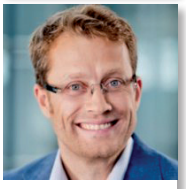
L'industrie automobile a besoin de normes pratiques pour la conduite autonome

Les méthodologies établies comme la norme de sûreté de fonctionnement ISO 26262 et les extensions comme SOTIF fournissent des exigences de haut niveau pour la sûreté des systèmes embarqués dans les véhicules plus ou moins autonomes, mais elles ne fournissent aucune instruction sur la manière de les atteindre. Pour Elektrobit, il est nécessaire d'entamer des discussions sur les bonnes pratiques en termes de configuration architecturale et d'algorithmes, qui puissent permettre de définir l'état de l'art des systèmes de conduite autonomes sûrs.

AUTEUR



Björn Giesler, responsable des technologies de conduite autonome, Elektrobit.

Les fonctionnalités de conduite hautement automatisée de niveau 3 et 4 nécessitent la présence d'une personne dans le véhicule, tant sur le plan technique que légal, même si elles autorisent le conducteur à détourner son attention de la conduite. Cela signifie qu'elles doivent au moins être capables de reconnaître si telle ou telle situation peut être gérée par le système ou s'il est nécessaire que le conducteur intervienne. Au niveau 4, les fonctionnalités doivent être préparées à l'éventualité que le conducteur ne puisse pas prendre le relais et doivent pouvoir s'adapter à toutes les situations de manière sûre. Au niveau 5, le véhicule doit être capable de fonctionner sans conducteur; cependant, déjà dans les niveaux inférieurs, le véhicule autonome, lorsqu'il est sous contrôle automatisé, peut être considéré comme un véhicule fonctionnant sans conducteur.

La sûreté de fonctionnement, comme elle est proposée par la norme ISO 26262, repose sur l'hypothèse que n'importe quel danger peut être décrit et classé en catégories d'exposition (quelle est la probabilité pour qu'un danger se produise?), la sévérité (dans quelle mesure le danger peut-il entraîner des blessures voire un décès?) et la contrôlabilité (dans quelle mesure peut-on s'attendre à ce que le conducteur humain soit capable de réagir au danger et de l'éviter?). Cette dernière catégorie s'avère problématique pour les systèmes automatisés. Etant donné que le conducteur, bien que présent physiquement, ne fait pas forcément attention à ce qui se passe, on ne peut absolument pas s'attendre à ce qu'il soit en mesure de gérer la situation. C'est pour cela que la question de savoir si la norme ISO 26262 est applicable aux systèmes sans conducteur n'est pas dénuée d'intérêt. Au minimum, sous le contrôle automatisé, tous les dangers doivent être considérés de catégorie C3 ou fondamentalement incontrôlables.

Cette évaluation entraîne des exigences de sûreté de fonctionnement élevées pour les logiciels et le matériel mis en œuvre pour la conduite automatisée; pour le chemin critique de sûreté, l'ASIL D est le niveau le plus courant. Pour simplifier, cela signifie qu'aucun composant du principal chemin critique d'exécution ne peut avoir une occurrence d'erreur supérieure à 10^{-8} par heure, ou ne peut pas échouer statistiquement plus d'une fois toutes les 11 704 années.

