B.); manuela.tufo@kineton.it (M.T.); luigi.novella@kineton.it (L.N.); giovanni.fiengo@kineton.it (G.F.)
3 Department of Engineering, University of Sannio, 82100 Benevento, Italy
4 Department of Electrical Engineering and Information Technology, University of Naples Federico II, Via Claudio, 21, 80125 Naples, Italy
* Correspondence: stefania.santini@unina.it

Abstract: In recent years, advancements in Intelligent and Connected Vehicles (ICVs) have led to a significant increase in the amount of information to the driver through Human–Machine Interfaces (HMIs). To prevent driver cognitive overload, the development of Adaptive HMIs (A-HMIs) has emerged. Indeed, A-HMIs regulate information flows by dynamically adapting the presentation to suit the contextual driving conditions. This paper presents a novel methodology, based on multi-objective optimization, that offers a more generalized design approach for adaptive strategies in A-HMIs. The proposed methodology is specifically tailored for designing an A-HMI that, by continuously monitoring the Driver–Vehicle–Environment (DVE) system, schedules actions requested by applications and selects appropriate presentation modalities to suit the current state of the DVE. The problem to derive these adaptive strategies is formulated as an optimization task where the objective is to find a set of rules to manage information flow between vehicle and driver that minimizes both the driver's workload and the queuing of actions. To achieve these goals, the methodology evaluates through two indexes how applications' requests impact the driver's cognitive load and the waiting queue for actions. The optimization procedure has been solved offline to define adaptive strategies for scheduling five application requests, i.e., forward collision warning, system interaction, turn indicators, infotainment volume increase, and phone calls. A theoretical analysis has demonstrated the effectiveness of the proposed framework in optimizing the prioritization strategy for actions requested by applications. By adopting this approach, the design of rules for the scheduling process of the A-HMI architecture is significantly streamlined while gaining adaptive capabilities to prevent driver cognitive overload.

Keywords: adaptive human–machine interface; intelligent and connected vehicle; multi-objective optimization

check for updates

Citation: Tufano, F.; Bahadure, S.W.; Tufo, M.; Novella, L.; Fiengo, G.; Santini, S. An Optimization Framework for Information Management in Adaptive Automotive Human–Machine Interfaces. *Appl. Sci.* **2023**, *13*, 10687. https://doi.org/10.3390/app131910687

Academic Editor: João M. F. Rodrigues

Received: 31 July 2023
Revised: 14 September 2023
Accepted: 19 September 2023
Published: 26 September 2023

1. Introduction

The future Intelligent Transportation Systems (ITSs) hold the potential to revolutionize mobility, offering an integrated and sustainable solution to modern transportation challenges [1]. They are envisioned as dynamic eco-systems consisting of various vehicles and distributed services, controlled by computing devices. These sophisticated frameworks promote efficient information exchange and collaboration among vehicles, road infrastructure, individuals, and cloud-based platforms [2]. At the core of these visionary transportation systems lies the concept of Intelligent and Connected Vehicles (ICVs). Empowered with advanced onboard sensors, controllers, and actuators, ICVs have garnered significant attention and recognition in both research and industrial domains [3]. Indeed, ICVs play a pivotal role in ensuring safe, comfortable, and energy-efficient transportation [4] through the utilization of cutting-edge technologies. These technologies encompass complex environmental awareness [5], intelligent decision-making, and collaborative control mechanisms [6]. The ICVs hinge on the convergence of two fundamental technologies: Autonomous Driving (AD) and vehicular networking. The first has laid the groundwork for the new generation of Autonomous Vehicles (AVs). These AVs possess the ability to autonomously perceive their surroundings and to realize automatic driving, all solely by onboard sensors and control systems [7]. In contrast, Connected Vehicles (CVs) leverage network systems to augment their environmental perception, enabling advanced decision-making control [8]. Through a strategic combination of vehicle-to-X communication and in-vehicle sensors, ICVs gain an empowered perception of their surroundings, beyond the constraints of traditional onboard sensors [9]. This synergistic combination of AV and CV capabilities has swiftly emerged as the dominant developmental paradigm for ICVs [10]. Contextually, the networking and intellectualization in vehicles have given rise to a new dimension of interaction between humans and cars [11]. This paradigm shift creates promising opportunities for the innovative development of Human–Machine Interfaces (HMIs). HMI technology plays a key role as the medium of interaction between drivers and ICVs. Across the industry, and the research community, there is a collective commitment to advancing integrated interfaces capable of seamlessly receiving, storing, processing, and presenting information effectively [12]. The trajectory of HMI interfaces demonstrates a discernible trend characterized by cross-modal fusion and the progressive expansion of the interaction area. This trend emphasizes the harmonious integration of diverse sensory inputs and the broadening of the scope of interactions between users and ICVs. As ongoing advancements in HMI technology unfold, the ultimate goal is to create intuitive, user-friendly interfaces that elevate the overall driving experience, promoting heightened safety, efficiency, and connectivity [13]. The achievement of this multi-dimensional human–vehicle interaction has been made possible by the recent advancements in Adaptive HMI (A-HMI) [14].

1.1. Overview of A-HMIs in ICVs

A-HMIs are designed with a primary purpose in mind: to provide a natural and intuitive means of facilitating complex digital operations within vehicles. Adaptivity refers to the system's ability to dynamically respond in diverse ways, tailored to individual situations and users, achieved by continuously tracking and sensing information about their users, current tasks, and the environment [15]. For this purpose, A-HMI encompasses a range of input and output functions. On the input side, a diverse array of channels, such as microphones, cameras, and touch screens, are seamlessly integrated to collect and interpret human behavioral characteristics (eye movement, pose, voice, gesture, gaze, etc.) and physiological signals (electroencephalogram, heart rate, etc.) [16]. Through robust data analysis and fusion with information obtained via AD and vehicular networking technologies, these multi-modal interactive inputs are harnessed to enhance the user experience [17]. By leveraging comprehensive knowledge about the current driving situation, both context-aware information temporal management and modality manipulation strategies come into play [18]. These encompass message prioritization and modality selection, intelligently reducing the flow of information while ensuring an optimal and appropriate presentation. Indeed, interactive output is also diversified to establish feedback through various sensory modalities. These include tactile sensations, auditory cues, olfactory responses, gravity balance, temperature changes, and more [19]. Such diverse feedback mechanisms enable ICVs to deliver a comprehensive and engaging interaction experience to users. By capitalizing on the complementary advantages and disadvantages of different interfaces, A-HMI addresses the diverse interaction demands in ICVs effectively. For instance, auditory and tactile interfaces empower drivers to perform interactive tasks without being overly dependent on visual cues [20].

1.2. A-HMIs as a Solution to Reduce Driver Distraction

One of the key accomplishments of A-HMI lies in its significant contribution to reducing the workload and distractions faced by drivers by effectively addressing the challenges posed by the abundant and diverse information generated by In-Vehicle Information Systems (IVIS), Advanced Driver Assistance Systems (ADAS) [21], and nomadic devices [22]. Indeed, except for fully AVs, drivers are still expected to uphold attention towards the driving environment and be prepared to assume control of the driving task if the situation demands [23]. However, too much information causes cognitive overload, resulting in drivers disengaging from the driving task [24]. A-HMI, by carefully managing the information flows to the driver, are capable of mitigating issues of cognitive overload and distraction [13]. More specifically, A-HMIs exhibit the ability to adjust amount and format of information conveyed to the driver, contingent upon the driver's cognitive focus level or the cognitive demands of the surrounding environment [25]. Examples of this functionality encompass actions such as suspending phone calls during challenging driving scenarios or enhancing the prominence of alerts when the driver's distraction level is assessed to be high [26].

1.2.1. Related Work

Drawing from a literature review, eight strategies have been commonly employed to exploit adaptive functions in A-HMIs [26]. These strategies can be categorized into two groups, each addressing different aspects of attention management. The first focuses on optimizing attentional demands to prevent overload or underload. Within this category, there are five strategies, i.e., Limiting, Simplifying, Filtering, Delaying, and Activating. The first strategy involves limiting access to functions of non-driving tasks to prevent distractions, particularly during high-demand situations [13]. In the Simplifying approach, displays and alerts are simplified to reduce workload. This simplification may entail, for example, increasing text size or combining alerts into a single summary alert [27]. In contrast, the Filtering strategy is designed to allow only critical alerts and notifications to reach the driver during demanding scenarios [28]. The delaying strategy involves postponing alerts, notifications, etc. when drivers are experiencing high workload conditions [29]. On the other hand, the last strategy is employed in situations characterized by underload, typically by initiating a non-driving task to increase driver workload [30]. The remaining three strategies, i.e., Advancing, Supplementing, and Augmenting, fall under the second category that aims to redirect the driver's focus back to the driving task. The Advancing approach involves the adjustment of alert timing and issuing alerts prior to their standard timing, shifting the driver's attention back to the driving task [31]. In contrast, the Supplementing strategy serves to complement existing warnings by supplying extra warnings when the driver's attention is compromised [32]. Finally, within the Augmenting strategy, alerts are modified, for example, utilizing different sensory modalities, to increase awareness [33].

A-HMIs generally achieve their adaptability by employing one or a combination of these strategies. For instance, in [34], a sophisticated workload management system has been implemented to suppress non-essential messages during challenging driving scenarios, allowing the driver to concentrate on resolving the situation without unnecessary distractions. An additional innovative approach for information management has been developed in [35], focusing on the scheduling of driver warning messages. This method presents a compelling solution by formulating the message scheduling as a resource-constrained scheduling problem. This problem encompasses a dynamic set of presentation requests, contending for limited resources, and incorporates modifying actions as conflict resolution strategies. These strategies include postponing, preponing, shortening, canceling tasks, and switching resources. To effectively address the scheduling problem, a transformation into a tree search problem is executed. The work presented in [36], following the blackboard design pattern, introduces an A-HMI architecture that offers an integration of multiple strategies to manipulate and adapt in-vehicle information flow, depending on the driving context. An innovative fusion strategy for messages is a key highlight of this architecture, built on a taxonomic message model. This sophisticated fusion strategy facilitates the combination of two or more low-level messages, resulting in the creation of a higher-level message. Noteworthy research on information management can be found in [37]. It presents the Interaction and Communication Assistant (ICA) system, which is a highly advanced approach to efficiently manage information through continuous monitoring of the driver, vehicle, and environment. In the ICA system, access to Input/Output (I/O) devices is granted through a request-response mechanism. Applications interact with the ICA by submitting access requests, and in response, the ICA provides an assigned channel through a suitable reply. To optimize system efficiency, the ICA prioritizes and schedules requests, employing concatenated rules-based decision modules to select appropriate modalities and channels.

