

PROJET MO103 : RAPPORT INTERMÉDIAIRE

PHASE D'INGÉNIERIE SYSTÈME

Groupe 4

Janvier 2021

1 Cadre général

1.1 Produit et son concept

Nous cherchons à construire un robot sumo autonome dont l'objectif est de remporter un combat face à un autre robot sumo. Ce combat se déroule sur un dohyo d'environ 1m^2 . Un robot est déclaré vainqueur s'il parvient à pousser l'autre en dehors du dohyo ou bien, en cas d'égalité à la fin du temps imparti, s'il est le plus léger. Pour réaliser ce robot sumo, nous nous appuyons sur le "Bioloid Beginner Kit" ainsi que sur des accessoires en papier ou en carton.

Un tel robot cherche donc à répondre au besoin : "gagner le combat de sumo". Le robot devra donc être capable, de manière autonome, de pousser son adversaire en dehors du dohyo. De plus, il doit effectuer cette action sans être endommagé.

1.2 Cycle de vie

Le premier soir, tout le groupe s'est réuni afin de trouver des idées et tenter de prendre une décision quant au plan mécanique à adopter pour le robot. Le plan mécanique devant correspondre à une stratégie de combat déterminée, il fallait en même temps développer des idées de programmation en accord avec la géométrie choisie pour le robot. Le lendemain matin, les choix de géométrie et de stratégie à adopter pour le robot étaient faits. Le plan choisi était de concevoir un robot à 4 roues avec deux bras à l'avant pour tenter de prendre en tenaille l'adversaire. Une fois ce choix effectué, le groupe s'est divisé en deux sous-parties : la première dont l'objectif était de construire au plus vite le robot et la deuxième était chargée de programmer le robot pour qu'il remporte ses combats. Les deux équipes communiquaient fréquemment pour faire part de leurs avancements et préparer les tests qu'il allait falloir effectuer dès que le robot serait terminé, que ce soit pour obtenir des données nécessaires à la programmation ou des données mécaniques afin d'optimiser ensuite la construction du robot.

Les roues ont été fixées de telle sorte que le robot soit le plus bas possible afin d'avoir une stabilité optimale. Une première idée avait été de fixer deux roues sur la partie centrale et les deux autres sur chacun des bras mais nous l'avons rejetée car cela aurait empêché le robot d'avancer facilement. Les bras ont ensuite été fixés en assemblant les pièces restantes. Nous avons dû modifier l'extrémité des bras qui initialement étaient constituées de deux grandes plaques. En effet, celles-ci touchaient le sol et auraient engendré des frottements qui pouvaient freiner le robot donc nous avons opté pour des pinces afin que le robot conserve un design original. Ensuite, le capteur a été fixé à l'avant du robot entre les deux bras.

Le mercredi matin, après une journée et demie de travail, la construction mécanique du robot était achevée et la programmation était en bonne voie. Les tests pouvaient donc commencer.

Le jeudi a permis de réaliser de très nombreux tests en conditions réelles (cf 4.3 pour plus de détails) car le robot était enfin finalisé ainsi que des simulations plus réalistes des combats potentiels du vendredi après-midi.

Enfin, le vendredi matin a été consacré aux derniers ajustements en fonction des stratégies que nous avons pu observer chez nos adversaires et à la finalisation du rapport. Par ailleurs, nous avons réfléchi à différentes options de décorations pour rendre notre robot sumo unique sur le dohyo.