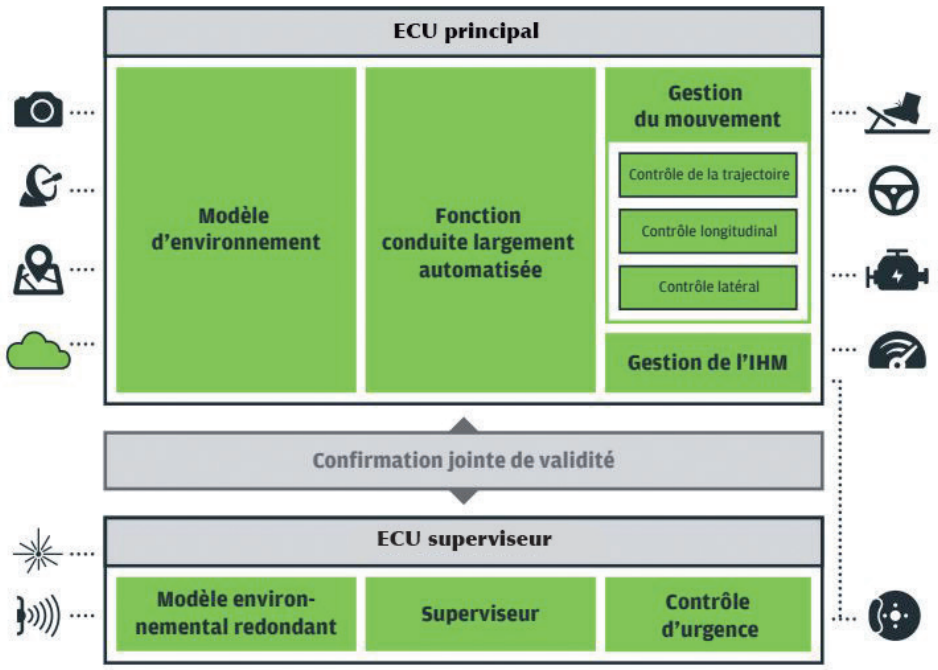
Le danger? Les algorithmes

La norme ISO 26262 fournit des instructions sur la façon d'atteindre cet objectif pour les logiciels ainsi que pour les dispositifs électroniques et électriques de qualité. Cependant, les véhicules largement automatisés basent leurs décisions sur les capteurs et les algorithmes. L'expérience montre que la plupart des dangers ne sont pas causés par des problèmes électroniques ou électriques, ni même par des bogues logiciels (bien que ceux-ci restent bien plus fréquents que ceux dus au matériel électronique), mais par des algorithmes ne parvenant pas à interpréter correctement les données des capteurs. Pour des systèmes autonomes de niveau 3 et plus, il n'est pas acceptable que l'automatisation commette une erreur qui n'est pas due à une erreur logicielle, électrique ou électronique alors que tout fonctionne comme lors du test, et que ce soit l'algorithme qui commette des erreurs fatales.

Les développements récents ne se sont ainsi pas concentrés seulement sur la sûreté de fonctionnement, mais aussi sur la sûreté de la fonctionnalité prévue (SOTIF, Safety of the Intended Functionality) qui peut être considérée comme une extension de la norme ISO 26262 prenant en compte les dangers dus à la perception et à l'algorithme via une vue encore plus aérienne du système. Nous pensons que cette approche est valable pour une évaluation globale d'un système automatisé, mais qu'elle n'aborde pas la façon dont on pourrait atteindre la sûreté nécessaire. Elle offre une liberté maximale dans l'implémentation mais ne permet pas d'atteindre l'état de l'art dans la mise en œuvre pratique de systèmes. Nous pensons qu'il serait judicieux de définir, non seulement la façon dont nous devrions fixer les

1 UN MOYEN THÉORIQUE DE RENDRE UN SYSTÈME SÛR

Une approche répandue à un niveau macrosystémique consiste à élaborer une « fonction principale » et une « fonction de supervision », chacune d'entre elles étant alimentée par des modèles environnementaux mutuellement redondants et capables de se confirmer mutuellement la validité de leur évaluation de la situation actuelle. Cette méthode permet la répartition des exigences de sûreté de fonctionnement entre deux ECU.



objectifs pour la sûreté de la conduite automatisée mais aussi sur la façon de les atteindre en pratique.

Redondance algorithmique, ou comment construire un système sûr

Il y a plusieurs façons de rendre un système sûr, du moins en théorie. Une approche répandue à un niveau macrosystémique consiste à élaborer une « fonction principale » et une « fonction de supervision », chacune d'entre elles étant alimentée par des modèles environnementaux mutuellement redondants et capables de se confirmer mutuellement la validité de leur évaluation de la situation actuelle. Cette méthode permet la répartition des exigences de sûreté de fonctionnement entre deux ECU (ou deux TCU sur une ECU) et deux équipes de conception de logiciel (figure 1).

