

# Robotique collective et auto-assemblage, une étude mécatronique

THÈSE N° 4875 (2010)

PRÉSENTÉE LE 19 NOVEMBRE 2010

À LA FACULTÉ SCIENCES ET TECHNIQUES DE L'INGÉNIEUR

LABORATOIRE DE SYSTÈMES ROBOTIQUES 1

PROGRAMME DOCTORAL EN SYSTÈMES DE PRODUCTION ET ROBOTIQUE

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Michael BONANI

acceptée sur proposition du jury:

Dr A. Karimi, président du jury  
Dr F. Mondada, directeur de thèse  
Dr G. Caprari, rapporteur  
Prof. M. Lauria, rapporteur  
Dr J.-C. Zufferey, rapporteur



ÉCOLE POLYTECHNIQUE  
FÉDÉRALE DE LAUSANNE

Suisse  
2010





# Abstract

We present a study of collective robotics by including a mechatronics point of view. In the field it is usually claimed that collective robots are simple and relatively cheap because they are produced in large quantities. Instead in our study we show that collective robots are not simple because they need sensors and actuators for additional work. Our experience in designing robots and producing them allows us to analyze the manufacturing costs of different collective robots .

We present in this work four robots developed prior and during to this thesis. Chapter 2 concerns the e-puck robot. This is a robot designed for education, however, it is used in research to experiment collective behaviors. The s-bot robot is presented in Chapter 3. This is a robot that has the collective ability to self-assemble to form larger structures and increase its performance. In Chapter 4 we present the marXbot robot. It is a modular robot developed in the laboratory to meet the needs of different research hubs. One of its modules allows it to self-assemble with its teammates and with the robot handbot presented in Chapter 5. The handbot is a robot that can climb and handle objects. It can climb for example a shelf to grasp a book. On the ground it is transported by marXbot.

In Chapter 6 the performance of robots are presented. We expose a study of their performance gain when the robots are self-assembled. Finally we compare in Chapter 7 the four robots from the mechatronics point of view and in respect to their cost.

**Keywords:** autonomous mobile robot, collective, self-assembly, mechatronics, modular, handling, climbing, cost, complexity.



# Résumé

Ce travail est une étude sur les robots collectifs d'un point de vue mécatronique. Les prétentions de certains spécialistes du domaine présentent les robots collectifs comme simples et relativement bon marché car produits en grande quantité. Au contraire, nous montrons dans notre étude que les robots collectifs ne sont pas si simples notamment parce qu'ils ont besoin de capteurs et d'actionneurs additionnels pour collaborer. Grâce à notre expérience dans la conception de robots et dans la mise en production, nous avons fait une analyse des coûts de fabrication de différents robots collectifs.

Nous exposons dans ce travail quatre robots développés dans le cadre de travaux précédents et pendant cette thèse. Le chapitre 2 est consacré au robot e-puck. C'est un robot conçu pour l'éducation, néanmoins, il est utilisé en recherche pour des expériences de comportements collectifs. Le robot s-bot est présenté dans le chapitre 3. C'est un robot collectif qui a la capacité de s'auto-assembler pour former des structures plus grandes et ainsi augmenter ses performances. Dans le chapitre 4, nous présentons le robot marXbot. C'est un robot modulaire développé dans le laboratoire pour répondre aux besoins de différents pôles de recherche. Un de ses modules lui permet de s'auto-assembler à ses pairs et au robot handbot présenté dans le chapitre 5. Ce dernier est un robot grimpeur pouvant manipuler des objets. Il est capable par exemple de gravir une étagère et de prendre un livre. Au sol, il est transporté par les marXbots.

Dans le chapitre 6, nous analysons les différentes performances de chacun des robots. Nous exposons une étude de leurs gains en performance lorsque les robots sont auto-assemblés. Finalement, nous comparons, dans le chapitre 7, les quatre robots d'un point de vue mécatronique et selon leur prix de revient.