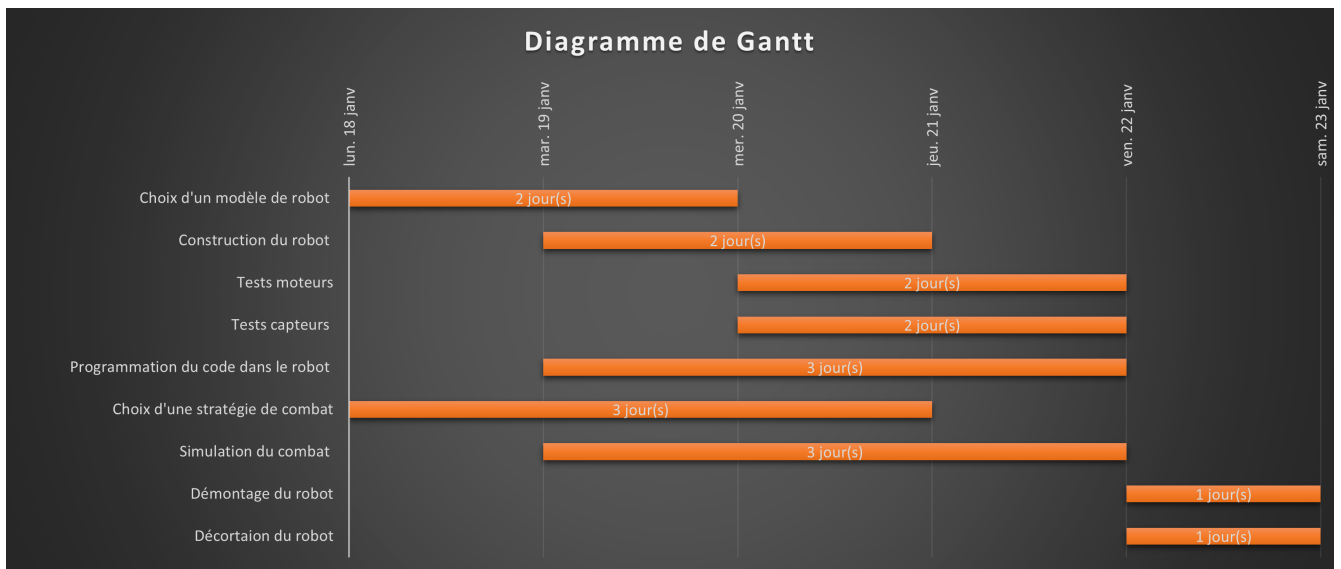


FIGURE 1 – Diagramme de Gantt

1.3 Exigence principale

Le robot est créé avant tout pour remporter un tournoi de robot sumo contre d'autres robots sumo.