Modéliser l'environnement

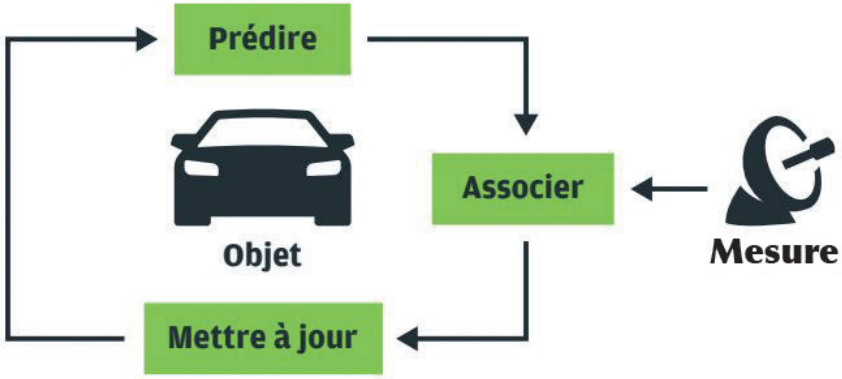
Ce type de redondance présente un gros potentiel à un niveau bien plus approfondi et détaillé. Prenons comme exemple le temps à la collision (TTC) qu'un système de conduite automatisée doit calculer. L'information la plus importante pour n'importe quelle fonction de contrôle d'un véhicule relative à la sûreté sera toujours « vous allez entrer en collision dans x secondes ». Dans de nombreuses configurations actuelles, le TTC est calculé à l'aide d'un modèle environnemental connu sous le nom de fusion d'objets.

L'idée derrière la fusion des objets est que la plupart des contributeurs potentiels de collision dans l'environnement de la voiture sont soit statiques, soit dynamiques et que leur mouvement (s'il y en a un) peut être modélisé avec une fonction mathématique donnée. A titre d'exemple, un autre véhicule, doté d'un essieu directeur et d'un essieu mécanisé, va se déplacer en fonction de l'angle du premier et de la vitesse du second. Ainsi, on peut s'attendre à ce que le véhicule ne fasse pas subitement un écart ou s'arrête de façon instantanée, mais plutôt à ce qu'il se déplace selon un modèle de mouvement spécifique. Cette propriété peut être utilisée pour prédire la trajectoire de l'objet et ainsi éviter des erreurs de calcul. Le cycle habituel d'un composant de fusion d'objets se décompose ainsi en trois phases répétitives : prédire la position d'un objet, associer l'objet prédit à la nouvelle mesure du capteur, mettre à jour le modèle du mouvement de l'objet pour qu'il corresponde aux nouvelles données du capteur (figure 2).

Pour ce type de modélisation, on utilise traditionnellement un filtre de Kalman ou une de ses variantes, une méthode datant de 1960 utilisée dans la recherche en robotique depuis les années 1980 pour la modélisation de la trajectoire des objets. Cette méthode, considérée comme l'état de l'art, est utilisée dans presque tous les capteurs environnementaux automobiles. Si les hypothèses concernant le modèle de trajectoire choisi sont correctes et que toutes les sources d'erreurs dans le système sont modélisées correctement, le résultat est sans aucun doute la meilleure façon possible de décrire l'objet. Si cet algorithme est développé correctement, le système est alors aussi sûr que possible, n'est-