In addition to adaptation strategies, A-HMIs can also be categorized into several Levels of Adaptive Sensitive Responses (LASR) [22]. The LASR framework outlines 5 levels of adaptivity, from a level 0, where no adaptivity is present, to a level 4, representing systems that can make connections between system behavior, context, user reactions, and even infer mood, emotional state, or personality [38]. Between these, in LASR 1, responses are managed by saved adaptations based on user selections. Within LASR 2, the system reacts according to predefined rules [39]. Finally, LASR 3 refers to systems that adapt based on real-time rules learned in the vehicle. With respect to these categorizations, a recent study demonstrated that a significant proportion of users may prefer A-HMIs falling under LASR 2 [22]. This makes A-HMIs driven by predefined rules worthy of research.

1.2.2. Research Gaps

Despite the fact that A-HMIs that react according to predefined rules have been demonstrated as particularly suited for guiding and interpreting adaptation in the context of automotive A-HMIs [40], it is essential to acknowledge the well-known issue associated with these adaptive functions. Indeed, the development of this kind of A-HMI requires comprehensive strategies to cope with all diverse tasks involved in the information management process [41]. These strategies are essential to establish resilient adaptive functions that significantly enhance driver–vehicle interactions. Notably, the brittleness of A-HMIs becomes evident when confronted with missing or unexpected input values, potentially affecting the system's adaptability and robustness [42]. This issue is made frequent by the vast number of potential driving conditions and the extensive information that needs to be managed within the context of new ICVs [16]. To tackle these limitations effectively, it is essential to move beyond the conventional empirical approach to A-HMI development, which heavily relies on the competences of expert pools [43]. Instead, it is essential to adopt novel frameworks for optimizing adaptation functions in A-HMIs, promoting the establishment of a more generalized and adaptable design approach. By embracing these innovative optimization approaches [44,45], A-HMI technology could achieve heightened effectiveness and performance, ultimately elevating the overall driving experience and enhancing safety for all users, as investigated in other application fields [46].

1.3. Objectives and Scope of the Study

Based on the aforementioned facts, this work introduces a novel methodology aimed at addressing the optimization problem inherent in designing rules for A-HMIs based on adaptive predefined strategies. Specifically, the methodology proposes a formal and generalized approach, leveraging the multi-objective optimization to define strategies for human–machine adaptation. To enhance clarity of the study, this methodology is applied to tackle and solve the problem of estimating adaptation rules to be used in a Filtering strategy. A theoretical analysis has demonstrated the effectiveness of the proposed framework. Indeed, by embracing this novel approach, the design of adaptive strategies in A-HMIs can be automatized. This automated approach expedites the development of highly efficient and effective A-HMI systems, streamlining the development process.

Finally, the article is organized as follows. Section 2 focuses on the description of an A-HMI architecture whose optimization is addressed with the proposed methodology.

Section 3 presents the problem statement. The proposed multi-objective optimization methodology to define adaptive strategies for A-HMI is suggested in Section 4. Section 5 discloses the implementation of the introduced approach in a case study and analyzes results obtained. Conclusions are drawn in Section 6.

2. System Architecture

This study focuses on an A-HMI whose architecture concept is depicted in Figure 1. It is functionally characterized by the following key features:

- Multi-modal HMIs, i.e., I/O devices shared by different ADAS and IVIS, such as Liquid Crystal Displays (LCD), Head-Up Displays (HUD), speech I/O, haptic I/O.
- An Information Management System (IMS), the control authority of the A-HMI, i.e., a centralized intelligence that performs information prioritization and scheduling.
- A gateway to connect nomadic devices to the in-vehicle system for sharing data, applications, and I/O devices. Thus, the functionality of the nomadic devices can be used by the in-vehicle system and vice versa.
- A system for real-time monitoring of driver and driving situation to assess information about Driver, Vehicle, and Environment (DVE status recognition and monitoring).

Scenarios that the A-HMI should address are defined using two parameters [47]: (i) actions; (ii) Driver–Vehicle–Environment (DVE) conditions. The actions represent an event by an application, i.e., component offering a specific functionality to the user (such as navigation, phone, lane departure warning, forward collision warning, etc.), classified into priority classes on the basis of their importance for the driver. In contrast, the DVE conditions are discrete values representing the current state of driver, vehicle, and environment perceived by a DVE state recognition and monitoring system, which give a description regarding the driver's ability and availability to drive the vehicle. It is precisely the perception of the current driving scenario and its impact on the driver which enable the adaptive functions of A-HMI according to the driving conditions. Indeed, the DVE state, together with the applications and their priority classes as well, are considered in the problem-solving process of the IMS control framework to assign priorities to each output request from the applications, to schedule their presentation, and to select modality, channel, and layout for the outputs. The DVE state recognition and monitoring system and the IMS, which are the composing modules of the A-HMI, are detailed in the following.

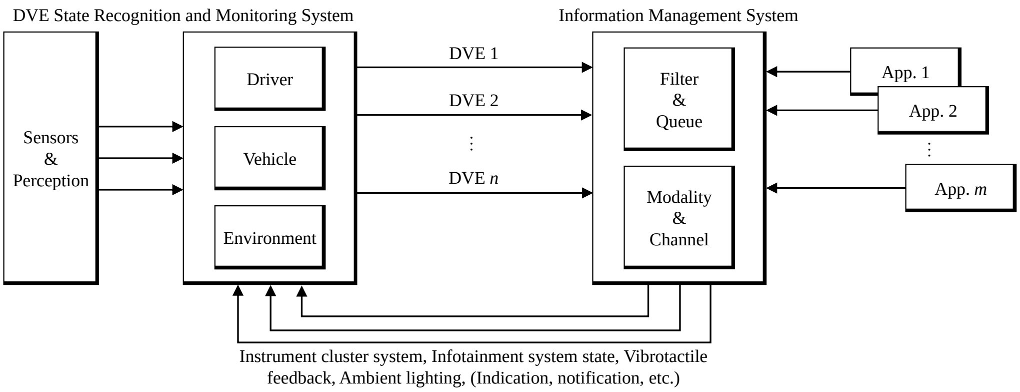

Figure 1. A-HMI system architecture concept.

2.1. DVE State Recognition and Monitoring System

Introduction of DVE status recognition and monitoring technologies into automotive HMIs helps to improve the safety and reliability of vehicles by recognizing the status of driver, vehicle, and surrounding environment. In particular, recent rapid advances in intelligent computing platforms have promoted the development of complex multi-sensor computing systems for active perception of driver's status [48], including cognitive load [49], secondary activities [50], emotion [51], positions [52], fatigue [53,54], etc.

To totally assess danger of driving conditions, these systems usually estimate the risks related to traffic and environmental factors as well [55,56]. An example of this advanced approach for DVE monitoring has been proposed by the AIDE (Adaptive Integrated Driver–vehicle interfacE) project (2004–2008) [37]. Here, a set of five modules, each of them designed to address a different dimension of the DVE state, i.e., Traffic and Environment Risk Assessment module, Driver Characteristic module, Driver Availability Estimator module, Driver State Degradation module, and Cockpit Activity Assessment module, derive a complete set of information (summarized in many parameters) about driver, vehicle, and environment conditions.

In this study, the A-HMI includes a non-invasive and low-cost DVE state recognition and monitoring system that, by common in-vehicle sensors, computes in real-time a set of five DVE parameters needed to enable adaptive management functions of the A-HMI. These give an abstract discrete representation of the environment, drivers, and vehicle state perceived, which is a key driving factor to estimate drivers' ability to process actions requested by applications, which could be a potential source of distraction. The five DVE parameters are defined as follows.

- DVE 1 $\in \mathbb{B} = \{0, 1\}$: driver's eyes off the road (0—No, driver is currently looking at the road ahead; 1—Yes, driver is currently looking at something other than the road ahead). Calculated by using driver head and eye movements and vehicle speed as input data, it gives a discrete representation of driver cognitive load or shift of visual attention away from the road ahead, induced by an external event or a secondary task.
- DVE 2 $\in \mathbb{B} = \{0, 1\}$: driver impairment (0—Normal, alert and few driving errors; 1—Dangerous, drowsy with some driving errors, critical driving time, and long trip duration). Driver state of drowsiness is calculated based on lane positioning, driving time, and PERCLOS [57–59] (the percentage of eyelid closure over the pupil over time) detected by an in-cabin camera with a face-mesh technique. It describes the physical ability of the driver to drive (fatigue, sleepiness, etc.).
- DVE 3 $\in \mathbb{B} = \{0, 1\}$: traffic risk (0—Low, no risk of collision with any other vehicle in path; 1—High, there is a risk of collision with other vehicles in path). This parameter represents the level of risk concerning the traffic density. It is driven by traffic risk established by Forward Collision Warning (FCW). Specifically, FCW warning sets the traffic risk to 1, otherwise it assumes value 0.
- DVE 4 $\in \mathbb{B} = \{0, 1\}$: environment risk low visibility (0—Low, no risk associated with low environment visibility; 1—High, there is a risk for low environment visibility). Poor weather conditions (rain, fog, etc.) and unlit roads detected from the vehicle's indicators and controls (e.g., wipers, rain detection sensor, fog lights, etc., and dipped headlights, high beams, etc., respectively), along with road and traffic attributes (e.g., high traffic density and high curvature of road until a certain distance ahead of the vehicle), are used to assess risk for low environment visibility [47]. At the current state of implementation, this parameter gives a discrete representation of the environmental conditions estimated only from wiper actuation state and rain detection sensor. In particular, if the wiper is active or the rain detection is true, as there is low visibility due to rain, this parameter is set to 1. Otherwise, it is set to 0.
- DVE 5 $\in \mathbb{B} = \{0, 1\}$: environment risk low audibility (0—Low, no risk associated with low audibility of warning stimuli by the environment and vehicle safety critical systems; 1—High, there is a risk for low audibility). It describes the in-cabin noise level that is monitored by a microphone. High noise level sets the environment risk low audibility to 1, otherwise, it is set to 0.

Each of these 5 parameters can assumes 2 possible values $\in \mathbb{B} = \{0, 1\}$, therefore, the DVE conditions set is composed by 32 states (2^5 combinations of the DVE parameters). All measured inputs for the DVE parameters calculation are read from CAN-bus (Controller Area Network) sequentially, and they are updated according to a sampling time of 33 ms.

2.2. Information Management System

The IMS is the control authority of the A-HMI, as it coordinates the information flow between vehicle and driver and manages visual, auditory, and haptic channels shared among various applications, according to driving safety criteria [43]. The generic application request, encoded as a vector that specifies the action and the priority class, is processed by the IMS. Specifically, taking into account the DVE state, the IMS performs the following adaptive decision-management functions [60]:

- change of output modality and channel (displays, audio devices, etc.);
- change of physical layout (color, font, format, etc.);
- I/O action coordination based on prioritization, i.e., termination, interruption, retardation, resumption, or suppression of output messages.