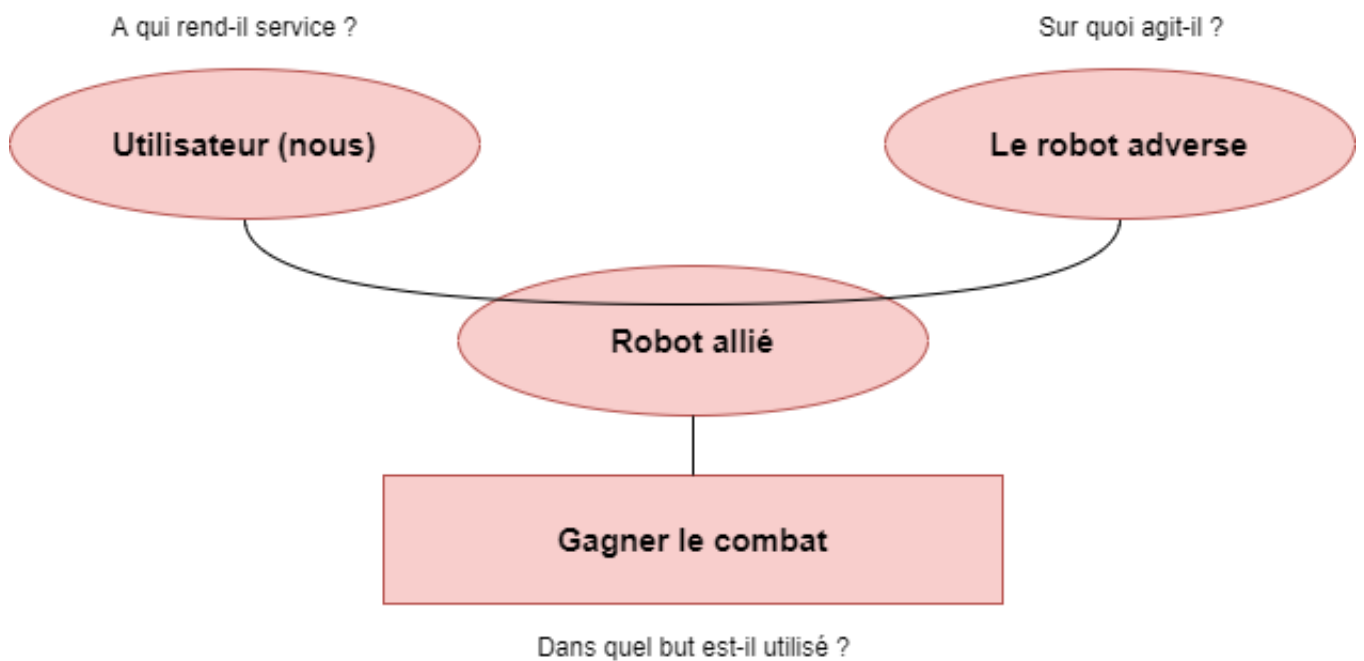


FIGURE 2 – Diagramme "bête à corne"

Ainsi les parties prenantes au projet sont uniquement les concepteurs du robot (nous) et le robot adverse. Les enjeux d'une telle réalisation sont d'obtenir un robot autonome lors du combat.

1.4 Fonctions

Pour réaliser cet objectif, le robot sumo doit donc avoir comme fonctions principales de se déplacer, d'attaquer son adversaire, de se protéger et de détecter ou repérer des objets. Mais il doit aussi satisfaire certaines fonctions contraintes notamment rester dans le terrain, commencer à se déplacer que 3 secondes après son allumage et ne pas s'endommager lors du combat.

Fonctions principales	Description	Fonctions contraintes	Description
FP 1	Se déplacer	FC 1	Rester dans le dohyo
FP 2	Attaquer	FC 2	Attendre au début du combat
FP 3	Se protéger	FC 3	Ne pas endommager le matériel
FP 4	Détecter, repérer des objets	FC 4	Ne pas perdre au poids
		FC 5	Le robot doit rentrer sur le terrain

TABLE 1 – Fonctions principales et contraintes

Nombre de ces fonctions dépendent d'interactions avec des éléments du milieu extérieur comme le montre le diagramme suivant.

