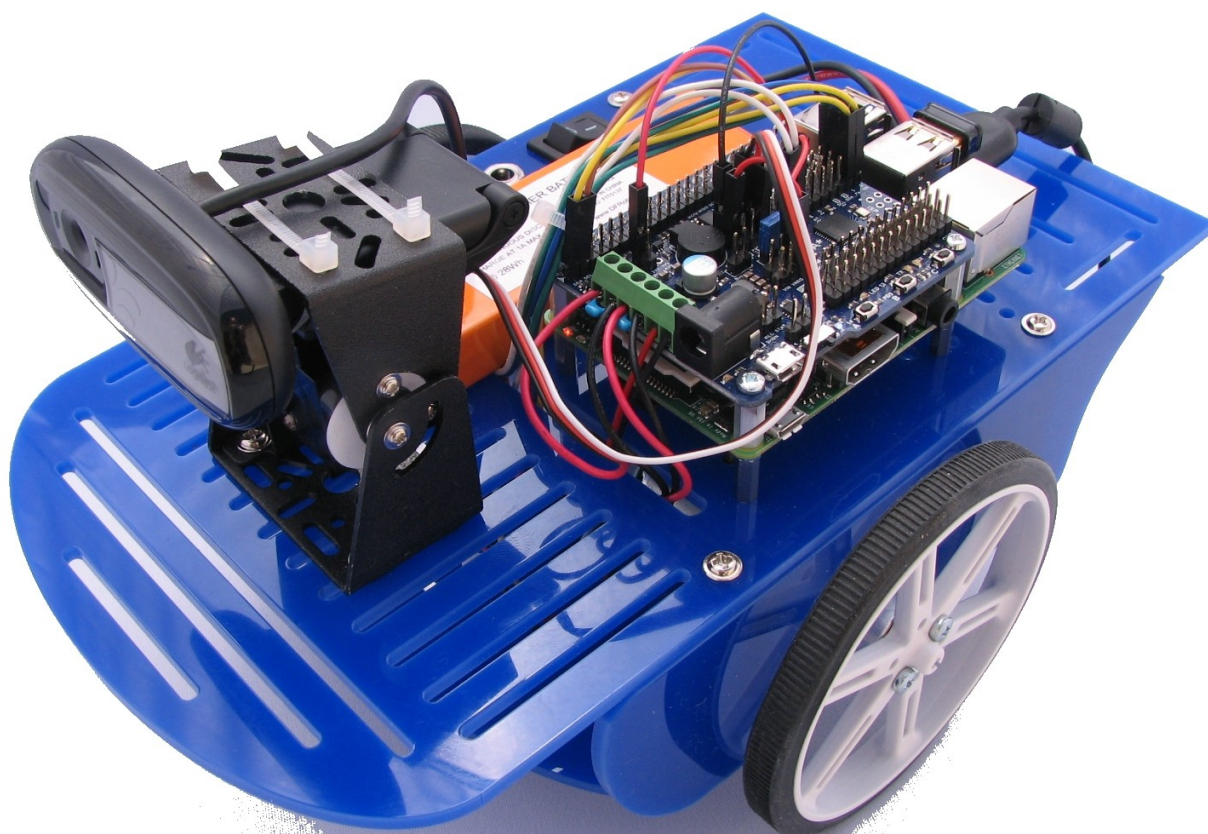




# GEEROS RASPBERRY C / PYTHON

## DOCUMENTATION COMPLÈTE



Date de dernière mise à jour : 15/11/2016

## Table des matières

---

<b>1 - Introduction.....</b>	<b><u>4</u></b>
<b>2 - Matériel inclus.....</b>	<b><u>4</u></b>
2.1 - Eléments constituant Geeros Raspberry C / Python.....	<u>5</u>
2.2 - Diagramme des échanges de signaux.....	<u>6</u>
<b>3 - Conformité.....</b>	<b><u>7</u></b>
<b>4 - Installation de l'IDE Arduino.....</b>	<b><u>8</u></b>
4.1 - Installation principale.....	<u>8</u>
4.2 - Installation du support de la carte A-Star.....	<u>8</u>
4.3 - Installation de la bibliothèque complémentaire FlexiTimer2.....	<u>9</u>
4.4 - Installation de la bibliothèque complémentaire I2Cdev.....	<u>9</u>
4.5 - Installation de la bibliothèque complémentaire MPU6050.....	<u>9</u>
4.6 - Installation de la bibliothèque complémentaire SoftwareServo.....	<u>10</u>
4.7 - Téléchargement d'un programme sur la carte A-Star.....	<u>11</u>
<b>5 - Précautions d'emploi.....</b>	<b><u>12</u></b>
5.1 - Connexions d'alimentation sur la carte A-Star.....	<u>12</u>
5.2 - Recharge de la batterie.....	<u>12</u>
5.3 - Précautions d'utilisation.....	<u>13</u>
<b>6 - Logiciels.....</b>	<b><u>14</u></b>
6.1 - Programmes C/C++.....	<u>14</u>
6.2 - Programmes Python.....	<u>15</u>
6.3 - Application Web de pilotage.....	<u>15</u>
<b>7 - Utilisation.....</b>	<b><u>16</u></b>
7.1 - Mise en route standard.....	<u>16</u>
7.2 - Mise en route spectaculaire.....	<u>17</u>
7.1 - Communication Wifi.....	<u>17</u>
7.2 - Pilotage par ordinateur.....	<u>18</u>
7.3 - Pilotage par smartphone ou tablette.....	<u>21</u>
7.4 - Télémétrie.....	<u>23</u>
7.5 - Comment arrêter Geeros ?.....	<u>26</u>
<b>8 - Fonctionnement d'un gyropode.....</b>	<b><u>27</u></b>
8.1 - Maintien en équilibre.....	<u>27</u>
8.2 - Mouvement.....	<u>28</u>

- 8.3 - Démarrage.....[28](#)
- 8.4 - Arrêt.....[29](#)
- 9 - Expériences additionnelles.....[30](#)**
  - 9.1 - Caractéristiques du moteur à courant continu avec codeur incrémental associé.....[30](#)
    - 9.1.1 - Calcul de la vitesse de rotation du moteur.....[31](#)**
    - 9.1.2 - Comptage du nombre d'impulsions.....[33](#)**
  - 9.2 - Commande des moteurs en tension (en Python).....[34](#)
  - 9.3 - Asservissement des moteurs en vitesse (en Python).....[37](#)
  - 9.4 - Commande des moteurs en tension (en C/C++).....[41](#)
  - 9.5 - Asservissement des moteurs en vitesse (en C/C++).....[44](#)
- 10 - Autres documentations disponibles.....[48](#)**
- 11 - Important.....[49](#)**

## 1 - Introduction

Geeros Raspberry C / Python est un robot gyropode ouvert et open-source, un concentré de technologie vous permettant de faire de nombreuses expériences. Il est basé sur une carte Raspberry Pi 3, différents capteurs et deux moteurs électriques permettant de le mettre en mouvement tout en assurant son maintien en équilibre.

Il intègre une liaison Wifi vous permettant de le piloter à distance à partir d'un ordinateur, d'un smartphone ou d'une tablette.

Il embarque enfin une Webcam capable de faire de la transmission vidéo en temps-réel des images prises par le robot.

Des ressources associées à Geeros Raspberry C / Python sont téléchargeables sur notre site Web à l'adresse suivante:

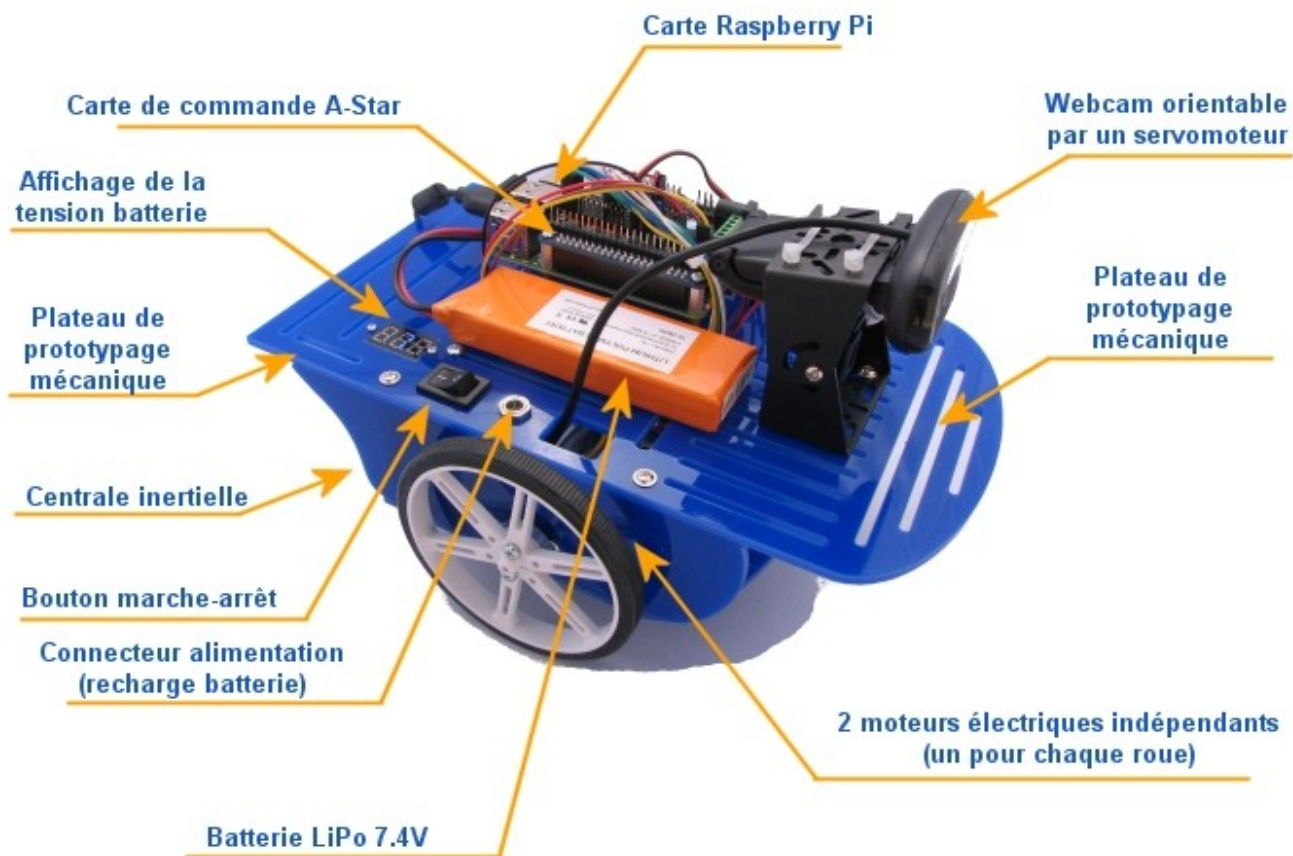
<http://www.3sigma.fr/telechargements>

## 2 - Matériel inclus

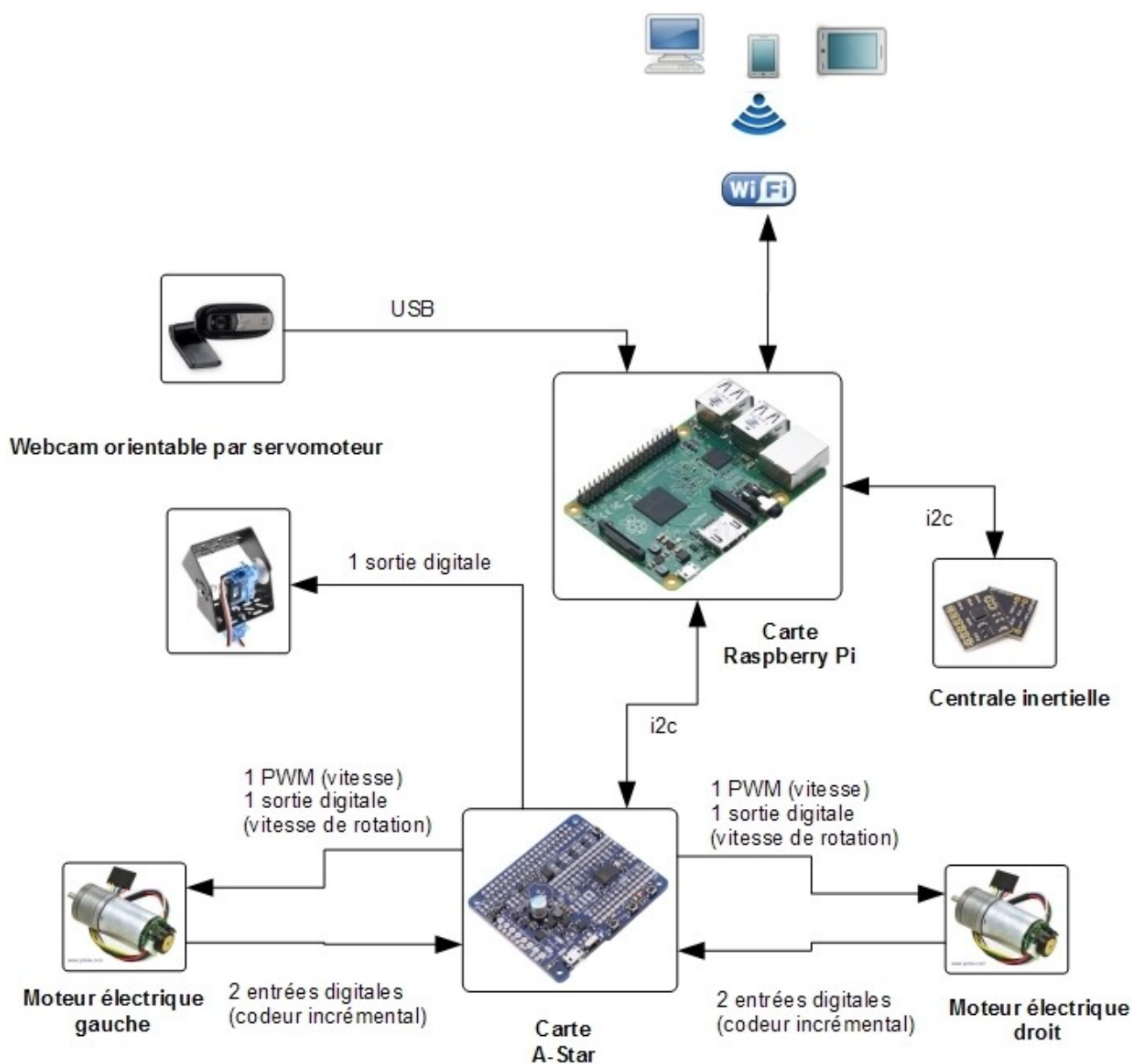
Geeros est livré monté et **fonctionnel (programmé et testé par nos soins avant la livraison)**. Il est composé des éléments suivants:

- Le robot gyropode lui-même
- 1 chargeur de batterie LiPo 8.4V, 1A
- 1 câble mini USB pour reprogrammer, éventuellement, la carte micro-contrôleur
- 1 clé allen pour visser ou dévisser les roues
- 1 adaptateur micro-SD pour sauvegarder le système de la Raspberry Pi
- 2 boules omnidirectionnelles permettant de stabiliser Geeros afin de faire des expériences sans se préoccuper de l'asservissement de verticalité

## 2.1 - Eléments constituant Geeros Raspberry C / Python



## 2.2 - Diagramme des échanges de signaux



Remarque :

Geeros Raspberry C / Python intègre une carte Pololu A-Star (compatible Arduino) pour le comptage des impulsions des codeurs incrémentaux des moteurs. En effet, la carte Raspberry Pi ne peut pas le faire d'une façon aussi performante et fiable (ce n'est pas un système temps-réel, contrairement à un micro-contrôleur dédié).

Ce manque de performance n'est pas gênant pour un robot mobile classique mais il est préjudiciable aux asservissements d'un système aussi contraignant qu'un gyropode.

La communication entre les cartes A-Star et Raspberry Pi (la première envoie à la seconde le nombre d'impulsions codeur de chaque moteur toutes les 10 ms) se fait par i2c.

## 3 - Conformité

Le robot Geeros, **dans sa configuration livrée aux clients**, est conforme à la directive 1999/EC.



## 4 - Installation de l'IDE Arduino

Geeros intègre une carte A-Star, compatible Arduino. A la livraison, cette carte est programmée pour effectuer les opérations de bas niveau (calcul de la vitesse de rotation des moteurs et commande de ces derniers) en relation avec la Raspberry Pi, qui réalise les asservissements de bas niveau.

Cependant, il est également possible de programmer l'A-Star pour que ce soit elle qui effectue la totalité du contrôle du robot. Ceci nécessite de télécharger un nouveau programme et pour ce faire, vous devez au préalable installer l'IDE Arduino et quelques bibliothèques additionnelles.

### 4.1 - Installation principale

Télécharger et installer l'IDE Arduino (<http://arduino.cc/en/Main/Software>). Il suffit de décompresser l'archive téléchargée dans le répertoire de votre choix.

### 4.2 - Installation du support de la carte A-Star

Il faut effectuer les opérations suivantes pour ajouter le support de la carte A-Star dans l'IDE Arduino :

- Télécharger et décompresser l'archive suivante dans le répertoire de votre choix : [https://www.pololu.com/file/download/a-star-2.0.0.zip?file\\_id=0J743](https://www.pololu.com/file/download/a-star-2.0.0.zip?file_id=0J743)
- Ouvrir le sous-répertoire « drivers ». Faire un clic bouton droit sur le fichier a-star.inf et sélectionner « Installer »
- Repérer l'emplacement de votre « Répertoire de croquis » en sélectionnant Fichiers → Préférences dans l'IDE Arduino. Ce répertoire est indiqué à l' « Emplacement du carnet de croquis »
- Copier dans ce répertoire de croquis le répertoire nommé « pololu », que vous trouverez dans l'archive précédemment téléchargée
- Redémarrer l'IDE Arduino
- Dans le menu Outils → Type de carte, choisir « Pololu A-Star 32u4 »
- Dans le menu Croquis → Inclure ou bibliothèque, sélectionner « Gérer les bibliothèques »
- Chercher AStar32U4
- Sélectionner l'entrée AStar32U4 trouvée et cliquer sur « Installer »
- Redémarrer l'IDE Arduino



## 4.3 - Installation de la bibliothèque complémentaire FlexiTimer2

Cette bibliothèque permet d'exécuter à cadence fixe une partie du programme Arduino.

Vous pouvez la télécharger à l'adresse suivante: <http://www.3sigma.fr/telechargements/FlexiTimer2.zip>.

Une fois téléchargée, décompressez-là dans le répertoire des bibliothèques de votre installation Arduino (typiquement, Documents\Arduino\libraries).

## 4.4 - Installation de la bibliothèque complémentaire I2Cdev

Cette bibliothèque permet de communiquer avec des composants utilisant le protocole I2C, comme le capteur accélérométrique et gyroscopique 3 axes intégré à Geeros.

Vous pouvez la télécharger à l'adresse suivante:  
<http://www.3sigma.fr/telechargements/I2Cdev.zip>.

Une fois téléchargée, décompressez-là dans le répertoire des bibliothèques de votre installation Arduino (typiquement, Documents\Arduino\libraries).

## 4.5 - Installation de la bibliothèque complémentaire MPU6050

Cette bibliothèque intègre de nombreuses fonctions permettant d'accéder aux informations mesurées par le MPU6050, qui est le capteur accélérométrique et gyroscopique 3 axes intégré à Geeros.

Vous pouvez la télécharger à l'adresse suivante:  
<http://www.3sigma.fr/telechargements/MPU6050.zip>.

Une fois téléchargée, décompressez-là dans le répertoire des bibliothèques de votre installation Arduino (typiquement, Documents\Arduino\libraries).

## 4.6 - Installation de la bibliothèque complémentaire SoftwareServo

Geeros « Lynx » est muni d'une caméra orientable dans le plan vertical grâce à un servomoteur. Il existe bien une bibliothèque « Servo » en standard dans l'environnement Arduino, mais celle-ci n'est pas compatible avec le fonctionnement sur « interruption timer » d'une partie du programme du robot, conduisant à des perturbations importantes sur le positionnement du servomoteur.

La bibliothèque « SoftwareServo » permet de remédier à ce problème en pilotant le servomoteur de façon synchronisée avec l' « interruption timer » réalisant les asservissements du robot.

Vous pouvez la télécharger à l'adresse suivante:

<http://www.3sigma.fr/telechargements/SoftwareServo.zip>.

Une fois téléchargée, décompressez-la dans le répertoire des librairies de votre installation Arduino (typiquement, Documents\Arduino\libraries).

### ATTENTION !

L'environnement Arduino doit être redémarré après l'installation des bibliothèques complémentaires.

## 4.7 - Téléchargement d'un programme sur la carte A-Star

### ATTENTION !

Pour télécharger un programme sur la carte A-Star, il est impératif de basculer au préalable le bouton marche-arrêt du robot sur la position « I ». Geeros doit par ailleurs être posé sur l'arrière.

La procédure à suivre pour la programmation est la suivante:

- Poser Geeros sur l'arrière pour qu'il ne se mette pas en route à la fin du téléchargement (en effet, le programme téléchargé démarre automatiquement à la fin de son téléchargement). Si vous modifiez les programmes livrés, vous devez vous assurer de conserver un mode de fonctionnement dans lequel le contrôle de verticalité est désactivé pour ne pas voir le robot s'enfuir sitôt terminé le téléchargement de votre nouveau programme
- Connecter le câble USB reliant l'ordinateur et la carte A-Star
- Mettre le robot sous tension en appuyant sur le bouton marche-arrêt
- Lancer le téléchargement
- Eteindre et rallumer le robot

## 5 - Précautions d'emploi

Nous insistons sur le fait que Geeros et un robot d'étude et de développement qui nécessite un certain nombre de précautions d'emploi.

### 5.1 - Connexions d'alimentation sur la carte A-Star

Il est impératif de faire très attention aux connexions de l'alimentation de la carte A-Star. Une erreur de connexion sur les bornes d'alimentation risque d'entraîner la destruction du sous-ensemble de gestion d'alimentation de la carte et de rendre celle-ci inutilisable. Le robot étant livré connecté et fonctionnel, il est préférable de ne pas modifier les branchements sur les connecteurs d'alimentation.

### 5.2 - Recharge de la batterie

Le chargeur 8.4V, 1A fourni est destiné à recharger la batterie LiPo du robot en le connectant sur le connecteur jack situé à côté du bouton de marche-arrêt. Bien que cette batterie possède une protection intégrée contre les surcharges et les sous-charges, il n'est pas conseillé de la recharger avec une alimentation pouvant fournir un courant supérieur à 1A.

Si une autre alimentation devait être utilisée pour la recharger, il est important de vérifier que sa polarité est « positif au centre du connecteur ».

Il faut compter environ 2 heures pour une demi-recharge de la batterie et 8 heures pour une recharge complète.

Les robots Geeros Raspberry C / Python possèdent par ailleurs un afficheur LCD permettant de visualiser la tension en sortie de la batterie. Lorsqu'une batterie est totalement chargée, sa tension de sortie est d'environ 8.4 V. Lorsque la tension descend au dessous de 7V, il est bon d'envisager une recharge de la batterie.

## 5.3 - Précautions d'utilisation

---

Geeros doit être utilisé dans les conditions suivantes:

- Au sol. Ne jamais l'utiliser en hauteur (sur une table, par exemple) à cause des risques de chute
- A l'intérieur. Les sols extérieurs présentent souvent des aspérités et des obstacles surdimensionnés par rapport à la taille du robot. Par ailleurs, Geeros craint l'eau et l'humidité

Geeros peut être reprogrammé à votre guise. Vous pouvez par exemple faire des programmes permettant d'étudier la commande des moteurs électriques. Mais attention: il est **fortement déconseillé** de faire des expériences de fonctionnement « rotor bloqué » avec une tension d'alimentation du moteur trop élevée. Ce type d'expérience peut générer des courants trop forts qui réduisent la durée des vie des éléments.