**Mots clés :** robot mobile autonome, collectif, auto-assemblage, mécatronique, modulaire, manipulation, grimpe, coût, complexité.



# Remerciements

Je remercie premièrement mon directeur de thèse Francesco Mondada. Notre longue collaboration, depuis mon diplôme, nous a amenés à faire de nombreux projets qui j'espère auront également du succès dans le futur. J'apprécie sa rigueur dans le travail et j'estime encore plus la place première que tient sa famille dans sa vie. Cet équilibre apporte une sérénité que chaque personne devrait apprécier à sa juste valeur.

Je remercie le projet de recherche Swarmanoid (subvention IST-FET 022 888), ainsi que l'EPFL, pour m'avoir fourni le financement au cours de ce travail.

J'exprime ma gratitude aux membres du comité de thèse : le Prof. Michel Lauria, le Dr Gilles Caprari, le Dr Jean-Christophe Zufferey et Dr. Alireza Karimi pour avoir accepté d'évaluer ce travail.

Je suis reconnaissant envers mes collègues de laboratoire : Stéphane Magnenat, Philippe Retornaz, Alexey Gribovskiy, Valentin Longchamp, Daniel Burnier, Frédéric Rochat, Patrick Schoeneich, Tarek Baaboura, Pierre Noirat, Fanny Riedo, Mariza Freire, Roderich Gross, Anne-Remillet Schaller et Florian Vaussard. Ils ont toujours créé une atmosphère chaleureuse dans le laboratoire. Avec eux, la collaboration est plus que facile et a permis d'avancer dans la tribulation des projets.

Je remercie surtout Stéphane qui, par ses compétences informatiques, a apporté une réelle plus-value aux robots que nous avons développés et qui m'a beaucoup aidé dans mon travail. Je remercie également Philippe, qui a su deverminer et programmer tous les circuits du marXbot et handbot. Un grand merci à Dany, qui m'a énormément aidé pour le routage et la production des circuits électroniques et sans qui le bureau serait encore plus apocalyptique. J'exprime ma gratitude à Tarek qui a prototypé des pièces très compliquées et simplifié une grande partie de celles qui l'étaient inutilement.

J'ai énormément apprécié de la part d'André Guignard le partage de son expérience inestimable pour le design de robots. Je remercie particulièrement Peter Bruelmeier pour son expertise dans le routage de circuit électroniques. Un grand chapeau à toute l'équipe de ACI, qui a su fabriquer les nombreuses versions des PCB aux formes les plus improbables et aux exigences de plus en plus pointues.

Je remercie également mes collègues des différents laboratoires (ASL1, ASL2 ,IRIDIA) où j'ai travaillé, ce fut à chaque fois des expériences enrichissantes.

Je remercie chaleureusement Nathalie, Mariza Freire et Francesco Cimini pour avoir corrigé les fautes de grammaire et d'orthographe que j'ai produites en masse durant l'écriture de ce document.

Je suis très reconnaissant envers tous mes amis, Simon, Fred, Nadja, Bal, Colpo, Sergino, Julien, les "Poissards" et les autres qui m'ont par leur présence et leur écoute soutenu toutes ces années. Merci pour leur amitié et tous ces bons moments, ballades, vacances et fêtes sans quoi le vie serait trop monotone.

Je remercie ma famille et belle famille pour leur soutien. Finalement je remercie mes parents pour m'avoir permis de faire des études et pour leur soutien inconditionnel dans mes projets.