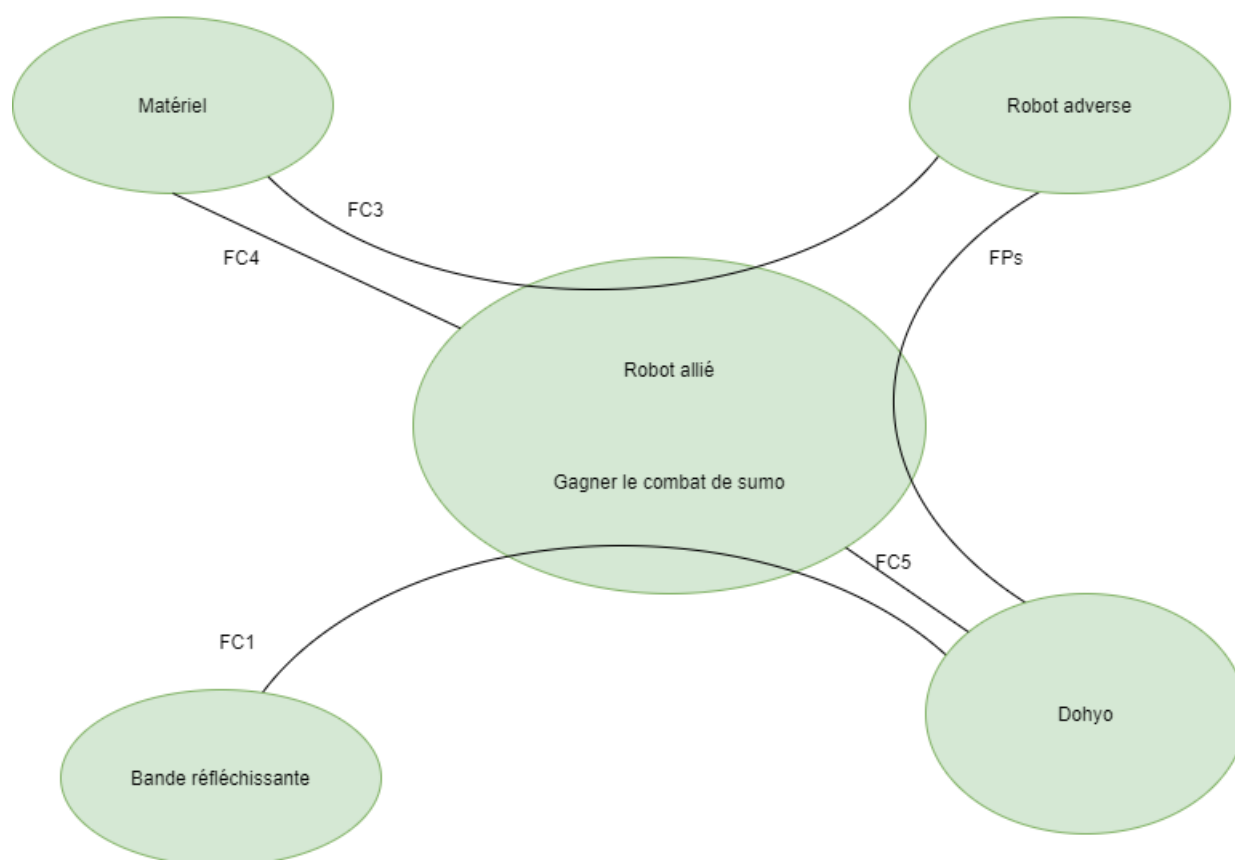


FIGURE 3 – Diagramme "pieuvre"

Au fil des jours, nous avons souhaité rendre notre robot plus "fun" et pour se faire nous nous sommes fixés quelques objectifs optionnels.

Objectif Optionnel	Description
OO 1	Avoir une forme originale
OO 2	Jouer de la musique PENDANT le combat
OO 3	Faire fonctionner les lumières au rythme de la musique

TABLE 2 – Objectifs optionnels

1.5 Description et chaîne fonctionnelle

Comme les fonctions principales nous l'indiquent, notre robot doit donc être en mesure de se déplacer, d'attaquer et de se défendre. La fonction principale visant à détecter des objets peut se décomposer en deux fonctions secondaires : attaquer et se protéger. En effet, pour attaquer, il faut être capable de détecter l'adversaire et pour se protéger, il faut pouvoir rester dans le terrain et donc repérer les limites du terrain (les bandes réfléchissantes).

En ce qui concerne les déplacements, notre robot ayant quatre roues, ils se résument à des rotations et des translation vers l'avant ou vers l'arrière. Ces déplacements sont aussi indispensables à l'attaque car le fait de pousser le robot adverse pour le faire sortir du terrain nécessite avant tout que notre robot avance et donc fasse une translation vers l'avant.

Enfin, que ce soit pour se défendre des attaques ennemies ou pour attaquer notre adversaire, le robot se doit d'être robuste car il ne doit pas être abîmé lors des combats.