## 6 - Logiciels

Geeros Raspberry C / Python associe deux systèmes « intelligents » :

- Une carte A-Star (compatible Arduino) à base de micro-contrôleur ATmega32u4
- Une carte Raspberry Pi 3

Les programmes en C/C++ (à la « mode » Arduino) sont destinés à être exécutés sur la carte A-Star alors que les programmes Python sont exécutés par la carte Raspberry Pi.

Les différentes fonctionnalités du robot (commande des moteurs en tension, asservissement des moteurs en vitesse ou contrôle total du robot) peuvent être gérées par l'une ou l'autre des deux cartes. Cependant, lorsque ce sont les programmes Python de la Raspberry Pi qui sont mis en œuvre, la carte A-Star reste sollicitée pour effectuer les opérations les plus directement liées au matériel, à savoir le comptage des impulsions des codeurs incrémentaux et la génération des PWM pour la commande des moteurs électriques. Il est indispensable de charger sur l'A-Star un programme, que nous appellerons par la suite « firmware », dès que l'on souhaite utiliser les programmes Python de la Raspberry Pi.

### 6.1 - Programmes C/C++

Les programmes C/C++ se téléchargent sur la page <http://www.3sigma.fr/Telechargements-Geeros.html#raspberry>, section « Programmes de la carte A-Star ».

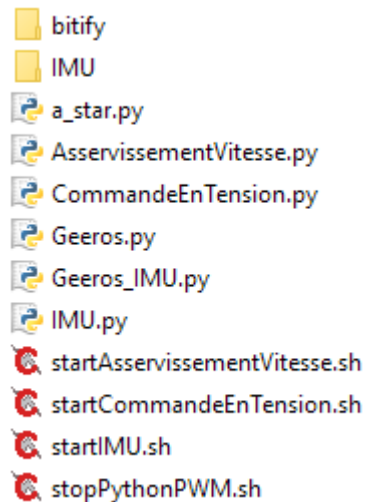
Ces programmes sont les suivants :

- Firmware de communication avec la carte Raspberry Pi : doit impérativement être chargé si on souhaite exécuter les programmes Python de la Raspberry Pi. En effet, ce firmware gère les entrées / sorties bas niveau et la communication avec la Raspberry Pi.
- Asservissement de verticalité et de mouvement : permet d'obtenir un asservissement de verticalité du robot, exécuté entièrement par la carte A-Star
- Commande des moteurs en tension (exécutée uniquement par la carte A-Star)
- Asservissement des moteurs en vitesse (exécutée uniquement par la carte A-Star)

## 6.2 - Programmes Python

Un certain nombre de programmes Python sont fournis pour piloter le robot, la tension des moteurs et la vitesse de ces derniers.

Ces programmes se trouvent dans le répertoire `/root/programmes_python` de la carte Raspberry Pi:



## 6.3 - Application Web de pilotage

Le pilotage du robot via un ordinateur ou un smartphone est une application Web dont le code source se trouve dans le répertoire `/www` de la carte Raspberry Pi.

## 7 - Utilisation

### 7.1 - Mise en route standard

Le programme pré-chargé dans le robot à la livraison lui permet de tenir en équilibre sur ses deux roues et de recevoir des consignes de mouvement (transmises par liaison Wifi) provenant d'un ordinateur, d'un smartphone ou d'une tablette.

La procédure à suivre est simple:

- poser le robot au sol sans l'allumer, penché vers l'arrière
- mettre le robot sous tension en basculant l'interrupteur sur « I »
- attendre environ une minute que la led verte de la Webcam s'allume
- redresser doucement le robot : il se maintient alors en équilibre même en cas de perturbations
- pour piloter le robot, se connecter en Wifi sur le réseau de SSID « Geeros ». Le mot de passe Wifi est « geerosraspberry »
- sur votre ordinateur, naviguer à l'adresse <http://192.168.0.199>
- choisir le pilotage par ordinateur
- vous pouvez alors utiliser le joypad virtuel pour donner des consignes de mouvement au robot

**Remarque :**

Si le robot est parfaitement équilibré quand il est éteint (ce qui est normalement le cas lorsque vous le recevez), l'activation de l'asservissement de verticalité s'accompagne d'un mouvement de translation quasi-nul. En revanche, si le robot est naturellement déséquilibré (ce qui peut arriver si vous rajoutez de nouveaux capteurs, par exemple), l'activation de l'asservissement de verticalité s'accompagnera d'un mouvement de translation, le temps que le robot détermine automatiquement l'angle d'inclinaison lui permettant de se maintenir en équilibre.



## 7.2 - Mise en route spectaculaire

---

Il est possible de démarrer le robot lorsqu'il est posé sur l'avant à l'arrêt. Dans ce cas, après une attente d'environ 20 secondes pour que tout le système démarre, le robot se redresse et retrouve tout seul son équilibre.

Ce mode de démarrage est très spectaculaire mais attention ! Il faut prévoir quelques dizaines de centimètres sans obstacles pour qu'il ait la place d'avancer pour se redresser.

Le reste de la procédure est identique à ce qui présenté dans la section précédente:

- pour piloter le robot, se connecter en Wifi sur le réseau de SSID « Geeros ». Le mot de passe Wifi est « geerosraspberry »
- sur votre ordinateur, naviguer à l'adresse <http://192.168.0.199>
- choisir le pilotage par ordinateur
- vous pouvez alors utiliser le joypad virtuel pour donner des consignes de mouvement au robot

## 7.1 - Communication Wifi

---

Geeros embarque une carte Raspberry Pi 3 avec Wifi intégré, lui permettant d'être piloté par certains appareils possédant une connectivité Wifi.

Lorsqu'il est démarré et que la led verte de la caméra est allumée, la connexion est disponible. Geeros se comporte alors comme point d'accès Wifi autonome: un nouveau réseau Wifi devient alors visible sur vos appareils, avec le SSID « Geeros ». Ce réseau est sécurisé, le mot de passe est « geerosraspberry ». Notez les points suivants:

- Un appareil donné ne peut pas en général se connecter en même temps à deux réseaux Wifi différents: la connexion au réseau Geeros entraînera la déconnexion du réseau auquel vous étiez éventuellement connecté au préalable. Vous n'aurez alors plus accès à Internet sur votre appareil, sauf si celui-ci est
  - un ordinateur connecté en parallèle en filaire (liaison Ethernet). Attention cependant: si ce réseau Ethernet utilise la même plage d'adresse (192.168.0.xxx), vous devrez probablement le désactiver pour vous connecter à l'adresse 192.168.0.199 de Geeros en Wifi
  - un smartphone ou une tablette connecté à un réseau 3G en parallèle

- Geeros fonctionne comme un point d'accès « maître ». Il ne peut pas, dans sa configuration de base, se connecter comme client du réseau Wifi de votre organisation
- L'adresse de Geeros est fixée à 192.168.0.199 (pour la modifier, voir la documentation « développeur »). Pour que votre appareil puisse accéder au robot, il doit accepter l'adressage automatique (adresse fournie par serveur DHCP). Si votre appareil n'accepte pas l'adressage automatique, cela signifie que vous lui avez donné une adresse IP fixe. Dans ce cas, celle-ci doit commencer par 192.168.0. (elle doit donc être de la forme 192.168.0.xxx). Si ce n'est pas le cas, vous ne pourrez pas communiquer avec Geeros

## 7.2 - Pilotage par ordinateur

Pour pouvoir piloter Geeros en Wifi grâce à votre ordinateur, les conditions suivantes doivent être respectées:

- Votre ordinateur doit posséder une interface Wifi. C'est le cas de la quasi-totalité des ordinateurs portables modernes ; si vous utilisez un ordinateur de bureau, vous devrez probablement connecter une « clé Wifi » à ce dernier pour lui apporter ce mode de communication
- Vous devez avoir rejoint le réseau Wifi « Geeros » (voir section précédente)
- Vous devez utiliser un navigateur Web récent

Si toutes ces conditions sont remplies, la connexion de votre navigateur à l'adresse 192.168.0.199 conduira à la page suivante:



Trois choix sont alors possibles:

- Pilotage par ordinateur (voir ci-dessous)
- Pilotage par smartphone: si vous être connecté avec un ordinateur, cliquer sur cette icône ne vous apportera rien car votre ordinateur ne possède a priori pas d'accéléromètre (utilisé pour le pilotage avec les appareils mobiles). Cependant, si la détection automatique du type d'appareil ne fonctionne pas, l'application Web pourrait ne pas reconnaître votre smartphone ou tablette et faire pointer son navigateur sur ce tableau de bord. Le clic sur « Pilotage par smartphone » permet alors à votre appareil mobile d'accéder à l'application de pilotage qui lui est dédiée
- Aide: résumé des 2 points précédents pour ceux qui ne souhaitent pas lire la présente documentation

Pour piloter Geeros avec votre ordinateur, vous devez bien sûr cliquer sur la première icône. Vous obtenez alors la page suivante:

**Geeros** Pilotage par ordinateur

**Mouvement** **Camera**

Consigne de vitesse longitudinale: 0 cm/s  
Consigne de vitesse de rotation: 0 deg/s  
Angle d'élévation de la caméra: 45 degrés

**Asservissement de vitesse**

**P:** 1.0  
**I:** 1.0

1.0 1.0 1.0

www.geeros.com

Cette interface présente différents éléments

- Le joystick « Mouvement » permet de donner:
  - une consigne de vitesse de rotation uniquement, lorsque le centre de la boule jaune se trouve dans la moitié supérieure du disque gris
  - une consigne de vitesse de rotation et une consigne de vitesse de translation lorsque le centre de la boule jaune se trouve dans la moitié supérieure de l'anneau bleu
  - une consigne de vitesse de translation (négative) uniquement, lorsque le centre de la boule jaune se trouve dans la moitié inférieure de l'anneau bleu
  - aucune consigne lorsque le centre de la boule jaune se trouve dans la moitié inférieure du disque gris
- Le curseur « Caméra » permet de modifier l'orientation de la caméra. L'angle correspond directement à la commande du servomoteur. La position initiale de ce dernier est de 45°, ce qui correspond à une caméra horizontale. Il est possible de faire varier l'angle du servomoteur entre 15° et 75°
- Les curseurs « P » et « I » permettent de modifier les paramètres du régulateur Proportionnel-Intégral de l'asservissement de vitesse du robot. Une diminution de « P » diminue la rapidité de l'asservissement. Une diminution de I dégrade la précision. Ces paramètres varient entre 0.5 et 2.0.
- A droite de l'écran, vous pouvez voir la retransmission vidéo en temps-réel de la webcam embarquée (pointée ici sur l'écran de l'ordinateur, d'où l'effet d' « infini »).