The IMS reply, formulated as a vector as well, sets out the outcome of the request (accepted or delayed), the selected channel(s), and the provision modality for the submitted request, suitably adapted for duration, enhancement, and simplification of DVE actual state, as indicated by the IMS rules. From the management of global information flow perspective, the IMS avoids conflicts and overload of information by queuing application messages appropriately on the basis of DVE state information. The control cycle of the IMS consists of a two-stage process, whose computational logic is implemented in two modules, respectively: (1) Filter and Queue; (2) Modality and Channel. Each module exploits a rules-based logic, developed according to its specific target, by using the requested action vector, the DVE state, and the channels status as inputs of the logic.

2.2.1. Filter and Queue Module

The Filter and Queue module accomplishes action prioritization and scheduling. The core of this module is built by a rules-based strategy, which is responsible for the manipulation of the flow of the messages. Application request(s) (messages, notifications, etc. to the driver) are filtered dynamically due to their base priority. The filtering is based on a truth table that describes a strategy, defined according to safety relevance criteria, of prioritization and scheduling of application requests to be forwarded to the driver on the basis of current DVE state [37]. It is precisely the dependency by actual DVE state that makes the filtering a dynamic process. This module includes the rules to enable or postpone the presentation of the incoming actions: negative outcome of the IMS means that the action will be queued; in contrast, with a positive outcome, we proceed to the Modality and Channel module.

2.2.2. Modality and Channel Module

The Modality and Channel module defines the suitable mode and adequate style of information presentation to the driver in order to minimize distraction effects. For example, in visually challenging situations due to high traffic density and/or poor weather conditions, visual presentation of information should be avoided. Furthermore, under illuminated conditions, visual messages are hard to perceive, and, therefore, auditory warnings should be preferred. Equivalently, in noisy environments, visual presentation should be preferred, as audio information could be missed by the driver. Similarly to Filter and Queue, this module uses a truth table to select the most suited modality in which the application requests should be provided to the driver on the basis of current DVE state [37]. A channel selector is also involved to choose the output channel, solving conflicting channel requests through preemption of high-priority requests over low-priority requests.

3. Problem Statement

The objective of the A-HMI is to provide a substantial contribution to:

- using all I/O on ADAS, IVIS, and nomadic devices safely;
- keeping driver's workload at a level that does not affect a safe driving performance;
- avoiding interference of multiple information sources;

- minimizing demanding interactions for the driver;
- decreasing the interaction complexity, according to the DVE state, via reconfiguration of the infotainment system;
- reducing response time for decision making.

All these objectives are driven by the design of the IMS, which performs the management functions of the A-HMI, i.e., meta-functions responsible for managing ADAS, IVIS, and nomadic devices with respect to their interaction with the driver (e.g., the block on phone calls in demanding driving situations). The definition of the relevant driving scenarios is the first step for the IMS design, which, as already mentioned, is formalized using the actions and DVE conditions. The actions are classified into a set of five priority classes, as follows.

- W: safety critical warnings (FCW, lane keeping, etc.).
- D: system interaction, etc.
- OP1: mandatory messages or important info related to the driving task (turn immediately notification, driver status, etc.).
- OP2: temporary info related to the driving task, requiring an action in the near future (high engine temperature, low oil pressure, etc.), or messages related to infotainment system.
- OP3: permanent status info related to the driving task, not requiring an action in the near future, or output messages related to secondary tasks (incoming phone call, chat notification, etc.).

Concerning the DVE conditions, as described in Section 2.1, they can be discretely represented by several parameters. The IMS decision-management process also takes into account the length of the actions' waiting queue through the parameter LQue, described as follows.

- LQue $\in \mathbb{B} = \{0, 1\}$: queue state (0—Low, no waiting queue for message, notifications, indications, etc.; 1—High, there is a waiting queue).

When receiving a request vector from the applications with instruction about actions that have to be dispatched, the IMS applies in sequence:

1. the strategy implemented in the Filter and Queue module, described in Section 2.2.1, that is based on truth tables describing prioritization rules for applications' requests as functions of DVE states;
2. the rules implemented in the Modality and Channel module, described in Section 2.2.2, which, too, are founded on truth tables containing criteria for selecting representation modality of messages, notifications, indications, etc. suited to DVE states.

Therefore, to schedule output requests from the applications, and to select modality, channel, and layout for their presentation, a key task concerns the design of truth tables.

This study proposes a methodology to derive truth tables for the Filter and Queue module, which can be easily extended to also define truth tables for the Modality and Channel module. A general formulation of a truth table is represented in Table 1 for scheduling actions (messages/notifications/indications to be forwarded to the driver) on the basis of $n \in \mathbb{R}$ DVE parameters (DVE 1, ..., DVE n), queue state (LQue), and $m \in \mathbb{R}$ application requests (App. 1, ..., App. m).

More specifically, for prioritization purpose, action-specific logical expressions (truth functions) $f_i(\text{DVE 1}, \ldots, \text{DVE } n, \text{LQue}, \text{App. 1}, \ldots, \text{App. } m)$, with $i \in \{1, \ldots, m\}$ the number of actions, sets out functional values in the Boolean domain $f_i \in \mathbb{B} = \{0, 1\}$ for each combination of truth values taken by the $n + 1 + m$ variables (the arity of the function), i.e., DVE 1, ..., DVE n, LQue, App. 1, ..., App. m, which, too, are defined in the Boolean domain $\mathbb{B}$. Through these functional values, the truth table, based on current DVE states, suggests if messages could be forwarded to the driver (value 1) or not (value 0).

The information management process must consider, therefore, $n + 1 + m$ parameters among DVE states, queue length, and applications requests, each of them with two possible discrete values $\in \{0, 1\}$, whose combination gives rise to 2^{n+1+m} possible driving scenarios.

Furthermore, for a specific driving scenario, i.e., considering a single set of values for the n DVE states and queue length, there are $2^m + 2^{m-1} + \ldots + 2^0$ possible outcomes for scheduling the m actions. Herein, there are $2^{n+1}\left(2^m + 2^{m-1} + \ldots + 2^0\right)$ total cases to be investigated in order to optimize the logical expressions f_i and to derive the truth table. The design of the truth tables employed in the rules-based strategies of IMSs is in general carried out empirically, starting with the analysis of [43]:

- collection of solutions to relevant driving scenarios and to different use cases;
- requirements of the A-HMI and other indications coming from the technical literature state-of-the-art;
- the perceived effects assessing proposals of HMI adaptation, made by a pool of experts.

Since the decision-management process generally covers a large number of possible driving scenarios, a formal and generalized method to define truth tables for the filtering processes is needed. In the next section, a methodology to develop truth tables for IMS's Filter and Queue module is presented.

Table 1. Truth table for prioritization of actions.

DVE, Queue, Application Requests	DVE 1
	…
	DVE n
	LQue
	App. 1
	…
Outcome of Requests	App. m
App. 1	$f_1\left(\text{DVE }1, \ldots, \text{DVE }n, \text{LQue}, \text{App. }1, \ldots, \text{App. }m\right) : \mathbb{B}^{n+1+m} \to \mathbb{B}$
…	…
App. m	$f_m\left(\text{DVE }1, \ldots, \text{DVE }n, \text{LQue}, \text{App. }1, \ldots, \text{App. }m\right) : \mathbb{B}^{n+1+m} \to \mathbb{B}$

4. Design of an Optimization Framework for the IMS

In this section, the methodology proposed for tackling and solving the offline truth tables estimation problem is detailed. As mentioned, a generalized approach is derived to define truth tables to be used in the Filter and Queue module for prioritization of actions. In particular, the estimation problem is formulated as an optimization task where the objective is to find a set of rules to manage information flow between vehicle and driver that minimizes both the driver workload and the queuing of the undelivered messages/notifications/indications to the driver. In this optimization task, whose overall scheme is depicted in Figure 2, the evaluation of the effects of application requests on driver cognitive load and on actions waiting queue plays a key role. Herein, two indexes $J_{Dw}(\mathbf{x})$ and $J_{Aq}(\mathbf{x})$, on the basis of actual DVE states and messages to be forwarded to the driver, are used to estimate the driver workload and actions queue, respectively. These two indexes are exploited to identify the functional values of truth tables by searching the $x_i \in \mathbb{B}$ action-specific multipliers, with $i \in \{1, \ldots, m\}$, belonging in a vector of decision variables $\mathbf{x} \in \mathbb{B}^m$, which simultaneously minimize $J_{Dw}(\mathbf{x})$ and $J_{Aq}(\mathbf{x})$. These are two conflicting objectives that make the process a multi-objective optimization problem, addressed in the next sections.

4.1. Driver Workload Index

The main objective of the IMS's Filter and Queue module is to schedule application requests in order to avoid driver cognitive overload. To achieve this goal, the truth tables used in the filtering process for the prioritization of actions must be derived to minimize the driver's level of workload associated with forwarding application requests to the driver. To optimize the filtering process, a methodology to assess driving workload in several conditions is essential. Many studies have focused on evaluating the driver's resources allocated to a driving task [61,62]. These have shown that driver workload has three main features. First, drivers are capable of reporting the task demands on separate workload dimensions (perceptual, cognitive, and motor dimensions) [63]. Second, age differences significantly affect driver workload [64]. Third, performance measurements may estimate workload only partially because of the potential dissociation of performance and mental workload [65]. In accordance with these three properties, a workload assessment method is expected to capture the multi-dimensional property of the cognitive load and to account for driver age differences. Among several techniques to assess the level of the driver's workload, subjective methods are the most frequently used in practice [66]: they make use of driver's reports concerning subjective judgments of the effort and expenditure that was experienced during the task [67]. The most popular subjective method is the National Aeronautic and Space Administration Task Load indeX (NASA-TLX) [68], which was originally designed for aviation pilots. It assesses the workload through six rating scales, i.e., mental demand, physical demand, temporal demand, performance, effort, and frustration levels. The Driving Activity Load Index (DALI) [69] is a technique for workload estimation obtained by revising the NASA-TLX to specifically adapt it to the car driving task. Similarly to NASA-TLX, the DALI estimation is based on six rating scales, where, however, the main factors composing the workload score were chosen to be more adapted to the car driving context. In particular, workload dimensions of the DALI are effort of attention, visual demand, auditory demand, temporal demand, interference, and situational stress.