2 MODÈLE ENVIRONNEMENTAL CONNU SOUS LE NOM DE FUSION D'OBJETS

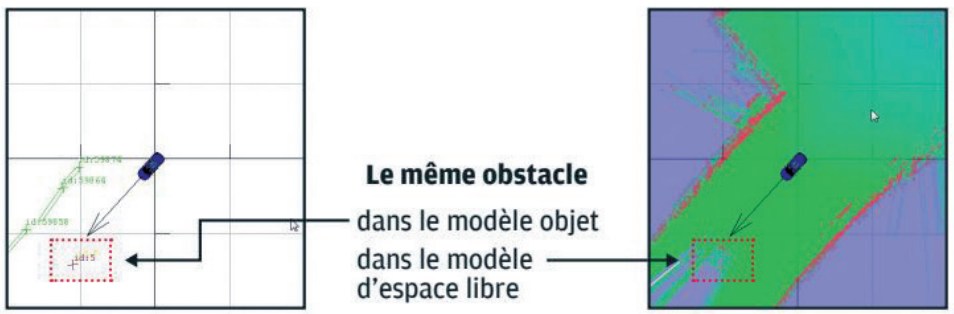
Le cycle habituel d'un composant de fusion d'objets se décompose en trois phases répétitives : prédire la position d'un objet, associer l'objet prédit à la nouvelle mesure du capteur, mettre à jour le modèle du mouvement de l'objet pour qu'il corresponde aux nouvelles données du capteur.



ce-pas ? Pas forcément en fait.
La modélisation de l'objet repose sur différentes hypothèses : le modèle du mouvement doit être le bon. Par exemple, il ne faut pas confondre la voiture que nous modélisons avec un autre type d'objet ; ainsi, si c'était un piéton, il pourrait faire brusquement un écart. Étant donné que cela n'aurait pas été modélisé, notre algorithme serait fortement surpris, il considérerait cela comme une erreur et ne le signalerait pas comme telle à notre fonction. L'association doit également être correcte. Si nous associons de façon erronée la mesure d'un lampadaire situé à côté de la route à celle de la voiture modélisée, cela produirait un mouvement brusque chez la voiture qui n'existe pas dans la réalité. Toutes les erreurs doivent être correctement modélisées. Et ainsi de suite.
Cela ne signifie pas que la modélisation d'objet n'est pas adaptée dans cette situation. Cependant, étant basée sur des hypothèses qui ne sont pas toujours correctes, elle peut se révéler fausse de temps en temps. Aucune de ces hypothèses n'est en fait garantie par une quelconque norme de sûreté de fonctionnement. La conclusion logique à ce problème consiste à la renforcer avec un algorithme redondant, à l'état de l'art, et qui fournirait la même information d'une façon différente.
Une autre façon de modéliser l'environnement de la voiture consiste à imaginer une grille de cellules d'une certaine taille, distinctes les unes des autres, chacune d'entre elles pouvant être occupée, vide ou inconnue si elle n'a pas encore été aperçue par les capteurs du véhicule. Lorsqu'un capteur aperçoit un obstacle à une distance et à un angle donnés du véhicule, la grille de cellules est marquée « occupée » à l'emplacement correspondant. Pour certains types de capteurs, les cellules de la grille entre le capteur et l'obstacle observé sont marquées comme vides.
Cette façon de modéliser l'environnement est apparue en 1985. Elle est également considérée comme éprouvée et testée dans le domaine des capteurs environnementaux automobiles, même si elle n'a peut-être pas été aussi utilisée que le filtre de Kalman. Elle peut également fournir des informations sur le temps à la collision (TTC) car elle modélise aussi les obstacles sur le chemin du véhicule. Elle se base aussi sur des hypothèses, comme par exemple le fait qu'une zone est vide si les rayons du capteur peuvent passer à travers (ce qui, en fonction du capteur et de l'objet, n'est pas toujours le cas) et sur le dynamisme de l'environnement, c'est à dire, combien de mesures doivent être réalisées pour une seule cellule avant qu'elle soit reconnue comme vide ou occupée. Cependant, ces hypothèses sont fondamentalement différentes de celles utilisées dans la modélisation d'objet.

3 MODÉLISER L'ENVIRONNEMENT PAR PLUSIEURS MÉTHODES À DES FINS DE REDONDANCE

Une façon d'éviter les erreurs de jugement consiste à utiliser simultanément plusieurs méthodes pour modéliser l'environnement et à n'autoriser une action que si toutes ces méthodes s'accordent sur sa sûreté. Dans le schéma ci-dessous, le véhicule ne pourrait continuer à rouler que si le modèle de l'« objet » et le modèle de l'« espace libre » de l'environnement s'accordent sur le fait qu'il n'y a rien sur le chemin du véhicule.