*Je dédie cette thèse à mon épouse Nathalie, sans qui rien n'est pour moi possible.  
Merci pour ton soutien, tes sacrifices et notre amour.*

# Table des matières

<b>Abstract</b>	<b>i</b>
<b>Résumé</b>	<b>iii</b>
<b>Remerciements</b>	<b>v</b>
<b>Table des matières</b>	<b>vii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.1.1 Prétentions du domaine . . . . .	1
1.1.2 Nos observations . . . . .	2
1.2 Objectifs . . . . .	3
1.3 État de l’art . . . . .	4
1.4 Approche . . . . .	7
1.4.1 Structure de la thèse . . . . .	8
<b>2 L’e-puck</b>	<b>9</b>
2.1 Design de l’e-puck . . . . .	10
2.1.1 La mécanique . . . . .	11
2.1.2 L’électronique . . . . .	13
2.1.3 Les techniques de production . . . . .	16
2.1.4 Les extensions . . . . .	17
2.2 Caractéristiques de l’e-puck . . . . .	19
2.3 L’e-puck en recherche sur la robotique collective . . . . .	21
2.4 Leçons apprises, limites et améliorations . . . . .	22
2.5 Conclusion . . . . .	23
<b>3 Le s-bot</b>	<b>25</b>
3.1 Design du s-bot . . . . .	26
3.1.1 La mécanique . . . . .	28
3.1.2 L’électronique . . . . .	36
3.1.3 Les logiciels . . . . .	46
3.1.4 Les techniques de production . . . . .	49
3.2 Caractéristiques du s-bot . . . . .	51
3.2.1 La capacité d’auto-assemblage . . . . .	53

3.3	Leçons apprises, limites et améliorations . . . . .	54
3.4	Conclusion . . . . .	56
<b>4</b>	<b>Le MarXbot</b>	<b>57</b>
4.1	Design du marXbot . . . . .	58
4.1.1	La mécanique . . . . .	60
4.1.2	L'électronique . . . . .	72
4.1.3	Les logiciels . . . . .	85
4.1.4	Les techniques de production . . . . .	90
4.2	Caractéristiques du marXbot . . . . .	91
4.2.1	La capacité d'auto-assemblage . . . . .	94
4.2.2	SLAM . . . . .	96
4.2.3	La station d'échange de la batterie . . . . .	97
4.3	Leçons apprises, limites et améliorations . . . . .	99
4.4	Conclusion . . . . .	100
<b>5</b>	<b>Le Handbot</b>	<b>103</b>
5.1	Design du handbot . . . . .	104
5.1.1	La mécanique . . . . .	107
5.1.2	L'électronique . . . . .	116
5.1.3	Les techniques de production . . . . .	124
5.2	Caractéristiques du handbot . . . . .	125
5.2.1	La prise d'un livre sur une étagère . . . . .	127
5.3	La prise d'un livre sans escalade . . . . .	131
5.4	Leçons apprises, limites et améliorations . . . . .	131
5.5	Conclusion . . . . .	134
<b>6</b>	<b>Capacités collectives en auto-assemblage</b>	<b>135</b>
6.1	Super-linéarité des performances des robots . . . . .	135
6.1.1	La super-linéarité des performances du s-bot . . . . .	136
6.1.2	La super-linéarité des marXbots . . . . .	143
6.1.3	La super-linéarité des handbots . . . . .	144
6.1.4	Analyse de la super-linéarité du handbot . . . . .	150
6.2	Exemples d'expériences collectives complètes . . . . .	151
6.2.1	Les s-bots à la recherche de nourriture . . . . .	151
6.2.2	Le scénario de Swarmanoid . . . . .	155
6.3	Conclusion . . . . .	155
<b>7</b>	<b>Comparaison des quatre robots</b>	<b>159</b>
7.1	Comparatif des caractéristiques et coûts . . . . .	159
7.2	Conclusion . . . . .	163



<b>8 Conclusion</b>	<b>165</b>
8.1 Conclusion . . . . .	165
8.2 Mes contributions à l'état de l'art . . . . .	166
<b>Bibliographie</b>	<b>169</b>
<b>Curriculum vitæ</b>	<b>175</b>



# 1 Introduction

## 1.1 Motivation

La robotique collective est un domaine vaste. Nous proposons un projet à mi-chemin entre deux approches de la robotique collective. La première est la robotique en essaim, où un grand nombre de robots autonomes relativement simples interagissent pour produire des comportements complexes [4]. La seconde est la robotique auto-reconfigurable, où les modules simples avec quelques degrés de liberté sont joints les uns aux autres pour créer des structures plus grandes avec des formes différentes [28,80].