In this study, the DALI method is used to evaluate offline the workload under various complexities of the driving context. More specifically, several driving conditions are set up to induce on purpose various levels of workload for the driver, i.e., with and without secondary activities, under several different states of the driver–vehicle–environment system, and by varying application requests. During each driving condition, subjective measures are executed to assess the magnitude of the six factors on a scale. Then, the levels of workloads are quantified by comparing the sets of six factors evaluated in the tested driving conditions. In doing so, relative weights on driver workload between various DVE states $\left(c_{\mathrm{DVE}\ j}, \text{with } j \in \{1,\ \ldots,\ n\}\right)$ and several applications requests $\left(c_{\mathrm{App.}\ i}, \text{with } i \in \{1,\ \ldots,\ m\}\right)$ are computed. Identification of these relative weights is a preliminary step to calculate the driver workload index $J_{Dw}(\mathbf{x})$. Indeed, it is defined as a weighted sum of the DVE parameters $\left(\mathrm{DVE}\ j, \text{with } j \in \{1,\ \ldots,\ n\}\right)$ and application request $\left(\mathrm{App.}\ i, \text{with } i \in \{1,\ \ldots,\ m\}\right)$ contributions on driver cognitive load:

$$K = \sum_{j=1}^{n} \mathrm{DVE}\ j \cdot c_{\mathrm{DVE}\ j} \tag{1}$$

$$J_{Dw} = K \cdot \sum_{i=1}^{m} x_i \cdot \mathrm{App.}\ i \cdot c_{\mathrm{App.}\ i} \tag{2}$$

where each decision variable x_i assumes a value $\in \{0,\ 1\}$ that determines if the multiplied action $(\mathrm{App.}\ i)$ is executed or not.

4.2. Actions Queue Index

Application requests are classified into a set of priority classes (W, D, OP1, OP2, and OP3, described in Section 3) on the basis of their relevance regarding safety, mobility, and the impact on the level of driver cognitive load [34]. Every one of these three dimensions is ranked by its magnitude (from 'low' to 'high') given a-priori statically [36]. Hence, their combination can be interpreted as an indicator for the base priority of an application request. For example, the priority of a turn indicator's request is described by:

- safety relevance = high;
- mobility relevance = medium;
- workload impact = medium.

A comparison between the base priority of several possible application requests allows us to determine their relative weights on the actions queue $\left(p_{\text{App. }i}, \text{with } i \in \{1, \ldots, m\}\right)$. More specifically, each action weight represents the relative effect on waiting queue derived by not executing the action and queuing it. Through these, the actions queue index can be calculated as a weighted sum of the application requests' $\left(\text{App. } i, \text{with } i \in \{1, \ldots, m\}\right)$ contributions to the queue:

$$J_{Aq} = \sum_{i=1}^{m} \left(p_{\text{App. }i} \cdot cos(x_i) \cdot \text{App. } i\right) + \left[\text{LQue} \cdot \left(p_{Q,\text{ App. }i} \cdot cos(x_i) \cdot \text{App. } i\right)\right] \tag{3}$$

Moreover, when a long waiting queue is assessed $(\text{LQue} = 1)$, to each application request is associated a multiplier $p_{Q,\text{ App. }i}$, with $i \in \{1, \ldots, m\}$, that increases the cost of not executing the corresponding action.

4.3. Multi-Variable Multi-Objective Optimization

As mentioned, to simultaneously achieve the two conflicting goals of minimizing both the driver workload level and the length of actions waiting queue, the truth table estimation problem is formulated as a multi-objective optimization process. More specifically, a multi-variable multi-objective optimization process has been proposed according to the scheme represented in Figure 2, which is defined as follows [70]:

$$min\, \mathbf{J}(\mathbf{X}) = min\left(\mathbf{J_{Dw}}(\mathbf{X}), \mathbf{J_{Aq}}(\mathbf{X})\right) \tag{4}$$

where $\mathbf{X} \in \mathbb{B}^{m \times 2^{(n+1+m)}}$ is a space of decision variables, $\mathbf{J}(\mathbf{X}) : \mathbb{B}^{m \times 2^{(n+1+m)}} \to \mathbb{R}^{2 \times 2^{(n+1+m)}}$ is a space of objectives, $\mathbf{J_{Dw}}(\mathbf{X}) : \mathbb{B}^{m \times 2^{(n+1+m)}} \to \mathbb{R}^{2^{(n+1+m)}}$ is a vector of driver workload indexes, and $\mathbf{J_{Aq}}(\mathbf{X}) : \mathbb{B}^{m \times 2^{(n+1+m)}} \to \mathbb{R}^{2^{(n+1+m)}}$ is a vector of actions queue length indexes. In this process, the inputs are distinguished in:

- $n + 1 + m$ parameters that describe the driving scenario, i.e., DVE 1, …, DVE n, LQue, App. 1, …, App. m;
- m decision variables $x_i \in \mathbb{B}$, each of them associated with one of the m application requests App. i, and determines if the corresponding action is executed $(x_i = 1)$ or not $(x_i = 0)$.

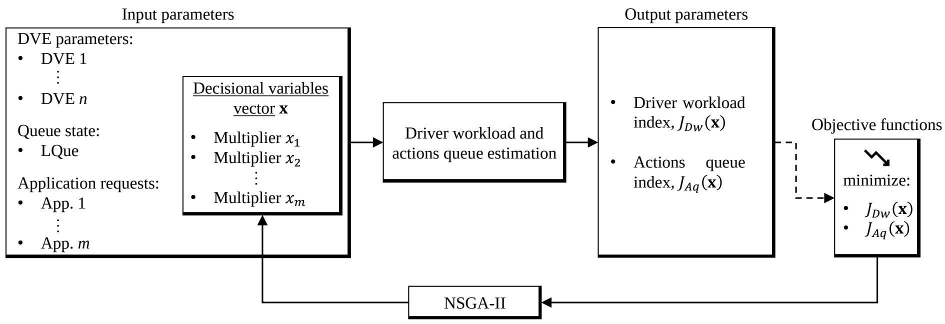

Figure 2. Scheme of the multi-variable multi-objective optimization.

From these inputs, driver workload level and length of action waiting queue are estimated by leveraging the two indexes $J_{Dw}(\mathbf{x})$ and $J_{Aq}(\mathbf{x})$ described in Sections 4.1 and 4.2, respectively. The objective of the proposed optimization procedure is to find, for each of 2^{n+1+m} possible driving scenarios, a vector of decision variables $\mathbf{x} = [x_1, \ldots, x_m]^T$ that simultaneously minimizes the two indexes $J_{Dw}(\mathbf{x})$ and $J_{Aq}(\mathbf{x})$. The trade-off between these two conflicting goals can be optimized by exploiting a Non-dominated Sorting Genetic Algorithm II (NSGA-II) [71] in order to search the optimal decision variables. The NSGA-II, first, randomly initializes a parent population of individuals. All individuals are then sorted into different front levels, and to each front level is assigned a rank, which equals its non-domination level. In the same front level, the location of the finite number of solutions is expected to be distributed uniformly. Therefore, the crowding distance criterion is adopted to select solutions by evaluating the local aggregation of individuals. Since extreme points are desired to be kept every generation, they are assigned a crowding distance to the maximum value. After sorting, the selection operator, crossover operator, and mutation operator are implemented to generate the offspring population. Thereafter, the new parent population is generated by being filled with non-dominated solutions. Then, the optimization is completed by the iterative application of the procedure until the number of maximum generation is reached.

At each iteration step, a space of decision variables $\mathbf{X}$ is determined, containing the decision variable vectors $\mathbf{x}_\mu$ of the $2^{(n+1+m)}$ possible driving scenarios, with $\mu \in \left\{1,\ldots,2^{(n+1+m)}\right\}$. The quality of each searched decision variables vector $\mathbf{x}_\mu$ is then evaluated by the two indexes J_{Dw} and J_{Aq} to be minimized. Repeating this evaluation over the whole space of decision variables, two vectors of driver workload indexes and actions queue length indexes, $\mathbf{J_{Dw}}(\mathbf{X})$ and $\mathbf{J_{Aq}}(\mathbf{X})$, respectively, are determined. In doing so, a $2 \times 2^{(n+1+m)}$-dimensional hyperspace of objectives $\mathbf{J}(\mathbf{X})$ is constructed. Among the non-dominated solutions of the Pareto front, optimal solutions are identified according to the criterion of minimum distance to the origin of the objectives hyperspace by normalizing the objective functions.

5. Results and Analysis

In this section, we evaluate if the proposed optimization framework, described in Section 4, could be successfully used to derive truth tables for scheduling application requests in order to minimize both the driver workload and the queuing of actions. The analysis concerns with the development of the truth table are represented in Table 2. This truth table is suitable for scheduling of $m = 5$ application requests, each of them belonging to one of the priority classes listed in Section 3: FCW $\in$ W; system interaction $\in$ D; turn indicators $\in$ OP1; volume increase of infotainment system $\in$ OP2; phone call $\in$ OP3.

Specifically, the truth table is carried out by defining five truth functions f_i, with $i \in \{1, \ldots, 5\}$, which suggest if the application requests could be executed or not. In this decision process are involved 11 parameters: $n = 5$ DVE parameters, i.e., DVE 1 , DVE 2, DVE 3, DVE 4, and DVE 5, measured by the status recognition and monitoring system, that, as described in Section 2.1, allow us to represent 32 possible DVE states; queue length parameter LQue; and $m = 5$ application requests (App. 1 = FCW, App. 2 = system interaction, App. 3 = turn indicators, App. 4 = volume increase, App. 5 = phone call) to take into account multiple and simultaneous actions.

5.1. Implementation of the Optimization Framework

As mentioned, the 11 inputs in the optimization problem have discrete values in the Boolean domain. Herein, the truth table complies with 2^{11} possible driving scenarios. As mentioned in Section 2.1, the parameter DVE 3, on the basis of FCW status, gives a discrete evaluation of collision risk with other vehicles. By this direct relationship between the DVE 3 and the App. 1, the variables of the decision problem are reduced to 10, and the number of scenarios decreases in turn. Moreover, considering that for high risk of collision with other vehicles in path, no messages, notifications, indication, etc. can be delivered to the driver, the decision problem can be further simplified. Indeed, to an FCW warning status, and therefore to a traffic risk (DVE 3) set to 1, the IMS can only formulate a reply that commands to deliver the warning alone. Hence, the possible driving scenarios can be reduced to $1 + 2^9$.

To impose no notifications to the driver when FCW is enabled, the coefficient $c_{\text{DVE 3}}$ in Equations (1)–(3) assumes a value two orders of magnitude greater than other coefficients in the same equations ($c_{\text{DVE 3}} = 1000$), and the Expression (2) of the driver workload index is reformulated as follows:

$$\begin{aligned} K =&(\text{DVE 1} \cdot c_{\text{DVE 1}}) + (\text{DVE 2} \cdot c_{\text{DVE 2}}) + (\text{DVE 3} \cdot 1000) \\ &+ (\text{DVE 4} \cdot c_{\text{DVE 4}}) + (\text{DVE 5} \cdot c_{\text{DVE 5}}) \end{aligned} \tag{5}$$

$$J_{Dw} = (\text{DVE } 3 \cdot 1000) + K \cdot \sum_{i=2}^{5} x_i \cdot \text{App. } i \cdot c_{\text{App. } i} \tag{6}$$

Moreover, the optimization problem is subject to an inequality constraint defined as:

$$(J_{Dw} - 1000) \cdot \text{DVE } 3 \leq 0 \tag{7}$$

that forces optimization to suppress all the notifications during FCW state.

