


# Installation de MATLAB

---

## Télécharger

La première étape consiste à créer un compte sur le site internet [MathWorks](#). Lorsque vous remplissez la question « How will you use MathWorks software ? » sélectionnez « Academic use [...] ».

Après avoir créé votre compte, demandez la version d'essai pour la compétition « Maker Faire Paris 2014 ».

Une fois que votre demande a été validée (sous  jours) vous recevrez un email (sur l'adresse mail de votre compte MathWorks). Il devrait ressembler à ça :

Dear Arthur DENT,

...

Thank you for your interest in MATLAB and Simulink products. Your trial NNNNNN starts today and expires on **July 31, 2014**.

[Start Your Trial](#)

This trial includes the following R2014a products:

- MATLAB
- Simulink
- ...

Helpful tips:

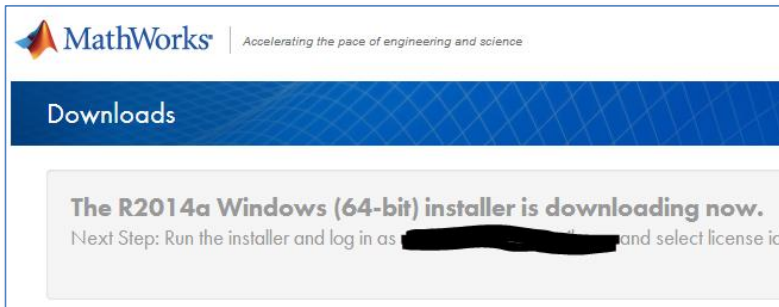
- If you do not have a MathWorks Account, one will be created for you. Check your email inbox for an email titled "MathWorks Trial Login Information".
- Having trouble downloading, installing, or activating your trial software? [Contact support](#) for help.

Best regards,

MathWorks Sales Team

Cliquez alors sur le gros bouton orange « Start Your Trial ».

Une fenêtre d'installation téléchargera depuis votre navigateur internet les fichiers d'installation. L'installation démarrant automatiquement :

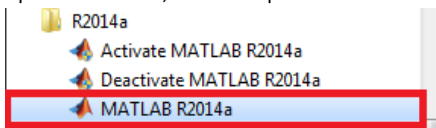


## Premiers pas

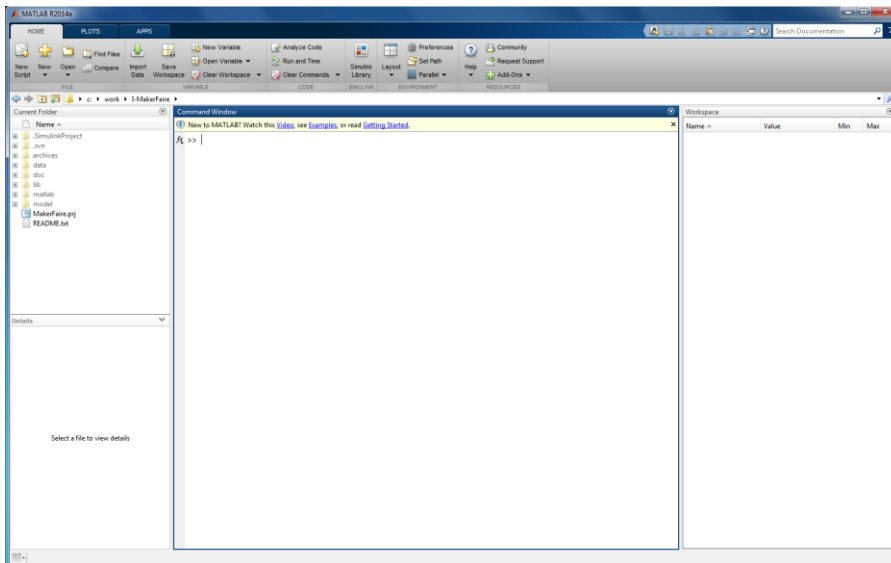
---

### Lancer MATLAB

Après installation, une icône permettant de lancer MATLAB doit avoir été créée dans le menu démarrer.