Notre approche est composée de robots mobiles et autonomes qui peuvent s'auto-assembler pour former des structures plus grandes ou accroître leurs performances. C'est un domaine particulier de la robotique en essaim à cause des interactions physiques où l'accrochage se rapproche de la robotique auto-reconfigurable.

### 1.1.1 Prétentions du domaine

La plupart des chercheurs des domaines de la robotique en essaim et auto-reconfigurable affirment que des robots "simples" et "peu coûteux" peuvent réaliser des tâches complexes. Ils suggèrent également que la robotique collective est plus robuste et polyvalente que des solutions basées sur un seul robot spécialisé.

- Bonabeau et al. [5] affirment que : "distributed control and optimization algorithms... tend to exhibit a high degree of flexibility and robustness in a dynamic environment." and "Swarm Intelligence ... is promising in the perspective of miniaturization and low-cost robotics (see for example the Alice micro-robots...)"
- Groß et Dorigo dans [32] déclarent que : "The transport of heavy objects by groups of robots can be motivated by low cost of manufacture, high robustness, high failure tolerance, or high flexibility - all desirable properties for a robotic system to have. Instead of using one robot powerful enough to transport the object (also called prey hereafter) on its own, the task is accomplished in cooperation by a group of simpler robots."
- Seyfried et al. [72] considèrent que : "We believe that the availability of a (possibly commercially available) low-cost, mass fabricated swarm

micro robot will have a great impact in the fields of education, science and possibly also entertainment."

- Peleg [65] résume comme suit : "Systems consisting of a group of autonomously operating mobile robots (sometimes referred to as robot swarms) have attracted considerable interest throughout the past twenty years, due to their potential for providing flexible, low-cost solutions in hazardous situations (e.g., military operations, toxic environments or fire fighting). The idea is to deal with such applications using swarms consisting of many small and simple robots, with very limited capabilities (e.g., low energy sources, limited communication means and weak processors). The use of tiny, functionally simple and cheap robots may make it acceptable to lose some of the robots, so long as the team manages to achieve its collective goals."
- Yim et al [79] résument les motivations pour la robotique reconfigurable : "There are three key motivations for designing modular selfreconfigurable robotic systems :
  - Versatility : Self-reconfigurable robotic systems are potentially more adaptive than conventional systems. The ability to reconfigure allows a robot or a group of robots to disassemble and reassemble machines to form new morphologies that are better suited for new tasks, such as changing from a legged robot to a snake robot and then to a rolling robot.
  - Robustness : Since robot parts are interchangeable (within a robot and between different robots), machines can also replace faulty parts autonomously, leading to self-repair.
  - Low Cost : Self-reconfigurable robotic systems can potentially lower overall robot cost by making many copies of one (or relatively few) type of modules so economies of scale and mass production come into play. Also, a range of complex machines can be made from one set of modules, saving costs through reuse and generality of the system.

These three advantages have not yet been fully realized."

### 1.1.2 Nos observations

Bien que les théories précitées ne sont pas totalement fausses, en particulier dans le cas de tâches et d'environnements simples, elles n'ont jamais été démontrées. La plupart des expériences sont réalisées en simulation et non pas avec des robots réels. Lorsque on regarde les derniers travaux à succès [31,60,61,74] obtenus avec des robots réels, nous constatons que la plupart de ces robots ont un grand nombre de capteurs, une haute puissance de calcul, des capacités de communication élevées et beau-

coup de degrés de liberté. D'un point de vue mécatronique, même si les comportements peuvent être décrits par des règles simples, les individus eux-mêmes ne sont pas simples. La collaboration entre des robots nécessite des capteurs évolués pour localiser ses coéquipiers, des mécanismes de connexion physique et des dispositifs de communication. Par contre, au niveau global, nous sommes d'accord que les comportements du robot et les interactions entre les robots peuvent souvent être modélisés par des règles simples.