La redondance algorithmique crée des systèmes plus sûrs

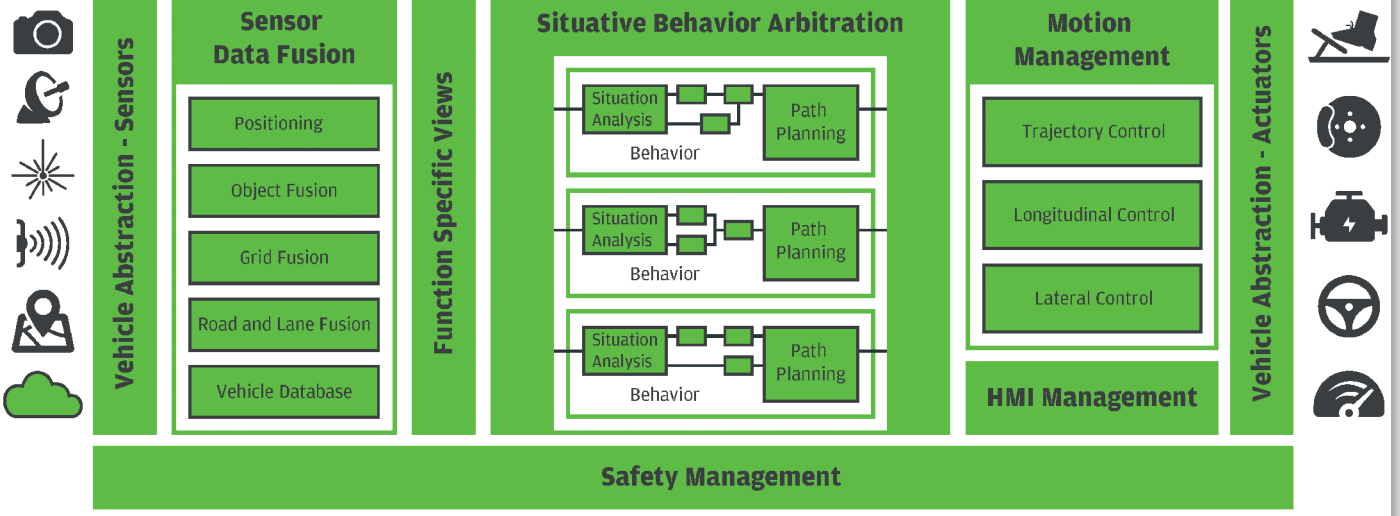
Nous venons d'examiner, dans un même exemple, deux types de mécanismes largement utilisés pour modéliser l'environnement. Tous deux ont largement été testés sur le terrain et peuvent être développés à des niveaux élevés de sûreté de fonctionnement (pour des coûts de développement arbitrairement élevés). Aucun des deux ne fera jamais d'erreur dans tous les cas. Quand la sûreté de fonctionnement et la sûreté SOTIF exigent un taux d'erreur inférieur ou égal à un pour 10^8 heures de fonctionnement, on peut facilement supposer que pendant ce temps de fonctionnement, chacun des algorithmes entraînera des erreurs, car ni la sûreté de fonctionnement ni la sûreté SOTIF ne donnent d'instructions pour les éviter. Un système autonome dans sa globalité regorge d'autres exemples de ce type. Cela s'avère très problématique pour un système automatisé s'il fait des erreurs de jugement car il peut mettre des personnes en danger.
Une façon d'éviter ces erreurs de jugement consiste à utiliser simultanément plusieurs méthodes pour modéliser l'environnement et à n'autoriser une action que si toutes ces méthodes s'accordent sur sa sûreté. Dans l'exemple ci-dessus, notre véhicule ne pourrait continuer à rouler que si le modèle de l'« objet » et le modèle de l'« espace libre » de l'environnement s'accordaient sur le fait qu'il n'y a rien sur le chemin du véhicule (figure 3). Etant donné que les hypothèses sous-jacentes à ces deux méthodes sont fondamentalement différentes, il est probable qu'elles ne sont pas fausses toutes les deux en même temps.