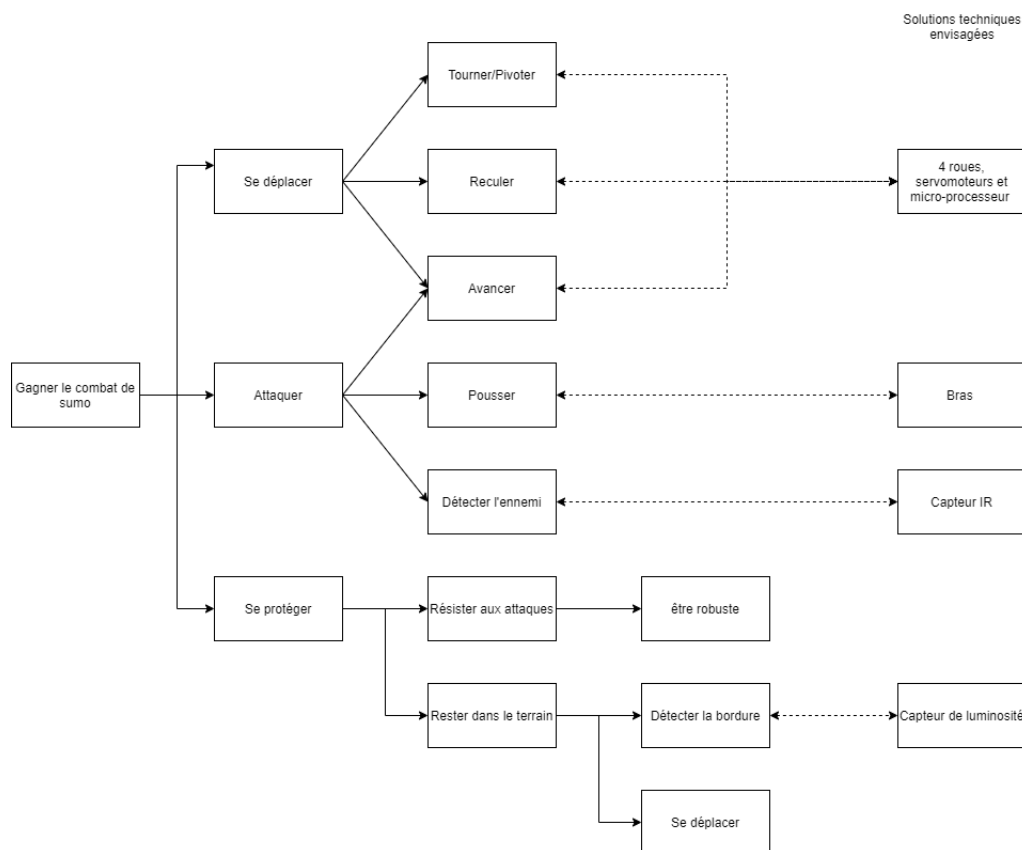


FIGURE 4 – Décomposition fonctionnelle

2 Documentation de référence

2.1 Cahier des Charges Fonctionnel

Ici, le cahier des charges consiste à respecter le règlement de la compétition. Voici le lien du document en question : <https://perso.ensta-paris.fr/~toralba/M0103/docs/cahierDesCharges.pdf>

2.2 Articles

Pour construire notre robot sumo, nous nous sommes appuyés sur le guide d'utilisateur du robot Bioloid produit par l'entreprise, qui explique le fonctionnement de chaque composant et qui donne aussi des exemples de robots à réaliser. De plus, nous nous sommes également basés sur un document de l'ENSTA intitulé "Mise en oeuvre du robot" qui détaille lui aussi mais de manière plus synthétique comment doit fonctionner le robot ainsi que les caractéristiques techniques propres aux composants, notamment aux capteurs et aux moteurs qui

nous ont permis d'envisager des tests précis pour vérifier nos hypothèses de programmation et de construction mécanique. Un document présentant une manière détaillée de coder les robots nous a aussi été utile. Voici les liens des documents mentionnés précédemment :

- <https://perso.ensta-paris.fr/~toralba/M0103/docs/guide-bioloid.pdf>
- <https://perso.ensta-paris.fr/~toralba/M0103/docs/quick-start-bioloid.pdf>
- <https://perso.ensta-paris.fr/~toralba/M0103/docs/MiseEnOeuvreDuRobot.pdf>
- <https://perso.ensta-paris.fr/~toralba/M0103/docs/programmationCM530.pdf>

3 Exigences et contraintes

3.1 Exigences fonctionnelles et contraintes techniques

Fonctions principales	Description	Critères	Précision
FP 1	Se déplacer		
FP 2	Attaquer	Vitesse, force, équilibre...	$v > 8\text{cm/s}$
FP 3	Se protéger	Résister à une attaque frontale	
FP 4	Détecter, repérer des objets	Portée du capteur IR	20cm