Des travaux récents sur les insectes sociaux, qui sont la principale source d'inspiration pour la robotique en essaim, montrent qu'à côté des comportements généralement régis par des règles simples, certains comportements spécifiques, mais critiques exigent des interactions complexes [21,78] ou un contrôle pointu [76].

### 1.2 Objectifs

Les principaux objectifs de ce travail sont :

- Comprendre et expliquer pourquoi les robots "complexes" et "chers" sont obligatoires pour la robotique collective, en opposition aux prétentions dans le domaine.
- Analyser les gains en performance par l'auto-assemblage d'un groupe de robots.

Nous avons conçu dans des travaux passés et récents, différents types de robots collectifs. Cette expérience nous place dans une situation idéale pour pouvoir analyser leur structure, leur coût et leur performance. Nous allons comparer les différents robots en termes de degrés de liberté, de puissance de calcul, d'interaction, de capacité de communication.

Plusieurs chercheurs qui suggèrent d'utiliser les robots collectifs, affirment que leur solution est bon marché. Le principal argument théorique est que des modules identiques de petites tailles et de production de masse sont moins chers à construire que les gros robots personnalisés. Nous allons résumer les prix des différents robots produits en quantité et comparer leurs volumes et leur complexité. Notre objectif est de montrer que l'approche collective n'est pas moins chère, mais comporte d'autres avantages.

Les robots ayant des capacités d'auto-assemblages augmentent leurs performances de façon super-linéaire ce qui est très intéressant. Nous allons étudier ce phénomène plus en détail sur différents types de robots.

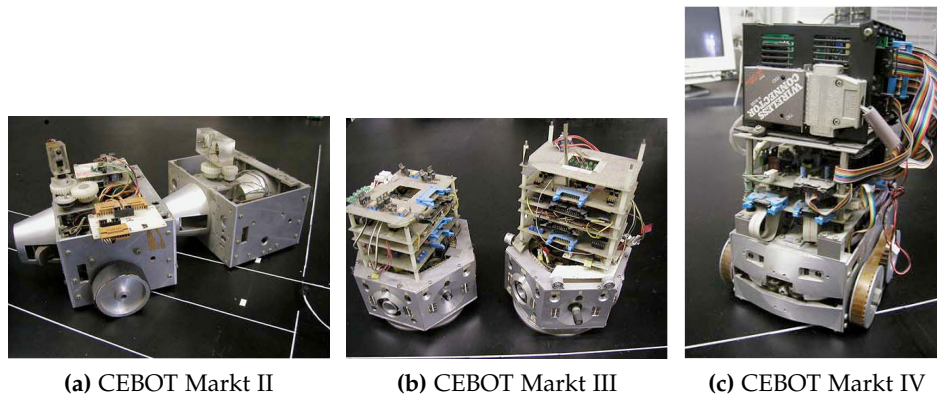


Figure 1.1

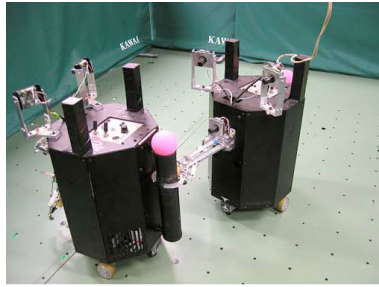
### 1.3 État de l'art

La robotique collective est un domaine vaste avec beaucoup de sous-disciplines spécifiques. Nous allons dans cette thèse nous concentrer sur la robotique collective, ayant des capacités d'interaction inter-individus et pouvant s'auto-assembler.