La nécessité d'une vérification par des pairs

On ne doit pas avoir à « espérer » que les véhicules autonomes soient sûrs, ils doivent être « reconnus » comme tels. La sûreté SOTIF ne précise pas les règles de décomposition pour les hypothèses algorithmiques, elle précise en revanche qu'elles ne peuvent pas être fausses plus d'une fois toutes les 10^8 heures pour un système ASIL D. C'est également vrai pour d'autres types de systèmes (les mécanismes de frein par exemple). Cependant, dans notre cas, le problème est bien moins compris et ne peut pas être testé en laboratoire. C'est pourquoi la preuve nécessaire ne peut actuellement être atteinte que par des tests sur le terrain, ce qui nécessiterait de

4 LA SOLUTION DE CONDUITE AUTONOME OPEN ROBINOS PROMUE PAR ELEKTROBIT

En juin 2016, Elektrobit a publié une description de sa solution sous le nom d'open robinos, solution conçue pour pouvoir être exécutée sur les PC standard et sur certaines ECU automobiles. Les spécifications (c'est-à-dire la description du système à faire réviser par des pairs) sont disponibles sous une licence Common Creative et l'accès est gratuit.



nombreux tests statistiques, dont chacun durerait 10^8 heures. Ne serait-ce que pour un seul test de conduite avec une voiture, les prix sont déjà prohibitifs, sans même évoquer le nombre d'échantillons nécessaires pour parvenir à une preuve statistique. Ainsi, d'un point de vue scientifique, le test de conduite avance à tâtons. Il est vrai que les essais de conduite assistée standard se sont très bien passés, mais étant donné le haut degré d'exigences de la norme ISO sur la sûreté du système, on ne peut pas, à l'heure actuelle, garantir que les tests soient suffisants. Cela signifie qu'il n'y a actuellement aucun moyen technique de valider la sûreté de toutes solutions algorithmiques pour les véhicules automatisés.

Il existe cependant des solutions pour contourner ce problème. En science, on fait appel à la révision par des pairs : une méthode donnée, dont on pense qu'elle pourrait résoudre un problème, est publiée et décrite en détail. Elle est examinée par d'autres, ses lacunes étant mises en évidence par des examens théoriques et des tests pratiques. Des améliorations sont apportées, publiées et testées à nouveau. C'est le cycle typique pour n'importe quelle solution, au terme duquel de nombreuses solutions, légèrement différentes les unes des autres, seront trouvées puis comparées. A la fin, la meilleure solution émerge et perdure pendant un certain temps, jusqu'à ce que de nouvelles informations ou avancées dans d'autres domaines produisent de nouvelles idées. Alors, le cycle recommence. Les deux algorithmes décrits ci-dessus ont passé ce cycle et sont considérés comme les meilleures solutions à leurs problèmes spécifiques. La vérification par des pairs ne garantit pas une solution sûre à 100 %, mais cela assure qu'à un instant donné, personne n'a trouvé de meilleure solution.

Intelligence artificielle : une solution à très court terme

Dans le contexte de la course à la première place sur le marché, la recherche sur la conduite largement automatisée semble néanmoins prendre la direction opposée à ce que nous venons de décrire. Il est vrai que des solutions de pointe sont apportées à des problèmes spécifiques. Cependant la combinaison de ces solutions pour bâtir des systèmes complets est considérée comme la propriété intellectuelle de l'entreprise et n'est pas divulguée. Le marketing public sur les dernières avancées dans la recherche sur les réseaux de neurones semble indiquer que même l'utilisation d'algorithmes de pointe n'est plus à la mode et qu'à la place, ce sera l'intelligence artificielle qui résoudra le problème par une magie encore inconnue.