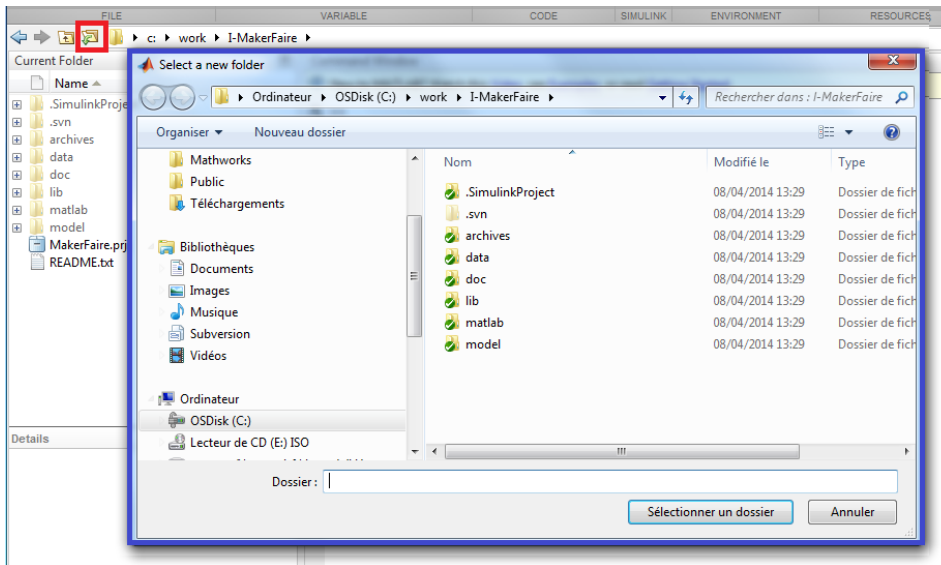


Cliquer sur cette icône permet de lancer l'environnement MATLAB, et ouvre une fenêtre similaire à la figure suivante.

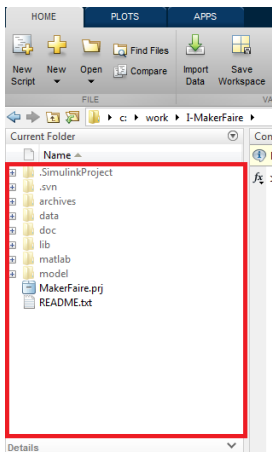


## Ouvrir le modèle

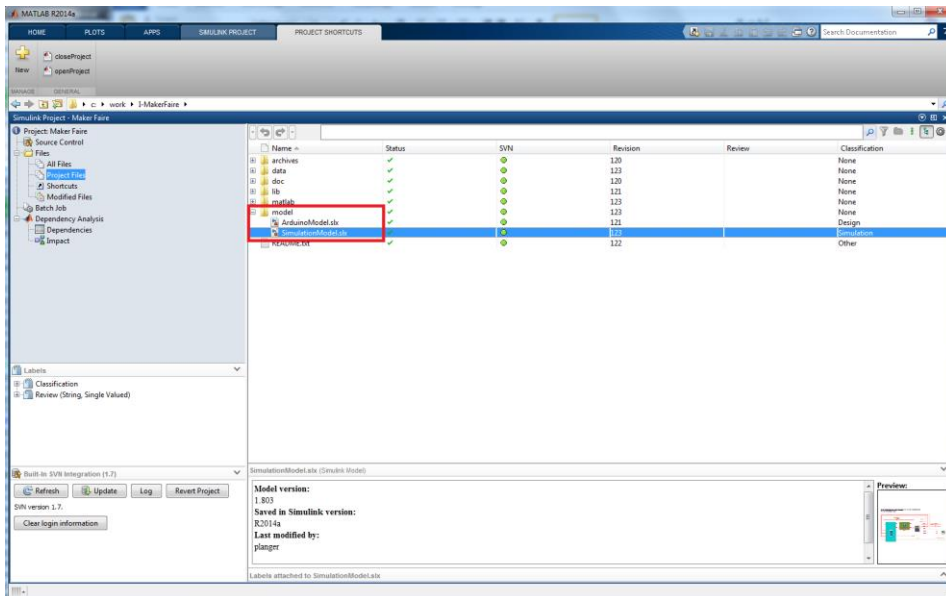
Utilisez le bouton surligné en rouge sur la figure suivante pour sélectionner le répertoire où vous avez installé le modèle de simulation. Ce bouton « Browse for folder » ouvre une boîte de dialogue (surlignée en bleu) permettant de choisir le répertoire.



