

D0001/99A

ECOLE NATIONALE POLYTECHNIQUE

DEPARTEMENT D'ELECTRONIQUE

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

THESE DE DOCTORAT D'ETAT
EN ELECTRONIQUE

Présentée Par: **Mr. A. Chohra**

Magister en Cybernétique option: Robotique

Thème

PLANIFICATION ET CONTROLE DE LA NAVIGATION
DES VEHICULES AUTONOMES INTELLIGENTS (VAI)
EN ENVIRONNEMENTS DYNAMIQUES

Soutenue Le 16 Mars 1999.

Membres du Jury:

Président du Jury: **Mr. R. Toumi**, Professeur,

Rapporteur: **Mr. A. Farah**, Professeur,

Co-Rapporteur: **Mr. C. Benmehrez**, Maître de Recherches,

Examineur: **Mr. N. Berrached**, Maître de Conférences,

Examineur: **Mr. H. Bessalah**, Maître de Recherches,

Examineur: **Mr. F. Boudjemaa**, Maître de Conférences,

Invité: **Mr. N. Ghouali**, Professeur.

DEDICACES

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

Je dédie cette thèse à mes parents et à toutes les personnes qui me sont chères.

REMERCIEMENTS

Une thèse se réalise rarement 'toute seule', il faut souvent l'aider ! et s'aider de diverses contributions d'amis et collègues, aussi leur dois-je au minimum de sincères remerciements.

Je tiens à remercier particulièrement:

Melle. O. Azouaoui,

Mr. A. Bouzit,

Mr. M. Djafour,

ainsi que:

Mr. A. Farah, Rapporteur et Mr. C. Benmehrez, Co-Rapporteur,

Mr. H. Bessalah, Directeur du CDTA,

Messieurs: le Président et les Membres du Jury.

AVANT PROPOS



Les travaux de recherches présentés dans cette thèse, dirigée par le Professeur A. Farah et co-dirigée par Mr. C. Benmehrez, Maître de Recherches, ont été menés au *Laboratoire de Robotique et d'Intelligence Artificielle (LRIA)* du *Centre de Développement des Technologies Avancées (CDTA)*, en collaboration avec le *Laboratoire Techniques Digitales et Systèmes (LTDS)* de l'*Ecole Nationale Polytechnique (ENP)*.

ملخص

الهدف الأساسي من هذا البحث يتمثل في تطوير تقنيات لتخطيط و مراقبة حركة العربات المستقلة و الذكية في بيئات متغيرة. لقد تأكدت حديثا ضرورة إستعمال الشبكات العصبية، المنطق الغامض، الألوغورثمات الجينية، "Adaptive Resonance"، والأنظمة الخبيرة لتوفير مزيدا من الإستقلالية و الذكاء مع قدرات كبيرة لتحليل المعلومات لهاته العربات. بالفعل، إستعمال مثل هذه التقنيات ضروري لتطوير سلوكات العربات بإتجاه سلوكات الإنسان خاصة في حصر الهدف و إجتنااب الحواجز، في أخذ القرارات، وفي الفعل بالتعلم و التأقلم. لهذا الغرض، إقترحنا إستراتيجيتين للتحرك كلتا هما متكونتان من السلوكات الذكية الآتية : سلوك حصر الهدف، سلوك إجتنااب الحواجز، سلوك أخذ القرارات، و سلوك الفعل. أما إكتساب تلك السلوكات الذكية بالتعلم و التأقلم فقد أنجز بأنظمة ذكية مختلفة (مستعملة التقنيات الحديثة على أفراد أو مع بعضها البعض) إقترحت لكل إستراتيجية على حدة. بالفعل، إهتمام خاص قد أعطي للأنظمة الذكية المستعملة للشبكات العصبية أو للمنطق الغامض على أفراد، وللأنظمة المستعملة للتقنيات الحديثة مع بعضها البعض:

- الشبكات العصبية و الألوغورثمات الجينية،
- الشبكات العصبية و المنطق الغامض،
- الشبكات العصبية، المنطق الغامض، و "Adaptive Resonance"،
- الأنظمة الخبيرة، الشبكات العصبية، و المنطق الغامض.

مكننا النتائج المماثلة لهاته الأنظمة من إختيار أنجع الأنظمة لكل سلوك على حدة و بالتالي من إقترح أنظمة ذكية شاملة (محتوية كل السلوكات) لتخطيط و مراقبة الإستراتيجيتين. لقد تبين أيضا من نتائج المماثلة لهاته الأنظمة الشاملة أن الإستراتيجيتين قادرتان على توفير للعربات المستقلة و الذكية قدرات التحرك الذكي بإتجاه الأهداف، مجتنبتين الحواجز، في بيئات متغيرة. كما أثبتت قدرات التعلم و التأقلم لهاته التقنيات الحديثة في التخطيط و المراقبة. أما من الناحية التطبيقية، لقد إقترحنا تطبيقا على منهجية "Field-Programmable Gate Array" لنظام ذكي معتمدا على شبكة عصبية لسلوك إجتنااب الحواجز.

الكلمات المرجعية : العربات المستقلة و الذكية، البيئات المتغيرة، التحرك، الإستقلالية، الذكاء، القدرات الكبيرة لتحليل المعلومات، التخطيط، المراقبة، السلوكات الذكية، حصر الهدف، إجتنااب الحواجز، أخذ القرارات، الفعل، الشبكات العصبية، المنطق الغامض، الألوغورثمات الجينية، "Adaptive Resonance"، الأنظمة الخبيرة، التعلم، التأقلم، الأنظمة الذكية، "Field-Programmable Gate Array".

Abstract

The main objective, in this thesis, is the development of *planning* and *control* approaches of the *navigation* for *Intelligent Autonomous Vehicles (IAV)* in dynamic environments. To ensure more *autonomy* and more *intelligence* with *real-time* processing capabilities to *IAV*, these approaches are based on the *Soft Computing (SC)* involving particularly the *Neural Networks (NN)*, *Fuzzy Logic (FL)*, *Genetic Algorithms (GA)*, *Adaptive Resonance (AR)*, and *Expert Systems (ES)*. Indeed, the use of *SC* is necessary to bring the *IAV* behavior near the human one in the recognition, learning, adaptation, reasoning and decision-making, and action. Two (02) navigation strategies are then suggested, *NAV_1* and *NAV_2*, having the same global architecture built of the intelligent behaviors of target localization (*TAR*), obstacle avoidance (*OBS*), decision-making (*DEC*), and action (*ACT*). The acquisition of these intelligent behaviors by *learning* or/and *adaptation* is carried out by *Intelligent Systems (IS)* and *Hybrid Intelligent Systems (HIS)* suggested for each navigation strategy. Indeed, a particular interest has been oriented to *IS* based either on *NN* or on *FL* and to *HIS* based on the combinations *NN-GA*, *NN-FL*, *NN-FL-AR*, *ES-NN-FL*. The simulation results of training and making use of these *IS* and *HIS* have allowed to determine the best systems, in terms of making use facility and performance, for each intelligent behavior. Consequently, global architectures are suggested as *HIS*, including all the intelligent behaviors, for the planning and control of the *NAV_1* and *NAV_2* strategies. Then, simulation results show that these strategies allow *IAV* to intelligently navigate towards their targets, without collisions, in dynamic environments. They illustrate also the *learning*, *adaptation*, and *generalization* capabilities of these *SC* based planning and control approaches. Concerning the implementation aspect, a *Field-Programmable Gate Array (FPGA)* architecture is suggested for a *NN* based *IS* achieving the intelligent behavior *OBS*.

Key Words: *Intelligent Autonomous Vehicles (IAV)*, *Dynamic Environments*, *Navigation*, *Autonomy*, *Intelligence*, *Real-Time*, *Planning*, *Control*, *Intelligent Behaviors*, *Target Localization (TAR)*, *Obstacle Avoidance (OBS)*, *Decision-Making (DEC)*, *Action (ACT)*, “*Soft Computing*” (*SC*), *Neural Networks (NN)*, *Fuzzy Logic (FL)*, *Genetic Algorithms (GA)*, *Adaptive Resonance (AR)*, *Expert Systems (ES)*, *Supervised Learnings*, *Adaptation*, *Intelligent Systems (IS)*, *Hybrid Intelligent Systems (HIS)*, and *Field-Programmable Gate Array (FPGA)*.

Résumé

L'objectif principal, dans cette thèse, est le développement d'approches de planification et de contrôle de la navigation des *Véhicules Autonomes Intelligents (VAI)* en environnements dynamiques. Afin qu'elles puissent assurer aux *VAI* plus d'*autonomie* et plus d'*intelligence* avec des capacités de traitement *temps réel*, ces approches sont basées sur la collection de méthodologies appelée "Soft Computing" (*SC*) impliquant particulièrement les Réseaux de Neurones (*RN*), la Logique Floue (*LF*), les Algorithmes Génétiques (*AG*), la Résonance Adaptative (*RA*), et les Systèmes Experts (*SE*). En effet, l'utilisation du *SC* est nécessaire afin de faire évoluer le comportement des *VAI* vers celui de l'être humain dans la reconnaissance, l'apprentissage, l'adaptation, le raisonnement et la prise de décisions, et l'action. Deux (02) stratégies de navigation sont alors proposées, *NAV_1* et *NAV_2*, dont l'architecture globale se compose des comportements intelligents de localisation de la cible (*CIB*), d'évitement d'obstacles (*OBS*), de prise de décisions (*DEC*), et d'action (*ACT*). L'acquisition de ces comportements intelligents par *apprentissage* ou/et par *adaptation* est prise en charge par des Systèmes Intelligents (*SI*) et des Systèmes Hybrides Intelligents (*SHI*) proposés pour chaque stratégie de navigation. En effet, un intérêt particulier a été porté aux *SI* basés soit sur les *RN* soit sur la *LF* ainsi qu'aux *SHI* basés sur les combinaisons *RN-AG*, *RN-LF*, *RN-LF-RA*, et *SE-RN-LF*. Les résultats de simulation des entraînements et des mises en oeuvre de ces *SI* et *SHI* ont permis de dégager les meilleurs systèmes, en termes de facilité de mise en oeuvre et de performance, pour chaque comportement intelligent. Par conséquent, des architectures globales sont proposées en *SHI*, englobant tous les comportements intelligents, pour la planification et le contrôle des stratégies *NAV_1* et *NAV_2*. Les résultats de simulation montrent que ces stratégies permettent alors aux *VAI* de naviguer intelligemment vers leurs cibles, sans collisions, dans des environnements dynamiques. Ils illustrent également les capacités d'*apprentissage*, d'*adaptation*, et de *généralisation* de ces approches de planification et de contrôle basées sur le *SC*. Concernant l'aspect implémentation, une architecture "Field-Programmable Gate Array" (*FPGA*) est proposée pour un *SI* basé sur un *RN* réalisant le comportement intelligent *OBS*.

Mots Clés: Véhicules Autonomes Intelligents (*VAI*), Environnements Dynamiques, Navigation, Autonomie, Intelligence, Temps Réel, Planification, Contrôle, Comportements Intelligents, Localisation de la Cible (*CIB*), Evitement d'Obstacles (*OBS*), Prise de Décisions (*DEC*), Action (*ACT*), "Soft Computing" (*SC*), Réseaux de Neurones (*RN*), Logique Floue (*LF*), Algorithmes Génétiques (*AG*), Résonance Adaptative (*RA*), Systèmes Experts (*SE*), Apprentissages Supervisés, Adaptation, Systèmes Intelligents (*SI*), Systèmes Hybrides Intelligents (*SHI*), et "Field-Programmable Gate Array" (*FPGA*).

TABLE DES MATIERES

LISTE DES ABREVIATIONS

LISTE DES FIGURES

LISTE DES TABLEAUX

INTRODUCTION GENERALE

	1
I. NAVIGATION DES VEHICULES AUTONOMES INTELLIGENTS (VAI) EN ENVIRONNEMENTS DYNAMIQUES	8
I. 1. INTRODUCTION	9
I. 1. 1. Véhicules Autonomes Intelligents (VAI)	10
I. 1. 2. Environnements Dynamiques	12
I. 1. 3. Navigation	12
I. 2. EVOLUTION DES VEHICULES AUTONOMES INTELLIGENTS (VAI)	13
I. 2. 1. Domaines d'Applications	13
I. 2. 1. 1. Applications Terrestres	13
I. 2. 1. 2. Applications Sous-Marines	15
I. 2. 1. 3. Applications Spatiales	16
I. 2. 2. Architecture, Mobilité, Perception, Traitement, et Communication	17
I. 2. 2. 1. Exigences d'Autonomie	17
I. 2. 2. 2. Stratégies Sensorielles Intelligentes (Fusion de Données des Capteurs)	18
I. 2. 2. 3. Calculateurs Massivement Parallèles	19
I. 2. 3. Planification et Contrôle de la Navigation	19
I. 2. 3. 1. Approches Classiques	20
I. 2. 3. 2. Approches Actuelles	22
I. 3. LA PROBLEMATIQUE DE NAVIGUER INTELLIGEMMENT EN ENVIRONNEMENTS DYNAMIQUES	25



I. 3. 1. Comportement Intelligent de Localisation de la Cible (<i>CIB</i>)	26
I. 3. 2. Comportement Intelligent d'Evitement d'Obstacles (<i>OBS</i>)	26
I. 3. 3. Comportement Intelligent de Prise de Décisions (<i>DEC</i>)	27
I. 3. 4. Comportement Intelligent d'Action (<i>ACT</i>)	27
I. 4. OBJECTIFS VISES	
I. 4. 1. Applications	
I. 4. 1. 1. Applications Visées	27
I. 4. 1. 2. Environnements Dynamiques Partiellement Structurés	28
I. 4. 2. Architecture, Mobilité, Perception, Traitement, et Communication	29
I. 4. 2. 1. Véhicules Visés	29
I. 4. 2. 2. "Autonomous Guided Vehicle" (<i>AGV</i>)	29
I. 4. 3. Planification et Contrôle de la Navigation	31
I. 4. 3. 1. Planification et Contrôle Visés	32
I. 4. 3. 2. Approches Basées sur le "Soft Computing" (<i>SC</i>)	32
I. 5. SYNTHESE SUR LES APPROCHES BASEES SUR LE "SOFT COMPUTING" (<i>SC</i>)	33
I. 5. 1. Systèmes Intelligents (<i>SI</i>): Réseaux de Neurones (<i>RN</i>)	33
I. 5. 2. Systèmes Intelligents (<i>SI</i>): Logique Floue (<i>LF</i>)	33
I. 5. 3. Systèmes Hybrides Intelligents (<i>SHI</i>): Réseaux de Neurones (<i>RN</i>) et Algorithmes Génétiques (<i>AG</i>)	34
I. 5. 4. Systèmes Hybrides Intelligents (<i>SHI</i>): Réseaux de Neurones (<i>RN</i>) et Logique Floue (<i>LF</i>)	35
I. 5. 4. 1. Propriétés des <i>RN</i> et de la <i>LF</i>	35
I. 5. 4. 2. Analyse Critique sur les <i>RN</i> et sur la <i>LF</i>	35
I. 5. 4. 3. Synthèse sur la Navigation Basée sur les <i>RN</i> et sur la <i>LF</i>	36
I. 5. 4. 4. Analyse Critique sur la Navigation Basée sur les <i>RN</i> et sur la <i>LF</i>	38
I. 5. 5. Systèmes Hybrides Intelligents (<i>SHI</i>): Réseaux de Neurones (<i>RN</i>), Logique Floue (<i>LF</i>), et Résonance Adaptative (<i>RA</i>)	38
I. 5. 6. Systèmes Hybrides Intelligents (<i>SHI</i>): Systèmes Experts (<i>SE</i>), Réseaux de Neurones (<i>RN</i>), et Logique Floue (<i>LF</i>)	40
I. 6. CONCLUSION	40

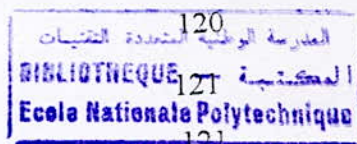
II. PLANIFICATION ET CONTROLE DE LA NAVIGATION: Vers une	
Autonomie et une Intelligence des Véhicules	42
II. 1. INTRODUCTION	43
II. 2. STRATEGIES DE NAVIGATION, NAV_1 ET NAV_2, PROPOSEES	43
II. 2. 1. Stratégie NAV_1	44
II. 2. 1. 1. Stratégie NAV_1_Simplifiée	45
II. 2. 1. 2. Stratégie NAV_1_Généralisée	45
II. 2. 2. Stratégie NAV_2	50
II. 2. 2. 1. Stratégie NAV_2_Simplifiée	51
II. 2. 2. 2. Stratégie NAV_2_Généralisée	55
II. 2. 3. Comportement d'Action (ACT) Proposé aux Stratégies NAV_1_NAV_2_Généralisées	59
II. 3. SI ET SHI PROPOSES POUR L'ACQUISITION DES COMPORTEMENTS DE LOCALISATION DE LA CIBLE (CIB) ET D'EVITEMENT D'OBSTACLES (OBS)	64
II. 3. 1. Stratégie NAV_1_Simplifiée	64
II. 3. 1. 1. SI_1: CIB_NAV_1_Simplifiée-RN/RPG	65
II. 3. 1. 2. SI_1: OBS_NAV_1_Simplifiée-RN/RPG	66
II. 3. 1. 3. SI_2: OBS_NAV_1_Simplifiée-LF	67
II. 3. 1. 4. SHI_1: OBS_NAV_1_Simplifiée-RN/AG	69
II. 3. 1. 5. SHI_2: OBS_NAV_1_Simplifiée-RN/AG-RPG	69
II. 3. 1. 6. SHI_3: OBS_NAV_1_Simplifiée-RNF/RPG	70
II. 3. 1. 7. SHI_4: OBS_NAV_1_Simplifiée-RNAMF/AMFS	72
II. 3. 2. Stratégie NAV_1_Généralisée	73
II. 3. 2. 1. SI_1: CIB_NAV_1_Généralisée-RN/RPG	75
II. 3. 2. 2. SI_1: OBS_NAV_1_Généralisée-RN/RPG	75
II. 3. 2. 3. SHI_4: CIB_NAV_1_Généralisée-RNAMF/AMFS	75
II. 3. 2. 4. SHI_4: OBS_NAV_1_Généralisée-RNAMF/AMFS	76
II. 3. 3. Stratégie NAV_2_Généralisée	77
II. 3. 3. 1. SHI_3: OBS_NAV_2_Généralisée-RNF/RPG: Degrés du Danger Statique	78



II. 3. 3. 2. <i>SHI_3: OBS_NAV_2_Généralisée-RNF/RPG</i> : Degrés du Danger Dynamique	80
II. 4. SI PROPOSES POUR L'ACQUISITION DU COMPORTEMENT DE PRISE DE DECISIONS (DEC)	81
II. 4. 1. <i>SI_3: DEC_NAV_1_Généralisée-RN/REE</i>	81
II. 4. 1. 1. Association du Comportement <i>CIB</i> aux Actions Favorables	83
II. 4. 1. 2. Association du Comportement <i>OBS</i> aux Actions Favorables	84
II. 4. 1. 3. Coordination des Comportements <i>CIB</i> et <i>OBS</i>	84
II. 4. 2. <i>SI_1: DEC_NAV_2_Généralisée-RN/RPG</i>	85
II. 5. SI PROPOSE POUR L'ACQUISITION DU COMPORTEMENT D'ACTION (ACT): SI_2: ACT_NAV_1_NAV_2_Généralisées-LF	87
II. 6. SYSTEMES HYBRIDES INTELLIGENTS (SHI) PROPOSES POUR LA PLANIFICATION ET LE CONTROLE DE LA NAVIGATION	89
II. 6. 1. Stratégie <i>NAV_1_Généralisée</i>	90
II. 6. 1. 1. <i>SHI_A_NAV_1_Généralisée</i>	90
II. 6. 1. 2. <i>SHI_B_NAV_1_Généralisée</i>	91
II. 6. 2. Stratégie <i>NAV_2_Généralisée: SHI_SECOV_NAV_2_Généralisée</i>	92
II. 7. CONCLUSION	94
III. SIMULATIONS, TESTS, ET RESULTATS	95
III. 1. INTRODUCTION	96
III. 2. ENVIRONNEMENT DE SIMULATION	96
III. 2. 1. Simulation des Environnements Dynamiques Partiellement Structurés	96
III. 2. 2. Simulation des Véhicules	97
III. 2. 3. Simulation des Capteurs à UltraSons (<i>US</i>)	97
III. 3. ENTRAINEMENTS ET MISE EN OEUVRE DES SI ET SHI POUR LES COMPORTEMENTS CIB ET OBS	97
III. 3. 1. Stratégie <i>NAV_1_Simplifiée</i>	97
III. 3. 1. 1. <i>SI_1: CIB_NAV_1_Simplifiée-RN/RPG</i>	98
III. 3. 1. 2. <i>SI_1: OBS_NAV_1_Simplifiée-RN/RPG</i>	99
III. 3. 1. 3. <i>SI_2: OBS_NAV_1_Simplifiée-LF</i>	99
III. 3. 1. 4. <i>SHI_1: OBS_NAV_1_Simplifiée-RN/AG</i>	99

III. 3. 1. 5. <i>SHI_2: OBS_NAV_1 Simplifiée-RN/AG-RPG</i>	
III. 3. 1. 6. <i>SHI_3: OBS_NAV_1 Simplifiée-RNF/RPG</i>	100
III. 3. 1. 7. <i>SHI_4: OBS_NAV_1 Simplifiée-RNAMF/AMFS</i>	100
III. 3. 2. <i>Stratégie NAV_1 Généralisée</i>	101
III. 3. 2. 1. <i>SI_1: CIB_NAV_1 Généralisée-RN/RPG</i>	101
III. 3. 2. 2. <i>SI_1: OBS_NAV_1 Généralisée-RN/RPG</i>	102
III. 3. 2. 3. <i>SHI_4: CIB_NAV_1 Généralisée-RNAMF/AMFS</i>	103
III. 3. 2. 4. <i>SHI_4: OBS_NAV_1 Généralisée-RNAMF/AMFS</i>	103
III. 3. 3. <i>Stratégie NAV_2 Généralisée</i>	103
III. 3. 3. 1. <i>SHI_3: OBS_NAV_2 Généralisée-RNF/RPG: Degrés du Danger</i> Statique	103
III. 3. 3. 2. <i>SHI_3: OBS_NAV_2 Généralisée-RNF/RPG: Degrés du Danger</i> Dynamique	104
III. 4. ENTRAINEMENTS ET MISE EN OEUVRE DES SI POUR LE COMPORTEMENT DEC	104
III. 4. 1. <i>SI_3: DEC_NAV_1 Généralisée-RN/REE</i>	104
III. 4. 1. 1. Association du Comportement <i>CIB</i> aux Actions Favorables	105
III. 4. 1. 2. Association du Comportement <i>OBS</i> aux Actions Favorables	106
III. 4. 1. 3. Coordination des Comportements <i>CIB</i> et <i>OBS</i>	108
III. 4. 2. <i>SI_1: DEC_NAV_2 Généralisée-RN/RPG</i>	109
III. 5. MISE EN OEUVRE DU SI_2: ACT_NAV_1_NAV_2 Généralisées-LF POUR LE COMPORTEMENT ACT	110
III. 6. RESULTATS DE LA SIMULATION DES STRATEGIES DE NAVIGATION	111
III. 6. 1. <i>Stratégie NAV_1 Généralisée</i>	112
III. 6. 1. 1. <i>SHI_A_NAV_1 Généralisée</i>	112
III. 6. 1. 2. <i>SHI_B_NAV_1 Généralisée</i>	114
III. 6. 2. <i>Stratégie NAV_2 Généralisée: SHI_SECOV_NAV_2 Généralisée</i>	116
III. 7. CONCLUSION	119

IV. DISCUSSIONS DES RESULTATS DE SIMULATION, IMPLEMENTATION DES RESEAUX DE NEURONES (RN), ET PERSPECTIVES



IV. 1. INTRODUCTION

IV. 2. DISCUSSIONS DES RESULTATS DE SIMULATION

IV. 2. 1. *SI* et *SHI* pour l'Acquisition des Comportements Intelligents 121

IV. 2. 1. 1. Performance des *SI* et *SHI* 121

IV. 2. 1. 2. Comparaison des *SI* et *SHI* Proposés au Comportement *OBS*
(*NAV_1_Simplifiée*) 121

IV. 2. 1. 3. *SI* et *SHI* pour l'Acquisition des Comportements *CIB*, *OBS*,
DEC, et *ACT* 123

IV. 2. 2. *SHI* pour la Planification et le Contrôle des Stratégies
NAV_1_Généralisée et *NAV_2_Généralisée* 125

IV. 3. IMPLEMENTATION D'UN RESEAU DE NEURONES (RN) SUR
"FIELD-PROGRAMMABLE GATE ARRAY" (FPGA) 127

IV. 4. PERSPECTIVES

IV. 4. 1. Apprentissage Non-Supervisé 129

IV. 4. 2. Systèmes Hybrides Intelligents (*SHI*) 129

IV. 4. 3. Cibles Dynamiques 129

IV. 4. 4. Navigation en Groupes de *VAI* 130

IV. 4. 5. Implémentation des *SI* et *SHI* Proposés 131

IV. 4. 6. Implémentation des Stratégies *NAV_1_Généralisée* et
NAV_2_Généralisée Proposées sur l'*AGV* 131

IV. 5. CONCLUSION

CONCLUSION GENERALE 134

REFERENCES BIBLIOGRAPHIQUES 138

ANNEXES 148

ANNEXE A: DESCRIPTION DE L'"AUTONOMOUS GUIDED VEHICLE"
(AGV) 149

A. 1. Architecture de l'AGV	149
A. 2. Roues	49
A. 3. Moteurs-Réducteurs	50
A. 4. Codeurs	50
A. 5. Capteurs à UltraSons (US)	150
A. 6. Capteurs à Infrarouges	151
A. 7. Batteries	151
ANNEXE B: PROPRIETES DES RN ET DE LA LF	152
B. 1. Propriétés des RN et de la LF	152
B. 2. Propriétés Complémentaires des RN et de la LF	153
ANNEXE C: APPRENTISSAGES SUPERVISES	154
C. 1. Rétro-Propagation du Gradient (RPG)	154
C. 1. 1. Topologie des RN et Convergence par la RPG	154
C. 1. 2. Apprentissage Supervisé par la RPG	156
C. 2. Algorithmes Génétiques (AG) ou AG-RPG	158
C. 2. 1. Apprentissage Supervisé par les AG	158
C. 2. 2. Apprentissage Supervisé par AG-RPG	160
C. 3. "ArtMap" Floue Simplifiée (AMFS)	160
C. 3. 1. Réseaux de Neurones Basés sur la "ArtMap" Floue (RNAMEF)	161
C. 3. 2. Apprentissage Supervisé par la AMFS	161
C. 4. Renforcement par Essai et Erreur (REE)	165
C. 4. 1. Association Entre Situations et Actions	165
C. 4. 2. Mise à Jour des Poids Entre Situations et Actions	166
ANNEXE D: BASES DE REGLES FLOUES	167
D. 1. Base de Règles Floues 1	167
D. 2. Base de Règles Floues 2	168
D. 3. Base de Règles Floues 3	169
D. 4. Base de Règles Floues 4	169
D. 5. Distances d'Accélération/Décélération pour Chaque Combinaison des Vitesse Finale, Vitesse Initiale, et Distance-But	170
D. 6. Base de Règles Floues 5	174

LISTE DES ABREVIATIONS

- 1) *ACT* (Comportement d'*ACT*ion).
- 2) *AG* (Algorithmes Génétiques).
- 3) *AGV* ("Autonomous Guided Vehicle").
- 4) *AMFS* ("ArtMap" Floue Simplifiée).
- 5) *Cib* (Cible).
- 6) *CIB* (Comportement de Localisation de la *CIB*le).
- 7) *DEC* (Comportement de Prise de *DEC*isions).
- 8) *FPGA* ("Field-Programmable Gate Array").
- 9) *LF* (Logique Floue).
- 10) *NAV_1_Simplifiée* (Stratégie de *NAV*igation 1 *Simplifiée*).
- 11) *NAV_1_Généralisée* (Stratégie de *NAV*igation 1 *Généralisée*).
- 12) *NAV_2_Simplifiée* (Stratégie de *NAV*igation 2 *Simplifiée*).
- 13) *NAV_2_Généralisée* (Stratégie de *NAV*igation 2 *Généralisée*).
- 14) *Obs* (*Ob*stacle).
- 15) *OBS* (Comportement d'Evitement d'*OB*stacles).
- 16) *RA* (Résonance Adaptative).
- 17) *REE* (Renforcement par *Essai* et *Erreur*).
- 18) *Rég* (Région).
- 19) *RN* (Réseaux de Neurones).
- 20) *RNAMF* (Réseaux de Neurones basés sur la "ArtMap" Floue).
- 21) *RNF* (Réseaux de Neurones Flous).
- 22) *RPG* (Rétro-Propagation du Gradient).
- 23) *SC* ("Soft Computing").
- 24) *SE* (Systèmes Experts).
- 25) *SECOV* (Système Expert pour le *CO*ntrôle des *Véhicules*).
- 26) *SENF* (Systèmes Experts Neuro-Flous).
- 27) *SHI* (Systèmes Hybrides Intelligents).
- 28) *SI* (Systèmes Intelligents).
- 29) *US* (UltraSons).
- 30) *VAI* (Véhicules Autonomes Intelligents).
- 31) *Véh* (Véhicule).

LISTE DES FIGURES

- Figure I. 1. Architecture Globale des Comportements Intelligents de la Navigation.
- Figure I. 2. Environnement Dynamique Partiellement Structuré.
- Figure I. 3. Configuration de l'“Autonomous Guided Vehicle” (AGV).
- Figure II. 1. Synoptique de la Stratégie de Navigation *NAV_1*.
- Figure II. 2. Véhicule, Capteurs, et Situations de Position de la Cible $C = [C_1, \dots, C_{j1}, \dots, C_6]$ (*NAV_1_Simplifiée*).
- Figure II. 3. Situations d'Evitement d'Obstacles $O = [O_1, \dots, O_{j2}, \dots, O_6]$ (*NAV_1_Simplifiée*).
- Figure II. 4. Véhicule, Capteurs, et Situations de Position de la Cible $C = [C_1, \dots, C_{j1}, \dots, C_6]$ (*NAV_1_Généralisée*).
- Figure II. 5. Situations d'Evitement d'Obstacles $O = [O_1, \dots, O_{j2}, \dots, O_{30}]$ (*NAV_1_Généralisée*).
- Figure II. 6. Synoptique de la Stratégie de Navigation *NAV_2*.
- Figure II. 7. Véhicule, Capteurs, et Situations de Position de la Cible $C = [C_1, \dots, C_i, \dots, C_{11}]$ (*NAV_2_Simplifiée* et *NAV_2_Généralisée*).
- Figure II. 8. Modèle d'Evitement d'Obstacles du Véhicule (*NAV_2_Simplifiée* et *NAV_2_Généralisée*).
- Figure II. 9. Fonctions d'Appartenance de la Direction.
- Figure II. 10. Fonctions d'Appartenance de la Distance.
- Figure II. 11. Fonctions d'Appartenance des Degrés des Dangers Statique et Dynamique.
- Figure II. 12. Fonctions d'Appartenance de la Direction de la Vitesse.
- Figure II. 13. Fonctions d'Appartenance de la Vitesse.
- Figure II. 14. Bloc-Diagramme de l'Algorithme de Décisions (*NAV_2_Simplifiée*).
- Figure II. 15. (a). Sélection de Points Obstacles: Cas d'un Obstacle Occupant une Seule Région.
- Figure II. 15. (b). Sélection de Points Obstacles: Cas d'un Obstacle Occupant Plus d'une Région.
- Figure II. 16. Bloc-Diagramme de l'Algorithme de Décisions (*NAV_2_Généralisée*).
- Figure II. 17. Profil de la Vitesse du Véhicule sous le Contrôle Temps Minimum.
- Figure II. 18. Fonctions d'Appartenance de la Vitesse.
- Figure II. 19. Fonctions d'Appartenance de la Distance-But.
- Figure II. 20. Fonctions d'Appartenance de la Distance Courante.

Figure II. 21. *SI* et *SHI* pour les Comportements *CIB* et *OBS* (*NAV_1_Simplifiée* et *NAV_1_Généralisée*).

Figure II. 22. Architecture du *SI_2: OBS_NAV_1_Simplifiée-LF*.

Figure II. 23. Architecture du *SHI_3: OBS_NAV_1_Simplifiée-RNF/RPG*.

Figure II. 24. Architecture du *SHI_4: OBS_NAV_1_Simplifiée-RNAMF/AMFS*.

Figure II. 25. (a). Architecture du *SHI_4: CIB_NAV_1_Généralisée-RNAMF/AMFS*.

Figure II. 25. (b). Architecture du *SHI_4: OBS_NAV_1_Généralisée-RNAMF/AMFS*.

Figure II. 26. *SHI* pour le Comportement *OBS* (*NAV_2_Généralisée*).

Figure II. 27. (a). Architecture du *SHI_3: OBS_NAV_2_Généralisée-RNF/RPG*: Degrés du Danger Statique.

Figure II. 27. (b). Architecture du *SHI_3: OBS_NAV_2_Généralisée-RNF/RPG*: Degrés du Danger Dynamique.

Figure II. 28. *SI* pour le Comportement *DEC* (*NAV_1_Généralisée* et *NAV_2_Généralisée*).

Figure II. 29. Architecture du *SI_3: DEC_NAV_1_Généralisée-RN/REE*.

Figure II. 30. Représentation des Actions Favorables pour les Situations de Position de la Cible (*NAV_1_Généralisée*).

Figure II. 31. Architecture du *SI_1: DEC_NAV_2_Généralisée-RN/RPG*.

Figure II. 32. *SI* pour le Comportement *ACT* (*NAV_1_NAV_2_Généralisées*).

Figure II. 33. Architecture du *SI_2: ACT_NAV_1_NAV_2_Généralisées-LF*.

Figure II. 34. *SHI_A_NAV_1_Généralisée*.

Figure II. 35. *SHI_B_NAV_1_Généralisée*.

Figure II. 36. (a). *SHI_SECOV_NAV_2_Généralisée*.

Figure II. 36. (b). *SHI_SECOV_NAV_2_Généralisée* (Base de Faits).

Figure III. 1. Environnement Dynamique Partiellement Structuré.

Figure III. 2. Environnement d'Entraînement (*CIB_OBS_NAV_1_Simplifiée*).

Figure III. 3. (a). Environnement d'Entraînement (*CIB_NAV_1_Généralisée*).

Figure III. 3. (b). Environnement d'Entraînement (*OBS_NAV_1_Généralisée*).

Figure III. 4. (a). Matrice d'Association (*CIB_ACT_NAV_1_Généralisée*).

Figure III. 4. (b). Matrice d'Association (*OBS_ACT_NAV_1_Généralisée*).

Figure III. 5. Courbes des Distances et Vitesses du *SI_2: ACT_NAV_1_NAV_2_Généralisées-LF*.

Figure III. 6. Obstacles Statiques (*SHI_A_NAV_1_Généralisée*).

Figure III. 7. Obstacles Dynamiques Intelligents (*SHI_A_NAV_1_Généralisée*)

Figure III. 8. Obstacles Dynamiques Non-Intelligents (*SHI_A_NAV_1_Généralisée*).

Figure III. 9. Environnement Dynamique Partiellement Structuré
(*SHI_A_NAV_1_Généralisée*).

Figure III. 10. Obstacles Statiques (*SHI_B_NAV_1_Généralisée*).

Figure III. 11. Obstacles Dynamiques Intelligents (*SHI_B_NAV_1_Généralisée*).

Figure III. 12. Obstacles Dynamiques Non-Intelligents (*SHI_B_NAV_1_Généralisée*).

Figure III. 13. Environnement Dynamique Partiellement Structuré
(*SHI_B_NAV_1_Généralisée*).

Figure III. 14. Obstacles Statiques (*SHI_SECOV_NAV_2_Généralisée*).

Figure III. 15. Obstacles Dynamiques Intelligents (*SHI_SECOV_NAV_2_Généralisée*).

Figure III. 16. Obstacle Dynamique Non-Intelligent (*SHI_SECOV_NAV_2_Généralisée*).

Figure III. 17. Environnement Dynamique Partiellement Structuré
(*SHI_SECOV_NAV_2_Généralisée*).

Figure IV. 1. Architecture *FPGA* pour le *SI_1: OBS_NAV_1_Simplifiée-RN/RPG*.

Figure IV. 2. Exigences Actuelles d'Autonomie et d'Intelligence des *VAI*.

Figure A. 1. Architecture de l'*AGV*.

Figure A. 2. Positions des Capteurs à *US* sur l'*AGV*.

Figure C. 1. Architecture des *SI* et *SHI* Basés sur les *RN/RPG*, les *RN/AG*, et les *RN/AG-RPG*.

Figure C. 2. Surface d'Erreur Typique pour les *RN* Multicouches avec des Fonctions
d'Activation Non-Linéaires.

Figure C. 3. Synoptique de l'Apprentissage Supervisé par les *AG* ou par *AG-RPG*.

Figure C. 4. Codage des Chromosomes, Croisement, et Mutation.

Figure C. 5. Architecture Générale des *RNAMF*.

Figure C. 6. Configuration Initiale du *SHI_4: OBS_NAV_1_Simplifiée-RNAMF/AMFS*.

LISTE DES TABLEAUX



Tableau II. 1. Décision de la Direction d'Evitement.

Tableau IV. 1. Paramètres des *SI* et *SHI* Proposés au Comportement *OBS*
(*NAV_1_Simplifiée*).

Tableau B. 1. Propriétés des *RN* et de la *LF*.

Tableau B. 2. Propriétés Complémentaires des *RN* et de la *LF*.

Tableau D. 1. Base de Règles Floues 1.

Tableau D. 2. Base de Règles Floues 2.

Tableau D. 3. Base de Règles Floues 3.

Tableau D. 4. Base de Règles Floues 4.

Tableau D. 5. Distances d'Accélération/Décélération pour Chaque Combinaison des Vitesse
Finale, Vitesse Initiale, et Distance-But.

Tableau D. 6. Base de Règles Floues 5.

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

INTRODUCTION GENERALE

De nos jours les Véhicules Autonomes Intelligents (VAI) qui réalisent des tâches sans intervention d'un opérateur sont nécessaires dans plusieurs domaines d'application et peuvent ainsi accomplir des tâches complexes dans divers environnements dynamiques par eux-mêmes tel un être humain. En fait, les applications spatiales, sous-marines, et terrestres de ces systèmes sont d'une importance de plus en plus grande pour plusieurs raisons à savoir: la minimisation de la fatigue de l'opérateur humain, la réduction des risques d'exposition de l'opérateur, la minimisation du coût des opérations, ainsi que le contrôle de qualité des produits ou des opérations. Ces différentes raisons ont conduit indéniablement vers la conception et le développement de VAI capables d'un certain degré d'*auto-suffisance*, d'*autonomie*. Ces VAI doivent alors être capables d'exécuter des comportements *réactifs* et *réfléchis* dans des environnements réels. En fait, ils obtiennent des entrées à partir de l'environnement à travers des capteurs et doivent agir sur celui-ci à travers leurs actionneurs. En effet, la connexion entre la perception et la mise en action implique un système englobant la prise de décisions, l'interprétation de buts, et d'autres aspects du raisonnement.

Les récents développements dans les exigences d'autonomie, des composants intelligents, des systèmes multi-véhicules, et des calculateurs massivement parallèles ont ouvert le champ d'application des VAI à une large variété de domaines. En effet, ces dernières années ont été très riches d'exploits dans la recherche et le développement des VAI en particulier dans les explorations spatiales et sous-marines, ainsi que les applications terrestres (industrielles, agricoles, médicales, et sur autoroutes) [1-11]. Nous citons entre autres:

- Le 24 Mars 1995, le véhicule guidé à distance, Kaiko, a navigué dans la plus profonde région Mariana Trough de l'océan Pacifique.
- En Juillet 1997, la NASA envoie pour la première fois le "Pathfinder rover", le robot mobile Sojourner, sur la planète Mars [10] ; l'utopie fut alors réalité en moins d'une décennie.
- En Août 1997, la première coupe du monde de "football" des robots appelée RoboCup97 s'est déroulée à Nagoya, Japon, avec la participation de plus de quarante (40) équipes. RoboCup97 a été une initiative importante visant à donner une nouvelle impulsion à la recherche en intelligence artificielle et en robotique intelligente.
- En Septembre 1997, l'Université de Tokyo présente le robot humanoïde, Saika, de taille humaine et se démarquant, par rapport à d'autres robots humanoïdes par son intelligence, son faible poids et son faible coût [8]. En fait, le premier robot humanoïde dans le monde, WABOT-1 (WAseda roBOT 1), a été développé en 1973 au Japon par I. Kato et amélioré en

WABOT-2, le robot 'musicien', qui a participé à une manifestation internationale en 1985 à Tsukuba, Japon [8].

- En 1997, la démonstration, Demo'97, des systèmes de véhicules intelligents sur autoroutes [2, 3, 4, 7, 9] a confirmé que l'automatisation véhicule-autoroute est souhaitable, et surtout réalisable [7, 9].

En dépit de l'aspect spectaculaire de ces événements, ils témoignent d'une avancée technologique certaine à la fois dans la recherche en robotique et en intelligence artificielle, particulièrement dans la planification et le contrôle de la navigation des *VAI* en environnements dynamiques. Les concepteurs de ces *VAI* cherchent alors à créer des systèmes dynamiques capables de naviguer et de réaliser des *comportements intelligents* similaires à ceux de l'être humain dans des environnements dynamiques réels où les conditions sont difficiles. Cependant, la complexité des environnements est un problème spécifique à résoudre puisque ces environnements peuvent être *imprécis, vastes, dynamiques, et partiellement ou non structurés*.

Dans les approches classiques de planification et de contrôle de la navigation, il a été souvent supposé une connaissance à priori et exacte du véhicule, de ses capteurs, et plus encore de son environnement. Ceci a limité le type d'environnements où de tels véhicules peuvent accomplir des tâches, et par conséquent le type de tâches que ces véhicules peuvent résoudre. En effet, ces limitations sont dues principalement aux difficultés qu'ont les concepteurs à fournir des modèles exacts des véhicules et de leurs environnements. Même si une telle connaissance est disponible, la rendant accessible par ordinateur, le codage à la main de modèles explicites a souvent exigé des temps de programmation non raisonnables. Par ailleurs, atteindre une facilité de contrôle du calcul ("*tractability*") est un sérieux obstacle pour la conception de structures de contrôle pour les *VAI* dans des environnements dynamiques complexes [12].

De nouvelles approches ont alors été développées pour la planification et le contrôle de la navigation où la supposition d'une connaissance parfaite de l'environnement a été supprimée. Elles ont également supprimé la représentation *explicite* de la connaissance pour une représentation *implicite* basée sur l'*acquisition de comportements intelligents* qui habilite le véhicule à interagir efficacement avec son environnement [13]. Par conséquent, les *VAI* se heurtent à des environnements moins prévisibles et plus complexes, ils ont à s'orienter, à explorer leurs environnements de manière autonome, à s'auto-dépanner, et à exécuter des familles complètes de tâches en *temps réel*. De plus, si les véhicules manquent d'une connaissance initiale les concernant eux-mêmes et leurs environnements, l'*apprentissage*

devient inévitable. Cet apprentissage se réfère à une variété d'algorithmes caractérisés par leurs capacités de remplacer une connaissance de l'environnement manquante ou incorrecte par l'*expérimentation*, l'*observation*, et la *généralisation*. En d'autres termes, l'apprentissage de véhicules compte sur l'interaction avec l'environnement pour extraire l'information utile afin d'atteindre un but donné [12].

Ainsi, les *VAI* doivent être capables de comprendre la structure de leurs environnements pour atteindre leurs cibles sans collisions avec d'éventuels obstacles rencontrés. Ils doivent donc avoir la capacité de réaliser des comportements intelligents de localisation de la cible (*CIB*), d'évitement d'obstacles (*OBS*), de prise de décisions (*DEC*), et d'action (*ACT*). Ces comportements intelligents font appel à la reconnaissance (traitement de données imprécises issues de capteurs, construction d'une base de connaissances, et établissement d'une carte de l'environnement), l'apprentissage, l'adaptation, le raisonnement et la prise de décisions, et l'action. En fait, ces derniers constituent les principaux problèmes de la navigation des *VAI* en environnements dynamiques.

Afin de résoudre ces problèmes, les approches classiques [14–24] ont rapidement été remplacées par les approches actuelles basées sur la collection de méthodologies appelée "Soft Computing" (*SC*) impliquant les Réseaux de Neurones (*RN*), la Logique Floue (*LF*), les Algorithmes Génétiques (*AG*), la Résonance Adaptative (*RA*), et les Systèmes Experts (*SE*) utilisés individuellement en Systèmes Intelligents (*SI*) ou en différentes combinaisons en Systèmes Hybrides Intelligents (*SHI*) [12, 13, 25–60]. En fait, d'une part le but du *SC* est de faire évoluer le comportement des *VAI* vers celui de l'être humain dans la reconnaissance, l'apprentissage, l'adaptation, le raisonnement et la prise de décisions, et l'action [6, 11, 52, 61–74]. D'autre part, ces *SI* et *SHI* sont reconnus pour leur amélioration des capacités d'*apprentissage*, d'*adaptation*, et de *généralisation* relatives aux variations dans des environnements où l'information est *qualitative*, *imprécise*, *incertaine*, ou *incomplète*.

Etant donné que nous visons une planification et un contrôle de la navigation qui puissent assurer aux *VAI* plus d'*autonomie* et plus d'*intelligence* avec des capacités de traitement *temps réel*, les comportements intelligents *CIB*, *OBS*, *DEC*, et *ACT* doivent être pris en charge impérativement par les approches actuelles de planification et de contrôle de la navigation. Notons que nous visons essentiellement les véhicules de type voiture ou tricycle en environnements dynamiques partiellement structurés et en vue de futures implémentations, une

attention particulière est portée vers le véhicule de type tricycle: "Autonomous Guided Vehicle" (AGV) développé au Laboratoire de Robotique et d'Intelligence Artificielle (LRIA) du Centre de Développement des Technologies Avancées (CDTA).

Ainsi, afin de faire évoluer le comportement de ces VAI vers celui de l'être humain notamment dans la reconnaissance, l'apprentissage, l'adaptation, le raisonnement et la prise de décisions, et l'action, nous nous sommes proposés de résoudre le problème de la planification et du contrôle de la navigation des VAI en environnements dynamiques utilisant les approches actuelles basées sur le SC. Ces approches qui fournissent aux VAI plus d'autonomie et plus d'intelligence avec des capacités de traitement *temps réel* leur permettent ainsi de naviguer *intelligemment* dans des environnements *dynamiques partiellement structurés*.

Dans cette thèse, nous présentons, en quatre (04) Chapitres, le thème de recherche de la planification et du contrôle de la navigation des VAI en environnements dynamiques, les problèmes que nous nous sommes proposés à résoudre, les objectifs visés, les solutions proposées, les résultats obtenus, ainsi que les perspectives envisagées.

Dans le Chapitre I, nous donnons les définitions de base permettant de comprendre les différents aspects des VAI, des environnements dynamiques, et de la navigation. Nous présentons ensuite, l'évolution des VAI par rapport à différentes applications (terrestres, sous-marines, et spatiales), à leurs architecture, mobilité, perception, traitement de données, et communication, ainsi qu'à la planification et le contrôle de la navigation.

Nous définissons alors les principaux problèmes à résoudre dans la problématique de naviguer *intelligemment* en environnements *dynamiques*. De ce fait, les réponses aux questions suivantes deviennent alors primordiales:

Question 1: par quels systèmes les VAI peuvent-ils acquérir des comportements intelligents *CIB*, *OBS*, *DEC*, et *ACT* dans des environnements dynamiques ?

Question 2: sur quels types d'approches se basent de tels systèmes ?

Question 3: comment développer et mettre en oeuvre ces approches ?

L'établissement des objectifs visés devient donc nécessaire notamment par rapport aux types d'applications, aux types de véhicules, et aux types d'approches de planification et contrôle de la navigation. Ensuite, à travers une synthèse sur les différentes approches de planification et de contrôle des VAI, basées sur le SC, la nécessité, les propriétés, ainsi que les avantages de ces approches sont présentées et discutées. En effet, un intérêt particulier est

porté aux approches basés soit sur des *SI* utilisant soit les *RN* soit la *LF*, ainsi que sur des *SHI* combinant les *RN*, la *LF*, les *AG*, la *RA*, et les *SE*.

L'objectif principal du Chapitre II est de développer des approches de planification et de contrôle de la navigation capables de fournir aux *VAI* plus d'*autonomie* et plus d'*intelligence* avec des capacités de traitement *temps réel*. Ce qui se traduit par le développement et la mise en oeuvre de *SI* et *SHI*, basés sur le *SC*, pour l'acquisition de comportements intelligents *CIB*, *OBS*, *DEC*, et *ACT* de la navigation. Deux (02) stratégies de navigation *NAV_1* et *NAV_2* sont alors proposées en versions simplifiées et généralisées: *NAV_1_Simplifiée* et *NAV_1_Généralisée* et *NAV_2_Simplifiée* et *NAV_2_Généralisée*.

Ainsi, ces deux (02) stratégies de navigation utilisent différents *SI* et *SHI* proposés pour l'acquisition par *apprentissage* ou/et *adaptation* des comportements intelligents *CIB* et *OBS*, *DEC*, et *ACT*. Nous nous intéressons particulièrement aux *SI* basés soit sur les *RN* entraînés par l'*apprentissage supervisé* par la *Rétro-Propagation du Gradient (RPG)*: *RN/RPG*, soit sur la *LF*, ainsi qu'aux *SHI* basés sur les combinaisons *RN-AG*, *RN-LF*, *RN-LF-RA*, et *SE-RN-LF*:

- les *RN* entraînés par l'*apprentissage supervisé* par les *AG*: *RN/AG*,
- les *RN* entraînés par l'*apprentissage supervisé* par les *AG* suivis de la *RPG*: *RN/AG-RPG*,
- les Réseaux de Neurones Flous (*RNF*) entraînés par la *RPG*: *RNF/RPG*,
- les *RN* basés sur la "ArtMap" Floue (*RNAMF*) entraînés par l'*apprentissage supervisé* par la "ArtMap" Floue Simplifiée (*AMFS*): *RNAMF/AMFS*,
- les *RN* entraînés par l'*apprentissage supervisé* par Renforcement par Essai et Erreur (*REE*): *RN/REE*,
- les Systèmes Experts Neuro-Flous (*SENF*).

Les meilleurs de ces systèmes, pour chaque comportement, en termes de performance et de facilité de mise en oeuvre sont alors proposés en architectures globales: *SHI* (*SHI_A_NAV_1_Généralisée*, *SHI_B_NAV_1_Généralisée*, *SHI_SECOV_NAV_2_Généralisée*), englobant tous les comportements, pour la planification et le contrôle des deux (02) stratégies de navigation.

L'environnement de simulation, les entraînements et la mise en oeuvre des *SI* et *SHI* proposés pour les comportements intelligents *CIB*, *OBS*, *DEC*, et *ACT* ainsi que les résultats

de la simulation des stratégies *NAV_1_Généralisée* et *NAV_2_Généralisée*, sont présentés et discutés dans le Chapitre III.

Des discussions sous différents aspects des résultats obtenus par rapport aux *SI* et *SHI* proposés pour l'acquisition des comportements intelligents, ainsi qu'aux *SHI* proposés pour la planification et le contrôle des stratégies de navigation sont présentées dans le Chapitre IV. Dans ce même Chapitre, nous proposons une architecture "Field-Programmable Gate Array" (*FPGA*), basée sur la technologie Xilinx, pour l'implémentation d'un *SI* basé sur un *RN* réalisant le comportement intelligent *OBS*. Enfin, nous présentons différentes perspectives jugées intéressantes tant sur l'aspect théorique que celui d'implémentation.

Chapitre I

***NAVIGATION DES VEHICULES AUTONOMES
INTELLIGENTS (VAI) EN ENVIRONNEMENTS
DYNAMIQUES***

I. 1. INTRODUCTION

Aujourd'hui, l'application des contrôles autonomes intelligents s'introduit activement dans divers systèmes robotiques qui était jusque là en général semi-autonomes et ayant besoin d'une certaine supervision humaine. En effet, plusieurs systèmes robotiques traditionnels déjà utilisées dans certaines applications tendent peu à peu vers des systèmes guidées à distance ou même complètement autonomes, grâce au développement de la technologie des calculateurs de contrôle. Ces nouveaux systèmes appelés systèmes robotiques autonomes, particulièrement les *Véhicules Autonomes Intelligents (VAI)*, *flexibles* et *robustes* sont une composante clé de l'usine du futur et ont un avenir brillant dans diverses applications dans les explorations spatiales et sous-marines, l'industrie, la construction, les mines, l'agriculture, etc. L'industrie automobile p.ex., a investi énormément dans le développement des systèmes de perception et de contrôle afin de rendre les nouvelles voitures plus sécurisantes et plus faciles à conduire. La conduite autonome dans certaines conditions sera alors une réalité dans un futur proche. D'un autre côté, les chercheurs conçoivent, dans le domaine des structures et des systèmes mécaniques, de nouveaux types de *VAI*, pour les applications aussi bien terrestres, sous-marines, que spatiales [6, 11].

Ces systèmes robotiques autonomes peuvent résoudre les problèmes posés dans certaines tâches de travaux intensifs ou même pénibles utilisées dans les lignes d'assemblage industriel nécessitant l'habileté humaine. Plus particulièrement, les tâches répétitives peuvent d'un côté altérer le contrôle de qualité p.ex., les tâches d'inspection visuelle et d'un autre côté elles peuvent exposer l'opérateur humain à des problèmes de fatigue ou blessure associés à ces tâches. Au-delà de ces problèmes, il pourrait y avoir des risques de santé significatifs associés à l'environnement opérationnel tel que l'exposition à des matières dangereuses comme les déchets radioactifs ou chimiques ou encore à de hautes pressions dans les applications sous-marines. Ces raisons relatives à la productivité, au contrôle de qualité, au coût, et à la sécurité ont généralement été les principaux facteurs motivant le développement de tels systèmes robotiques autonomes, particulièrement des *VAI*. Toutefois, d'un point de vue technique, la fusion de données des capteurs, le traitement de données, la planification et le contrôle de tâches, la mise au point de calculateurs efficaces utilisant des processeurs de petites tailles et à faibles coûts, ont ainsi nécessité le développement de plusieurs sous-systèmes. La disponibilité de ces processeurs ainsi que la croissance des capacités de communication étant devenues une réalité, de petits systèmes autonomes (sans liens physiques) vont certainement jouer un rôle

dans l'expansion des capacités humaines dans des domaines tels que l'inspection des régions océaniques et l'exploration spatiale. Leur utilisation permet le transport d'instruments de mesures ainsi que le prélèvement d'échantillons dans différents sites à intérêts scientifiques p.ex., roches et crevasses. Ce qui augmente la flexibilité des missions, en particulier la capacité à réagir d'une manière adaptative à de réelles observations dans ce type de sites [5, 75]. Ces systèmes autonomes accomplissent des tâches alors sans contraintes en intégrant une énergie et une intelligence suffisantes, impliquant alors une communication avec l'utilisateur à un plus haut niveau, comparée à celles utilisées par les systèmes guidés à distance [6, 11].

Dans ce Chapitre, nous passons en revue quelques définitions concernant les *VAI*, les environnements dynamiques, et la navigation. Nous présentons dans § I. 2 l'évolution des *VAI*, leurs applications, architecture, mobilité, perception, traitement, communication ainsi que la planification et le contrôle intelligent de leur navigation. Nous posons alors la problématique de la navigation des *VAI* en environnements dynamiques, dans § I. 3, ainsi que les objectifs visés notamment ceux relatifs aux applications et environnements, aux types de *VAI*, et à la planification et le contrôle de la navigation, dans § I. 4. Enfin, nous présentons une synthèse sur les différentes approches de planification et de contrôle, basées sur le "Soft Computing" (*SC*) introduisant Réseaux de Neurones (*RN*), Logique Floue (*LF*), Algorithmes Génétiques (*AG*), Résonance Adaptative (*RA*), et Systèmes Experts (*SE*), dans § I. 5.

I. 1. 1. Véhicules Autonomes Intelligents (*VAI*)

Les performances des *VAI* sont essentiellement évaluées par rapport à l'exécution et la difficulté des tâches réalisées [6, 11]. Ce qui se résume par la question suivante: à quel point ces *VAI* remplissent les critères de réactivité et d'intelligence ?

- **Autonomie et Intelligence**

C'est la capacité des *VAI* à prendre en charge les tâches dans divers environnements par eux-mêmes imitant ainsi le comportement humain. Ils déterminent leurs propres actions, dans des environnements incertains, en se basant sur l'information sensorielle, et s'adaptent à ces environnements par l'acquisition de nouvelles connaissances par eux-mêmes à travers l'apprentissage [76, 52].