TABLE 3 – Fonctions principales et exigences

Fonctions contraintes	Description	Critères	Précision
FC 1	Rester dans le dohyo	Taille du terrain : 1m x 1m	20%
FC 2	Attendre au début du combat	Temps : délai au démarrage	$> 3\text{s}$
FC 3	Ne pas endommager le matériel	Matériaux de construction	
FC 4	Ne pas perdre au poids	Masse du robot	$< 2\text{kg}$
FC 5	Le robot doit rentrer sur le terrain	Taille du robot	$< 1\text{m}$

TABLE 4 – Fonctions contraintes et exigences

3.2 Exigences opérationnelles

L'aspect opérationnel le plus important est l'autonomie du robot lors du combat. Ainsi il faut qu'il puisse se déplacer seul et trouver son adversaire seul. Pour se faire, plusieurs stratégies ont été testées. De plus, il doit rester alimenter pour fonctionner. On observe donc les exigences opérationnelles suivantes :

Exigence	Besoin	Critères	Précision
Pouvoir suivre la stratégie de combat	Être autonome	Code	Aucune erreur
Maintenir l'alimentation du robot pendant le combat	Fonctionner	Câble d'alimentation	$> 3\text{m}$

TABLE 5 – Exigences opérationnelles

3.3 Exigences d'environnement

La zone de combat, constituée du dohyo et du robot adverse, est l'environnement principal de "vie" de notre robot. Ceci implique certaines exigences.

Exigence	Besoin	Critères	Précision
Pouvoir réutiliser les pièces des robots (allié et adverse)	Ne pas les endommager	Matériaux	
Limiter les frottements sur la moquette	Avancer	Contact avec le sol	Aucun (hors roues)

TABLE 6 – Exigences d'environnement

3.4 Exigences d'interface

La stratégie retenue a donc dû être codé sur l'unité de commande du robot et un travail d'équipe a dû être fourni. Ceci a imposé certaines contraintes.

Exigence	Besoin	Critères	Précision
Coder la stratégie	Être autonome	Coder en C	Uniquement en C
Constituer une équipe de 10	Travailler en équipe	Humain	+/- 1 personne
Ne pas être plus de 6 dans une salle	Respecter les normes sanitaires	Humain	Aucune

TABLE 7 – Exigences d'interface

3.5 Exigences d'études et solutions imposées

Ce projet utilise les ressources fournies par l'école et s'inscrit dans le cadre d'un tournoi (interne à l'école) qui possède donc son propre règlement. Ceci génère certaines exigences.

Exigence	Besoin	Critères	Précision
Utiliser la boîte "Bioloid Beginner Kit"	Construire un robot	Boîte	1 seule boîte

TABLE 8 – Exigences d'études et solutions imposées

3.6 Exigences d'assurance de résultat

Comment s'assurer que notre robot soit bien capable de remporter un combat de sumo ? Tout cela ne peut se faire que par des tests (détaillés dans le plan de validation en 4.2) qui devront vérifier certaines exigences.

Exigence	Besoin	Critères	Précision
Détecter l'autre robot	Gagner le combat	Porté du capteur IR et vitesse de rotation	<6s/tour
Détecter les limites	Gagner le combat	Seuil du capteur IR	Détection possible
Pousser le robot adverse en dehors du dohyo	Gagner le combat	Vitesse en translation	>8cm/s

TABLE 9 – Exigences d'assurance de résultat

4 Tests et plans

4.1 Plan de vérification

Le plan de vérification vise à assurer que le système respecte les exigences du système, il est tout d'abord composé des tests visant à satisfaire les fonctions principales du robot. Il faut donc faire des tests pour s'assurer que le robot :

- peut bien se déplacer donc effectuer des rotations et des translations ;
- peut attaquer et donc être capable de déplacer des objets de dimension similaire à celle que pourrait avoir un robot ennemi ;
- peut être capable de résister à une charge ennemi sans s'abîmer ;
- peut détecter des objets et les repérer : en particulier un robot ennemi et la ligne de démarcation du terrain.