Après choix du répertoire, la section « current folder » de MATLAB devrait afficher un contenu similaire à celui encadré à la figure suivante.



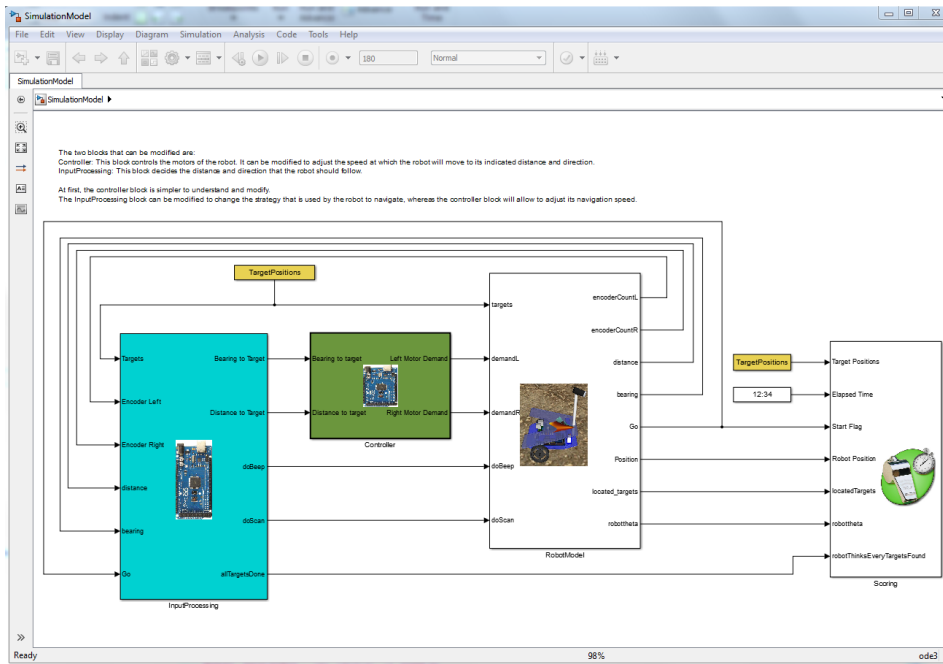
Le projet contenant le modèle et l'environnement de simulation peut être ouvert en double-cliquant sur l'icône « MakerFaire.prj ».



Une fois le projet ouvert, le modèle s'ouvre en double cliquant sur « SimulationModel.slx » dans le répertoire « model » du projet (encadré en rouge sur la figure précédente).

Notez qu'il peut être nécessaire de déplier le contenu du répertoire « model » pour faire apparaître son contenu.

L'ouverture du modèle devrait créer une nouvelle fenêtre similaire à la fenêtre suivante :

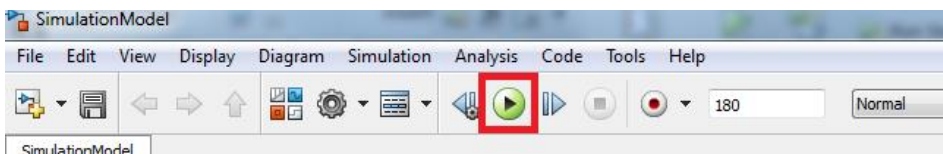


Cette fenêtre contient le modèle de simulation.