## 7.3 - Pilotage par smartphone ou tablette

Pour pouvoir piloter Geeros en Wifi grâce à votre smartphone ou votre tablette (nous utiliserons par la suite le nom générique « appareil »), les conditions suivantes doivent être respectées:

- Votre appareil doit posséder une interface Wifi
- Il doit posséder un accéléromètre
- Il doit être orienté en mode « paysage », comme ceci:

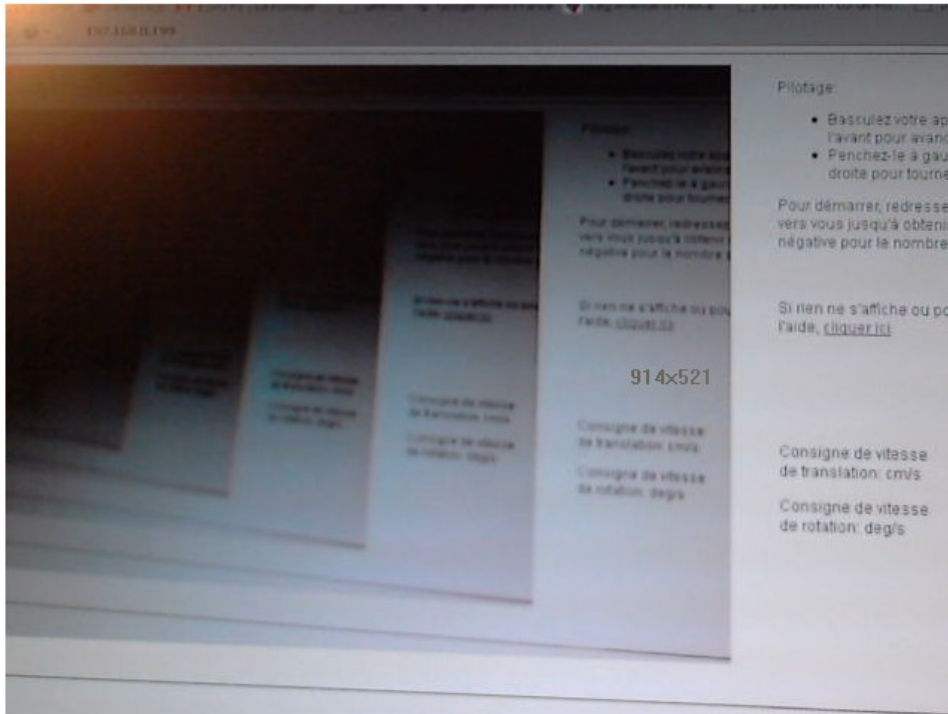


- Vous devez avoir rejoint le réseau Wifi « Geeros » (voir plus haut)
  - sur iPhone, iPad et iPod: votre appareil doit être sous iOS 4.2 ou supérieur
  - sur Android: vous devez utiliser un navigateur Web récent, de type Firefox ou Opera mobile
  - les autres systèmes n'ont pas encore été testés mais peuvent malgré tout fonctionner

**Remarque :**

Lorsque Geeros a été développé, le choix d'une application Web comme interface de pilotage a semblé intéressant en terme de portabilité. Cependant, le monde des navigateurs mobiles évolue très vite et pas toujours dans le bon sens: on nous a en effet rapporté des problèmes de performances sur des versions très récentes. Si vous avez des soucis quelconques, n'hésitez pas à nous en faire part ([contact@geeros.com](mailto:contact@geeros.com)). Il est possible que nous développiions à l'avenir des applications natives (et donc malheureusement moins portables) si certains problèmes devenaient trop récurrents ou insolubles avec les interfaces Web.

Lorsque vous connectez le navigateur Web de votre appareil à l'adresse 192.168.0.199, vous obtenez tout d'abord la page suivante:



Pilotage:

- Basculez votre appareil vers l'avant pour avancer
- Penchez-le à gauche ou à droite pour tourner

Pour démarrer, redressez l'appareil vers vous jusqu'à obtenir une valeur négative pour le nombre suivant:

Si rien ne s'affiche ou pour avoir de l'aide, [cliquer ici](#).

Consigne de vitesse de translation: cm/s

Consigne de vitesse de rotation: deg/s

Cette page permet de tester si votre appareil est compatible. Si tout se passe bien, vous devez voir, à droite de la caméra, un nombre changer en fonction de l'inclinaison de votre appareil. Ce nombre correspond à l'accélération de la pesanteur mesurée par votre appareil, normalisée entre 0 et 1. Si vous ne voyez pas ce nombre, cela signifie:

- que votre appareil ne possède pas d'accéléromètre
- ou bien qu'il n'est pas compatible avec cette application Web de pilotage (iPhone, iPad ou iPod avec iOS < 4.2, par exemple)
- ou bien que le navigateur utilisé n'est pas compatible

Dans ce cas, merci de nous contacter à l'adresse [contact@geeros.com](mailto:contact@geeros.com) en nous indiquant les caractéristiques précises du smartphone ou de la tablette que vous utilisez. Une solution personnalisée est peut-être possible.

Si vous voyez l'accélération évoluer en fonction de l'inclinaison de votre appareil, tout va bien. Avant d'accéder au pilotage à proprement parler, vous devez activer l'application. C'est très simple, il suffit de redresser votre appareil jusqu'à obtenir une accélération nulle. Une fois ceci fait, l'interface change légèrement.

L'accélération n'est plus visible, seules restent les consignes de vitesse de translation et de rotation (en plus de la transmission vidéo temps-réel de la Webcam, à gauche de l'écran). Piloter Geeros est alors très facile:

- Lorsque vous basculez votre appareil vers l'avant, vous modifiez la consigne de vitesse de translation (entre 0 et 50 cm/s)
- Lorsque vous inclinez votre appareil sur la gauche ou sur la droite, vous modifiez la consigne de vitesse de rotation (entre -180 et 180 deg/s), comme avec un volant. Si l'inclinaison est trop forte à gauche ou à droite, l'application détecte que votre appareil est en mode portrait et annule les consignes

**Remarque :**

Dans ce mode de pilotage par smartphone ou tablette, il est impossible de faire reculer le robot car il est impossible de lui donner une consigne de vitesse négative.

## 7.4 - Télémétrie

---

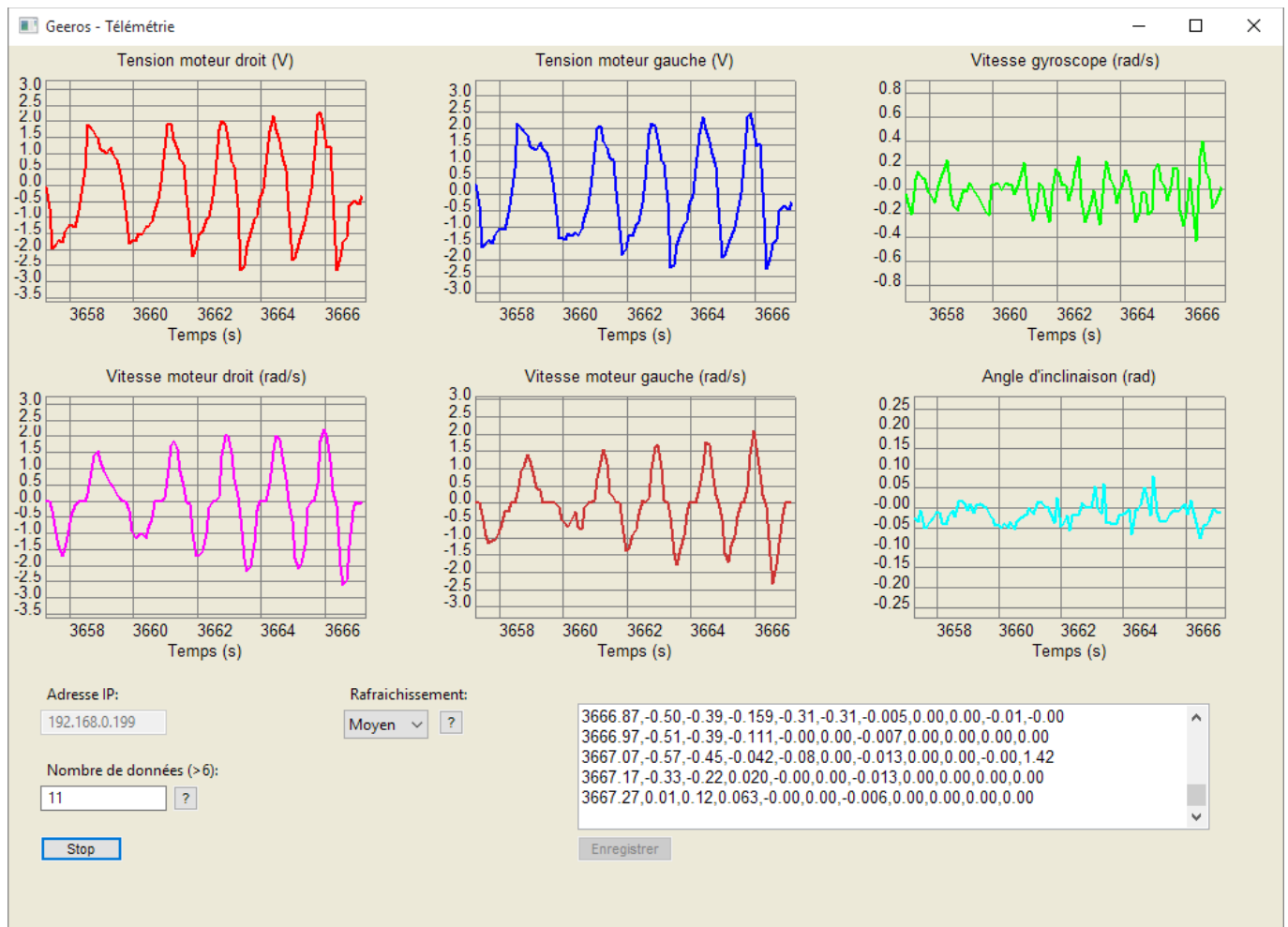
Une application de télémétrie permet d'afficher sur un ordinateur 6 données mesurées sur le robot en fonctionnement, à distance, via la liaison Wifi. Elle peut se télécharger à l'adresse suivante:

<http://www.3sigma.fr/Telechargements-Geeros.html>

Son nom est de la forme GeerosTelemetrie-Linux\_x.y.zip (x.y correspond au numéro de version du programme).

Pour l'installer, il suffit de décompresser l'archive dans le répertoire de votre choix. Pour l'exécuter, double-cliquer sur GeerosTelemetrie-Linux.exe. Si Geeros est en mode gyropode, vous pourrez visualiser (et éventuellement enregistrer) ces données.

Voici une capture d'écran de l'interface de l'application de télémétrie:



Les différents éléments sont les suivants (de haut en bas)

- Courbes: l'interface permet de visualiser
  - la tension de commande des moteurs droit et gauche (V, en rouge et bleu)
  - les vitesses mesurées des moteurs (rad/s, en magenta et orange)
  - la vitesse de rotation par rapport à la verticale, mesurée par le gyroscope (rad/s, en vert)
  - l'angle d'inclinaison mesuré par l'accéléromètre (rad, en cyan)
- Zones de sélection de l'adresse IP sur laquelle est connecté votre robot. Les valeurs affichées au lancement du programme sont les valeurs par défauts fonctionnant avec Geeros lorsque vous le recevez.
- Nombre de données : correspond au nombre de données envoyées par le programme Geeros.py. Ce nombre doit être impérativement strictement supérieur à 6 et ne possède pas de limite supérieure théorique. Cependant :
  - le temps doit être impérativement la première donnée
  - seules les 7 premières données sont tracées (le temps en abscisse + les 6 suivantes en ordonnées). En revanche, toutes les données sont affichées dans la zone de texte
  - plus le nombre de données est important, plus la liaison sera chargée



- Bouton « Acquisition ». Cliquer dessus pour voir les courbes se mettre en jour en temps-réel et défiler des valeurs numériques dans la zone d'affichage à droite
- « Rafraîchissement »: ceci pilote la fréquence de rafraîchissement des courbes. En fonction de la vitesse de votre ordinateur, vous pouvez choisir parmi les 4 valeurs « Minimum », « Lent », « Moyen » et « Rapide ». Plus la fréquence de rafraîchissement est élevée, moins le tracé est saccadé. Mais si votre ordinateur n'est pas très rapide, vous risquez d'observer un retard entre les consignes et l'affichage. Dans ce cas, il faut choisir un rafraîchissement plus lent
- Bouton « Enregistrer »: il permet d'enregistrer les valeurs numériques affichées dans la zone de « log » vers un fichier texte pour une utilisation ultérieure avec n'importe quel logiciel permettant de lire des données séparées par des virgules dans un fichier texte.  
**Attention** : ce bouton n'est actif qu'après une séquence de mesure. Il est grisé le reste du temps (au démarrage du programme et pendant une séquence de mesure)

## 7.5 - Comment arrêter Geeros ?

---

Trois options sont possibles:

- Vous souhaitez l'arrêter totalement: il suffit de basculer le bouton de marche-arrêt sur « 0 »
- Vous souhaitez arrêter de programme d'asservissement de verticalité mais pas arrêter totalement le robot pour continuer à communiquer en Wifi: il suffit d'exécuter la commande :  
`/root/programmes_python/stopPythonPWM.sh`
- Vous souhaitez désactiver temporairement l'asservissement de verticalité sans l'arrêter: il suffit de le basculer vers l'arrière jusqu'à l'arrêt de la rotation des roues. Vous pouvez ensuite le poser, toujours sur l'arrière, il ne démarrera pas tant que vous ne le redresserez pas à la verticale

## 8 - Fonctionnement d'un gyropode

Ce chapitre décrit le principe de fonctionnement d'un gyropode, qu'il est utile de connaître pour comprendre son comportement.