Dans l'article "Self-Assembly at the Macroscopic Scale" [33] Groß et Dorigo passent en revue un demi-siècle de recherches sur la conception des systèmes physiques permettant l'auto-assemblage de composants macroscopiques. Nous allons nous focaliser sur un sous-ensemble de ce domaine où les systèmes dans lesquels chaque module est auto-propulsé, s'approche et s'assemble avec d'autres modules. Dans ces systèmes, les modules peuvent être considérés comme des robots mobiles :

#### CEBOT

Fukuda et al. ont proposé le concept de robotique modulaire reconfigurable et ont réalisé la première mise en oeuvre avec le CEBOT [25,29]. Le CEBOT est un système hétérogène composé de modules avec des fonctions différentes (par exemple : bouger, se pencher, pivoter et glisser). Une série de prototypes ont été réalisés. Le premier prototype, le CEBOT Mark I [25,26], est de forme cubique avec des connecteurs actifs et passif sur les côtés opposés. Un actionneur à alliage à mémoire de forme (SMA) peut verrouiller dans un trou une goupille à partir du module de raccordement. Il a été montré qu'un module équipé de deux roues motorisées peut approcher l'arrière d'un autre module. Toutefois, cette approche s'est révélée inefficace pour le couplage des deux modules, car le mécanisme contraignant avait besoin d'un alignement précis. Pour le CEBOT Mark II [24,27]



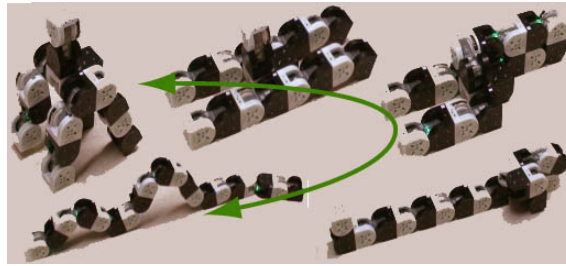
**Figure 1.2** Deux enfants de Super Mechano Colony

(figure 1.1a) et CEBOT Mark IV [11] (figure 1.1c), un crochet mécanique est utilisé pour la connexion. En outre, une partie en forme de cône est fixée sur l'avant de chaque module qui correspond à une contrepartie fixée à l'arrière pour faciliter l'alignement pendant l'approche. Avec le CEBOT Mark III [30] (figure 1.1b), les modules ont une forme hexagonale. Les six faces sont fournies avec trois connecteurs actifs et trois passifs. Le mécanisme de liaison est similaire à celui employé dans CEBOT Mark I. Les broches des connecteurs actifs sont faites d'un matériau élastique. Le module est équipé de six buses qui assurent la propulsion sur un terrain plat.

Fukuda et al. ont montré l'auto-assemblage réussi d'un module mobile avec un module statique avec la CEBOT Mark II, Mark III et Mark IV. Dans chaque expérience, les modules ont été placés à une position spécifique (à des distances allant jusqu'à 60 cm et à des déplacements angulaires jusqu'à 25°). La coordination a été réalisée en faisant usage de capteurs infrarouges.

### **Super Mechano Colony (SMC)**

Super Mechano Colony (SMC) [22, 38] (figure 1.2) est un concept modulaire robotique composé d'un module parent et de plusieurs modules enfants qui s'y rattachent. Les modules enfants font partie intégrante de la locomotion du système. En outre, les modules enfants peuvent se disperser pour accomplir des missions distinctes et autonomes, puis se reconnecter une fois les missions accomplies. Hirose et al. a présenté un premier prototype du SMC. Deux roues motorisées et deux passives assurent la mobilité sur un terrain plat. Chaque module est équipé d'un bras manipulateur qui peut être élevé avec au bout une pince. La partie supérieure du corps (y compris la pince) peut être tournée par rapport au châssis au moyen d'un axe vertical motorisé. Gross et al. [34] ont montré qu'un module enfant peut s'assembler avec un autre module enfant en l'abordant avec un angle maximum de 150°. Dans 91 des 92 essais, les modules ont correctement



**Figure 1.3** Métamorphose par des robots auto-reconfigurables M-TRAN III

établi une connexion. Une autre expérience a été menée pour étudier la formation contrôlée de structures distinctes dans un groupe de quatre modules enfants.