- **Adaptation et Réactivité**

C'est la capacité d'un *VAI* à s'adapter et à répondre aux changements dans un environnement imprévisible sans aucune carte ou connaissance à priori des positions ou des vitesses des éventuels obstacles statiques ou dynamiques. Un des objectifs du système est de réagir en temps-réel, dans des environnements dynamiques en utilisant p.ex., la notion d'apprentissage pour compléter sa réactivité, diminuant ainsi le temps de réponse.

- **Détection de Défaillances et Tolérance aux Défaillances**

C'est la capacité de *robustesse* d'un *VAI* à détecter rapidement, s'occuper, ou à remédier à des défaillances internes du système (réparer des pannes). La détection autonome de défaillances (pannes, bruits, ou erreurs) en *robotique* est devenue de plus en plus importante puisque les *VAI* sont utilisés dans des environnements de plus en plus inaccessibles et de plus en plus dangereux (applications sous-marines et spatiales). Avec cette capacité, le système est capable d'isoler effectivement les composants défaillants et de continuer l'exécution de sa tâche sans intervention humaine immédiate [77].

- **Fiabilité**

C'est la capacité d'un *VAI* à exécuter sans pannes la fonction exigée sous des conditions environnementales spécifiées durant un intervalle de temps donné. Elle dépend de la conception du système: la fiabilité de ses sous-systèmes, ses composants [78, 79].

- **Sécurité**

C'est la capacité d'un *VAI* à ne pas se mettre en danger ou mettre en danger d'autres intervenants (d'autres *VAI*, des objets statiques ou dynamiques) dans l'environnement. Elle dépend des caractéristiques des composants du système [78].

- **Utilité**

Ce critère est relatif généralement à la nécessité d'un *VAI* dans une application réelle et à son coût comparé à sa performance.

- **Innovation**

C'est le degré de nouveauté, de changement et de créativité de la conception des sous-systèmes des *VAI*.

En général, quand l'*autonomie* d'un *VAI* augmente, sa *robustesse*, sa détection de pannes et sa tolérance aux pannes augmentent également mais la *fiabilité* tend à diminuer à cause de l'augmentation de la complexité et du nombre de composants. Par ailleurs, un degré d'*autonomie* raisonnable aide à réduire le danger de l'erreur humaine et donc à augmenter la *sécurité*. Elle augmente de plus l'efficacité du système qui se manifeste par une réduction de la durée que le système doit passer dans un environnement dangereux et par conséquent elle a une influence positive sur la *fiabilité* [6, 11, 78].

I. 1. 2. Environnements Dynamiques

Les concepteurs des *VAI* cherchent à créer des systèmes dynamiques capables de naviguer et de réaliser des comportements intelligents comme ceux de l'être humain dans des environnements réels (dynamiques), où les conditions sont pénibles. Cependant, la complexité de ces environnements, qui peuvent être imprécis, vastes, dynamiques (inconnus), partiellement ou non structurés, voire dangereux est un problème spécifique à résoudre. La nature de ces environnements impose ainsi aux *VAI* des contraintes physiques (architecture, mobilité, masse, volume), sensorielles (perception d'obstacles, localisation, etc.), de traitement (calculateurs performants), de communication (stations fixes, ou d'autres *VAI*), de planification et de contrôle de la navigation (localisation de la cible, évitement d'obstacles, prise de décisions, et action).

I. 1. 3. Navigation

La navigation est l'art de contrôler l'évolution d'un véhicule au moment où il traverse l'environnement (terre, mer ou air) [80]. Les principaux problèmes de la navigation se résument en fait à la reconnaissance, l'apprentissage, l'adaptation, le raisonnement et la prise de décisions, et l'action. Trois phases sont nécessaires alors pour la reconnaissance à savoir: le traitement de données imprécises (issues de capteurs), la construction d'une base de connaissances et l'établissement d'une carte de l'environnement.

I. 2. EVOLUTION DES VEHICULES AUTONOMES INTELLIGENTS (VAI)

Après avoir donné quelques définitions essentielles, ce Paragraphe traite l'évolution, le comportement, et l'intelligence des VAI ainsi que leurs interactions avec les environnements dynamiques. Le but est de faire une étude approfondie sur les exigences d'autonomie et ses composants intelligents qui assurent réactivité et intelligence à ce type de véhicules.

I. 2. 1. Domaines d'Application

Nous présentons, dans ce Paragraphe, l'évolution, le comportement, et l'intelligence des VAI à travers quelques applications terrestres, sous-marines, et spatiales [6, 11].

I. 2. 1. 1. Applications Terrestres

La plupart des applications des VAI sont des applications terrestres notamment les applications médicales, industrielles, et des systèmes de véhicules intelligents sur autoroutes.

- **Applications Médicales**

Le nombre de plus en plus grand d'applications de la robotique dans le domaine médical démontre que la communauté médicale est réceptive aux nouveaux dispositifs. En effet, la chirurgie assistée par des robots est déjà une réalité en neuro-chirurgie où un robot peut exécuter une biopsie dans le tissu nerveux. Différents types de robots ont également été utilisés afin d'améliorer l'insertion de prothèse dans la hanche et d'exécuter une résection trans-urétrale de la prostate sous la supervision d'un urologue [81]. De plus, la micro-chirurgie exige de la pratique qui requiert un entraînement et un équipement spéciaux. Les tâches exécutées exigent alors des niveaux de dextérité se rapprochant au maximum des capacités humaines (sans assistance), impliquant des mouvements aussi petits que quelques microns et des forces aussi délicates que quelques grammes [82]. Les systèmes de télé-opération de mouvements graduels sont alors développés en utilisant généralement deux (02) poignets se soulevant magnétiquement, un maître et un esclave, qui partagent une base commune positionnée sur le site chirurgical par un robot de transport à mouvement-grossier [82]. Plus récemment, en chirurgie "laparoscopique", une camera "laparoscopique" et des instruments spéciaux insérés

dans la cavité abdominale à travers la paroi abdominale sont utilisés. Le robot manipule le "laparoscope" qui fournit une image vidéo en se déplaçant dans la cavité abdominale. Le robot agit comme un "camera-man" et remplace ainsi l'assistant de chirurgie [81]. Simultanément, les activités de recherche actuelles incluent le développement de stratégies de contrôle, pour un manipulateur médical miniature, basées à la fois sur le contrôle flou (en boucle fermée et pour la compensation de friction) et sur le contrôle neuronal. Ce dernier doit être capable de réduire les effets de blocage-glissement et d'améliorer la performance du système. Ce type de manipulateur est utilisé, dans la micro-chirurgie, comme un système de positionnement pour divers instruments chirurgicaux.

- **Applications Industrielles**

Plusieurs systèmes robotiques autonomes sont déjà utilisés dans des applications industrielles telles que la manutention de matériels, la soudure par points, la peinture par pulvérisation, et la construction. De plus, ces dernières années d'autres applications classiques sont en pleine expansion telles que l'assemblage, le désassemblage, l'exploitation minière, et l'agriculture. Le projet Panorama, "Helsinki University of Technology", par exemple a permis de développer une perception avancée et un système de navigation dédiés à l'automatisation des robots mobiles industriels dans des environnements d'extérieur partiellement structurés, tel le site réel de tests d'applications "Open Mine". Un autre exemple est le système Neptune qui est en fait un robot mobile utilisé pour inspecter à distance les réservoirs de stockage sur terre contenant des produits pétroliers. Neptune s'informe sur l'état de corrosion du fond et des murs latéraux selon les directives de "The American Petroleum Institute" [83]. Par ailleurs, les véhicules télé-contrôlés embarquant des robots manipulateurs, appelés manipulateurs mobiles, sont utilisés pour les activités de décontamination telles que la manutention de matériel radioactif, la restauration de sites de déchets, et le prélèvement d'échantillons de matériel à risques, voire dangereux. Le prélèvement d'échantillons de déchets à risques, utilisant un manipulateur mobile, a comme avantages l'amélioration de la sécurité de l'ouvrier et la réduction du coût par échantillon [84]. Une autre utilisation de robots coopératifs (manipulateur et robot mobile) est dans les applications de service telles que l'ouverture ou la fermeture d'une porte ainsi que pousser le bouton d'un élévateur [85].

- **Systèmes de Véhicules Intelligents sur Autoroutes**

L'industrie automobile a fourni un effort important dans le développement des systèmes de perception et de contrôle afin de rendre la voiture moderne plus sécurisante et plus facile à conduire. Ainsi, le récent développement dans les systèmes de véhicules intelligents sur autoroutes [2-4] p.ex., le programme de recherche "California PATH" et le projet "European PROMETHEUS", rend la conduite autonome un objectif atteignable dans un futur proche. Plus récemment, le but de la démonstration "Demo'97" a permis de confirmer que l'automatisation véhicule-autoroute est souhaitable et surtout réalisable [7, 9].

I. 2. 1. 2. Applications Sous-Marines

L'importance des véhicules sous-marins sans équipages embarquant des robots manipulateurs pour la recherche marine et le développement de sous-marins est en expansion rapide puisque leurs équivalents avec équipages sont beaucoup plus chers à développer et à entretenir. Ils sont également un élément clé pour l'exploitation des ressources marines [86]. En effet, quelques applications ont été développées telles que l'inspection et la maintenance des forages des sources pétrolières marines, l'immersion et l'inspection des câbles de communication, et les relevés géologiques et géophysiques des fonds marins. En fait, ce n'est que récemment qu'on a découvert, en utilisant des sous-marins avec équipages, qu'une grande quantité de dioxyde de carbone vient du fond-marin et des groupes extraordinaires d'organismes vivent dans des régions à 'trous' hydro-thermiques. Le 24 Mars 1995, le véhicule guidé à distance, Kaiko, a navigué dans la plus profonde région de l'océan Pacifique, Mariana Trough. Ce véhicule a plongé avec succès à une profondeur de 10058,4 mètres et a montré instantanément des scènes de la tranchée à travers une caméra vidéo. De tels outils permettent d'acquérir donc des connaissances de lieux mystérieux. Toutefois, l'utilisation de sous-marins avec équipages et de véhicules contrôlés à distance est limitée à de rares applications à cause des coûts opérationnels trop élevés, de la fatigue de l'opérateur, et des problèmes de sécurité. Malgré ces difficultés, la demande de robots sous-marins à haute technologie augmente de plus en plus et arrivera alors à des véhicules sous-marins complètement autonomes, spécialisés, et fiables.

1. 2. 1. 3. Applications Spatiales

Ces dernières années, plusieurs applications robotiques spatiales incluant des "rovers" planétaires avec des robots manipulateurs, des robots 'volants' tous usages à l'intérieur de stations spatiales, des robots manipulateurs pour l'industrie spatiale, et des robots pour la construction de stations spatiales sont fonctionnelles. Par exemple, le manipulateur à double bras [87] est défini comme une classe de robot manipulateur hautement redondant, multi-axes, multi-bras. Ce type de manipulateur peut être utilisé dans diverses applications telle qu'une grue spatiale pour l'assemblage de gros systèmes dans l'espace comme c'est le cas, pour la première fois dans l'histoire d'un vol spatial où un robot manipulateur multi-capteurs, embarqué sur un vaisseau spatial, a exécuté un nombre de tâches prototypes telles que l'ouverture et la fermeture de connecteurs, l'assemblage de structures à partir de pièces détachées, et la 'capture d'objets libres-flottants' [88]. Un autre exemple est le projet assigné à "Johnson Space Center" de la NASA qui consiste à intégrer les sous-systèmes de la Station Spatiale Internationale assurant l'exécution des tests fondamentaux de tâches sur orbite: tâches d'enlèvement/insertion d'unités remplaçables. Ces unités doivent être entretenues par le manipulateur habile (avec dextérité) Canadien à usages spécifiques, Canadarm [89], qui a engendré une nouvelle génération de manipulateurs: le "Mobile Servicing System" (MSS). Ce dernier joue un rôle essentiel dans la construction, la maintenance, et le fonctionnement de la Station Spatiale Internationale. Le MSS est un système autonome constitué de deux manipulateurs dont l'un est plus petit et plus habile. Un calculateur gère le fonctionnement de ces manipulateurs, démontrant des niveaux d'automatisation et de robotisation avancés. Par ailleurs, dans les missions spatiales, les systèmes robotiques interagissent avec des objets se déplaçant librement, sans contraintes. Ces objets peuvent présenter un mouvement complexe dû à une distribution asymétrique de leur masse. Par conséquent, il est nécessaire pour ce type de robots d'estimer et de prédire les mouvements tri-dimensionnels de ces objets [90]. Le robot manipulateur de la "Robotics Research Corporation", monté sur une plate-forme mobile est bien adapté à des tâches exigeant le positionnement et l'orientation d'une charge utile avec dextérité. Ce manipulateur mobile est capable d'exécuter un grand nombre de tâches exigeant la télé-opération et la manipulation autonome dans des environnements dynamiques non-structurés à la fois terrestres et spatiales [91]. Un autre système robotique conçu par le "Jet Propulsion Laboratory" est utilisé pour le prélèvement d'échantillons et le positionnement d'instruments. Deux exigences majeures de prélèvement d'échantillons terre/roche et d'images

panoramiques sont considérées par ce système, qui utilise deux types de manipulateurs, entièrement opérationnel sur le prototype "Rocky 7 Mars rover" [10].

Plus récemment, le "Pathfinder rover" de la NASA, appelé robot mobile Sojourner, a été le premier à être envoyé sur la planète Mars en Juillet 1997. En effet, Sojourner est conçu pour démontrer la viabilité de l'exploration de la planète Mars par des robots mobiles [10]. Au-delà de cette mission, plusieurs organisations internationales planifient de revisiter la Lune et la planète Mars, aussi bien que des excursions sur les astéroïdes et les comètes.

Dans un futur proche, les *VAI* vont exiger de hauts niveaux d'autonomie locale, incluant la capacité d'exécuter une navigation locale, d'identifier des régions d'intérêt scientifique potentiel, de réguler des ressources embarquées, et d'ordonnancer des activités, limitant ainsi l'intervention d'un contrôle terrestre.

I. 2. 2. Architecture, Mobilité, Perception, Traitement, et Communication

Afin d'exécuter avec succès les critères à satisfaire par les *VAI* pour être intelligents et réactifs, plusieurs exigences d'autonomie doivent être satisfaites et des composants intelligents sont nécessaires [6, 11].

I. 2. 2. 1. Exigences d'Autonomie

Les exigences d'autonomie ont été nécessitées par différentes spécificités des applications des *VAI*. Actuellement, ces spécificités nécessitent notamment des *VAI* une autonomie liée à leur masse et leur volume, à leur conception mécanique, à leur énergie, à la température ambiante, et à la communication.

- **Masse et Volume**

La masse et le volume sont deux des plus grands problèmes que rencontrent les *VAI* surtout dans les applications sous-marines et spatiales. Aussi, les exigences d'autonomie dépendent fortement de ces contraintes de masse et de volume.

- **Conception Mécanique**

La conception mécanique des *VAI* est le résultat d'une approche d'intégration considérant plusieurs critères relatifs à la perception, le contrôle, et la planification en plus de la conception de structures ainsi qu'aux contraintes mécaniques [92].

- **Energie**

Afin d'accomplir des tâches de manière autonome pour une période spécifique, l'énergie est une ressource très limitée pour les systèmes robotiques terrestres, sous-marins, et spatiaux. Ainsi, les *VAI* sont habituellement dotés d'un système d'énergie rechargeable, batteries embarquées de tailles appropriées.

- **Température**

Dans les applications spatiales, sous des conditions particulières, la conception thermique doit être prise en compte où la température peut varier sensiblement. Comme les composants mécaniques et électro-mécaniques doivent fonctionner sans régulation thermique à températures ambiantes, l'équipement électronique embarqué sensible à la température (comme les bancs d'essais scientifiques, les capteurs, et les caméras) doivent être placés dans un compartiment isolé thermiquement [5].

- **Gestion de la Communication**

Dans les applications sous-marines et spatiales, un système de gestion de données est habituellement nécessaire au transfert de données à partir des *VAI* aux stations terrestres de stockage et de traitement à l'aide d'un lien de communication à double sens [5, 75].

I. 2. 2. 2. Stratégies Sensorielles Intelligentes (Fusion de Données des Capteurs)

Etant donné que les *VAI* sont appelés à accomplir des tâches dans des environnements dynamiques incertains, les applications robotiques avancées nécessitent un flot continu d'informations riches et fiables qui doivent être rapidement recueillies à partir de capteurs.

Malheureusement, aucun capteur ne peut tout le temps garantir de délivrer des informations exactes de manière acceptable. Toutefois, les capteurs ont des caractéristiques assez différentes où les limites de fonctionnement d'un capteur ne correspondent pas souvent à celles d'un autre, proposant alors que deux (02) capteurs ou plus peuvent être utilisés avantageusement en combinaison. Ainsi, la redondance peut être utile pour les systèmes sensoriels robotiques autonomes comme dans le système sensoriel humain [6, 11, 93]. La fusion efficace de données à partir des différentes sources permettra au système de répondre immédiatement au monde réel. L'intégration multi-capteurs a alors reçu une attention particulière, ces dernières années, due à la disponibilité des capteurs, des moteurs, et des processeurs. Par exemple, des approches comportementales ont été proposées dans [94] comme une technique pour l'organisation de l'intégration multi-capteurs et le contrôle d'une tâche spécifique telles que la manipulation habile (avec dextérité) pour un robot manipulateur et l'évitement d'obstacles pour un robot mobile.

I. 2. 2. 3. Calculateurs Massivement Parallèles

Un composant intelligent important des *VAI* est le système de traitement qui doit être massivement parallèle pour des calculs en temps réel comme le calculateur "NCUBE/ten" et la bien connue "Connection Machine (CM)" [95]. Principalement, un tel composant est exigé par les approches émergentes, basées sur le *SC*, pour les systèmes de contrôle intelligent utilisant les *RN*, la *LF*, et les *AG* où le parallélisme est une propriété inhérente. De plus, la conception des systèmes utilisant un calculateur devrait supporter la conception de diverses fonctions des systèmes hautement autonomes, telles que le contrôle d'opérations, la tolérance aux défaillances (pannes, bruits, fautes), la communication, la planification, et l'ordonnancement.

I. 2. 3. Planification et Contrôle de la Navigation

L'objectif de la planification et du contrôle des *VAI* est l'amélioration de l'autonomie, de l'intelligence et des temps de réponse de ces véhicules qui impliquent trois aspects essentiels. En premier, ces véhicules doivent être capables d'exécuter efficacement des tâches telles que la reconnaissance, l'apprentissage, l'adaptation, le raisonnement et la prise de décisions, et l'action. Ils doivent également être capables de réduire la charge d'un opérateur par l'utilisation

d'un langage naturel et d'un sens commun de la connaissance afin de permettre une facilité dans la prise de décisions. Finalement, ils doivent accomplir des tâches à un niveau humain à l'aide de capacités d'*apprentissage*, d'*adaptation*, et de *généralisation* [63]. La différence entre l'intelligence de l'être humain et celle de la machine est due particulièrement aux concepts abstraits tels que la compréhension de l'intention, la généralisation, la prise de décisions dans des situations floues, etc. En effet, penser et raisonner en termes imprécis, non-quantitatifs et flous sont des aptitudes propres au cerveau humain. Ces aptitudes permettent aux humains de déchiffrer un texte mal écrit, de comprendre un discours bruité et de sélectionner l'information appropriée et utile à une prise de décisions [96].

Les chercheurs tentent alors de concevoir des *VAI* capables de naviguer intelligemment et de manière autonome dans un environnement réel. En d'autres termes, ils devraient posséder des capacités hautement intégrées: de perception de l'environnement, de planification et de contrôle de la navigation, d'apprentissage, d'adaptation, de raisonnement et de prise de décisions, d'action, et de traitement de données.

I. 2. 3. 1. Approches Classiques

Dans les approches classiques de planification et de contrôle de la navigation, il a été souvent supposé une connaissance à priori et exacte du véhicule, de ses capteurs, et plus encore de son environnement. Ceci a limité le type d'environnements où de tels véhicules peuvent accomplir des tâches, et par conséquent le type de tâches que ces véhicules peuvent résoudre. En effet, ces limitations sont dues principalement aux difficultés qu'ont les concepteurs à fournir des modèles exacts des véhicules et de leurs environnements. Même si une telle connaissance est disponible, la rendant accessible par ordinateur, le codage à la main de modèles explicites a souvent exigé des temps de programmation non raisonnables. Par ailleurs, atteindre une facilité de contrôle du calcul ("*tractability*") est un sérieux obstacle pour la conception de structures de contrôle pour les *VAI* dans des environnements dynamiques complexes [12, 58].

Les approches classiques de planification et de contrôle de la navigation ont été largement développées [14-24]. Nous citons notamment les approches basées sur le raisonnement géométrique, sur la configuration d'espaces, sur les heuristiques appliquées généralement aux environnements *statiques*, ainsi que celles basées sur le champ de potentiel artificiel appliquées aux environnements *statiques* et *dynamiques*.

- **Approches Basées sur le Raisonnement Géométrique**

Ces approches, basées sur le raisonnement géométrique, simplifient généralement le problème de planification et de contrôle en le ramenant souvent à celui d'un véhicule modélisé par un point de masse ou alors à une forme géométrique, dans un environnement *statique*, complètement connu. Ce problème est alors transformé en un problème purement géométrique de planification de chemins et de génération de trajectoires pour le contrôle [24]. Ainsi, ces approches ne peuvent être utilisées pour des applications *temps réel* vu la complexité des calculs qu'elles requièrent.

- **Approches Basées sur la Configuration d'Espaces**

Ces approches de planification et de contrôle, basées sur la configuration d'espaces, utilisent généralement un modèle de grille afin de représenter des environnements *statiques*, complètement connus [17, 19]. En plus du fait que les environnements sont supposés connus, ces approches prennent un temps considérable dans la division des noeuds ce qui les rend non appropriées pour des applications *temps réel*.

- **Approches Basées sur les Heuristiques**

Ces approches de planification et de contrôle, basées sur les heuristiques, sont appliquées à la fois aux environnements *statiques* et *dynamiques* [14, 18]. Bien que ces approches présentent plus de possibilités d'évaluation des points de collisions pour des obstacles dynamiques, elles sont exclues d'une éventuelle application *temps réel* puisqu'elles requièrent un temps de traitement considérable.

- **Approches Basées sur le Champ de Potentiel Artificiel**

Ces approches, basées sur le champ de potentiel artificiel développé par O. Khatib [97], ont été les plus utilisées pour la planification et le contrôle de la navigation [20–22]. Par contre, ces approches peuvent être envisagées pour une éventuelle application *temps réel* à condition de résoudre le problème majeur de 'serrure' c.-à-d., dans le cas d'équilibre des forces.

Ces approches classiques qu'elles soient basées sur le raisonnement géométrique, sur la configuration d'espaces, sur les heuristiques, ou sur le champ de potentiel artificiel, ne répondent pas aux exigences actuelles (*temps réel, autonomie, et intelligence*) des *VAI* dans leur navigation en environnements dynamiques. De plus, sur le plan stratégie de navigation, elles ne se basent pas sur la manière dont se comporte un être humain pour naviguer dans de tels environnements.

I. 2. 3. 2. Approches Actuelles

Dans les approches conventionnelles de planification et de contrôle, le raisonnement sur le comportement du système robotique et sur le comportement global désiré se base sur le formalisme de descriptions mathématiques caractérisé par un haut niveau d'abstraction. Dans le cas où le modèle mathématique du système robotique n'est pas disponible ou ne peut être obtenu qu'avec un grand effort et un coût élevé comme c'est le cas avec les *VAI*, la stratégie de contrôle devrait être basée sur une approche complètement différente. Au lieu d'une description mathématique, une description comportementale du processus est nécessaire se basant à la fois sur des expressions qualitatives et sur l'expérience humaine [98].

Afin de prendre alors en charge tous les problèmes en environnements dynamiques réels, ce qui représente le défi de ces dernières années, on doit introduire de nouvelles approches de planification et de contrôle, s'inspirant du comportement humain, appelées contrôles intelligents [6, 11, 98]. Ces nouvelles approches se basent essentiellement sur le *SC* introduisant particulièrement les *RN*, la *LF*, les *AG*, la *RA*, ainsi que les *SE*. Le *SC* est un domaine regroupant spécifiquement les approches visant à comprendre et à mimer le fonctionnement du cerveau humain. En effet, ces approches sont développées pour les *VAI* afin d'assurer: l'auto-calibration des véhicules et l'auto-réparation (dépannage ou remise en état), le mouvement habile (avec dextérité), l'auto-reproduction de structures intelligentes, l'apprentissage et l'évolution (apprentissage de machines, comportement adaptatif, et mécanismes d'évolution).

- **Réseaux de Neurones (*RN*)**

La capacité des êtres humains à se rappeler et à apprendre, plusieurs choses sans nécessairement oublier celles apprises dans le passé, représente la principale fonction utilisée

dans les *RN* pour mimer le cerveau humain [67, 71, 72]. Historiquement, l'intérêt porte aux *RN* provient du souhait de comprendre les principes menant en quelque sorte à la compréhension des fonctions du cerveau humain et de construire des machines capables d'exécuter des tâches complexes exigeant un traitement massivement parallèle [62, 72]. Essentiellement, les *RN* sont capables d'effectuer des tâches ayant rapport avec la connaissance telles que l'apprentissage, l'adaptation, la généralisation, et l'optimisation qui sont intéressantes pour la résolution de problèmes non-linéaires, incertains, incomplets, ou non-stationnaires.

- **Logique Floue (*LF*)**

L'écart entre l'intelligence des humains et celle des machines est due particulièrement aux concepts abstraits comme la compréhension de l'intention, la généralisation, la prise de décisions dans des situations floues, etc. En effet, penser et raisonner en termes inexacts, non-quantitatifs, et flous sont des aptitudes propres au cerveau humain. Reconnaître l'environnement et interagir avec ce dernier implique des appréciations abstraites des concepts naturels relatifs à la proximité, le degré de danger, etc. Le langage naturel impliqué doit être représenté alors par des sous-ensembles flous. Ces derniers impliquent des classes aux limites indéterminées dans lesquelles la transition d'appartenance à non-appartenance est plutôt graduelle que brusque. En effet, le raisonnement humain n'est pas basé sur la logique binaire classique, à deux (02) valeurs, puisque ce processus implique des vérités floues, des règles de déduction floue, etc. C'est la raison pour laquelle la *LF* est plus proche de la pensée humaine et du langage naturel que la logique classique. Fondamentalement, elle fournit des moyens efficaces pour saisir la nature approximée et inexacte de l'environnement réel [61, 99–102]. Ainsi, la *LF* peut être vue comme une tentative d'évolution des concepts mathématiques conventionnels précis vers ceux de prise de décisions de l'être humain [96, 99–101]. Les systèmes flous sont également présentés comme des systèmes parallèles à raisonnement associatif, générés à partir de la connaissance d'un expert ou des données d'entraînement [67].

- **Algorithmes Génétiques (*AG*)**

Les *AG* ont récemment émergé de l'étude du mécanisme d'évolution et sont des stratégies de recherche efficaces afin de trouver la solution optimale dans un large espace de paramètres [66, 70]. Ils ont été introduits pour la première fois en 1975 par J. Holland et ses étudiants à l'Université du Michigan et sont basés essentiellement sur le mécanisme de la

génétique naturelle et de la sélection naturelle [70].

Les approches évolutives ont été appliquées avec succès à une variété de problèmes difficiles. Ceux-ci incluent l'optimisation numérique, l'apprentissage de la machine, le contrôle optimal, la modélisation de processus par lesquels un être vivant acquiert des informations sur son environnement, les problèmes classiques de recherche (voyageur de commerce, 'havesac', transport, distribution, ordonnancement, partitionnement, etc.), conception en ingénierie, intégration de systèmes, robotique, traitement du signal, etc.

Les approches de calcul évolutif basées sur les *AG* sont des algorithmes stochastiques dont les méthodes de recherche modélisent certains phénomènes naturels tel que l'héritage génétique pour la 'survivable'. La popularité de ces méthodes peut être expliquée par leur simplicité et leur robustesse: l'idée générale de propagation de bonnes caractéristiques d'une solution potentielle dans des populations ultérieures est simple et indépendante des problèmes. Les meilleurs individus sont sélectionnés pour la reproduction et leur descendants remplacent les plus mauvais individus. Génération après génération la population s'améliore d'elle-même, en fin de compte l'algorithme converge et le meilleur individu représente une solution proche de l'optimum. Ainsi, à partir d'une population initiale de structures génétiques, des opérations d'héritage génétique basées sur la sélection, le croisement, et la mutation sont exécutées afin de générer une descendance qui rentre en compétition pour la 'survivable' des mieux adaptés afin de former la prochaine génération de structures de population [74].

Dans les *AG*, la connaissance est représentée par une population à M structures génétiques (ou individus) à l'instant t : A, B, \dots, M où chacune est représentée par plusieurs allèles [74]. Les structures sont le plus souvent choisies comme des chaînes de caractères telles que des chaînes binaires ou des nombres réels. Les opérations pour modéliser la génétique naturelle sont le plus souvent choisies comme étant un croisement et une mutation [74]. Les *AG* travaillent avec une population de solutions potentielles où chacune d'elles est codée par une chaîne ou chromosome. Ils utilisent les opérateurs basés sur la génétique qui peuvent être groupés en trois (03) catégories de base: les croisements, les mutations, et les gradients [103]. Ces opérateurs sont utilisés comme moyens de préservation des informations avantageuses avec l'objectif global d'arriver aux meilleures solutions au problème.

- **Résonance Adaptative (RA)**

La *RA* a été introduite par S. Grossberg en 1976 comme une théorie de traitement de l'information des processus par lesquels un être humain acquiert des informations sur son

environnement. Elle a depuis mené à un développement de séries de modèles combinant *RN* et *RA* pour un apprentissage *stable* et *rapide* et pour la reconnaissance de catégories (types ou classes) [74, 104, 105].

- **Systemes Experts (*SE*)**

Les *SE* sont des systèmes basés sur la connaissance représentant une approche symbolique du raisonnement. Ils sont utilisés communément pour les systèmes consultatifs dans lesquels la connaissance de l'expert, dans un domaine d'expertise spécifique, est mise à la disposition de l'utilisateur [73, 106]. En effet, un *SE* est un programme qui fonctionne dans un domaine restreint ayant rapport avec une connaissance spécialisée généralement en possession d'experts humains. De plus, c'est un système capable de tirer des conclusions sans avoir recours à toutes les informations [73, 107].

- **Systemes Intelligents (*SI*) et Systemes Hybrides Intelligents (*SHI*)**

Les approches actuelles de planification et de contrôle, impliquant les *RN*, la *LF*, les *AG*, la *RA*, et les *SE*, sont utilisées soit en *SI* se basant sur l'une de ces approches actuelles soit en *SHI* se basant sur l'une des différentes combinaisons de ces approches, exploitant leurs propriétés complémentaires [73]. En effet, l'intégration des *RN* et des *SE* s'est avérée une solution pour développer des applications utiles dans des environnements réels, et les *SHI* comprenant la *LF*, les *RN*, les *AG*, et les systèmes basés sur la connaissance reçoivent actuellement une attention de plus en plus grande [73]. D'autres *SHI* introduisant la *RA* [71] ont émergé tel que les *RN* basés sur la "ArtMap" Floue (*RNAMF*) [105].

Ces *SI* et *SHI* sont reconnus pour leur amélioration des capacités d'*apprentissage*, d'*adaptation*, et de *généralisation* relatives aux variations dans des environnements où l'information est *qualitative*, *imprécise*, *incertaine*, ou *incomplète*.

I. 3. LA PROBLEMATIQUE DE NAVIGUER INTELLIGEMMENT EN ENVIRONNEMENTS DYNAMIQUES

La reconnaissance, l'apprentissage, et l'adaptation pour les comportements intelligents de localisation de la cible (*CIB*) et d'évitement d'obstacles (*OBS*), le raisonnement et la prise de

décisions pour le comportement intelligent de prise de décisions (*DEC*) et l'exécution de l'action pour le comportement intelligent d'action (*ACT*) constituent les principaux problèmes de la navigation des *Véhicules Autonomes Intelligents (VAI)* [3, 4, 12, 13, 30, 34, 37, 42, 43, 108]. La problématique de naviguer intelligemment en environnements dynamiques se traduit alors par l'acquisition des comportements intelligents *CIB*, *OBS*, *DEC*, et *ACT* dont le synoptique de l'architecture globale est présenté dans la Figure I. 1.

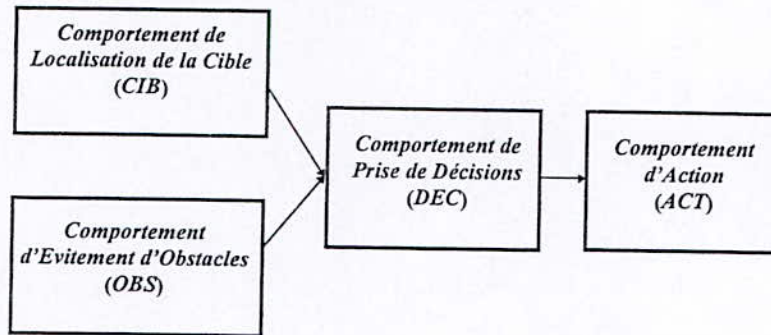


Figure I. 1. Architecture Globale des Comportements Intelligents de la Navigation.

I. 3. 1. Comportement Intelligent de Localisation de la Cible (*CIB*)

Le comportement *CIB* permet à chaque véhicule de localiser sa cible par rapport à lui-même ce qui lui permettra ensuite de se diriger vers cette cible.

I. 3. 2. Comportement Intelligent d'Évitement d'Obstacles (*OBS*)

Le comportement *OBS* permet à chaque véhicule de percevoir, de positionner, et de reconnaître les obstacles se trouvant dans un rayon correspondant à l'étendue des capteurs utilisés pour la détection d'obstacles ce qui lui permettra d'éviter ces obstacles. Notons que dans le cas des capteurs à *US*, pour une détection fiable, l'étendue utilisée est de deux (02) mètres [20, 109]. En effet, l'être humain perçoit les situations d'évitement d'obstacles comme des situations topologiques: couloir, chambres, virages à gauche, etc. Ainsi, dans la tentative de saisir le comportement *OBS* des êtres humains, il est nécessaire de structurer de tels environnements en situations topologiques [58].

I. 3. 3. Comportement Intelligent de Prise de Décisions (*DEC*)

Le comportement *DEC* permet à chaque véhicule de raisonner, à partir des résultats des comportements *CIB* et *OBS*, et de décider de l'action appropriée lui permettant à la fois de se diriger vers sa cible et d'éviter les éventuels obstacles rencontrés.

I. 3. 4. Comportement Intelligent d'Action (*ACT*)

Le comportement *ACT* permet à chaque véhicule d'exécuter l'action décidée par le comportement *DEC* c.-à-d., donner la commande appropriée aux moteurs du véhicule exécutant cette action. En fait, le système réalisant ce comportement devra générer des trajectoires (mouvements) lisses en contrôlant la vitesse des *VAI*.

I. 4. OBJECTIFS VISES

A travers ce Paragraphe, nous définissons les objectifs que nous nous sommes fixés notamment par rapport aux:

- types d'applications visés,
- types d'architecture, de mobilité, de perception, de traitement, et de communication des véhicules visés,
- types d'approches de planification et de contrôle de la navigation visés.

I. 4. 1. Applications

La définition d'un type d'applications des *VAI* implique automatiquement un certain type d'environnement spécifique dans lequel évoluent ces véhicules.

I. 4. 1. 1. Applications Visées

Les applications des *VAI* que nous visons concernent essentiellement les applications terrestres notamment le transport et la surveillance en environnements dynamiques, voire dangereux p.ex., les gares, les ports, les aéroports, les usines, et les sites dangereux.

I. 4. 1. 2. Environnements Dynamiques Partiellement Structurés

Les applications de transport et de surveillance impliquent des environnements d'intérieur ou d'extérieur dynamiques et généralement partiellement structurés. L'incertitude de la connaissance de l'environnement réel est le principal problème des *VAI* [24]. Cette incertitude est due au fait que la représentation des environnements se base essentiellement sur les systèmes de perception des véhicules qui doit être prise en compte dans la représentation de données.

Pour les stratégies de navigation que nous proposons dans le Chapitre II, le système de perception devra utiliser des capteurs à *UltraSons (US)* pour la détection d'obstacles nécessaire au comportement *OBS*. En effet, le problème d'une évaluation correcte de données bruitées et incorrectes pour l'interprétation des signaux des capteurs à *US* est souvent rencontré [35]. Ce problème sera pris en compte dans ces stratégies de navigation par les *RN* avec leurs caractéristiques inhérentes d'adaptation et de haute tolérance aux défaillances par rapport aux capteurs défectueux ou aux données bruitées (erronées) des capteurs [58]. Par ailleurs, l'intérêt dans la robotique mobile particulièrement dans les environnements tels que: les usines, les gares, les ports, et les aéroports a conduit à une description d'éventuels obstacles rencontrés dans de tels environnements. Ces environnements sont dynamiques partiellement structurés avec des obstacles statiques, dynamiques intelligents, et dynamiques non-intelligents.

En réalité, les obstacles statiques p.ex., Obs1, Obs2, Obs3, et Obs4 dans la Figure I. 2, où Véh: Véhicule, Obs: Obstacle, et Cib: Cible, de différentes formes représentent des murs, des piliers, des machines, des bureaux, des tables, des chaises, etc.

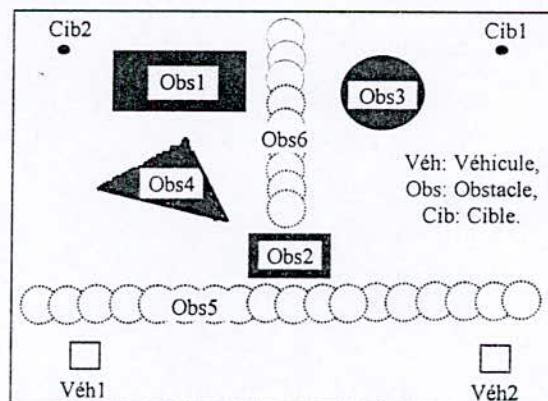


Figure I. 2. Environnement Dynamique Partiellement Structuré.

Les obstacles dynamiques intelligents (p.ex., Véh1 par rapport au Véh2 et réciproquement dans la Figure I. 2) représentent en réalité des *VAI* contrôlés par la même

stratégie de navigation où chacun considère l'autre comme un obstacle.

Les obstacles dynamiques non-intelligents, oscillant horizontalement (p.ex., Obs5 dans la Figure I. 2) ou verticalement (p.ex., Obs6 dans la Figure I. 2) entre deux points fixes, représentent en réalité des véhicules pré-programmés, téléguidés, ou guidés. Ce type d'obstacles ne possède ni système de perception ni approche d'évitement d'obstacles.

I. 4. 2. Architecture, Mobilité, Perception, Traitement, et Communication

L'architecture, la mobilité, la perception, le traitement, et la communication sont les principaux éléments qui définissent le type de véhicule envisagé.

I. 4. 2. 1. Véhicules Visés

Nous visons essentiellement les véhicules de type voiture ou tricycle en environnements dynamiques partiellement structurés. Notons la similarité par rapport aux aspects cinématique et dynamique dans ce type de véhicules. En vue de futures implémentations, une attention particulière est portée vers le véhicule de type tricycle: "Autonomous Guided Vehicle" (AGV) développé au Laboratoire de Robotique et d'Intelligence Artificielle (LRIA) du Centre de Développement des Technologies Avancées (CDTA).

I. 4. 2. 2. "Autonomous Guided Vehicle" (AGV)

L'AGV est un véhicule mobile de type tricycle dont une description est donné en ANNEXE A. Trois (03) phases ont été planifiées dans la conception et le développement de l'AGV: la phase de guidage par radio-commande (télé-commande), la phase de vidéo guidage (contrôle à distance à partir de caméra), et la phase autonome grâce à une camera embarquée.

L'AGV possède une (01) roue avant directionnelle et de traction ainsi que deux (02) roues arrières passives (porteuses), embarquant une carte de puissance, une carte de commande des moteurs, et une carte de gestion de douze (12) capteurs à infrarouges ainsi que huit (08) capteurs à US [110]. Les directions de l'AGV sont possibles entre -45° et $+45^\circ$ par rapport à son axe principal.

La configuration de l'AGV [111-113] est représentée dans la Figure I. 3.

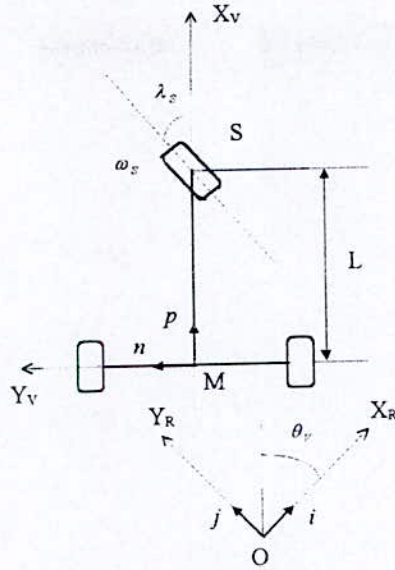


Figure I. 3. Configuration de l'“Autonomous Guided Vehicle” (AGV).

où $R(O, i(X_R), j(Y_R))$: repère absolu de référence,

M: milieu de l'essieu arrière,

$V(M, p(X_V), n(Y_V))$: repère local du véhicule,

S: point de direction,

L: distance ($L = 115$ cm) entre le milieu de l'essieu arrière M et le point de direction S,

θ_V : angle entre l'axe principal de l'AGV et l'axe des X_R du repère $R(O, i, j)$,

λ_S : angle de direction de la roue avant,

ω_S : vitesse tangentielle de la roue avant.

Les vecteurs unités p et n , qui sont respectivement parallèle et perpendiculaire à l'axe principal de l'AGV, peuvent être exprimés dans le repère R comme suit:

$$\begin{aligned} p &= \cos\theta_V i + \sin\theta_V j \\ n &= -\sin\theta_V i + \cos\theta_V j \end{aligned} \quad (I. 1)$$

Les transformations de coordonnées sont alors données:

$$\begin{bmatrix} \frac{d}{dt} X_V \\ \frac{d}{dt} Y_V \\ \frac{d}{dt} \theta_V \end{bmatrix} = \begin{bmatrix} \cos \lambda_S \\ 0 \\ (\sin \lambda_S) / L \end{bmatrix} \omega_S \quad (I. 2)$$

$$\begin{bmatrix} \frac{d}{dt} X_R \\ \frac{d}{dt} Y_R \end{bmatrix} = \begin{bmatrix} \cos \theta_S & 0 \\ 0 & \sin \theta_V \end{bmatrix} \begin{bmatrix} \frac{d}{dt} X_V \\ \frac{d}{dt} Y_V \end{bmatrix} \quad (I. 3)$$

I. 4. 3. Planification et Contrôle de la Navigation

Par rapport aux approches classiques, les approches actuelles de planification et de contrôle de la navigation ont supprimé la supposition d'une connaissance parfaite de l'environnement. Elles ont également supprimé la représentation *explicite* de la connaissance pour une représentation *implicite* basée sur l'*acquisition de comportements intelligents* qui habilite le véhicule à interagir efficacement avec son environnement [13]. Par conséquent, les *VAI* se heurtent à des environnements moins prévisibles et plus complexes, ils ont à s'orienter, à explorer leurs environnements de manière autonome, à s'auto-dépanner, et à exécuter des familles complètes de tâches en *temps réel*. De plus, si les véhicules manquent d'une connaissance initiale les concernant eux-mêmes et leurs environnements, l'*apprentissage* devient inévitable. Cet apprentissage se réfère à une variété d'algorithmes caractérisés par leurs capacités de remplacer une connaissance de l'environnement manquante ou incorrecte par l'*expérimentation*, l'*observation*, et la *généralisation*. En d'autres termes, l'apprentissage de véhicules compte sur l'interaction avec l'environnement pour extraire l'information utile afin d'atteindre un but donné [12].

Ainsi, les *VAI* doivent être capables de comprendre la structure de leurs environnements pour atteindre leurs cibles sans collisions avec d'éventuels obstacles rencontrés. Ils doivent donc avoir la capacité de réaliser des comportements intelligents de localisation de la cible (*CIB*), d'évitement d'obstacles (*OBS*), de prise de décisions (*DEC*), et d'action (*ACT*). Ces comportements intelligents font appel à la reconnaissance (traitement de données imprécises issues de capteurs, construction d'une base de connaissances, et établissement d'une carte de

l'environnement), l'apprentissage, l'adaptation, le raisonnement et la prise de décisions, et l'action. En fait, ces derniers constituent les principaux problèmes de la navigation des *VAI* en environnements dynamiques.

Afin de résoudre ces problèmes, les approches classiques [14–24], ont rapidement été remplacées par les approches actuelles basées sur la collection de méthodologies appelée “Soft Computing” (*SC*) impliquant les Réseaux de Neurones (*RN*), la Logique Floue (*LF*), les Algorithmes Génétiques (*AG*), la Résonance Adaptative (*RA*), et les Systèmes Experts (*SE*) utilisés individuellement en Systèmes Intelligents (*SI*) ou en différentes combinaisons en Systèmes Hybrides Intelligents (*SHI*) [12, 13, 25–60]. En fait, d'une part, le but du *SC* est de faire évoluer le comportement des *VAI* vers celui de l'être humain dans la reconnaissance, l'apprentissage, l'adaptation, le raisonnement et la prise de décisions, et l'action [6, 11, 52, 61–74]. D'autre part, ces *SI* et *SHI* sont reconnus pour leur amélioration des capacités d'apprentissage, d'adaptation, et de généralisation relatives aux variations dans des environnements où l'information est qualitative, imprécise, incertaine, ou incomplète.

I. 4. 3. 1. Planification et Contrôle Visés

Nous visons une planification et un contrôle de la navigation qui puissent assurer aux *VAI*, particulièrement à l'*AGV*, plus d'autonomie et plus d'intelligence avec des capacités de traitement *temps réel*. Par conséquent, les comportements intelligents de localisation de la cible (*CIB*), d'évitement d'obstacles (*OBS*), de prise de décisions (*DEC*), et d'action (*ACT*) doivent se baser sur les approches actuelles du “Soft Computing” (*SC*).

I. 4. 3. 2. Approches Basées sur le “Soft Computing” (*SC*)

L'objectif principal des *VAI* est relatif au développement de la théorie et des applications des systèmes robotiques capables d'un certain degré d'*auto-suffisance*. Le centre d'intérêt est orienté vers la capacité à naviguer et à être *auto-suffisant* dans des environnements dynamiques. Actuellement, les chercheurs ont à leur disposition des technologies avancées concernant le “hardware”, le “software”, et les capteurs afin de construire de tels *VAI*. En effet, ils sont particulièrement en possession d'approches puissantes de traitement tels que les *RN*, la *LF*, les *AG*, la *RA*, et les *SE* qui sont plus efficaces dans la conception et le développement des *VAI* que celles basées sur la logique des prédicats de l'intelligence

artificielle traditionnelle. Ces approches proviennent de la collection de méthodologies appelée *SC*, dont le principe directeur est l'exploitation de la tolérance à l'imprécision, à l'incertitude, et à la vérité partielle dans le but de réaliser une facilité de contrôle du calcul ("*tractability*"), une *robustesse*, un faible coût de la solution, et un meilleur rapport avec la réalité. ♣

I. 5. SYNTHÈSE SUR LES APPROCHES BASEES SUR LE "SOFT COMPUTING" (SC)

Dans cette synthèse sur les approches actuelles de planification et de contrôle de la navigation, basées sur le *SC*, un intérêt particulier est porté aux Systèmes Intelligents (*SI*) basés soit sur les *RN* soit sur la *LF* ainsi qu'aux Systèmes Hybrides Intelligents (*SHI*) basés sur les combinaisons *RN-AG*, *RN-LF*, *RN-LF-RA*, et *SE-RN-LF*.

I. 5. 1. Systèmes Intelligents (*SI*): Réseaux de Neurones (*RN*)

L'approche des *SI* introduisant les *RN*, a été activement étudiée et appliquée aux domaines tels que la planification et le contrôle de la navigation des *VAI* [30, 31, 34–36, 38, 41–43, 46, 51, 58]. Ces approches basées sur les *RN* exploitent les caractéristiques des *RN* telles que:

- la représentation *implicite* de la connaissance,
- l'*apprentissage* et la *généralisation* à partir d'expériences (à partir d'exemples),
- la *robustesse* relative à la *tolérance aux défaillances* (pannes, bruits, ou erreurs) par rapport aux capteurs défectueux ou aux données bruitées (erronées) des capteurs,
- le traitement massivement parallèle (*temps réel*).

D'où l'intérêt que nous portons particulièrement aux *SI* suivants:

- *SI* utilisant les *RN* multicouches par *apprentissage supervisé* par la *RPG* [62, 64, 67, 69, 71, 72, 74],
- *SI* utilisant les *RN* multicouches par *apprentissage supervisé* par le *REE* [30].

I. 5. 2. Systèmes Intelligents (*SI*): Logique Floue (*LF*)

L'approche des *SI* introduisant la *LF*, a été également activement étudiée et appliquée à la planification et le contrôle de la navigation des *VAI* [25–29, 32, 37, 39, 40, 43, 50]. Ces

approches exploitent les caractéristiques de la *LF* telles que:

- la représentation *explicite* de la connaissance,
- l'*adaptation* et la *généralisation* à partir de règles floues,
- la capacité à saisir la nature *approximative* et *imprécise* de l'environnement réel,
- le traitement parallèle (règles à mémoires associatives floues).

D'où l'intérêt que nous portons particulièrement aux *SI* utilisant la *LF* par *adaptation* 'supervisée' traduite par l'utilisation des règles floues données par un expert humain (superviseur), par la fuzzification basée sur des fonctions d'appartenance triangulaires voire trapézoïdales, par l'inférence utilisant les opérateurs MAX-MIN, et par la défuzzification par l'opérateur MAX [29, 61, 67, 99-102].

I. 5. 3. Systèmes Hybrides Intelligents (*SHI*): Réseaux de Neurones (*RN*) et Algorithmes Génétiques (*AG*)

Les *AG* sont caractérisés par une performance *robuste* par l'utilisation des propriétés d'*adaptation* et de 'survivance' imitant celle de l'évolution biologique. Il a été prouvé théoriquement et empiriquement qu'ils fournissent une recherche robuste dans des espaces complexes, vu qu'ils ne sont pas limités par des suppositions restrictives de continuité, d'existence de dérivées, d'unimodalité, etc. [70, 74]. Ils ont été largement appliqués dans le domaine scientifique, d'ingénierie à savoir: la reconnaissance de formes (types), l'optimisation de fonctions, l'ordonnancement, l'apprentissage, la classification, le contrôle de processus, la conception en ingénierie, etc. [74].

D'une part, l'apprentissage supervisé le plus répandu des *RN* multicouches est celui par la *RPG* [64, 71, 72, 74]. En fait, la *RPG* est sensible aux paramètres tels que le taux d'apprentissage et le "momentum" en plus du fait qu'elle se base sur le principe de la descente du gradient qui souffre du problème des minimums *locaux* [114]. D'autres part, les deux (02) techniques les plus importantes de l'apprentissage des machines, basé sur la génétique, sont les systèmes classificateurs et la programmation génétique [70]. En fait, les *AG* peuvent être utilisés afin d'entraîner un *RN* multicouches [103, 114]. Cet entraînement peut être plus *robuste*, n'impliquant pas le principe de descente du gradient, avec moins de paramètres à ajuster comparé à celui par la *RPG* [114]. Bien que la performance des *AG* ne dépend pas des paramètres tels que le taux d'apprentissage et le "momentum", le problème de lenteur leur est inhérent.

Les approches de planification et de contrôle de la navigation, basées sur les *AG*, sont caractérisées par la *généralité*, la *flexibilité*, et l'*adaptabilité*. Elles ont souvent été appliquées aux environnements *statiques*. Récemment, ces approches se sont appliquées afin de planifier et contrôler la navigation dans des environnements *dynamiques* à l'aide d'une implémentation parallèle [45, 48, 56].

D'où l'intérêt que nous portons particulièrement aux *SHI* suivants:

- *SHI* utilisant les *RN* multicouches par *apprentissage supervisé* par les *AG* [70, 71, 103, 114, 115],

- *SHI* utilisant les *RN* multicouches par *apprentissage supervisé* par les *AG-RPG* [71, 114].

I. 5. 4. Systèmes Hybrides Intelligents (*SHI*): Réseaux de Neurones (*RN*) et Logique Floue (*LF*)

Actuellement, la plupart des approches de planification et de contrôle de la navigation des *VAI* se basent essentiellement sur la combinaison *particulière RN-LF*. La particularité de cette combinaison est due au fait que les *RN* et la *LF* sont effectivement des *technologies intelligentes*. Ce fait nous a conduit à effectuer une étude de synthèse sur cette combinaison beaucoup plus approfondie que celles effectuées pour les autres combinaisons.

I. 5. 4. 1. Propriétés des *RN* et de la *LF*

En général, la *LF* et les *RN* sont bien appropriés pour le traitement de données imprécises et les systèmes basés sur la connaissance [43, 61, 63, 67, 71, 72, 74, 116, 117]. Ceci est dû à leurs propriétés telles que le parallélisme, la classification, l'apprentissage, l'adaptation, la généralisation, l'aide à la décision, l'optimisation, et la tolérance aux défaillances (mémoire distribuée), présentées dans le Tableau B. 1 (ANNEXE B).

I. 5. 4. 2. Analyse Critique sur les *RN* et sur la *LF*

☛ L'intérêt d'établir des relations entre les *RN* et la *LF* est en partie relatif à des notions telles que: *l'acquisition de la connaissance* et la *généralisation*. En effet, L. A. Zadeh et P. J. Werbos ont affirmé que ces relations sont fondamentalement *complémentaires* plutôt

qu'équivalentes ou compétitives [63, 68]. Cet aspect de complémentarité a motivé leur fusion permettant un nouveau concept associant à la fois les connaissances *implicite* et *explicite*. En fait, les systèmes basés sur les *RN* sont capables d'apprendre, alors que ceux basés sur la *LF* peuvent exprimer la connaissance mais ne possèdent pas cette capacité (ils s'adaptent plutôt). Les *RN* expriment alors la connaissance *implicitement* dans les poids, alors que la connaissance est exprimée *explicitement* dans les règles par la *LF*.

Nous présentons les propriétés complémentaires dans le Tableau B. 2 (ANNEXE B), alors que celles équivalentes sont globalement relatives aux différents comportements similaires à ceux de l'être humain et aux domaines d'applications [61, 63, 67, 71, 72, 74, 116, 117].

Par ailleurs, il est intéressant d'analyser le point de vue des *RN* par rapport à la *LF* et réciproquement. En fait, du point de vue neuronal, la *LF* évite l'utilisation de seuils, introduit des transitions graduelles, permet une initialisation correcte des poids, et peut accélérer l'apprentissage. Du point de vue flou, les *RN* permettent l'identification à la fois des parties antécédent et conclusion des règles. De plus, la connaissance acquise, après apprentissage, dans un *RN* peut être traduite sous forme de règles floues. Par conséquent, différentes approches de fusion exploitant toutes les analogies sous-jacentes entre les *RN* et la *LF* ont été établies [43, 63, 73, 116].

I. 5. 4. 3. Synthèse sur la Navigation Basée sur les *RN* et sur la *LF*

La navigation est en relation directe avec les problèmes de reconnaissance de l'environnement (traitement de données imprécises), de planification et contrôle, de prise de décisions et d'action.

- **Traitement de Données Imprécises**

Les données imprécises sont traitées afin de prendre connaissance de l'environnement et d'établir une carte de l'environnement. La difficulté dans l'établissement de cette carte réside dans la représentation de la connaissance.

L'utilisation de la *LF* ou des *RN* pour traiter des données imprécises issues à partir de l'environnement et des capteurs a souvent été plus efficace que les techniques déterministes traditionnelles [25, 29, 34, 35]. Ceci est dû d'une part à la représentation de la *LF* à travers les fonctions d'appartenance et leur discrétisation en un vocabulaire linguistique. D'autres part,

quand les données sensorielles sont incorrectes ou hautement bruitées, l'utilisation des *RN* est nécessaire puisqu'ils sont hautement tolérants aux défaillances.

- **Planification et Contrôle**

La planification basée sur les *RN* proposée dans [31] a été formulée comme un problème de classification où les types de classes sont cartographiées sur l'ensemble des mouvements manoeuvrables du robot. De plus, pour déterminer un chemin sans danger dans des environnements cartographiés en grille, le traitement distribué et parallèle, en temps réel, d'un modèle basé sur les *RN* a prouvé son efficacité [36]. L'approche développée dans [34] se basant également sur un système *neuronal* de contrôle fournit au robot "Mobile Adaptive Visual Navigator (MAVIN)" la capacité à s'adapter à son environnement et à apprendre à partir d'expériences. Ce système utilise les *RN* pour la perception, le contrôle de mouvements, l'apprentissage de types (formes) et la reconnaissance d'objets, les associations d'objets, les actions comportementales, etc. Les *RN* sont également utilisés pour contrôler le comportement de repérage et de poursuite d'un véhicule autonome dans [35], alors qu'ils sont utilisés pour contrôler les comportements *CIB* et *OBS* dans un environnement dynamique dans [41]. De plus, le système NEURO-NAV développé dans [42] pour des véhicules d'intérieur se base sur les *RN* qui sont entraînés afin d'interpréter les informations visuelles (par l'image) et d'exécuter des comportements de la navigation tels que le suivi de mur (dans un couloir) et la détection de points de repérage (au sol).