Table 2. Truth table for scheduling: FCW; system interaction; turn indicators; volume increase of infotainment system; phone call.

DVE and Application Request / **Outcome of Request**	DVE 1 DVE 2 DVE 3 DVE 4 DVE 5 LQue App. 1 = FCW App. 2 = system interaction App. 3 = turn indicators App. 4 = volume increase App. 5 = phone call
FCW	$f_1(\text{DVE } 1, \ldots, \text{DVE } 5, \text{LQue}, \text{App. } 1, \ldots, \text{App. } 5) : \mathbb{B}^{11} \to \mathbb{B}$
system interaction	$f_2(\text{DVE } 1, \ldots, \text{DVE } 5, \text{LQue}, \text{App. } 1, \ldots, \text{App. } 5) : \mathbb{B}^{11} \to \mathbb{B}$
turn indicators	$f_3(\text{DVE } 1, \ldots, \text{DVE } 5, \text{LQue}, \text{App. } 1, \ldots, \text{App. } 5) : \mathbb{B}^{11} \to \mathbb{B}$
volume increase	$f_4(\text{DVE } 1, \ldots, \text{DVE } 5, \text{LQue}, \text{App. } 1, \ldots, \text{App. } 5) : \mathbb{B}^{11} \to \mathbb{B}$
phone call	$f_5(\text{DVE } 1, \ldots, \text{DVE } 5, \text{LQue}, \text{App. } 1, \ldots, \text{App. } 5) : \mathbb{B}^{11} \to \mathbb{B}$

The truth table derived must address $1 + 2^9$ possible different driving scenarios. Such a finite number of scenarios can be solved offline using the proposed optimization approach. As described in [72,73], the exploited optimization procedure, implemented in Python environment (see Supplementary Materials Pseudocode S1), by leveraging the NSGA-II, can search the driver workload index J_{Dw} and the action queue index J_{Aq} optimal parameters. These two indexes are characterized by 10 inputs and 21 coefficients. The 10 inputs (DVE 1, …, DVE 5, LQue, App. 2, …, App. 5) are used to take into account possible driving scenarios addressed by the truth table. The 21 coefficients consist of 9 weights to consider effects of DVE parameters and application requests on driver workload $\left(c_{\text{DVE } j}\right.$ and $c_{\text{App. } i}$, respectively$\left.\right)$, 8 weights on impact of application requests on actions queue $\left(p_{\text{App. } i}\right.$ and $\left.p_{Q,\,\text{App. } i}\right)$, and 4 decision variables $(x_i \in \mathbb{B})$ to be identified. More specifically, the four decision variables, which indicate if the corresponding four actions are executed, are tuned during the optimization process to minimize both the driver cognitive load and messages queue. The population size, which is the parameters space of potential solutions in which the NSGA-II searches the optimal value of these unknown parameters, is set according to the technical literature to 1000. Regarding, instead, the crossover and the mutation, the simulated binary crossover method and the polynomial mutation are implemented (for details about the mentioned techniques see [74]).

Table 3 reports the weights used to calculate the driver workload and action queue indexes, evaluated following the procedures described in Sections 4.1 and 4.2, respectively.

Table 3. Weights of driver workload and action queue indexes.

Coefficient	Value	Coefficient	Value
$c_{\text{DVE 1}}$	10 [-]	$p_{\text{App. 2}}$	5 [-]
$c_{\text{DVE 2}}$	20 [-]	$p_{\text{App. 3}}$	4 [-]
$c_{\text{DVE 4}}$	10 [-]	$p_{\text{App. 4}}$	3 [-]
$c_{\text{DVE 5}}$	2 [-]	$p_{\text{App. 5}}$	2 [-]
$c_{\text{App. 2}}$	2 [-]	$p_{\text{Q, App. 2}}$	4 [-]
$c_{\text{App. 3}}$	1.5 [-]	$p_{\text{Q, App. 3}}$	4 [-]
$c_{\text{App. 4}}$	1 [-]	$p_{\text{Q, App. 4}}$	3 [-]
$c_{\text{App. 5}}$	2 [-]	$p_{\text{Q, App. 5}}$	2 [-]

Implementation of the Optimized Filtering Strategy

The optimized truth table served as the foundation for the development of the project demonstrator, as depicted in Figure 3. This is an intelligent system that, by collecting and processing information from various sources, is capable of prioritizing safety, customization, and distraction-free interaction for drivers. It was instrumental in developing a multi-modal A-HMI for the concept car named "Kinecar" [75], an innovative project spearheaded by Kineton. In this communication framework, applications convey their requests via Ethernet and CAN-bus to the Kinecar IVIS, equipped with a 15-inch display and based on the Renesas R-Car-H3e board. Within this framework, the A-HMI manages these requests using a request-response mechanism. Notably, the IVIS engages in Ethernet-based interactions with the Camera Control Unit (CCU), which is driven by the Nvidia Jetson AGX Xavier platform. The CCU plays as the central intelligent system of the whole setup, responsible for executing both the IMS and the DVE state recognition and monitoring system, detailed in Section 2. More specifically, the CCU's operations include running monitoring system modules, crucial for evaluating the five DVE parameters (described in Section 2.1) related to traffic and environmental conditions [55,56], driver distraction [76] and PERCLOS [77]. These evaluations are carried out by processing measurements from a driver monitoring camera and a forward-facing camera, both received via USB. Subsequently, the CCU, utilizing the DVE parameters, filters these requests based on the previously optimized offline truth table, effectively scheduling the necessary actions.

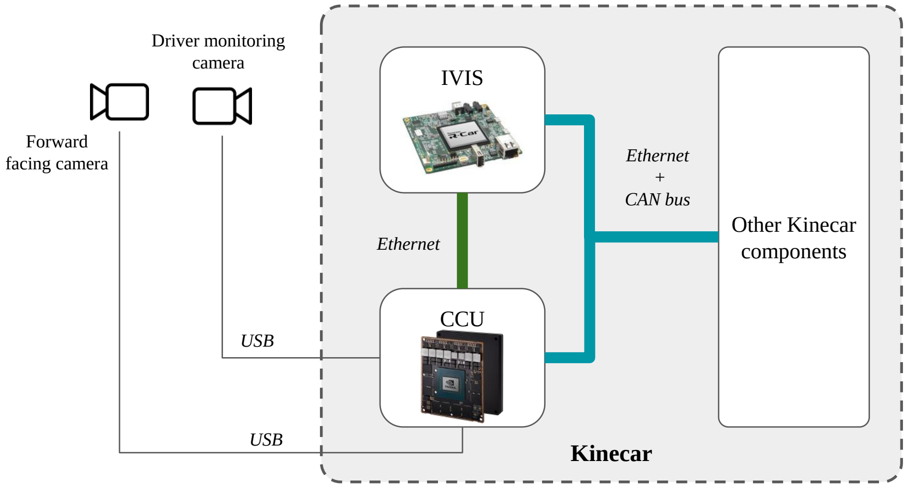

Figure 3. Scheme of the physical implementation of the offline optimized truth table for the Filtering strategy in Kinecar's A-HMI.

5.2. Results

The number of generations in a multi-objective optimization procedure is set to cover all possible combinations of DVE states and application requests. Solutions belong to the 1026-dimensional hyperspace $\left(\mathbf{J}(\mathbf{X}) \in \mathbb{R}^{2\times 513}\right)$ constituting the Pareto frontier.

Indeed, for each of the 513 input combinations, an optimal solution set of 4 decision variables is determined. In doing so, the output of the optimization process is a vector of decision variables of dimension $\mathbf{X} \in \mathbb{B}^{4\times513}$. For the sake of brevity, only some sample outputs $\mathbf{x}_\mu \in \mathbf{X}$, with $\mu \in \{1, \ldots, 513\}$, are here theoretically analyzed to highlight effectiveness of the proposed procedure to derive truth tables for scheduling application requests. Specifically, solutions for three driving scenarios are presented below.

- Scenario 1: the Filter and Queue module must schedule the driver request to increase the volume of the infotainment system and an incoming phone call. The decision management process is carried out, taking into account the output from the DVE state recognition and monitoring system, which infers a high driver workload level. The input vector of this driving scenario is reported in Table 4.
- Scenario 2: wipers actuation state, measurements of rain detection sensors, and microphone warn of a high level of environmental risk for both low visibility and audibility. According to these conditions, the truth table must filter two application requests, i.e., activate turn indicators and an incoming phone call. Table 5 shows the corresponding input vector.
- Scenario 3: the monitoring system detects high risk in four of the five dimensions of the driver–vehicle–environment system, i.e., driver distraction, driver drowsiness, environment low visibility, and environment low audibility. The IMS addresses a complex decision management problem, where, in the high waiting queue condition, three requests are performed, i.e., system interaction, activate turn indicators, and increase the volume of the infotainment system. This driving scenario corresponds to the input vector in Table 6.

Table 4. Input vector for Scenario 1.

Input (DVE Parameters)	Value	Input (Queue State, App. Requests)	Value
DVE 1	1 [-]	LQue	0 [-]
DVE 2	0 [-]	App. 2 (system interaction)	0 [-]
DVE 3, App. 1 (FCW)	0 [-]	App. 3 (turn indicators)	0 [-]
DVE 4	0 [-]	App. 4 (volume increase)	1 [-]
DVE 5	0 [-]	App. 5 (phone call)	1 [-]

Table 5. Input vector for Scenario 2.

Input (DVE Parameters)	Value	Input (Queue State, App. Requests)	Value
DVE 1	0 [-]	LQue	0 [-]
DVE 2	0 [-]	App. 2 (system interaction)	0 [-]
DVE 3, App. 1 (FCW)	0 [-]	App. 3 (turn indicators)	1 [-]
DVE 4	1 [-]	App. 4 (volume increase)	0 [-]
DVE 5	1 [-]	App. 5 (phone call)	1 [-]

Table 6. Input vector for Scenario 3.

Input (DVE Parameters)	Value	Input (Queue State, App. Requests)	Value
DVE 1	1 [-]	LQue	1 [-]
DVE 2	1 [-]	App. 2 (system interaction)	1 [-]
DVE 3, App. 1 (FCW)	0 [-]	App. 3 (turn indicators)	1 [-]
DVE 4	1 [-]	App. 4 (volume increase)	1 [-]
DVE 5	1 [-]	App. 5 (phone call)	0 [-]

From the single driving scenario perspective, the solutions of the optimization process can be represented in scenario-specific two-dimensional surfaces. Therefore, on each of these surfaces belong possible outcomes identified for scheduling the four actions for a specific input vector, i.e., for a single set of DVE parameters and applications requests.