C'est une solution à très court terme. Il est probable qu'au bout d'un certain temps, une personne sera blessée dans un accident impliquant un système largement automatisé, que cela soit dû au système ou non. Dans ce cas, une procédure de justice devra déterminer si le système a été conçu dans le respect de l'état de l'art. Si à ce moment-là aucun état de l'art n'a encore été établi, on peut imaginer que de nombreuses discussions houleuses auront lieu. En fonction du jugement du tribunal et des procès qui pourraient s'ensuivre, ce ne sont pas seulement les entreprises concernées qui pourraient en souffrir, mais l'industrie de la voiture autonome dans son ensemble, dont on estime qu'elle devrait atteindre un marché de 42 milliards de dollars d'ici à 2025.

La nécessité d'avoir des normes

C'est la raison pour laquelle nous voudrions proposer d'entamer une discussion dans l'industrie sur les meilleures pratiques à mettre en œuvre pour le développement des systèmes de conduite largement automatisés, et ce, à un niveau technique approfondi. Il n'est pas nécessaire que les constructeurs dévoilent leur propres solutions dans le détail (bien que cela sera souhaitable à

long terme), il suffirait de publier une solution qui pourrait être revue par des pairs et améliorée à travers le temps. Selon nous, cela ne menace en aucune façon la propriété intellectuelle des constructeurs.

En juin 2016, Elektrobit a publié une description de sa solution sous le nom d'open robinos (https://www.elektrobit.com/products/eb-robinos/eb-robinos-specification/), solution conçue pour pouvoir être exécutée sur les PC standard et sur certaines ECU automobiles. Les spécifications (c'est-à-dire la description du système à faire réviser par des pairs) sont disponibles sous une licence Common Creative et l'accès est gratuit. Nous mettons actuellement en place un consortium autour de ces spécifications ; celui-ci entamera le processus de révision de ces spécifications et publiera des améliorations à leur apporter. Notre objectif consiste à créer une implémentation de référence par rapport à laquelle les autres systèmes pourraient être comparés et testés, et finalement, de construire un standard ouvert pour le développement de systèmes autonomes (figure 4).

Une norme a d'autres implications au-delà de simples algorithmes relatifs à la sûreté et révisés par des pairs. Elle permet aux concepteurs de véhicules, de logiciels et de matériels électroniques d'utiliser certaines parties de la norme et d'en remplacer d'autres par leurs propres solutions. Elle permet aussi de développer leur propre IP ou bien de développer les caractéristiques propres à leur marque sans avoir à développer le système complet. Cela crée un marché pour les modules système, de sorte que la compétition peut se faire sur la base du module et non sur la base du système. Une norme permet par ailleurs de concevoir de meilleures solutions technologiques et de répartir les coûts de développement et des tests sur toute l'industrie plutôt que sur un seul modèle de voiture. Une norme peut faire l'objet d'une discussion avec le législateur, être utilisée pour créer des procédures de test pour des agences comme NHTSA ou TÜV et garantir au consommateur qu'un produit y répond et est donc considéré sûr d'utilisation.

Conclusion

Le but de ce document n'est pas de publier une nouvelle solution technique. Nous proposons une très vielle solution pour réussir à trouver les bonnes avancées technologiques. Celle-ci n'est actuellement pas employée dans la course aux voitures autonomes en-dehors de la recherche universitaire. Nous pensons que, actuellement, les problèmes liés à la conduite largement automatisée ne sont pas assez bien compris pour qu'une seule entreprise ou conglomérat puisse les résoudre seul, et que même s'ils pouvaient l'être, cela ne résoudrait pas le problème de la conception de voitures autonomes sûres dans leur ensemble. Les principes de la conduite automatisée doivent, au moins, être vérifiés, testés, approuvés et finalement standardisés par l'industrie.

Nous estimons que cette proposition ne nuit en aucun cas aux affaires de toute entreprise ni n'entrave la vitesse de développement des voitures autonomes. Nous pensons au contraire que cela permettra une meilleure utilisation des ressources financières et humaines, accélérera le processus de développement et contribuera à l'élaboration de systèmes plus sûrs.

Si vous souhaitez rejoindre le processus de spécifications et d'évaluation, ainsi qu'échanger sur les algorithmes et les architectures des voitures autonomes sécurisées, merci de vous inscrire sur le site http://www.open-robinos.com ■