Alors que, dans les systèmes basés sur la connaissance, rapidement il a été reconnu que la détermination des règles du contrôle flou est l'un des problèmes clés à résoudre [26–28]. Les approches développées pour déterminer les règles fondamentales du contrôle flou sont basées essentiellement sur l'expérience d'un opérateur, la connaissance d'un ingénieur de contrôle ou sur la modélisation floue des actions de contrôle d'un opérateur. Une particularité des travaux basés sur des règles floues, peut être leurs capacités de classer les règles floues selon le degré de danger ou de sécurité [25, 32], selon les règles de reconnaissance et les règles de contrôle [37] et selon les règles d'évitement d'obstacles et les règles de guidage [39].

- **Prise de Décisions et Action**

- Globalement, les approches floues et neuronales de planification et de contrôle de la navigation se basent essentiellement sur la manière avec laquelle les êtres humains raisonnent,

prennent leurs propres décisions (actions appropriées), et exécutent ces actions dans des environnements dynamiques, incertains, et imprécis [25–32, 34–39, 41, 42, 46, 50, 51, 58].

I. 5. 4. 4. Analyse Critique sur la Navigation Basée sur les RN et sur la LF

L'utilisation des *RN* pour résoudre les problèmes de planification et de contrôle de la navigation s'avère une approche *intéressante* et *nécessaire*, si les critères de classification ou les règles de généralisation sont *inconnus*, puisqu'ils sont capables d'apprendre et de généraliser à partir d'exemples sans connaissance de règles, et leur capacité d'adaptation est d'un grand intérêt pour les problèmes évolutifs. Si par contre les critères de classification ou les règles de généralisation sont *connus*, l'utilisation de la *LF* s'avère une approche *intéressante* et *efficace* ; p.ex., dans le cas où les fonctions d'appartenance et les règles de classification des obstacles par leurs degrés de sécurité sont bien définies (connues).

La combinaison de la *LF* et des *RN* est reconnue comme une amélioration, pour les *SI*, des capacités d'*apprentissage* et d'*adaptation* dans des environnements où l'information est *qualitative*, *imprécise*, *incertaine* ou *incomplète*. La plupart des systèmes *neuro-flous*, actuellement en utilisation, sont fondamentalement basés sur des règles floues induites par les *RN* à partir d'observations. Cependant, une autre tendance est devenue perceptible: l'utilisation de la *LF* dans la conception des *RN*. En particulier, il apparaît que les capacités des *RN* pourraient être significativement rehaussées par la capacité de traitement de l'information floue [63]. En effet, la tendance dans la direction de l'utilisation de la *LF* dans la conception des *RN* conduisant aux *RNF* est devenue très perceptible [47, 73, 118–120]. Elle combine la force du *raisonnement* et la capacité d'*adaptation* de la *LF* et la puissance des capacités d'*apprentissage* et de *généralisation* des *RN*.

D'où l'intérêt que nous portons particulièrement aux *SHI* utilisant les *RNF* [47, 49, 54, 60, 73, 118–120] par *apprentissage supervisé* par la *RPG*.

I. 5. 5. Systèmes Hybrides Intelligents (SHI): Réseaux de Neurones (RN), Logique Floue (LF), et Résonance Adaptative (RA)

La *RA* a mené au développement de collections de modèles combinant *RN* et *RA* (*RA1*: classer des types d'entrée binaires, *RA2*: classer des types d'entrée analogiques ou binaires, et

RA3 peut prendre en charge une recherche parallèle ou un test d'hypothèse des codes distribués de reconnaissance en une hiérarchie multi-niveaux de réseau) pour l'apprentissage de catégories et la reconnaissance de types [64, 104].

Le modèle *RA1* a été utilisé afin de concevoir une architecture de réseaux hiérarchiques "*ArtMap*" qui peut rapidement auto-organiser des correspondances de catégories stables entre des vecteurs d'entrée à i -dimensions et des vecteurs de sortie à n -dimensions [104]. Le "*Map*" se rapporte à l'action que le *RN* exécute: il apprend une correspondance à partir d'une entrée à une sortie. La "*ArtMap*" a été utilisée à l'origine pour apprendre des correspondances entre des vecteurs d'entrée binaires et des vecteurs de sortie binaires. Le modèle de la "*Art*" Floue développé dans [104] généralise la *RA1* à l'apprentissage et reconnaissance stable de catégories en réponse à la fois aux types d'entrée analogiques et binaires. Ce modèle a été incorporé en une architecture "*ArtMap*" Floue qui est capable d'un apprentissage stable et rapide de reconnaissance de catégories en réponse aux séquences arbitraires de types d'entrée analogiques ou binaires. L'architecture "*ArtMap*" Floue réalise ainsi une synthèse de la *LF* et les *RN* utilisant la *RA* par l'exploitation d'une proche similarité formelle entre les calculs des sous-ensembles flous et du choix de catégories par la *RA*, la résonance, et l'apprentissage [105].

L'approche développée dans [44] utilise les *RNAMF* pour une classification d'espaces, à partir de la perception, pour le comportement *OBS*. Les *RNAMF* ont aussi été utilisés dans un contrôleur de planification de mouvements pour le suivi de trajectoires afin de reconnaître les images de caméra [53] et apprendre un positionnement qualitatif d'un robot mobile d'intérieur équipé de capteurs à UltraSons (*US*) [55]. Dans ces approches, les *RNAMF* ont été utilisés pour leurs capacité de *généralisation*, leur *robustesse*, et leur *apprentissage rapide et stable* [57, 59, 74, 104, 105, 121, 122]. En fait, les *RNAMF* ont des capacités significatives de généralisation de la connaissance. Par ailleurs, les capteurs à *US* et les images de caméra sont très utilisés pour le comportement *OBS*, mais leurs signaux sont souvent bruités donnant des données incorrectes (erronées). Les approches basées sur les *RNAMF*, avec leurs caractéristiques inhérentes d'*adaptation* et de *grande tolérance aux défaillances*, prennent efficacement en charge ce problème rendant ainsi ces approches effectivement *robustes*.

D'où l'intérêt que nous portons particulièrement aux *SHI* utilisant les *RNAMF* [57, 59, 74, 104, 105] par *apprentissage supervisé* par la *AMFS* [74, 121].

I. 5. 6. Systèmes Hybrides Intelligents (*SHI*): Systèmes Experts (*SE*), Réseaux de Neurones (*RN*), et Logique Floue (*LF*)

La première intégration, et la plus avancée, des technologies intelligentes est la combinaison (hybride) des *RN* et des *SE*. La poussée majeure de ce type de *SHI* est de synthétiser la capacité des *SE* à saisir la connaissance d'un expert d'un domaine donné dans un système basé sur l'inférence avec la puissance des *RN* entraînés à partir d'exemples [73]. Par ailleurs, la base de connaissances est la source majeure d'informations incertaines dans les *SE*, les causes peuvent être des informations non-fiables, des langages de description imprécis, une inférence avec des informations incomplètes et une pauvre combinaison de la connaissance à partir de différents experts [73, 107]. Ainsi, les *SE* flous offrent l'opportunité de produire des systèmes avec une meilleure connaissance, se focalisant sur le besoin de prendre en charge les incertitudes, les ambiguïtés et les imprécisions associées aux problèmes en environnements réels [73].

Dans les *SENF* (*SHI* combinant *SE*, *RN*, et *LF*), les *RNF* sont utilisés pour focaliser l'inférence, guider les recherches, et exécuter l'appariement des types alors que les *SE* sont utilisés pour interpréter les résultats des *RNF*, fournir des facilités d'explication, et établir une connexion entre réseaux. Cette combinaison offre une flexibilité de conception et une intégration robuste et a l'avantage de réduire les communications et de l'amélioration de la durée d'exploitation. Ainsi, plusieurs *SHI* basés sur cette combinaison ont été développés [106, 107, 123–125] et vont probablement être omniprésents dans la résolution des problèmes en environnements réels, particulièrement pour l'acquisition de comportements *intelligents* des *VAI* [54, 60].

D'où l'intérêt que nous portons particulièrement aux *SHI* utilisant les *SENF* [54, 60, 107, 123–125].

I. 6. CONCLUSION

Les récents développements dans les exigences d'autonomie, les composants intelligents, les systèmes multi-véhicules, et les calculateurs massivement parallèles rendent le futur des *VAI* très prometteur dans les applications terrestres, sous-marines, et spatiales. En effet, par l'introduction de la planification et du contrôle caractérisés par l'émulation des systèmes biologiques, les *VAI* vont être de plus en plus *autonomes* et *intelligents* avec des capacités de

traitement *temps réel*. Ainsi, les approches de planification et de contrôle de la navigation, basées sur le *SC*, sont donc nécessaires afin de faire évoluer le comportement de ces *VAI* vers celui de l'être humain notamment dans les comportements *CIB*, *OBS*, *DEC*, et *ACT*, en environnements dynamiques.

A partir de la synthèse que nous avons effectuée, dans ce Chapitre, nous nous sommes intéressés à l'acquisition de ces comportements intelligents par *apprentissage* ou/et *adaptation* notamment par les *SI* et *SHI* suivants:

- les Réseaux de Neurones (*RN*) entraînés par l'*apprentissage supervisé* par la Rétro-Propagation du Gradient (*RPG*): *RN/RPG*,
- la Logique Floue (*LF*),
- les *RN* entraînés par l'*apprentissage supervisé* par les *AG*: *RN/AG*,
- les *RN* entraînés par l'*apprentissage supervisé* par les *AG* suivis de la *RPG*: *RN/AG-RPG*,
- les Réseaux de Neurones Flous (*RNF*) entraînés par la *RPG*: *RNF/RPG*,
- les *RN* basés sur la "ArtMap" Floue (*RNAMF*) entraînés par l'*apprentissage supervisé* par la "ArtMap" Floue Simplifiée (*AMFS*): *RNAMF/AMFS*,
- les *RN* entraînés par l'*apprentissage supervisé* par Renforcement par Essai et Erreur (*REE*): *RN/REE*,
- les Systèmes Experts Neuro-Flous (*SENF*).

Ces *SI* et *SHI* devront fournir aux *VAI* des capacités afin de naviguer intelligemment dans des environnements dynamiques partiellement structurés. En fait, l'acquisition des comportements intelligents par *apprentissage* ou/et *adaptation* va aider les *VAI* à réagir avec de nouvelles actions à de nouveaux environnements.

Chapitre II

PLANIFICATION ET CONTROLE DE LA NAVIGATION: Vers une Autonomie et une Intelligence des Véhicules

II. 1. INTRODUCTION

L'objectif principal de ce Chapitre est de développer des approches, basées sur le "Soft Computing" (SC), de planification et de contrôle de la navigation capables de fournir aux Véhicule Autonomes Intelligents (VAI) plus d'autonomie et plus d'intelligence avec des capacités de traitement *temps réel*. Ce qui se traduit par le développement de Systèmes Intelligents (SI) et Systèmes Hybrides Intelligents (SHI) pour l'acquisition de comportements intelligents de localisation de la cible (CIB), d'évitement d'obstacles (OBS), de prise de décisions (DEC), et d'action (ACT).

Durant la navigation, les Véhicule Autonomes Intelligents (VAI) doivent construire une carte interne implicite (c.-à-d., cible, obstacles, et espaces libres) à partir des capteurs, mettre à jour cette carte et l'utiliser afin de planifier et contrôler intelligemment leur navigation. Afin de naviguer dans des environnements dynamiques partiellement structurés, les VAI doivent atteindre leurs cibles sans collisions avec les éventuels obstacles rencontrés c.-à-d., ils doivent avoir la capacité à réaliser les comportements CIB et OBS. Ensuite, à partir des résultats de ces comportements il doivent avoir la capacité à réaliser le comportement DEC leur permettant le choix de l'action appropriée. Finalement, ils doivent avoir la capacité à réaliser le comportement ACT.

Dans ce Chapitre, deux (02) stratégies de navigation, NAV_1 et NAV_2, sont proposées aux VAI, dans § II. 2. Ensuite, différents SI et SHI sont proposés pour l'acquisition, par *apprentissage* ou/et par *adaptation*, des comportements CIB, OBS, DEC, et ACT, dans § II. 3, § II. 4, et § II. 5. Enfin, les SHI_A_NAV_1_Généralisée et SHI_B_NAV_1_Généralisée ainsi que le SHI_SENF_NAV_2_Généralisée sont proposés, dans § II. 6, pour la planification et le contrôle de la navigation des VAI.

II. 2. STRATEGIES DE NAVIGATION, NAV_1 ET NAV_2, PROPOSEES

Nous proposons deux (02) stratégies de navigation, en versions simplifiées (NAV_1_Simplifiée et NAV_2_Simplifiée) et en versions généralisées (NAV_1_Généralisée et NAV_2_Généralisée) où:

- NAV_1_Généralisée est la stratégie que nous généralisons et proposons à partir de la stratégie NAV_1_Simplifiée initialement développée par E. Sorouchyari [30],

- *NAV_2_Généralisée* est la stratégie que nous généralisons et proposons à partir de la stratégie *NAV_2_Simplifiée* initialement développée par Y. Maeda [32].

Notons que nous utilisons *NAV_1_Simplifiée* comme un banc d'essais simple à travers lequel nous appliquons différentes approches de planification et de contrôle, basées sur le *SC*, uniquement pour l'acquisition du comportement *OBS*. Cela nous permet de passer en revue et de développer toutes ces approches à travers un exemple simple ; excepté celle des Réseaux de Neurones (*RN*) entraînés par l'*apprentissage supervisé* par le Renforcement par *Essai et Erreur (REE)*. Une fois ces approches maîtrisées, elles sont proposées pour l'acquisition des comportements *CIB*, *OBS*, *DEC*, et *ACT* pour les stratégies *NAV_1_Généralisée* et *NAV_2_Généralisée*.

II. 2. 1. Stratégie *NAV_1*

La stratégie de navigation *NAV_1*, dont le synoptique est présenté dans la Figure II. 1, est constituée des comportements intelligents:

- *CIB* (classification de situations de position de la cible),
- *OBS* (perception, et classification de situations d'évitement d'obstacles),
- *DEC* (associations et coordination afin de décider de l'action appropriée),
- *ACT*.

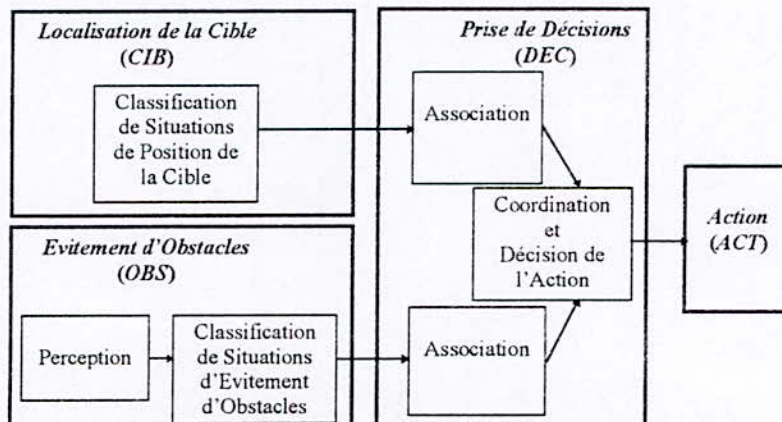


Figure II. 1. Synoptique de la Stratégie de Navigation *NAV_1*.

Dans cette stratégie, les mouvements des véhicules sont possibles seulement dans trois (03) directions pour *NAV_1_Simplifiée* alors qu'ils sont possibles dans cinq (05) directions pour *NAV_1_Généralisée*. Quant aux situations d'évitement d'obstacles (éventuels obstacles

rencontrés), elles doivent être perçus en situations topologiques. Ce qui conduit à structurer les éventuels obstacles rencontrés en six (06) et trente (30) situations topologiques pour *NAV_1_Simplifiée* et *NAV_1_Généralisée*, respectivement. Pour cela, le système de perception nécessaire à la détection d'obstacles *statiques*, donnant les distances véhicules-obstacles, devra se baser essentiellement sur les transducteurs (émetteurs-récepteurs) ultrasoniques, voir § A. 5 (ANNEXE'A).

II. 2. 1. 1. Stratégie *NAV_1_Simplifiée*

Les mouvements des véhicules sont possibles seulement dans trois (03) directions et par conséquent seulement trois (03) actions possibles A_i ($i=1, \dots, 3$) sont définies comme l'action de se déplacer à Droite, en Avant, et à Gauche présentées dans la Figure II. 2. Elles sont exprimées par le vecteur des actions $\mathbf{A} = [A_1, A_2, A_3]$.

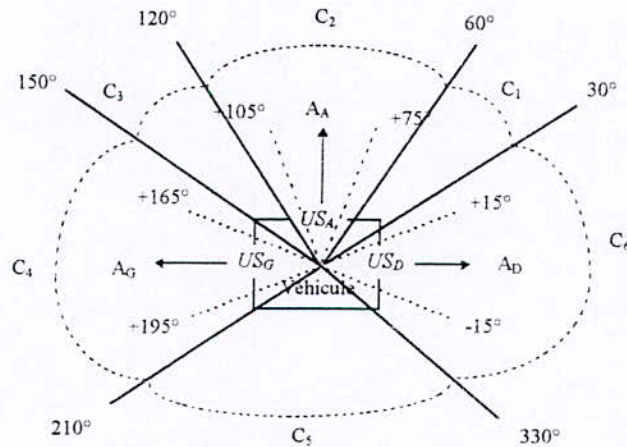


Figure II. 2. Véhicule, Capteurs, et Situations de Position de la Cible

$$C = [C_1, \dots, C_{j1}, \dots, C_6] \text{ (NAV}_1 \text{ Simplifiée).}$$

Afin de détecter les éventuels obstacles rencontrés, un certain nombre de capteurs à *US*, voir ANNEXE A, sont nécessaires pour obtenir les distances (véhicule-obstacle) couvrant les régions: à Droite US_D de -15° à $+15^\circ$, en Avant US_A de 75° à 105° , et à Gauche US_G de 165° à 195° , présentées dans la Figure II. 2.

- **Comportement de Localisation de la Cible (CIB)**

Afin de localiser et d'atteindre leurs cibles, la stratégie du champ de température développée dans [30] est utilisée menant vers la modélisation de l'environnement du véhicule en six (06) régions correspondant à toutes les positions de la cible appelées situations de

position de la cible présentées dans la Figure II. 2. Ces situations sont définies par six (06) Classes $C_1, \dots, C_{j1}, \dots, C_6$ où ($j1 = 1, \dots, 6$):

- Si $30^\circ \leq \gamma < 60^\circ$ (Classe C_1),
 - Si $60^\circ \leq \gamma < 120^\circ$ (Classe C_2),
 - Si $120^\circ \leq \gamma < 150^\circ$ (Classe C_3),
 - Si $150^\circ \leq \gamma < 210^\circ$ (Classe C_4),
 - Si $210^\circ \leq \gamma < 330^\circ$ (Classe C_5),
 - Si $330^\circ \leq \gamma < 390^\circ$ (Classe C_6),
- (II. 1)

où γ est l'angle de direction de la cible.

Ainsi, à chaque pas, un champ de température est défini dans l'environnement du véhicule avec un vecteur champ de température $X_C = [X_{C1}, X_{C2}, X_{C3}]$ à partir des températures, au voisinage du véhicule, à Droite t_D , en Avant t_A , et à Gauche t_G calculées dans § II. 3. 1. 1. La composante maximale de X_C c.-à-d., la température maximale de ce champ correspond à la situation C_{j1} où la cible est localisée:

La tâche du véhicule est alors de reconnaître à partir du vecteur d'entrée X_C dans quelle situation C_{j1} il trouve sa cible afin de se diriger vers ce maximum c.-à-d., dans la direction correspondant à cette situation.

• **Comportement d'Evitement d'Obstacles (OBS)**

Les mouvements possibles des véhicules conduisent à structurer les éventuels obstacles rencontrés en six (06) situations topologiques appelées situations d'évitement d'obstacles présentées dans la Figure II. 3. Ces situations sont définies par six (06) Classes $O_1, \dots, O_{j2}, \dots, O_6$ où ($j2 = 1, \dots, 6$).

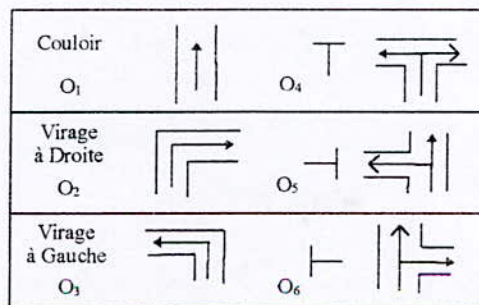


Figure II. 3. Situations d'Evitement d'Obstacles

$O = [O_1, \dots, O_{j2}, \dots, O_6]$ (NAV_1 Simplifiée).

Ainsi, à chaque pas, un vecteur distance $X_O = [X_{O1}, X_{O2}, X_{O3}]$ est calculé, dans § II. 3. 1. 2, à partir des distances véhicule-obstacle, au voisinage du véhicule, à Droite d_D , en Avant d_A , et à Gauche d_G .

La tâche du véhicule est alors de reconnaître à partir du vecteur d'entrée X_O dans quelle situation d'évitement d'obstacles O_{j2} il se trouve afin d'éviter les éventuels obstacles rencontrés.

II. 2. 1. 2. Stratégie NAV_1_Généralisée

La stratégie NAV_1_Généralisée que nous proposons [58] conduit à:

- une augmentation du nombre de mouvements possibles du véhicule de trois (03) à cinq (05) mouvements,
- une nouvelle distribution des situations de position de la cible par rapport au véhicule dans le comportement CIB,
- une augmentation du nombre de situations d'évitement d'obstacles de six (06) à trente (30) situations dans le comportement OBS.

Les mouvements des véhicules sont possibles dans cinq (05) directions et par conséquent cinq (05) actions possibles A_i ($i=1, \dots, 5$) sont définies comme action de se déplacer à 30° , à 60° , à 90° , à 120° , et à 150° , présentées dans la Figure II. 4, et exprimées par le vecteur des actions $A = [A_1, \dots, A_i, \dots, A_5]$.

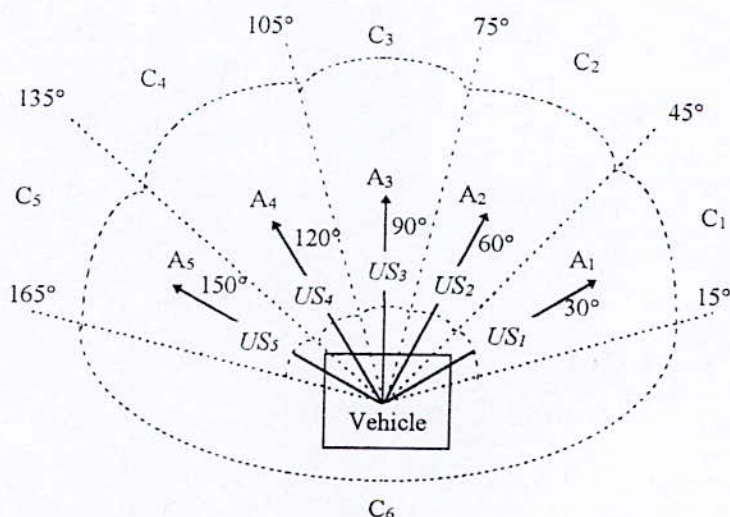


Figure II. 4. Véhicule, Capteurs, et Situations de Position de la Cible

$C = [C_1, \dots, C_{j1}, \dots, C_6]$ (NAV_1 Généralisée).

Afin de détecter les éventuels obstacles rencontrés, un certain nombre de capteurs à *US*, voir ANNEXE A, sont nécessaires pour obtenir les distances (véhicule-obstacle) couvrant la région 15° à 165° : US_1 de 15° à 45° , US_2 de 45° à 75° , US_3 de 75° à 105° , US_4 de 105° à 135° , et US_5 de 135° à 165° , présentée dans la Figure II. 4.

- **Comportement de Localisation de la Cible (CIB)**

Afin de localiser et d'atteindre leurs cibles, la même stratégie définie pour la stratégie *NAV_1_Simplifiée*, est utilisée sauf qu'il y'a une nouvelle distribution des six (06) situations de position de la cible présentées dans la Figure II. 4. Ces situations sont définies par six (06) Classes $C_1, \dots, C_{j1}, \dots, C_6$ où ($j1 = 1, \dots, 6$):

- $Si\ 15^\circ \leq \gamma < 45^\circ$ (Classe C_1),
 - $Si\ 45^\circ \leq \gamma < 75^\circ$ (Classe C_2),
 - $Si\ 75^\circ \leq \gamma < 105^\circ$ (Classe C_3),
 - $Si\ 105^\circ \leq \gamma < 135^\circ$ (Classe C_4),
 - $Si\ 135^\circ \leq \gamma < 165^\circ$ (Classe C_5),
 - $Si\ 165^\circ \leq \gamma < 375^\circ$ (Classe C_6),
- (II. 2)

où γ est l'angle de direction de la cible.

Ainsi, à chaque pas, un champ de température est défini dans l'environnement du véhicule avec un vecteur champ de température $X_C = [X_{C1}, \dots, X_{Ci}, \dots, X_{C5}]$ à partir des températures, au voisinage du véhicule, t_{30} , t_{60} , t_{90} , t_{120} , et t_{150} calculées dans § II. 3. 2. 1. La composante maximale de X_C c.-à-d., la température maximale de ce champ correspond à la situation C_{j1} où la cible est localisée.

La tâche du véhicule est alors de reconnaître à partir du vecteur d'entrée X_C dans quelle situation C_{j1} il trouve sa cible afin de se diriger vers ce maximum c.-à-d., dans la direction correspondant à cette situation.

- **Comportement d'Evitement d'Obstacles (OBS)**

Les mouvements possibles des véhicules conduisent à structurer les éventuels obstacles rencontrés en trente (30) situations topologiques appelées situations d'évitement d'obstacles présentées dans la Figure II. 5. Ces situations sont définies par trente (30) Classes $O_1, \dots, O_{j2}, \dots, O_{30}$ où ($j2 = 1, \dots, 30$). Dans la Figure II. 5, les directions montrées, dans chaque situation, correspondent à celles où des obstacles existent.

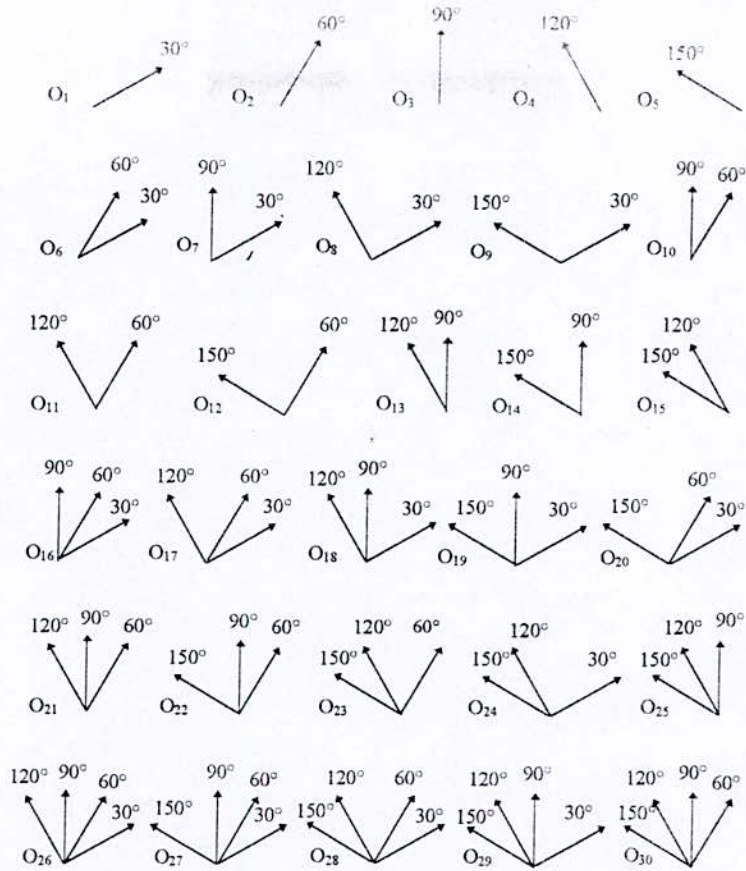


Figure II. 5. Situations d'Evitement d'Obstacles $O = \{O_1, \dots, O_{j_2}, \dots, O_{30}\}$
(NAV_1 Généralisée).

Ainsi, à chaque pas, un vecteur distance $X_O = [X_{O_1}, \dots, X_{O_i}, \dots, X_{O_5}]$ est calculé, dans § II. 3. 2. 2, à partir des distances véhicule-obstacle, au voisinage du véhicule, d_{30} dans la direction 30° , d_{60} dans la direction 60° , d_{90} dans la direction 90° , d_{120} dans la direction 120° , et d_{150} dans la direction 150° .

La tâche du véhicule est alors de reconnaître à partir du vecteur d'entrée X_O dans quelle situation d'évitement d'obstacles O_{j_2} il se trouve afin d'éviter les éventuels obstacles rencontrés.

- **Comportement de Prise de Décisions (DEC)**

Le comportement *DEC* proposé à la stratégie *NAV_1_Généralisée*, voir § II. 4. 1, prend en charge:

- l'association du comportement *CIB* avec les actions possibles,
- l'association du comportement *OBS* avec les actions possibles,
- la coordination des comportements *CIB* et *OBS* et la décision de l'action appropriée.

II. 2. 2. Stratégie NAV_2

La stratégie de navigation NAV_2, dont le synoptique est présenté dans la Figure II. 6, est constituée des comportements intelligents:

- CIB (traitement),
- OBS (perception, et classification des degrés des dangers statique et dynamique),
- DEC (classification des directions d'évitement et décision de l'action appropriée),
- ACT.

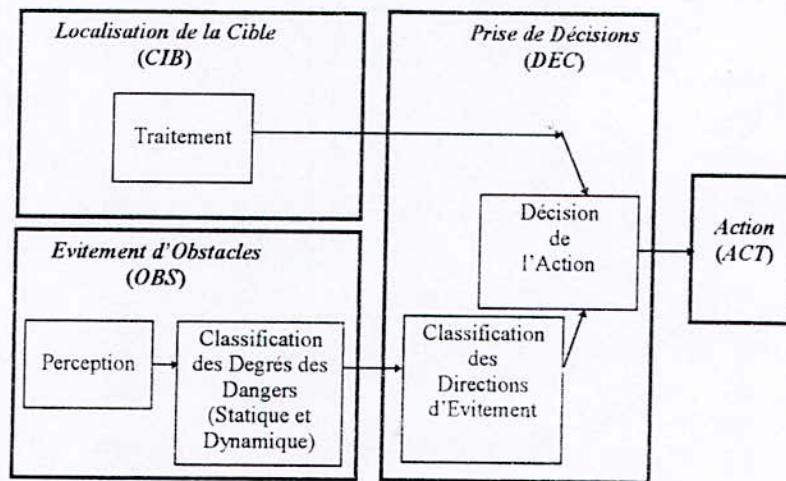


Figure II. 6. Synoptique de la Stratégie de Navigation NAV_2.

Dans cette stratégie, les mouvements des véhicules sont possibles dans onze (11) directions pour les stratégies NAV_2_Simplifiée et NAV_2_Généralisée. Quant aux éventuels obstacles rencontrés, ils doivent être perçus et traités par rapport aux degrés des dangers statique et dynamique qu'ils présentent aux véhicules [32]. Pour cela, le système de perception nécessaire à la détection d'obstacles *statiques*, donnant les distances véhicules-obstacles, devra se baser essentiellement sur les transducteurs ultrasoniques, voir § A. 5 (ANNEXE A), alors que la détection d'obstacles *dynamiques* devra se baser sur des capteurs et un traitement spécifiques donnant les vitesses des obstacles.

Les mouvements des véhicules sont possibles dans onze (11) directions et par conséquent onze (11) actions possibles A_i ($i=1, \dots, 11$) sont définies comme action de se déplacer à +5, à +4, à +3, à +2, à +1, à 0, à -1, à -2, à -3, à -4, et à -5 présentées dans la Figure II. 7. Elles sont exprimées par le vecteur des actions $A = [A_1, \dots, A_i, \dots, A_{11}]$.

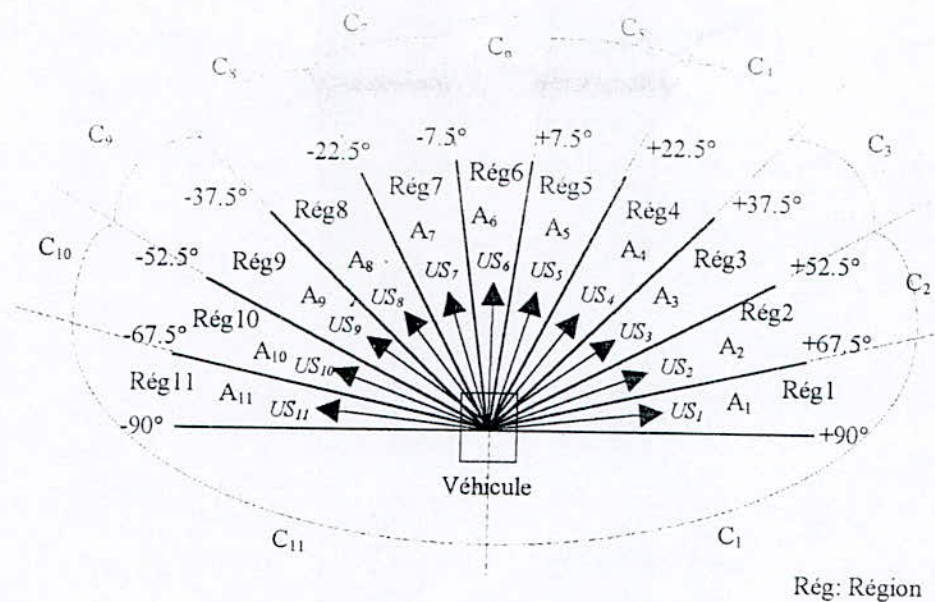


Figure II. 7. Véhicule, Capteurs, et Situations de Position de la Cible
 $C = [C_1, \dots, C_i, \dots, C_{11}]$ (*NAV_2_Simplifiée* et *NAV_2_Généralisée*).

Afin de détecter les éventuels obstacles rencontrés, un certain nombre de capteurs à *US*, voir ANNEXE A, sont nécessaires pour obtenir les distances (véhicule-obstacle) couvrant la région $+90^\circ$ à -90° : US_1, US_2, \dots , et US_{11} couvrant les Régions 1, 2, ..., et 11, respectivement, présentées dans la Figure II. 7.

- **Comportement de Localisation de la Cible (*CIB*)**

Afin de localiser et d'atteindre leurs cibles, la même stratégie définie pour la stratégie *NAV_1* est également proposée pour cette stratégie *NAV_2* avec une nouvelle distribution des situations de position de la cible. Ces situations sont définies par onze (11) Classes $C_1, \dots, C_i, \dots, C_{11}$, voir Figure II. 7.

II. 2. 2. 1. Stratégie *NAV_2_Simplifiée*

- **Comportement d'Evitement d'Obstacles (*OBS*)**

Le comportement *OBS*, développé par Y. Maeda [32], est proposé aux stratégies *NAV_2_Simplifiée* et *NAV_2_Généralisée*. En premier, à partir des vecteurs donnés de position et de vitesse absolues du véhicule \mathbf{P}_v et \mathbf{V}_v , respectivement et à partir d'une position

absolue donnée de la cible P_C , le véhicule détermine, en utilisant les données des capteurs et un traitement, les vecteurs absolus de la position et de la vitesse de l'obstacle $P(\text{Obs})$ et $V(\text{Obs})$ respectivement présentés dans la Figure II. 8.

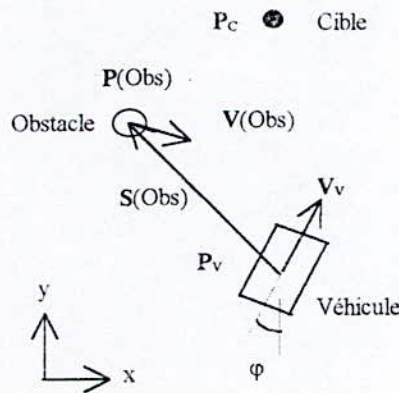


Figure II. 8. Modèle d'Evitement d'Obstacles du Véhicule
(NAV_2_Simplifiée et NAV_2_Généralisée).

Après mise à jour de sa position P_V et du vecteur position de la cible P_C basée sur les coordonnées cartésiennes, il transforme $P(\text{Obs})$ et $V(\text{Obs})$ en vecteurs de position et de vitesse relatives $S(\text{Obs})$ et $U(\text{Obs})$ basées sur les coordonnées polaires du véhicule.

Ensuite, le véhicule reconnaît les degrés des dangers statique O_{α_j} et dynamique O_{β_j} entre lui-même et l'obstacle en utilisant un raisonnement et une inférence flous. En fait, O_{α_j} et O_{β_j} sont décidés à partir de $S(\text{Obs})$ et $U(\text{Obs})$ respectivement.

Ainsi, le comportement *OBS* se base essentiellement sur la classification des degrés des dangers statique O_{α_j} et dynamique O_{β_j} d'un obstacle.

Degré du Danger Statique O_{α_j} : Le degré du danger statique O_{α_j} d'un obstacle est déterminé à partir du vecteur de position relative S_O du véhicule-obstacle.

$$\varphi = \pi/2 - \tan^{-1}(V_{Vx}, V_{Vy}) \text{ avec } \mathbf{V}_V = [V_{Vx}, V_{Vy}]^T \quad (\text{II. 3})$$

$$\mathbf{S}_O = \mathbf{R} (\mathbf{P}_O - \mathbf{P}_V) = [S_{Ox}, S_{Oy}]^T \quad (\text{II. 4})$$

$$\text{où } \mathbf{R} = \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix} \quad (\text{II. 5})$$

φ : direction du véhicule basée sur les coordonnées cartésiennes absolues.

Ensuite, la direction θ_S et la distance L_S de l'obstacle basées sur les coordonnées polaires du véhicule, dont l'angle est défini 0° en face, les valeurs positives à droite, et les valeurs négatives à gauche du véhicule, sont données par:

$$\theta_S = \pi/2 - \tan^{-1}(S_{Oy}/S_{Ox}) \quad (\text{II. 6})$$

$$L_S = \sqrt{S_{Ox}^2 + S_{Oy}^2} \quad (\text{II. 7})$$

La perception des obstacles est possible à l'intérieur des limites de 180° en avant, c.-à-d., entre -90° et $+90^\circ$, et les fonctions d'appartenance de la direction θ_S sont représentées dans la Figure II. 9, où les variables linguistiques floues sont définies par Gauche Grand (GG), Gauche Petit (GP), Droite Petit (DP), et Droite Grand (DG).

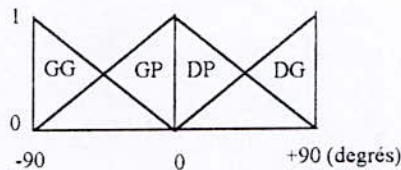


Figure II. 9. Fonctions d'Appartenance de la Direction.

De plus, le véhicule est supposé percevoir un obstacle à l'intérieur d'une distance de 150 cm, et les fonctions d'appartenance de distance L_S sont présentées dans la Figure II. 10, où les variables linguistiques floues sont définies par Proche (P), Moyen (M), et Loin (L).

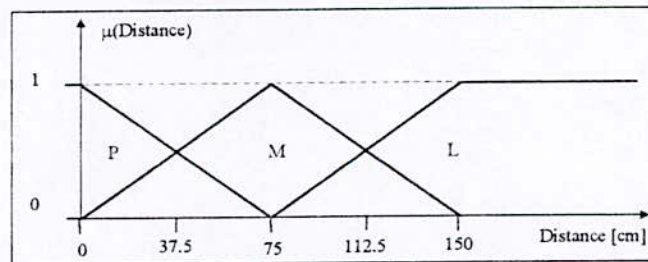


Figure II. 10. Fonctions d'Appartenance de la Distance.

Les fonctions d'appartenance de O_{α_j} sont présentées dans la Figure II. 11, où les variables linguistiques floues sont définies par les degrés de danger sur le côté gauche Danger Gauche Grand (DGG), Danger Gauche Petit (DGP), Sécurité Gauche Petit (SGP), et Sécurité Gauche Grand (SGG), alors que les degrés de danger sur le côté droit Danger Droite Grand (DDG), Danger Droite Petit (DDP), Sécurité Droite Petit (SDP), et Sécurité Droite Grand (SDG).

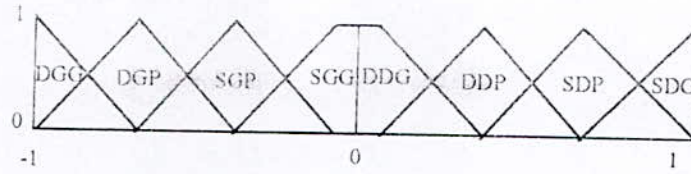


Figure II. 11. Fonctions d'Appartenance des Degrés des Dangers Statique et Dynamique.

Degré du Danger Dynamique $O_{\beta j}$: Le degré du danger dynamique $O_{\beta j}$ d'un obstacle est déterminé à partir du vecteur de vitesse relative U_O du véhicule-obstacle, où R est la matrice de transformation définie dans l'Equation (II. 5):

$$U_O = R (V_O - V_V) = [U_{Ox}, U_{Oy}]^T \quad (II. 8)$$

U_O est transformé au vecteur de vitesse relative U_O' basé sur les coordonnées cartésiennes du véhicule, où l'axe y est confondu avec la direction vers l'obstacle.

$$U_O' = R_s U_O = [U_{Ox}', U_{Oy}']^T \quad (II. 9)$$

$$R_s = \begin{bmatrix} \cos\theta_s & -\sin\theta_s \\ \sin\theta_s & \cos\theta_s \end{bmatrix} \quad (II. 10)$$

Ainsi, la direction de la vitesse Φ_U et la vitesse W_U de l'obstacle basées sur les coordonnées polaires du véhicule, dont l'angle est défini 0° dans la direction de l'obstacle, les valeurs positives à droite, et les valeurs négatives à gauche, sont données par:

$$\Phi_U = \pi/2 - \tan^{-1}(S_{Oy}/S_{Ox}) \quad (II. 11)$$

$$W_U = \sqrt{U_{Ox}'^2 + U_{Oy}'^2} \quad (II. 12)$$

La direction de la vitesse est limitée entre -180° et $+180^\circ$, et les fonctions d'appartenance de la direction de la vitesse Φ_U sont présentées dans la Figure II. 12, où les variables linguistiques floues sont définies par Gauche Grand (GG), Gauche Moyen (GM), Gauche Petit (GP), Droite Petit (DP), Droite Moyen (DM), et Droite Grand (DG).

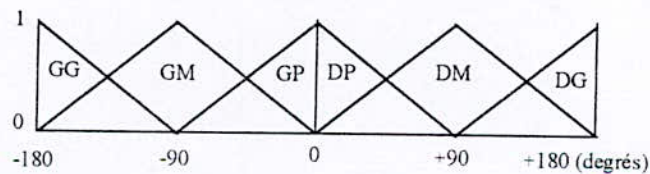


Figure II. 12. Fonctions d'Appartenance de la Direction de la Vitesse.

La vitesse du véhicule est supposée varier entre 0 et 100 cm/s, et les fonctions d'appartenance de W_V sont présentées dans la Figure II. 13, où les variables linguistiques floues sont définies par Lent (L) et Rapide (R). Notons que les fonctions d'appartenance de O_{β_j} sont les mêmes que celles de O_{α_j} présentées dans la Figure II. 11.

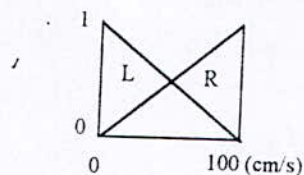


Figure II. 13. Fonctions d'Appartenance de la Vitesse.

Bien que le comportement *OBS* développé par Y. Maeda se basant sur l'évaluation des degrés des dangers statique et dynamique a prouvé son efficacité et son intelligence, le majeur problème de ce comportement est qu'un véhicule ne peut pas rencontrer plusieurs obstacles à la fois et que la taille des obstacles n'est pas prise en compte.

- **Comportement de Prise de Décisions (*DEC*)**

Le comportement *DEC*, se basant sur celui développé par Y. Maeda, devra tout d'abord utiliser O_{α_j} et O_{β_j} afin de décider de la direction d'évitement O_i , et alors du vecteur direction d'évitement \mathbf{O} . Ensuite, en utilisant les résultats issus du comportement *CIB* et ceux du comportement *OBS*, le comportement *DEC* devra décider de l'action appropriée.

Le bloc-diagramme de l'algorithme de décisions se basant principalement sur les comportements *OBS* et *DEC* développés par Y. Maeda [32] est présenté dans la Figure II. 14.

II. 2. 2. 2. Stratégie *NAV_2_Généralisée*

La stratégie *NAV_2_Généralisée* que nous proposons [54, 60] conduit à:

- une procédure de sélection de points obstacles ainsi qu'une classification des degrés des dangers statique et dynamique de chaque point obstacle, afin de prendre en charge la taille d'un obstacle dans le comportement *OBS* développé par Y. Maeda,
- une classification des directions d'évitement de chaque point obstacle ainsi qu'une procédure de classification en ensembles d'évitement et de décision final du vecteur direction d'évitement, afin de prendre en charge plusieurs obstacles à la fois dans le comportement *DEC* développé par Y. Maeda.

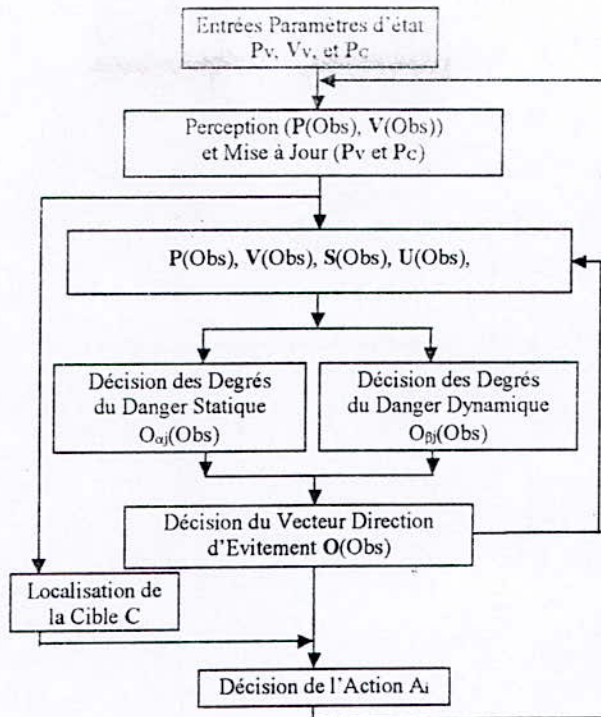


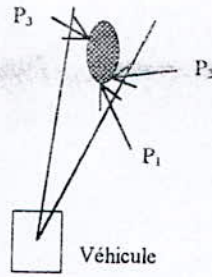
Figure II. 14. Bloc-Diagramme de l'Algorithme de Décisions (*NAV_2 Simplifiée*).

- **Comportement d'Evitement d'Obstacles (OBS)**

Afin de prendre en charge la taille d'un obstacle, nous proposons alors la *Procédure 1* appelée *Sélection de Points Obstacles* permettant de sélectionner les points obstacles à prendre en considération, ainsi que la classification des degrés des dangers statique et dynamique de chacun de ces points obstacles sélectionnés.

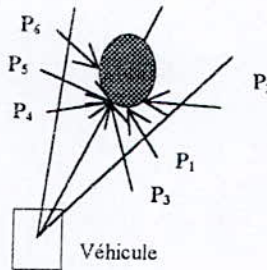
Sélection de Points Obstacles: Cette procédure sélectionnant plusieurs points obstacles est réalisée en utilisant des données à partir de la perception de l'environnement. En fait, le véhicule peut percevoir la région à l'intérieur des limites de 180° en avant. Cette région est modélisée par onze (11) régions, selon les onze (11) directions possibles du véhicule, présentées dans la Figure II. 7.

- Pour un obstacle occupant une (01) seule région, les points obstacles sélectionnés par rapport au véhicule sont le point le plus proche P_1 , le point le plus à droite P_2 , et le point le plus à gauche P_3 , voir Figure II. 15. (a). Pour chaque région, trois (03) points sont alors sélectionnés constituant les obstacles à prendre en considération.



**Figure II. 15. (a). Sélection de Points Obstacles:
Cas d'un Obstacle Occupant une Seule Région.**

- Dans le cas d'un obstacle occupant plus d'une région, trois (03) points doivent être sélectionnés pour chaque région p.ex., dans le cas présenté dans la Figure II. 15. (b): le point le plus proche P_1 , le point le plus à droite P_2 , et le point le plus à gauche P_3 pour la région de droite ainsi que le point le plus proche P_4 , le point le plus à droite P_5 , et le point le plus à gauche P_6 pour la région de gauche. Un obstacle d'une certaine taille est alors considéré comme plusieurs points obstacles.



**Figure II. 15. (b). Sélection de Points Obstacles:
Cas d'un Obstacle Occupant Plus d'Une Région.**

Classification des Degrés des Dangers Statique et Dynamique: La tâche du véhicule est alors de reconnaître les degrés des dangers statique O_{α_j} et dynamique O_{β_j} de chaque point obstacle c.-à-d., $O_{\alpha_j}(\text{Obs1})$, $O_{\alpha_j}(\text{Obs2})$, ... et $O_{\beta_j}(\text{Obs1})$, $O_{\beta_j}(\text{Obs2})$, ...

- **Comportement de Prise de Décisions (DEC)**

Afin de prendre en charge plusieurs obstacles à la fois, nous proposons la classification des directions d'évitement ainsi que la *Procédure 2* appelée *Classification et Décision Finale*

permettant la classification en ensembles d'évitement et de décision final du vecteur direction d'évitement.

Classification des Directions d'Évitement: Le véhicule doit reconnaître la direction d'évitement pour chaque point obstacle c.-à-d., $O(\text{Obs1})$, $O(\text{Obs2})$, etc. Il devra utiliser les degrés des dangers statique O_{α_j} et dynamique O_{β_j} afin de reconnaître la direction d'évitement O_i et donc du vecteur direction d'évitement \mathbf{O} pour chaque point obstacle.

Classification et Décision Finale: Cette procédure prend en charge la classification en ensembles d'évitement et la décision final du vecteur direction d'évitement.

- La classification se base sur la définition de trois (03) ensembles d'évitement particuliers.

* l'ensemble d'évitement possible est constitué de onze (11) mouvements possibles du véhicule,

* l'ensemble d'évitement dangereux est constitué des directions qui doivent être éliminées, pour éviter les collisions avec les obstacles, p.ex. si un obstacle est détecté dans la région Rég3, voir Figure II. 7, la direction +3 (A_3) correspondante doit être éliminée,

* l'ensemble d'évitement prioritaire est constitué des directions données par la décision des vecteurs direction d'évitement $O(\text{Obs1})$, $O(\text{Obs2})$, etc.

- La décision finale du vecteur direction d'évitement $\mathbf{O}(\text{Obstacles})$ consiste à sélectionner parmi l'ensemble d'évitement prioritaire la direction appropriée.

Le bloc-diagramme de l'algorithme de décisions, permettant d'améliorer les comportements *OBS* et *DEC* développés par Y. Maeda, est présenté dans la Figure II. 16.

Notons que la décision de l'action appropriée A_i est obtenue à partir des résultats des comportements *CIB* et *OBS*, vecteur position de la cible \mathbf{C} et vecteur direction d'évitement $\mathbf{O}(\text{Obstacles})$, respectivement.

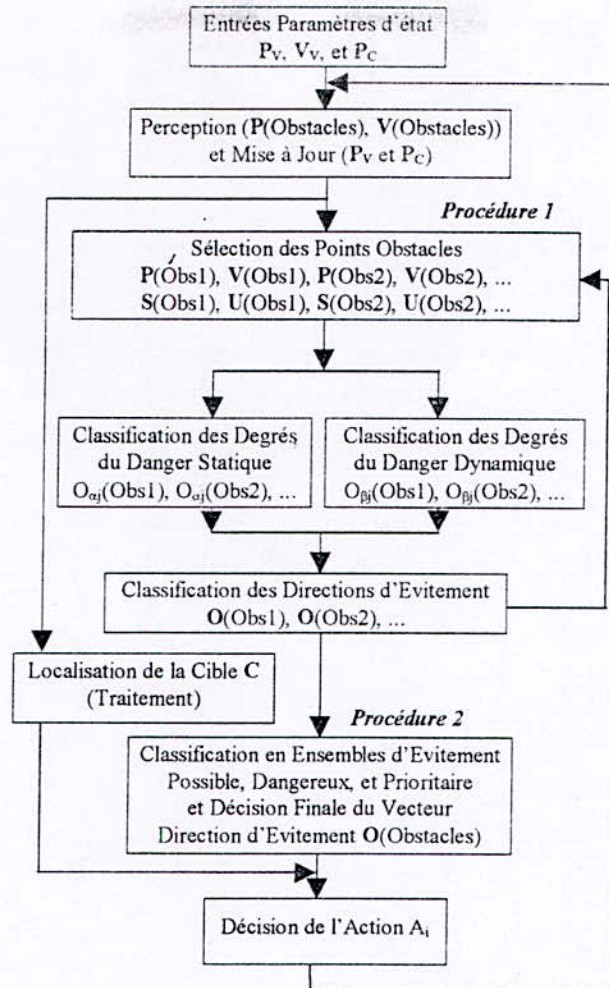


Figure II. 16. Bloc-Diagramme de l'Algorithme de Décisions (*NAV_2 Généralisée*).

II. 2. 3. Comportement d'Action (ACT) Proposé aux Stratégies *NAV_1 NAV_2 Généralisées*

L'objectif principal des chercheurs dans le domaine des *VAI* est orienté surtout vers leur capacité à naviguer et à être *auto-suffisants* dans des environnements dynamiques partiellement structurés. En effet, afin d'atteindre leurs cibles sans collisions, avec d'éventuels obstacles rencontrés, les *VAI* doivent avoir la capacité à réaliser les comportements intelligents *CIB*, *OBS*, et *DEC* donnant, à chaque instant, une action A_i (orientation) appropriée. Le comportement intelligent *ACT* devra alors permettre aux *VAI* d'avoir la capacité à générer des trajectoires (mouvements) *lisses* à partir de cette action A_i (orientation) et une distance-but du mouvement à exécuter. Ainsi, un contrôleur de mouvements devra alors utiliser cette information sur l'orientation ainsi que sur la distance-but afin de contrôler la vitesse de ces

véhicules. Il devra satisfaire l'exigence de ces véhicules: la génération de mouvements *lisses en temps réel avec intelligence*.

Afin de remédier aux problèmes des approches analytiques et aux imprécisions conduisant à des effets indésirables, le contrôleur de mouvements que nous proposons, aux stratégies *NAV_1_NAV_2_Généralisées*, est un *Système Intelligent (SI)* qui se base sur le contrôle *temps minimum* [126] et sur la *LF* [29, 40]. En fait, le contrôle hiérarchique intelligent permet d'utiliser les données de capteurs avec divers niveaux de précision alors que la *LF* prend en charge les incertitudes inhérentes à la représentation des connaissances de l'environnement. Les données d'entrée de ce *SI* sont:

- la Vitesse Finale: V_{finale} ,
- la Vitesse Initiale (courante): $V_{initiale}$, $V_{courante}$,
- la Distance-But (courante): d_{but} , $d_{courante}$.

Ces données d'entrée sont données à un haut niveau de planification et de contrôle gérant les comportements intelligents *CIB*, *OBS*, *DEC*, et *ACT*.

- **Nécessité du Contrôle Temps Minimum et de la LF**

Le contrôle classique de mouvements des véhicules introduit des contrôleurs analytiques et ceux basés sur des règles, où les règles de contrôle de mouvements sont dérivées à partir d'un modèle analytique. De plus, le plus bas niveau de contrôle qui implique l'exécution d'un certain mouvement nécessite un modèle mathématique de la dynamique des véhicules. Dans leur dérivation des règles, les approches analytiques impliquent des approximations et des simplifications dans le modèle mathématique. Une représentation *plus exacte*, que celle des approches analytiques, du comportement des véhicules est alors possible en utilisant les approches expérimentales, dans ce cas les règles floues sont dérivées expérimentalement à partir d'expériences basées sur le contrôle *temps minimum*, connu par le contrôle "*bang-bang*" [29, 40, 126]. Les règles dérivées par des approches expérimentales peuvent être utilisées afin de construire un contrôleur de mouvements au plus haut niveau de contrôle en utilisant un modèle linguistique de la dynamique des véhicules, basé sur la *LF*.