Figure 4 shows solutions identified by the proposed optimization approach for Scenario 1 in terms of driver workload $J_{Dw}(\mathbf{x})$ and actions queue $J_{Aq}(\mathbf{x})$ indexes, both normalized. Among the four feasible outcomes, the optimal output is represented by a red dot in the figure, which corresponds to the output vector in Table 7. Since the monitoring system, by driver head and eye movements, detects driver distraction, the truth table should suggest to reject an incoming phone call for minimizing workload level. In contrast, to minimize the action queue, it should allow to increase the volume of the infotainment system.

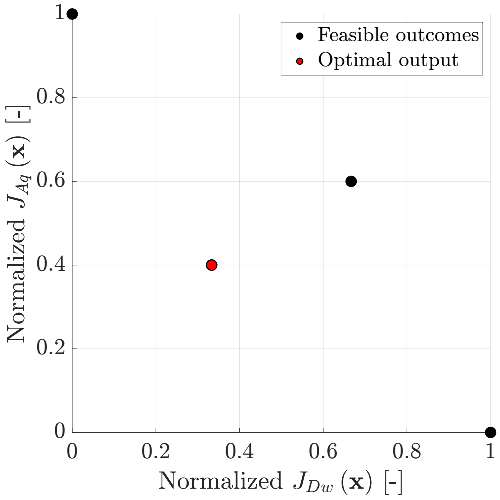

Figure 4. Solutions for Scenario 1: black dots are feasible outcomes; red dot is the optimal output.

Table 7. Output vector for Scenario 1.

Output (Action)	Value	Output (Action)	Value
system interaction	0 [-]	volume increase	1 [-]
turn indicators	0 [-]	phone call	0 [-]

Similarly, Figure 5 shows, in the plane of the normalized $J_{Dw}(\mathbf{x})$ and $J_{Aq}(\mathbf{x})$, four possible outcomes for Scenario 2 calculated during the optimization process.

In addition, here, the red dot is the optimal solution, whose output vector is represented in Table 8. In visually challenging situations due to poor weather conditions and in noisy environments, the truth table should reject an incoming phone call to avoid dangerous situations due to driver overload. The request to activate turn indicators, on the other hand, should be accomplished to reduce traffic crashes risk.

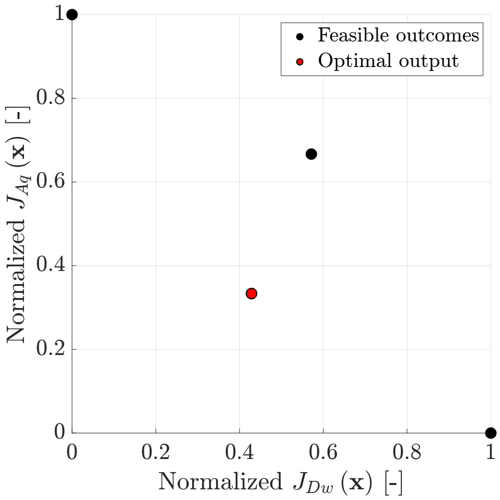

Figure 5. Solutions for Scenario 2: black dots are feasible outcomes; red dot is the optimal output.

Table 8. Output vector for Scenario 2.

Output (Action)	Value	Output (Action)	Value
system interaction	0 [-]	volume increase	0 [-]
turn indicators	1 [-]	phone call	0 [-]

Finally, Figure 6 shows optimization process results for Scenario 3, which is a challenging scenario for the decision management process. Indeed, DVE parameters show high driver workload level, driver physical impairment affecting driving performance, as well as poor environment visibility and audibility that decrease driver's attention and reaction level. Under these poor DVE conditions, the IMS must schedule many application requests (three of four possible requests) while facing a high waiting queue for actions. All these factors make the decision management problem complex.

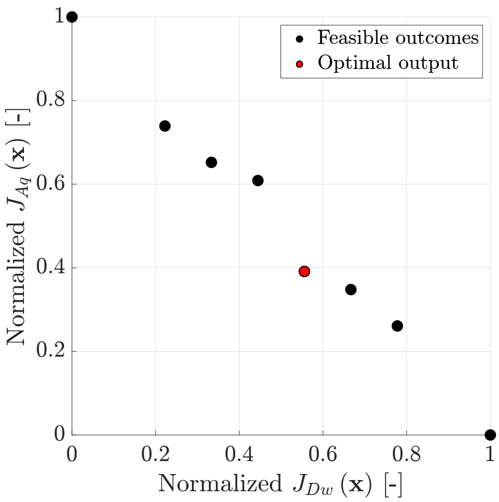

Figure 6. Solutions for Scenario 3: black dots are feasible outcomes; red dot is the optimal output.

The identified solution, in Table 9, optimizes the trade-off between driver workload and action queue minimization. Specifically, the turn indicators request is accepted since it is significant for traffic safety; the same response is given for the action of increase the volume of the infotainment system in order to decrease the waiting queue. In contrast, the truth table should reject system interaction for avoiding driver cognitive overload.

Table 9. Output vector for Scenario 3.

Output (Action)	Value	Output (Action)	Value
system interaction	0 [-]	volume increase	1 [-]
turn indicators	1 [-]	phone call	0 [-]

5.3. Discussion

This paper introduces an innovative methodology aimed at providing a more generalized and adaptable approach for designing rules driving the adaptive strategies of A-HMIs. The primary objective of this methodology is to design an A-HMI architecture geared towards reducing driver distraction. Within this architecture, the A-HMI effectively manages application requests through a request-response mechanism. Various applications submit requests for driver interaction, and these requests are overseen by the IMS. The IMS employs a combination of rules-based strategies that allow decisions capable of minimizing driver distraction. These strategies could encompass Limiting, Simplifying, Filtering, Delaying, Activating, Advancing, Supplementing, and Augmenting, among others [26]. Such rules-based strategies are implemented using truth tables. These truth tables take as input requests of the applications and the DVE parameters, generating action-management decisions as output. DVE parameters are assessed through a DVE state recognition and monitoring system, which processes data from in-vehicle sensors and cameras. It is this dependence on the current DVE state that contributes to endowing the strategies with adaptability [37]. At the core of the proposed methodology to design rules for adaptivity strategies lies the application of multi-objective optimization. Using this optimization methodology, this study is focused on designing a truth table specifically tailored for a Filtering strategy. According to this strategy, the truth table has been designed for optimizing attentional demands to mitigate overload or underload. The overarching goals of this optimization endeavor were twofold: to minimize the driver's cognitive workload and reduce actions queuing. To achieve these objectives, a crucial component of the optimization process was the comprehensive evaluation of how requests of applications impact the driver's cognitive load and the queue of pending actions. To facilitate this assessment, two indexes were introduced, grounded in the actual state of the DVE and requests of the applications. These indexes serve as invaluable tools for estimating the driver's cognitive workload and the queue of pending actions throughout the optimization procedure. The proposed approach has undergone evaluation through a case study, with the objective of formulating a truth table tailored for scheduling five application requests: FCW, system interaction, turn indicators, volume adjustment of the infotainment system, and phone calls. These requests are inputs of the truth table, together with the five DVE parameters assessed via the DVE state recognition and monitoring system. Additionally, the IMS decision-making process takes into account the length of the action waiting queue, as represented by the parameter LQue. Collectively, these elements contribute to the development of a truth table characterized by 11 inputs, each with discrete values within the Boolean domain. Notably, this truth table encompasses $2^{11} = 2048$ potential driving scenarios. It is important to note that, considering the simplifications outlined in Section 5.1, the truth table condenses to encompass $1 + 2^9 = 513$ feasible driving scenarios. Furthermore, for each distinct DVE system state and queue length, there exist $2^5 + 2^4 + \ldots + 2^0 = 63$ plausible outcomes concerning the scheduling of the five actions. Consequently, the total cases to be investigated for optimizing the 513 possible outcomes of the truth table amount to $2^{4+1}\left(2^5 + 2^4 + \ldots + 2^0\right) = 2016$. The study's results theoretically demonstrated the remarkable capability of the proposed optimization framework in automatically addressing this large amount of feasible driving scenarios. This capability has enabled the derivation of an effective strategy for prioritizing actions requested by applications. Compared to existing methodologies documented in the technical literature [25,28,39], the proposed approach has significantly streamlined the design of rules driving the Filtering strategy within the A-HMI architecture. Notably, it has endowed the system with adaptive capabilities, effectively mitigating driver cognitive overload. Indeed, many known approaches for designing predefined rules for adaptive strategies heavily rely on empirical methods, leaning on the competences of expert pools [43]. However, when facing a large number of possible driving scenarios and in-vehicle applications to be managed, this conventional approach proves to be complex and time-consuming. The process of deriving truth tables, in particular, escalates in complexity as the number of DVE parameters and application requests to be managed increase. More specifically, the total cases to be investigated for optimizing the truth table exponentially increase with these two variables. This complexity issue is compounded by the sheer volume of potential driving conditions and the extensive data handling necessitated by new ICVs [16]. In stark contrast, our proposed optimization approach significantly simplifies the rule design process for the Filtering strategy, making A-HMI development within the context of ICVs automatic and less labor-demanding. Furthermore, our methodology facilitates the formulation of comprehensive strategies to effectively manage the diverse tasks associated with the information management process [41]. It is noteworthy that the multi-objective optimization procedure has explored all combinations of inputs, i.e., DVE states and application requests. Specifically, for each of the 513 input combinations, an optimal solution set of 4 decision variables, each with value in the Boolean domain, has been determined. Each decision variable corresponds to one of the application requests and dictates whether the corresponding action is executed or not. Consequently, the outcome of our optimization process is a vector of decision variables with a dimension of $\mathbf{X} \in \mathbb{B}^{4\times513}$. In contrast, when adhering to conventional empirical approaches, obtaining comprehensive strategies to cope with all diverse tasks involved in the information management process is not a trivial task [42]. This issue accentuates a well-recognized issue of rules-based strategies. Specifically, the brittleness of rules-based A-HMIs becomes evident when faced with missing or unforeseen input values, which can compromise the system's adaptability and robustness.

The main limitation of the proposed methodology pertains to the user-specific characteristics of the optimized truth table. It is essential to recognize that the truth table should exhibit adaptability over time to accommodate changes in a driver's experience and the effects of aging. Indeed, age differences represent one of the most significant factors influencing driver workload. The aging process induces a deceleration in information processing across perceptual, cognitive, and psychomotor domains for older drivers [64]. Consequently, when processing an equivalent amount of information within a defined time period, older drivers typically experience higher levels of mental workload in contrast to their younger counterparts. Furthermore, rules-based strategies should be attuned to changes of individual drivers since the assessment of workload levels may diverge from one user to another, owing to differences in the multi-dimensional properties of cognitive load [78]. One promising solution for mitigating this limitation, which will be a focal point of future research endeavors, consists of revolutionizing A-HMIs by applying cloud computing and Internet of Things (IoT) technologies of ICVs [79]. By harnessing these technologies, it can be possible to collect and transmit driver data, as well as other relevant information from the vehicle and environment, to a cloud-based platform. This platform can then be utilized to construct a comprehensive data-driven model of driver workload [80]. Using this model, it will become feasible to dynamically assess driver distraction and establish correlations between application requests and their impact on workload over time.