### 8.1 - Maintien en équilibre

Deux capteurs sont essentiels pour le maintien en équilibre: un accéléromètre et un gyroscope. Geeros est équipé d'un capteur qui communique avec la carte Raspberry Pi via une interface I2C. Ce capteur combine un accéléromètre 3 axes et un gyroscope 3 axes. Seul un axe accélérométrique et un axe gyroscopique sont utilisés dans l'asservissement de verticalité.

Un accéléromètre 3 axes permet de mesurer l'accélération du gyropode **plus** l'accélération de la pesanteur dans 3 axes orthogonaux liés au robot. L'algorithme de maintien en équilibre utilise la mesure de l'accélération longitudinale: en effet, quand le gyropode est incliné sans mouvement, l'accéléromètre mesure la pesanteur multipliée par l'angle d'inclinaison:

$$\text{acc} = g \cdot \sin(\theta)$$

La valeur de la pesanteur ( $g$ ) étant connue ( $9.81 \text{ m/s}^2$ ) et l'angle d'inclinaison étant en général faible ( $\sin(\theta) \sim \theta$ ), cela nous permet d'obtenir:

$$\theta = \text{acc}/g$$

Mais la connaissance de l'angle d'inclinaison n'est pas suffisante. En effet, deux cas de figure peuvent se présenter:

- le gyropode est incliné et est en train de tomber (l'inclinaison augmente)
- le gyropode est incliné mais est en train de se redresser (l'inclinaison diminue)

Dans le premier cas, il faudra agir plus fortement que dans le second cas. La distinction entre les deux cas se fait grâce à la mesure de la vitesse de chute (ou de redressement) fournie par un des axes (celui qui est parallèle à l'axe des roues) de mesure du gyroscope.

Avec ces deux mesures (1 axe accélérométrique et 1 axe gyroscopique), nous possédons en théorie assez d'informations pour maintenir correctement le gyropode à la verticale. Cependant, nous avons fait précédemment l'hypothèse d'un gyropode incliné sans mouvement pour la mesure de l'accélération. Or, dans le cas général le gyropode est en mouvement et peut avoir une accélération longitudinale qui va se superposer à l'accélération de la pesanteur:

$$\text{acc\_mesurée} = g \cdot \sin(\theta) + \text{acc\_longitudinale}$$

Cette accélération longitudinale étant inconnue, on ne peut pas en théorie remonter à l'angle d'inclinaison. Pour cette raison, l'asservissement de verticalité de Geeros intègre un filtre permettant d'estimer l'angle d'inclinaison à partir des deux mesures de l'accéléromètre et du gyroscope.

**Remarque :**

Geeros est capable de se maintenir en équilibre même s'il n'est pas parfaitement équilibré en « statique » (lorsqu'il est éteint). Dans ce cas, il penchera vers l'avant (même à l'arrêt) s'il y a trop de poids sur l'arrière et vice-versa.

## 8.2 - Mouvement

---

Avant tout, il est important de bien comprendre comment fonctionne un gyropode: lorsqu'il est en mouvement, il est penché en avant dans le sens de la marche pour maintenir son équilibre. En effet, la force de motorisation ne s'applique pas sur le centre de gravité du robot mais au point de contact avec le sol. La force de motorisation engendre donc un couple de rotation du gyropode autour de son centre d'inertie et a tendance à le faire tomber. Pour contrer ce couple de chute, le gyropode asservi se penche en avant pour utiliser la pesanteur.

Le gyropode peut donc rester à l'équilibre (mais pas à la verticale) en mouvement lorsque le couple de rotation généré par les moteurs est compensé par le couple de rotation généré par la pesanteur. Il se déplace alors penché dans le sens de la marche.

## 8.3 - Démarrage

---

Comme indiqué précédemment, Geeros est penché vers l'avant lorsqu'il se déplace. Or, il est vertical à l'arrêt. La phase de démarrage doit permettre de passer d'un état à l'autre (tout se fait automatiquement). Cette phase permettant de passer de l'état « vertical à l'arrêt » à l'état « penché vers l'avant en mouvement » est la suivante:

- Les moteurs vont tout d'abord tourner dans le sens inverse de la marche
- Cela a pour effet de faire pencher le gyropode vers l'avant
- Les moteurs tournent ensuite dans le sens de la marche, leur vitesse de rotation et l'angle d'inclinaison se compensant mutuellement pour éviter la chute

Nous nous trouvons donc en présence du paradoxe suivant: pour avancer, un gyropode doit d'abord (légèrement) reculer.

**IMPORTANT !**

Si vous démarrez Geeros collé à un mur à l'arrière, il ne pourra par conséquent pas démarrer puisqu'il ne pourra pas initialement reculer !

## 8.4 - Arrêt

---

Nous avons vu précédemment que lorsque Geeros est en mouvement, il est penché en avant dans le sens de la marche pour maintenir son équilibre. Quand il doit s'arrêter et retrouver la verticale, il doit transitoirement se pencher vers l'arrière, dans le sens contraire au mouvement, pour freiner (sinon, il chute vers l'avant). La seule façon de réaliser ça est de faire en sorte que la base du gyropode aille transitoirement plus vite que son centre de gravité. Par conséquent, pour s'arrêter, le gyropode doit tout d'abord accélérer !

S'il roule déjà à la vitesse maximale permise par les moteurs, il ne pourra plus accélérer et sera donc incapable de s'arrêter. La vitesse du robot est limitée à 50 cm/s dans les différentes applications de pilotage, ce qui permet de conserver aux moteurs une réserve d'accélération suffisante pour que ceux-ci puisse réaliser un arrêt du robot.

Une dernière conséquence est que l'arrêt immédiat d'un robot gyropode n'est pas possible. Si on coupe brutalement l'alimentation des moteurs, le robot va chuter, entraîné par sa vitesse.

## 9 - Expériences additionnelles

Geeros est un système pédagogique multi-expériences qui permet non seulement d'étudier de près le fonctionnement d'un gyropode, mais également d'autres aspects plus spécifiques, comme la commande de moteurs électriques.

### ATTENTION !

Veillez à sécuriser le robot lorsque vous réalisez les expériences décrites par la suite, pour qu'il ne risque pas de tomber d'une table, par exemple.

### 9.1 - Caractéristiques du moteur à courant continu avec codeur incrémental associé

Ce robot embarque deux moteurs à courant continu 6V, de rapport de réduction 34:1, avec codeur incrémental 48 CPR (Counts Per Revolution).

Les équations d'un moteur sont les suivantes:

$$\frac{d}{dt} \omega_m(t) = \frac{\text{ratio} K i_m(t) - d \omega_m(t)}{J \cdot \text{ratio}^2}$$

$$\frac{d}{dt} i_m(t) = \frac{V(t) - R i_m(t) - K \cdot \text{ratio} \omega_m(t)}{L}$$

Avec :

- R : résistance électrique interne: 3.0 Ohms
- L : inductance des enroulements: 90 mH
- J : moment d'inertie du rotor:  $5 \cdot 10^{-6}$  kg.m<sup>2</sup>
- K : constante de couple = constante de fem: 0.01 N.m/A
- d : coefficient de frottement visqueux: 0.005 N.m.s/rad
- $\omega_m$  : vitesse de rotation de l'arbre de sortie du réducteur (rad/s)
- $i_m$  : courant dans le moteur (A)
- V : tension d'alimentation (V)

Ces paramètres ont été identifiés à partir d'un essai de réponse d'un moteur à un échelon de tension.

La fonction de transfert entre l'entrée  $V(t)$  et la sortie  $\omega_m(t)$  est la suivante :

$$\frac{K_{ratio}}{JL_{ratio}^2 s^2 + (JR_{ratio}^2 + Ld) s + K^2_{ratio} + R d}$$

La fonction de transfert entre l'entrée  $V(t)$  et la sortie  $i_m(t)$  est la suivante :

$$\frac{J_{ratio}^2 s + d}{JL_{ratio}^2 s^2 + (JR_{ratio}^2 + Ld) s + K^2_{ratio} + R d}$$

### 9.1.1 - Calcul de la vitesse de rotation du moteur

Le codeur incrémental fournit deux signaux carrés en quadrature, comme sur la capture ci-dessous:



Ces deux signaux permettent de mesurer à la fois la vitesse et le sens de rotation. La mesure de la vitesse se fait simplement en comptant le nombre d'impulsions pendant un temps fixe.

Les données du problème sont les suivantes :

- Le codeur est fixé à l'arbre moteur et non pas à l'arbre de sortie du réducteur (celui utilisé pour l'entraînement). Le rapport de réduction étant 34:1, l'arbre moteur fait 34 tours lorsque l'arbre « principal » en fait 1
- Le codeur génère 48 impulsions à chaque fois qu'il fait un tour. Cependant, la carte Arduino Pro Mini qui gère le comptage des impulsions ne dispose que de deux lignes d'interruption (une pour chaque moteur). Par conséquent, on ne prend en compte qu'une interruption sur 2
- La cadence d'échantillonnage utilisée pour l'asservissement sera de 0.01 s

Par conséquent, lorsque l'arbre principal fait un tour, le codeur génère :  
 $34 * 12 = 816$  impulsions.

Si N est le nombre d'impulsions comptées en 0.01 s, la vitesse est (en rad/s, l'unité standard, sachant qu'un tour fait  $2*\pi$  radians) :  
 $2*\pi*N/(0.01*816)$

**ATTENTION !**

Bien que le codeur soit placé sur l'arbre moteur, le calcul ci-dessus donne la vitesse en sortie du réducteur.

Un point très important concerne la résolution de la mesure, c'est-à-dire la plus petite valeur qu'il est possible de calculer. La formule est la suivante (en rad/s) :

$$2*\pi/(Ts*CPR*ratio)$$

avec :

- Ts : cadence d'échantillonnage
- CPR : nombre d'impulsions par tour du codeur
- ratio : rapport de réduction du moteur

Dans notre cas de figure, la résolution serait la suivante

$$2*\pi/(0.01*816) = 0,77 \text{ rad/s}$$

Enfin, pour l'améliorer, nous faisons une moyenne sur les 10 derniers échantillons. Ceci introduit un retard (à prendre en compte dans les asservissements) mais permet d'avoir une résolution de l'ordre de 0.08 rad/s.



## 9.1.2 - Comptage du nombre d'impulsions

Dans le cas du robot Geeros Raspberry C / Python, le comptage des impulsions revient à compter le nombre de fronts montants du signal jaune représenté sur l'image ci-dessus. Pour ce faire, la seule méthode viable consiste à brancher ce signal (le fil jaune du codeur utilisé) sur une des deux entrées « interruption » de la carte A-Star. Le fil blanc est branché sur une entrée digitale classique et es deux derniers fils (bleu et vert) seront respectivement branchés sur le 5 V et sur la masse de la carte A-Star.

L'intérêt d'une ligne d'interruption est qu'elle permet, comme son nom l'indique, d'interrompre le déroulement des calculs sur le processeur pour effectuer un traitement spécifique, en l'occurrence la mise à jour du compteur d'impulsions, avant de rendre la main à la boucle principale.

La seule « difficulté » est de savoir s'il faut incrémenter ou décrémenter le compteur dans le traitement de l'interruption. Il suffit pour cela d'observer les courbes ci-dessus, obtenues alors que le moteur tourne dans le sens positif. On constate que:

- Lorsque la voie A (en jaune) passe au niveau haut, la voie B (en bleu) est au niveau bas
- Lorsque la voie A passe au niveau bas, la voie B est au niveau haut

Quand le moteur tourne dans le sens positif, lors d'une interruption sur la voie A, les niveaux de A et B sont donc inversés. Pour connaître le sens de rotation du moteur, il faut donc mesurer le niveau de B lorsque l'interruption survient.

## 9.2 - Commande des moteurs en tension (en Python)

Cette expérience permet de changer la vitesse de rotation des moteurs en appliquant une tension variable par l'intermédiaire d'une application qui s'exécute sur votre ordinateur.

Le programme Python gérant cette expérience se nomme `CommandeEnTension.py`. Il est situé dans le répertoire `/root/programmes_python`.