Par ailleurs, les données de capteurs aux divers niveaux de contrôle ainsi que l'environnement des véhicules impliquent des incertitudes qui doivent être prises en charge dans le but de réaliser des mouvements lisses. En effet, les éventuelles sources de telles incertitudes peuvent être relatives aux capteurs (limitations ou défaillances des capteurs), à la

représentation des connaissances de l'environnement, ou alors à la spécification des actions elles-mêmes. De telles incertitudes impliquent des imprécisions dans le comportement *ACT* et par suite elles impliquent des imprécisions dans les mouvements des véhicules conduisant à des effets indésirables p.ex., changements brusques de vitesse. Le contrôle hiérarchique intelligent permet alors d'utiliser les données de capteurs avec divers niveaux de précision tandis que la *LF* prend en charge les incertitudes inhérentes à la représentation des connaissances de l'environnement. Le contrôle optimal peut être ainsi utilisé efficacement dans la conception et la réalisation des niveaux les plus bas de contrôle des mouvements des véhicules.

- **Contrôle Temps Minimum et Fuzzification des Variables d'Etat**

Une stratégie de contrôle est utilisée au plus bas niveau de contrôle afin de dériver les actions appropriées de contrôle. La stratégie de contrôle choisie est le contrôle *temps minimum*, puisque le problème est considéré être celui d'atteindre un sous-but en un temps minimum avec une certaine précision. Pour le mouvement du véhicule le long d'une trajectoire linéaire, la vitesse et la distance-but sont les variables d'état et l'accélération linéaire est l'entrée de contrôle.

Notons que l'erreur finale de position admissible est considérée être la contrainte majeure dans la construction d'un vocabulaire admissible linguistique pour ce *SI*. De plus, les niveaux de précision des variables d'état sont définis de telle manière que le nombre d'intervalles discrets pour les variables d'état est minimisé tout en tenant compte de la contrainte d'erreur finale de position admissible. Cette minimisation a en fin de compte un effet sur la dimension de la base de règles floues, puisqu'une règle floue particulière doit être assignée à chaque combinaison éventuelle d'intervalles discrets (de vitesse et de distance-but). Ainsi, afin d'obtenir l'action appropriée de contrôle pour chaque état éventuel du véhicule, des expériences sont exécutées. Ces expériences sont prises en charge sur la base du critère contrôle *temps minimum*. La solution bien connue du contrôle temps minimum peut simplement être exprimée par la règle:

Accélérer jusqu'à ce que le véhicule atteigne sa vitesse maximale, alors maintenir cette vitesse jusqu'à ce que la distance véhicule-position désirée (distance-but) soit suffisante (juste assez) pour décélérer à la vitesse finale désirée. Lorsqu'une telle distance est atteinte, commencer la décélération jusqu'à ce que les conditions finales soient satisfaites.

Le profil théorique de la vitesse du véhicule sous le contrôle *temps minimum* est présenté dans la Figure II. 17.

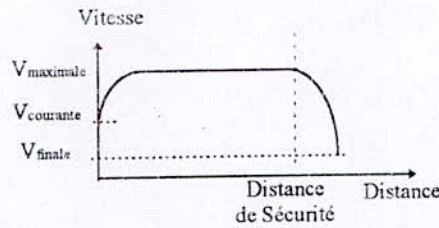


Figure II. 17. Profil de la Vitesse du Véhicule sous le Contrôle Temps Minimum.

Dans le cas de l' "AGV", voir § I. 4. 2. 2, dont la vitesse maximale est de 0.4 m/s, la taille d'un intervalle de vitesse est de 0.1 m/s, les tailles des intervalles de vitesse sont alors: vitesse $v_i \in \{v_1, v_2, v_3, v_4\}$ avec $i = 1, \dots, 4$, où $v_1 = [0.0, 0.1]$, $v_2 = [0.1, 0.2]$, $v_3 = [0.2, 0.3]$, et $v_4 = [0.3, 0.4]$.

Rappelons que le système de perception utilisé pour la détection d'obstacles se base sur des capteurs à *US* et que pour une détection fiable des obstacles par ce type de capteurs, seulement la détection dans l'étendue de deux (02) mètres est prise en compte [20, 109]. Ainsi, la taille d'un intervalle de distance-but est de $\Delta d = 0.375$ m et par suite les tailles d'intervalles de distance-but sont alors: distance-but $d_j \in \{d_1, d_2, d_3, d_4, d_5\}$ avec $j = 1, \dots, 5$, où $d_1 = [0.000$ m, 0.375 m], $d_2 = [0.375$ m, 0.750 m], $d_3 = [0.750$ m, 1.125 m], $d_4 = [1.125$ m, 1.500 m], et $d_5 = [1.500$ m, 2.000 m]. Notons que si l'étendue des erreurs de mesure des capteurs s'approche des tailles d'intervalles de distance-but, la transformation floue devient imprécise.

En utilisant les tailles d'intervalles discrets de vitesse et celles de distance-but, l'espace d'état du véhicule est discrétisé en variables linguistiques relatives aux intervalles discrets. Par conséquent, quatre (04) intervalles discrets de la vitesse conduisent à quatre (04) variables linguistiques présentées dans la Figure II. 18, où TL, L, R, TR correspondent à Très Lente, Lente, Rapide, et Très Rapide, respectivement.

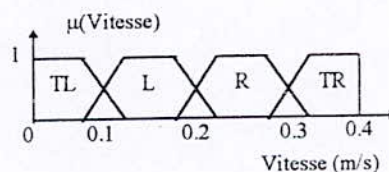


Figure II. 18. Fonctions d'Appartenance de la Vitesse.

Quant aux cinq (05) intervalles discrets de la distance-but, ils conduisent à cinq (05) variables linguistiques présentées dans la Figure II. 19, où TP, P, L, TL, et HE correspondent à

Très Proche, Proche, Loin, Très Loin, et Hors Etendue, respectivement.

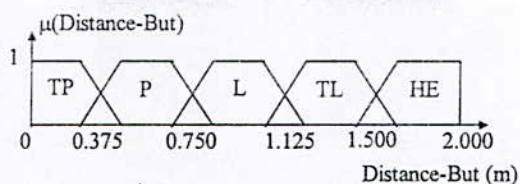


Figure II. 19. Fonctions d'Appartenance de la Distance-But.

Les fonctions d'appartenance de la distance courante, sont données dans la Figure II. 20, où ACC, MAI, et DEC correspondent à ACCélérer, MAIntenir, et DECélérer, respectivement.

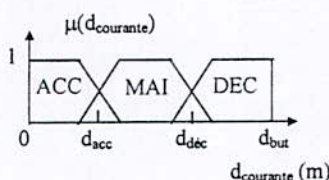


Figure II. 20. Fonctions d'Appartenance de la Distance Courante.

- **Expériences et Règles Floues**

La commande d'action de mouvements CA_k (avec $k = 1, \dots, 6$) doit être choisie parmi les commandes d'actions suivantes afin d'actionner les moteurs du véhicule: $CA_1 = \{\text{Stop, stopper l'actionnement du moteur } (V_0)\}$, $CA_2 = \{\text{Croisière, maintenir la même vitesse}\}$, $CA_3 = \{\text{Changer la vitesse à } V_4\}$, $CA_4 = \{\text{Changer la vitesse à } V_3\}$, $CA_5 = \{\text{Changer la vitesse à } V_2\}$, et $CA_6 = \{\text{Changer la vitesse à } V_1\}$. Notons que les valeurs des vitesses finales désirées V_0, V_1, V_2, V_3 , et V_4 sont déterminées durant les expériences, voir § II. 5.

La tâche est maintenant de déterminer expérimentalement les distances d'accélération/décélération pour chaque combinaison de la vitesse et de la distance-but c.-à-d., pour chaque Vitesse Finale v_i , Vitesse Initiale v_i , et Distance-But d_j . A partir de ces expériences les règles floues de contrôle de mouvements sont dérivées, voir § II. 5, à partir de la règle suivante:

Si Vitesse Courante est v_i et Distance-But est d_j ,

Alors Changer la Vitesse à V_i avec Accélération/Décélération Maximale.

II. 3. *SI* ET *SHI* PROPOSES POUR L'ACQUISITION DES COMPORTEMENTS DE LOCALISATION DE LA CIBLE (*CIB*) ET D'ÉVITEMENT D'OBSTACLES (*OBS*)

Nous proposons, dans ce Paragraphe, différents *SI* et *SHI* pour l'acquisition, par *apprentissage* ou/et *adaptation*, des comportements *CIB* et *OBS* pour les stratégies *NAV_1_Simplifiée*, *NAV_1_Généralisée* et *NAV_2_Généralisée*.

II. 3. 1. Stratégie *NAV_1_Simplifiée*

Les *SI* et *SHI* proposés, pour l'acquisition des comportements *CIB* et *OBS*, à la stratégie *NAV_1_Simplifiée*, sont cités dans la Figure II. 21.

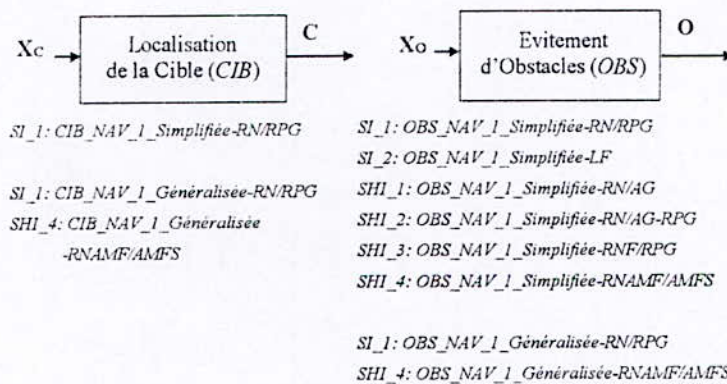


Figure II. 21. *SI* et *SHI* pour les Comportements *CIB* et *OBS* (*NAV_1_Simplifiée* et *NAV_1_Généralisée*).

• Comportement *CIB*

Le véhicule doit apprendre à reconnaître les situations de position de la cible, présentées dans la Figure II. 2, pour sa localisation ; ce qui lui permettra de se diriger vers cette cible. En fait, le comportement *CIB* se base sur une reconnaissance acquise par *apprentissage* ou/et *adaptation* à partir de données obtenues par la stratégie du champ de température [30]. Après apprentissage, pour chaque vecteur d'entrée X_c , ces *SI* et *SHI* devront fournir au véhicule la capacité à décider de la situation de position de la cible C_{j1} exprimée par la sortie C_{j1} ayant la plus grande activation.

- Comportement *OBS*

Ce véhicule doit également apprendre à reconnaître les situations d'évitement d'obstacles, présentées dans la Figure II. 3, ce qui lui permettra de les éviter. En fait, le comportement *OBS* se base sur une reconnaissance acquise par *apprentissage* ou/et *adaptation* à partir de données des capteurs à *US*, présentés dans l'ANNEXE A, donnant les distances véhicule-obstacles. Après apprentissage, pour chaque vecteur d'entrée X_0 , ces *SI* et *SHI* devront fournir au véhicule la capacité à décider de la situation d'évitement d'obstacles O_{j2} exprimée par la sortie O_{j2} ayant la plus grande activation.

II. 3. 1. 1. *SI_1: CIB_NAV_1_Simplifiée-RN/RPG*

Le *SI_1: CIB_NAV_1_Simplifiée-RN/RPG* [46], proposé au comportement *CIB*, doit classifier les situations de position de la cible en utilisant les *RN*. Cette classification se base sur l'*apprentissage supervisé* par la *RPG* [62, 64, 67, 69, 72, 71, 74], présenté dans l'ANNEXE C. L'architecture de ce *SI* est présenté dans la Figure C. 1 (ANNEXE C).

Ce *SI* est entraîné, voir § III. 3. 1. 1, à partir de données obtenues de la stratégie du champ de température. Cette stratégie est définie dans l'environnement du véhicule et la tâche du véhicule est alors de localiser sa cible correspondant à l'unique température maximale de ce champ. Les températures au voisinage du véhicule sont définies par t_D , t_A , et t_G correspondant aux températures à Droite, en Avant, et à Gauche, respectivement. Ces températures sont calculées en utilisant les fonctions sinus et cosinus comme suit:

Si $30^\circ \leq \gamma < 60^\circ$ (Classe C_1),

$$\text{Alors } t_D = \cos(\gamma), t_A = \sin(\gamma), t_G = \sin(\gamma)/2,$$

Si $60^\circ \leq \gamma < 120^\circ$ (Classe C_2),

$$\text{Si } 60^\circ \leq \gamma \leq 90^\circ$$

$$\text{Alors } t_D = \cos(\gamma), t_A = \sin(\gamma), t_G = \sin(\gamma)/2,$$

$$\text{Si } 90^\circ < \gamma < 120^\circ$$

$$\text{Alors } t_D = \sin(\gamma)/2, t_A = \sin(\gamma), t_G = -\cos(\gamma),$$

Si $120^\circ \leq \gamma < 150^\circ$ (Classe C_3),

$$\text{Alors } t_D = \sin(\gamma)/2, t_A = \sin(\gamma), t_G = -\cos(\gamma),$$

Si $150^\circ \leq \gamma < 210^\circ$ (Classe C_4),

Alors $t_D = \sin(\gamma)/2$, $t_A = \sin(\gamma)$, $t_G = -\cos(\gamma)$,

Si $210^\circ \leq \gamma < 330^\circ$ (Classe C_5),

Si $210^\circ \leq \gamma \leq 270^\circ$

Alors $t_D = \sin(\gamma)/2$, $t_A = \sin(\gamma)$, $t_G = -\cos(\gamma)$,

Si $270^\circ < \gamma < 330^\circ$

Alors $t_D = \cos(\gamma)$, $t_A = \sin(\gamma)$, $t_G = \sin(\gamma)/2$,

Si $330^\circ \leq \gamma < 390^\circ$ (Classe C_6),

Alors $t_D = \cos(\gamma)$, $t_A = \sin(\gamma)$, $t_G = \sin(\gamma)/2$, (II.

13)

Ces composantes sont prétraitées, normalisées entre 0 et 1, afin de constituer le vecteur d'entrée X_C de ce SI :

$$X_{C1} = (1/\rho_C) \exp(-t_D/a_C),$$

$$X_{C2} = (1/\rho_C) \exp(-t_A/a_C),$$

$$X_{C3} = (1/\rho_C) \exp(-t_G/a_C), \quad (\text{II. 14})$$

où ρ_C : norme du vecteur d'entrée X_C and a_C : facteur de prétraitement d'entrée avec $a_C > 1$.

II. 3. 1. 2. $SI_{1: OBS_NAV_1_Simplifiée-RN/RPG}$

Le $SI_{1: OBS_NAV_1_Simplifiée-RN/RPG}$ [46], proposé au comportement OBS , doit classifier les situations d'évitement d'obstacles en utilisant les RN . Cette classification se base sur l'apprentissage supervisé par la RPG [62, 64, 67, 69, 71, 72, 74], détaillé dans l'ANNEXE C. L'architecture de ce SI est présentée dans la Figure C. 1 (ANNEXE C).

Ce SI est entraîné, voir § III. 3. 1. 2, à partir de données des capteurs à US obtenues de l'environnement donnant les distances véhicule-obstacle. Ces distances sont définies au voisinage du véhicule par d_D , d_A , et d_G correspondant aux distances à Droite, en Avant, et à Gauche, respectivement. Ces composantes sont prétraitées afin de constituer le vecteur d'entrée X_O de ce SI :

$$X_{O1} = (1/\rho_O) \exp(-d_D/a_O),$$

$$X_{O2} = (1/\rho_O) \exp(-d_A/a_O),$$

$$X_{O3} = (1/\rho_O) \exp(-d_G/a_O), \quad (\text{II. 15})$$

où ρ_O : norme du vecteur d'entrée X_O et a_O : facteur de prétraitement d'entrée avec $a_O > 1$.

II. 3. 1. 3. *SI_2: OBS_NAV_1_Simplifiée-LF*

Le *SI_2: OBS_NAV_1_Simplifiée-LF* [50], proposé au comportement *OBS*, doit classifier les situations d'évitement d'obstacles en utilisant la *LF*. Cette classification se base sur une adaptation 'supervisée', voir § I. 5. 2. En effet, ce *SI* est utilisé afin de saisir le comportement d'un expert humain tout en contrôlant le comportement *OBS* c.-à-d., il doit mimer la correspondance d'entrée/sortie de cet expert humain. En effet, ce dernier en formulant sa connaissance sous une forme linguistique fournit une explication afin de contrôler ce comportement. Pour mimer ce contrôle, la formulation linguistique floue est utilisée et un ensemble de règles floues sont alors établies. Ces règles floues sont alors utilisées afin d'inclure cette connaissance de l'expert humain dans l'architecture de ce *SI* présentée dans la Figure II. 22.

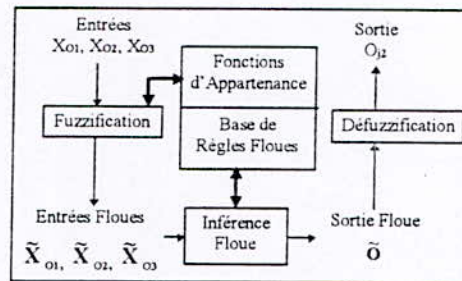


Figure II. 22. Architecture du *SI_2: OBS_NAV_1_Simplifiée-LF*.

Les distances d_D , d_A , et d_G constituent le vecteur d'entrée X_O de ce *SI*:

$$\begin{aligned} X_{O1} &= d_D, \\ X_{O2} &= d_A, \\ X_{O3} &= d_G. \end{aligned} \tag{II. 16}$$

Les composantes du vecteur d'entrée (distances) ont les mêmes variables linguistiques floues Proche (P), Moyen (M), et Loin (L) définies par les fonctions d'appartenance présentées dans la Figure II. 10.

Fuzzification. L'opération de fuzzification calcule le degré des données mesurées membres des trois fonctions d'appartenance pour les trois variables d'entrée p.ex., pour X_{O1} , on obtient $\{\mu_P(X_{O1}), \mu_M(X_{O1}), \mu_L(X_{O1})\}$ représentant les degrés d'appartenance des sous-ensembles flous P, M, et L, respectivement. Ainsi, \tilde{X}_{O1} , \tilde{X}_{O2} , \tilde{X}_{O3} , et \tilde{O} représentent les entrées X_{O1} floue, X_{O2} floue, et X_{O3} floue, ainsi que le vecteur O flou, respectivement.

Base de Règles Floues 1. Un expert humain a établi la base de règles floues présentée

dans le Tableau D. 1 (ANNEXE D) afin d'acquérir le comportement *OBS*.

Les cas particuliers (X_{O1} est P et X_{O2} est P et X_{O3} est P) et (X_{O1} est L et X_{O2} est L et X_{O3} est L) doivent être pris en charge par le système de navigation à un niveau supérieur.

Notons chaque vecteur \tilde{O} dans la base de règles floues comme suit:

$$\tilde{O} = [\mu(X_{O1m}, X_{O2m}, X_{O3m})(O_1), \dots, \mu(X_{O1m}, X_{O2m}, X_{O3m})(O_6)], \quad (\text{II. 17})$$

où $\mu(X_{O1m}, X_{O2m}, X_{O3m})(O_{j2})$ représente le degré d'appartenance de O_{j2} où $m = 1$ ou 2 , voir la simplification présentée ci-après.

Inférence Floue. L'inférence floue est réalisée par l'algorithme, de recherche et d'appariement utilisant les opérateurs MAX-MIN, développé dans [29]. La particularité du paramètre distance est que pour chaque distance donnée le degré d'appartenance, parmi les trois (O_3), qui est égal à zéro ne doit pas être considéré d'où la simplification, voir [127]. A partir de là, pour chaque situation spécifique du véhicule, la correspondance des valeurs des distances sur les intervalles discrets forme les sous-ensembles \tilde{X}_{O1} , \tilde{X}_{O2} , et \tilde{X}_{O3} exprimés par:

$$\begin{aligned} \tilde{X}_{O1} &= \{\mu_1(X_{O1}), \mu_2(X_{O1})\}, \\ \tilde{X}_{O2} &= \{\mu_1(X_{O2}), \mu_2(X_{O2})\}, \\ \tilde{X}_{O3} &= \{\mu_1(X_{O3}), \mu_2(X_{O3})\}, \end{aligned} \quad (\text{II. 18})$$

où $\mu_m(X_{O_i})$, avec $m = 1$ ou 2 , sont les degrés d'appartenance des fonctions d'appartenance correspondant aux distances. Avec cette description, nous avons huit (O_8) conditions possibles correspondant à huit (O_8) règles floues. Le niveau de certitude de chaque triplet de condition μ_1 , ou μ_2 , ..., ou μ_8 peut alors être calculé en utilisant l'opérateur MIN:

$$\begin{aligned} \mu_{\text{cond}}(X_{O11}, X_{O21}, X_{O31}) &= \text{MIN}(\mu_1(X_{O1}), \mu_1(X_{O2}), \mu_1(X_{O3})) = \mu_1, \\ &\dots \\ \mu_{\text{cond}}(X_{O12}, X_{O22}, X_{O32}) &= \text{MIN}(\mu_2(X_{O1}), \mu_2(X_{O2}), \mu_2(X_{O3})) = \mu_8, \end{aligned} \quad (\text{II. 19})$$

où *cond* représente l'ensemble flou des conditions qui est alors écrit comme suit:

$$\text{cond} = \{\mu_1, \dots, \mu_8\}. \quad (\text{II. 20})$$

Chaque triplet possible de condition est associé à un ensemble flou des situations O_{j2} . La certitude de chaque situation est donc obtenue par les opérateurs MAX-MIN comme suit:

$$\mu_{O1} = \text{MAX}\{ \text{MIN}(\mu_1, \mu(X_{O11}, X_{O21}, X_{O31})(O_1)), \dots, \text{MIN}(\mu_8, \mu(X_{O12}, X_{O22}, X_{O32})(O_1)) \},$$

$$\mu_{O_6} = \text{MAX}\{ \text{MIN}(\mu_1, \mu(X_{O_{11}}, X_{O_{21}}, X_{O_{31}})(O_6)), \dots, \text{MIN}(\mu_8, \mu(X_{O_{12}}, X_{O_{22}}, X_{O_{32}})(O_6)) \}, \quad (\text{II. 21})$$

où $\mu(X_{O_{1m}}, X_{O_{2m}}, X_{O_{3m}})(O_{j2})$ est le degré d'appartenance de la situation O_{j2} de la règle floue associée.

La collection des situations forme l'ensemble flou final \tilde{O} :

$$\tilde{O} = \{ \mu_{O_1}(O_1), \dots, \mu_{O_{j2}}(O_{j2}), \dots, \mu_{O_6}(O_6) \} \quad (\text{II. 22})$$

Défuzzification. L'opération de défuzzification utilise l'opérateur MAX, pour obtenir la sortie de ce *SI* correspondant à la situation O_{j2} par :

$$O_{j2} = \text{MAX}\{ \mu_{O_1}(O_1), \dots, \mu_{O_{j2}}(O_{j2}), \dots, \mu_{O_6}(O_6) \} \quad (\text{II. 23})$$

II. 3. 1. 4. *SHI_1: OBS_NAV_1_Simplifiée-RN/AG*

Le *SHI_1: OBS_NAV_1_Simplifiée-RN/AG*, proposé au comportement *OBS*, doit classifier les situations d'évitement d'obstacles en utilisant les *RN*. Cette classification se base sur l'apprentissage supervisé par les *AG* [70, 71, 103, 114, 115], détaillé dans l'ANNEXE C. L'architecture de ce *SHI* est présentée dans la Figure C. 1 (ANNEXE C).

Ce *SHI* est entraîné, voir § III. 3. 1. 4, à partir de données des capteurs à *US* obtenues de l'environnement donnant les distances véhicule-obstacle. Les distances d_D , d_A , et d_G sont prétraitées comme dans l'Equation (II. 15), pour constituer le vecteur d'entrée X_O de ce *SHI*.

II. 3. 1. 5. *SHI_2: OBS_NAV_1_Simplifiée-RN/AG-RPG*

Le *SHI_2: OBS_NAV_1_Simplifiée-RN/AG-RPG*, proposé au comportement *OBS*, doit classifier les situations d'évitement d'obstacles en utilisant les *RN*. Cette classification se base sur l'apprentissage supervisé par *AG-RPG* [71, 114], détaillé dans l'ANNEXE C. L'architecture de ce *SHI* est présentée dans la Figure C. 1 (ANNEXE C).

L'approche proposée consiste à appliquer les *AG* afin d'obtenir une meilleure configuration des poids [103, 115], ensuite la *RPG* est appliquée prenant comme poids initiaux cette configuration des poids.

Ce *SHI* est entraîné, voir § III. 3. 1. 5, à partir de données des capteurs à *US* obtenues de l'environnement donnant les distances véhicule-obstacle. Les distances d_D , d_A , et d_G sont prétraitées comme dans l'Equation (II. 15), pour constituer le vecteur d'entrée X_O de ce *SHI*.

II. 3. 1. 6. *SHI_3: OBS_NAV_1_Simplifiée-RNF/RPG*

Le *SHI_3: OBS_NAV_1_Simplifiée-RNF/RPG* [49], proposé au comportement *OBS*, doit classifier les situations d'évitement d'obstacles en utilisant les *RNF* [47, 73, 118–120]. Cette classification se base sur l'*apprentissage supervisé* par la *RPG*, détaillé dans l'ANNEXE C. L'architecture de ce *SHI* est présentée dans la Figure II. 23.

Ce *SHI* est entraîné, voir § III. 3. 1. 6, à partir de données des capteurs à *US* obtenues de l'environnement donnant les distances véhicule-obstacle. Les distances d_D , d_A , et d_G , comme définies dans l'Equation (II. 16), constituent le vecteur d'entrée \mathbf{X}_O de ce *SHI*.

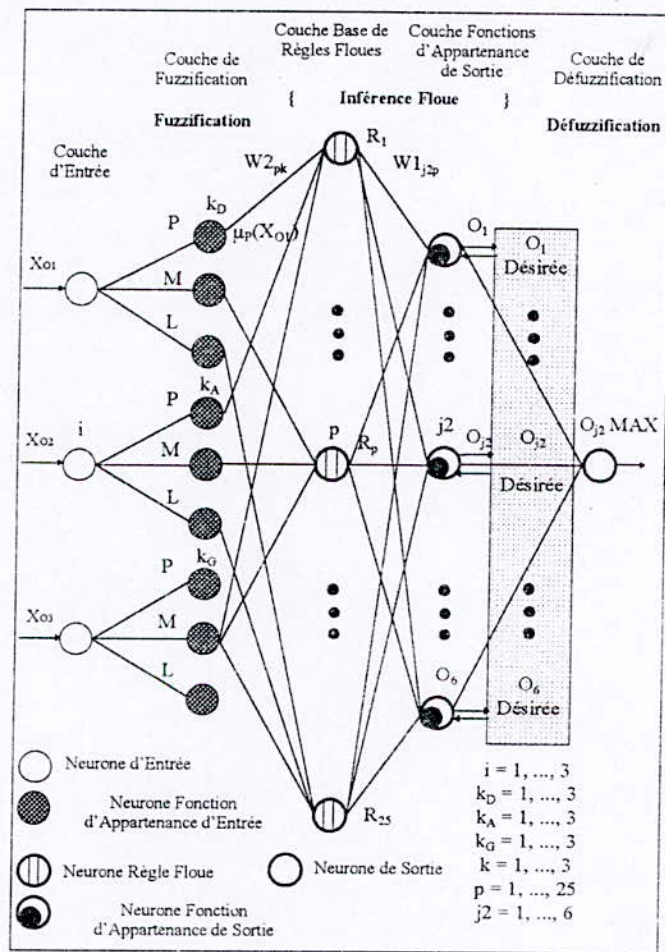


Figure II. 23. Architecture du *SHI_3: OBS_NAV_1_Simplifiée-RNF/RPG*.

Les composantes X_{01} , X_{02} , et X_{03} du vecteur d'entrée ont les mêmes variables linguistiques floues Proche (P), Moyen (M), et Loin (L) définies par les fonctions d'appartenance présentées dans la Figure II. 10, alors que leurs vecteurs flous sont définis par:

$$\tilde{\mathbf{X}}_{01} = \{\mu_P(X_{01}), \mu_M(X_{01}), \mu_L(X_{01})\},$$

$$\tilde{X}_{O2} = \{\mu_P(X_{O2}), \mu_M(X_{O2}), \mu_L(X_{O2}),\},$$

$$\tilde{X}_{O3} = \{\mu_P(X_{O3}), \mu_M(X_{O3}), \mu_L(X_{O3}),\}, \quad (\text{II. 24})$$

avec $\mu_P(X_{O_i})$, $\mu_M(X_{O_i})$, et $\mu_L(X_{O_i})$ les degrés d'appartenance aux sous-ensembles flous P, M, et L, respectivement.

Les règles résultantes de la formulation linguistique floue de la connaissance d'un expert humain sont établies dans la Base de Règles Floues 2 dans le Tableau D. 2 (ANNEXE D).

Les cas particuliers (X_{O1} est P et X_{O2} est P et X_{O3} est P) et (X_{O1} est L et X_{O2} est L et X_{O3} est L) sont pris en charge par le système de navigation à un niveau supérieur.

Ce *SHI* est constitué de cinq (05) couches et présenté dans la Figure II. 23.

Couche d'Entrée: ayant trois (03) neurones d'entrée, elle reçoit les composantes du vecteur d'entrée X_O et transmet ces entrées à leurs fonctions d'appartenance de la couche suivante.

Couche de Fuzzification: ayant neuf (09) neurones fonctions d'appartenance d'entrée, elle représente ainsi les fonctions d'appartenance des entrées. Chaque neurone calcule le degré d'appartenance à un sous-ensemble flou de l'entrée correspondante. Cette couche n'est pas complètement connectée à la couche précédente, les connexions existent seulement entre les neurones d'entrée et les neurones fonctions d'appartenance d'entrée correspondants. Les poids de ses connexions sont mis à 1.

Couche Base de Règles Floues: ayant vingt cinq (25) neurones règles floues, elle représente ainsi la base de règles floues. L'activation des neurones est réalisée par l'opérateur MIN, suivie de la fonction sigmoïde de sortie f:

$$R_p = f(\text{MIN}(W_{2pkD} \mu_{kD}(X_{O1}), W_{2pkA} \mu_{kA}(X_{O2}), W_{2pkG} \mu_{kG}(X_{O3}))) \quad (\text{II. 25})$$

Cette couche n'est également pas complètement connectée à la couche précédente, chaque neurone reçoit seulement trois (03) connexions des neurones fonctions d'appartenance d'entrée correspondants. Les poids W_{2pk} de ces connexions sont ajustés après apprentissage.

Couche Fonctions d'Appartenance de Sortie: ayant six (06) neurones fonctions d'appartenance de la sortie, elle représente ainsi les fonctions d'appartenance de la sortie. L'activation des neurones est réalisée par l'opérateur \sum , suivie de la fonction sigmoïde de sortie f:

$$O_{j2} = f\left(\sum_p R_p W_{1j2p}\right). \quad (\text{II. 26})$$

Cette couche est entièrement connectée à la couche précédente et les poids W_{1j2p} de ces connexions sont ajustés après apprentissage.

Ainsi, l'inférence floue est réalisée par les deux (02) couches: couche base de règles floues et couche fonctions d'appartenance de sortie.

Couche de Défuzzification: ayant un (01) neurone de sortie, elle réalise l'opération de défuzzification par l'opérateur MAX. Cette couche est entièrement connectée à la couche précédente et les poids de ses connexions sont mis à 1. Notons que cette couche n'est utilisée qu'après apprentissage c.-à-d., durant l'application.

II. 3. 1. 7. *SHI_4: OBS_NAV_1_Simplifiée-RNAMF/AMFS*

Le *SHI_4: OBS_NAV_1_Simplifiée-RNAMF/AMFS* [57], proposé au comportement *OBS*, doit classifier les situations d'évitement d'obstacles en utilisant les *RNAMF* [74, 104, 105] voir Figure C. 5 (ANNEXE C). Cette classification se base sur l'*apprentissage supervisé* par la *AMFS* développé par T. Kasuba [74, 121] détaillé dans l'ANNEXE C. L'architecture de ce *SHI*, constituée d'une couche (01) d'entrée 'brute', d'une (01) couche d'entrée, d'une (01) couche des catégories de sortie, et d'une (01) couche de catégories, est présentée dans la Figure II. 24.

Ce *SHI* est entraîné, voir § III. 3. 1. 7, à partir de données des capteurs à *US* obtenues de l'environnement donnant les distances véhicule-obstacle. Les distances d_D , d_A , et d_G sont prétraitées comme dans l'Equation (II. 15), pour constituer le vecteur d'entrée X_O de ce *SHI*.

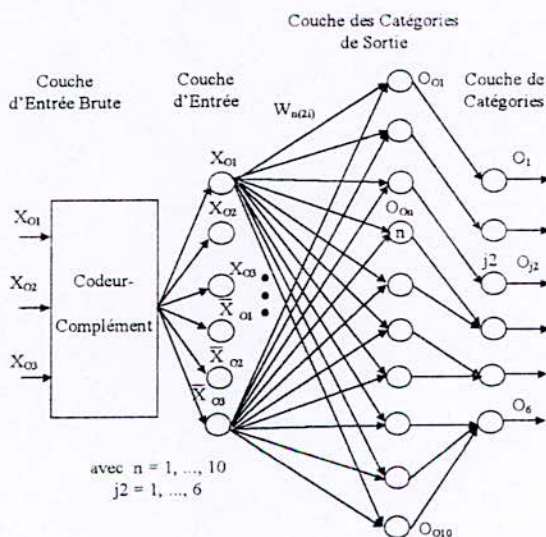


Figure II. 24. Architecture du *SHI_4: OBS_NAV_1_Simplifiée-RNAMF/AMFS*.

II. 3. 2. Stratégie NAV_1_Généralisée

Les *SI* et *SHI* proposés, pour l'acquisition des comportements *CIB* et *OBS*, à la stratégie *NAV_1_Généralisée*, sont cités dans la Figure II. 21.

- **Comportement *CIB***

Le véhicule doit apprendre à reconnaître les situations de position de la cible, présentées dans la Figure II. 4, pour sa localisation ; ce qui lui permettra de se diriger vers cette cible. En fait, le comportement *CIB* se base sur une reconnaissance acquise par *apprentissage* ou/et *adaptation* à partir de données obtenues par la stratégie du champ de température [30]. Cette stratégie est définie dans l'environnement du véhicule et la tâche du véhicule est alors de localiser sa cible correspondant à l'unique température maximale de ce champ. Les températures au voisinage du véhicule sont définies par t_{30} dans la direction 30° , t_{60} dans la direction 60° , t_{90} dans la direction 90° , t_{120} dans la direction 120° , et t_{150} dans la direction 150° . Ces températures sont calculées en utilisant les fonctions sinus et cosinus comme suit:

Si $15^\circ \leq \gamma < 45^\circ$ (Classe C_1)

$$\begin{aligned} \text{Alors } t_{30} &= \cos(\gamma), t_{60} = \sin(\gamma), t_{90} = \sin(\gamma)/2, \\ t_{120} &= \sin(\gamma)/3, \text{ et } t_{150} = \sin(\gamma)/4 \end{aligned}$$

Si $45^\circ \leq \gamma < 75^\circ$ (Classe C_2)

$$\begin{aligned} \text{Alors } t_{30} &= \cos(\gamma), t_{60} = \sin(\gamma), t_{90} = \cos(\gamma), \\ t_{120} &= \cos(\gamma)/2, \text{ et } t_{150} = \cos(\gamma)/3 \end{aligned}$$

Si $75^\circ \leq \gamma < 105^\circ$ (Classe C_3)

$$\begin{aligned} \text{Alors } t_{30} &= \sin(\gamma)/3, t_{60} = \sin(\gamma)/2, t_{90} = \sin(\gamma), \\ t_{120} &= \sin(\gamma)/2, \text{ et } t_{150} = \sin(\gamma)/3 \end{aligned}$$

Si $105^\circ \leq \gamma < 135^\circ$ (Classe C_4)

$$\begin{aligned} \text{Alors } t_{30} &= -\cos(\gamma)/3, t_{60} = -\cos(\gamma)/2, t_{90} = -\cos(\gamma), \\ t_{120} &= \sin(\gamma), \text{ et } t_{150} = -\cos(\gamma) \end{aligned}$$

Si $135^\circ \leq \gamma < 165^\circ$ (Classe C_5)

$$\begin{aligned} \text{Alors } t_{30} &= \sin(\gamma)/4, t_{60} = \sin(\gamma)/3, t_{90} = \sin(\gamma)/2, \\ t_{120} &= \sin(\gamma), \text{ et } t_{150} = -\cos(\gamma) \end{aligned}$$

Si $165^\circ \leq \gamma < 375^\circ$ (Classe C_6)

Si $165^\circ \leq \gamma < 270^\circ$

Alors $t_{30} = -\cos(\gamma)/2$, $t_{60} = -\cos(\gamma)/3$, $t_{90} = -\cos(\gamma)/4$,

$t_{120} = -\cos(\gamma)/3$, et $t_{150} = -\cos(\gamma)$

Si $\gamma = 270^\circ$

Alors $t_{30} = -\sin(\gamma)$, $t_{60} = -\sin(\gamma)/3$, $t_{90} = -\sin(\gamma)/4$,

$t_{120} = -\sin(\gamma)/3$, et $t_{150} = -\sin(\gamma)/2$

Si $270^\circ < \gamma < 375^\circ$

Alors $t_{30} = \cos(\gamma)$, $t_{60} = \cos(\gamma)/3$, $t_{90} = \cos(\gamma)/4$,

$t_{120} = \cos(\gamma)/3$, et $t_{150} = \cos(\gamma)/2$

(II. 27)

Ces composantes sont prétraitées afin de constituer le vecteur d'entrée \mathbf{X}_C :

$$X_{C1} = (1/\rho_C) \exp(-t_{30}/a_C),$$

$$X_{C2} = (1/\rho_C) \exp(-t_{60}/a_C),$$

$$X_{C3} = (1/\rho_C) \exp(-t_{90}/a_C),$$

$$X_{C4} = (1/\rho_C) \exp(-t_{120}/a_C),$$

$$X_{C5} = (1/\rho_C) \exp(-t_{150}/a_C).$$

(II. 28)

Après apprentissage, pour chaque vecteur d'entrée \mathbf{X}_C , ces *SI* et *SHI* devront fournir au véhicule la capacité à décider de la situation de position de la cible C_{j1} exprimée par la sortie C_{j1} ayant la plus grande activation.

• Comportement *OBS*

Ce véhicule doit également apprendre à reconnaître les situations d'évitement d'obstacles, présentées dans la Figure II. 5, ce qui lui permettra de les éviter. En fait, le comportement *OBS* se base sur une reconnaissance acquise par *apprentissage* ou/et *adaptation* à partir de données des capteurs à *US*, présentés dans l'ANNEXE A, donnant les distances véhicule-obstacles. Ces distances sont définies au voisinage du véhicule par: d_{30} dans la direction 30° , d_{60} dans la direction 60° , d_{90} dans la direction 90° , d_{120} dans la direction 120° , et d_{150} dans la direction 150° . Ces composantes sont prétraitées afin de constituer le vecteur d'entrée \mathbf{X}_O :

$$X_{O1} = (1/\rho_O) \exp(-d_{30}/a_O),$$

$$\begin{aligned}
 X_{O2} &= (1/\rho_0) \exp(-d_{60}/a_0), \\
 X_{O3} &= (1/\rho_0) \exp(-d_{90}/a_0), \\
 X_{O4} &= (1/\rho_0) \exp(-d_{120}/a_0), \\
 X_{O5} &= (1/\rho_0) \exp(-d_{150}/a_0).
 \end{aligned}
 \tag{II. 29}$$

Après apprentissage, pour chaque vecteur d'entrée X_O , ces *SI* et *SHI* devront fournir au véhicule la capacité à décider de la situation d'évitement d'obstacles O_{j2} exprimée par la sortie O_{j2} ayant la plus grande activation.

II. 3. 2. 1. *SI_1: CIB_NAV_1_Généralisée-RN/RPG*

Le *SI_1: CIB_NAV_1_Généralisée-RN/RPG* [58], proposé au comportement *CIB*, doit classifier les situations de position de la cible en utilisant les *RN*. Cette classification se base sur l'apprentissage supervisé par la *RPG* détaillé dans l'ANNEXE C. L'architecture de ce *SI* est présentée dans la Figure C. 1 (ANNEXE C).

Ce *SI* est entraîné, voir § III. 3. 2. 1, à partir de données obtenues de la stratégie du champ de température. Les températures t_{30} , t_{60} , t_{90} , t_{120} , et t_{150} sont calculées par l'Equation (II. 27) et prétraitées par l'Equation (II. 28) afin de constituer le vecteur d'entrée X_C de ce *SI*.

II. 3. 2. 2. *SI_1: OBS_NAV_1_Généralisée-RN/RPG*

Le *SI_1: OBS_NAV_1_Généralisée-RN/RPG* [58], proposé au comportement *OBS*, doit classifier les situations d'évitement d'obstacles en utilisant les *RN*. Cette classification se base sur l'apprentissage supervisé par la *RPG* détaillé dans l'ANNEXE C. L'architecture de ce *SI* est présentée dans la Figure C. 1 (ANNEXE C).

Ce *SI* est entraîné, voir § III. 3. 2. 2, à partir de données des capteurs à *US* obtenues de l'environnement donnant les distances véhicule-obstacles. Les distances d_{30} , d_{60} , d_{90} , d_{120} , et d_{150} sont prétraitées par l'Equation (II. 29) afin de constituer le vecteur d'entrée X_O de ce *SI*.

II. 3. 2. 3. *SHI_4: CIB_NAV_1_Généralisée-RNAMF/AMFS*

Le *SHI_4: CIB_NAV_1_Généralisée-RNAMF/AMFS* [59], proposé au comportement *CIB*, doit classifier les situations de position de la cible en utilisant les *RNAMF*. Cette

classification se base sur l'apprentissage supervisé par la AMFS détaillé dans l'ANNEXE C. L'architecture de ce SHI, composée d'une (01) couche d'entrée 'brute', d'une (01) couche d'entrée, d'une (01) couche des catégories de sortie, et d'une (01) couche de catégories, est présentée dans la Figure II. 25. (a).

Ce SHI est entraîné, voir § III. 3. 2. 3, à partir de données obtenues de la stratégie du champ de température. Les températures t_{30} , t_{60} , t_{90} , t_{120} , et t_{150} sont calculées par l'Equation (II. 27) et prétraitées par l'Equation (II. 28) afin de constituer le vecteur d'entrée X_C de de SHI.

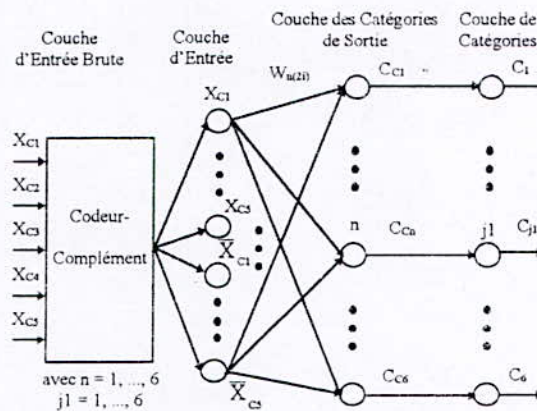


Figure II. 25. (a). Architecture du SHI_4: CIB_NAV_1_Généralisée-RNAMF/AMFS.

II. 3. 2. 4. SHI_4: OBS_NAV_1_Généralisée-RNAMF/AMFS

Le SHI_4: OBS_NAV_1_Généralisée-RNAMF/AMFS [59], proposé au comportement OBS, doit classifier les situations d'évitement d'obstacles en utilisant les RNAMF. Cette classification se base sur l'apprentissage supervisé par la AMFS détaillé dans l'ANNEXE C. L'architecture de ce SHI, composée d'une (01) couche d'entrée 'brute', d'une (01) couche d'entrée, d'une couche (01) des catégories de sortie, et d'une (01) couche de catégories, est présentée dans la Figure II. 25. (b).

Ce SHI est entraîné, voir § III. 3. 2. 4, à partir de données des capteurs à US obtenues de l'environnement donnant les distances véhicule-obstacles. Les distances d_{30} , d_{60} , d_{90} , d_{120} , et d_{150} sont prétraitées par l'Equation (II. 29) afin de constituer le vecteur d'entrée X_O de ce SHI.

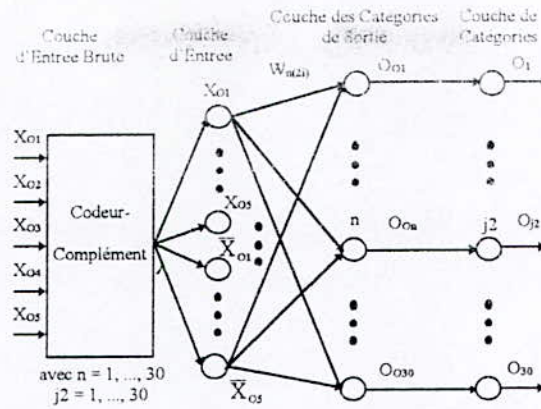


Figure II. 25. (b). Architecture du *SHI_4: OBS_NAV_1_Généralisée-RNAMF/AMFS*.

II. 3. 3. Stratégie *NAV_2_Généralisée*

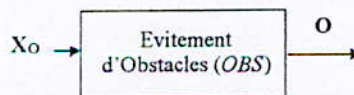
- **Comportement *CIB***

Le véhicule doit apprendre à reconnaître les situations de position de la cible, présentées dans la Figure II. 7, pour sa localisation ; ce qui lui permettra de se diriger vers cette cible. Le traitement proposé, pour l'acquisition du comportement *CIB*, à la stratégie *NAV_2_Généralisée*, calcule le vecteur $C = [C_1, \dots, C_i, \dots, C_{11}]$ à partir des vecteurs position de la cible P_C et position du véhicule P_V .

Notons, que ce comportement devrait être lui aussi acquis, par *apprentissage* ou/et *adaptation*, de la même manière que le *SI_1: CIB_NAV_1_Généralisée-RN/RPG* présenté dans § II. 3. 2. 1 ou le *SHI_4: CIB_NAV_1_Généralisée-RNAMF/AMFS* présenté dans § II. 3. 2. 3.

- **Comportement *OBS***

Les *SHI* proposés, pour l'acquisition du comportement *OBS*, à la stratégie *NAV_2_Généralisée*, sont cités dans la Figure II. 26.



SHI_3: OBS_NAV_2_Généralisée-RNF/RPG: Degrés du Danger Statique
SHI_3: OBS_NAV_2_Généralisée-RNF/RPG: Degrés du Danger Dynamique

Figure II. 26. *SHI* pour le Comportement *OBS (NAV_2_Généralisée)*.

Le véhicule doit également apprendre à reconnaître les degrés de danger statique et dynamique de chaque point obstacle, présentés dans la Figure II. 11, ce qui lui permettra de les éviter.

En fait, il est important de noter que la reconnaissance du degré de danger statique se base sur une acquisition, par *apprentissage* et *adaptation*, à partir de données des capteurs à *US*, présentés dans l'ANNEXE A, donnant les distances véhicules-obstacles. Tandis que la reconnaissance du degré de danger dynamique se base sur une acquisition, par *apprentissage* et *adaptation*, à partir de données des capteurs donnant les vitesses des obstacles, ce type de capteurs est différent de celui présenté dans l'ANNEXE A.

Après apprentissage, pour chaque vecteur d'entrée X_O , ces *SHI* devront fournir au véhicule la capacité à décider de son évitement d'obstacles, reconnaissant le degré de danger statique O_{α_j} (ou dynamique O_{β_j}) exprimée par la sortie O_{α_j} (ou O_{β_j}) ayant la plus grande activation.

II. 3. 3. 1. *SHI_3: OBS_NAV_2_Généralisée-RNF/RPG: Degrés du Danger Statique*

Le *SHI_3: OBS_NAV_2_Généralisée-RNF/RPG* (Degrés du Danger Statique) [54, 60], proposé au comportement *OBS*, doit classifier les degrés du danger statique O_{α_j} en utilisant les *RNF*. Cette classification se base sur l'*apprentissage supervisé* par la *RPG* détaillé dans l'ANNEXE C. L'architecture de ce *SHI*, constituée de cinq (05) couches, est présentée dans la Figure II. 27. (a). Notons que O_{α_1} , O_{α_2} , O_{α_3} , O_{α_4} , O_{α_5} , O_{α_6} , O_{α_7} , et O_{α_8} correspondent à DGG, DGP, SGP, SGG, DDG, DDP, SDP, et SDG, respectivement.

Ce *SHI* est entraîné, voir § III. 3. 3. 1, à partir des formulations linguistiques floues. La direction θ_s et la distance L_s pour le degré du danger statique O_{α_j} , constituent les composantes X_{O1} et X_{O2} , du vecteur d'entrée X_O de ce *SHI*, définies dans l'Equation (II. 30).

$$\begin{aligned} X_{O1} &= \theta_s, \\ X_{O2} &= L_s, \\ X_{O3} &= \Phi_U, \\ X_{O4} &= W_U. \end{aligned} \tag{II. 30}$$

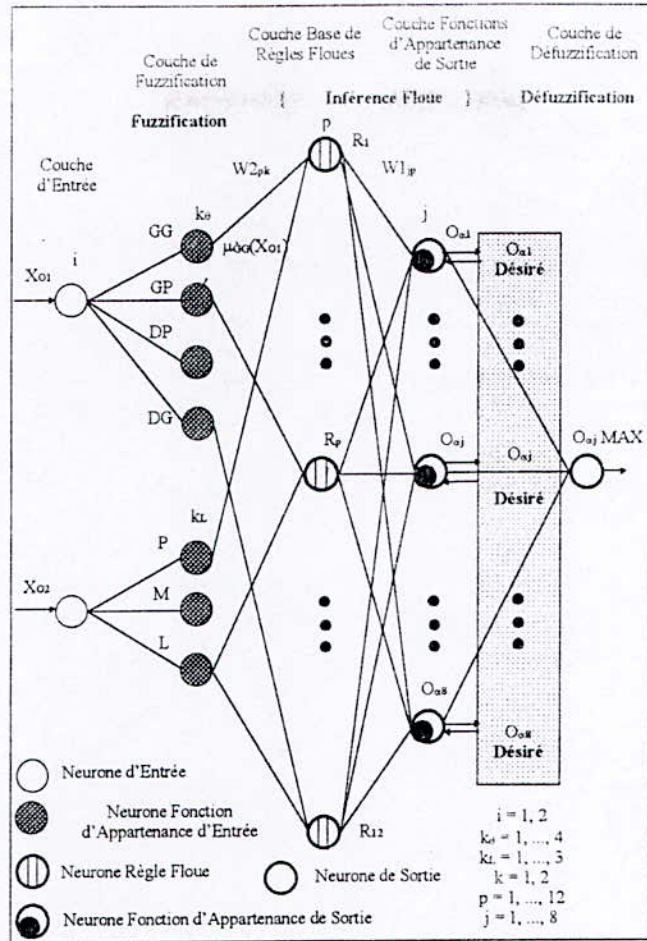


Figure II. 27. (a). Architecture du SHI_3:

OBS_NAV_2_Généralisée-RNF/RPG: Degrés du Danger Statique.

Les composantes du vecteur d'entrée X_{01} et X_{02} ont les variables linguistiques floues définies par les fonctions d'appartenance présentées dans la Figure II. 9 et la Figure II. 10, respectivement, alors que leurs vecteurs flous sont définis par:

$$\begin{aligned} \tilde{X}_{01} &= \{\mu_{GG}(X_{01}), \mu_{GP}(X_{01}), \mu_{DP}(X_{01}), \mu_{DG}(X_{01})\}, \\ \tilde{X}_{02} &= \{\mu_P(X_{02}), \mu_M(X_{02}), \mu_L(X_{02})\}, \end{aligned} \quad (II. 31)$$

avec $\mu_{GG}(X_{0i}), \mu_{GP}(X_{0i}), \mu_{DP}(X_{0i}),$ et $\mu_{DG}(X_{0i})$ les degrés d'appartenance aux sous-ensembles flous GG, GP, DP, et DG, respectivement ; alors que $\mu_P(X_{0i}), \mu_M(X_{0i}),$ et $\mu_L(X_{0i})$ les degrés d'appartenance aux sous-ensembles flous P, M, et L, respectivement.

Les règles résultantes de la formulation linguistique floue de la connaissance d'un expert humain sont établies dans la Base de Règles Floues 3 [32, 54, 60] dans le Tableau D. 3 (ANNEXE D).

II. 3. 3. 2. *SHI_3: OBS_NAV_2_Généralisée-RNF/RPG: Degrés du Danger Dynamique*

Le *SHI_3: OBS_NAV_2_Généralisée-RNF/RPG* (Degrés du Danger Dynamique) [54, 60], proposé au comportement *OBS*, doit classifier les degrés du danger dynamique O_{β_j} en utilisant les *RNF*. Cette classification se base sur l'apprentissage supervisé par la *RPG* détaillé dans l'ANNEXE C. L'architecture de ce *SHI*, constituée de cinq (05) couches, est présentée dans la Figure II. 27. (b). Notons que O_{β_1} , O_{β_2} , O_{β_3} , O_{β_4} , O_{β_5} , O_{β_6} , O_{β_7} , et O_{β_8} correspondent à DGG, DGP, SGP, SGG, DDG, DDP, SDP, et SDG, respectivement.

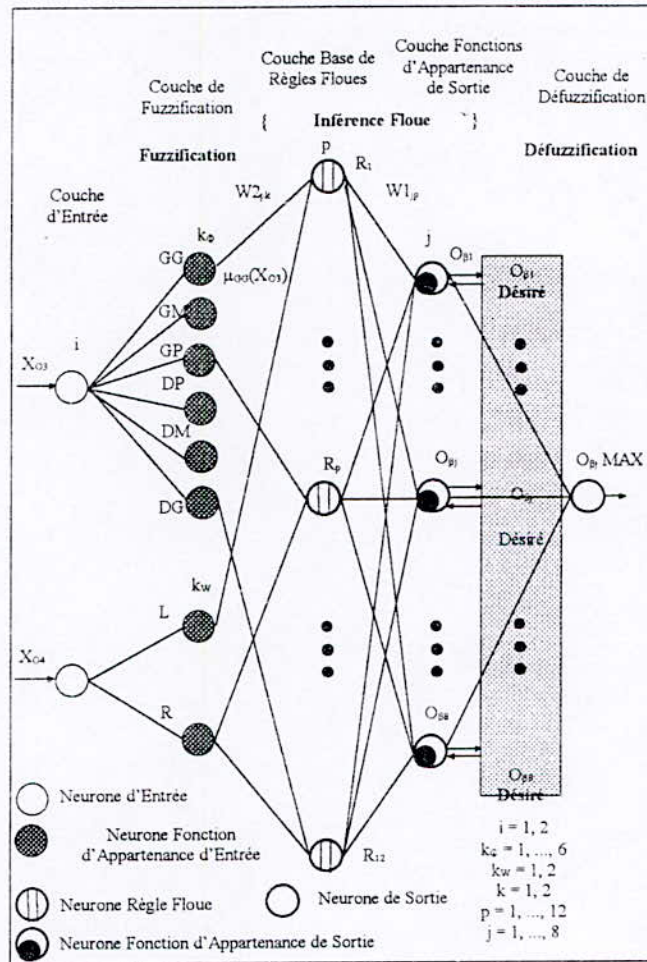


Figure II. 27. (b). Architecture du *SHI_3*:

OBS_NAV_2_Généralisée-RNF/RPG: Degrés du Danger Dynamique.

Ce *SHI* est entraîné, voir § III. 3. 3. 2, à partir des formulations linguistiques floues. La direction de la vitesse Φ_U et la vitesse W_U pour le degré du danger dynamique O_{β_j} constituent les composantes X_{O3} et X_{O4} , du vecteur d'entrée X_O de ce *SHI*, définies dans l'Equation (II. 30).

Les composantes du vecteur d'entrée X_{O3} et X_{O4} ont les variables linguistiques floues définies par les fonctions d'appartenance présentées dans la Figure II. 12 et la Figure II. 13, respectivement, alors que leurs vecteurs flous sont définis par:

$$\begin{aligned}\tilde{X}_{O3} &= \{\mu_{GG}(X_{O3}), \mu_{GM}(X_{O3}), \mu_{GP}(X_{O3}), \mu_{DP}(X_{O3}), \mu_{DM}(X_{O3}), \mu_{DG}(X_{O3})\}, \\ \tilde{X}_{O4} &= \{\mu_L(X_{O4}), \mu_R(X_{O4})\},\end{aligned}\tag{II. 32}$$

avec $\mu_{GG}(X_{O_i})$, $\mu_{GM}(X_{O_i})$, $\mu_{GP}(X_{O_i})$, $\mu_{DP}(X_{O_i})$, $\mu_{DM}(X_{O_i})$, et $\mu_{DG}(X_{O_i})$ les degrés d'appartenance aux sous-ensembles flous GG, GM, GP, DP, DM, et DG, respectivement ; alors que $\mu_L(X_{O_i})$, et $\mu_R(X_{O_i})$ les degrés d'appartenance aux sous-ensembles flous L et R, respectivement.

Les règles résultantes de la formulation linguistique floue de la connaissance d'un expert humain sont établies dans la Base de Règles Floues 4 [32, 54, 60] dans le Tableau D. 4 (ANNEXE D).