6. Conclusions

This paper introduces an innovative methodology, based on multi-objective optimization, to tackle the challenges of designing rules-based adaptive strategies for A-HMIs with the objective of reducing driver distraction. Using this optimization methodology, the study focuses on designing a truth table tailored for a Filtering strategy. More specifically, this truth table has been designed to optimize the scheduling of five application requests: FCW, system interaction, turn indicators, volume adjustment, and phone calls. This optimization has defined an adaptation strategy aimed at mitigating both driver's overload and underload, based on five DVE parameters and the length of the action waiting queue. A theoretical analysis has been carried out to assess the effectiveness of the proposed optimization framework. The results of the study have clearly demonstrated the remarkable effectiveness of the proposed framework in optimizing the Filtering strategy for actions requested by applications. By adopting this approach, the design of rules has been automated and significantly streamlined while gaining adaptive capabilities to prevent driver cognitive overload.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/app131910687/s1 or https://github.com/francescotufano91/A-HMI-optimization.git, Pseudocode S1: pseudocode, developed in Python environment, implementing the proposed optimization method.

Author Contributions: All authors contributed extensively to the work presented in this paper. Conceptualization, investigation, and writing, F.T.; conceptualization, methodology, and software, S.W.B.; conceptualization, review, and editing, M.T., L.N. and G.F.; review, formal analysis, and supervision, S.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Italian Ministry of Economic Development (MISE)'s Fund for Sustainable Growth (F.C.S) under grant agreement (CUP) B61B19000410008, project KINE-BRAIN ('Key Interaction among Entertainment and BRAIN').

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Haydari, A.; Yılmaz, Y. Deep reinforcement learning for intelligent transportation systems: A survey. *IEEE Trans. Intell. Transp. Syst.* **2020**, *23*, 11–32. [CrossRef]
2. Coppola, A.; Lui, D.G.; Petrillo, A.; Santini, S. Cooperative driving of heterogeneous uncertain nonlinear connected and autonomous vehicles via distributed switching robust PID-like control. *Inf. Sci.* **2023**, *625*, 277–298. [CrossRef]
3. Arthurs, P.; Gillam, L.; Krause, P.; Wang, N.; Halder, K.; Mouzakitis, A. A taxonomy and survey of edge cloud computing for intelligent transportation systems and connected vehicles. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 6206–6221. [CrossRef]
4. Petrillo, A.; Prati, M.V.; Santini, S.; Tufano, F. Improving the NOx reduction performance of an Euro VI d SCR System in real-world condition via nonlinear model predictive control. *Int. J. Engine Res.* **2023**, *24*, 823–842. [CrossRef]
5. Baratta, M.; Kheshtinejad, H.; Laurenzano, D.; Misul, D.; Brunetti, S. Modelling aspects of a CNG injection system to predict its behavior under steady state conditions and throughout driving cycle simulations. *J. Nat. Gas Sci. Eng.* **2015**, *24*, 52–63. [CrossRef]
6. Coppola, A.; Lui, D.G.; Petrillo, A.; Santini, S. Eco-driving control architecture for platoons of uncertain heterogeneous nonlinear connected autonomous electric vehicles. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 24220–24234. [CrossRef]
7. Albarella, N.; Lui, D.G.; Petrillo, A.; Santini, S. A Hybrid Deep Reinforcement Learning and Optimal Control Architecture for Autonomous Highway Driving. *Energies* **2023**, *16*, 3490. [CrossRef]
8. Bifulco, G.N.; Coppola, A.; Petrillo, A.; Santini, S. Decentralized cooperative crossing at unsignalized intersections via vehicle-to-vehicle communication in mixed traffic flows. *J. Intell. Transp. Syst.* **2022**, 1–26. [CrossRef]
9. Caiazzo, B.; Coppola, A.; Petrillo, A.; Santini, S. Distributed nonlinear model predictive control for connected autonomous electric vehicles platoon with distance-dependent air drag formulation. *Energies* **2021**, *14*, 5122. [CrossRef]
10. Musa, A.; Pipicelli, M.; Spano, M.; Tufano, F.; De Nola, F.; Di Blasio, G.; Gimelli, A.; Misul, D.A.; Toscano, G. A review of model predictive controls applied to advanced driver-assistance systems. *Energies* **2021**, *14*, 7974. [CrossRef]
11. Bengler, K.; Rettenmaier, M.; Fritz, N.; Feierle, A. From HMI to HMIs: Towards an HMI framework for automated driving. *Information* **2020**, *11*, 61. [CrossRef]
12. Kun, A.L. Human-machine interaction for vehicles: Review and outlook. *Found. Trends Hum. Comput. Interact.* **2018**, *11*, 201–293. [CrossRef]
13. Piechulla, W.; Mayser, C.; Gehrke, H.; König, W. Reducing drivers' mental workload by means of an adaptive man–machine interface. *Transp. Res. Part Traffic Psychol. Behav.* **2003**, *6*, 233–248. [CrossRef]
14. Hasenjäger, M.; Heckmann, M.; Wersing, H. A survey of personalization for advanced driver assistance systems. *IEEE Trans. Intell. Veh.* **2019**, *5*, 335–344. [CrossRef]
15. Feigh, K.M.; Dorneich, M.C.; Hayes, C.C. Toward a characterization of adaptive systems: A framework for researchers and system designers. *Hum. Factors* **2012**, *54*, 1008–1024. [CrossRef]
16. Tan, Z.; Dai, N.; Su, Y.; Zhang, R.; Li, Y.; Wu, D.; Li, S. Human–machine interaction in intelligent and connected vehicles: A review of status quo, issues, and opportunities. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 13954–13975. [CrossRef]
17. Nakagawa, T.; Nishimura, R.; Iribe, Y.; Ishiguro, Y.; Ohsuga, S.; Kitaoka, N. A human machine interface framework for autonomous vehicle control. In Proceedings of the 2017 IEEE 6th Global Conference on Consumer Electronics (GCCE), Nagoya, Japan, 24–27 October 2017; pp. 1–3.
18. Árnason, J.I.; Jepsen, J.; Koudal, A.; Schmidt, M.R.; Serafin, S. Volvo intelligent news: A context aware multi modal proactive recommender system for in-vehicle use. *Pervasive Mob. Comput.* **2014**, *14*, 95–111. [CrossRef]
19. Wiegand, G.; Mai, C.; Holländer, K.; Hussmann, H. Incarar: A design space towards 3D augmented reality applications in vehicles. In Proceedings of the 11th international conference on automotive user interfaces and interactive vehicular applications, Utrecht, The Netherlands, 22–25 September 2019; pp. 1–13.
20. Gray, R.; Ho, C.; Spence, C. A comparison of different informative vibrotactile forward collision warnings: Does the warning need to be linked to the collision event? *PLoS ONE* **2014**, *9*, e87070. [CrossRef]
21. Brancati, R.; Tufano, F. Indirect Estimation of Tire Pressure on Several Road Pavements via Interacting Multiple Model Approach. *Machines* **2022**, *10*, 1221. [CrossRef]
22. Rittger, L.; Engelhardt, D.; Schwartz, R. Adaptive user experience in the car—Levels of adaptivity and adaptive HMI design. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 4866–4876. [CrossRef]
23. Wandtner, B.; Schömig, N.; Schmidt, G. Secondary task engagement and disengagement in the context of highly automated driving. *Transp. Res. Part Traffic Psychol. Behav.* **2018**, *58*, 253–263. [CrossRef]