Ces exigences fonctionnelles vont souvent de pair avec les fonctions contraintes et donc les tests seront souvent les mêmes sauf pour vérifier que le robot ne commence à bouger avant 3 secondes. De plus, comme nous nous sommes fixés des objectifs optionnels, il faut aussi mettre au point des tests pour veiller à leur bonne réalisation :

- vérifier que la musique et les lumières n'empêchent pas le bon fonctionnement du robot (test avec et sans musique pour vérifier qu'il effectue les mêmes actions) ;
- s'assurer que les lumières s'activent bien en rythme.

4.2 Plan de validation

Afin de vérifier que notre système réponde bien au besoin spécifié : gagner le combat, nous devons nous assurer que le robot soit autonome lors du combat : libéré dans le dohyo, il doit être capable d’y rester tout en cherchant son adversaire et le cas échéant le pousser vers l’extérieur du terrain. Nous devons notamment vérifier que notre robot puisse réagir face à différentes options de stratégies de combat de l’ennemi, par exemple lors des simulations informatiques.

4.3 Tests

4.3.1 Premiers tests unitaires réalisés

Les différentes fonctions du robot ont été testées séparément dans l’ordre dans lesquelles elles ont été mises en place.

- Tests sur les capteurs : ce sont les premiers à avoir été réalisés par l’équipe programmation car le corps du robot était parallèlement en construction par l’autre équipe. Une fonction affichant les valeurs brutes des capteurs a permis de déterminer les seuils à utiliser pour la détection infrarouge du robot adverse, ainsi que celui utilisé pour la détection de la bande lumineuse. Il a aussi permis d’invalider l’utilisation du capteur de luminosité (résultats parfaitement non exploitables).
- Tests sur un moteur unique : tester les différentes vitesses ainsi que le fonctionnement du code embarqué. Il s’agit simplement de fixer la vitesse.
- Tests sur plusieurs moteurs : vérifier que les moteurs tournent dans le bon sens et déterminer les vitesses à utiliser pour tourner.
- Tests du buzzer et des LEDs : vérifier qu’ils fonctionnent, préparer des fonctions permettant de les utiliser facilement et de manière programmable.

Nous avons retenu le capteur infrarouge pour la détection des bandes lumineuses, mais celui-ci avait tendance à saturer lorsqu’il était utilisé sur un sol en plastique (comme celui des salles de l’ENSTA non équipée du tapis). Il a donc été désactivé pour les autres tests.

4.3.2 Mise en place d’une simulation / Programmation plus avancée

Nous avons assigné en parallèle du développement un membre de l’équipe à l’élaboration d’une stratégie de combat. Afin de pouvoir tester les stratégies plus simplement avant leur implémentation en C et la fin de la construction du robot, il a décidé de mettre en place une simulation numérique en Python du robot et d’un adversaire. Cela nous a permis de nous rendre compte par exemple des stratégies qui ne couvrent pas l’intégralité du terrain.

Le reste de l’équipe de développement s’est consacré à implémenter les fonctions “standard” permettant le contrôle du robot et celles correspondant aux différentes stratégies à adopter en fonction des états.

Une particularité du code retenu est l’objectif optionnel OO2 nécessitant de jouer de la musique pendant le combat. Ne pouvant interrompre l’exécution du code, nous avons en effet dû mettre en place une horloge en modifiant les fonctions fournies avec le SDK (Software Development Kit) afin de pouvoir jouer des notes à des instants prédéterminés.

4.3.3 Suite des tests (plus aboutis)

Une fois la construction du robot globalement achevée, le code arrivait en parallèle à maturité, ce qui nous a permis de procéder à une seconde batterie de tests, cette fois-ci plus longs et en conditions réelles (sur le dohyo).