II. 4. SI PROPOSES POUR L'ACQUISITION DU COMPORTEMENT DE PRISE DE DECISIONS (DEC)

Nous proposons, dans ce Paragraphe, différents SI pour l'acquisition, par *apprentissage*, du comportement DEC pour les stratégies *NAV_1_Généralisée* et *NAV_2_Généralisée*.

Les SI proposés, pour l'acquisition du comportement DEC, aux stratégies *NAV_1_Généralisée* et *NAV_2_Généralisée*, sont cités dans la Figure II. 28.

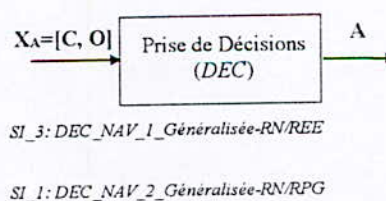


Figure II. 28. SI pour le Comportement DEC (*NAV_1_Généralisée* et *NAV_2_Généralisée*).

II. 4. 1. SI_3: DEC_NAV_1_Généralisée-RN/REE

Le véhicule doit apprendre à décider de l'action appropriée A_i à partir des résultats des comportements *CIB* et *OBS*. Ainsi, le *SI_3: DEC_NAV_1_Généralisée-RN/REE* [58], proposé

au comportement *DEC*, doit associer les comportements *CIB* et *OBS* avec les actions possibles du véhicule, et coordonner ces associations afin de décider de l'action appropriée en utilisant un *RN*. Ces associations et cette coordination se base sur l'*apprentissage supervisé* par le *REE* [30] détaillé dans l'ANNEXE C. L'architecture de ce *SI*, constituée de deux (02) couches (couche d'entrée et couche de sortie), est présentée dans la Figure II. 29.

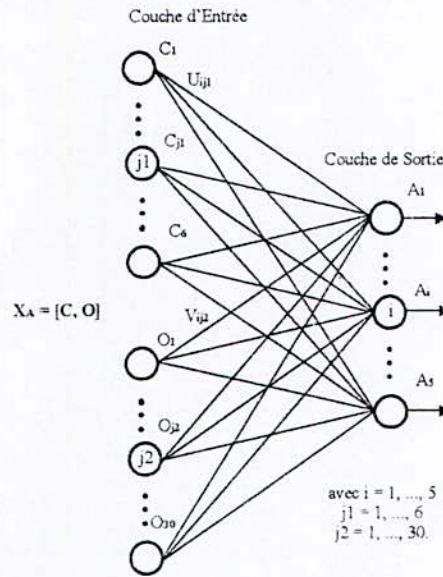


Figure II. 29. Architecture du *SI_3: DEC_NAV_1_Généralisée-RN/REE*.

En effet, d'une part, ces deux (02) étapes d'association des comportements *CIB* et *OBS* aux actions favorables (parmi les actions possibles) sont prises en charge séparément par ce *SI*:

- étape d'association entre le comportement *CIB* et les actions favorables: association des situations C_{j1} aux actions favorables A_i par apprentissage par le *REE*,
- étape d'association entre le comportement *OBS* et les actions favorables: association des situations O_{j2} aux actions favorables A_i par apprentissage par le *REE*,

D'autres part, la coordination de ces étapes d'association permet alors la prise de décisions de l'action appropriée.

Après apprentissage, ce *SI* devra fournir au véhicule la capacité à décider de l'action appropriée exprimée par la sortie A_i ayant la plus grande activation.

Ce *SI* est entraîné, voir § III. 4. 1, à partir de données d'un expert humain ainsi que celles de capteurs à *US* obtenues de l'environnement. Les composantes de la matrice d'entrée X_A sont le vecteur **C** et le vecteur **O**: $X_A = [C, O]$.

Couche d'Entrée: cette couche est constituée de trente six (36) neurones d'entrée recevant les composantes des vecteurs C et O . Cette couche transmet ces entrées à tous les neurones de la couche suivante. Chaque neurone C_{j1} est connecté à tous les neurones A_i avec les poids de connexion U_{ij1} , voir § III. 4. 1. 1, et chaque neurone O_{j2} est connecté à tous les neurones A_i avec les poids de connexion V_{ij2} , voir § III. 4. 1. 2.

Couche de Sortie: cette couche est constituée de cinq (05) neurones de sortie qui sont obtenues par:

$$A_i = g \left(\sum_{j1} C_{j1} U_{ij1} + \sum_{j2} O_{j2} V_{ij2} \right) + N_i, \quad (\text{II. 33})$$

$$\text{avec } g(x) = \begin{cases} x & \text{si } x > 0 \\ 0 & \text{ailleurs} \end{cases}, \quad (\text{II. 34})$$

où N_i est une variable de distribution aléatoire entre $[0, \beta]$, où β est une constante définie dans § III. 4. 1. 3.

II. 4. 1. 1. Association du Comportement CIB aux Actions Favorables

L'entraînement dans cette étape, voir § III. 4. 1. 1, est guidé seulement par un processus de retour, c.-à-d., guidé par un signal P_1 fourni par le superviseur: un expert humain. Les situations de position de la cible sont alors associées aux actions favorables dans un environnement libre d'obstacles (c.-à-d., $O = 0$).

Les actions favorables sont définies, pour chaque situation C_{j1} , par l'expert humain (superviseur) qui fournit le vecteur $Z = [Z_1, Z_2, Z_3, Z_4, Z_5]$ où chaque composante Z_i est déterminée par rapport à chaque action possible A_i . Si $Z_i = 1$ alors A_i est une action favorable, tandis que si $Z_i = 0$ alors A_i est une action défavorable. Pour chaque situation C_{j1} seulement les composantes $Z_i = 1$ du vecteur Z correspondant aux actions favorables sont représentées dans la Figure II. 30.

Par exemple, pour la situation T_3 : $Z_1 = 0, Z_2 = 1, Z_3 = 1, Z_4 = 1$, et $Z_5 = 0$ ce qui veut dire que seulement A_2, A_3 , et A_4 sont considérées comme des actions favorables tandis que A_1 et A_5 sont considérées comme des actions défavorables.

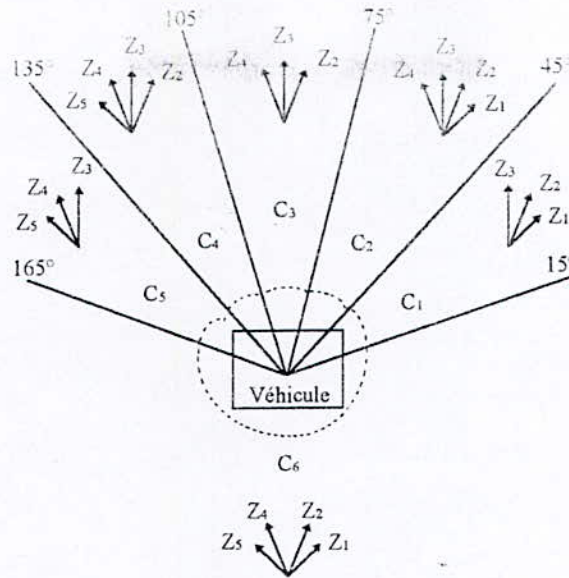


Figure II. 30. Représentation des Actions Favorables pour les Situations de Position de la Cible (NAV_1_Généralisée).

II. 4. 1. 2. Association du Comportement *OBS* aux Actions Favorables

L'entraînement dans cette étape, voir § III. 4. 1. 2, est guidé seulement par un processus de retour, c.-à-d., guidé par un signal P_2 fourni par le superviseur: données de capteurs à *US* obtenues de l'environnement. Les situations d'évitement d'obstacles sont alors associées aux actions favorables sans considérer le champ de température (c.-à-d., $C = 0$).

Pour chaque situation O_{j2} , les actions favorables sont définies par celles qui correspondent aux directions dans lesquelles aucun obstacle n'est détecté (aucune collision), tandis que les actions défavorables sont définies par celles qui correspondent aux directions où un obstacle est détecté (collision). Par exemple, pour la situation O_{23} présentée dans la Figure II. 5: seulement A_1 et A_3 sont considérées comme actions favorables alors que A_2 , A_4 , et A_5 sont considérées comme actions défavorables.

II. 4. 1. 3. Coordination des Comportements *CIB* et *OBS*

La coordination doit fournir au véhicule la capacité à accomplir, en même temps, les deux comportements *CIB* et *OBS* et à décider de l'action appropriée A_i . Afin d'assurer la coordination des deux (02) étapes d'association, voir § III. 4. 1. 3, l'action appropriée A_i est obtenue à partir de l'Equation (II. 33).

II. 4. 2. *SI_1: DEC_NAV_2_Généralisée-RN/RPG*

Pour chaque point obstacle, le véhicule doit apprendre à décider de la direction d'évitement O_i , et donc du vecteur direction d'évitement $\mathbf{O}(\text{Obs})$ à partir des résultats du comportement OBS : $O_{\alpha_j}(\text{Obs})$ et $O_{\beta_j}(\text{Obs})$. Ainsi, le *SI_1: DEC_NAV_2_Généralisée-RN/RPG* [54, 60], proposé au comportement *DEC*, doit classifier la direction d'évitement O_i ($i = 1, \dots, 11$) (parmi les mouvements possibles A_i du véhicule) en utilisant un *RN*. Rappelons que les onze (11) mouvements possibles du véhicule $\mathbf{A} = [+5, +4, +3, +2, +1, 0, -1, -2, -3, -4, -5]$, présentés dans la Figure II. 7, correspondent aux directions $75^\circ, 60^\circ, 45^\circ, 30^\circ, 15^\circ, 0^\circ, -15^\circ, -30^\circ, -45^\circ, -60^\circ, -75^\circ$, respectivement. Cet classement se base sur l'apprentissage supervisé par la *RPG*, présenté dans l'ANNEXE C. L'architecture de ce *SI* est présentée dans la Figure II. 31.

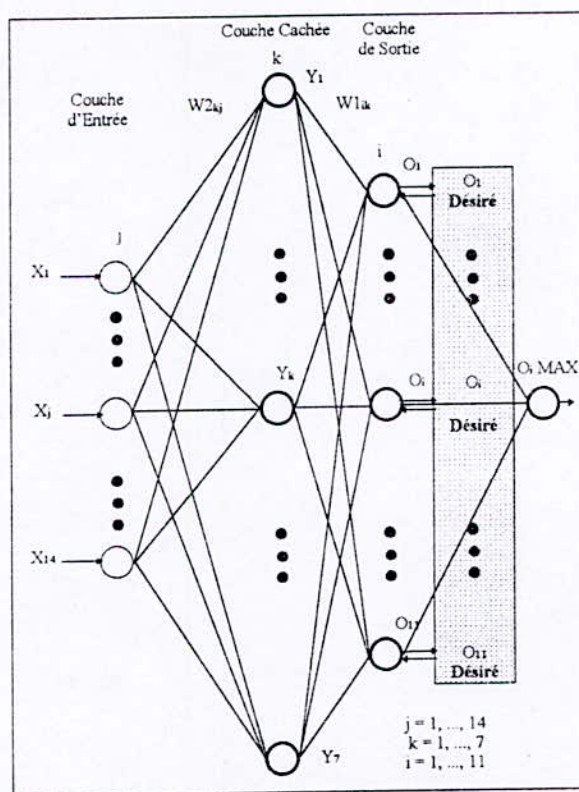


Figure II. 31. Architecture du *SI_1: DEC_NAV_2_Généralisée-RN/RPG*.

Ce *SI* est entraîné, voir § III. 4. 2, à partir des degrés des dangers statique et dynamique codés ainsi que la formulation d'un expert humain établie, dans le Tableau II. 1, pour la reconnaissance des directions d'évitement [32].

Tableau II. 1. Décision de la Direction d'Évitement.

		Danger			Sécurité				Danger		
$\beta(\text{Obs}) \backslash \alpha(\text{Obs})$	-1	-0.8	-0.6	-0.4	-0.2	0	0.2	0.4	0.6	0.8	1
Danger -1	+3	+2	+1	0	0	0	-2	-3	+4	+5	
-0.8	+2	+1	0	0	0	0	-1	-1	+3	+4	
-0.6	+1	0	0	0	0	0	0	0	-1	-2	
-0.4	0	0	0	0	0	0	0	0	0	-1	
-0.2	0	0	0	0	0	0	0	0	0	0	
Sécurité 0	0	0	0	0	0	0	0	0	0	0	
0.2	0	0	0	0	0	0	0	0	0	0	
0.4	+1	0	0	0	0	0	0	0	0	0	
0.6	+2	+1	0	0	0	0	0	0	0	-1	
0.8	-4	-3	+1	+1	0	0	0	0	-1	-2	
Danger 1	-5	-4	+3	+2	0	0	-0	-1	-2	-3	

En fait, les composantes $O_{\alpha j}$ et $O_{\beta j}$ sont codées en utilisant un code binaire sur sept (07) bits résultant en sept (07) composantes $x_{\alpha 1}, \dots, x_{\alpha 7}$ relatives à $O_{\alpha j}$ et en sept (07) autres $x_{\beta 1}, \dots, x_{\beta 7}$ relatives à $O_{\beta j}$. Ces composantes sont alors prétraitées afin de constituer les composantes du vecteur d'entrées \mathbf{X} du SI :

$$X_1 = (1/\rho_X) \exp(-x_{\alpha 1}/a_X), \dots, X_7 = (1/\rho_X) \exp(-x_{\alpha 7}/a_X),$$

$$X_8 = (1/\rho_X) \exp(-x_{\beta 1}/a_X), \dots, X_{14} = (1/\rho_X) \exp(-x_{\beta 7}/a_X), \quad (\text{II. 35})$$

où ρ_X : norme du vecteur d'entrée \mathbf{X} et a_X : facteur de prétraitement d'entrée avec $a_X > 1$.

Après apprentissage, pour chaque vecteur d'entrée \mathbf{X} , ce SI devra fournir au véhicule la capacité à décider de sa direction d'évitement, reconnaissant la direction d'évitement O_i exprimée par la sortie O_i ayant la plus grande activation, et donc du vecteur direction d'évitement $\mathbf{O}(\text{Obs})$ pour chaque point obstacle: $\mathbf{O}(\text{Obs1}), \mathbf{O}(\text{Obs2}), \text{etc.}$

Notons que la décision de l'action appropriée A_i est obtenue, voir § II. 2. 2. 2, à partir des résultats des comportements CIB et DEC , vecteur \mathbf{C} et vecteur $\mathbf{O}(\text{Obstacles})$, respectivement. D'une part, le vecteur \mathbf{C} est obtenu par un traitement, voir § II. 2. 2 et II. 3. 3. D'autres part, le vecteur $\mathbf{O}(\text{Obstacles})$ est obtenu par la *Procédure 2* à partir des vecteurs directions d'évitement $\mathbf{O}(\text{Obs1}), \mathbf{O}(\text{Obs2}), \text{etc.}$, voir § II. 2. 2. 2.

II. 5. SI PROPOSE POUR L'ACQUISITION DU COMPORTEMENT D'ACTION (ACT): SI₂: ACT_NAV_1_NAV_2_Généralisées-LF

Nous proposons, dans ce §, un SI pour l'acquisition, par *adaptation*, du comportement ACT pour les stratégies NAV_1_Généralisée et NAV_2_Généralisée.

Le véhicule doit apprendre à exécuter l'action appropriée A_i résultat des comportements CIB, OBS, et DEC.

Le SI proposé, pour l'acquisition du comportement ACT, aux stratégies NAV_1_Généralisée et NAV_2_Généralisée, est cité dans la Figure II. 32.

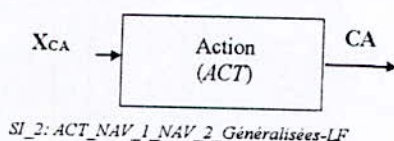


Figure II. 32. SI pour le Comportement ACT (NAV_1_NAV_2_Généralisées).

Le SI₂: ACT_NAV_1_NAV_2_Généralisées-LF [29, 40], proposé au comportement ACT, doit classifier les commandes d'action CA_k, dans le cas de l'AGV. Cette classification se base sur une *adaptation 'supervisée'*, voir § I. 5. 2. L'architecture de ce SI est présentée dans la Figure II. 33.

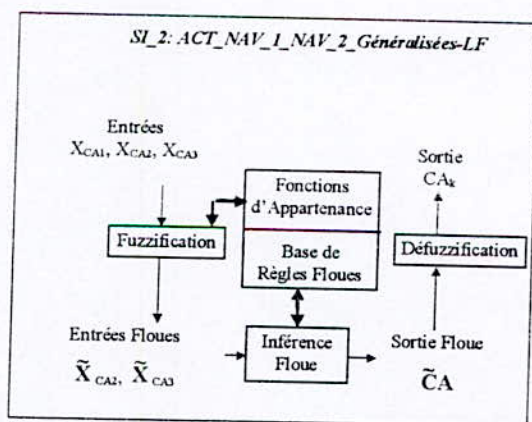


Figure II. 33. Architecture du SI₂: ACT_NAV_1_NAV_2_Généralisées-LF.

La vitesse finale V_{finale} , la vitesse initiale $V_{initiale}$ (vitesse courante $V_{courante}$), et la distance-but d_{but} (distance courante $d_{courante}$), voir § II. 2. 3, constituent les composantes du vecteur d'entrée X_{CA} de ce SI:

$$X_{CA1} = V_{finale},$$

$$X_{CA2} = V_{initiale}, V_{courante}$$

$$X_{CA3} = d_{but}, d_{courante} \quad (II. 36)$$

Ces données d'entrée doivent être recueillies, à au haut niveau de contrôle, à partir du système de navigation contrôlant les comportements intelligents *CIB*, *OBS*, et *DEC*.

Fuzzification. L'opération de fuzzification calcule le degré des données mesurées membres des trois fonctions d'appartenance pour les trois variables d'entrée p.ex., pour X_{CA1} , on obtient $\{\mu_{TL}(X_{CA1}), \mu_L(X_{CA1}), \mu_R(X_{CA1}), \mu_{TR}(X_{CA1})\}$ représentant les degrés d'appartenance des sous-ensembles flous TL, L, R, et TR, respectivement. Ainsi, les \tilde{X}_{CA1} , \tilde{X}_{CA2} , \tilde{X}_{CA3} , et \tilde{CA} représentent les entrées X_{CA1} floue, X_{CA2} floue, et X_{CA3} floue, ainsi que le vecteur CA flou, respectivement.

Base de Règles Floues 5. Dans le but d'établir cette base de règles floues des expériences ont été effectuées et ont données les valeurs des distances d'accélération/décélération, pour chaque combinaison de la vitesse finale, la vitesse initiale, et la distance-but présentées dans le Tableau D. 5 (ANNEXE D). A partir des résultats de ces expériences, les règles floues de contrôle de mouvements sont dérivées: Base de Règles Floues 5 présentée dans le Tableau D. 6 (ANNEXE D). Notons que cette base de règles floue est établie pour les vitesses finales désirées $V_0 = 0$ m/s, $V_1 = 0.1$ m/s, $V_2 = 0.2$ m/s, $V_3 = 0.3$ m/s, et $V_4 = 0.4$ m/s.

Notons chaque vecteur \tilde{CA} dans la base de règles floues comme suit:

$$\tilde{CA} = [\mu_{(X_{CA2m}, X_{CA3m})}(CA_1), \dots, \mu_{(X_{CA2m}, X_{CA3m})}(CA_6)], \quad (II. 37)$$

où $\mu_{(X_{CA2m}, X_{CA3m})}(CA_k)$ représente le degré d'appartenance de CA_k où $m = 1$ ou 2 , voir la simplification présentée dans § II. 3. 1. 3.

Inférence Floue. L'inférence floue est réalisée par un algorithme de recherche et d'appariement, utilisant les opérateurs MIN-MAX, développé dans [29]. Pour chaque situation spécifique du véhicule, la correspondance des valeurs des vitesses initiales et des distances-buts sur les intervalles discrets forme les sous-ensembles \tilde{X}_{CA2} , et \tilde{X}_{CA3} exprimés par:

$$\begin{aligned} \tilde{X}_{CA2} &= \{\mu_1(X_{CA2}), \mu_2(X_{CA2})\}, \\ \tilde{X}_{CA3} &= \{\mu_1(X_{CA3}), \mu_2(X_{CA3})\}, \end{aligned} \quad (II. 38)$$

où $\mu_m(X_{CAi})$, avec $m = 1$ ou 2 , sont les degrés d'appartenance des fonctions d'appartenance correspondant aux vitesses initiales et aux distances-buts. Avec cette description, nous avons quatre (04) conditions possibles correspondant à quatre (04) règles floues. Le niveau de

certitude de chaque paire de condition μ_1 , ou μ_2 , ..., ou μ_4 peut alors être calculé en utilisant l'opérateur MIN:

$$\begin{aligned} \mu_{\text{cond}}(X_{CA21}, X_{CA31}) &= \text{MIN}(\mu_1(X_{CA2}), \mu_1(X_{CA3})) = \mu_1, \\ &\dots \\ \mu_{\text{cond}}(X_{CA22}, X_{CA32}) &= \text{MIN}(\mu_2(X_{CA2}), \mu_2(X_{CA3})) = \mu_2, \end{aligned} \quad (\text{II. 39})$$

où cond représente l'ensemble flou des conditions qui est alors écrit comme suit:

$$\text{cond} = \{\mu_1, \dots, \mu_4\}. \quad (\text{II. 40})$$

Chaque paire possible de condition est associée à un ensemble flou des situations CA_k . La certitude de chaque situation est donc obtenue par les opérateurs MAX-MIN comme suit:

$$\begin{aligned} \mu_{CA1} &= \text{MAX}\{ \text{MIN}(\mu_1, \mu(X_{CA21}, X_{CA31})(CA_1)), \dots, \text{MIN}(\mu_4, \mu(X_{CA22}, X_{CA32})(CA_1)) \}, \\ &\dots \\ \mu_{CA6} &= \text{MAX}\{ \text{MIN}(\mu_1, \mu(X_{CA21}, X_{CA31})(CA_6)), \dots, \text{MIN}(\mu_4, \mu(X_{CA22}, X_{CA32})(CA_6)) \}, \end{aligned} \quad (\text{II. 41})$$

où $\mu(X_{CA2m}, X_{CA3m})(CA_k)$ est le degré d'appartenance de la situation CA_k de la règle floue associée.

La collection des situations forme l'ensemble flou final \tilde{CA} :

$$\tilde{CA} = \{ \mu_{CA1}(CA_1), \dots, \mu_{CAk}(CA_k), \dots, \mu_{CA6}(CA_6) \} \quad (\text{II. 42})$$

Défuzzification. L'opération de défuzzification utilise l'opérateur MAX pour obtenir la sortie du SI correspondant à la situation CA_k par:

$$CA_k = \text{MAX}\{ \mu_{CA1}(CA_1), \dots, \mu_{CAk}(CA_k), \dots, \mu_{CA6}(CA_6) \} \quad (\text{II. 43})$$

Pour chaque vecteur d'entrée X_{CA} , ce SI devra fournir au véhicule la capacité à décider de la commande d'action CA_k appropriée.

II. 6. SYSTEMES HYBRIDES INTELLIGENTS (SHI) PROPOSES POUR LA PLANIFICATION ET LE CONTROLE DE LA NAVIGATION

Nous proposons, dans ce Paragraphe, différents SHI pour la planification et le contrôle des stratégies de navigation *NAV_1_Généralisée* et *NAV_2_Généralisée*. En fait, les meilleurs

SI ou SHI, en termes de performance et de facilité de mise en oeuvre, pour chaque comportement intelligent ont été sélectionnés pour chaque stratégie, voir § III. 3, III. 4, III. 5, et IV. 2. 1.

Ainsi, les SHI appelés *SHI_A_NAV_1_Généralisée*, *SHI_B_NAV_1_Généralisée*, ainsi que *SHI_SECOV_NAV_2_Généralisée*, Système Expert pour le COntôle des Véhicules (*SECOV*), sont proposés afin de planifier et de contrôler la navigation des VAI en environnements dynamiques.

II. 6. 1. Stratégie NAV_1_Généralisée

II. 6. 1. 1. SHI_A_NAV_1_Généralisée

Le *SHI_A_NAV_1_Généralisée*, voir Figure II. 34, est constitué des Systèmes Intelligents (SI) suivants:

Comportement CIB: Classification de situations de position de la cible par le *SI_1: CIB_NAV_1_Généralisée-RN/RPG*.

Comportement OBS: Classification de situations d'évitement d'obstacles par le *SI_1: OBS_NAV_1_Généralisée-RN/RPG*,

Comportement DEC: Décision de l'action appropriée A_i (associations et coordination) par le *SI_3: DEC_NAV_1_Généralisée-RN/REE*,

Comportement ACT: Exécution de l'action A_i par le *SI_2: ACT_NAV_1_NAV_2_Généralisées-LF*.

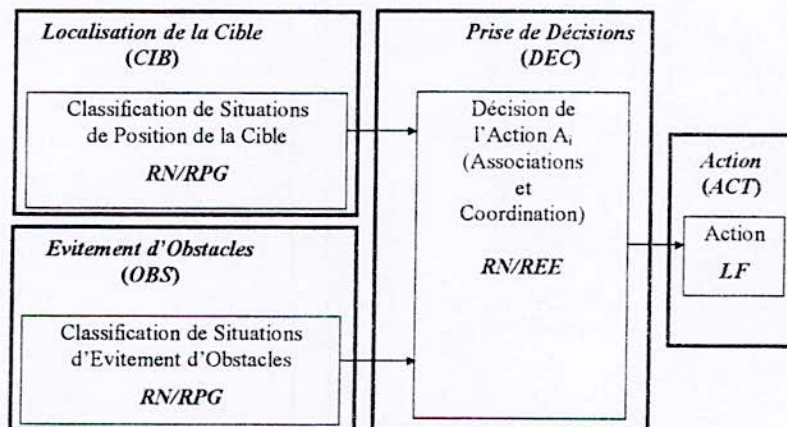


Figure II. 34. *SHI_A_NAV_1_Généralisée*.

L'architecture globale de planification et de contrôle proposée à cette stratégie permet aux *VAI* de réaliser:

- les comportements *CIB* et *OBS*, acquis par *apprentissage supervisé* par la *RPG* des *RN*,
- le comportement *DEC*, acquis par *apprentissage supervisé* par le *REE* d'un *RN*,
- le comportement *ACT*, acquis par *adaptation 'supervisée'* par la *LF*.

II. 6. 1. 2. *SHI_B_NAV_1_Généralisée*

Le *SHI_B_NAV_1_Généralisée*, voir Figure II. 35, est constitué des *SI* et *SHI* suivants:

Comportement *CIB*: Classification de situations de position de la cible par le *SHI_4: CIB_NAV_1_Généralisée-RNAMF/AMFS*,

Comportement *OBS*: Classification de situations d'évitement d'obstacles par le *SHI_4: OBS_NAV_1_Généralisée-RNAMF/AMFS*,

Comportement *DEC*: Décision de l'action appropriée A_i (associations et coordination) par le *SI_3: DEC_NAV_1_Généralisée-RN/REE*,

Comportement *ACT*: Exécution de l'action A_i par le *SI_2: ACT_NAV_1_NAV_2_Généralisées-LF*.

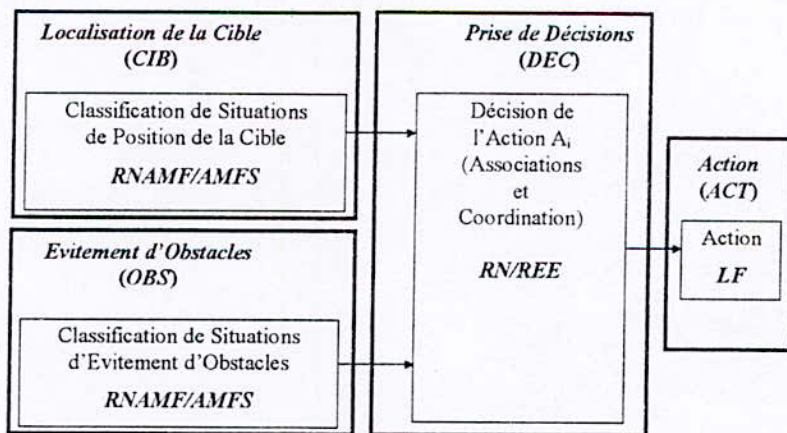


Figure II. 35. *SHI_B_NAV_1_Généralisée*.

L'architecture globale de planification et de contrôle proposée à cette stratégie permet aux *VAI* de réaliser:

- les comportements *CIB* et *OBS*, acquis par *apprentissage supervisé* par la *AMFS* des *RNAMF*,
- le comportement *DEC*, acquis par *apprentissage supervisé* par le *REE* d'un *RN*,
- le comportement *ACT*, acquis par *adaptation 'supervisée'* par la *LF*.

II. 6. 2. Stratégie NAV_2_Généralisée: SHI_SECOV_NAV_2_Généralisée

Le *SHI_SECOV_NAV_2_Généralisée*, voir Figure II. 36. (a), est constitué d'un *SE*, appelé *SECOV*, incorporant les *SI* et *SHI* suivants:

Comportement CIB: Ce comportement est réalisé par un traitement, voir § II. 3.3,

Comportement OBS: Classification des degrés des dangers statique et dynamique de chaque point obstacle par le *SHI_3 OBS_NAV_2_Généralisée-RNF/RPG*,

Comportement DEC: Classification des directions d'évitement par le *SI_1: DEC_NAV_2_Généralisée-RN/RPG*,

Comportement ACT: Exécution de l'action A_i par le *SI_2: ACT_NAV_1_NAV_2_Généralisées-LF*.

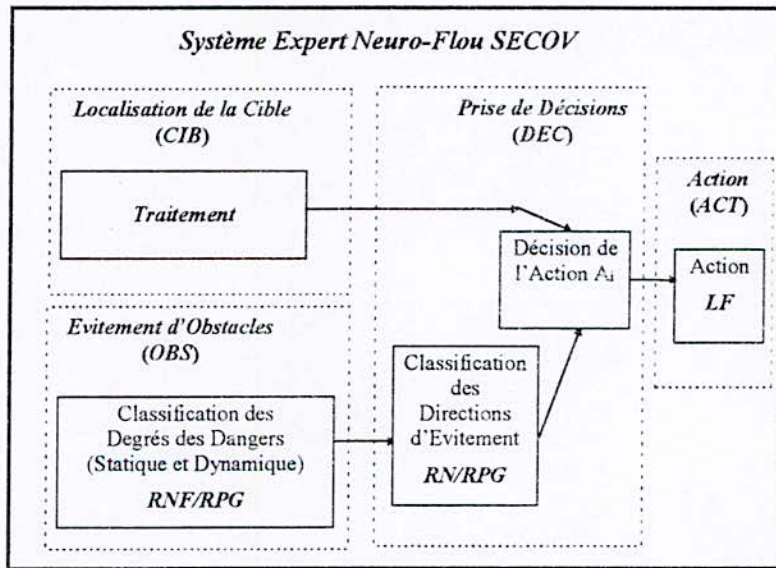


Figure II. 36. (a). *SHI_SECOV_NAV_2_Généralisée*.

Le *SECOV*, présenté dans la Figure II. 36. (b), intègre des *Procédures*, des *RNF*, et un *RN* dans un *SE* spécifique [54, 60]. Il est globalement inspiré de l'outil expert ERIC détaillé dans [33], et il est constitué d'une interface utilisateur permettant en plus de ces fonctions de base une facilité d'explication, d'une base de données, et d'une base de faits. La principale investigation dans *SECOV* est l'utilisation de deux (02) *RNF* et d'un (01) *RN* dans la tâche du raisonnement flou et de l'inférence floue de la base de faits présentée dans la Figure II. 36. (b).

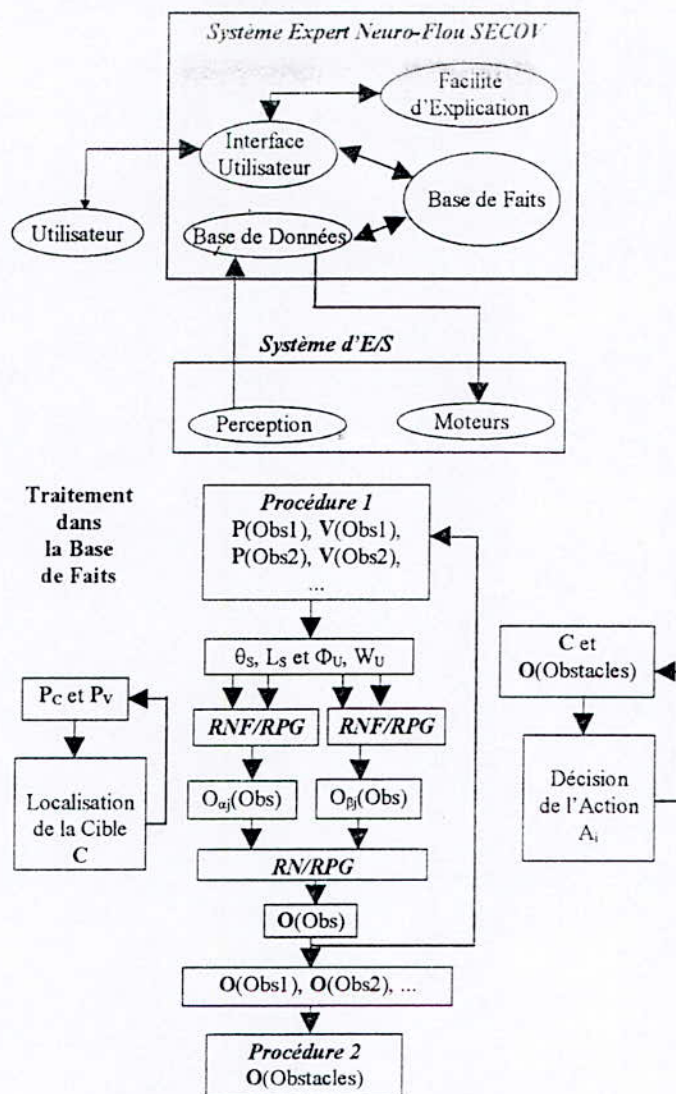


Figure II. 36. (b). *SHI_SECOV_NAV_2_Généralisée* (Base de Faits).

L'architecture globale de planification et de contrôle proposée à cette stratégie permet aux *VAI* de réaliser:

- le comportement *CIB* par un traitement ; notons qu'il devra être acquis par *apprentissage supervisé* par la *RPG* d'un *RN* (ou bien par *apprentissage supervisé* par la *AMFS* d'un *RNAMF*),
- le comportement *OBS*, acquis par *apprentissage supervisé* par la *RPG* des *RNF*,
- le comportement *DEC*, acquis par *apprentissage supervisé* par la *RPG* d'un *RN*,
- le comportement *ACT*, acquis par *adaptation 'supervisée'* par la *LF*.

Ces *SHI_A_NAV_1_Généralisée*, *SHI_A_NAV_1_Généralisée*, et *SHI_SECOV_NAV_2_Généralisée* devront alors permettre aux *VAI* de se diriger vers leurs cibles, d'éviter les éventuels obstacles rencontrés, de décider de l'action appropriée, et de générer des mouvements lisses.

II. 7. CONCLUSION

Nous nous sommes intéressés, dans ce Chapitre, à deux (02) stratégies de navigation, *NAV_1_Simplifiée* et *NAV_2_Simplifiée*, développées à partir d'études et d'observations des comportements de l'être humain en environnements dynamiques. En fait, ces stratégies intègrent essentiellement deux (02) manières différentes du raisonnement de l'être humain dans ses comportements intelligents en environnements dynamiques. Par ailleurs, bien que ces stratégies soient intéressantes elles présentent des problèmes dus principalement à des simplifications.

Afin de remédier à ces problèmes, nous avons proposé des généralisations:

- *NAV_1_Simplifiée* en *NAV_1_Généralisée* notamment dans l'augmentation du nombre de mouvements possibles de trois (03) à cinq (05),

- *NAV_2_Simplifiée* en *NAV_2_Généralisée* notamment dans la prise en charge de plusieurs obstacles à la fois de différentes tailles.

La principale différence entre ces stratégies de navigation réside dans leurs comportements *OBS*. Ce comportement se base sur une reconnaissance des situations topologiques (situations d'évitement d'obstacles) pour la stratégie *NAV_1_Généralisée*, alors qu'il se base sur une évaluation des degrés des dangers statique et dynamique des obstacles pour la stratégie *NAV_2_Généralisée*.

Le second problème a été alors d'acquérir les comportements intelligents *CIB*, *OBS*, *DEC*, et *ACT* de chaque stratégie de navigation par *apprentissage* ou/et *adaptation*. Pour cela, nous avons proposé des *SI* et des *SHI*, basés sur le *SC*, exploitant les capacités d'*apprentissage* ou/et d'*adaptation*, et de *généralisation* des *RN/RPG*, de la *LF*, des *RN/AG*, des *RN/AG-RPG*, des *RNF/RPG*, des *RNAMF/AMFS*, et des *RN/REE*. A partir de ces systèmes, nous avons proposé des architectures globales en *SHI* (*SHI_A_NAV_1_Généralisée*, *SHI_B_NAV_1_Généralisée*, et *SHI_SECOV_NAV_2_Généralisée*) pour la planification et le contrôle de la navigation des *VAI* en environnements dynamiques. Ces *SHI* devront fournir aux *VAI* plus d'*autonomie* et plus d'*intelligence* ainsi que des capacités de traitement en *temps réel*.

Chapitre III

SIMULATIONS, TESTS, ET RESULTATS

III. 1. INTRODUCTION

Nous présentons et discutons dans ce Chapitre, l'environnement de simulation, les entraînements et la mise en oeuvre des *Systèmes Intelligents (SI)* et *Systèmes Hybrides Intelligents (SHI)* proposés, ainsi que les résultats de la simulation des stratégies de navigation planifiées et contrôlées par les *SHI* suivants:

- *SHI_A_NAV_1_Généralisée*,
- *SHI_B_NAV_1_Généralisée*,
- *SHI_SECOV_NAV_2_Généralisée*.

III. 2. ENVIRONNEMENT DE SIMULATION

L'environnement de simulation est constitué des environnements dynamiques partiellement structurés (obstacles statiques et dynamiques non-intelligents), des véhicules (positions, vitesses, orientations, et capteurs à *US*), ainsi que leurs cibles.

III. 2. 1. Simulation des Environnements Dynamiques Partiellement Structurés

La simulation des environnements dynamiques partiellement structurés, voir Figure III. 1, implique la simulation des obstacles statiques, dynamiques intelligents, et dynamiques non-intelligents détaillés dans § I. 4. 1. 2. Dans cette simulation, les obstacles statiques sont représentés par des carrés, des rectangles, des triangles, et des cercles de couleur noire (pixels noirs). Les obstacles dynamiques non-intelligents sont représentés par des cercles de couleur noire (pixels noirs) évoluant avec une vitesse inférieure ou égale à celle des véhicules.

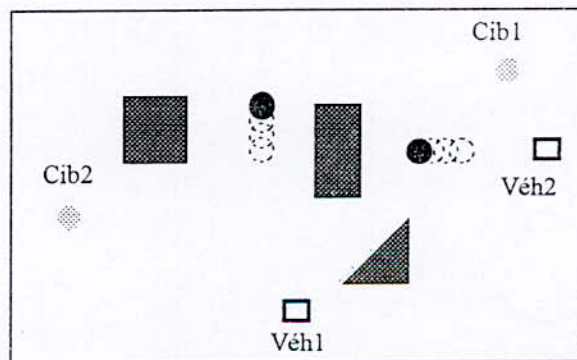


Figure III. 1. Environnement Dynamique Partiellement Structuré.

III. 2. 2. Simulation des Véhicules

Les véhicules sont simulés par un carré de bordure noire (pixels noirs) évoluant avec une même vitesse, voir Figure III. 1. Chaque véhicule est représenté par sa position (une composante selon l'axe des x et une composante selon l'axe des y) et son orientation puisque, dans cette simulation, les véhicules sont contrôlés en orientation. Notons que chaque véhicule considère les autres véhicules comme étant des obstacles dynamiques intelligents.

III. 2. 3. Simulation des Capteurs à UltraSons (US)

A chaque pas, les régions de perception, définies pour chaque stratégie de navigation, sont parcourues pixel par pixel, sur une distance d'un (01) jusqu'à vingt (20) pixels dans chaque région de perception. Ainsi, les obstacles ne seront considérés qu'à une distance du véhicule égale ou inférieure à vingt (20) pixels correspondant à deux (02) mètres. La détection de pixels noirs correspond alors à la détection d'obstacles.

Le développement de l'environnement de simulation nous a permis non seulement d'entraîner et de mettre en oeuvre les différents *SI* et *SHI* proposés, mais aussi de simuler les stratégies de navigation planifiées et contrôlées par le *SHI_A_NAV_1_Généralisée*, le *SHI_B_NAV_1_Généralisée*, et le *SHI_SECOV_NAV_2_Généralisée* dans des environnements dynamiques partiellement structurés.

III. 3. ENTRAÎNEMENTS ET MISE EN OEUVRE DES *SI* ET *SHI* POUR LES COMPORTEMENTS *CIB* ET *OBS*

Les principaux résultats des entraînements et de la mise en oeuvre des *SI* et *SHI*, proposés pour l'acquisition des comportements *CIB* et *OBS*, sont présentés dans ce Paragraphe.

III. 3. 1. Stratégie *NAV_1_Simplifiée*

L'entraînement des *SI* et *SHI* proposés à la stratégie *NAV_1_Simplifiée*, s'est effectué dans l'environnement d'entraînement présenté dans la Figure III. 2.

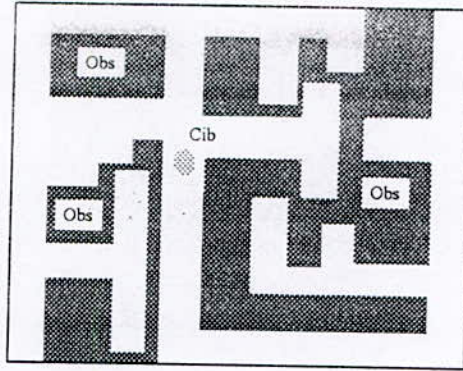


Figure III. 2. Environnement d'Entraînement (*CIB_OBS_NAV_1_Simplifiée*).

Cet environnement permet au véhicule de se trouver dans toutes les situations d'évitement d'obstacles ainsi que toutes les situations de position de la cible. Pendant l'entraînement, le véhicule se déplace selon les chemins libres d'obstacles, ce qui lui permettra d'être dans différentes positions, par rapport à la cible et par rapport aux obstacles, dans l'environnement. Ces positions correspondent en fait aux différentes situations de position de la cible et d'évitement d'obstacles.

Notons que cinquante quatre (54) positions du véhicule correspondant à cinquante quatre (54) exemples d'entraînement sont sélectionnés pour tous les entraînements des *SI* et *SHI* proposés pour l'acquisition du comportement *OBS* de la stratégie *NAV_1_Simplifiée*.

III. 3. 1. 1. *SI_1: CIB_NAV_1_Simplifiée-RN/RPG*

Les paramètres de la *RPG* pour l'entraînement de ce *SI* sont:

- facteur de prétraitement d'entrée $a_c = 50$,
- nombre de neurones dans la couche cachée $NNCC_c = 5$,
- initialisation aléatoire des poids entre $[-1, +1]$,
- taux d'apprentissage $\eta_c = 0.1$,
- erreur admissible $E_{ac} = 0.01$,
- nombre de cycles $N_{ec} = 48$.

Notons que le facteur de prétraitement d'entrée doit être choisi de telle manière à séparer les composantes du vecteur d'entrée ainsi que les vecteurs d'entrée aux maximum possible.

III. 3. 1. 2. *SI_1: OBS_NAV_1_Simplifiée-RN/RPG*

Les paramètres de la *RPG* pour l'entraînement de ce *SI* sont:

- facteur de prétraitement d'entrée $a_0 = 3$,
- nombre de neurones dans la couche cachée $NNCC_0 = 5$,
- initialisation aléatoire des poids entre $[-1, +1]$,
- taux d'apprentissage $\eta_0 = 0.1$,
- erreur admissible $E_{a0} = 0.01$,
- nombre de cycles $N_{c0} = 19$.

III. 3. 1. 3. *SI_2: OBS_NAV_1_Simplifiée-LF*

Les principales caractéristiques de ce *SI* sont:

- fonctions d'appartenance triangulaires et trapézoïdales pour la fuzzification,
- vingt cinq (25) règles floues constituant la base de règles floues,
- opérateurs MIN-MAX pour l'inférence floue,
- opérateur MAX pour la défuzzification.

III. 3. 1. 4. *SHI_1: OBS_NAV_1_Simplifiée-RN/AG*

Les paramètres de l' *AG* pour l'entraînement de ce *SHI* sont:

- population initiale $pop_0 = 50$ chromosomes,
- nombre de neurones dans la couche cachée $NNCC_0 = 5$,
- initialisation aléatoire des poids entre $[-5, +5]$,
- valeur aléatoire v_{mut0} entre $[-0.25, +0.25]$,
- nombre de cycles $N_{c0} = 5500$.

III. 3. 1. 5. *SHI_2: OBS_NAV_1_Simplifiée-RN/AG-RPG*

Les paramètres de l' *AG* pour l'entraînement de ce *SHI* sont:

- population initiale $pop_0 = 50$ chromosomes,
- nombre de neurones dans la couche cachée $NNCC_0 = 5$,
- initialisation aléatoire des poids entre $[-5, +5]$,

- valeur aléatoire v_{mut0} entre $[-0.25, +0.25]$,
- nombre de cycles $N_{c0} = 5000$.

Les paramètres de la *RPG* pour l'entraînement de ce *SHI* sont:

- facteur de prétraitement d'entrée $a_0 = 3$,
- nombre de neurones dans la couche cachée $NNCC_0 = 5$,
- taux d'apprentissage $\eta_0 = 0.1$,
- erreur admissible $E_{a0} = 0.01$,
- nombre de cycles $N_{c0} = 17$.

III. 3. 1. 6. *SHI_3: OBS_NAV_1_Simplifiée-RNF/RPG*

Les principales caractéristiques et les paramètres de la *RPG* pour l'entraînement de ce *SHI* sont:

- fonctions d'appartenance triangulaires et trapézoïdales pour la fuzzification,
- vingt cinq (25) règles floues constituant la base de règles floues, ce qui implique le même nombre de neurones dans la couche cachée $NNCC_0 = 25$,
- opérateurs $MIN-\Sigma$ pour l'inférence floue,
- opérateur MAX pour la défuzzification,
- initialisation aléatoire des poids entre $[-1, +1]$,
- taux d'apprentissage $\eta_0 = 0.4$,
- erreur admissible $E_{a0} = 0.01$,
- nombre de cycles $N_{c0} = 49$.

III. 3. 1. 7. *SHI_4: OBS_NAV_1_Simplifiée-RNAMF/AMFS*

Les paramètres de la *AMFS* pour l'entraînement de ce *SHI* sont:

- taux d'apprentissage $\eta_0 = 1.0$,
- constante $\lambda_0 = 0.000001$,
- constante $\varepsilon_0 = 0.0001$,
- vigilance de base $\sigma_0 = 0.4$,
- nombre de cycles $N_{c0} = 1$,
- nombre de neurones de sortie $n_0 = 10$.

Notons que dans le cas des *RNAMF/AMFS*, la vigilance de base est le seul paramètre à ajuster, alors que dans celui des *RN/RPG*, les paramètres à ajuster sont: le facteur de prétraitement d'entrée, l'intervalle d'initialisation des poids, le nombre de neurones cachés, le taux d'apprentissage, ainsi que le nombre de cycles. Dans le cas des *RN/AG*, les paramètres à ajuster sont: la taille de la population initiale, le type de représentation des chromosomes, les types d'opérateurs génétiques et leurs paramètres, ainsi que le nombre de cycles.

Dans le cas de la *LF*, la difficulté de mise en oeuvre réside essentiellement dans l'établissement de la base de règles floues, ainsi que dans le choix des fonctions d'appartenance pour la fuzzification et des opérations pour l'inférence floue et la défuzzification.

III. 3. 2. Stratégie *NAV_1_Généralisée*

L'entraînement des *SI* et *SHI* proposés à la stratégie *NAV_1_Généralisée* s'est effectué dans différents environnements d'entraînement.

III. 3. 2. 1. *SI_1: CIB_NAV_1_Généralisée-RN/RPG*

L'entraînement de ce *SI* s'est effectué dans l'environnement d'entraînement présenté dans la Figure III. 3. (a). En effet, le véhicule se déplace selon les chemins (1, ..., 10) dans un environnement libre d'obstacles, où la cible est positionnée au centre de cet environnement. Ce qui permet au véhicule de se trouver dans différentes positions et orientations et par conséquent la cible va être dans différentes positions par rapport à ce véhicule. Ainsi, chaque couple (position, orientation) correspond à un exemple d'entraînement pour une situation de position particulière de la cible. Plusieurs exemples d'entraînement sont alors sélectionnés pour chaque situation de position de la cible. Ainsi, l'ensemble d'entraînement est composé de mille deux cents cinquante (1250) exemples pour les six (06) situations de position de la cible.

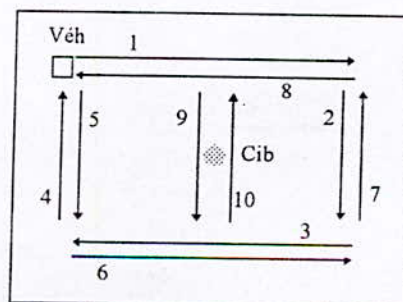


Figure III. 3. (a). Environnement d'Entraînement (*CIB_NAV_1_Généralisée*).

Les paramètres de la *RPG* pour l'entraînement de ce *SI* sont:

- facteur de prétraitement d'entrée $a_c = 5$,
- nombre de neurones dans la couche cachée $NNCC_c = 5$,
- initialisation aléatoire des poids entre $[-1, +1]$,
- taux d'apprentissage $\eta_c = 0.1$,
- erreur admissible $E_{aC} = 0.01$,
- nombre de cycles $N_{cC} = 43$.

III. 3. 2. 2. *SI_1: OBS_NAV_1_Généralisée-RN/RPG*

L'entraînement de ce *SI* s'est effectué dans l'environnement d'entraînement présenté dans la Figure III. 3. (b). En effet, le véhicule est simulé dans une position et orientation données dans cet environnement, où une configuration d'obstacles est simulée correspondant à un exemple d'une situation d'évitement d'obstacles particulière (p.ex., l'exemple d'entraînement présenté dans la Figure III. 3. (b) correspond à un exemple d'entraînement de la situation d'évitement d'obstacle O_{23}). Plusieurs exemples d'entraînement sont alors simulés pour chaque situation d'évitement d'obstacles. Ainsi, l'ensemble d'entraînement est composé de cent cinquante (150) exemples pour les trente (30) situations d'évitement d'obstacles.

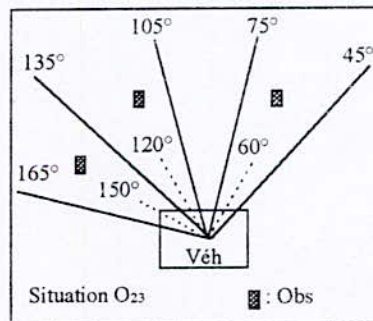


Figure III. 3. (b). Environnement d'Entraînement (*OBS_NAV_1_Généralisée*).

Les paramètres de la *RPG* pour l'entraînement de ce *SI* sont:

- facteur de prétraitement d'entrée $a_o = 10$,
- nombre de neurones dans la couche cachée $NNCC_c = 15$,
- initialisation aléatoire des poids entre $[-1, +1]$,
- taux d'apprentissage $\eta_o = 0.4$,
- erreur admissible $E_{aO} = 0.01$,
- nombre de cycles $N_{cO} = 50$.

III. 3. 2. 3. *SHI_4: CIB_NAV_1_Généralisée-RNAMF/AMFS*

Ce *SHI* est entraîné dans l'environnement d'entraînement présenté dans la Figure III. 3.

(a) en sélectionnant vingt quatre (24) exemples.

Les paramètres de la *AMFS* pour l'entraînement de ce *SHI* sont:

- taux d'apprentissage $\eta_C = 1.0$,
- constante $\lambda_C = 0.000001$,
- constante $\epsilon_C = 0.0001$,
- vigilance de base $\sigma_C = 0.4$,
- nombre de cycles $N_{eC} = 1$,
- nombre de neurones de sortie $n_C = 6$.

III. 3. 2. 4. *SHI_4: OBS_NAV_1_Généralisée-RNAMF/AMFS*

Ce *SHI* est entraîné dans l'environnement d'entraînement présenté dans la Figure III. 3.

(b) en simulant cent cinquante (150) exemples.

Les paramètres de la *AMFS* pour l'entraînement de ce *SHI* sont:

- taux d'apprentissage $\eta_O = 1.0$,
- constante $\lambda_O = 0.000001$,
- constante $\epsilon_O = 0.0001$,
- vigilance de base $\sigma_O = 0.25$,
- nombre de cycles $N_{eO} = 1$,
- nombre de neurones de sortie $n_O = 30$.

III. 3. 3. Stratégie *NAV_2_Généralisée*

L'entraînement des *SHI* proposés à la stratégie *NAV_2_Généralisée* s'est effectué dans un environnement d'entraînement où plusieurs exemples d'entraînement sont simulés, voir § III. 3. 2. 2, pour chaque degré de danger.

III. 3. 3. 1. *SHI_3: OBS_NAV_2_Généralisée-RNF/RPG: Degrés du Danger Statique*

Les principales caractéristiques et les paramètres de la *RPG* pour l'entraînement de ce *SHI* sont:

- fonctions d'appartenance triangulaires et trapézoïdales pour la fuzzification,

- douze (12) règles floues constituent la base de règles floues, ce qui implique le même nombre de neurones dans la couche cachée $NNCC_0 = 12$,
- opérateurs $MIN-\Sigma$ pour l'inférence floue,
- opérateur MAX pour la défuzzification,
- initialisation aléatoire des poids entre $[-1, +1]$,
- taux d'apprentissage $\eta_0 = 0.4$,
- erreur admissible $E_{a0} = 0.01$,
- nombre de cycles $N_{c0} = 75$.

III. 3. 3. 2. *SHI_3: OBS_NAV_2_Généralisée-RNF/RPG: Degrés du Danger Dynamique*

Les principales caractéristiques et les paramètres de la *RPG* pour l'entraînement de ce *SHI* sont:

- fonctions d'appartenance triangulaires et trapézoïdales pour la fuzzification,
- douze (12) règles floues constituent la base de règles floues, ce qui implique le même nombre de neurones dans la couche cachée $NNCC_0 = 12$,
- opérateurs $MIN-\Sigma$ pour l'inférence floue,
- opérateur MAX pour la défuzzification,
- initialisation aléatoire des poids entre $[-1, +1]$,
- taux d'apprentissage $\eta_0 = 0.3$,
- erreur admissible $E_{a0} = 0.01$,
- nombre de cycles $N_{c0} = 55$.

III. 4. ENTRAÎNEMENTS ET MISE EN OEUVRE DES *SI* POUR LE COMPORTEMENT *DEC*

Les principaux résultats des entraînements et de la mise en oeuvre des *SI*, proposés pour l'acquisition du comportements *DEC*, sont présentés dans ce Paragraphe.

III. 4. 1. *SI_3: DEC_NAV_1_Généralisée-RN/REE*

L'entraînement de ce *SI*, proposé à la stratégie *NAV_1_Généralisée*, est réalisé par l'entraînement en deux (02) étapes d'association (les poids U_{ij1} et V_{ij2}) ainsi que leur coordination (les actions A_i).

III. 4. 1. 1. Association du Comportement *CIB* aux Actions Favorables

L'association du comportement *CIB* aux actions favorables consiste à renforcer les poids U_{ij1} conduisant à des actions favorables et à pénaliser les poids U_{ij1} conduisant à des actions défavorables. La mise à jour de ces poids est réalisée par l'Equation (C. 31), voir ANNEXE C, où $M_{ij} = U_{ij1}$, $Classe_j = C_{j1}$, ($j = j1' = 1, \dots, 6$) et:

$$P = \begin{cases} P_1 & \text{si } Z_i = 0 \\ 0 & \text{si } Z_i = 1 \end{cases} \quad (\text{III. 1})$$

Afin d'obtenir les poids U_{ij1} dans cette étape, l'entraînement est réalisé dans un environnement libre d'obstacles (c.-à-d., $\mathbf{O} = 0$): dans l'environnement d'entraînement présenté dans la Figure III. 3. (a), où seulement un (01) exemple d'entraînement pour chaque situation de position de la cible est sélectionné. L'ensemble d'entraînement est composé alors de six (06) exemples d'entraînement.

Ce processus d'association peut être décrit à travers un exemple: à $t = 0$, $U_{ij1}(0) = 0$ et $P = 0$, le véhicule est alors positionné de manière à ce que la position de sa cible corresponde à la situation de position de la cible C_1 . Cette situation active la sortie du *SI_1*: *CIB_NAV_1_Généralisée-RN/RPG* (ou *SHI_4: CIB_NAV_1_Généralisée-RNAMF/AMFS*) correspondante (c.-à-d., $C_1 = 1$). Par conséquent, le vecteur $\mathbf{Z} = [1, 1, 1, 0, 0]$, voir Figure II. 30, c.-à-d., dans cette situation A_1 , A_2 , et A_3 sont des actions favorables alors que A_4 et A_5 sont des actions défavorables.

