Brainstorm :

* Robot
* doyho de 1 mètre de côté : trop grand pour ne pas se déplacer
* Estimer position des roues pour rester sur tapis
* Foncer dans le tas?
* Utiliser le capteur pour s’orienter vers l’ennemi? ou pour détecter les marques au sol
* Option 4 roues
* Option 3 roues

Proposition de stratégies :

* rampe (difficile à estimer: le véhicule va-t-il vraiment monter dessus?)
* esquive (contournement? suivre le marquage?).
* foncer dans le tas (tout le monde va faire ça, style “missile tête chercheuse''). On peut l’intégrer à l’esquive: foncer dans le tas ET faire des petites manoeuvres de contournement ou qqchose
* contrôle à distance (cheaté)
* Leurre + Contournement
* Obstruer les capteurs ennemis
* Adaptatif (phase observation, puis choix d’une stratégie)
* Chasse neige avec angle (idée de Lionel)

Analyse des besoins :

* Se déplacer
* Stratégie d’attaque/défense

| Programmation (3.5) | Test (3) | Assemblage (4.5) |
| --- | --- | --- |
| Gabriel (respo G) | Lionel | Lucas (scribe) |
| Antoine | Seta (scribe) | Bastien (respo test) |
| Lionel (respo test) | Annie | Manon |
| Arsène (scribe) | Bastien | Kawtar |
|  |  | Bruce |

Idées de stratégie :

* bras articulés (sur les côtés) pour faire bouger l’adversaire
* rampe pour déstabiliser l'adversaire
* leurre puis se décaler pour pousser l’adversaire

| Réflexion 1 | Réflexion 2 | Réflexion 3 |
| --- | --- | --- |
| Gabriel | Bruce | Arsène |
| Manon | Seta | Bastien |
| Lionel | Annie | Antoine |
| Lucas |  | Kawtar |
|  |  |  |

à faire :

* comment trouver l’autre robot
* comment on gère la non-sortie de terrain (avec automate : état contact, état recherche)

Phase 1 :

→ aller vers le centre pour s’éloigner des bords et essayer de trouver l’autre robot

(dès que le robot ennemi est détecté, on change de mode → on passe en mode attaque ou défense)

→ chercher le robot ennemi (les 2 en même temps?)

Phase 2 :

→ aller taper le robot ennemi en se positionnant à un angle donné

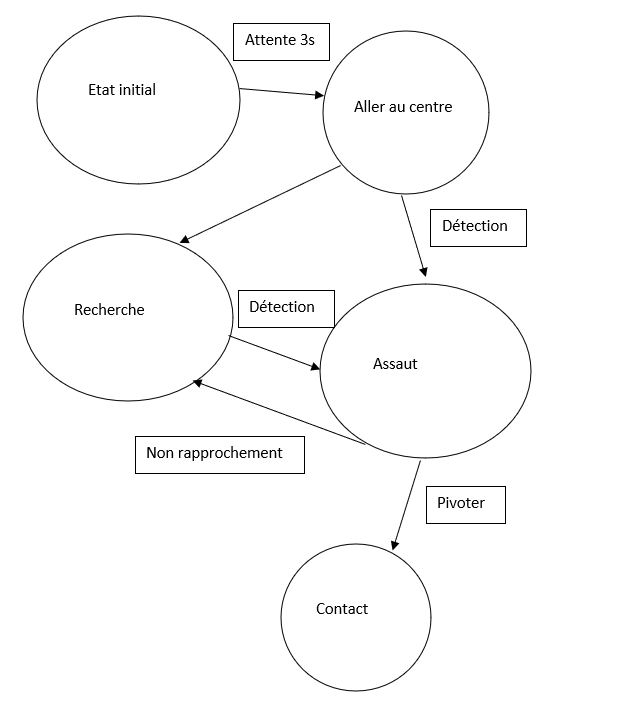
Phase 3 :

→ contact : option 1 : on pousse avec un angle choisi en phase 2

option 2 : défense (essayer de savoir si on a des roues en l’air par comparaison des vitesses des roues et dans ce cas reculer pour récupérer notre équilibre)

Peu importe la situation :

→ Cas d’erreur si on arrive au bord

 1/ on a repéré l’adversaire

→ Phase assaut : (il est dans le demi-cercle décrit par les 2 bras : connaître l’angle et la longueur des bras : environ 30 cm)

2/ on pivote (essayer d’optimiser dans quel sens)

On choisit un sens de rotation arbitraire. On effectue une mesure de la vitesse des roues du moteur en temps normal lors d’une rotation. Si à un moment cette vitesse s’annule voire s’inverse alors on a percuté le robot ennemi et on pivote dans le mauvais sens donc on change le sens de rotation. Si la vitesse des roues est seulement réduite, alors on a bien pivoté dans le bon sens et on peut entrer en phase contact car on a commencé à pousser le robot adverse.

condition d’échec de la manoeuvre de rotation : si on a tourné pendant une certaine durée (ou d’un certain angle) et qu’on a pas de détection sur le robot adverse (pas de changement de la vitesse des roues) → on repasse en mode recherche car on a perdu le robot ennemi

Envisager le cas où le robot adverse cherche à éviter le combat → adopter une stratégie plus offensive

3/ on se rapproche jusqu’à limite du capteur infra-rouge

essayer de repérer quand on tourne par accoups (lorsqu’un bras heurte le robot adverse)

gérer le cas où l’on arrive au bord du terrain

Principe du code de la phase d’assaut :

Au préalable, on a repéré l’ennemi qui se trouve donc entre les bras du robot.