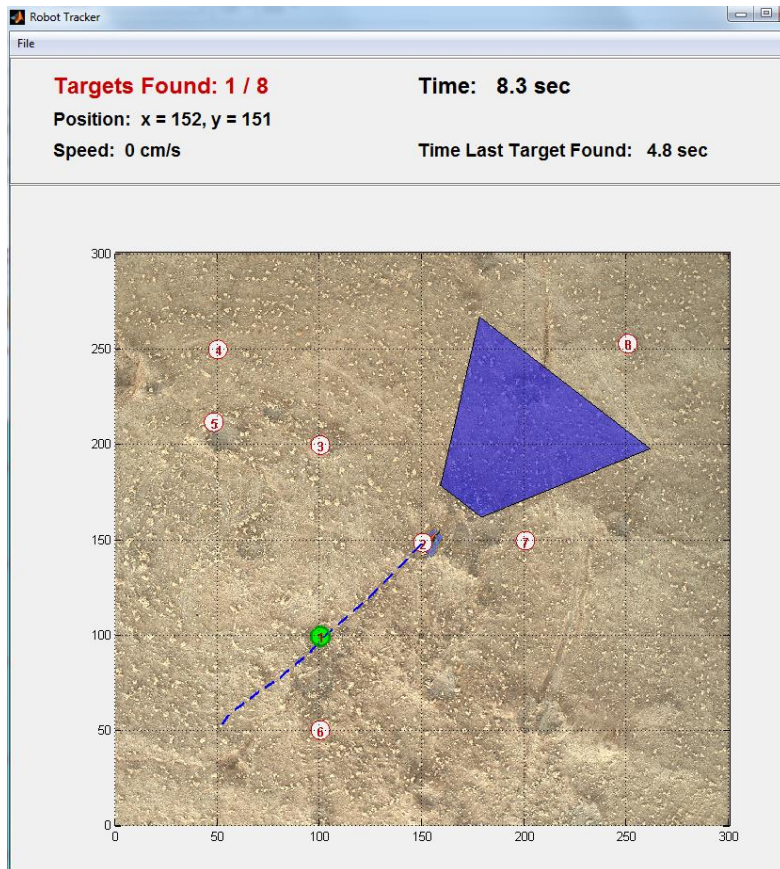
## Lancer la simulation

Avant d'examiner et de modifier le modèle, il est important de comprendre comment lancer la simulation, afin de pouvoir observer les résultats de ses modifications ou expérimentations.

Pour lancer la simulation, il suffit de cliquer sur le bouton « Run » illustré à la figure suivante. Ce bouton vérifie les éventuelles erreurs dans le modèle, et lance la simulation dans le cas où aucune erreur n'est rencontrée.



Le modèle initial ne contient (normalement) pas d'erreur. Aussi après quelques instants correspondant à la vérification et la compilation du modèle, la fenêtre suivante devrait apparaître.