Il comporte de nombreux commentaires permettant de comprendre facilement son fonctionnement.

Le principe de pilotage des moteurs consiste à envoyer des signaux PWM sur le shield de commande moteur connecté à la carte Raspberry Pi afin de faire varier la tension d'alimentation des moteurs. Cette tension peut être modifiée interactivement via une application qui s'exécute sur votre ordinateur. Elle peut se télécharger à l'adresse suivante:

<http://www.3sigma.fr/Telechargements-Geeros.html>

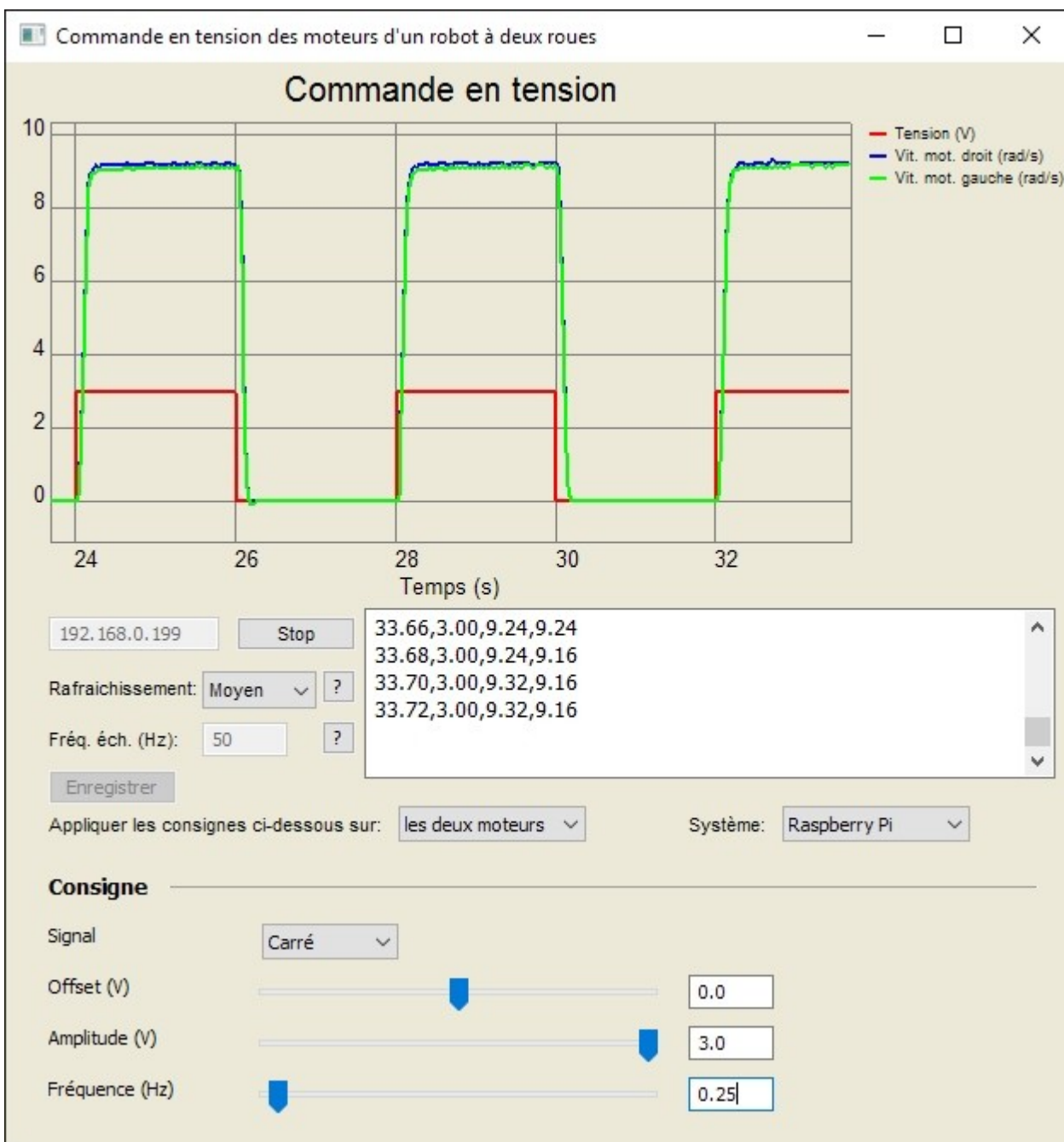
Son nom est de la forme `InterfaceReseauCommandeDeuxMoteursEnTension_x.y.zip` (x.y correspond au numéro de version du programme).

Pour l'installer, il suffit de décompresser l'archive dans le répertoire de votre choix. Pour l'exécuter, double-cliquer sur `InterfaceReseauCommandeDeuxMoteursEnTension.exe`.

Pour piloter la tension d'alimentation (et donc la vitesse, mais sans asservissement de cette dernière) des moteurs depuis votre ordinateur, il vous suffit de suivre les étapes suivantes :

- mettre le système sous tension en positionnant le bouton marche-arrêt sur « I »
- se connecter au système en Wifi à l'adresse 192.168.0.199
- lancer l'application « `InterfaceReseauCommandeDeuxMoteursEnTension` »
- entrer l'adresse IP du robot dans la zone correspondante
- sélectionner le système « Raspberry Pi » sous la zone blanche
- cliquer sur « Connexion ». L'application démarre alors à distance le programme Python de commande des moteurs sur la carte Raspberry Pi. Vous pouvez alors interagir avec l'interface graphique pour faire vos expérimentations

Voici une capture d'écran de l'interface de l'application de pilotage:



Les différents éléments sont les suivants (de haut en bas)

- courbes: l'interface permet de visualiser
  - la tension de commande (V, en rouge)
  - les vitesses mesurées (rad/s, en bleu et vert)
- zone de sélection de l'adresse IP: entrer l'adresse IP de votre robot et cliquer sur le bouton « Connexion ». Vous verrez alors les courbes se mettre en jour automatiquement et défiler des valeurs numériques dans la zone d'affichage à droite
- « Rafraîchissement »: ceci pilote la fréquence de rafraîchissement des courbes. En fonction de la vitesse de votre ordinateur, vous pouvez choisir parmi les 4 valeurs « Minimum », « Lent », « Moyen » et « Rapide ». Plus la fréquence de rafraîchissement est élevée, moins le tracé est saccadé. Mais si votre ordinateur n'est pas très rapide, vous risquez d'observer un retard entre les consignes et l'affichage. Dans ce cas, il faut choisir un rafraîchissement plus lent
- « Fréq. éch. (Hz) »: ce paramètre correspond à la fréquence d'échantillonnage des mesures dans le programme `CommandeEnTension.py` (20 ms par défaut)
- Bouton « Enregistrer »: il permet d'enregistrer les valeurs numériques affichées dans la zone de « log » vers un fichier texte pour une utilisation ultérieure avec n'importe quel logiciel permettant de lire des données séparées par des virgules dans un fichier texte.  
**Attention** : ce bouton n'est actif qu'après une séquence de mesure. Il est grisé le reste du temps (au démarrage du programme et pendant une séquence de mesure)
- « Appliquer les consignes ci-dessous sur: »: permet d'appliquer les consignes sur le moteur gauche, le moteur droit ou les deux moteurs
- « Système: »: cette interface étant commune avec les Geeros pcDuino et Raspberry C / Python, cette option permet de choisir le système du robot utilisé
- Zone « Consigne »: elle permet de modifier la consigne de tension avec les curseurs. La vitesse des moteurs varie alors. Vous pouvez la visualiser, ainsi que la consigne et la tension de commande, sur le graphique qui s'affiche en temps-réel

### IMPORTANT !

Si vous comparez la vitesse de rotation des deux moteurs, vous obtiendrez probablement une valeur différente, bien que la tension de commande soit la même : c'est normal, ceci est dû à la disparité des caractéristiques des moteurs à courant continu.

D'un point de vue pédagogique, ce point permet de souligner la nécessité de réaliser un asservissement de vitesse : si l'on souhaite une vitesse de rotation précise, on ne peut pas se contenter d'une commande en boucle ouverte.

## 9.3 - Asservissement des moteurs en vitesse (en Python)

Cette expérience permet d'asservir la vitesse de rotation des moteurs en appliquant une consigne de vitesse par l'intermédiaire d'une application qui s'exécute sur votre ordinateur.

Le programme Python gérant cette expérience se nomme AsservissementVitesse.py. Il est situé dans le répertoire /root/programmes\_python.

Il comporte de nombreux commentaires permettant de comprendre facilement son fonctionnement.

Le principe de pilotage des moteurs consiste à calculer, grâce à un régulateur de type PID, la tension de commande à appliquer aux moteurs (via commande PWM de la carte A-Star connectée sur la carte Raspberry Pi) pour qu'il suive la consigne de vitesse spécifiée.

Notez que dans ce programme, la vitesse mesurée est en fait la moyenne glissante des 10 derniers échantillons de mesure de vitesse instantanée, ce qui permet d'avoir une mesure plus lisse. En effet, la résolution de la mesure instantanée est la suivante:

$$2*\pi/(Ts*CPR*ratio)$$

avec :

- Ts : cadence d'échantillonnage (0.01 s)
- CPR : nombre d'impulsions par tour du codeur (48)
- ratio : rapport de réduction du moteur (34)

Ceci donne une résolution de  $2*\pi/(0.01*1632) = 0.4$  rad/s. Le moyennage permet d'améliorer cette valeur.

La consigne de vitesse peut se fixer interactivement via une application qui s'exécute sur votre ordinateur. Elle peut se télécharger à l'adresse suivante:

<http://www.3sigma.fr/Telechargements-Geeros.html>

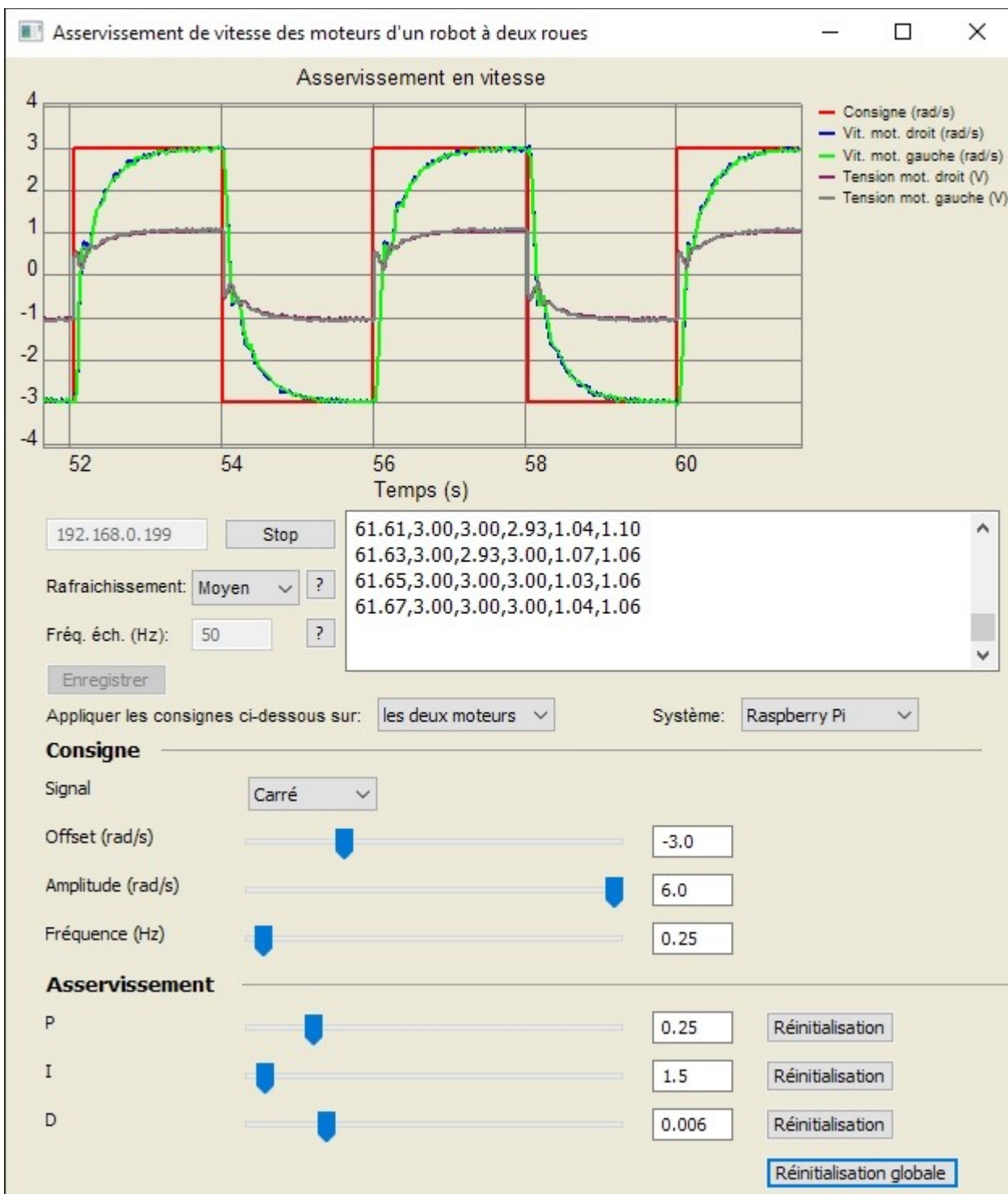
Son nom est de la forme InterfaceReseauAsservissementDeuxMoteursEnVitesse\_x.y.zip (x.y correspond au numéro de version du programme).