### **PolyBot, CONRO et M-TRAN**

Le PolyBot [79, 80], CONRO [14, 15] et M-TRAN [41, 57] sont tous des systèmes robotiques auto-reconfigurables. Leurs modules qui n'ont qu'un voire deux degrés de liberté ne peuvent pas se déplacer seuls. Par contre, un petit nombre de modules assemblés peut se mouvoir et s'auto-assembler avec un autre groupe ou un module fixe.

Yim et al. [81] ont démontré la capacité d'un bras de robot modulaire composé de six modules PolyBot G2 d'approcher et de saisir un autre module sur un terrain plat.

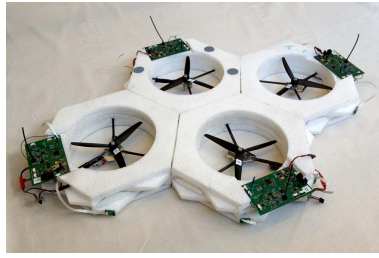
Rubenstein et al. [71] ont démontré la capacité de deux robots CONRO distincts à s'auto-assembler. Chaque robot est constitué par une chaîne de deux modules CONRO linéairement liés. Pour veiller à ce que les deux chaînes se perçoivent mutuellement, ils ont été mis en place à des distances n'excédant pas 15 cm, face à face avec un déplacement angulaire ne dépassant pas 45°.

En utilisant la plate-forme M-TRAN III figure 1.3, Murata et al. [56] ont démontré l'accostage d'un robot mobile composé de trois modules avec un robot modulaire stationnaire. La procédure s'est révélée efficace pour une variété de positions et d'orientations initiales. En outre, une séquence intégrée comprenant à la fois l'auto-assemblage et l'auto-reconfiguration a été démontrée avec 18 modules au total. Ainsi, l'entité réunie peut changer de forme en déplaçant les modules à l'intérieur de sa structure.

### **The Distributed Flight Array**

The Distributed Flight Array [63, 64] est un travail en développement. Des modules autonomes mono-hélice peuvent se décaler, s'auto-assembler avec leurs pairs et voler de manière coordonnée. Des individus munis





**Figure 1.4** Deux modules enfants de Super Mechano Colony

d'une seule hélice ne peuvent pas voler de façon contrôlée. Une fois assemblés comme un quadrirotor (figure 1.4), ils peuvent voler d'une manière stabilisée. Les modules échangent de l'information venant de leurs capteurs et déterminent individuellement la poussée à fournir pour stabiliser le groupe en fonction de leur position dans la matrice.

Il s'agit d'un système complexe et dynamique qui laisse beaucoup d'espace pour explorer diverses stratégies de contrôle distribués.

#### 1.4 Approche

Cette thèse s'est déroulée dans le cadre du projet européen FET "Swarmanoid" [23]. Dans ce projet et dans le cadre de cette thèse, nous avons développé deux robots.

Le premier robot est le marXbot et sera décrit dans le chapitre 4. Le marXbot est un robot modulaire : pour le projet "Swarmanoid" nous avons ajouté un module pour se connecter aux autres robots.

Le second robot est le handbot et sera présenté dans le chapitre 5. La particularité de ce robot est d'avoir deux pinces qui permettent la manipulation d'objets, un lanceur de corde qui permet au robot de grimper dans l'axe vertical et deux ventilateurs pour diriger le robot quand il est attaché au plafond. Le handbot ne peut pas se mouvoir sur le sol, car il sera principalement transporté par les marXbots. Pour se faire, le handbot a un anneau assorti au module de connexion du marXbot.