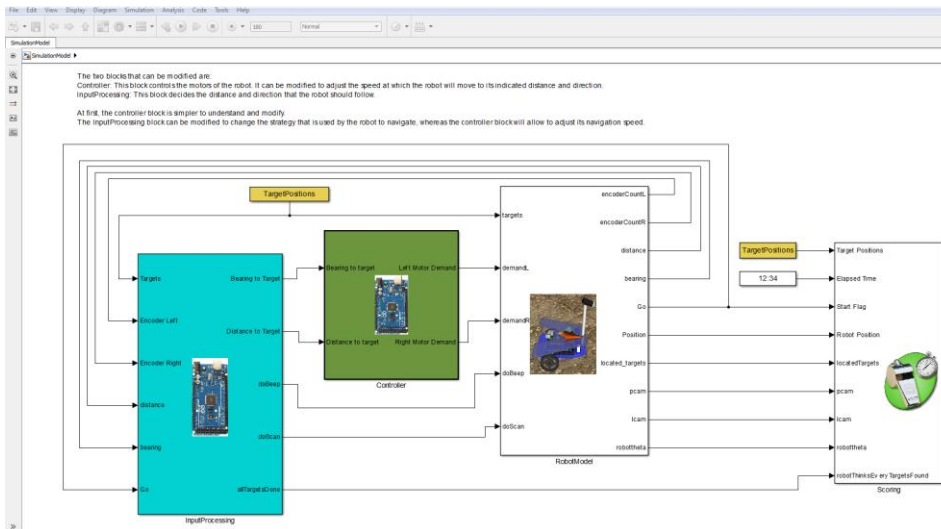


Cette fenêtre montre la trajectoire suivie par le robot, comme illustré sur la capture d'écran suivante. Le quadrilatère bleu décrit la zone « vue » par la caméra du robot, et les sites deviennent verts lorsqu'elles ont été correctement parcourues par le robot. Cette fenêtre affiche aussi le temps et le nombre de sites correctement atteintes par le robot.

### Vue d'ensemble du modèle

La figure ci-dessous représente le modèle de simulation du système tel qu'il apparaît lorsque vous l'ouvrez. Il comprend à la fois le programme du robot, et une modélisation de l'environnement.

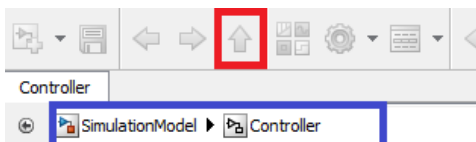
Les blocs blancs (« RobotModel » et « Scoring ») correspondent à l'environnement de simulation, et à la modélisation physique du robot, et ne doivent donc pas être modifiés.



Les blocs correspondant au programme du robot, et qui peuvent donc être modifiés sont le bloc vert (« Controller ») et le bloc bleu (« InputProcessing »):

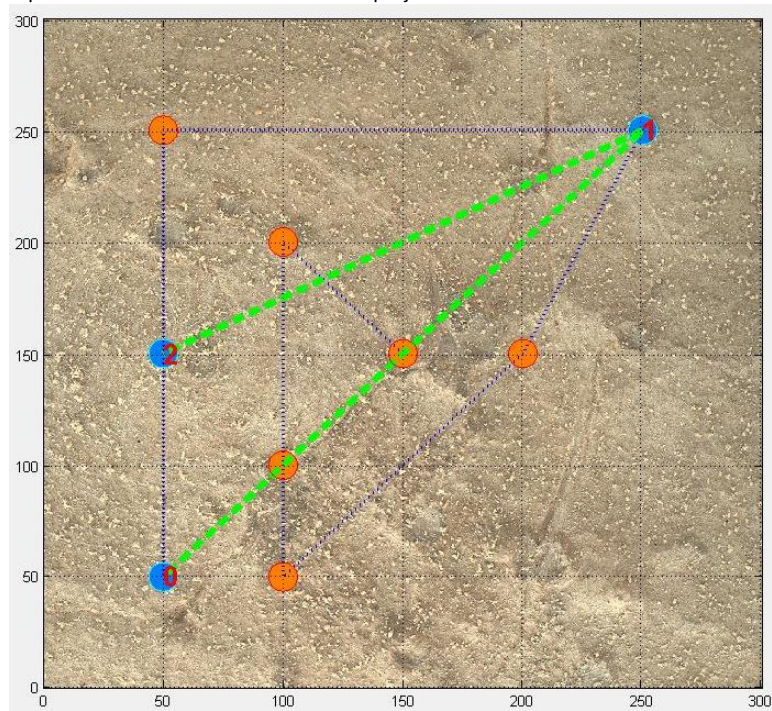
- Le bloc vert (« Controller ») contrôle les moteurs des roues du robot. Le modifier peut permettre au robot d'avancer ou de tourner plus rapidement.
- Le bloc bleu implémente la stratégie de déplacement du robot. Etant donné la position estimée du robot, et les informations de la caméra, ce bloc décide de la direction que le robot va prendre.