Pour les actions favorables ($P = 0$), l'Equation (C. 31) se sature à α quand t s'incrémente avec $\tau = 3$ s comme constante de temps. Ainsi, U_{11} , U_{21} , et U_{31} sont alors augmentés, ce qui se traduit par:

- l'association entre C_1 et A_1 s'est renforcée avec $t = t_1$ à saturation,
- l'association entre C_1 et A_2 s'est renforcée avec $t = t_2$ où $t_2 < t_1$,
- l'association entre C_1 et A_3 s'est renforcée avec $t = t_3$ où $t_3 < t_2$.

Pour les actions défavorables ($P = P_1$) l'Equation (C. 31) se sature à $(\alpha - P_1)$ avec une valeur négative quand t s'incrémente avec la même constante de temps $\tau = 3$ s. Ainsi U_{41} et U_{51} deviennent négatifs, ce qui se traduit par:

- A_4 s'est dissociée de C_1 avec t à saturation,
- A_5 s'est dissociée de C_1 avec t à saturation.

Les poids U_{ij1} sont alors ajustés afin d'obtenir les actions renforcées parmi les actions favorables représentées par la matrice d'association *CIB_ACT_NAV_1_Généralisée* présentée dans la Figure III. 4. (a), où les cercles pleins correspondent aux poids positifs qui représentent les actions favorables indiquant une association renforcée dont la valeur est proportionnelle à la taille du cercle correspondant et l'action la plus renforcée est ainsi celle ayant le plus grand poids positif. Quant aux cercles vides, ils correspondent aux poids négatifs qui représentent les actions dissociées.

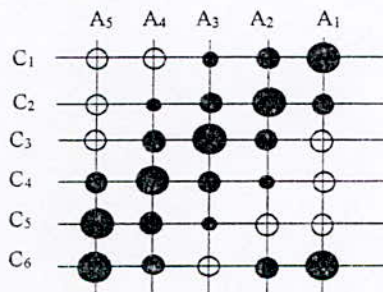


Figure III. 4. (a). Matrice d'Association (*CIB_ACT_NAV_1_Généralisée*).

Notons que le choix de l'action la plus renforcée est guidé par le principe suivant: le véhicule doit prendre l'action la plus directe le conduisant vers sa cible.

III. 4. 1. 2. Association du Comportement *OBS* aux Actions Favorables

L'association du comportement *OBS* aux actions favorables consiste à renforcer les poids V_{ij2} conduisant à des actions favorables et à pénaliser les poids V_{ij2} conduisant à des actions défavorables. La mise à jour de ces poids est réalisée par l'Equation (C. 31), voir ANNEXE C, où $M_{ij} = V_{ij2}$, Classe_j = O_{j2} , ($j = j2 = 1, \dots, 30$) et:

$$P = \begin{cases} P_2 & \text{si collision} \\ 0 & \text{sinon} \end{cases}, \quad (\text{III. 2})$$

Afin d'obtenir les poids V_{ij2} dans cette étape, l'entraînement est réalisé dans un environnement sans prendre en considération la cible (c.-à-d., $C = 0$): dans l'environnement d'entraînement, présenté pour la situation O_{23} dans la Figure III. 3. (b), où seulement un (01) exemple d'entraînement pour cette situation d'évitement d'obstacles est simulé. L'ensemble d'entraînement est composé alors de trente (30) exemples d'entraînement.

Ce processus d'association peut être décrit à travers un exemple: à $t = 0$, $V_{ij2}(0) = 0$ et $P = 0$, le véhicule est alors positionné de manière à ce que la position des obstacles corresponde à la situation d'évitement d'obstacles O_1 . Cette situation active la sortie du SI_1 : $OBS_NAV_1_Généralisée-RN/RPG$ (ou SHI_4 : $OBS_NAV_1_Généralisée-RNAMF/AMFS$) correspondante (c.-à-d., $O_1 = 1$). Par conséquent, dans cette situation A_2 , A_3 , A_4 , et A_5 sont des actions favorables alors que A_1 est une action défavorable, voir Figure II. 5.

Pour les actions favorables ($P = 0$), l'Equation (C. 31) se sature à α quand t s'incrémente avec $\tau = 3$ s comme constante de temps. Ainsi, V_{21} , V_{31} , V_{41} et V_{51} sont alors augmentés, ce qui se traduit par:

- l'association entre O_1 et A_2 s'est renforcée avec $t = t_1$ à saturation,
- l'association entre O_1 et A_3 s'est renforcée avec $t = t_2$ où $t_2 < t_1$,
- l'association entre O_1 et A_4 s'est renforcée avec $t = t_3$ où $t_3 < t_2$,
- l'association entre O_1 et A_5 s'est renforcée avec $t = t_4$ où $t_4 < t_3$.

Pour les actions défavorables ($P = P_2$) l'Equation (C. 31) se sature à $(\alpha - P_2)$ avec une valeur négative quand t s'incrémente avec la même constante de temps $\tau = 3$ s. Ainsi V_{11} devient négatif, ce qui se traduit par:

- A_1 s'est dissociée de O_1 avec t à saturation.

Les poids V_{ij2} sont alors ajustés afin d'obtenir les actions renforcées parmi les actions favorables représentées par la matrice d'association $OBS_ACT_NAV_1_Généralisée$ présentée dans la Figure III. 4. (b), où les cercles pleins correspondent aux poids positifs, qui représentent les actions favorables, indiquant une association renforcée dont la valeur est proportionnelle à la taille du cercle correspondant et l'action la plus renforcée est ainsi celle ayant le plus grand poids positif. Quant aux cercles vides, ils correspondent aux poids négatifs qui représentent les actions dissociées.

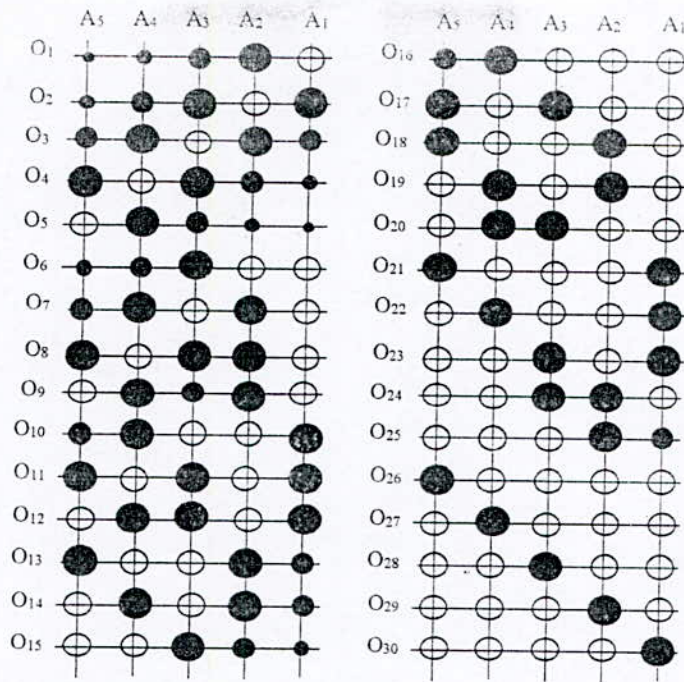


Figure III. 4. (b). Matrice d'Association (*OBS_ACT_NAV_1_Généralisée*).

Notons que le choix de l'action la plus renforcée est guidé par le principe suivant: le véhicule doit prendre l'action lui permettant d'éviter les collisions avec les obstacles sans trop s'éloigner de ces obstacles.

Comme expliqué auparavant, l'Equation (C. 31) se sature à α dans les deux (02) étapes d'association quand les actions sont favorables, elles doivent alors être renforcées c.-à-d., avec des poids positifs ($\alpha > 0$). Par ailleurs, l'Equation (C. 31) se sature à $(\alpha - P)$ dans les deux (02) étapes d'association quand les actions sont défavorables, elles doivent alors être dissociées c.-à-d., avec des poids négatifs ($P > \alpha$). Ceci conduit aux relations suivantes:

$$\alpha > 0, P_1 > \alpha, \text{ et } P_2 > \alpha. \quad (\text{III. 3})$$

III. 4. 1. 3. Coordination des Comportements *CIB* et *OBS*

Dans la coordination des deux (02) étapes d'association, la détection du maximum du champ de température doit être interprété comme le *but* du véhicule alors que les actions générées par la présence d'obstacles doivent être interprétées comme le *réflexe* du véhicule. Les actions générées par le comportement *OBS* doivent alors avoir la priorité sur celles générées par le comportement *CIB* c.-à-d., les constantes P_1 et P_2 doivent être définies tel que:

$$P_2 > P_1. \quad (\text{III. 4})$$

La variable de distribution aléatoire N_i , voir l'Equation (II. 33), est utilisée afin de décider entre deux (02) ou bien parmi plusieurs actions favorables dans des cas de conflits p.ex., dans le cas des situations C_3 et O_{19} ($C_3 = 1$ et $O_{19} = 1$): A_1 et A_5 sont dissociées de C_3 voir Figure III. 4. (a) alors que A_1 , A_3 , et A_5 sont dissociées de O_{19} voir Figure III. 4. (b). A partir de l'Equation (II. 33) ni A_1 , ni A_3 , ni A_5 ne pourrait être alors la sortie la plus activée sans prendre en considération N_i . Ainsi, A_2 et A_4 se sont renforcées à C_3 avec la même valeur tandis que ces mêmes actions A_2 et A_4 se sont renforcées à O_{19} avec la même valeur.

Dans ce cas, l'action A_i la plus activée correspond à l'action ayant la plus grande valeur N_i . Notons que les valeurs N_i doivent être différentes. Ainsi, les constantes β et α doivent être définies comme suit:

$$0 < \beta < \alpha \quad \text{(III. 5)}$$

Les valeurs utilisées des différentes constantes dans les deux (02) étapes d'association et leur coordination sont:

- $\beta = 1$,
- $\alpha = 5$,
- $P_1 = 7$,
- $P_2 = 9$.

Notons que les actions considérées comme défavorables dans le comportement *CIB* pourraient être prises dans certains cas alors que celles considérées comme défavorables dans le comportement *OBS* ne sont jamais prises.

III. 4. 2. *SI_1: DEC_NAV_2_Généralisée-RN/RPG*

L'entraînement de ce *SI* proposé à la stratégie *NAV_2_Généralisée* s'est effectué dans un environnement d'entraînement où plusieurs exemples d'entraînement sont simulés, voir § III. 3. 2. 2, pour chaque direction d'évitement.

Les paramètres de la *RPG* pour l'entraînement de ce *SI* sont:

- facteur de prétraitement d'entrée $a_x = 1.5$,
- nombre de neurones dans la couche cachée $NNCC_0 = 7$,

- initialisation aléatoire des poids entre $[-1, +1]$,
- taux d'apprentissage $\eta_0 = 0.39$,
- erreur admissible $E_{a0} = 0.03$,
- nombre de cycles $N_{c0} = 100$.

III. 5. MISE EN OEUVRE DU *SI_2: ACT_NAV_1_NAV_2_Généralisées-LF* POUR LE COMPORTEMENT *ACT*

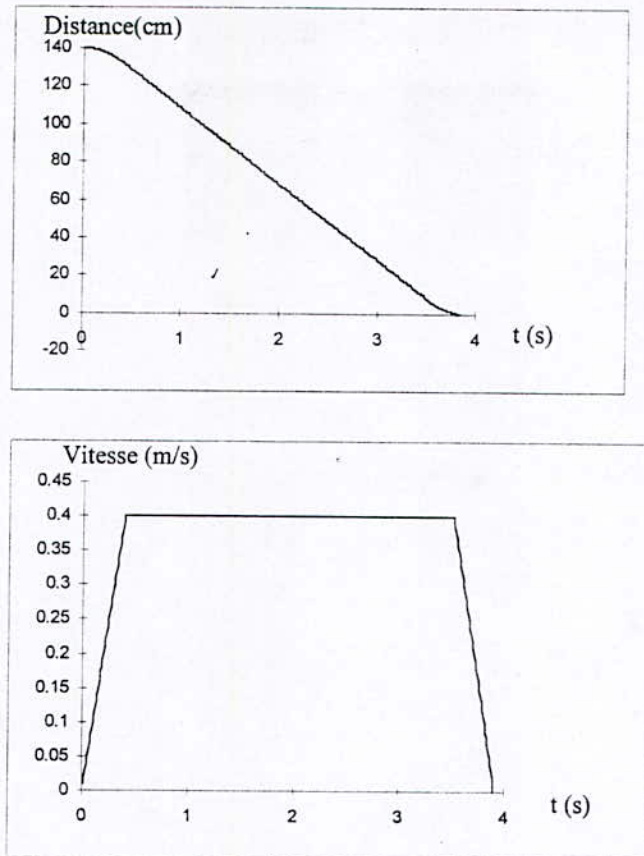
Les principaux résultats de la mise en oeuvre du *SI_2: ACT_NAV_1_NAV_2_Généralisées-LF*, proposé pour l'acquisition du comportement *ACT*, sont présentés dans ce Paragraphe.

Les principales caractéristiques de ce *SI* sont:

- fonctions d'appartenance trapézoïdales pour la fuzzification,
- trois cents (300) règles floues constituant la base de règles floues,
- opérateurs MIN-MAX pour l'inférence floue,
- opérateur MAX pour la défuzzification.

Les résultats en simulation, présentés dans la Figure III. 5, montrent la courbe des distances en fonction du temps et celle des vitesses en fonction du temps pour le cas Vitesse Finale = 0 m/s, Vitesse Initiale = 0 m/s, et distance-but = 1.4 m. La première figure montre que la distance-but est parcourue de (1.4 m à 0 m) en moins de 4 secondes en générant une trajectoire (mouvement) lisse. La seconde figure montre bien que cette distance (1.4 m) a été parcourue en temps minimum en accélération maximale jusqu'à atteindre la vitesse maximale (0.4 m/s) ensuite en maintenant cette vitesse jusqu'à ce que la distance véhicule-position désirée (distance-but) est suffisante (juste assez) pour décélérer à la vitesse finale désirée (0 m/s).

Ces résultats montrent que ce *SI*, se basant sur la capacité d'*adaptation* et l'*intelligence* de la *LF*, est capable de générer des mouvements *lisses*.



Vitesse Finale = 0 m/s,
 Vitesse Initiale = 0 m/s,
 Distance-But = 1.4 m.

**Figure III. 5. Courbes des Distances et Vitesses du *SI_2*:
*ACT_NAV_1_NAV_2_Généralisées-LF.***

III. 6. RESULTATS DE LA SIMULATION DES STRATEGIES DE NAVIGATION

Afin de refléter les comportements des véhicules acquis par apprentissage ou/et adaptation, et de démontrer la capacité de généralisation des approches proposées de planification et de contrôle de la navigation, la navigation des véhicules est simulée dans différents environnements statiques et dynamiques partiellement structurés.

Chaque véhicule simulé ne peut disposer que de deux (02) données connues au départ: ses positions initiale et finale (cible), ainsi qu'une orientation initiale. A partir de ces données, il doit atteindre sa cible tout en évitant les éventuels obstacles rencontrés en utilisant les stratégies de navigation proposées *NAV_1_Généralisée* et *NAV_2_Généralisée* planifiées et contrôlées par les approches proposées. Dans cette simulation, le comportement *ACT* n'est pas simulé, ce qui implique que le véhicule contrôle uniquement son orientation. De plus, les

obstacles dynamiques non-intelligents sont supposés avoir une vitesse inférieure ou égale à celles des véhicules. Quant à la vitesse des véhicules (obstacles dynamiques intelligents), ils ont tous la même vitesse.

Afin d'estimer les résultats des différentes stratégies de navigation proposées, nous présentons pour chaque stratégie les résultats dans différents environnements: à obstacles statiques, à obstacles dynamiques intelligents, à obstacles dynamiques non-intelligents, et dans un environnement dynamique partiellement structuré.

III. 6. 1. Stratégie *NAV_1_Généralisée*

Nous présentons, dans ce Paragraphe, les résultats de simulation de la stratégie de navigation, *NAV_1_Généralisée*, planifiée et contrôlée par le *SHI_A_NAV_1_Généralisée* ainsi que ceux de la même stratégie planifiée et contrôlée par le *SHI_B_NAV_1_Généralisée*.

III. 6. 1. 1. *SHI_A_NAV_1_Généralisée*

Dans cette simulation, les *VAI*, dont la navigation est planifiée et contrôlée par le *SHI_A_NAV_1_Généralisée*, sont simulés dans différents environnements.

- **Obstacles Statiques**

Testé dans un environnement contenant des obstacles statiques, illustré dans la Figure III. 6, le véhicule arrive avec succès à éviter les obstacles statiques tout en se dirigeant vers sa cible.



Figure III. 6. Obstacles Statiques (*SHI_A_NAV_1_Généralisée*).

- **Obstacles Dynamiques Intelligents**

Dans le cas illustré dans la Figure III. 7, les quatre (04) véhicules Véh1, Véh2, Véh3, et Véh4 se dirigent vers leurs cibles respectives tout en s'évitant mutuellement. En effet, ces véhicules naviguent avec la même approche utilisant le *SHI_A_NAV_1_Généralisée* où chaque véhicule considère les autres comme étant des obstacles dynamiques intelligents.

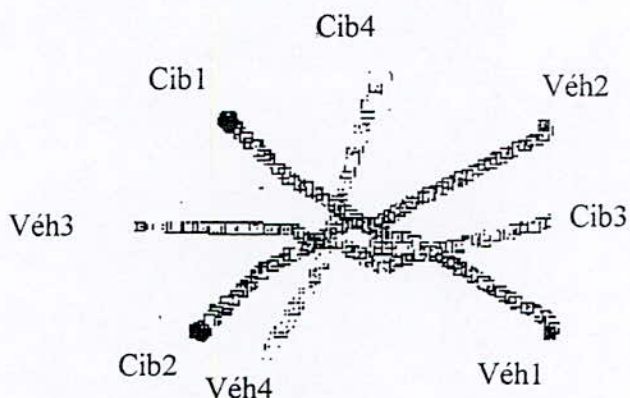


Figure III. 7. Obstacles Dynamiques Intelligents (*SHI_A_NAV_1_Généralisée*).

- **Obstacles Dynamiques Non-Intelligents**

Dans le cas d'obstacles dynamiques non-intelligents, l'un oscillant verticalement et l'autre horizontalement entre deux (02) points fixes, illustré dans la Figure III. 8, le véhicule les évite et atteint sa cible avec succès.

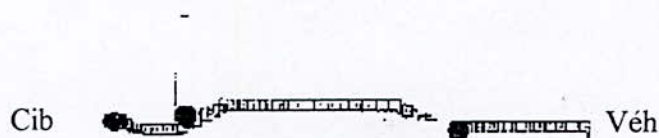


Figure III. 8. Obstacles Dynamiques Non-Intelligents (*SHI_A_NAV_1_Généralisée*).

- **Environnement Dynamique Partiellement Structuré**

Dans le cas illustré dans la Figure III. 9, les trois (03) véhicules atteignent leurs cibles, sans collisions, dans un environnement dynamique partiellement structuré.

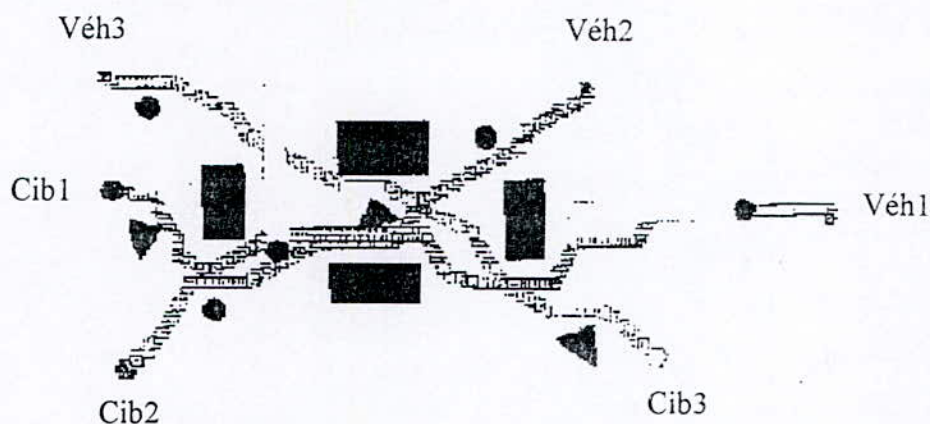


Figure III. 9. Environnement Dynamique Partiellement Structuré
(*SHI_A_NAV_1_Généralisée*).

Ces résultats illustrent les capacités d'*apprentissage* et de *généralisation* des *RN/RPG* et du *RN/REE* utilisés par le *SHI_A_NAV_1_Généralisée*.

III. 6. 1. 2. *SHI_B_NAV_1_Généralisée*

Dans cette simulation, les *VAI*, dont la navigation est planifiée et contrôlée par le *SHI_B_NAV_1_Généralisée*, sont simulés dans différents environnements.

- **Obstacles Statiques**

Testé dans un environnement contenant des obstacles statiques, illustré dans la Figure III. 10, le véhicule atteint sa cible sans collisions avec les obstacles statiques rencontrés.

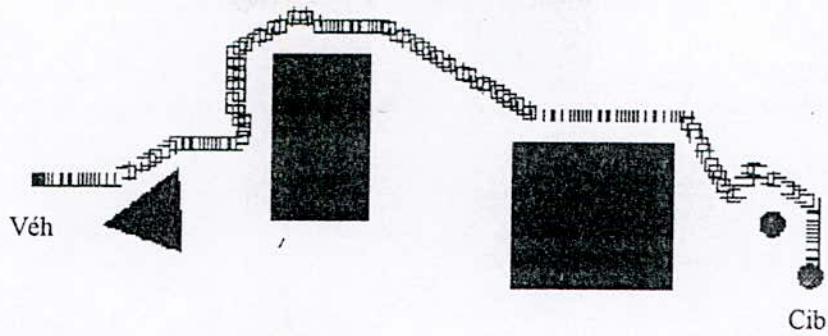


Figure III. 10. Obstacles Statiques (*SHI_B_NAV_1_Généralisée*).

- Obstacles Dynamiques Intelligents

Dans le cas illustré dans la Figure III. 11, les quatres (04) véhicules Véh1, Véh2, Véh3, et Véh4 atteignent leurs cibles respectives tout en s'évitant.

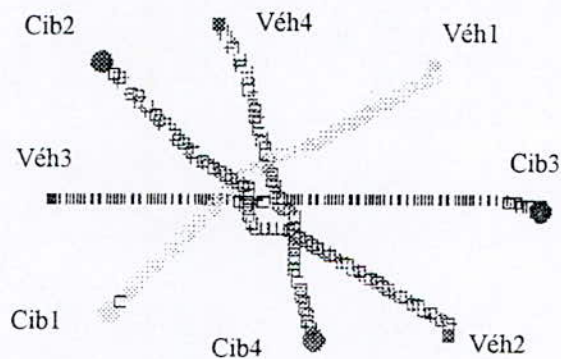


Figure III. 11. Obstacles Dynamiques Intelligents (*SHI_B_NAV_1_Généralisée*).

- Obstacles Dynamiques Non-Intelligents

Dans le cas d'obstacles dynamiques non-intelligents, illustré dans la Figure III. 12, le véhicule les évite et atteint sa cible avec succès.



Figure III. 12. Obstacles Dynamiques Non-Intelligents (*SHI_B_NAV_1_Généralisée*).

- **Environnement Dynamique Partiellement Structuré**

Dans le cas illustré dans la Figure III. 13, les trois (03) véhicules atteignent leurs cibles sans collisions avec les obstacles statiques et dynamiques rencontrés.

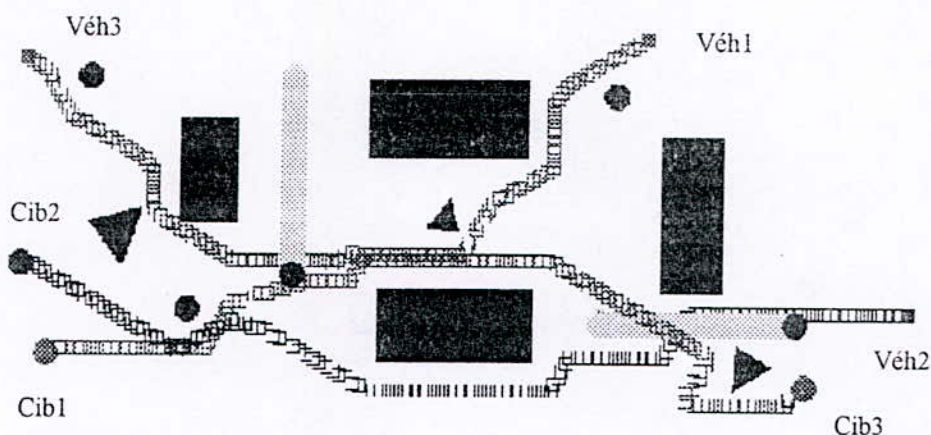


Figure III. 13. Environnement Dynamique Partiellement Structuré
(*SHI_B_NAV_1_Généralisée*).

Ces résultats illustrent les capacités d'*apprentissage*, d'*adaptation*, et de *généralisation* des *RNAMF/AMFS* ainsi que celles d'*apprentissage* et de *généralisation* du *RN/REE* utilisés par le *SHI_B_NAV_1_Généralisée*.

III. 6. 2. Stratégie *NAV_2_Généralisée*: *SHI_SECOV_NAV_2_Généralisée*

Dans cette simulation, les *VAI*, dont la navigation est planifiée et contrôlée par le *SHI_SECOV_NAV_2_Généralisée*, sont simulés dans différents environnements.

- **Obstacles Statiques**

Testé dans un environnement contenant des obstacles statiques, illustré dans la Figure III. 14, le véhicule atteint sa cible sans collisions avec les obstacles statiques rencontrés. Notons que le véhicule prend en considération la taille de l'obstacle pendant son évitement.

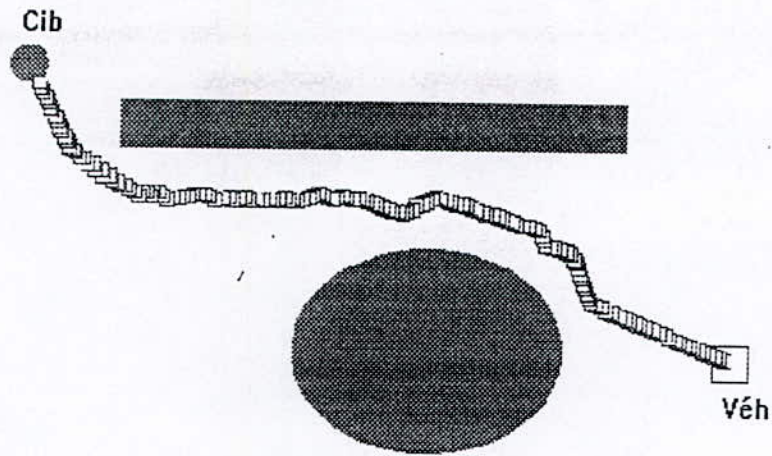


Figure III. 14. Obstacles Statiques (*SHI_SECOV_NAV_2_Généralisée*).

- **Obstacles Dynamiques Intelligents**

Dans le cas illustré dans la Figure III. 15, les trois (03) véhicules Véh1, Véh2, et Véh3 atteignent leurs cibles respectives tout en s'évitant. Notons que les véhicules prennent en considération plusieurs obstacles en même temps.

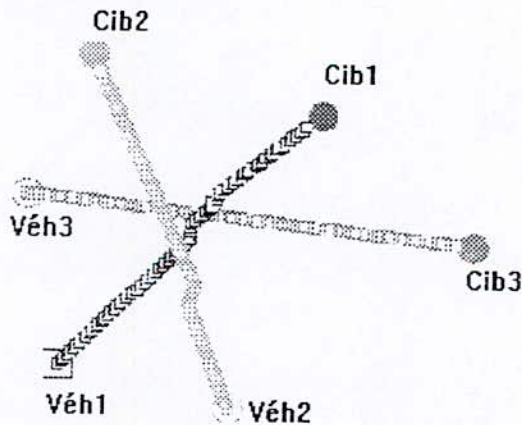


Figure III. 15. Obstacles Dynamiques Intelligents (*SHI_SECOV_NAV_2_Généralisée*).

- **Obstacles Dynamiques Non-Intelligents**

• Dans le cas d'un obstacle dynamique non-intelligent, illustré dans la Figure III. 16, le véhicule l'évite et atteint sa cible avec succès.

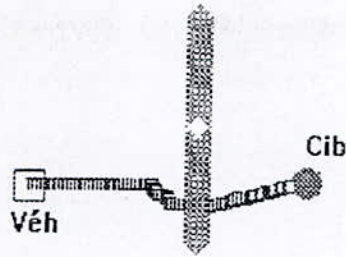


Figure III. 16. Obstacle Dynamique Non-Intelligent
(*SHI_SECOV_NAV_2_Généralisée*).

- Environnement Dynamique Partiellement Structuré

Dans le cas illustré dans la Figure III. 17, les trois (03) véhicules atteignent leurs cibles sans collisions avec les obstacles statiques et dynamiques rencontrés. Notons que les véhicules prennent en considération non seulement plusieurs obstacles en même temps mais aussi leurs tailles.

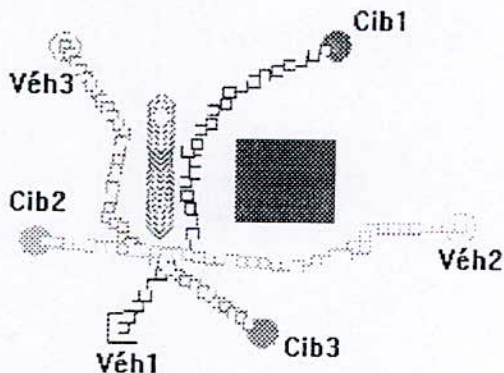


Figure III. 17. Environnement Dynamique Partiellement Structuré
(*SHI_SECOV_NAV_2_Généralisée*).

Ces résultats illustrent les capacités d'*apprentissage*, d'*adaptation*, et de *généralisation* des *RNF/RPG* ainsi que celles d'*apprentissage* et de *généralisation* du *RN/RPG* utilisés par le *SHI_SECOV_NAV_2_Généralisée*. Ils illustrent aussi l'*efficacité* de *SECOV* dans la prise en considération de plusieurs obstacles à la fois avec différentes tailles.

III. 7. CONCLUSION

Les résultats de simulation, illustrent les capacités d'*apprentissage* ou/et d'*adaptation*, ainsi que celle de *généralisation* des différents *SI* et *SHI* proposés pour l'acquisition des comportements intelligents *CIB*, *OBS*, et *DEC*. Quant au *SI* proposé pour l'acquisition du comportement intelligent *ACT*, les résultats montrent qu'un tel système, se basant sur la capacité d'*adaptation* de la *LF*, est capable de fournir aux *VAI*, la capacité de génération de mouvements *lisses*.

Les résultats de simulation des stratégies de navigation *NAV_1_Généralisée* et *NAV_2_Généralisée* en environnements dynamiques partiellement structurés, planifiées et contrôlées par le *SHI_A_NAV_1_Généralisée*, par le *SHI_B_NAV_1_Généralisée*, ou par le *SHI_SECOV_NAV_2_Généralisée*, ont montré d'une part que ces *SHI* fournissent aux *VAI* plus d'*autonomie* et plus d'*intelligence*. D'autres part, ces résultats de simulation ont démontré l'*efficacité* des généralisations proposées à ces stratégies de navigation notamment par plus de possibilités de mouvements pour *NAV_1_Généralisée* et par la prise en charge de plusieurs obstacles à la fois de différentes tailles pour *NAV_2_Généralisée*.

Chapitre IV

***DISCUSSIONS DES RESULTATS DE
SIMULATION, IMPLEMENTATION DES
RESEAUX DE NEURONES (RN), ET
PERSPECTIVES***

IV. 1. INTRODUCTION

Nous présentons, dans Ce Chapitre, des discussions des résultats de simulation obtenus des Systèmes Intelligents (*SI*) et Systèmes Hybrides Intelligents (*SHI*) proposés pour l'acquisition des comportements intelligents de localisation de la cible (*CIB*), d'évitement d'obstacles (*OBS*), de prise de décisions (*DEC*), et d'action (*ACT*) ainsi que ceux des *SHI* proposés pour la planification et le contrôle de la navigation. Nous proposons ensuite une architecture "Field-Programmable Gate Array" (*FPGA*) pour l'implémentation d'un *SI* basé sur les *RN* réalisant le comportement *OBS*. Enfin, nous présentons quelques perspectives jugées intéressantes tant sur l'aspect théorique que celui d'implémentation.

IV. 2. DISCUSSIONS DES RESULTATS DE SIMULATION

IV. 2. 1. *SI* et *SHI* pour l'Acquisition des Comportements Intelligents

IV. 2. 1. 1. Performances des *SI* et *SHI*

Après avoir choisi une méthodologie basée sur le *SC*, et après avoir opté pour son implémentation en perspectives voir § IV. 4. 5, la réponse à la question suivante s'impose:

Question: comment estimer la *qualité* des systèmes résultants ?

En général, pour des systèmes impliquant des calculs, il existe deux (02) principaux critères [128]:

1- la *performance* résultante du système, spécialement pour les systèmes de contrôle, un critère typique de la qualité est la *stabilité* du contrôle résultant ;

2- la *rapidité* et la *précision* des calculs (algorithmes *temps réel*).

Pour une application donnée, les principales caractéristiques désirées de la performance sont: la *stabilité* de contrôle, la *rapidité* (déterminer la complexité de calcul de chaque algorithme ou mesurer le temps), la *précision* (par rapport à l'erreur), la *robustesse* (tolérance aux défaillances), et la capacité de *généralisation* (apprentissage et adaptation) [74].

IV. 2. 1. 2. Comparaison des *SI* et *SHI* Proposés au Comportement *OBS* (*NAV_1_Simplifiée*)

Les paramètres des différents *SI* et *SHI* proposés au comportement *OBS* pour la stratégie *NAV_1_Simplifiée* sont présentés dans le Tableau IV. 1.

**Tableau IV. 1. Paramètres des *SI* et *SHI* Proposés au Comportement *OBS*
(*NAV_1_Simplifiée*).**

	Nombre de Poids	Nombre de Cycles (<i>AG</i>)	Nombre de Neurones Cachés	Taux d'Apprentissage	Nombre de Cycles (<i>RPG</i> ou <i>AMFS</i>)	Nombre de Règles
<i>RN/RPG</i>	45	---	5	0.1	19	---
<i>RN/AG</i>	45	5500	5	---	---	---
<i>RN/AG-RPG</i>	45	5000	5	0.1	17	---
<i>RNF/RPG</i>	225	---	25	0.4	49	25
<i>RNAMF/AMFS</i>	60	---	---	1	1	---
<i>LF</i>	---	---	---	---	---	25

Notons que les mêmes cinquante quatre (54) exemples d'entraînement ainsi que la même erreur admissible ont été utilisés pour chaque entraînement de chaque *SI* ou *SHI* proposé à la stratégie *NAV_1_Simplifiée*. Concernant, le *SI_2: OBS_NAV_1_Simplifiée-LF*, cette erreur admissible a été prise en considération durant l'établissement de la base de règles floues.

• **Comparaison entre *RN/RPG*, *RN/AG*, *RN/AG-RPG*, et *RNAMF/AMFS***

Les apprentissages supervisés par la *Rétro-Propagation du Gradient (RPG)*, les *Algorithmes Génétiques (AG)*, et les *AG-RPG* ayant à entraîner le même Réseaux de Neurones (*RN*) de quarante cinq (45) poids et cinq (05) neurones cachés, avec le même taux d'apprentissage (0.1) pour la *RPG*, ont donnés pratiquement les mêmes résultats par rapport à la capacité de *généralisation*. Alors que la *RPG* présente des difficultés de convergence (ajustement de plusieurs paramètres liés à la convergence: risque de non-convergence), les *AG* posent plutôt le problème de lenteur de convergence. Le principal avantage pour l'entraînement des *RN* multicouches par les *AG* réside dans le fait qu'ils peuvent trouver les minimums *globaux* sans être 'piégés' dans les minimums *locaux*. L'apprentissage par les *AG-RPG* peut alors non seulement remédier aux problèmes des minimums *locaux* mais aussi à celui de la lenteur.

L'apprentissage supervisé par la "ArtMap" Floue Simplifiée (*AMFS*) ne présente ni difficultés de convergence, ni problème de lenteur (avec un seul paramètre à ajuster, un taux d'apprentissage égal à 1, et en 1 cycle), cet apprentissage est par conséquent *stable* et *rapide*.

En conclusion, lorsqu'un ensemble d'exemples d'entraînement est disponible, l'utilisation des Réseaux de Neurones basés sur la "ArtMap" Floue (*RNAMF*) est particulièrement conseillée que celle des *RN* puisque l'apprentissage par la *AMFS* a prouvé sa *rapidité* et sa *stabilité* ainsi que sa grande capacité de *généralisation* surpassant les performances des autres types d'apprentissage, comme celui par la *RPG*. De plus, pour une application de classification donnée par les *RN*, si la *RPG* présente des difficultés de convergence alors il faudra utiliser les *AG*, à ce moment là si ces derniers mettent beaucoup de temps à converger, il faudra alors utiliser les *AG-RPG*.

- **Comparaison entre *RNF/RPG* et *LF***

Dans le cas où une base de règles floues est disponible, l'utilisation des Réseaux de Neurones Flous (*RNF*), entraînés par la *RPG*, ou de la Logique Floue (*LF*) est plutôt conseillée. De plus, bien que la difficulté des *RNF* ou de la *LF* réside dans l'établissement de la base de règles floues ainsi que le choix du types de fonctions d'appartenance, ils présentent l'avantage d'être *stables*.

IV. 2. 1. 3. *SI* et *SHI* pour l'Acquisition des Comportements *CIB*, *OBS*, *DEC*, et *ACT*

Les *SI* et *SHI* proposés pour l'acquisition des comportements *CIB*, *OBS*, *DEC*, et *ACT* exploitent les capacités d'*apprentissage* ou/et d'*adaptation*, et de *généralisation* de la *LF*, des *RN/RPG*, des *RN/AG*, des *RN/AG-RPG*, des *RNF/RPG*, des *RNAMF/AMFS*, ou des *RN/REE*.

- ***RN/RPG*, *RN/AG*, et *RN/AG-RPG***

L'utilisation des *RN/RPG* en *SI* classificateurs pour l'acquisition des comportements *CIB*, *OBS*, *DEC*, et *ACT* s'avère intéressante et indispensable si les *critères de classification* ou les *règles de généralisation* sont *inconnus*, puisque les *RN* sont capables d'apprendre et de généraliser à partir d'exemples sans aucune connaissance a priori des règles. Les *RN* sont ainsi bien adaptés aux tâches non-structurées telles que la reconnaissance de types. Les *RN* peuvent également déterminer des relations d'appartenance p.ex., apprendre à formuler des fonctions non-linéaires complexes à partir d'exemples d'entraînement telles que les étendues des

fonctions d'appartenance multi-dimensionnelles qui sont difficiles à concevoir (p.ex., la température, l'humidité, et la vitesse du vent).

Bien que l'apprentissage par la *RPG* a été largement utilisé avec succès, il *ne* présente *pas* une facilité de contrôle du calcul ("tractability") à cause du risque de non-convergence, correspondant au problème d'être 'piégé' dans un minimum *local* durant la descente du gradient. Ceci soulève une question très importante:

Question: existe-t-il une solution générale efficace pour l'apprentissage des *RN* multicouches ?

Les *AG* peuvent résoudre le problème des minimums *locaux* de la *RPG* en assurant une solution optimale (minimum *global*), mais soulève un autre problème lié à l'inhérente lenteur de convergence des *AG* qui peut être résolu par l'apprentissage par les *AG* suivi de la *RPG*.

- *LF*

L'utilisation de la *LF* en classificateurs pour l'acquisition des comportements intelligents s'avère intéressante et efficace si les *critères de classification* ou les *règles de généralisation* sont *connus*, donnés par un expert. De plus, les règles floues peuvent exprimer une large étendue de relations condition/action impliquant ainsi moins de règles par rapport aux *SE* conventionnels. Par ailleurs, la *LF* peut bien exprimer des valeurs qualitatives de la logique de l'être humain et fournit des actions, de manière graduelle à travers des fonctions d'appartenance continues, efficaces et intéressantes notamment pour les systèmes de contrôle. Les systèmes basés sur la *LF* peuvent aussi être présentés comme des systèmes, à raisonnement associatif parallèle, générés à partir de connaissances d'experts ou de données d'entraînement.

- *RNF/RPG*

La représentation à la fois *implicite* et *explicite* de la connaissance est nécessaire dans plusieurs problèmes en environnements réels, plus particulièrement dans la navigation des *VAI* en environnements dynamiques. Les *RNF* présentent cet avantage de traiter deux (02) types de connaissances en même temps. Ils présentent également l'avantage d'utiliser la *LF* afin d'améliorer la performance des systèmes basés sur les *RN*: les *RNF* peuvent recevoir et traiter des entrées floues, ce prétraitement transforme les données 'brutes' en données floues et le *RNF* apprend à correspondre les données floues aux données 'brutes' ou floues de sortie.

Par ailleurs, d'autres intégrations de la *LF* et des *RN* peuvent être intéressantes p.ex. les *RN* peuvent être utilisés afin d'améliorer des systèmes flous de contrôle en affinant les fonctions d'appartenance.

- ***RNAMF/AMFS***

Dans les *RNAMF*, la "ArtMap" Floue est utilisée pour une classification par apprentissage supervisé *rapide* et *stable* de données 'brutes' dans des applications telle que la reconnaissance de types. La *LF* est alors utilisée, dans ce cas, dans des *RN* améliorant ainsi leur performance. Un tel *RNAMF/AMFS* a l'avantage d'avoir uniquement un paramètre vigilance de base à sélectionner et ne nécessite aucun travail pour déterminer sa configuration (topologie) initiale. Il évolue en créant un (01) ou plusieurs neurones de sortie pour représenter une classe donnée, en démarrant de zéro. Essentiellement, bien que les *RNAMF/AMFS* classificateurs sont *puissants*, ils sont faciles à comprendre. Comparés aux *RN* classificateurs, la facilité de prise de décisions de leurs poids "Top-Down" est si transparente que les règles de classification peuvent être littéralement lus carrément du réseau. Ainsi, par ses grandes capacités d'*apprentissage*, d'*adaptation*, et de *généralisation* d'ensembles d'entraînement en un (01) seul cycle, ce type de réseaux devrait gagner rapidement la popularité détenue longtemps par les *RN/RPG*.

- ***RN/REE***

L'apprentissage supervisé par le Renforcement par *Essai et Erreur (REE)* est particulier puisqu'il est spécialisé dans le renforcement des situations aux actions favorables et la dissociation des situations des actions défavorables en procédant par la technique dite par essai et erreur.

IV. 2. 2. *SHI* pour la Planification et le Contrôle des Stratégies *NAV_1_Généralisée* et *NAV_2_Généralisée*

- ***NAV_1_Généralisée* et *NAV_2_Généralisée***

Les résultats de simulation ont démontré aussi l'*efficacité* des généralisations proposées des stratégies:

- *NAV_1_Simplifiée* en *NAV_1_Généralisée* notamment dans l'augmentation du

nombre de mouvements possibles,

- de *NAV_2_Simplifiée* en *NAV_2_Généralisée* notamment dans la prise en charge de plusieurs obstacles de différentes tailles à la fois.

La principale différence entre ces stratégies de navigation réside essentiellement dans le comportement *OBS*. Ce comportement se base sur une reconnaissance des situations topologiques (situations d'évitement d'obstacles) pour la stratégie *NAV_1_Généralisée*, alors qu'il se base sur une évaluation des degrés des dangers statique et dynamique des obstacles pour la stratégie *NAV_2_Généralisée*.

- *SHI_A_NAV_1_Généralisée*, *SHI_B_NAV_1_Généralisée*, et *SHI_SECOV_NAV_2_Généralisée*

Les résultats de simulation des stratégies de navigation *NAV_1_Généralisée* et *NAV_2_Généralisée* en environnements dynamiques partiellement structurés, planifiées et contrôlées par le *SHI_A_NAV_1_Généralisée*, par le *SHI_B_NAV_1_Généralisée*, ou par le *SHI_SECOV_NAV_2_Généralisée*, ont montrés que ces *SHI* fournissent aux *VAI* plus d'*autonomie* et plus d'*intelligence*. Une fois implémentés, ils sont capables de doter ces véhicules de capacités de traitement *temps réel*, réhaussant ainsi leur *sécurité*, leur *fiabilité*, et leur "*operability*".

Les approches basées sur le *SC* que nous avons proposées, pour la planification et le contrôle de la navigation des *VAI* en environnements dynamiques, ainsi que la plupart des approches développées à ce propos p.ex., [32, 34, 35, 37, 39, 41, 42, 44, 47, 53, 55, 56] sont basées sur l'acquisition de différents comportements intelligents par *apprentissage* ou/et *adaptation* tels que le contrôle du mouvement visuel, la reconnaissance d'objets, la localisation de la cible, la poursuite de cible, l'évitement d'obstacles, etc. Récemment, une perspective très prometteuse d'apprentissage dit de 'longue vie' pour les *VAI* a été présentée dans [12]. Dans cet article, deux (02) approches de ce nouveau type d'apprentissage ont été développées pour la navigation du véhicule HERO-2000 dans des environnements d'intérieur et inconnus. Les auteurs affirment que l'apprentissage devient plus facile, s'il est introduit dans un contexte d'apprentissage de 'longue vie'. Ils proposent de ne pas étudier les problèmes d'apprentissage du véhicule d'une manière isolé, mais plutôt dans le contexte de multitude de problèmes d'apprentissage auxquels un véhicule sera confronté et devra faire face pendant toute sa durée de vie. Ainsi, l'apprentissage de 'longue vie' du véhicule ouvre l'opportunité pour le transfert

des connaissances apprises. Les véhicules qui mémorisent et transfèrent des connaissances comptent moins sur l'expérimentation de l'environnement réel et apprennent ainsi plus rapidement.

IV. 3. IMPLEMENTATION D'UN RESEAU DE NEURONES (RN) SUR "FIELD-PROGRAMMABLE GATE ARRAY" (FPGA)

Une architecture *FPGA*, basée sur la technologie Xilinx, présentée dans la Figure IV. 1, est proposée pour l'implémentation du *SI_1: OBS_NAV_1_Simplifiée-RN/RPG* [129-131].

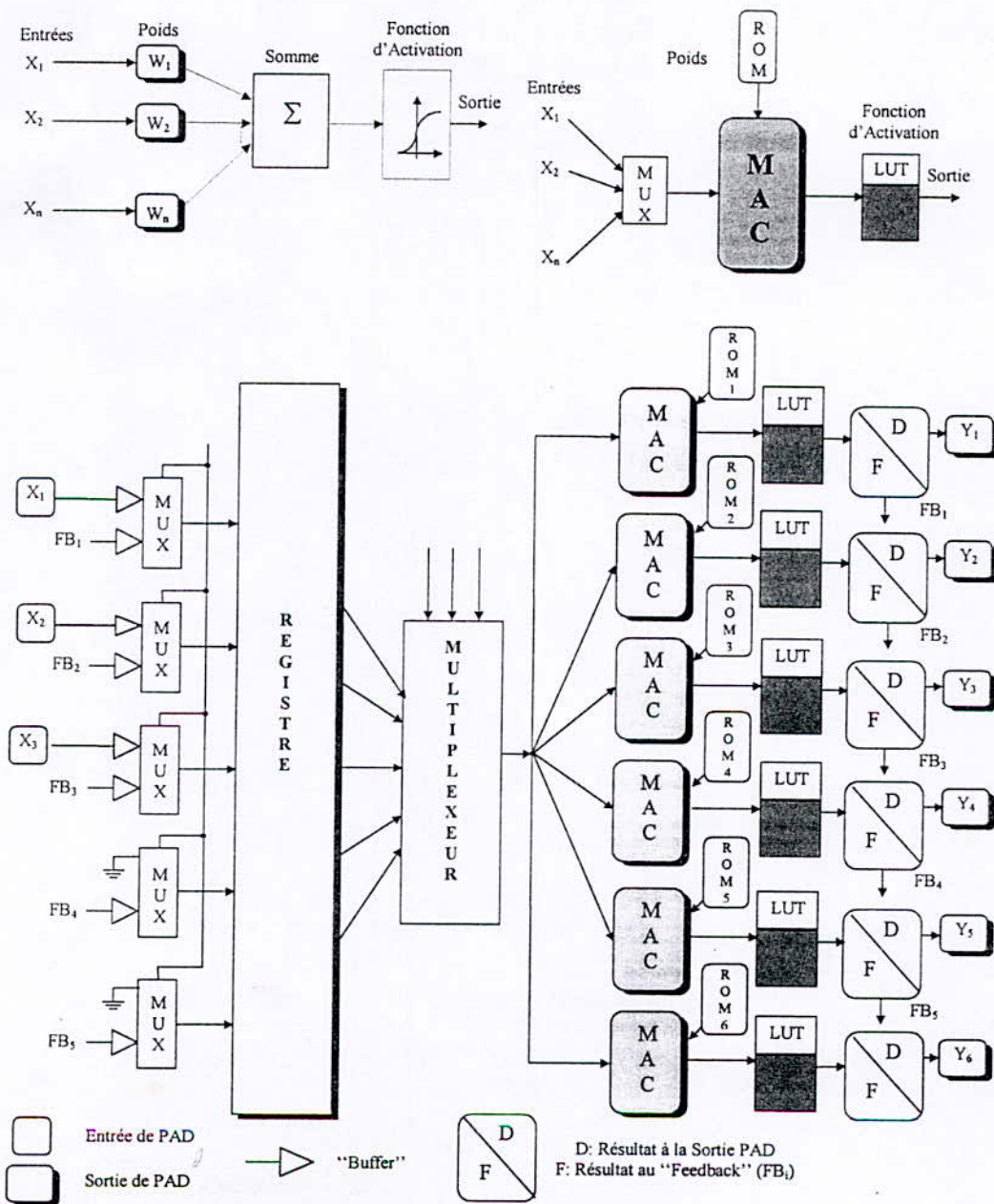


Figure IV. 1. Architecture *FPGA* pour le *SI_1: OBS_NAV_1_Simplifiée-RN/RPG*.

Chaque neurone des couches cachée et de sortie, de ce *SI*, est représenté par le modèle "hardware" de la Figure IV. 1. Ce modèle est principalement basé sur l'opérateur MAC ("Multiply and ACcumulate) qui calcule les sommes pondérées $\sum_i X_i W_{2ki}$ et $\sum_k Y_k W_{1jk}$. Le résultat du MAC pointe une LUT ("Look-Up Table") qui implémente la fonction d'activation sigmoïde.

L'algorithme décrivant la méthodologie adoptée est:

- 1- déterminer la couche, autre que la couche d'entrée, qui est constituée par le plus grand nombre de neurones N_n (dans ce cas c'est la couche de sortie $N_n = 6$),
- 2- le nombre de MAC requis dans l'architecture est alors $N_{MAC} = N_n$,
- 3- chaque MAC a sa propre ROM de poids, la profondeur de chaque ROM est égale au nombre de neurones constituant les couches d'entrée et cachée,
- 4- les entrées pour chaque neurone sont contrôlées par un multiplexeur (MUX), la taille S de ce multiplexeur est déterminée par le nombre de neurones qui constituent la plus importante couche autre que la couche de sortie (dans ce cas la taille est $S = 5:1$).

Cette architecture est implémentée sur un (01) seul *FPGA* XC4062EX de la technologie Xilinx et présente un grand degré de régularité. Elle peut être facilement modifiée (réduire ou augmenter le nombre de MAC et changer la taille du MUX).

Notons que cette architecture reste valable, sur un (01) *FPGA*, pour les *SI* et *SHI*:

- *SI_1: CIB_NAV_1_Simplifiée-RN/RPG*,
- *SHI_1: OBS_NAV_1_Simplifiée-RN/AG*,
- *SHI_2: OBS_NAV_1_Simplifiée-RN/AG-RPG*

en changeant uniquement les valeurs des poids.

Elle reste également valable, sur un (01) ou plusieurs *FPGA*, pour les *SI*:

- *SI_1: CIB_NAV_1_Généralisée-RN/RPG*,
- *SI_1: OBS_NAV_1_Généralisée-RN/RPG*,

en changeant les valeurs des poids, le nombre de MAC, et la taille des multiplexeurs.

IV. 4. PERSPECTIVES

Dans ce Paragraphe, nous présentons différentes perspectives jugées intéressantes tant sur l'aspect théorique que celui d'implémentation.

IV. 4. 1. Apprentissage Non-Supervisé

Il serait intéressant de développer des *SI* ou *SHI* pour l'acquisition des comportements intelligents par un apprentissage *non-supervisé* notamment celui développé par T. Kohonen [132] ainsi que celui basé sur la "ArtMap" Floue développé par G. A. Carpenter [74, 104, 105]. En fait, ce type d'apprentissage est nécessaire, voire indispensable, dans le cas où ni un ensemble d'exemples d'entraînement, ni une base de règles floues, et ni un expert humain ne sont disponibles. Dans un tel cas, un *RN* à apprentissage *non-supervisé* peut p.ex., analyser des données dans le but d'établir des règles floues.

IV. 4. 2. Systèmes Hybrides Intelligents (*SHI*)

Il serait intéressant de développer des *SHI* pour l'acquisition des comportements intelligents combinant la *LF*, les *AG*, et les *RN*. En effet, les structures génétiques peuvent représenter les paramètres de fonctions d'appartenance floues, les règles *Si ... Alors*, les connexions ou valeurs des poids dans un *RN*, la surface (l'étendue) de réponse d'un système de contrôle non-linéaire, etc. [74]. De plus, les *AG* peuvent développer des fonctions d'appartenances floues à représentations très satisfaisantes sans aide d'expertise humaine [95]. Par ailleurs, d'autres *SHI* peuvent être également intéressants en combinant les *RN* et la théorie du chaos [71, 133].

De plus, afin de rendre les *VAI* de plus en plus *robustes* et plus *fiables*, l'introduction de bases de vastes connaissances est recommandée. Par exemple, une base de vastes connaissances dans laquelle non seulement la compréhension et les expériences d'un conducteur humain sont stockés, mais aussi les expériences *accumulées* durant des missions de conduite vont être une nécessité dans les systèmes du futur [4]. De plus, le niveau de traitement des connaissances de tels systèmes devrait être organisé de telle manière à permettre l'application de concepts et résultats, des sciences de la psychologie et des processus par lesquels un être humain acquiert des informations, sur son environnement. Ceci offrira la possibilité de considérer ces systèmes basés sur la connaissance dans un cadre plus général [4].

IV. 4. 3. Cibles Dynamiques

Dans la plupart des applications actuelles des *VAI*, les cibles sont statiques correspondants à des tâches de transport ou de surveillance (navigation d'une position initiale à

une position cible données), de maintenance ou de chirurgie sur un site donné (position cible) par le biais d'un robot manipulateur (spécifique à chaque tâche) embarqué, d'explorations sous-marine ou spatiale, etc.

Le cas de cibles dynamiques implique nécessairement l'application des *VAI* à la tâche de poursuite. Cette tâche de poursuite est particulièrement intéressante dans la navigation en groupe. En effet, dans ce type de navigation, un véhicule considéré comme cible dynamique est poursuivi par les autres véhicules du groupe, menant ainsi tout le groupe à l'accomplissement d'une application donnée. Cette poursuite peut être en fait réalisée par la stratégie *NAV_1_Généralisée* où le comportement *CIB* proposé s'adapte *efficacement* aux changements de la position de la cible.

IV. 4. 4. Navigation en Groupes de *VAI*

Pour une tâche très complexe, un seul véhicule ne peut réaliser la tâche par lui-même à cause de ses limitations fonctionnelles. Dans ce cas, il est nécessaire de travailler en coopération avec d'autres véhicules, d'où la nécessité d'un comportement coorditatif dans les systèmes multi-véhicules. Ce comportement coorditatif nécessite le contrôle coopératif, l'interaction, les architectures distribuées de contrôle, et une grande capacité d'adaptation de ces véhicules [45, 134–138].

La navigation en groupes de *VAI* fait alors appel au domaine des systèmes multi-agents [139] impliquant un grand nombre de concepts nouveaux tels que la coopération, la coordination d'actions, la négociation, les conflits, la satisfaction, l'engagement, l'action et la réaction. Ces concepts trouvent leurs origines dans la *Sociologie* et la *Biologie*, voire dans l'*Ethologie*.

La recherche sur le comportement en groupe peut concerner plusieurs types tels que l'analyse de modélisation du comportement en groupe, la planification ou la prise de décisions comme le comportement en groupe et l'évolution du comportement en groupe. Le premier type est relativement proche du comportement des êtres vivants, alors que le second a été pris en charge particulièrement par l'intelligence artificielle distribuée où plusieurs agents coordonnent ou coopèrent pour prendre des décisions. Le troisième type est relativement proche de l'organisation et l'évolution des systèmes sociaux ou du travail d'équipe (collaboration).