- Tests de détection d’objets et des bandes réfléchissantes : ces tests nous ont permis de vérifier que le code de détection d’objets et de bandes lumineuses fonctionnait correctement. Il s’agissait dans un premier temps d’un simple retour de valeurs des capteurs, afin de vérifier que les résultats des premiers tests de capteurs s’appliquent bien à la forme de notre robot. Il en est ressorti que nous devons surélever le capteur infrarouge pour éviter la saturation lors de la détection des bandes blanches.
- Tests de déplacements : nous avons vérifié que le robot roulait correctement sur le sol de moquette du dohyo, et mesuré sa vitesse pour affiner la simulation. Nous avons aussi vérifié sa capacité à pousser le robot adverse.
- Tests de suivi d’objets et de réaction aux bandes : nous avons vérifié que le robot réagissait correctement en cas de détection d’un adversaire ou d’une bande (qu’il attaquait ou reculait pour rester sur le terrain).

Nous avons aussi vérifié (et corrigé), les gestions d'événements pour que le robot puisse interrompre son action à tout moment si elle n'était plus conforme à la stratégie.

- Tests de la parallélisation de la musique avec les autres tâches : comme expliqué précédemment, pouvoir jouer de la musique sans interruption de la stratégie a été difficile, et nous avons vérifié que cette addition ne perturbait pas l'exécution.

4.3.4 Compte-rendu de la phase de tests

- Les tests ont mis en exergue des problèmes de construction mécanique (capteur infrarouge trop près du sol), qui ont nécessité une modification du robot.
- Ils ont permis de mettre au point un simulateur numérique qui nous a aidé pour l'élaboration de la stratégie.
- Ils ont permis de vérifier le bon fonctionnement du code.

5 Conclusion

Notre démarche de conception s'inscrit dans une démarche d'ingénierie système car nous avons commencé par analyser le besoin rencontré, en prenant en compte l'environnement dans lequel il se trouve. Puis nous avons détaillé les contraintes et les exigences auxquelles notre robot devait satisfaire. Nous avons ensuite identifié les tâches et les fonctions que notre robot devait réaliser afin de commencer à suggérer des propositions de solutions. Pour ce faire, nous avons divisé le groupe en plusieurs parties, chacune assignée à une tâche bien précise afin de répondre à des exigences bien spécifiques du robot.

Pour la conception de l'architecture nous avons commencé à y réfléchir chacun de notre côté après avoir posé un cadre rapide (le matériel avait été récupéré à ce moment-là). Ensuite, nous avons exposé nos suggestions, puis débattu chaque idée. L'architecture n'était pas décrite en détail mais nous avons déjà choisi une configuration globale pour notre robot (2 bras à l'avant, 4 roues et nous avons tout d'abord envisagé une rampe à l'avant qui fut abandonnée).

De plus, pour évaluer au mieux les performances de notre robot qui restait simplement théorique dans un premier temps, nous avons fait des simulations avant de réaliser nos tests qui nous ont permis d'améliorer notre modèle. Les résultats donnés par les simulations ont ainsi pu être comparés aux tests réels, ce qui a permis d'affiner petit à petit les simulations pour se rapprocher autant que possible de la réalité et aboutir à la stratégie la plus efficace possible en combat.

Finalement, notre démarche a suivi une évolution bien précise, propre à l'ingénierie système :

- analyse des besoins ;
- spécification des exigences ;
- conception de l'architecture ;
- conception détaillée et réalisation ;
- tests unitaires ;
- intégration, vérification et validation.

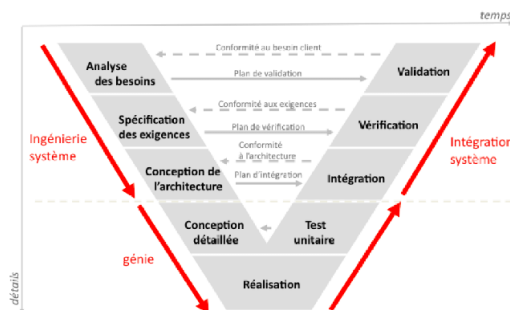


FIGURE 5 – Cycle en V

PS : La description complète de la conception de l'architecture sera faite dans le second rapport.