Pour examiner le contenu des blocs, il suffit de double-cliquer dessus. Lorsqu'un sous-bloc est ouvert, il est possible de remonter au bloc englobant à l'aide de la flèche haut de la barre d'outils (en rouge sur la figure suivante), ou de la barre de navigation (en bleu sur la figure suivante).



## Modifier l'ordre de parcours

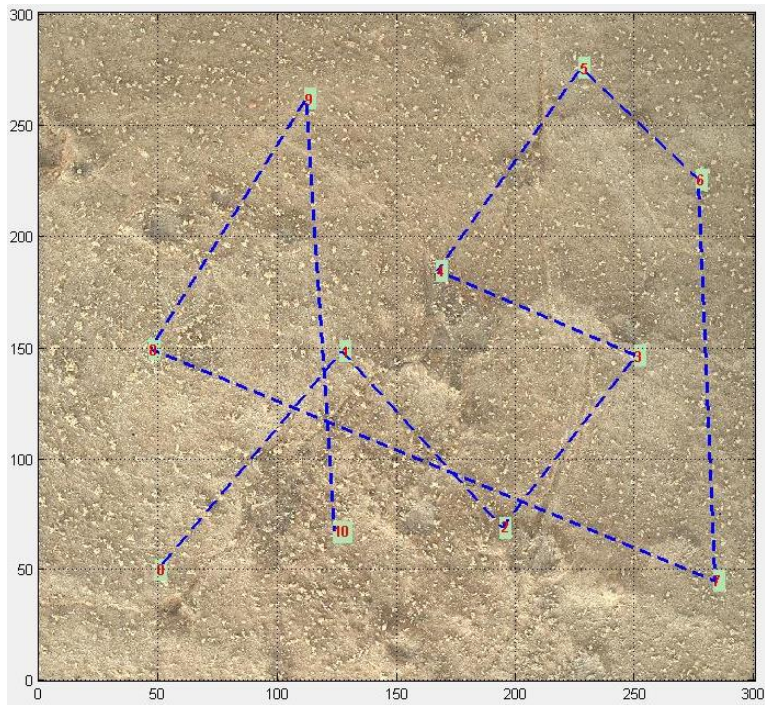
Vous pouvez graphiquement et très facilement changer l'ordre de parcours du robot à l'aide de la fonction `changeSitesOrder` (Pour plus d'informations cliquer [ici](#)). Cette fonction se trouve dans le répertoire `matlab\utilities` du projet :



**Commented [AR1]:** Je serai assez pour mettre le réordonancement des sites ici, car c'est une première chose simple à faire qui peut permettre d'appréhender le modèle, et c'est important de savoir qu'on peut le faire.

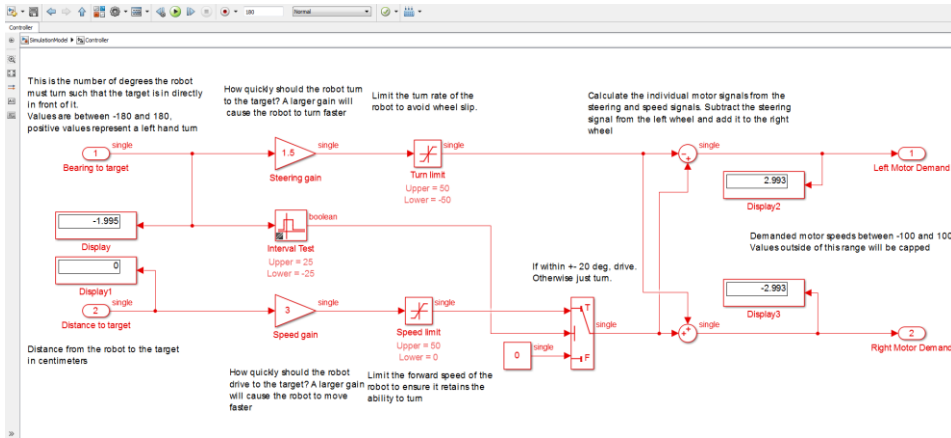
## Définir la position initiale du robot et les sites