Pour l'installer, il suffit de décompresser l'archive dans le répertoire de votre choix. Pour l'exécuter, double-cliquer sur InterfaceReseauAsservissementDeuxMoteursEnVitesse.exe.

Pour changer la consigne de vitesse du moteur depuis votre ordinateur, il vous suffit de suivre les étapes suivantes (remarque: n'exécutez pas de nouveau celles que vous avez déjà effectuées):

- mettre le système sous tension en positionnant le bouton marche-arrêt sur « I »
- se connecter à la carte Raspberry Pi en Wifi à l'adresse 192.168.0.199
- lancer l'application «InterfaceReseauAsservissementDeuxMoteursEnVitesse»
- entrer l'adresse IP du robot dans la zone correspondante
- sélectionner le système « Raspberry Pi » sous la zone blanche
- cliquer sur « Connexion ». L'application démarre alors à distance le programme Python d'asservissement des moteurs sur la carte Raspberry Pi. Vous pouvez alors interagir avec l'interface graphique pour faire vos expérimentations

Voici une capture d'écran de l'interface de l'application de pilotage:



Les différents éléments sont les suivants (de haut en bas)

- courbes: l'interface permet de visualiser
  - la consigne de vitesse (rad/s, en rouge)
  - les vitesses mesurées (rad/s, en bleu et en vert)
  - les tensions de commande (V, en marron et en gris)
- zone de sélection de l'adresse IP: entrer l'adresse IP de votre robot et cliquer sur le bouton « Connexion ». Vous verrez alors les courbes se mettre en jour automatiquement et défiler des valeurs numériques dans la zone d'affichage à droite
- « Rafraîchissement »: ceci pilote la fréquence de rafraîchissement des courbes. En fonction de la vitesse de votre ordinateur, vous pouvez choisir parmi les 4 valeurs « Minimum », « Lent », « Moyen » et « Rapide ». Plus la fréquence de rafraîchissement est élevée, moins le tracé est saccadé. Mais si votre ordinateur n'est pas très rapide, vous risquez d'observer un retard entre les consignes et l'affichage. Dans ce cas, il faut choisir un rafraîchissement plus lent
- « Fréq. éch. (Hz) »: ce paramètre correspond à la fréquence d'échantillonnage des mesures dans le programme AsservissementVitesse.py.
- Bouton « Enregistrer »: il permet d'enregistrer les valeurs numériques affichées dans la zone de « log » vers un fichier texte pour une utilisation ultérieure avec n'importe quel logiciel permettant de lire des données séparées par des virgules dans un fichier texte.  
**Attention** : ce bouton n'est actif qu'après une séquence de mesure. Il est grisé le reste du temps (au démarrage du programme et pendant une séquence de mesure)
- « Appliquer les consignes ci-dessous sur: »: permet d'appliquer les consignes sur le moteur gauche, le moteur droit ou les deux moteurs
- « Système: »: cette interface étant commune avec les Geeros pcDuino et Raspberry C / Python, cette option permet de choisir le système du robot utilisé
- Zone « Consigne »: elle permet de modifier la consigne de vitesse avec les curseurs. La vitesse des moteurs varie alors. Vous pouvez la visualiser, ainsi que la consigne et la tension de commande, sur le graphique qui s'affiche en temps-réel
- Zone « Asservissement »: elle permet de modifier les gains du régulateur PID pendant le fonctionnement du système. Cela permet de voir l'influence de ces gains sur les performances de l'asservissement et de régler précisément ce dernier.



## 9.4 - Commande des moteurs en tension (en C/C++)

Cette expérience permet de changer la vitesse de rotation des moteurs en appliquant une tension variable par l'intermédiaire d'une application qui s'exécute sur votre ordinateur. La connexion entre l'ordinateur et le système A-Star (qui exécute le programme) se fait via une liaison série.

Le programme Arduino peut être téléchargé à l'adresse suivante:

<http://www.3sigma.fr/Telechargements-Geeros.html>

Son nom est de la forme GeerosAStarCommandeMoteursEnTension\_x.y.zip (x.y correspond au numéro de version du programme).

Il comporte de nombreux commentaires permettant de comprendre facilement son fonctionnement.

Le principe de pilotage des moteurs consiste à envoyer des signaux PWM sur le pont en H intégré à la carte A-Star afin de faire varier la tension d'alimentation des moteurs. Cette tension peut être modifiée interactivement via une application qui s'exécute sur votre ordinateur. Elle peut se télécharger à l'adresse suivante:

<http://www.3sigma.fr/Telechargements-Geeros.html> (voir dans la partie « Version Romeo »)

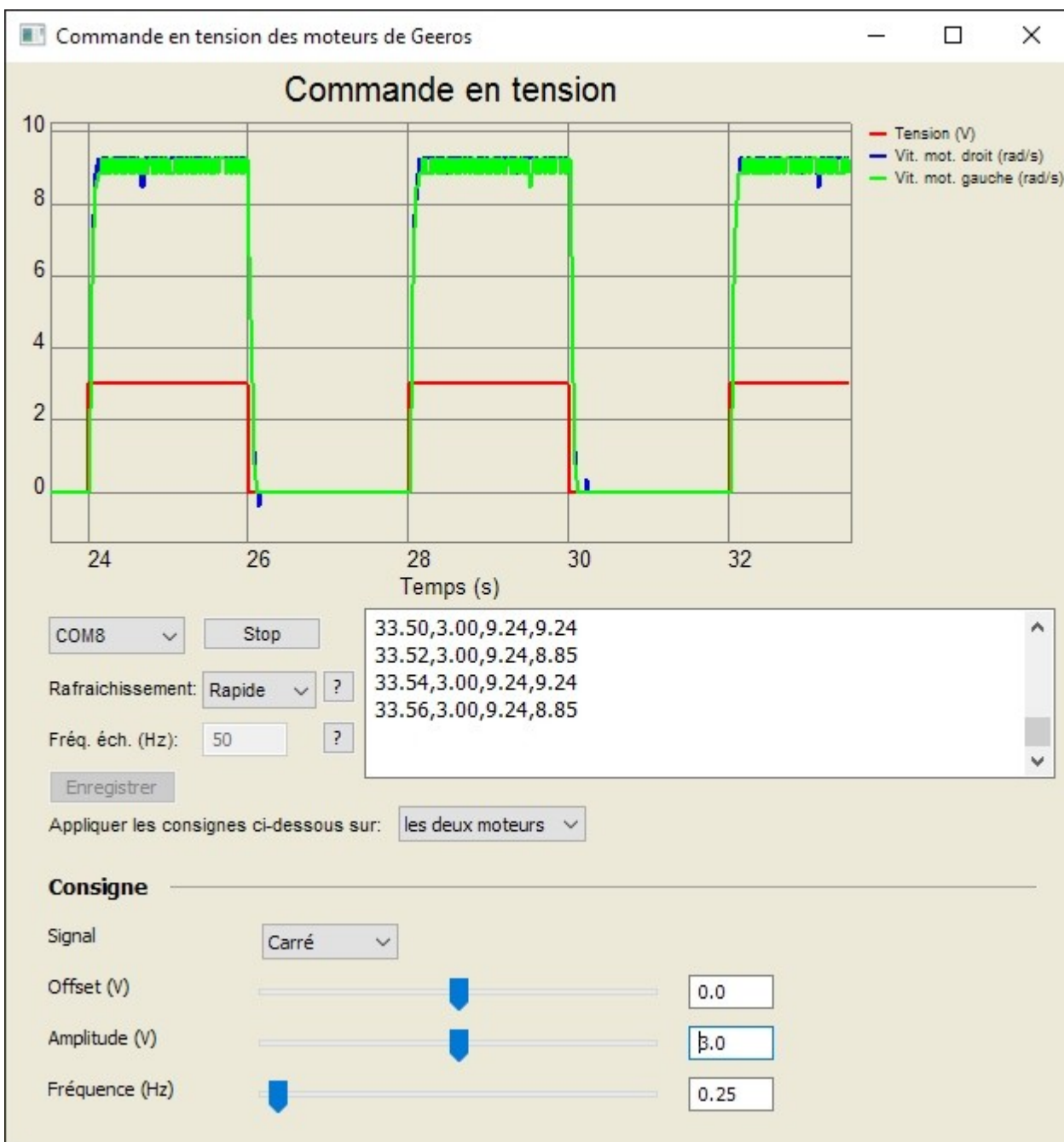
Son nom est de la forme GeerosInterfaceCommandeMoteursEnTension\_x.y.zip (x.y correspond au numéro de version du programme).

Pour l'installer, il suffit de décompresser l'archive dans le répertoire de votre choix. Pour l'exécuter, double-cliquer sur GeerosInterfaceCommandeMoteursEnTension\_x.y.exe.

Pour piloter la tension d'alimentation (et donc la vitesse) des moteurs depuis votre ordinateur, il vous suffit de suivre les étapes suivantes :

- mettre le système sous tension en positionnant le bouton marche-arrêt sur « I »
- connecter la carte A-Star à l'ordinateur avec le câble USB fourni
- télécharger le programme Arduino « GeerosAStarCommandeMoteursEnTension » sur la carte A-Star
- lancer l'application « GeerosInterfaceCommandeMoteursEnTension »

Voici une capture d'écran de l'interface de l'application de pilotage:



Les différents éléments sont les suivants (de haut en bas)

- courbes: l'interface permet de visualiser
  - la tension de commande (V, en rouge)
  - les vitesses mesurées (rad/s, en bleu et vert)
- zone de sélection du port série: choisir le port série sur lequel est connectée votre carte A-Star et cliquer sur le bouton « Connexion ». Vous verrez alors les courbes se mettre en jour automatiquement et défiler des valeurs numériques dans la zone d'affichage à droite
- « Rafraîchissement »: ceci pilote la fréquence de rafraîchissement des courbes. En fonction de la vitesse de votre ordinateur, vous pouvez choisir parmi les 4 valeurs « Minimum », « Lent », « Moyen » et « Rapide ». Plus la fréquence de rafraîchissement est élevée, moins le tracé est saccadé. Mais si votre ordinateur n'est pas très rapide, vous risquez d'observer un retard entre les consignes et l'affichage. Dans ce cas, il faut choisir un rafraîchissement plus lent
- « Fréq. éch. (Hz) »: ce paramètre correspond à la fréquence d'échantillonnage des mesures dans le programme Arduino. Attention: ce paramètre doit être spécifié en Hz, alors que les valeurs correspondantes dans le programme Arduino (CADENCE\_MS et TSDATA) sont spécifiées en ms. La relation entre les deux est la suivante:  
$$\text{freq (Hz)} = 1000 / \text{cadence (ms)}$$
- Bouton « Enregistrer »: il permet d'enregistrer les valeurs numériques affichées dans la zone de « log » vers un fichier texte pour une utilisation ultérieure avec n'importe quel logiciel permettant de lire des données séparées par des virgules dans un fichier texte.  
**Attention** : ce bouton n'est actif qu'après une séquence de mesure. Il est grisé le reste du temps (au démarrage du programme et pendant une séquence de mesure)
- « Appliquer les consignes ci-dessous sur: »: permet d'appliquer les consignes sur le moteur gauche, le moteur droit ou les deux moteurs
- Zone « Consigne »: elle permet de modifier la consigne de tension avec les curseurs. La vitesse des moteurs varie alors. Vous pouvez la visualiser, ainsi que la consigne et la tension de commande, sur le graphique qui s'affiche en temps-réel

#### IMPORTANT !

Si l'affichage ne suit pas les consignes même avec une fréquence de rafraîchissement très lente, veuillez diminuer la cadence d'échantillonnage dans le programme Arduino (augmenter les valeurs TSDATA et CADENCE\_MS)

#### IMPORTANT !

Si vous comparez la vitesse de rotation des deux moteurs, vous obtiendrez probablement une valeur différente, bien que la tension de commande soit la même : c'est normal, ceci est dû à la disparité des caractéristiques des moteurs à courant continu.