IV. 4. 5. Implémentation des *SI* et *SHI* Proposés

La perspective d'implémentation des *SI* et des *SHI* basés sur les *RN*, la *LF*, les *RNF*, ou les *RNAMF* est tout à fait intéressante puisque la particularité commune à tous ces systèmes est celle du *parallélisme* inhérente aux *RN* et à la *LF*. Une alternative intéressante est alors d'implémenter ces systèmes sur *FPGA* [129, 130].

IV. 4. 6. Implémentation des Stratégies *NAV_1_Généralisée* et *NAV_2_Généralisée* Proposées sur l'*AGV*

Les capteurs à *UltraSons* (*US*) sont très utilisés pour la détection d'obstacles qui devra servir pour le comportement *OBS*, mais leurs signaux sont souvent bruités ou alors ils sont défectueux donnant des données incorrectes [35, 45]. Les *RN*, les *RNF*, et les *RNAMF* avec leurs caractéristiques inhérentes d'*adaptation* et de *grande tolérance aux défaillances* (pannes, bruits, ou erreurs) prennent en charge ce problème rendant les approches de planification et de contrôle de la navigation, basées sur ces types de réseaux, *robustes*. En effet, la défaillance de l'un des capteurs n'altérera pas fortement le comportement *OBS*.

Néanmoins, en perspective d'éventuelle implémentation des stratégies *NAV_1_Généralisée* et *NAV_2_Généralisée* proposées sur l' "Autonomous Guide Vehicle" (*AGV*), un certain nombre de voies ont besoin davantage d'investigations:

- le véhicule devra être doté d'un (01) ou de plusieurs mouvements de marche arrière,
- pour la détection d'obstacles *dynamiques*, le véhicule devra se baser sur des capteurs spécifiques et sur un traitement spécifique des données issues de ces capteurs,
- intégrer le comportement *ACT* aux stratégies *NAV_1_Généralisée* et *NAV_2_Généralisée* proposées,
- pour la stratégie *NAV_2_Généralisée*, le comportement *CIB* devra être acquis par des *SI* ou *SHI*,
- pour le Système Expert pour le COntôle des Véhicules (*SECOV*), il serait intéressant de développer des *RNAMF* pour la tâche du raisonnement flou et d'inférence floue.

IV. 5. CONCLUSION

Les résultats des entraînements et des mises en oeuvre des *SI* et *SHI*, proposés pour l'acquisition des comportements *CIB*, *OBS*, *DEC*, et *ACT* ont montré que ces systèmes exploitent les capacités d'*apprentissage* ou/et d'*adaptation*, et de *généralisation* de la *LF*, des *RN/RPG*, des *RN/AG*, des *RN/AG-RPG*, des *RNF/RPG*, des *RNAMF/AMFS*, ou des *RN/REE*. Ces résultats nous ont permis de dégager les meilleurs systèmes, en termes de facilité de mise en oeuvre et de performance, pour chaque comportement intelligent. Par conséquent, nous avons proposé des architectures globales en *SHI*, englobant chacun tous les comportements intelligents, pour la planification et le contrôle des stratégies *NAV_1_Généralisée* et *NAV_2_Généralisée*:

- *SHI_A_NAV_1_Généralisée*,
- *SHI_B_NAV_1_Généralisée*,
- *SHI_SECOV_NAV_2_Généralisée*.

Les résultats de simulation des stratégies de navigation *NAV_1_Généralisée* et *NAV_2_Généralisée* en environnements dynamiques partiellement structurés, planifiées et contrôlées par le *SHI_A_NAV_1_Généralisée*, par le *SHI_B_NAV_1_Généralisée*, ou par le *SHI_SECOV_NAV_2_Généralisée*, ont montré d'une part que ces *SHI* fournissent aux *VAI* plus d'*autonomie* et plus d'*intelligence*. D'autres part, ces résultats de simulation ont démontré l'*efficacité* des généralisations proposées à ces stratégies de navigation notamment par plus de possibilités de mouvements pour *NAV_1_Généralisée* et par la prise en charge de plusieurs obstacles à la fois de différentes tailles pour *NAV_2_Généralisée*.

L'exploitation des caractéristiques inhérentes d'*adaptation* et de *grande tolérance aux défaillances* de ces *SI* et *SHI* rendent les approches de planification et de contrôle de la navigation plus *robustes*. Une fois implémentés, ces *SI* et *SHI* sont capables de doter ces véhicules de capacités de traitement *temps réel*, réhaussant ainsi leur *sécurité*, leur *fiabilité*, et leur "*operability*". Pour cela, nous avons proposé une architecture *FPGA* pour l'implémentation du *SI_1: OBS_NAV_1_Simplifiée-RN/RPG* basé sur un *RN* réalisant le comportement *OBS*. Cette architecture reste valable pour d'autres *SI* proposés, sur un (01) ou plusieurs *FPGA* en changeant uniquement les valeurs des poids, le nombre de *MAC*, et la taille des multiplexeurs.

Dans ce Chapitre, nous avons également discuté d'éventuelles implémentations sur l'*AGV*, d'extensions, et développements de futures applications de ces stratégies de navigation.

En effet, l'introduction de l'apprentissage non-supervisé, de l'apprentissage dit de 'longue vie', de bases de vastes connaissances, de la théorie du chaos, et de la combinaison de la *LF*, des *AG*, et des *RN* permettront d'augmenter les capacités d'*autonomie*, d'*intelligence*, et de traitement *temps réel* des *VAI* dotés de ce type de stratégies de navigation, voir Figure IV. 2. De telles stratégies peuvent alors être exploitées pour réaliser des tâches de poursuite de cibles dynamiques ou de comportement en groupes.

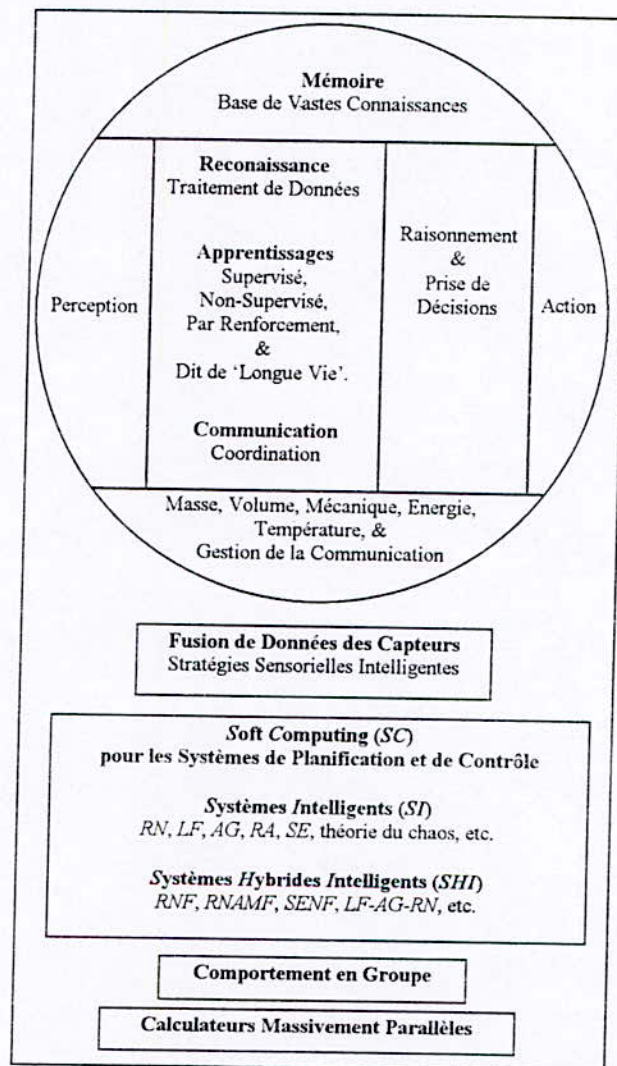


Figure IV. 2. Exigences Actuelles d'Autonomie et d'Intelligence des *VAI*.

CONCLUSION GENERALE

Nous avons développé, dans cette thèse, des approches de planification et de contrôle, basées sur le "Soft Computing" (SC), de la navigation des Véhicules Autonomes Intelligents (VAI) en environnements dynamiques. Ce travail a été effectué dans le but de faire évoluer les comportements de localisation de la cible (CIB), d'évitement d'obstacles (OBS), de prise de décisions (DEC), et d'action (ACT) de ces véhicules vers ceux de l'être humain dans de tels environnements. Ces comportements doivent fournir aux VAI plus d'autonomie et plus d'intelligence ainsi que des capacités de traitement temps réel.

Nous nous sommes alors intéressés à deux (02) stratégies de navigation, NAV_1_Simplifiée et NAV_2_Simplifiée, développées à partir d'études et d'observations des comportements de l'être humain en environnements dynamiques. En fait, ces stratégies intègrent essentiellement deux (02) manières différentes du raisonnement de l'être humain dans ses comportements intelligents en environnements dynamiques. Par ailleurs, bien que ces stratégies soient intéressantes elles présentent des problèmes dus principalement à des simplifications.

Afin de remédier à ces problèmes, nous avons proposé des généralisations:

- NAV_1_Simplifiée en NAV_1_Généralisée notamment dans l'augmentation du nombre de mouvements possibles de trois (03) à cinq (05),
- NAV_2_Simplifiée en NAV_2_Généralisée notamment dans la prise en charge de plusieurs obstacles à la fois de différentes tailles.

La principale différence entre ces stratégies de navigation réside dans leurs comportements OBS. Ce comportement se base sur une reconnaissance des situations topologiques (situations d'évitement d'obstacles) pour la stratégie NAV_1_Généralisée, alors qu'il se base sur une évaluation des degrés des dangers statique et dynamique des obstacles pour la stratégie NAV_2_Généralisée.

Le second problème a été alors d'acquérir les comportements intelligents CIB, OBS, DEC, et ACT de chaque stratégie de navigation par apprentissage ou/et adaptation. Pour cela, nous avons proposé des Systèmes Intelligents (SI) et des Systèmes Hybrides Intelligents (SHI) exploitant les capacités d'apprentissage ou/et d'adaptation, et de généralisation:

- des Réseaux de Neurones entraînés par la Rétro-Propagation du Gradient (RN/RPG),
- de la Logique Floue (LF),
- des RN entraînés par les Algorithmes Génétiques (RN/AG),
- des RN entraînés par les AG suivis de la RPG (RN/AG-RPG),

- des Réseaux de Neurones Flous entraînés par la *RPG* (*RNF/RPG*),
- des Réseaux de Neurones basés sur la "ArtMap" Floue entraînés par la "ArtMap" Floue Simplifiée (*RNAMF/AMFS*),
- des *RN* entraînés par le Renforcement par Essai et Erreur (*RN/REE*).

Les résultats de simulation des entraînements et des mises en oeuvre de ces *SI* et *SHI* nous ont permis de dégager les meilleurs systèmes, en termes de facilité de mise en oeuvre et de performance, pour chaque comportement intelligent. Par conséquent, nous avons proposé des architectures globales en *SHI*, englobant chacun tous les comportements intelligents, pour la planification et le contrôle des stratégies *NAV_1_Généralisée* et *NAV_2_Généralisée*:

- *SHI_A_NAV_1_Généralisée*,
- *SHI_B_NAV_1_Généralisée*,
- *SHI_SECOV_NAV_2_Généralisée*.

Les résultats de simulation des stratégies de navigation *NAV_1_Généralisée* et *NAV_2_Généralisée* en environnements dynamiques partiellement structurés, planifiées et contrôlées par le *SHI_A_NAV_1_Généralisée*, par le *SHI_B_NAV_1_Généralisée*, ou par le *SHI_SECOV_NAV_2_Généralisée*, ont montré d'une part que ces *SHI* fournissent aux *VAI* plus d'*autonomie* et plus d'*intelligence*. D'autre part, ces résultats de simulation ont démontré l'*efficacité* des généralisations proposées à ces stratégies de navigation notamment par plus de possibilités de mouvements pour *NAV_1_Généralisée* et par la prise en charge de plusieurs obstacles à la fois de différentes tailles pour *NAV_2_Généralisée*.

Par ailleurs, une fois implémentés, ces *SI* et *SHI* sont capables de doter ces véhicules de capacités de traitement *temps réel*, réhaussant ainsi leur *sécurité*, leur *fiabilité*, et leur "*operability*". Pour cela, nous avons proposé une architecture "Field-Programmable Gate Array" (*FPGA*) pour l'implémentation du *SI_1: OBS_NAV_1_Simplifiée-RN/RPG* basé sur un *RN* réalisant le comportement *OBS*.

En perspective d'une éventuelle implémentation des stratégies *NAV_1_Généralisée* et *NAV_2_Généralisée* proposées sur l'"Autonomous Guided Vehicle" (*AGV*), un certain nombre de voies ont besoin davantage d'investigations:

- le véhicule devra être doté d'un (01) ou de plusieurs mouvements de marche arrière,
- pour la détection d'obstacles *dynamiques*, le véhicule devra se baser sur des capteurs spécifiques et un traitement spécifique de données issues de ces capteurs,

- intégrer le comportement *ACT* aux stratégies *NAV_1_Généralisée* et *NAV_2_Généralisée* proposées,
- le comportement *CIB* devra être réalisé par des *SI* ou *SHI* pour la stratégie *NAV_2_Généralisée*,
- il serait intéressant de développer des *RNAMF* pour la tâche du raisonnement flou et d'inférence floue du Système Expert pour le *CON*trôle des Véhicules (*SECOV*),
- une alternative intéressante est alors d'implémenter les *SI* et *SHI* basés sur les *RN*, les *RNF*, et les *RNAMF* sur *FPGA*.

Nous pouvons également envisager l'utilisation de ces stratégies de navigation dans le cas de cibles dynamiques impliquant nécessairement l'application des *VAI* à la tâche de poursuite. Cette tâche de poursuite est particulièrement intéressante dans la navigation en groupe. En effet, dans ce type de navigation, un véhicule considéré comme cible dynamique est poursuivi par les autres véhicules du groupe, menant ainsi tout le groupe à l'accomplissement d'une application donnée. Cette poursuite peut être en fait réalisée par la stratégie *NAV_1_Généralisée* où le comportement *CIB* proposé s'adapte efficacement aux changements de la position de la cible.

Il serait intéressant aussi de développer les approches de planification et de contrôle par des *apprentissages non-supervisés* notamment celui développé par T. Kohonen ainsi que celui basé sur la "ArtMap" Floue développé par G. A. Carpenter. L'introduction également de l'apprentissage dit de 'longue vie', de bases de vastes connaissances, de la théorie du chaos, ainsi que la combinaison de la *LF*, des *AG*, et des *RN* s'avère intéressante.

Le progrès effectué, dans le développement de telles approches de planification et de contrôle de la navigation des *VAI* de plus en plus performantes notamment dans les exigences d'autonomie et les composants intelligents, ouvre la voie vers l'étude et l'exploitation des robots humanoïdes. Cette voie est appelée à jouer un rôle important dans le domaine de la recherche sur le fonctionnement du cerveau humain. Ainsi, des robots humanoïdes à bas-prix et à faible-poids vont contribuer à faire renaître l'intérêt dans l'intelligence des humanoïdes et pourquoi pas de fonctionner non seulement comme une machine mais de se joindre à notre environnement social et interagir avec les êtres humains, et jouer alors des rôles plus importants dans notre société.

REFERENCES BIBLIOGRAPHIQUES

- [1] Moreno, L., E. A. Puente, and M. A. Salichs. "World Modelling and Sensor Data Fusion in a Non Static Environment: Application to Mobile Robots." in *Proc. Int. IFAC Conf. Intelligent Components and Instruments for Control Applications*, Malaga, Spain, 1992, pp. 433-436.
- [2] Bosacchi, B. and I. Masaki. "Fuzzy Logic Technology & the Intelligent Highway System (IHS)." in *Proc. 2nd Int. IEEE Conf. Fuzzy Systems*, Vol. I, San Francisco, California, 1993, pp. 65-70.
- [3] Kujawski, C. "Deciding the Behaviour of an Autonomous Mobile Road Vehicle." in *Proc. 2nd Int. IFAC Conf. Intelligent Autonomous Vehicles*, Helsinki, Finland, June 12-14 1995, pp. 404-409.
- [4] Niegel, W. "Methodical Structuring of Knowledge Used in an Intelligent Driving System." in *Proc. 2nd Int. IFAC Conf. Intelligent Autonomous Vehicles*, Helsinki, Finland, June 12-14 1995, pp. 398-403.
- [5] Schilling, K. and C. Jungius. "Mobile Robots for Planetary Exploration." in *Proc. 2nd Int. IFAC Conf. Intelligent Autonomous Vehicles*, Helsinki, Finland, June 12-14 1995, pp. 110-120.
- [6] Chohra, A. and A. Farah. "Autonomy, Behaviour, and Evolution of Intelligent Vehicles." in *Proc. Int. IMACS IEEE-SMC Multiconf. Computational Engineering in Systems Applications*, Lille, France, 1996, pp. 36-41.
- [7] Bishop, R. "The Potential for Vehicle-Highway Automation in the United States." in *Proc. Int. IEEE/RSJ Conf. on Intelligent Robots and Systems*, Grenoble, France, 1997, pp. 1215-1216.
- [8] Konno, A., K. Nagashima, R. Furukawa, K. Nishiwaki, T. Noda, M. Inaba, and H. Inoue. "Development of a Humanoid Robot Saika." in *Proc. Int. IEEE/RSJ Conf. on Intelligent Robots and Systems*, Vol. 3, Grenoble, France, September 07-11, 1997, pp. 805-810.
- [9] Thorpe, C. "Mixed Traffic and Automated Highways." in *Proc. Int. IEEE/RSJ Conf. on Intelligent Robots and Systems*, Grenoble, France, 1997, pp. 1011-1017.
- [10] Volpe, R., T. Ohm, R. Petras, R. Welch, J. B. Balaram, and R. Ivlev. "A Prototype Manipulation System for Mars Rover Science Operations." *IEEE/RSJ Int. Conf on Intelligent Robots and Systems*, Grenoble, France, 1997, pp. 1486-1492.
- [11] Azouaoui, O. and A. Chohra. "Evolution, Behavior, and Intelligence Of Autonomous Robotic Systems (ARS)." in *Proc. 3rd Int. IFAC Conf. Intelligent Autonomous Vehicles*, Madrid, Spain, March 25-27 1998, pp. 139-145.
- [12] Thrun, S. and T. M. Mitchell. "Lifelong Robot Learning." *Robotics and Autonomous Systems*, 15, pp. 25-46, 1995.
- [13] Cherian, S. and W. Troxell. "Intelligent Behavior in Machines Emerging from a Collection of Interactive Control Structures." *Computational Intelligence*, Vol. 11, No. 4, pp. 565-592, November 1995.
- [14] Chattergy, R. "Some Heuristics for the Navigation of a Robot." *Int. J. Robotics Research*, Vol. 4, No. 1, pp. 59-66, 1985.
- [15] Crowley, J. L. "Navigation for an Intelligent Mobile Robot." *IEEE Journal of Robotics and Automation*, Vol. RA-1, No. 1, pp. 31-41, March 1985.
- [16] Jorgensen, C., W. Hamel, and C. Weisbin. "Autonomous Robot Navigation." *Byte*, pp. 223-235, January 1986.
- [17] Laumond, J.-P. "Feasible Trajectories for Mobile Robots with Kinematic and Environment Constraints." in *Proc. Int. Conf. on Autonomous Systems*, Amsterdam, Netherland, 1987, pp.---

- [18] Chohra, A. and F. Gaouar. *Stratégie de Navigation d'un Robot Mobile Simulation Physique: 2ème Partie*. Projet de Fin d'Etudes, Département d'Electronique, USTO, Oran, 28 Juin 1988.
- [19] Tsingpin, B. and L. Xueyin. "Safe Path Network: A New Approach To Path-Planning." in *Proc. Int. IEEE Conf. ---*, CH2556-9/88/0000-0917, ---, 1988, pp. 917-921.
- [20] Borenstein, J. and Y. Koren. "Real-Time Obstacle Avoidance for Fast Mobile Robots." *Trans. on Systems, Man, and Cybernetics*, Vol. 19, No. 5, pp. 1179-1187, September/October 1989.
- [21] Adams, M. D., H. Hu, and P. J. Probert. "Towards a Real-Time Architecture for Obstacle Avoidance and Path Planning in Mobile Robots." in *Proc. Int. IEEE Conf. ---*, CH2876-1/90/0000-0584, ---, 1990, pp. 584-589.
- [22] Chohra, A. *Pilotage et Asservissement Sensoriel d'un Robot Mobile Autonome*. Thèse de Magister, Laboratoire de Robotique et d'Intelligence Artificielle (LRIA) du Centre de Développements des Technologies Avancées (CDTA), Alger, 07 Juillet 1991.
- [23] Foux, G., M. Heymann, and A. Bruckstein. "Two-Dimensional Robot Navigation Among Unknown Stationary Polygonal Obstacles." *IEEE Trans. on Robotics and Automation*, Vol. 9, No. 1, pp. 096-102, February 1993.
- [24] Pignon, P., T. Hasegawa, and J. P. Laumond. "Structuration de l'Espace pour les Robots Mobiles." *Revue Scientifique et Technique de la Défense*, 2, pp. 17-31, 1994.
- [25] Hogle, R. A. and P. P. Bonissone. "A Fuzzy Algorithm for Path Selection in Autonomous Vehicle Navigation." in *Proc. 23rd Conf. on Decision and Control*, Las Vegas, NV, December 1984, pp. 898-900.
- [26] Sugeno, M. and K. Murakami. "Fuzzy Parking Control of Model Car." in *Proc. 23rd IEEE Conf. on Decision and Control*, Las Vegas, NV, December 1984, pp. 902-904.
- [27] Sugeno, M. and K. Murakami. "An Experimental Study on Fuzzy Parking Control Using a Model Car." *Industrial Applications of Fuzzy Control*, North-Holland Amsterdam, pp. 125-138, 1985.
- [28] Sugeno, M. and M. Nishida. "Fuzzy Control of Model Car." *Fuzzy Sets and Systems*, 16, North-Holland, pp. 103-113, 1985.
- [29] Ciliz, M. K. and C. Isik. "Fuzzy Rule-Based Motion Controller for an Autonomous Mobile Robot." *Robotica*, Vol. 7, pp. 37-42, 1989.
- [30] Sorouchyari, E. "Mobile Robot Navigation: A Neural Network Approach." in *Proc. Art du Colloque Neuromimétique*, Ecole Polytechnique de Lausanne, 1989, pp. 159-175.
- [31] Ho, A. W. and G. C. Fox. "Neural Network Near-Optimal Motion Planning for a Mobile Robot on Binary and Varied Terrains." *IEEE Int. Work. on Intelligent Robots and Systems, IROS'90*, 1990, pp. 593-600.
- [32] Maeda, Y. "Collision Avoidance Control Among Moving Obstacles for a Mobile Robot on the Fuzzy Reasoning." in *Proc. Eight CISM IFToMM Symp. Theory and Practice of Robots & Manipulators*, AGH, Cracow, Poland, 2-6 July 1990, pp. ---.
- [33] Takegaki, M. and T. Ishioka. "An Expert Tool for Real Time Intelligent Control: ERIC." in *Proc. Int. ISMM Conf. on Computer Applications in Design, Simulation and Analysis*, 1990, pp. 159-162.
- [34] Baloch, A. A. and A. M. Waxman. "Visual Learning, Adaptive Expectations, and Behavioral Conditioning of the Mobile Robot MAVIN." *Neural Networks*, Vol. 4, pp. 271-302, 1991.

- [35] Berns, K. and R. Dillmann. "A Neural Network Approach for the Control of a Tracking Behavior." in *Proc. Int. IEEE Conf. ---, 7803-0078/91/0600-0500, ---, 1991*, pp. 500-503.
- [36] Lin, C. S. and C. D. Wann. "A Parallel Processing Model for Robot Path Planning on Grid Terrains." *Int. J. Robotics and Automation*, Vol. 6, No. 1, pp. 01-11, 1991.
- [37] Maeda, M., Y. Maeda, and S. Murakami. "Fuzzy Drive Control of an Autonomous Mobile Robot." *Fuzzy Sets and Systems*, 39, North-Holland, pp. 195-204, 1991.
- [38] Shen, W., J. Shen, and J. P. Lallemand. "Finding the Shortest Path by Use of Neural Networks." in *Proc. Int. IEEE Conf. ---, 7803-0078/91/0600-1164, ---, 1991*, pp. 1164-1169.
- [39] Takeuchi, T. "An Autonomous Fuzzy Mobile Robot." *Advanced Robotics*, Vol. 5, No. 2, pp. 215-230, 1991.
- [40] Chohra, A. *Contrôle Intelligent d'un Robot Mobile Autonome*. Rapport Technique, Laboratoire de Robotique et d'Intelligence Artificielle (LRIA) du Centre de Développements des Technologies Avancées (CDTA), A. Chohra/CDTA.008, Octobre 1992.
- [41] Herbstreith, H., L. Gmeiner, and P. Preuß. "A Target-Directed Neurally Controlled Vehicle." in *Proc. Int. IFAC Conf. Artificial Intelligence in Real-Time Control*, Delft, The Netherlands, 1992, pp. 67-71.
- [42] Meng, M. and A. C. Kak. "Mobile Robot Navigation Using Neural Networks and Nonmetrical Environment Models." *IEEE Control Systems*, pp. 30-39, October 1993.
- [43] Chohra, A. and C. Benmehrez. "Planning and Intelligent Control of Autonomous Mobile Robots in Partially Structured Environments." in *Proc. Int. IMACS-IEEE/SMC Symp. Signal Processing, Robotics and Neural Networks*, Lille, France, 1994, pp..
- [44] Dubrawski, A. and J. L. Crowley. "Self-Supervised Neural System for Reactive Navigation." in *Proc. Int. IEEE Conf. on Robotics and Automation*, Vol. 3, San Diego, California, May 08-13, 1994, pp. 2076-2081.
- [45] Floreano, D. and F. Mondada. "Automatic Creation of an Autonomous Agent: Genetic Evolution of a Neural-Network Driven Robot." in *Proc. 3rd Int. Conf. on Simulation of Adaptive Behavior*, Brighton, 1994, pp..
- [46] Chohra, A., F. Sif, and S. Talaoubrid. "Neural Navigation Approach of an Autonomous Mobile Robot in a Partially Structured Environment." in *Proc. 2nd Int. IFAC Conf. Intelligent Autonomous Vehicles*, Helsinki, Finland, June 12-14 1995, pp. 238-243.
- [47] Hiraga, I., T. Furuhashi, Y. Uchikawa, and S. Nakayama. "An Acquisition of Operator's Rules for Collision Avoidance Using Fuzzy Neural Networks." *IEEE Trans. on Fuzzy Systems*, Vol. 3, No. 3, pp. 280-287, August 1995.
- [48] Ashiru, I., C. Czarnecki, and T. Routen. "Characteristics of a Genetic Based Approach to Path Planning for Mobile Robots." *Journal of Network and Computer Applications*, 19, pp. 149-169, 1996.
- [49] Chohra, A., A. Farah, and R. Benabbas. "Neuro-Fuzzy Navigation Approach for Autonomous Mobile Robots in Partially Structured Environments." in *Proc. Int. Conf. Application of Fuzzy Systems and Soft Computing*, Siegen, Germany, 1996, pp. 304-313.
- [50] Chohra, A. and A. Farah. "Hybrid Navigation Approach Combining Neural Networks and Fuzzy Logic for Autonomous Mobile Robots." in *Proc. 3rd Int. Conf. Motion and Vibration Control*, Chiba, Japan, 1996, pp..

- [51] Chohra, A., C. Benmehrez, A. Farah, and S. B. Bouguettaya. "Neural Navigation Approach for Autonomous Mobile Robots in Partially Structured Environments." *Algerian Journal of Technology*, Ecole Nationale Polytechnique International Publication: Electronique, Informatique, Electro-Technique, et Automatique, Série B, Vol. 12, No. 1, pp. 07-14, 1996.
- [52] Fukuda, T., F. Arai, and K. Shimojima. "Intelligent Robotic System." in *Proc. Int. IMACS IEEE-SMC Multiconf. Computational Engineering in Systems Applications*, Lille, France, 1996, pp. 01-10.
- [53] Szykarczyk, P. and A. Masiowski. "The Fuzzy ARTMAP Neural Networks as a Controller for the Mobile Robot." in *Proc. 3rd Int. Symp. on Methods and Models in Automation and Robotics*, Miedzyzdroje, Poland, September 10-13, 1996, pp. 1201-1206.
- [54] Chohra, A., A. Farah, and M. Belloucif. "Neuro-Fuzzy Expert System *E_S_CO_V* for the Obstacle Avoidance of Intelligent Autonomous Vehicles (IAV)." in *Proc. Int. IEEE/RSJ Conf. on Intelligent Robots and Systems*, Vol. 3, Grenoble, France, September 07-11, 1997, pp. 1706-1713.
- [55] Dubrawski, A. "Tuning Neural Networks with Stochastic Optimization." in *Proc. Int. IEEE/RSJ Conf. on Intelligent Robots and Systems*, Vol. 2, Grenoble, France, September 07-11, 1997, pp. 614-620.
- [56] Xiao, J., Z. Michalewicz, L. Zhang, and K. Trojanowski. "Adaptive Evolutionary Planner/Navigator for Mobile Robots." *IEEE Trans. on Evolutionary Computation*, Vol. 1, No. 1, pp. 18-28, April 1997.
- [57] Chohra, A. "Fuzzy ArtMap Neural Networks (FAMNN) Based Navigation for Intelligent Autonomous Vehicles (IAV) in Partially Structured Environments." in *Proc. 3rd Int. IFAC Conf. Intelligent Autonomous Vehicles*, Madrid, Spain, March 25-27 1998, pp. 304-309.
- [58] Chohra, A., A. Farah, and C. Benmehrez. "Neural Navigation Approach for Intelligent Autonomous Vehicles (IAV) in Partially Structured Environments." *The International Journal of Applied Intelligence: Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies*, Vol. 8, No. 3, Kluwer Academic Publishers, pp. 219-233, May/June 1998.
- [59] Chohra, A. "Fuzzy ArtMap Neural Networks (FAMNN) Based Navigation for Intelligent Autonomous Vehicles (IAV) in Partially Structured Environments." *ICSC Int. IFAC-IEEE Conf. on Neural Computation*, Austria, 1998, pp. 747-754.
- [60] A. Chohra, A. Farah, and M. Belloucif. "Neuro-Fuzzy Expert System *E_S_CO_V* for the Obstacle Avoidance Behavior of Intelligent Autonomous Vehicles (IAV)." *International Journal of Advanced Robotics*, Robotics Society of Japan, Vol. 12, No. 6, pp. 629-650, 1999.
- [61] Pedrycz, W. *Fuzzy Control and Fuzzy Systems*. John Wiley & Sons Inc., 1988.
- [62] Khanna, T. *Foundations of Neural Networks*. Addison-Wesley, New York, 1990.
- [63] Bosc, P., B. Bouchon-Meunier, D. Dubois, L. Foulloy, M. Grabisch, H. Prade, E. Sanchez, and L. A. Zadeh. "Principales Applications des Ensembles Flous." *La Lettre du Club CRIN Logique Floue*, No. Spécial, Mai 1992.
- Zadeh, L. A. "Soft Computing, Fuzzy Logic and Neural Networks." pp. 02-03.
- Foulloy, L. "La Commande Floue." pp. 03-04.
- Bosc, P. "Les Systèmes d'Information Flous." pp. 05-06.
- Dubois, D., and H. Prade. "Le Calcul avec des Quantités Floues." pp. 07-08.
- Bouchon-Meunier, B. "Les Raisonnements Approximatifs et Incertains." pp. 09-10.

- Grabisch, M. "Le Regroupement et la Classification Flous." pp. 11-12.
- Dubois, D., and H. Prade: "L'Agrégation Multi-Critère et l'Optimisation." pp. 13-14.
- Sanchez, E. "L'Apprentissage Flou et Neuronale." pp. 15-16.
- [64] Freeman, J. A. and D. M. Skapura. *Neural Networks: Algorithms, Applications, and Programming Techniques*. Addison-Wesley, New York, 1992.
- [65] Grabisch, M. *Le Phénomène Japonais: Mode ou Réalité ? Rapport*, Thomson-Sintra ASM, Avril 1992.
- [66] Holland, J. "Les Algorithmes Génétiques." *Pour La Science*, No. 179, pp. 44-51, Septembre 1992.
- [67] Kosko, B. *Neural Networks and Fuzzy Systems*. University of Southern California, Prentice Hall, A Simon & Schuster Company, Englewood Cliffs, New Jersey 07632, 1992.
- [68] Werbos, P. J. "Neurocontrol and Fuzzy Logic: Connections and Designs." *Int. J. of Approximate Reasoning*, Vol. 6, pp. 185-219, 1992.
- [69] Kung, S. Y. *Digital Neural Networks*. PTR Prentice-Hall, New Jersey 07632, 1993.
- [70] Goldberg, D. E. *Algorithmes Génétiques: Exploration, Optimisation et Apprentissage Automatique*. Addison-Wesley, France, 1994.
- [71] Welstead, S. T. *Neural Network and Fuzzy Logic Applications in C/C++*. Jhon Wiley & Sons Inc., Toronto, 1994.
- [72] Anderson, J. A. *An Introduction to Neural Networks*. The MIT Press, Cambridge, Massachusetts, London, England, 1995.
- [73] Medsker, L. R. *Hybrid Intelligent Systems*. Kluwer Academic Publishers, 1995.
- [74] Patterson, D. W. *Artificial Neural Networks: Theory and Applications*. Prentice-Hall, Simon & Schuster (Asia) Pte Ltd, Singapore, 1996.
- [75] Marco, D. B., A. J. Healey, and R. B. McGhee. "Autonomous Underwater Vehicles: Hybrid Control of Mission and Motion." *Autonomous Robots*, Vol. 3, No. 2/3, pp. 169-186, 1996.
- [76] Shibata, T. and T. Fukuda. "Hierarchical Intelligent Control for Robotic Motion." *IEEE Trans. on Neural Networks*, Vol. 5, No. 5, pp. 823-832, 1994.
- [77] Visinsky, M. L., I. D. Walker, and J. R. Cavallaro. "New Dynamic Model-Based Fault Detection Thresholds for Robot Manipulators." in *Proc. Int. IEEE Conf. on Robotics and Automation*, Vol. 2, San Diego, California, May 08-13, 1994, pp. 1388-1395.
- [78] Langen, A. and W. Baum "Reliability and Safety for Mobile Robots in Hostile Environment." in *Proc. 2nd Int. IFAC Conf. on Intelligent Autonomous Vehicles*, Helsinki, Finland, June 12-14 1995, pp. 326-330.
- [79] Lima, P. and J. Senteiro. "Research on Intelligent Control Methodologies at the Instituto de Sistemas e Robotica of Instituto Superior Técnico." *IMACS IEEE-SMC Multiconf. on Computational Engineering in Systems Applications*, Lille, France, 1996, pp. 623-628.
- [80] McKerrow, P. J. *Introduction to Robotics*. University of Wollongong, Australia, Addison-Wesley Publishing Company, 1991.
- [81] Hurteau, R., S. DeSantis, E. Begin, and M. Gagner. "Laparoscopic Surgery Assisted by a Robotic Cameraman: Concept and Experimental Results." in *Proc. Int. IEEE Conf. on Robotics and Automation*, Vol. 3, San Diego, California, May 08-13, 1994, pp. 2286-2289.

- [82] Salcudean, S. E. and J. Yan. "Towards a Force-Reflecting Motion-Scaling System for Microsurgery." in *Proc. Int. IEEE Conf. on Robotics and Automation*, Vol. 3, San Diego, California, May 08-13, 1994, pp. 2296-2301.
- [83] Schempf, H. "Neptune: Above-Ground Storage Tank Inspection Robot System." in *Proc. Int. IEEE Conf. on Robotics and Automation*, Vol. 2, San Diego, California, May 08-13, 1994, pp. 1403-1408.
- [84] Kim, K., M. W. Vannette, J. L. Hall, and D. R. Flugrad. "Remote Manipulator and its Teleoperated Guide Vehicle for Hazardous Waste Sampling Operations." in *Proc. Int. IEEE Conf. on Robotics and Automation*, Vol. 3, San Diego, California, May 08-13, 1994, pp. 2620-2625.
- [85] Tanaka, T., Y. Kojima, J. Ohwi, K. Yamafuji and S. V. Ulyanov. "Intelligent Control of Technology Operations for Robot of Service Use with Manipulator." in *Proc. Int. IMACS IEEE-SMC Multiconf. Computational Engineering in Systems Applications*, Lille, France, 1996, pp. 788-793.
- [86] McMillan, S., D. E. Orin, and R. B. McGhee. "Efficient Dynamic Simulation of an Unmanned Underwater Vehicle with a Manipulator." in *Proc. Int. IEEE Conf. on Robotics and Automation*, Vol. 2, San Diego, California, May 08-13, 1994, pp. 1133-1140.
- [87] Meghdari, A. "The Cooperative Dual-Arm Cam-Lock Manipulators." in *Proc. Int. IEEE Conf. on Robotics and Automation*, Vol. 2, San Diego, California, May 08-13, 1994, pp. 1279-1285.
- [88] Hirzinger, G., B. Brunner, J. Dietrich, and J. Heindl. "ROTEX - The First Remotely Controlled Robot in Space." in *Proc. Int. IEEE Conf. on Robotics and Automation*, Vol. 3, San Diego, California, May 08-13, 1994, pp. 2604-2611.
- [89] Hwang, J., E. Wu, A. Bell, A. Cordell, L. Stokes, and S. Hankins. "Design of a SPDM-Like Robotic Manipulator System for Space Station on Orbit Replaceable Unit Ground Testing - An Overview of the System Architecture." in *Proc. Int. IEEE Conf. on Robotics and Automation*, Vol. 2, San Diego, California, May 08-13, 1994, pp. 1286-1291.
- [90] Masutani, Y., T. Iwatsu, and F. Miyazaki. "Motion Estimation of Unknown Rigid Body under No External Forces and Moments." in *Proc. Int. IEEE Conf. on Robotics and Automation*, Vol. 2, San Diego, California, May 08-13, 1994, pp. 1066-1072.
- [91] Lim, D., T. S. Lee, and H. Seraji. "A Real-Time Control System for a Mobile Dexterous 7 DOF Arm." in *Proc. Int. IEEE Conf. on Robotics and Automation*, Vol. 2, San Diego, California, May 08-13, 1994, pp. 1188-1195.
- [92] Ollero, A., A. Simon, F. Garcia, and V. E. Torres. "Integrated Mechanical Design and Modelling of a New Mobile Robot." in *Proc. Int. IFAC Conf. Intelligent Components and Instruments for Control Applications*, Malaga, Spain, 1992, pp. 461-466.
- [93] Brady, M. "Special Issue on Sensor Data Fusion." *Foreword of Int. J. of Robotics Research*, Vol. 7, No. 6, pp. 02-04, 1988.
- [94] Henderson, T. C. and R. Grupen. "Logical Behaviors." *Journal of Robotic Systems*, 7 (3), pp. 309-336, 1990.
- [95] Kim, J., Y. Moon, and B. P. Zeigler. "Designing Fuzzy Net Controllers Using Genetic Algorithms." *IEEE Control Systems*, pp. 66-72, June 1995.

- [96] Kaufmann, A. *Introduction à la Théorie des Sous-Ensembles Flous à l'Usage des Ingénieurs (Fuzzy Sets Theory)*, Masson et Cie, Tome 1. *Eléments Théoriques de Base*, 2^e Edition 1977, Tome 2. *Applications à la Linguistique à la Logique et à la Sémantique*, 1975, Tome 3. *Applications à la Classification et à la Reconnaissance des Formes, aux Automates et aux Systèmes, au Choix des Critères*, 1975, Tome 4. *Compléments et Nouvelles Applications*, 1977.
- [97] Khatib, O. "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots." *Int. J. of Robotics Research*, Vol. 5, No. 1, pp. 90-98, 1986.
- [98] Krijgsman, A. J., H. B. Verbruggen, and M. G. Rodd. "Intelligent Control: Theory and Applications." in *Proc. Int. IFAC Conf. Intelligent Components and Instruments for Control Applications*, Malaga, Spain, 1992, pp. 57-67.
- [99] Zadeh, L. A. "Fuzzy Sets." *Information & Control*, Vol. 8, pp. 338-353, 1965.
- [100] Lee, C. C. "Fuzzy Logic in Control Systems: Fuzzy Logic Controller, Part I and Part II." *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 20, No. 2, pp. 404-435, March/April 1990.
- [101] Zadeh, L. A. "The Calculus of Fuzzy If/Then Rules." *AI Expert*, pp. 23-27, March 1992.
- [102] Zadeh, L. A. "Fuzzy Logic = Computing with Words." *IEEE Trans. on Fuzzy Systems*, Vol. 4, No. 2, pp. 103-111, May 1996.
- [103] Montana, D. J. and L. Davis. "Training Feedforward Neural Networks Using Genetic Algorithms." in *Proc. 11th Int. Joint Conf. on Artificial Intelligence*, Detroit MI., Morgan Kaufman, San Mateo CA., 1989, pp. 762-767.
- [104] Carpenter, G. A., S. Grossberg, and D. B. Rosen. "Fuzzy ART: Fast Stable Learning and Categorization of Analog Patterns by an Adaptive Resonance System." *Neural Networks*, Vol. 4, pp. 759-771, 1991.
- [105] Carpenter, G. A., S. Grossberg, N. Markuzon, J. H. Reynolds, and D. B. Rosen. "Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps." *IEEE Trans. on Neural Networks*, Vol. 3, No. 5, pp. 698-713, September 1992.
- [106] Yen, J. "Using Fuzzy Logic to Integrate Neural Networks and Knowledge-Based Systems." *2nd Joint Technical Workshop on Neural Networks and Fuzzy Logique*, Vol. 1, 1991, pp. 217-233.
- [107] Mitra, S. and S. K. Pal. "Fuzzy Multi-Layer Perceptron, Inferencing and Rule Generation." *IEEE Trans. on Neural Networks*, Vol. 6, No. 1, pp. 51-63, January 1995.
- [108] Chatila, R. "Deliberation and Reactivity in Autonomous Mobile Robots." *Robotics and Autonomous Systems*, 16, pp. 197-211, 1995.
- [109] Borenstein, J. and Y. Koren. "Obstacle Avoidance with Ultrasonic Sensors." *IEEE J. of Robotics and Automation*, Vol. 4, No. 2, pp. 213-218, April 1988.
- [110] Maslin, G. D. "A Simple Ultrasonic Ranging System." *Polaroid Ultrasonic Ranging System Handbook Application Notes/Technical Papers*, Polaroid Corporation, 1983.
- [111] Steer, B. "Trajectory Planning for a Mobile Robot." *Int. J. of Robotics Research*, Vol. 8, No. 5, pp. 03-14, October 1989.
- [112] Vasseur, H. A., F. G. Pin, and J. R. Taylor. "Navigation of Car-Like Mobile Robots in Obstructed Environments Using Convex Polygonal Cells." *Int. J. of Robotics and Autonomous Systems*, 10, pp. 133-146, 1992.

- [113] Vestli, S. J., N. Tschichold-Gürman, M. Adams, and S. Sulzberger. "Integration of Path Planning, Sensing and Control in Mobile Robotics." in *Proc. Int. IEEE Conf. on Robotics and Automation*, Vol. 3, Atlanta, Georgia, May 02-06, 1993, pp. 243-248.
- [114] McInerney, M. and A. P. Dhawan. "Use of Genetic Algorithms with Back Propagation in Training of Feed-Forward Neural Networks." in *Proc. Int. IEEE Conf. Neural Networks*, Vol. I, San Francisco, California, March 28-April 1 1993, pp. 203-208.
- [115] Kitano, H. "Empirical Studies on the Speed of Convergence of Neural Network Training Using Genetic Algorithms." in *Proc. 8th JMIT National Conf. on Artificial Intelligence*, Boston MASS, Vol. 2, 1990, pp. 789-795.
- [116] Glorennec, P. Y. "Les Réseaux Neuro-Flous Evolutifs: un Pont entre le Flou et le Neuronal." Exposé à Paris, dans le cadre du Projet de Recherche: "Commande Symbolique et Neuro-Mimétique", *INSA de Rennes*, pp. 01-24, 03 Octobre 1991.
- [117] Hammerstrom, D. "Working with Neural Networks." *IEEE Spectrum*, pp. 46-53, July 1993.
- [118] Hung, C. C. "Building a Neuro-Fuzzy Learning Control System." *AI Expert*, Vol. 8, No. 11, pp. 40-49, November 1993.
- [119] Pedrycz, W. "Fuzzy Neural Networks and Neurocomputations." *Fuzzy Sets and Systems*, Vol. 56, No. 1, pp. 01-28, May 1993.
- [120] Ishibuchi, H., K. Morioka, and I. B. Turksen. "Learning by Fuzzified Neural Networks." *Int. J. of Approximate Reasoning*, Vol. 13, No. 4, pp. 327-358, November 1995.
- [121] Kasuba, T. "Simplified Fuzzy ARTMAP." *AI Expert*, Vol. 8, No. 11, pp. 18-25, November 1993.
- [122] Lee, H.-M. and C.-S. Lai. "Supervised Extended ART: A Fast Neural Network Classifier Trained by Combining Supervised and Unsupervised Learning." *The International Journal of Applied Intelligence: Artificial Intelligence, Neural Networks, and Complex Problem-Solving Technologies*, 6, Kluwer Academic Publishers, pp. 117-128, 1996.
- [123] Hayashi, Y. "A Neural Expert System Using Fuzzy Teaching Input." in *Proc. Int. IEEE Conf. on Fuzzy Systems*, San Diego, 1992, pp. 485-491.
- [124] Keller, J. M., R. R. Yager, and H. Tahani. "Neural Network Implementation of Fuzzy Logic." *Fuzzy Sets and Systems*, Vol. 45, pp. 01-12, 1992.
- [125] Buckley, J. J. and Y. Hayashi. "Neural Nets for Fuzzy Systems." *Fuzzy Sets and Systems*, Vol. 71, No. 3, pp. 265-276, May 12, 1995.
- [126] Brogan, W. L. *Modern Control Theory*. Prentice-Hall, NJ, pp. 433-465, 1985.
- [127] Farreny, H. and H. Prade. "Tackling Uncertainty and Imprecision in Robotics." in *Proc. 3rd Int. Symp. on Robotics Research*, Gonvieux, 1985, pp. 85-91.
- [128] Nguyen, H. T. and V. Kreinovich. "Towards Theoretical Foundations of Soft Computing Applications." *Int. J. of Uncertainty, Fuzziness and Knowledge-Based Systems*, Vol. 3, No. 3, pp. 341-373, September 1995.
- [129] Cox, C. E. and W. E. Blanz. "GANGLION - A Fast Field-Programmable Gate Array Implementation of a Connectionist Classifier." *IEEE Journal of Solid-State Circuits*, Vol. 27, No. 3, pp. 288-299, March 1992.

- [130] Botros, N. M. and M. Abdul-Aziz. "Hardware Implementation of an Artificial Neural Network." in *Proc. Int. IEEE Conf. on Neural Networks*, Vol. III, San Francisco, California, March 28-April 01 1993, pp. 1252-1257.
- [131] El-Haffaf, Y. I., A. K. Oudjida, and A. Chohra. "Digital Artificial Neural Network Architecture for Obstacle Avoidance Suitable for FPGA." in *Proc. Int. Conf. on Engineering Applications of Neural Networks*, Stockholm, Sweden, June 16-18 1997, pp. 313-316.
- [132] Teuvo Kohonen, T. *Associative Memory: A System-Theoretical Approach*. Corrected Printing of the First Edition, Springer-Verlag, Berlin Heidelberg New York, 1978.
- [133] Dingle, A. A., J. H. Andreae, and R. D. Jones. "A Chaotic Neural Unit." in *Proc. Int. IEEE Conf. on Neural Networks*, Vol. I, San Francisco, California, March 28-April 01 1993, pp. 335-340.
- [134] Shibata, T. and T. Fukuda. "Coordinative Behavior in Evolutionary Multi-Agent System by Genetic Algorithm." in *Proc. Int. IEEE Conf. Neural Networks*, Vol. I, San Francisco, California, 1993, pp. 209-214.
- [135] Martinoli, A. and F. Mondada. "Collective and Cooperative Group Behaviours: Biologically Inspired Experiments in Robotics." *Fourth Int. Symp. on Exp. Rob. ISER '95*, California, 1995, pp.
- [136] Asama, H., M. Sato, N. Goto, H. Kaetsu, A. Matsumoto, and I. Endo. "Mutual Transportation of Cooperative Mobile Robots Using Forklift Mechanisms." in *Proc. IEEE Conf. on Robotics and Automation*, Minneapolis, Minnesota, April 1996, pp. 1754-1759.
- [137] Asama, H., T. Fujii, H. Kaetsu, and I. Endo. "Elemental Technologies for Collective Robots." *Journal of Robotics and Mechatronics*, Vol. 8, No. 5, pp. 420-426, 1996.
- [138] Arai, Y., T. Fujii, H. Asama, Y. Kataoka, H. Kaetsu, A. Matsumoto, and I. Endo. "Adaptive Behavior Acquisition of Collision Avoidance among Multiple Autonomous Mobile Robots." in *Proc. Int. IEEE/RSJ Conf. on Intelligent Robots and Systems*, Vol. 3, Grenoble, France, September 07-11, 1997, pp. 1762-1767.
- [139] Ferber, J. *Les Systèmes Multi-Agents, Vers une Intelligence Collective*. Inter Editions, Paris, 1995.

ANNEXES

A. 3. Moteurs-Réducteurs

Deux (02) moteurs-réducteurs sont montés sur la roue avant de l'AGV de telle manière à ce que le premier servira pour la direction (orientation) et le second pour la traction.

A. 4. Codeurs

Deux (02) codeurs optiques incrémentaux sont montés l'un sur l'axe du moteur-réducteur de la direction et l'autre sur l'axe du moteur-réducteur de la traction.

A. 5. Capteurs à UltraSons (US)

Le système de perception de l'AGV, pour la détection d'obstacles, se base essentiellement sur des capteurs à US de type transducteurs (émetteurs-récepteurs) ultrasoniques Polaroid au nombre de huit (08): chaque transducteur a un diamètre d'environ 0.0381 m permettant un angle maximal d'émission-réception d'environ 15° (c.-à-d., couvrant un angle de -15° à $+15^\circ \Rightarrow 30^\circ$) à 3 dB de largeur 50 kHz [110]. L'étendue de chaque transducteur est d'environ 0.15 m à 10.50 m. Notons qu'il existe plusieurs types de transducteurs de différents angles d'émission-réception. Par conséquent, la définition du nombre de transducteurs nécessaires pour la couverture d'une région donnée se fait en fonction du choix du type de transducteurs.

Les positions des capteurs à US sur l'AGV sont présentées dans la Figure A. 2.

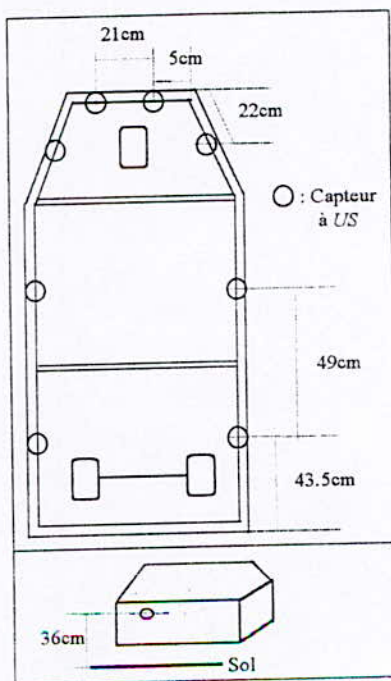


Figure A. 2. Positions des Capteurs à US sur l'AGV.

A. 6. Capteurs à Infrarouges

Il est également prévu un système de perception à infrarouges pour l'AGV avec probablement douze (12) capteurs à infrarouges. Ce système devra être utilisé pour la détection d'obstacles à moins de 25 cm, couvrant ainsi les zones dites 'mortes' que les capteurs à US ne peuvent couvrir fiablement.

A. 7. Batteries

L'alimentation de l'AGV est constituée de deux (02) batteries identiques embarquées, 12 volts, 60 Ah.

ANNEXE B: PROPRIETES DES RN ET DE LA LF

B. 1. Propriétés des RN et de la LF

Tableau B. 1. Propriétés des RN et de la LF.

Classification / Prise de Décisions

RN: Ils peuvent réaliser une classification complexe basée sur la capacité élémentaire de chaque neurone par sa fonction d'activation afin de distinguer les classes.

LF: La classification est une propriété intrinsèque puisque la notion de classes naturelles est essentiellement floue. En général, la classification est un problème de décision, c.-à-d., une division d'un espace d'attributs en régions de décisions.

Adaptation

RN: L'adaptation est en rapport direct avec la capacité d'apprentissage puisque le réseau est capable de prendre en considération et de répondre aux nouvelles contraintes et aux nouvelles données relatives à l'environnement extérieur. Comme ceux de l'être humain, un *RN* compte sur des exemples résolus antérieurement pour construire un système de neurones capables de prendre de nouvelles décisions, de classifier et d'établir des prévisions.

LF: L'auto-adaptation et la capacité de traiter les données imprécises sont dérivées à partir des notions fondamentales de la *LF*.

Généralisation

RN: A partir d'un ensemble d'exemples, l'apprentissage permet aux systèmes d'extraire des règles fondamentales ou d'émuler des comportements.

LF: Les systèmes flous sont essentiellement basés sur un ensemble de règles de généralisation et de règles de classification.

Optimisation

RN: Les *RN* sont utiles pour résoudre les problèmes d'optimisation comme prouvé par les applications dans divers domaines.

LF: L'intérêt des ensembles flous est essentiel puisqu'ils offrent un outil de représentation commun aux critères et aux contraintes flexibles.

Parallélisme / Développement

RN: Les architectures des *RN* impliquent un traitement massivement parallèle: les unités interconnectées traitent les données en même temps. La "mémoire" d'un fait correspond à une carte d'activation des neurones (codage et fait mémorisé). Cette distribution de la mémoire sur plusieurs entités mène à la tolérance aux défaillances (pannes, bruits, ou erreurs). La simulation d'un réseau, pour une application simple, implique un temps de traitement court ; alors que pour des applications complexes, l'utilisation de simulateurs et de cartes accélératrices est nécessaire.

Avantages: Apprentissage à partir d'exemples, parallélisme massif, tolérance aux défaillances (pannes,

bruits, ou erreurs) et les propriétés d'interpolation et d'approximation.

Inconvénients: Lenteur de l'apprentissage (initialisation aléatoire) et difficulté dans l'interprétation des poids.

LF: Le parallélisme est à la base des architectures des systèmes flous. En général, un système à mémoire floue effectue le codage et le traitement en parallèle, règles à mémoires associatives floues. Chaque entrée d'une mémoire associative floue active chaque règle à mémoire associative floue stockée à des niveaux différents. La fuzzification, la défuzzification et les mécanismes d'inférence sont faciles à utiliser et nécessitent un temps de traitement court, contrairement à l'apprentissage et la généralisation basés sur des règles floues.

Avantages: Utilisation de règles linguistiques, interpolation entre règles et intégration de la connaissance.

Inconvénients: Difficulté relative à la détermination des règles et à l'optimisation des fonctions d'appartenance.

B. 2. Propriétés Complémentaires des RN et de la LF

Tableau B. 2. Propriétés Complémentaires des RN et de la LF.

Connaissance

RN: La connaissance est exprimée implicitement dans des poids, après apprentissage.

LF: La connaissance est exprimée explicitement par des règles.

Classification / Généralisation

RN: Utiles si les critères de classification / règles de généralisation sont inconnus puisque les RN sont capables d'apprendre et de généraliser à partir d'exemples sans connaissance des règles. Ils peuvent être utilisés comme pré-processeur pour traiter l'information au bas niveau ou comme composant interne pour la classification / apprentissage ou la généralisation.

LF: Utile si les critères de classification / règles de généralisation sont connus.

Fonctions d'Appartenance

Les fonctions d'appartenance connues peuvent être affinées par les RN à partir d'informations sur des cas connus en entrée-sortie du réseau.

Le flou intervient dans les RN à différents niveaux: dans les données d'entrée (degré d'appartenance, fonctions d'appartenance floues), dans la propagation de l'information dans le réseau et dans l'interprétation des résultats en sortie.

ANNEXE C: APPRENTISSAGES SUPERVISES

C. 1. Rétro-Propagation du Gradient (RPG)

L'apprentissage supervisé par la RPG [62, 64, 67, 69, 71, 72, 74], présenté ci-après, est proposé pour entraîner différents RN ou RNF multicouches pour la classification de types (formes).

C. 1. 1. Topologie des RN et Convergence par la RPG

Généralement, pour une simple classification trois (03) couches sont suffisantes: une (01) couche d'entrée, une (01) couche cachée, et une (01) couche de sortie. C'est le cas des SI suivants dont l'architecture est présentée dans la Figure C. 1:

- SI_1: CIB_NAV_1_Simplifiée-RN/RPG,
- SI_1: OBS_NAV_1_Simplifiée-RN/RPG,
- SHI_1: OBS_NAV_1_Simplifiée-RN/AG,
- SHI_2: OBS_NAV_1_Simplifiée-RN/AG-RPG,
- SI_1: CIB_NAV_1_Généralisée-RN/RPG,
- SI_1: OBS_NAV_1_Généralisée-RN/RPG.