24. Verwey, W.B. On-line driver workload estimation. Effects of road situation and age on secondary task measures. *Ergonomics* **2000**, *43*, 187–209. [CrossRef]
25. Bischoff, D. Developing guidelines for managing driver workload and distraction associated with telematic devices. *SAE Paper* **2007**. https://www-esv.nhtsa.dot.gov/Proceedings/20/print6.pdf (accessed on 21 September 2023).
26. DeGuzman, C.A.; Kanaan, D.; Donmez, B. Attentive user interfaces: Adaptive interfaces that monitor and manage driver attention. In *User Experience Design in the Era of Automated Driving*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 305–334.
27. Galarza, M.; Paradells, J. Improving road safety and user experience by employing dynamic in-vehicle information systems. *IET Intell. Transp. Syst.* **2019**, *13*, 738–744. [CrossRef]
28. Birrell, S.; Young, M.; Stanton, N.; Jennings, P. Using adaptive interfaces to encourage smart driving and their effect on driver workload. In *Advances in Human Aspects of Transportation: Proceedings of the AHFE 2016 International Conference on Human Factors in Transportation, Walt Disney World®, Orlando, FL, USA, 27–31 July 2016*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 31–43.
29. Wintersberger, P.; Schartmüller, C.; Riener, A. Attentive user interfaces to improve multitasking and take-over performance in automated driving: The auto-net of things. *Int. J. Mob. Hum. Comput. Interact. (IJMHCI)* **2019**, *11*, 40–58. [CrossRef]
30. Oron-Gilad, T.; Ronen, A.; Shinar, D. Alertness maintaining tasks (AMTs) while driving. *Accid. Anal. Prev.* **2008**, *40*, 851–860. [CrossRef]
31. Jamson, A.H.; Lai, F.C.; Carsten, O.M. Potential benefits of an adaptive forward collision warning system. *Transp. Res. Part Emerg. Technol.* **2008**, *16*, 471–484. [CrossRef]
32. Gaspar, J.; Schwarz, C.; Kashef, O.; Schmitt, R.; Shull, E. Using Driver State Detection in Automated Vehicles. 2018; https://rosap.ntl.bts.gov/view/dot/42273 (accessed on 21 September 2023).
33. Reinmueller, K.; Koehler, L.; Steinhauser, M. Adaptive warning signals adjusted to driver passenger conversation: Impact of system awareness on behavioral adaptations. *Transp. Res. Part Traffic Psychol. Behav.* **2018**, *58*, 242–252. [CrossRef]
34. Heigemeyr, A.; Harrer, A. An integrated method for Adaptive automotive Human Machine Interfaces. In Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013), Hague, The Netherlands, 6–9 October 2013; pp. 558–564.
35. Endres, C. PRESTK: Situation-Aware Presentation of Messages and Infotainment Content for Drivers. 2012. Available online: http://dx.doi.org/10.22028/D291-25216 (accessed on 21 September 2023). [CrossRef]
36. Heigemeyr, A.; Harrer, A. Information management for adaptive automotive human machine interfaces. In Proceedings of the 6th International Conference on Automotive User Interfaces and Interactive Vehicular Applications, Seattle, WA, USA, 17–19 September 2014; pp. 1–8.
37. Amditis, A.; Andreone, L.; Pagle, K.; Markkula, G.; Deregibus, E.; Rue, M.R.; Bellotti, F.; Engelsberg, A.; Brouwer, R.; Peters, B.; et al. Towards the automotive HMI of the future: Overview of the AIDE-integrated project results. *IEEE Trans. Intell. Transp. Syst.* **2010**, *11*, 567–578. [CrossRef]
38. Peter, C.; Beale, R. *Affect and Emotion in Human-Computer Interaction: From Theory to Applications*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2008; Volume 4868.
39. Khan, I.; Khusro, S. Towards the design of context-aware adaptive user interfaces to minimize drivers' distractions. *Mob. Inf. Syst.* **2020**, *2020*, 8858886. [CrossRef]
40. Manawadu, U.E.; Kamezaki, M.; Ishikawa, M.; Kawano, T.; Sugano, S. A multimodal human-machine interface enabling situation-Adaptive control inputs for highly automated vehicles. In Proceedings of the 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, USA, 11–14 June 2017; pp. 1195–1200.
41. Amditis, A.; Polychronopoulos, A.; Andreone, L.; Bekiaris, E. Communication and interaction strategies in automotive adaptive interfaces. *Cogn. Technol. Work.* **2006**, *8*, 193–199. [CrossRef]
42. Prentzas, J.; Hatzilygeroudis, I. Categorizing approaches combining rule-based and case-based reasoning. *Expert Syst.* **2007**, *24*, 97–122. [CrossRef]
43. Deregibus, E.; Andreone, L.; Bianco, E.; Amditis, A.; Polychronopoulos, A.; Kussman, H. *The AIDE Adaptive and Integrated HMI Design: The Concept of the Interaction Communication Assistant*; ITS: London, UK, 2006.
44. Maino, C.; Misul, D.; Musa, A.; Spessa, E. Optimal mesh discretization of the dynamic programming for hybrid electric vehicles. *Appl. Energy* **2021**, *292*, 116920. [CrossRef]
45. Miretti, F.; Misul, D.; Spessa, E. DynaProg: Deterministic Dynamic Programming solver for finite horizon multi-stage decision problems. *SoftwareX* **2021**, *14*, 100690. [CrossRef]
46. De Santis, D. A framework for optimizing co-adaptation in body-machine interfaces. *Front. Neurorobotics* **2021**, *15*, 662181. [CrossRef] [PubMed]
47. Amditis, A.; Kubmann, H.; Polychronopoulos, A.; Engstrom, J.; Andreone, L. System architecture for integrated adaptive HMI solutions. In Proceedings of the 2006 IEEE Intelligent Vehicles Symposium, Meguro-Ku, Japan, 13–15 June 2006; pp. 388–393.
48. López-González, M. Today is to see and know: An argument and proposal for integrating human cognitive intelligence into autonomous vehicle perception. *Electron. Imaging* **2019**, *2019*, 54-1–54-9. [CrossRef]
49. Liang, Y.; Reyes, M.L.; Lee, J.D. Real-time detection of driver cognitive distraction using support vector machines. *IEEE Trans. Intell. Transp. Syst.* **2007**, *8*, 340–350. [CrossRef]
50. Muñoz, M.; Reimer, B.; Lee, J.; Mehler, B.; Fridman, L. Distinguishing patterns in drivers' visual attention allocation using Hidden Markov Models. *Transp. Res. Part Traffic Psychol. Behav.* **2016**, *43*, 90–103. [CrossRef]

51. Wang, X.; Liu, Y.; Wang, F.; Wang, J.; Liu, L.; Wang, J. Feature extraction and dynamic identification of drivers' emotions. *Transp. Res. Part Traffic Psychol. Behav.* **2019**, *62*, 175–191. [CrossRef]
52. Xing, Y.; Lv, C.; Zhang, Z.; Wang, H.; Na, X.; Cao, D.; Velenis, E.; Wang, F.Y. Identification and analysis of driver postures for in-vehicle driving activities and secondary tasks recognition. *IEEE Trans. Comput. Soc. Syst.* **2017**, *5*, 95–108. [CrossRef]
53. Sommer, D.; Golz, M. Evaluation of PERCLOS based current fatigue monitoring technologies. In Proceedings of the 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology, Buenos Aires, Argentina, 31 August–4 September 2010; pp. 4456–4459.
54. Li, K.; Gong, Y.; Ren, Z. A fatigue driving detection algorithm based on facial multi-feature fusion. *IEEE Access* **2020**, *8*, 101244–101259. [CrossRef]
55. Polychronopoulos, A.; Amditis, A.; Andreone, L. Stochastic reconstruction of the traffic scenario and applications for situation adaptive interfaces. In Proceedings of the 12th World Congress on Intelligent Transport SystemsITS AmericaITS JapanERTICO, San Francisco, CA, USA, 6–10 November 2005.
56. Polychronopoulos, A.; Amditis, A.; Andreone, L. Real time environmental and traffic supervision for adaptive interfaces in intelligent vehicles. *IFAC Proc. Vol.* **2005**, *38*, 115–120. [CrossRef]
57. Wierwille, W.W.; Wreggit, S.; Kirn, C.; Ellsworth, L.; Fairbanks, R. Research on Vehicle-Based Driver Status/Performance Monitoring; Development, Validation, and Refinement of Algorithms for Detection of Driver Drowsiness. Final Report; Technical Report; 1994. Available online: https://rosap.ntl.bts.gov/view/dot/2578 (accessed on 21 September 2023).
58. Dinges, D.F.; Grace, R. *PERCLOS: A Valid Psychophysiological Measure of Alertness as Assessed by Psychomotor Vigilance*; Publication Number FHWA-MCRT-98-006; US Department of Transportation, Federal Highway Administration: Washington, DC, USA, 1998.
59. Trutschel, U.; Sirois, B.; Sommer, D.; Golz, M.; Edwards, D. PERCLOS: An alertness measure of the past. In Proceedings of the Driving Assesment Conference, Santa Fe, NM, USA, 24–27 June 2011; University of Iowa: Iowa, IA, USA, 2011; Volume 6.
60. Amditis, A.; Andreone, L.; Polychronopoulos, A.; Engström, J. Design and development of an adaptive integrated driver-vehicle interface: Overview of the AIDE project. *IFAC Proc. Vol.* **2005**, *38*, 103–108. [CrossRef]
61. Lohani, M.; Payne, B.R.; Strayer, D.L. A review of psychophysiological measures to assess cognitive states in real-world driving. *Front. Hum. Neurosci.* **2019**, *13*, 57. [CrossRef]
62. Paxion, J.; Galy, E.; Berthelon, C. Mental workload and driving. *Front. Psychol.* **2014**, *5*, 1344. [CrossRef] [PubMed]
63. Annett, J. Subjective rating scales: Science or art? *Ergonomics* **2002**, *45*, 966–987. [CrossRef] [PubMed]
64. Tomporowski, P.D. Performance and perceptions of workload among young and older adults: Effects of practice during cognitively demanding tasks. *Educ. Gerontol.* **2003**, *29*, 447–466. [CrossRef]
65. Johnson, A.; Proctor, R.W. *Attention: Theory and Practice*; Sage: Newcastle-upon-Tyne, UK, 2004.
66. Tokunaga, R.A.; Hagiwara, T.; Kagaya, S.; Onodera, Y. Cellular telephone conversation while driving: Effects on driver reaction time and subjective mental workload. *Transp. Res. Rec.* **2000**, *1724*, 1–6. [CrossRef]
67. Patten, C.J.; Kircher, A.; Östlund, J.; Nilsson, L. Using mobile telephones: Cognitive workload and attention resource allocation. *Accid. Anal. Prev.* **2004**, *36*, 341–350. [CrossRef] [PubMed]
68. Hart, S.G.; Staveland, L.E. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In *Advances in Psychology*; Elsevier: Amsterdam, The Netherlands, 1988; Volume 52, pp. 139–183.
69. Pauzié, A. A method to assess the driver mental workload: The driving activity load index (DALI). *IET Intell. Transp. Syst.* **2008**, *2*, 315–322. [CrossRef]
70. Gimelli, A.; Sannino, R. A micro gas turbine one-dimensional model: Approach description, calibration with a vector optimization methodology and validation. *Appl. Therm. Eng.* **2021**, *188*, 116644. [CrossRef]
71. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [CrossRef]
72. Muccillo, M.; Gimelli, A.; Sannino, R. Multi-objective optimization and sensitivity analysis of a cogeneration system for a hospital facility. *Energy Procedia* **2015**, *81*, 585–596. [CrossRef]
73. Gimelli, A.; Mottola, F.; Muccillo, M.; Proto, D.; Amoresano, A.; Andreotti, A.; Langella, G. Optimal configuration of modular cogeneration plants integrated by a battery energy storage system providing peak shaving service. *Appl. Energy* **2019**, *242*, 974–993. [CrossRef]
74. Deb, K.; Sindhya, K.; Okabe, T. Self-adaptive simulated binary crossover for real-parameter optimization. In Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation, London, UK, 7–1 July 2007; pp. 1187–1194.
75. Kineton. Innovarion Lab, KineCar. 2023. Available online: https://www.kineton.it/innovation-lab/ (accessed on 5 September 2023).
76. Kutila, M.; Jokela, M.; Markkula, G.; Rué, M.R. Driver distraction detection with a camera vision system. In Proceedings of the 2007 IEEE International Conference on Image Processing, San Antonio, TX, USA, 16–19 September 2007; Volume 6, pp. VI-201–VI-204.
77. Helene, T.V.; Thierry, B.; Serge, B.; Matti, K.; Jouko, V.; Evangelos, B.; Maria, P.; Johan, E.; Anders, A. Development of a driver situation assessment module in the AIDE project. *IFAC Proc. Vol.* **2005**, *38*, 97–102. [CrossRef]
78. Wu, C.; Liu, Y. Queuing network modeling of driver workload and performance. *IEEE Trans. Intell. Transp. Syst.* **2007**, *8*, 528–537.

79. Fonsalas, F. Holistic HMI Architecture for Adaptive and Predictive Car Interiors. In *Electronic Components and Systems for Automotive Applications, Proceedings of the 5th CESA Automotive Electronics Congress, Paris, France, 5–6 December 2018*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 217–227.
80. Hu, Z.; Lv, C.; Hang, P.; Huang, C.; Xing, Y. Data-driven estimation of driver attention using calibration-free eye gaze and scene features. *IEEE Trans. Ind. Electron.* **2021**, *69*, 1800–1808. [CrossRef]