D'un point de vue pédagogique, ce point permet de souligner la nécessité de réaliser un asservissement de vitesse : si l'on souhaite une vitesse de rotation précise, on ne peut pas se contenter d'une commande en boucle ouverte.

## 9.5 - Asservissement des moteurs en vitesse (en C/C++)

Cette expérience permet d'asservir la vitesse de rotation des moteurs en appliquant une consigne de vitesse par l'intermédiaire d'une application qui s'exécute sur votre ordinateur. La connexion entre l'ordinateur et le système A-Star (qui exécute le programme) se fait via une liaison série.

Le programme Arduino peut être téléchargé à l'adresse suivante:

<http://www.3sigma.fr/Telechargements-Geeros.html>

Son nom est de la forme GeerosAStarAsservissementMoteursEnVitesse\_x.y.zip (x.y correspond au numéro de version du programme).

Il comporte de nombreux commentaires permettant de comprendre facilement son fonctionnement.

Le principe de pilotage des moteurs consiste à calculer, grâce à un régulateur de type PID, la tension de commande à appliquer aux moteurs (via commande PWM du pont en H intégré sur la carte A-Star) pour qu'il suive la consigne de vitesse spécifiée.

Notez que dans ce programme, la vitesse mesurée est en fait la moyenne glissante des 10 derniers échantillons de mesure de vitesse instantanée, ce qui permet d'avoir une mesure plus lisse. En effet, la résolution de la mesure instantanée est la suivante:

$$2*\pi/(Ts*CPR*ratio)$$

avec :

- Ts : cadence d'échantillonnage (0.01 s)
- CPR : nombre d'impulsions par tour du codeur (48)
- ratio : rapport de réduction du moteur (34)

Ceci donne une résolution de  $2*\pi/(0.01*1632) = 0.4$  rad/s. Le moyennage permet d'améliorer cette valeur.

La consigne de vitesse peut se fixer interactivement via une application qui s'exécute sur votre ordinateur. Elle peut se télécharger à l'adresse suivante:

<http://www.3sigma.fr/Telechargements-Geeros.html> (voir dans la partie « Version Romeo »)

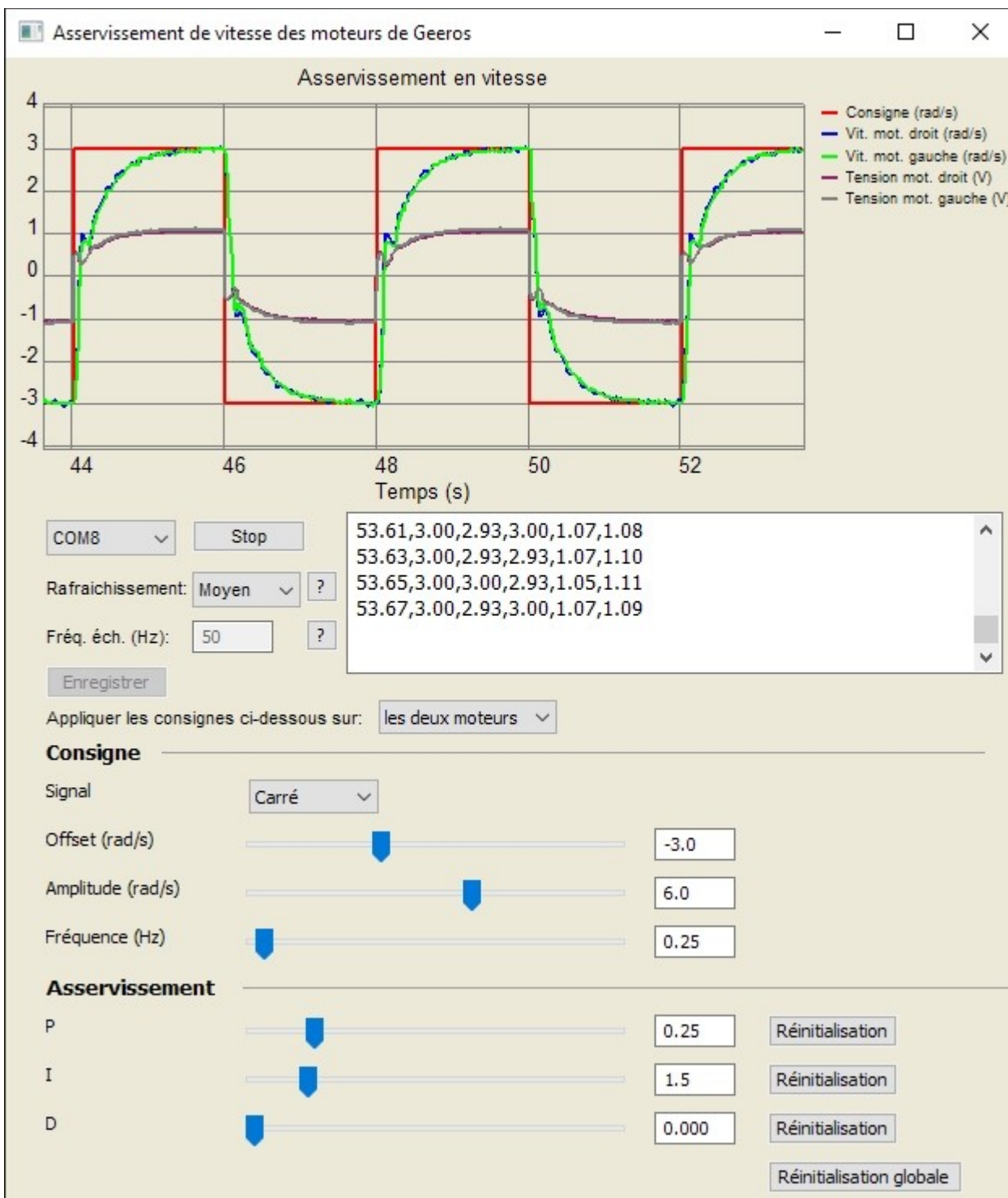
Son nom est de la forme GeerosInterfaceAsservissementMoteursEnVitesse\_x.y.zip (x.y correspond au numéro de version du programme).

Pour l'installer, il suffit de décompresser l'archive dans le répertoire de votre choix. Pour l'exécuter, double-cliquer sur GeerosInterfaceAsservissementMoteursEnVitesse\_x.y.exe.

Pour changer la consigne de vitesse du moteur depuis votre ordinateur, il vous suffit de suivre les étapes suivantes (remarque: n'exécutez pas de nouveau celles que vous avez déjà effectuées):

- mettre le système sous tension en positionnant le bouton marche-arrêt sur « I »
- connecter la carte A-Star à l'ordinateur avec le câble USB fourni
- télécharger le programme Arduino « GeerosAStarAsservissementMoteursEnVitesse » sur la carte A-Star
- lancer l'application « GeerosInterfaceAsservissementMoteursEnVitesse »

Voici une capture d'écran de l'interface de l'application de pilotage:



Les différents éléments sont les suivants (de haut en bas)

- courbes: l'interface permet de visualiser
  - la consigne de vitesse (rad/s, en rouge)
  - les vitesses mesurées (rad/s, en bleu et en vert)
  - les tensions de commande (V, en marron et en gris)
- zone de sélection du port série: choisir le port série sur lequel est connectée votre carte A-Star et cliquer sur le bouton « Connexion ». Vous verrez alors les courbes se mettre en jour automatiquement et défiler des valeurs numériques dans la zone d'affichage à droite
- « Rafraîchissement »: ceci pilote la fréquence de rafraîchissement des courbes. En fonction de la vitesse de votre ordinateur, vous pouvez choisir parmi les 4 valeurs « Minimum », « Lent », « Moyen » et « Rapide ». Plus la fréquence de rafraîchissement est élevée, moins le tracé est saccadé. Mais si votre ordinateur n'est pas très rapide, vous risquez d'observer un retard entre les consignes et l'affichage. Dans ce cas, il faut choisir un rafraîchissement plus lent
- « Fréq. éch. (Hz) »: ce paramètre correspond à la fréquence d'échantillonnage des mesures dans le programme Arduino. Attention: ce paramètre doit être spécifié en Hz, alors que la valeur correspondante dans le programme Arduino (TSDATA) est spécifiée en ms. La relation entre les deux est la suivante:  
$$\text{freq (Hz)} = 1000 / \text{cadence (ms)}$$
- Bouton « Enregistrer »: il permet d'enregistrer les valeurs numériques affichées dans la zone de « log » vers un fichier texte pour une utilisation ultérieure avec n'importe quel logiciel permettant de lire des données séparées par des virgules dans un fichier texte.  
**Attention** : ce bouton n'est actif qu'après une séquence de mesure. Il est grisé le reste du temps (au démarrage du programme et pendant une séquence de mesure)
- « Appliquer les consignes ci-dessous sur: »: permet d'appliquer les consignes sur le moteur gauche, le moteur droit ou les deux moteurs
- Zone « Consigne »: elle permet de modifier la consigne de vitesse avec les curseurs. La vitesse des moteurs varie alors. Vous pouvez la visualiser, ainsi que la consigne et la tension de commande, sur le graphique qui s'affiche en temps-réel
- Zone « Asservissement »: elle permet de modifier les gains du régulateur PID pendant le fonctionnement du système. Cela permet de voir l'influence de ces gains sur les performances de l'asservissement et de régler précisément ce dernier.

#### **IMPORTANT !**

Si l'affichage ne suit pas les consignes même avec une fréquence de rafraîchissement très lente, veuillez diminuer la cadence d'échantillonnage dans le programme Arduino (augmenter la valeur TSDATA). Ne pas modifier la valeur CADENCE\_MS, sous peine de déstabiliser l'asservissement.

## 10 - Autres documentations disponibles

Cette documentation, bien que baptisée « complète », ne détaille pas certains points qui sont présents dans deux autres documentations disponibles sur simple demande à l'adresse [contact@geeros.com](mailto:contact@geeros.com):

- Documentation « développeur »: elle regroupe toutes les informations utiles pour ceux qui souhaitent entrer dans le cœur du système Linux de la carte Raspberry Pi. Cette documentation n'est pas en accès libre car elle contient des informations qui, utilisées à mauvais escient, peuvent corrompre le bon fonctionnement de votre Geeros.
- Documentation pédagogique: elle contient tous les éléments relatifs à la mise en équations du robot ainsi que les documents Maple et modèles MapleSim associés. Cette documentation, utile dans le cadre de l'enseignement, n'est pas en accès libre car elle contient des informations qui peuvent être les solutions d'exercices proposés par les professeurs. Ceux-ci sont donc les seuls habilités à y avoir accès.



## **11 - Important**

Geeros est un produit « vivant » en constant développement pour améliorer ou lui ajouter de nouvelles fonctionnalités. Si vous avez des idées ou des besoins pour des développements spécifiques, n'hésitez pas à nous contacter ([contact@geeros.com](mailto:contact@geeros.com)).

**Ne restez jamais bloqué sans nous contacter !**

Pour tout problème ou toute requête, contactez-nous à l'adresse [contact@geeros.com](mailto:contact@geeros.com)