Couche d'Entrée: ayant i neurones d'entrée recevant les composantes du vecteur d'entrée X . Cette couche transmet ces entrées à tous les neurones de la couche suivante.

Couche Cachée: ayant k neurones cachés. La sortie de chaque neurone est obtenue en utilisant la fonction sigmoïde de sortie f comme suit:

$$\text{net}_k = \sum_i X_i W_{ki} \quad (\text{C. 1})$$

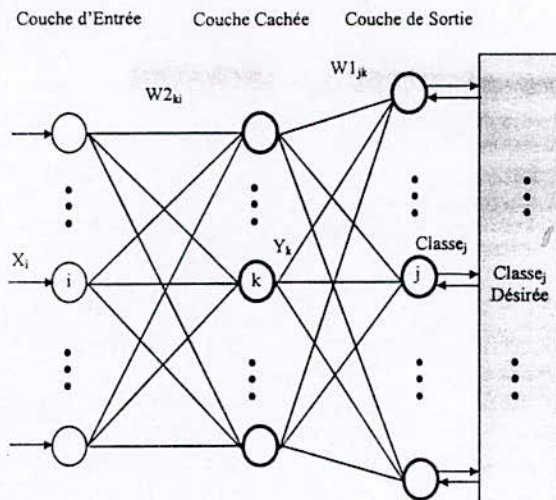
$$Y_k = f(\text{net}_k), \quad (\text{C. 2})$$

$$\text{où } f(x) = \frac{1}{1 + \exp(-x)} \quad (\text{C. 3})$$

Couche de Sortie: ayant j neurones de sortie qui sont obtenus par:

$$\text{net}_j = \sum_k Y_k W_{jk} \quad (\text{C. 4})$$

$$\text{Classe}_j = f(\text{net}_j). \quad (\text{C. 5})$$



SI_1: CIB_NAV_1_Simplifiée-RN/RPG

avec $X_i = X_{Ci}$ ($i = 1, \dots, 3$), Y_k ($k = 1, \dots, 5$), $Classe_j = C_{j1}$ ($j = j1 = 1, \dots, 6$).

SI_1: OBS_NAV_1_Simplifiée-RN/RPG

avec $X_i = X_{Oi}$ ($i = 1, \dots, 3$), Y_k ($k = 1, \dots, 5$), $Classe_j = O_{j2}$ ($j = j2 = 1, \dots, 6$).

SHI_1: OBS_NAV_1_Simplifiée-RN/AG

avec $X_i = X_{Oi}$ ($i = 1, \dots, 3$), Y_k ($k = 1, \dots, 5$), $Classe_j = O_{j2}$ ($j = j2 = 1, \dots, 6$).

SHI_2: OBS_NAV_1_Simplifiée-RN/AG-RPG

avec $X_i = X_{Oi}$ ($i = 1, \dots, 3$), Y_k ($k = 1, \dots, 5$), $Classe_j = O_{j2}$ ($j = j2 = 1, \dots, 6$).

SI_1: CIB_NAV_1_Généralisée-RN/RPG

avec $X_i = X_{Ci}$ ($i = 1, \dots, 5$), Y_k ($k = 1, \dots, 5$), $Classe_j = C_{j1}$ ($j = j1 = 1, \dots, 6$).

SI_1: OBS_NAV_1_Généralisée-RN/RPG

avec $X_i = X_{Oi}$ ($i = 1, \dots, 5$), Y_k ($k = 1, \dots, 15$), $Classe_j = O_{j2}$ ($j = j2 = 1, \dots, 30$).

Figure C. 1. Architecture des *SI* et *SHI*

Basés sur les *RN/RPG*, les *RN/AG*, et les *RN/AG-RPG*.

Avant de présenter les étapes de l'apprentissage par la *RPG*, il est intéressant de discuter de quelques problèmes relatifs à sa convergence. La Figure C. 2 illustre la possibilité de converger vers un minimum *local* dans l'espace des poids. Une fois un minimum est atteint, *local* ou *global*, l'apprentissage s'arrête. Si un minimum *local* est atteint, l'erreur aux sorties du réseau pourrait être encore largement inacceptable.

La Figure C. 2 montre une section d'intersection d'une surface d'erreur typique pour les *RN* multicouches avec des fonctions d'activation non-linéaires. Le point, z_{\min} est appelé le minimum *global*. Notez, cependant qu'il y a d'autres points minimums, z_1 et z_2 . Une recherche de descente du gradient du minimum *global* aurait pu trouver accidentellement un de ces minimums *locaux* au lieu du minimum *global*.

Heureusement, ce problème ne semble pas causer plus de difficulté en pratique. Si un réseau s'arrête d'apprendre avant d'atteindre une solution acceptable, un changement du

nombre de neurones cachés ou des paramètres d'apprentissage va souvent solutionner ce problème: nous pouvons p.ex., simplement relancer avec un ensemble différent des poids initiaux.

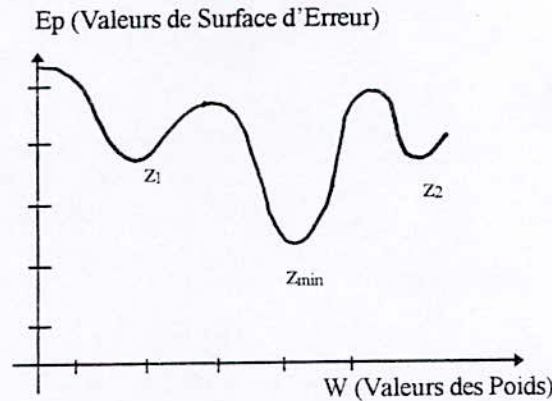


Figure C. 2. Surface d'Erreur Typique pour les RN Multicouches avec des Fonctions d'Activation Non-Linéaires.

En résumé, quand un réseau atteint une solution acceptable, il n'y a aucune garantie qu'il ait atteint un minimum *global* plutôt qu'un minimum *local*. Si la solution est jugée acceptable, il n'est pas important que le minimum soit *global* ou *local*, ou bien même que l'entraînement ait été arrêté à quelques points avant qu'un minimum *global* n'ait été atteint.

C. 1. 2. Apprentissage Supervisé par la RPG

Les étapes de l'apprentissage par la RPG sont comme suit:

- 1- Initialiser les poids aléatoirement entre $[-1, +1]$.
- 2- Appliquer un vecteur d'entrée X à la couche d'entrée.
- 3- Calculer net_k et les sorties Y_k de la couche cachée.
- 4- Calculer net_j et les sorties $Classe_j$ de la couche de sortie.
- 5- Calculer l'erreur δ_j pour les sorties de la couche de sortie:

$$\delta_j = (Classe_j \text{ Désirée} - Classe_j) f(net_j), \quad (C. 6)$$

puisque f : fonction sigmoïde $\Rightarrow f = f(1 - f)$,

$$\delta_j = (Classe_j \text{ Désirée} - Classe_j) Classe_j (1 - Classe_j). \quad (C. 7)$$

- 6- Calculer l'erreur δ_k pour les sorties de la couche cachée:

$$\delta_k = f(\text{net}_k) \sum_j \delta_j W_{1jk}, \quad (\text{C. 8})$$

$$\delta_k = Y_k (1 - Y_k) \sum_j \delta_j W_{1jk}. \quad (\text{C. 9})$$

7- Mettre à jour les poids de la couche de sortie:

$$W_{1jk}(t+1) = W_{1jk}(t) + \Delta W_{1jk}, \quad (\text{C. 10})$$

$$\text{avec } \Delta W_{1jk} = \eta \delta_j Y_k, \quad (\text{C. 11})$$

où η : taux d'apprentissage tel que $0 < \eta < 1$.

8- Mettre à jour les poids de la couche cachée:

$$W_{2ki}(t+1) = W_{2ki}(t) + \Delta W_{2ki}, \quad (\text{C. 12})$$

$$\text{avec } \Delta W_{2ki} = \eta \delta_k X_i. \quad (\text{C. 13})$$

9- Calculer l'erreur e :

$$e = (1/2) \sum_j (\text{Classe}_j \text{ Désiré} - \text{Classe}_j)^2. \quad (\text{C. 14})$$

10- Répéter 2- à 9- avec le même vecteur d'entrée \mathbf{X} (le même exemple d'apprentissage) jusqu'à ce que l'erreur e soit très proche de l'erreur admissible.

11- Répéter 2- à 10- pour chaque vecteur d'entrée \mathbf{X} (chaque exemple d'apprentissage).

12- Répéter 2- à 11- plusieurs passages complets par tous les exemples d'entraînement: cycles (*RPG*).

Les poids des connexions sont ainsi mis à jour jusqu'à ce que le réseau converge: un état permettant le codage, c.-à-d., la classification de tous les exemples d'entraînement (espace d'entrée). Cet état est atteint quand l'erreur e est très proche de l'erreur admissible c.-à-d., l'erreur pour tous les exemples d'entraînement est réduite à une valeur acceptable [64]. En pratique, l'erreur admissible est déterminée afin d'arrêter l'entraînement tout en empêchant un sur-entraînement (mémorisation non appropriée).

C. 2. Algorithmes Génétiques (AG) ou AG-RPG

Les apprentissages supervisés par les AG et par AG-RPG sont proposés pour entraîner différents RN multicouches pour la classification de types (formes). Le principal avantage de l'entraînement des RN par les AG réside dans le fait qu'ils peuvent trouver les minimums globaux sans être 'piégés' dans des minimums locaux. Ces apprentissages sont également proposés afin d'obtenir une meilleure configuration des poids des RN classificateurs.

C. 2. 1. Apprentissage Supervisé par les AG

Cet apprentissage consiste à appliquer les AG afin d'obtenir une meilleure configuration des poids des RN classificateurs [103, 114, 115]. Le synoptique de l'apprentissage supervisé par les AG est présenté dans la Figure C. 3.

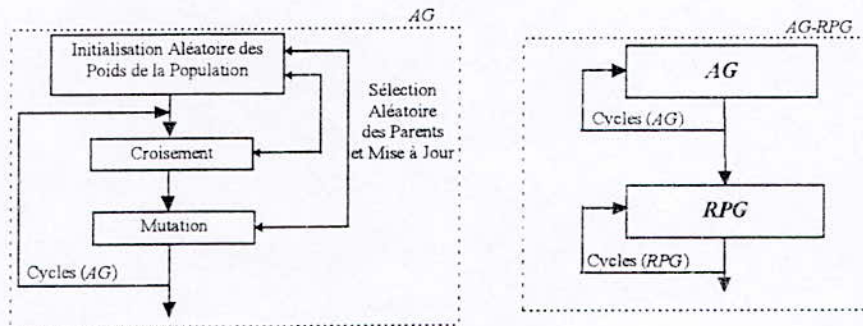


Figure C. 3. Synoptique de l'Apprentissage Supervisé par les AG ou par AG-RPG.

La taille de la population qui est un paramètre essentiel est habituellement pris égal à cinquante (50) chromosomes [103]. De plus, dans ce cas, les structures génétiques représentent les valeurs des poids des connexions du RN classificateurs: *SHI_1*: *OBS_NAV_1_Simplifiée-RN/AG*, voir Figure C. 1.

Chaque chromosome (c.-à-d., les poids dans le RN) est codé par une liste de nombres réels, voir Figure C. 4, avec l'erreur:

$$E_{pop} = e_1 + \dots + e_{ex} + \dots + e_{54} \quad (C. 15)$$

avec l'indice pop qui parcourt la population de chromosomes: $pop = 1, \dots, 50$

$$\text{et } e_{ex} = (1/2) \sum_j (\text{Classe}_j \text{ Dé siré} - \text{Classe}_j)^2 \quad (C. 16)$$

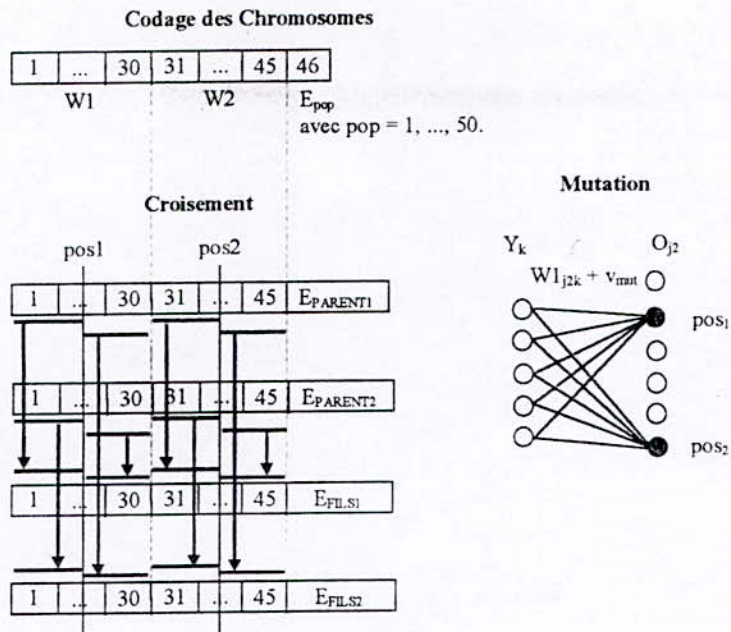


Figure C. 4. Codage des Chromosomes, Croisement, et Mutation.

Le choix des deux (02) opérateurs croisement des poids et mutation des neurones se base sur les résultats obtenus dans [103]. L'opérateur croisement prend deux (02) chromosomes parents et crée deux (02) chromosomes fils contenant quelques caractéristiques génétiques de chaque parent. L'opérateur mutation prend un (01) chromosome parent et change aléatoirement quelques entrées afin de créer un (01) chromosome fils.

Les étapes de l'apprentissage par les *AG* sont comme suit:

- 1- Initialisation aléatoire des poids de la population.
- 2- Croisement des Poids: cet opérateur sélectionne deux (02) parents et crée deux (02) fils en procédant comme suit:

+ sélection aléatoire de deux (02) chromosomes parents à partir de la population c.-à-d., parmi [1, ..., 50]: deux (02) chromosomes parents $chromosome_{PARENT1}$ et $chromosome_{PARENT2}$ et création de deux (02) chromosomes fils $chromosome_{FILS1}$ et $chromosome_{FILS2}$ présentés dans la Figure C. 4. Les deux (02) positions du croisement pos_1 et pos_2 sont déterminées aléatoirement parmi [1, ..., 30] et [31, ..., 45], respectivement.

+ ensuite, les deux (02) chromosomes ayant les erreurs les plus élevées de la population doivent être mis à jour par les deux (02) chromosomes ayant les erreurs les moins élevées parmi $chromosome_{PARENT1}$, $chromosome_{PARENT2}$, $chromosome_{FILS1}$, et $chromosome_{FILS2}$.

- 3- Mutation des Neurones: cet opérateur sélectionne un nombre nbr de neurones, n'appartenant pas aux neurones d'entrée, du réseau représenté par le chromosome parent.

Pour chaque une des connexions allant vers ces nbr neurones, l'opérateur ajoute une valeur aléatoire v_{mut} aux poids de cette connexion à partir d'une distribution aléatoire. Il code ensuite ce nouveau réseau sur le chromosome fils. Le but étant que les connexions allant vers un neurone forment un sous-groupe de toutes les connexions en termes d'opération du réseau. En limitant ses changements à un petit nombre de ces groupes, cet opérateur va probablement améliorer le chromosome fils. Dans notre cas $nbr = 2$. Cet opérateur sélectionne un (01) chromosome parent et crée un (01) chromosome fils en procédant comme suit:

- + sélection aléatoire d'un (01) chromosome parent à partir de la population c.-à-d., parmi $[1, \dots, 50]$: un (01) chromosome $chromosome_{PARENT}$ et création d'un (01) chromosome fils $chromosome_{FILS}$. Les deux (02) positions de la mutation pos_1 et pos_2 sont déterminées aléatoirement parmi les neurones n'appartenant pas aux neurones d'entrée c.-à-d., soit pos_1 et pos_2 parmi $[Y_1, \dots, Y_5]$ soit pos_1 et pos_2 parmi $[O_1, \dots, O_6]$ avec $pos_1 \neq pos_2$, voir Figure C. 1. Cette opération de mutation est expliquée à travers un exemple dans la Figure C. 4 c.-à-d., les neurones noirs représentent pos_1 et pos_2 , ensuite toutes les connexions allant à ces neurones doivent ajouter à leurs poids une valeur aléatoire v_{mut} .

- + ensuite, le chromosome parent de la population doit être mis à jour par le chromosome fils si l'erreur E_{FILS} est plus petite que E_{PARENT} .

4- Répéter 2- à 3- plusieurs fois: cycles (AG).

C. 2. 2. Apprentissage Supervisé par AG-RPG

Cet apprentissage consiste à appliquer les AG suivis de la RPG (AG-RPG) afin d'obtenir une meilleure configuration des poids des RN classificateurs [71, 114]. Le synoptique de l'apprentissage supervisé par AG-RPG est présenté dans la Figure C. 3. Ainsi, l'apprentissage du *SHI_2: OBS_NAV_1 Simplifiée-RN/AG-RPG*, voir Figure C. 1, commence par les AG ensuite continue par la RPG.

C. 3. "ArtMap" Floue Simplifiée (AMFS)

C'est un apprentissage supervisé, rapide, et stable spécialisé dans la classification de types (formes) et qui peut apprendre chaque type d'entraînement en seulement un (01) seul passage par tous les exemples d'entraînement: cycle (AMFS). Il démarre sans poids de connexion mais augmente en taille pour s'adapter au problème et possède seulement un

paramètre à sélectionner par l'utilisateur. Il a été prouvé que cet apprentissage incrémental est stable puisque tous les poids adaptatifs ne peuvent que diminuer dans le temps et qu'il est souvent plus performant que les autres techniques sur une variété de machines d'apprentissage de référence.

C. 3. 1. Réseaux de Neurones Basés sur la "ArtMap" Floue (RNAMF)

Les RNAMF décident si un (01) ou plusieurs neurones de sortie sont nécessaires afin de représenter une (01) catégorie particulière. En effet, ce type de réseau grandit par lui-même pour représenter un problème au lieu d'être fixé donc limité par un concepteur de réseau en une architecture *statique*. L'architecture d'un tel réseau est présentée dans la Figure C. 5 alors que les concepts mathématiques de l'apprentissage supervisé, basé sur la "ArtMap" Floue [74, 104, 105], sur lequel se base cette architecture sont donnés ci-après.

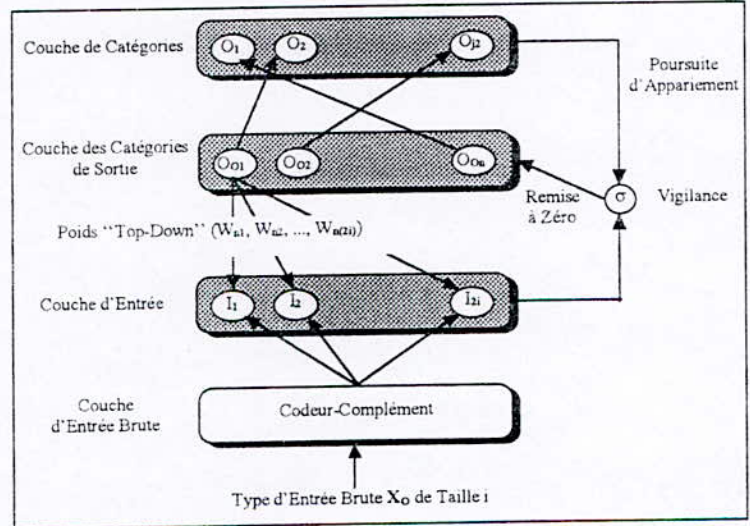


Figure C. 5. Architecture Générale des RNAMF.

C. 3. 2. Apprentissage Supervisé par la AMFS

Normalisation d'Entrées

Le codeur-complément est un processus de normalisation des entrées qui représente la présence et l'absence d'une caractéristique particulière dans le type d'entrée. Si X est le vecteur d'un type d'entrée donné de i caractéristiques, le vecteur complément \bar{X} représente l'absence de chaque caractéristique où \bar{X} est défini par:

$$\bar{X}_i = 1 - X_i \quad (\text{C. 17})$$

Comme dans la *LF*, toutes les valeurs d'entrée des *RNAMF* doivent être comprises entre $[0, 1]$. Par conséquent, le vecteur d'entrée complété **I** est donné par le vecteur à $2i$ -dimensions:

$$\mathbf{I} = (\mathbf{X}, \bar{\mathbf{X}}) = (X_1, \dots, X_i, \bar{X}_1, \dots, \bar{X}_i) \quad (\text{C. 18})$$

Une seconde caractéristique intéressante du codeur-complément est la normalisation automatique des vecteurs d'entrée telle que:

$$|\mathbf{I}| = |(\mathbf{X}, \bar{\mathbf{X}})| = \sum_i X_i + (i - \sum_i X_i) = i \quad (\text{C. 19})$$

où la norme $||$ est définie par:

$$|\mathbf{X}| = \sum_i X_i \quad (\text{C. 20})$$

Ainsi, la norme de n'importe quel vecteur d'entrée complété à i -dimensions est égale à i .

Activation du Neurone de Sortie

Le réseau nécessite un mécanisme pour former une activation au niveau de la couche de sortie en réponse à une entrée au réseau. Cette activation de sortie est indiquée par F_n pour le n ème neurone de sortie ainsi que ses poids \mathbf{W}_n . La fonction produisant cette activation est:

$$F_n(\mathbf{I}) = \frac{|\mathbf{I} \wedge \mathbf{W}_n|}{\lambda + |\mathbf{W}_n|} \quad (\text{C. 21})$$

où \wedge est l'opérateur ET Flou et λ est une constante proche de 0, habituellement égale à 0.000001.

Le neurone de sortie gagnant est le neurone possédant la plus grande activation:

$$\text{Gagnant} = \text{MAX}(F_n) \quad (\text{C. 22})$$

La catégorie associée au neurone de sortie gagnant est ainsi la classification du réseau du type d'entrée courant.

La fonction d'appariement est utilisée pour comparer les caractéristiques de l'entrée complétement et des poids d'un (01) neurone de sortie particulier afin d'aider à déterminer si l'apprentissage aurait lieu. La fonction d'appariement est définie par:

$$\frac{|I \wedge W_n|}{|I|} \quad (C. 23)$$

Dans la *LF*, la fonction d'appariement calcule le degré auquel l'entrée complétement *I* est un sous-ensemble flou de W_n . En utilisant l'Equation (C. 19), l'équation de la fonction d'appariement est réduite à:

$$\frac{|I \wedge W_n|}{i} \quad (C. 24)$$

Lorsqu'elle est utilisée concurremment avec le paramètre vigilance σ , la valeur de la fonction d'appariement détermine si l'entrée courante est un appariement assez bon pour un neurone de sortie particulier pour être codée par ce neurone de sortie ou par contre si un nouveau neurone de sortie devrait être créé pour coder ce type d'entrée. Le paramètre vigilance est compris entre $[0, 1]$; sa valeur initiale appelée vigilance de base σ est le seul paramètre à sélectionner par l'utilisateur. Il contrôle le degré du codage du neurone de sortie: de grandes valeurs du paramètre vigilance résulte des neurones de sortie plus sélectifs quant à la décision sur le codage des types d'entrée, alors que de petites valeurs de ce paramètre résulte des critères d'appariement moins sélectifs pour les neurones de sortie. En général, les grandes valeurs du paramètre vigilance impliquent la création d'un grand nombre de neurones des catégories de sortie. La poursuite d'appariement permet au réseau, voir Figure C. 5, d'auto-ajuster son paramètre vigilance durant l'entraînement, à partir de la vigilance de base σ , en réponse aux erreurs de classification. C'est à travers la poursuite d'appariement que le réseau ajuste son propre paramètre d'apprentissage lui permettant de décider quand faut-il créer des nouveaux neurones de sortie (reformer ses régions de décisions).

Si la valeur de la fonction d'appariement est plus grande que le paramètre de vigilance, le réseau est dit être en état de résonance:

$$\frac{|I \wedge W_n|}{i} > \sigma \quad (C. 25)$$

La résonance signifie que le neurone de sortie *n* est assez bon pour coder l'entrée *I*, stipulant que le neurone de sortie *n* représente la même catégorie que celle de l'entrée *I*. Ainsi,

si le neurone de sortie n code déjà une certaine catégorie j_2 et si l'entrée \mathbf{I} est de la même catégorie j_2 , le neurone de sortie n pourra alors mettre à jour ses poids pour apprendre le nouveau type d'entrée. Si le neurone de sortie code une catégorie différente de celle du type d'entrée d'entraînement, une erreur d'appariement de catégorie survient alors et les poids du neurone de sortie restent inchangés. A chaque présentation d'un type d'entrée d'entraînement, seulement un neurone de sortie peut mettre à jour ses poids.

Un état du réseau appelé remise à zéro de l'erreur d'appariement survient si la valeur de la fonction d'appariement est plus petite que celle du paramètre vigilance. Cet état indique que le neurone de sortie courant n'as pas atteint le degré du codage représenté par le paramètre vigilance et par conséquent ne peut mettre à jour ses poids même si la catégorie du type d'entrée est égale à la catégorie du neurone de sortie gagnant.

Si la valeur de la fonction d'appariement n'est pas supérieure ou égale au paramètre vigilance courant σ et si la catégorie de l'entrée courante est la même que celle du neurone courant, une erreur d'appariement de catégorie survient alors, ce qui implique que le paramètre vigilance est automatiquement augmenté d'une constante ε proche de 0, $\sigma = \sigma + \varepsilon$, habituellement égale à 0.0001.

Une fois un neurone de sortie gagnant n a été sélectionné afin d'apprendre un type d'entrée particulier \mathbf{I} , le vecteur de poids "Top-Down" \mathbf{W}_n du neurone de sortie est mis à jour selon l'équation:

$$\mathbf{W}_n(t+1) = \eta (\mathbf{I} \wedge \mathbf{W}_n(t)) + (1 - \eta) \mathbf{W}_n(t) \quad (\text{C. 26})$$

où plus η , taux d'apprentissage compris entre $0 < \eta \leq 1$, est grand et plus l'apprentissage est rapide.

Afin de simplifier l'équation de mise à jour des poids, le taux d'apprentissage η est pris égal à 1 pour un apprentissage rapide. Ainsi, la mise à jour des poids est réduite à:

$$\mathbf{W}_n(t+1) = (\mathbf{I} \wedge \mathbf{W}_n(t)) \quad (\text{C. 27})$$

Ces concepts mathématiques sont utilisés dans l'apprentissage supervisé, par la "ArtMap" Floue Simplifiée (AMFS), détaillé dans [121]. Notons le fait que cet apprentissage peut créer un (01) ou plusieurs neurones de sortie afin de représenter une même catégorie:

dans le cas d'une erreur d'appariement à un neurone codant la même catégorie que celle de l'entrée courante. Ceci implique que les *RNAMF* entraînés par la *AMFS* sont des réseaux à architecture *dynamique*.

Ainsi, pour l'apprentissage des six (06) classes d'évitement d'obstacles voir Figure I. 7, la configuration initiale du *SHI_4: OBS_NAV_1_Simplifiée-RNAMF/AMFS* est présentée dans la Figure C. 6, sans aucun neurone de sortie ni poids.

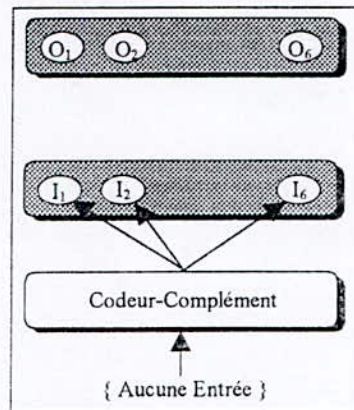


Figure C. 6. Configuration Initiale du *SHI_4: OBS_NAV_1_Simplifiée-RNAMF/AMFS*.

Ce classificateur, présenté après apprentissage dans la Figure II. 4, doit reconnaître la situation d'évitement d'obstacles O_{j2} exprimée par la sortie ayant la plus grande activation F_n .

C. 4. Renforcement par *Essai et Erreur (REE)*

L'apprentissage supervisé par le *REE* est un apprentissage spécialisé dans le renforcement des situations aux actions favorables ainsi que la dissociation de situations conduisant à des actions défavorables en procédant par la technique dite par *essai et erreur*.

C. 4. 1. Association Entre Situations et Actions

L'association entre une situation donnée et une action est habituellement mise en pratique par l'utilisation d'un signal fourni par un processus externe (p.ex., un superviseur), donnant la réponse désirée. Afin de réaliser une association correcte, la réponse désirée est acquise à travers l'apprentissage supervisé par *REE*. L'apprentissage, dans ce cas, est guidé seulement par un processus de retour, c.-à-d., guidé par un signal P fourni par le superviseur.

Ce signal conduit au renforcement de l'association entre une situation donnée et une action favorable si cette dernière mène à une conséquence favorable ; sinon, le signal P provoque une dissociation [30].

C. 4. 2. Mise à Jour des Poids Entre Situations et Actions

Pour cet apprentissage, l'équation suivante est utilisée pour mettre à jour les poids M_{ij} de l'association d'une situation (classe) à une action comme détaillé dans [30]:

$$\tau \frac{dM_{ij}(t)}{dt} = A_i \text{ Classe}_j (- M_{ij}(t) + \alpha - P), \quad (\text{C. 28})$$

avec τ constante de temps et α constante.

L'Equation (C. 28) conduit à l'équation linéaire, à coefficients constants, de premier ordre suivante:

$$\frac{dM_{ij}(t)}{dt} + \frac{A_i \text{ Classe}_j}{\tau} M_{ij}(t) = \frac{A_i \text{ Classe}_j}{\tau} (\alpha - P). \quad (\text{C. 29})$$

La solution générale de l'Equation (C. 29) est donnée par:

$$M_{ij}(t) = c e^{-\frac{A_i \text{ Classe}_j}{\tau} t} + (\alpha - P), \quad (\text{C. 30})$$

où c est une constante déterminée à $t = 0$: $M_{ij}(0) = 0$ et $P = 0 \Rightarrow c = -\alpha$.

Ainsi, l'équation finale pour mettre à jour les poids M_{ij} se réduit à:

$$M_{ij}(t) = -\alpha e^{-\frac{A_i \text{ Classe}_j}{\tau} t} + (\alpha - P), \quad (\text{C. 31})$$

Notons, dans notre cas les poids de l'association des situations de position de la cible aux actions, U_{ij1} , ainsi que ceux de l'association des situations d'évitement d'obstacles aux actions, V_{ij2} , sont mis à jour par l'Equation (C. 31) avec $M_{ij} = U_{ij1}$ et $\text{Classe}_j = C_{j1}$ ainsi que $M_{ij} = V_{ij2}$ et $\text{Classe}_j = O_{j2}$, respectivement.

ANNEXE D: BASES DE REGLES FLOUES

D. 1. Base de Règles Floues 1

Tableau D. 1. Base de Règles Floues 1.

- Si* (X_{O1} est P et X_{O2} est P et X_{O3} est M) *Alors* $\tilde{O} = [0, 0, 0.6, 0, 0, 0]$,
- Si* (X_{O1} est P et X_{O2} est P et X_{O3} est L) *Alors* $\tilde{O} = [0, 0, 1, 0, 0, 0]$,
- Si* (X_{O1} est P et X_{O2} est M et X_{O3} est P) *Alors* $\tilde{O} = [1, 0, 0, 0, 0, 0]$,
- Si* (X_{O1} est P et X_{O2} est M et X_{O3} est M) *Alors* $\tilde{O} = [0.2, 0, 0.2, 0, 0.9, 0]$,
- Si* (X_{O1} est P et X_{O2} est M et X_{O3} est L) *Alors* $\tilde{O} = [0.2, 0, 0, 0, 0.9, 0]$,
- Si* (X_{O1} est P et X_{O2} est L et X_{O3} est P) *Alors* $\tilde{O} = [1, 0, 0, 0, 0, 0]$,
- Si* (X_{O1} est P et X_{O2} est L et X_{O3} est M) *Alors* $\tilde{O} = [0.2, 0, 0, 0, 0.9, 0]$,
- Si* (X_{O1} est P et X_{O2} est L et X_{O3} est L) *Alors* $\tilde{O} = [0, 0, 0, 0, 1, 0]$,
- Si* (X_{O1} est M et X_{O2} est P et X_{O3} est P) *Alors* $\tilde{O} = [0, 0.6, 0, 0, 0, 0]$,
- Si* (X_{O1} est M et X_{O2} est P et X_{O3} est M) *Alors* $\tilde{O} = [0, 0.2, 0.2, 0.9, 0, 0]$,
- Si* (X_{O1} est M et X_{O2} est P et X_{O3} est L) *Alors* $\tilde{O} = [0, 0, 0.9, 0.2, 0, 0]$,
- Si* (X_{O1} est M et X_{O2} est M et X_{O3} est P) *Alors* $\tilde{O} = [0.2, 0.2, 0, 0, 0, 0.9]$,
- Si* (X_{O1} est M et X_{O2} est M et X_{O3} est M) *Alors* $\tilde{O} = [0.2, 0.2, 0.2, 0.9, 0.9, 0.9]$,
- Si* (X_{O1} est M et X_{O2} est M et X_{O3} est L) *Alors* $\tilde{O} = [0, 0, 0.2, 0.9, 0.9, 0]$,
- Si* (X_{O1} est M et X_{O2} est L et X_{O3} est P) *Alors* $\tilde{O} = [0.2, 0, 0, 0, 0, 0.9]$,
- Si* (X_{O1} est M et X_{O2} est L et X_{O3} est M) *Alors* $\tilde{O} = [0.2, 0, 0, 0, 0.9, 0.9]$,
- Si* (X_{O1} est M et X_{O2} est L et X_{O3} est L) *Alors* $\tilde{O} = [0, 0, 0, 0, 0.6, 0]$,
- Si* (X_{O1} est L et X_{O2} est P et X_{O3} est P) *Alors* $\tilde{O} = [0, 1, 0, 0, 0, 0]$,
- Si* (X_{O1} est L et X_{O2} est P et X_{O3} est M) *Alors* $\tilde{O} = [0, 0.2, 0, 0.9, 0, 0]$,
- Si* (X_{O1} est L et X_{O2} est P et X_{O3} est L) *Alors* $\tilde{O} = [0, 0, 0, 1, 0, 0]$,
- Si* (X_{O1} est L et X_{O2} est M et X_{O3} est P) *Alors* $\tilde{O} = [0.2, 0, 0, 0, 0, 0.9]$,
- Si* (X_{O1} est L et X_{O2} est M et X_{O3} est M) *Alors* $\tilde{O} = [0, 0.2, 0, 0.9, 0, 0.9]$,
- Si* (X_{O1} est L et X_{O2} est M et X_{O3} est L) *Alors* $\tilde{O} = [0, 0, 0, 0.6, 0, 0]$,
- Si* (X_{O1} est L et X_{O2} est L et X_{O3} est P) *Alors* $\tilde{O} = [0, 0, 0, 0, 0, 1]$,
- Si* (X_{O1} est L et X_{O2} est L et X_{O3} est M) *Alors* $\tilde{O} = [0, 0, 0, 0, 0, 0.6]$.
- avec $X_{O1} = d_D$, $X_{O2} = d_A$, et $X_{O3} = d_G$.

D. 2. Base de Règles Floues 2

Tableau D. 2. Base de Règles Floues 2.

- Si* (X_{01} est P et X_{02} est P et X_{03} est M) *Alors* O_3 ,
Si (X_{01} est P et X_{02} est P et X_{03} est L) *Alors* O_3 ,
Si (X_{01} est P et X_{02} est M et X_{03} est P) *Alors* O_1 ,
Si (X_{01} est P et X_{02} est M et X_{03} est M) *Alors* O_5 ,
Si (X_{01} est P et X_{02} est M et X_{03} est L) *Alors* O_5 ,
Si (X_{01} est P et X_{02} est L et X_{03} est P) *Alors* O_1 ,
Si (X_{01} est P et X_{02} est L et X_{03} est M) *Alors* O_5 ,
Si (X_{01} est P et X_{02} est L et X_{03} est L) *Alors* O_5 ,
Si (X_{01} est M et X_{02} est P et X_{03} est P) *Alors* O_2 ,
Si (X_{01} est M et X_{02} est P et X_{03} est M) *Alors* O_4 ,
Si (X_{01} est M et X_{02} est P et X_{03} est L) *Alors* O_3 ,
Si (X_{01} est M et X_{02} est M et X_{03} est P) *Alors* O_6 ,
Si (X_{01} est M et X_{02} est M et X_{03} est M) *Alors* O_6 ,
Si (X_{01} est M et X_{02} est M et X_{03} est L) *Alors* O_3 ,
Si (X_{01} est M et X_{02} est L et X_{03} est P) *Alors* O_1 ,
Si (X_{01} est M et X_{02} est L et X_{03} est M) *Alors* O_5 ,
Si (X_{01} est M et X_{02} est L et X_{03} est L) *Alors* O_5 ,
Si (X_{01} est L et X_{02} est P et X_{03} est P) *Alors* O_2 ,
Si (X_{01} est L et X_{02} est P et X_{03} est M) *Alors* O_2 ,
Si (X_{01} est L et X_{02} est P et X_{03} est L) *Alors* O_4 ,
Si (X_{01} est L et X_{02} est M et X_{03} est P) *Alors* O_1 ,
Si (X_{01} est L et X_{02} est M et X_{03} est M) *Alors* O_6 ,
Si (X_{01} est L et X_{02} est M et X_{03} est L) *Alors* O_4 ,
Si (X_{01} est L et X_{02} est L et X_{03} est P) *Alors* O_6 ,
Si (X_{01} est L et X_{02} est L et X_{03} est M) *Alors* O_6 ,

avec $X_{01} = d_D$, $X_{02} = d_A$, et $X_{03} = d_G$.

D. 3. Base de Règles Floues 3

Tableau D. 3. Base de Règles Floues 3.

- Si* (X_{O1} est GG et X_{O2} est P) *Alors* (α est DGP),
Si (X_{O1} est GG et X_{O2} est M) *Alors* (α est SGP),
Si (X_{O1} est GG et X_{O2} est L) *Alors* (α est SGG),
Si (X_{O1} est GP et X_{O2} est P) *Alors* (α est DGG),
Si (X_{O1} est GP et X_{O2} est M) *Alors* (α est DGP),
Si (X_{O1} est GP et X_{O2} est L) *Alors* (α est SGP),
Si (X_{O1} est DP et X_{O2} est P) *Alors* (α est DDG),
Si (X_{O1} est DP et X_{O2} est M) *Alors* (α est DDP),
Si (X_{O1} est DP et X_{O2} est L) *Alors* (α est SDP),
Si (X_{O1} est DG et X_{O2} est P) *Alors* (α est DDP),
Si (X_{O1} est DG et X_{O2} est M) *Alors* (α est SDP),
Si (X_{O1} est DG et X_{O2} est L) *Alors* (α est SDG),

avec $X_{O1} = \theta_S$ et $X_{O2} = L_S$.

D. 4. Base de Règles Floues 4

Tableau D. 4. Base de Règles Floues 4.

- Si* (X_{O3} est GG et X_{O4} est L) *Alors* (β est DGP),
Si (X_{O3} est GG et X_{O4} est R) *Alors* (β est SGP),
Si (X_{O3} est GM et X_{O4} est L) *Alors* (β est SGG),
Si (X_{O3} est GM et X_{O4} est R) *Alors* (β est DGG),
Si (X_{O3} est GP et X_{O4} est L) *Alors* (β est DGP),
Si (X_{O3} est GP et X_{O4} est R) *Alors* (β est SGP),
Si (X_{O3} est DP et X_{O4} est L) *Alors* (β est DDG),
Si (X_{O3} est DP et X_{O4} est R) *Alors* (β est DDP),
Si (X_{O3} est DM et X_{O4} est L) *Alors* (β est SDP),
Si (X_{O3} est DM et X_{O4} est R) *Alors* (β est DDP),
Si (X_{O3} est DG et X_{O4} est L) *Alors* (β est SDP),
Si (X_{O3} est DG et X_{O4} est R) *Alors* (β est SDG),

avec $X_{O3} = \Phi_U$ et $X_{O4} = W_U$.

D. 5. Distances d'Accélération/Décélération pour Chaque Combinaison des Vitesse Finale, Vitesse Initiale, et Distance-But

Tableau D. 5. Distances d'Accélération/Décélération pour Chaque Combinaison des Vitesse Finale, Vitesse Initiale, et Distance-But.

Notons que ces expériences sont établies pour les vitesses finales désirées:

$$V_0 = 0 \text{ m/s}, V_1 = 0.1 \text{ m/s}, V_2 = 0.2 \text{ m/s}, V_3 = 0.3 \text{ m/s}, \text{ et } V_4 = 0.4 \text{ m/s}.$$

$V_{\text{finale}} \text{ (m/s)}$	$V_{\text{initiale}} \text{ (m/s)}$	$d_{\text{but}} \text{ (m)}$	$d_{\text{acc}} \text{ (m)}$	$d_{\text{déc}} \text{ (m)}$
0	0	0.375	0.084050	0.292050
0.1	0	0.375	0.084050	0.296050
0.2	0	0.375	0.084050	0.312050
0.3	0	0.375	0.084050	0.336050
0.4	0	0.375	0.084050	0.372050
0	0.1	0.375	0.075000	0.295000
0.1	0.1	0.375	0.075000	0.299000
0.2	0.1	0.375	0.075000	0.315000
0.3	0.1	0.375	0.075000	0.339000
0.4	0.1	0.375	0.075000	0.375000
0	0.2	0.375	0.060000	0.292000
0.1	0.2	0.375	0.060000	0.300000
0.2	0.2	0.375	0.060000	0.312000
0.3	0.2	0.375	0.060000	0.340000
0.4	0.2	0.375	0.060000	0.372000
0	0.3	0.375	0.035000	0.295000
0.1	0.3	0.375	0.035000	0.299000
0.2	0.3	0.375	0.035000	0.315000
0.3	0.3	0.375	0.035000	0.339000
0.4	0.3	0.375	0.035000	0.375000
0	0.4	0.375	0	0.292000
0.1	0.4	0.375	0	0.300000
0.2	0.4	0.375	0	0.312000
0.3	0.4	0.375	0	0.340000

0.4	0.4	0.375	0	0.372000
-----	-----	-------	---	----------

V_{finale} (m/s)	$V_{initiale}$ (m/s)	d_{but} (m)	d_{acc} (m)	d_{dec} (m)
0	0	0.750	0.084050	0.668049
0.1	0	0.750	0.084050	0.672049
0.2	0	0.750	0.084050	0.688049
0.3	0	0.750	0.084050	0.712049
0.4	0	0.750	0.084050	0.748049
0	0.1	0.750	0.075000	0.666999
0.1	0.1	0.750	0.075000	0.674999
0.2	0.1	0.750	0.075000	0.686999
0.3	0.1	0.750	0.075000	0.714999
0.4	0.1	0.750	0.075000	0.746999
0	0.2	0.750	0.060000	0.668000
0.1	0.2	0.750	0.060000	0.672000
0.2	0.2	0.750	0.060000	0.687999
0.3	0.2	0.750	0.060000	0.711999
0.4	0.2	0.750	0.060000	0.747999
0	0.3	0.750	0.035000	0.667000
0.1	0.3	0.750	0.035000	0.675000
0.2	0.3	0.750	0.035000	0.686999
0.3	0.3	0.750	0.035000	0.714999
0.4	0.3	0.750	0.035000	0.746999
0	0.4	0.750	0	0.668000
0.1	0.4	0.750	0	0.672000
0.2	0.4	0.750	0	0.687999
0.3	0.4	0.750	0	0.711999
0.4	0.4	0.750	0	0.747999

V_{finale} (m/s)	$V_{initiale}$ (m/s)	d_{but} (m)	d_{acc} (m)	d_{dec} (m)
0	0	1.125	0.084050	1.044049
0.1	0	1.125	0.084050	1.048049
0.2	0	1.125	0.084050	1.064049

0.3	0	1.125	0.084050	1.088049
0.4	0	1.125	0.084050	1.124049
0	0.1	1.125	0.075000	1.042999
0.1	0.1	1.125	0.075000	1.046999
0.2	0.1	1.125	0.075000	1.062999
0.3	0.1	1.125	0.075000	1.086999
0.4	0.1	1.125	0.075000	1.122999
0	0.2	1.125	0.060000	1.043999
0.1	0.2	1.125	0.060000	1.047999
0.2	0.2	1.125	0.060000	1.063999
0.3	0.2	1.125	0.060000	1.087999
0.4	0.2	1.125	0.060000	1.123999
0	0.3	1.125	0.035000	1.042999
0.1	0.3	1.125	0.035000	1.046999
0.2	0.3	1.125	0.035000	1.062999
0.3	0.3	1.125	0.035000	1.086999
0.4	0.3	1.125	0.035000	1.122999
0	0.4	1.125	0	1.043999
0.1	0.4	1.125	0	1.047999
0.2	0.4	1.125	0	1.063999
0.3	0.4	1.125	0	1.087999
0.4	0.4	1.125	0	1.123999

V_{finale} (m/s)	V_{initiale} (m/s)	d_{but} (m)	d_{acc} (m)	d_{dec} (m)
0	0	1.500	0.084050	1.416049
0.1	0	1.500	0.084050	1.424049
0.2	0	1.500	0.084050	1.436049
0.3	0	1.500	0.084050	1.464049
0.4	0	1.500	0.084050	1.496049
0	0.1	1.500	0.075000	1.418999
0.1	0.1	1.500	0.075000	1.422999
0.2	0.1	1.500	0.075000	1.438999

0.3	0.1	1.500	0.075000	1.462999
0.4	0.1	1.500	0.075000	1.498999
0	0.2	1.500	0.060000	1.419999
0.1	0.2	1.500	0.060000	1.423999
0.2	0.2	1.500	0.060000	1.439999
0.3	0.2	1.500	0.060000	1.463999
0.4	0.2	1.500	0.060000	1.499999
0	0.3	1.500	0.035000	1.418999
0.1	0.3	1.500	0.035000	1.422999
0.2	0.3	1.500	0.035000	1.438999
0.3	0.3	1.500	0.035000	1.462999
0.4	0.3	1.500	0.035000	1.498999
0	0.4	1.500	0	1.419999
0.1	0.4	1.500	0	1.423999
0.2	0.4	1.500	0	1.439999
0.3	0.4	1.500	0	1.463999
0.4	0.4	1.500	0	1.499999

$V_{\text{finale}} \text{ (m/s)}$	$V_{\text{initiale}} \text{ (m/s)}$	$d_{\text{but}} \text{ (m)}$	$d_{\text{acc}} \text{ (m)}$	$d_{\text{dec}} \text{ (m)}$
0	0	2.000	0.084050	1.916058
0.1	0	2.000	0.084050	1.924058
0.2	0	2.000	0.084050	1.936058
0.3	0	2.000	0.084050	1.964059
0.4	0	2.000	0.084050	1.996060
0	0.1	2.000	0.075000	1.919007
0.1	0.1	2.000	0.075000	1.923007
0.2	0.1	2.000	0.075000	1.939009
0.3	0.1	2.000	0.075000	1.963008
0.4	0.1	2.000	0.075000	1.999009
0	0.2	2.000	0.060000	1.916006
0.1	0.2	2.000	0.060000	1.924007
0.2	0.2	2.000	0.060000	1.936007

0.3	0.2	2.000	0.060000	1.964007
0.4	0.2	2.000	0.060000	1.996008
0	0.3	2.000	0.035000	1.919006
0.1	0.3	2.000	0.035000	1.923006
0.2	0.3	2.000	0.035000	1.939007
0.3	0.3	2.000	0.035000	1.963007
0.4	0.3	2.000	0.035000	1.999008
0	0.4	2.000	0	1.916006
0.1	0.4	2.000	0	1.924006
0.2	0.4	2.000	0	1.936006
0.3	0.4	2.000	0	1.964007
0.4	0.4	2.000	0	1.996008

D. 6. Base de Règles Floues 5

Tableau D. 6. Base de Règles Floues 5.

Notons que la base de règles floues est établie pour les vitesses finales désirées:

$$V_0 = 0 \text{ m/s}, V_1 = 0.1 \text{ m/s}, V_2 = 0.2 \text{ m/s}, V_3 = 0.3 \text{ m/s}, \text{ et } V_4 = 0.4 \text{ m/s}.$$

$$X_{CA1} = V_{\text{finale}},$$

$$X_{CA2} = V_{\text{initiale}}, V_{\text{courante}}$$

$$X_{CA3} = d_{\text{but}}, d_{\text{courante}}$$

$$V_{\text{finale}} = V_0$$

Si V_{courante} est TL et d_{but} est TP et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_1

et d_{but} est P et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_1

et d_{but} est L et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_1

et d_{but} est TL et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_1

et d_{but} est HE et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_1

$$V_{\text{finale}} = V_1$$

Si V_{courante} est TL et d_{but} est TP et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_6

et d_{but} est P et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_6

et d_{but} est L et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_6

et d_{but} est TL et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_6

et d_{but} est HE et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_6

$$V_{\text{finale}} = V_2$$

Si V_{courante} est TL et d_{but} est TP et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_5

et d_{but} est P et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_5

et d_{but} est L et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_5

et d_{but} est TL et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_5

et d_{but} est HE et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_5

$$V_{\text{finale}} = V_3$$

Si V_{courante} est TL et d_{but} est TP et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est MAI *Alors CA₃*
 et d_{courante} est DEC *Alors CA₁*
 et d_{but} est TL et d_{courante} est ACC *Alors CA₃*
 et d_{courante} est MAI *Alors CA₃*
 et d_{courante} est DEC *Alors CA₁*
 et d_{but} est HE et d_{courante} est ACC *Alors CA₃*
 et d_{courante} est MAI *Alors CA₃*
 et d_{courante} est DEC *Alors CA₁*

$$V_{\text{finale}} = V_1$$

Si V_{courante} est L et d_{but} est TP et d_{courante} est ACC *Alors CA₃*
 et d_{courante} est MAI *Alors CA₃*
 et d_{courante} est DEC *Alors CA₆*
 et d_{but} est P et d_{courante} est ACC *Alors CA₃*
 et d_{courante} est MAI *Alors CA₃*
 et d_{courante} est DEC *Alors CA₆*
 et d_{but} est L et d_{courante} est ACC *Alors CA₃*
 et d_{courante} est MAI *Alors CA₃*
 et d_{courante} est DEC *Alors CA₆*
 et d_{but} est TL et d_{courante} est ACC *Alors CA₃*
 et d_{courante} est MAI *Alors CA₃*
 et d_{courante} est DEC *Alors CA₆*
 et d_{but} est HE et d_{courante} est ACC *Alors CA₃*
 et d_{courante} est MAI *Alors CA₃*
 et d_{courante} est DEC *Alors CA₆*

$$V_{\text{finale}} = V_2$$

Si V_{courante} est L et d_{but} est TP et d_{courante} est ACC *Alors CA₃*
 et d_{courante} est MAI *Alors CA₃*
 et d_{courante} est DEC *Alors CA₅*
 et d_{but} est P et d_{courante} est ACC *Alors CA₃*
 et d_{courante} est MAI *Alors CA₃*
 et d_{courante} est DEC *Alors CA₅*
 et d_{but} est L et d_{courante} est ACC *Alors CA₃*
 et d_{courante} est MAI *Alors CA₃*
 et d_{courante} est DÉC *Alors CA₅*
 et d_{but} est TL et d_{courante} est ACC *Alors CA₃*
 et d_{courante} est MAI *Alors CA₃*
 et d_{courante} est DEC *Alors CA₅*
 et d_{but} est HE et d_{courante} est ACC *Alors CA₃*

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_5

$$V_{\text{finale}} = V_3$$

Si V_{courante} est L et d_{but} est TP et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_4

et d_{but} est P et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_4

et d_{but} est L et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_4

et d_{but} est TL et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_4

et d_{but} est HE et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_4

$$V_{\text{finale}} = V_4$$

Si V_{courante} est L et d_{but} est TP et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_3

et d_{but} est P et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_3

et d_{but} est L et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_3

et d_{but} est TL et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_3

et d_{but} est HE et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_3

$$V_{\text{finale}} = V_0$$

Si V_{courante} est R et d_{but} est TP et d_{courante} est ACC Alors CA_3

et d_{courante} est MAI *Alors* CA_3
 et d_{courante} est DEC *Alors* CA_5
 et d_{but} est TL et d_{courante} est ACC *Alors* CA_3
 et d_{courante} est MAI *Alors* CA_3
 et d_{courante} est DEC *Alors* CA_5
 et d_{but} est HE et d_{courante} est ACC *Alors* CA_3
 et d_{courante} est MAI *Alors* CA_3
 et d_{courante} est DEC *Alors* CA_5

$$V_{\text{finale}} = V_3$$

Si V_{courante} est R et d_{but} est TP et d_{courante} est ACC *Alors* CA_3
 et d_{courante} est MAI *Alors* CA_3
 et d_{courante} est DEC *Alors* CA_4
 et d_{but} est P et d_{courante} est ACC *Alors* CA_3
 et d_{courante} est MAI *Alors* CA_3
 et d_{courante} est DEC *Alors* CA_4
 et d_{but} est L et d_{courante} est ACC *Alors* CA_3
 et d_{courante} est MAI *Alors* CA_3
 et d_{courante} est DEC *Alors* CA_4
 et d_{but} est TL et d_{courante} est ACC *Alors* CA_3
 et d_{courante} est MAI *Alors* CA_3
 et d_{courante} est DEC *Alors* CA_4
 et d_{but} est HE et d_{courante} est ACC *Alors* CA_3
 et d_{courante} est MAI *Alors* CA_3
 et d_{courante} est DEC *Alors* CA_4

$$V_{\text{finale}} = V_4$$

Si V_{courante} est R et d_{but} est TP et d_{courante} est ACC *Alors* CA_3
 et d_{courante} est MAI *Alors* CA_3
 et d_{courante} est DEC *Alors* CA_3
 et d_{but} est P et d_{courante} est ACC *Alors* CA_3
 et d_{courante} est MAI *Alors* CA_3
 et d_{courante} est DEC *Alors* CA_3
 et d_{but} est L et d_{courante} est ACC *Alors* CA_3
 et d_{courante} est MAI *Alors* CA_3
 et d_{courante} est DEC *Alors* CA_3
 et d_{but} est TL et d_{courante} est ACC *Alors* CA_3
 et d_{courante} est MAI *Alors* CA_3
 et d_{courante} est DEC *Alors* CA_3
 et d_{but} est HE et d_{courante} est ACC *Alors* CA_3

et d_{courante} est MAI Alors CA_3

et d_{courante} est DEC Alors CA_3

$$V_{\text{finale}} = V_0$$

Si V_{courante} est TR et d_{but} est TP et d_{courante} est ACC Alors CA_2

et d_{courante} est MAI Alors CA_2

et d_{courante} est DEC Alors CA_1

et d_{but} est P et d_{courante} est ACC Alors CA_2

et d_{courante} est MAI Alors CA_2

et d_{courante} est DEC Alors CA_1

et d_{but} est L et d_{courante} est ACC Alors CA_2

et d_{courante} est MAI Alors CA_2

et d_{courante} est DEC Alors CA_1

et d_{but} est TL et d_{courante} est ACC Alors CA_2

et d_{courante} est MAI Alors CA_2

et d_{courante} est DEC Alors CA_1

et d_{but} est HE et d_{courante} est ACC Alors CA_2

et d_{courante} est MAI Alors CA_2

et d_{courante} est DEC Alors CA_1

$$V_{\text{finale}} = V_1$$

Si V_{courante} est TR et d_{but} est TP et d_{courante} est ACC Alors CA_2

et d_{courante} est MAI Alors CA_2

et d_{courante} est DEC Alors CA_6

et d_{but} est P et d_{courante} est ACC Alors CA_2

et d_{courante} est MAI Alors CA_2

et d_{courante} est DEC Alors CA_6

et d_{but} est L et d_{courante} est ACC Alors CA_2

et d_{courante} est MAI Alors CA_2

et d_{courante} est DEC Alors CA_6

et d_{but} est TL et d_{courante} est ACC Alors CA_2

et d_{courante} est MAI Alors CA_2

et d_{courante} est DEC Alors CA_6

et d_{but} est HE et d_{courante} est ACC Alors CA_2

et d_{courante} est MAI Alors CA_2

et d_{courante} est DEC Alors CA_6

$$V_{\text{finale}} = V_2$$

Si V_{courante} est TR et d_{but} est TP et d_{courante} est ACC Alors CA_2

et d_{courante} est MAI *Alors CA₂*
et d_{courante} est DEC *Alors CA₃*
et d_{but} est TL et d_{courante} est ACC *Alors CA₂*
et d_{courante} est MAI *Alors CA₂*
et d_{courante} est DEC *Alors CA₃*
et d_{but} est HE et d_{courante} est ACC *Alors CA₂*
et d_{courante} est MAI *Alors CA₂*
et d_{courante} est DEC *Alors CA₃*