Etape 1 : choix arbitraire d’un sens de rotation et d’un angle limite/durée limite de rotation (à déterminer avec des tests)

Etape 2 : Commencer à pivoter + mesurer la vitesse de rotation des roues (v\_ini) | a voir si l’on peut en même temps qu’on l’on pivote, avancer légèrement en translation vers le robot ennemi

Etape 3 : Chercher s’il y a une modification de la vitesse des roues (contact avec le robot ennemi) et vérifier si l’on détecte toujours l’ennemi

* cas 1 : s’il existe une différence de vitesse des roues entre v\_actuelle et v\_ini alors on a percuté le robot ennemi et on avance vers lui
* cas 2 : si l’on a atteint la durée limite ou l’angle limite sans perturbation de la vitesse de rotation alors
  + sous cas 1 : on détecte toujours le robot, il est en face de nous et on avance vers lui
  + sous cas 2 : on ne le détecte plus, il a peut être reculé, on repasse en mode recherche

On passe en mode contact dès que l’on atteint la limite du capteur infra-rouge (on est alors à moins de 5cm du robot adverse)

Aller au centre :

Idée 1 :

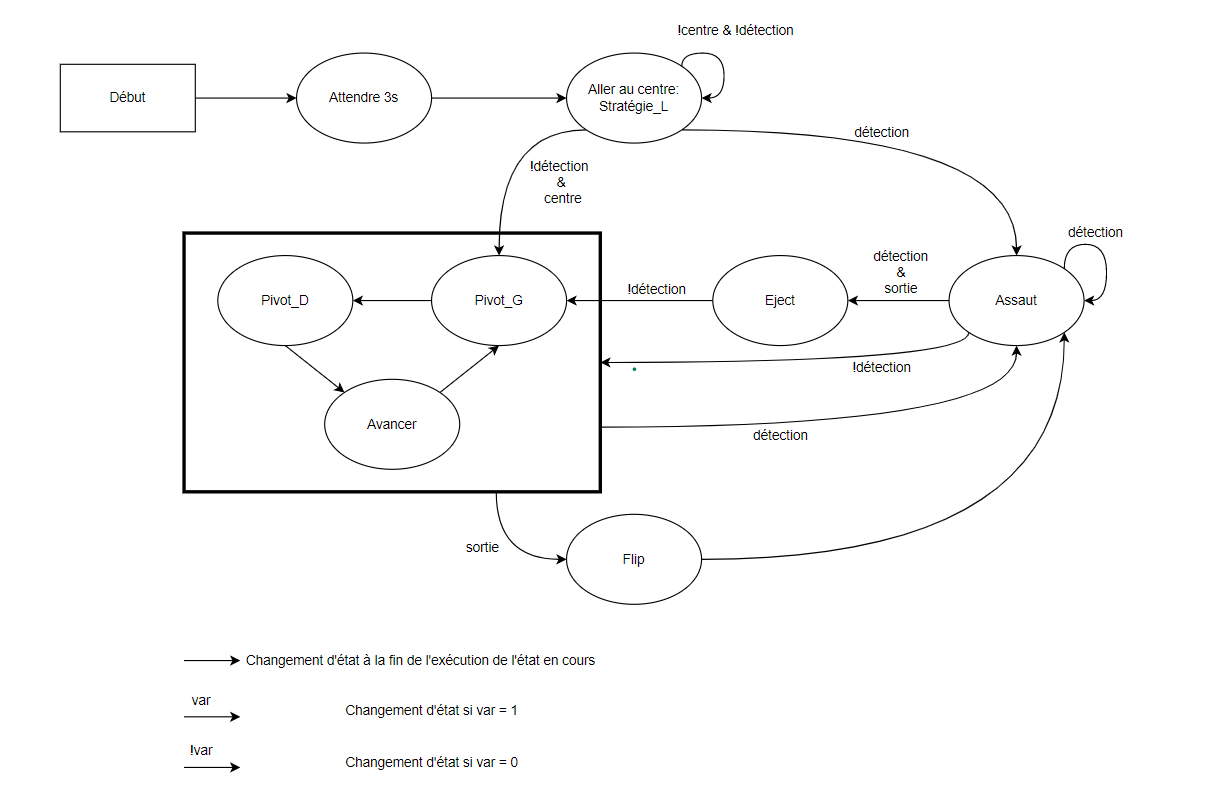
On se rend au centre en avançant tout droit en suivant la diagonale

Idée 2 : On avance sur le côté puis on tourne à partir d'une certaine distance pour arriver vers le centre.

ce qu’il reste à coder :

* aller au centre (voir si on peut le faire en mesurant le temps qu’il faut)
* comment obtenir la position du robot par rapport au dernier arrêt (en polaires)
* tracer un carré autour du centre en recherchant le robot ennemi
* transition vers la phase assaut lorsqu’on a détecté l’adversaire
* phase de contact
* modification du code si l’on détecte la limite du terrain

////Désolé ça n'a rien à faire là, mais je laisse ça là : Les deux versions sont équivalentes? ////



et 