Vous pouvez graphiquement et très facilement définir les sites à l'aide de la fonction `defineSitesPosition` (Pour plus d'informations cliquer [ici](#)). Cette fonction se trouve dans le répertoire `matlab\utilities` du projet :



### Une première modification du modèle

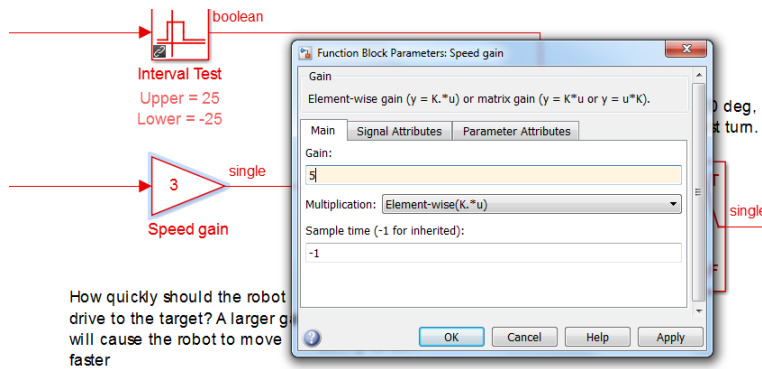
Pour débuter, le bloc controller (vert) est le plus simple à modifier. Il suffit de double-cliquer sur le bloc pour afficher son contenu. Dans le cas du bloc controller, double-cliquer dessus devrait donner un contenu similaire à la figure suivante :



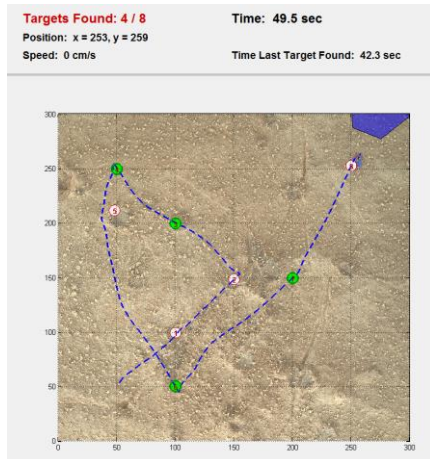
Pour débiter et se familiariser avec le modèle une première approche est de modifier les paramètres actuels, et d'observer comment le fonctionnement du robot est impacté. Ensuite, vous pouvez bien sûr complètement modifier la logique du modèle : le modèle donné est une base de départ que vous pouvez librement améliorer.

Pour modifier un paramètre, il suffit de double-cliquer sur un bloc, et utiliser la boîte de dialogue qui apparait pour changer les valeurs.

Sur ce schéma, une première expérimentation peut être d'augmenter le champ « Speed gain », en double-cliquant sur le triangle correspondant, et en augmentant la valeur comme ci-dessous.

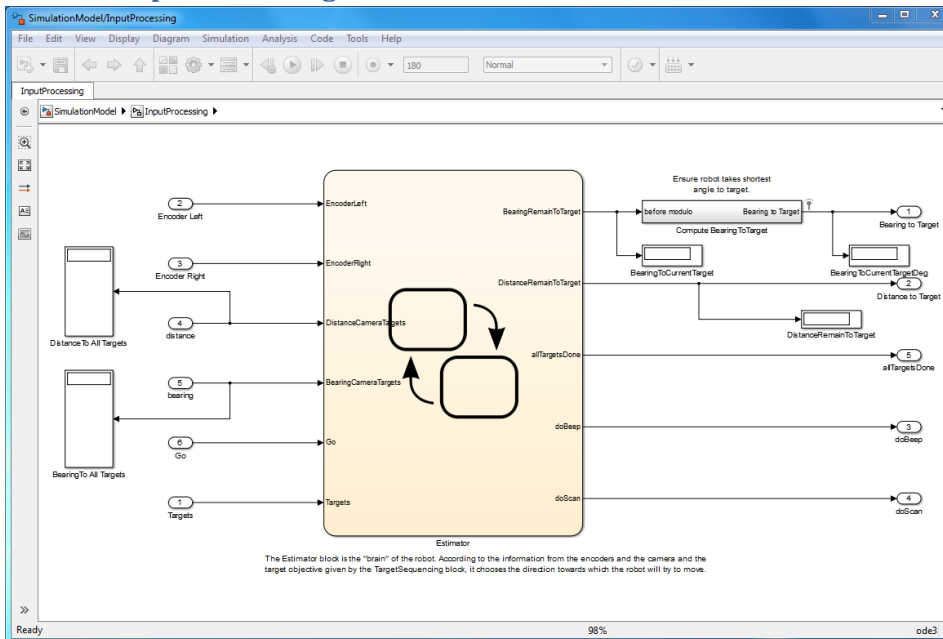


La modification effectuée, il est possible de relancer la simulation, pour obtenir le résultat ci-dessous. Le robot se déplace plus vite, mais maintenant il ne s'arrête plus correctement sur les sites, et ne trouve plus que quatre cibles. Il faut donc essayer autre chose...



Une fois que vous avez lancé la simulation et observé le comportement du robot, vous pouvez modifier le modèle, et relancer la simulation pour immédiatement observer le nouveau comportement.

## Le bloc « InputProcessing »



La logique de ce bloc est fournie par un diagramme Stateflow : le bloc « Estimator ». Ce bloc implémente la logique de parcours du robot.

Il prend en entrée :

- Les informations fournies par la caméra : liste et position des sites visibles depuis la caméra
- Les informations des encodeurs
- La liste des sites à parcourir. Sous forme de distance et d'angle par rapport au site précédent.

Les informations des encodeurs (rotation des roues gauches et droites) sont utilisées pour estimer la position actuelle du robot. Il faut cependant garder à l'esprit qu'il s'agit d'une estimation. Ainsi, si la taille réelle des roues n'est pas exactement égales à la taille des roues estimées, ou si les roues glissent ou patinent, cette estimation va présenter des erreurs qui vont s'accumuler.

Les informations de caméra sont quant à elles utilisées pour naviguer « à vue » lorsque le site est repéré. Cela permet de compenser les erreurs d'estimation de la position. Lorsque le robot ne voit pas le site, il suit la direction en utilisant les informations estimées.

## Pour finir

---

Le modèle fourni est une base de départ que vous êtes libre de complètement modifier.

Plusieurs exemples de pistes s'offrent à vous pour optimiser le modèle, de la plus simple à la plus complexe:

- Vous pouvez parcourir les sites dans l'ordre de votre choix, par exemple pour réduire la distance que devra parcourir le robot.
- Vous pouvez optimiser la navigation de votre robot, et définir par exemple la meilleure trajectoire.
- Vous pouvez aussi jouer avec les paramètres du robot, sa vitesse par exemple, ou bien décider de modifier la stratégie proposée.

Mais bien sûr, vous êtes libres d'inventer d'autres solutions. Gardez cependant à l'esprit que la puissance de calcul du robot est très limitée, et que des calculs complexes risquent de prendre trop de temps. Ainsi, il est recommandé de ne pas utiliser de nombres en double précision ou un trop grand nombre d'opérations trigonométriques.