La combinaison de ces deux robots nous permettra d'explorer les différents aspects mécatroniques de robots hétérogènes qui effectuent des tâches collectives avancées en utilisant l'auto-assemblage.

Au cours de mes activités précédentes, j'ai développé le robot e-puck [55] et il a été produit en quantité relativement importante (400). Il sera décrit dans le chapitre 2. Il n'a pas été conçu pour étudier la robotique collective, mais plusieurs groupes de recherche l'utilisent pour des expériences collectives. L'expérience de sa fabrication à grande échelle est une bonne contribution à l'objectif de la thèse.

Avant cette thèse, j'ai également contribué au robot s-bot dans le cadre du projet "Swarm-bots". Il sera présenté dans le chapitre 3. Les travaux à succès récents avec des robots réels en robotique collective capable de s'auto-assembler sont pour la plupart fondés sur le projet Swarm-bots.

Nous allons nous intéresser aux performances de ces robots quand ils sont assemblés.

Pour finir une étude comparative de ces quatre robots au niveau mécanique, complexité et coût de fabrication nous apportera des éléments pour répondre aux questions de performances de robots collectifs pouvant s'auto-assembler.

### 1.4.1 Structure de la thèse

Les chapitres 2,3,4 et 5 présentent les quatre robots utilisés dans cette thèse. Dans chacun de ces chapitres, est présentée une description du design mécanique et électronique. La structure logicielle et les techniques de production spécifiques à chacun sont également expliquées. Puis nous ferons une analyse des caractéristiques de chacun des robots, de leur complexité, de leur coût de fabrication et de leurs capacités. Une section "leçons apprises, limites et améliorations" termine chaque chapitre consacré aux robots. Celle-ci permet de transmettre au mieux l'expérience acquise, d'avoir une esprit critique sur les robots et d'ouvrir la voie à de travaux futurs.

Le chapitre 6 présente les performances collectives des robots lorsqu'ils sont auto-assemblés. Les scénarios finaux des projets Swarmbot et Swarmanoid sont également décrits dans ce chapitre.

Dans le chapitre 7, les caractéristiques des robots sont comparées. Une analyse de leur complexité, leur coût de fabrication et la part que représente leur spécificité collective apporte un regard différent sur la robotique collective pouvant s'auto-assembler.

Ces deux derniers chapitres permettent de tirer des conclusions qui sont rassemblées dans le chapitre 8.

## 2 L'e-puck



Le robot e-puck a été développé pour l'enseignement à l'EPFL. Néanmoins, il est utilisé dans la recherche par différents laboratoires. Son faible prix, la richesse de ses capteurs et sa modularité permettent de l'utiliser en robotique collective. Après un premier design de Francesco Mondada, j'ai développé et géré une première production de 400 pièces.

Dans ce chapitre, le robot e-puck est décrit et ses caractéristiques mécaniques relatives aux tâches collectives sont analysées.

### 2.1 Design de l'e-puck

#### Cahier des charges

Le cahier des charges était simple et demandait un robot pour l'éducation à l'EPFL. Il fallait qu'il soit polyvalent, utilisable dans divers cours, simple d'utilisation et d'entretien et qu'il ait suffisamment de possibilités pour motiver les étudiants. L'idée était aussi que les étudiants apprennent sur une plate-forme robotique commune, pouvant ainsi mieux se concentrer sur le sujet que sur l'apprentissage d'un nouveau matériel. Le but était également d'avoir un grand nombre de ces robots pour que chaque groupe d'étudiants puisse avoir son propre matériel. Ainsi le prix devait rester bas, permettant à l'EPFL de se doter d'un parc suffisant de e-pucks.

L'outil ne devait pas forcément être utilisé pour enseigner la robotique, mais devait surtout être une source de motivation pour les étudiants. L'e-puck est ainsi utilisé dans des cours de robotique, de robotique collective, de traitement de signaux, d'informatique et des cours d'automatique [54].

#### Approche

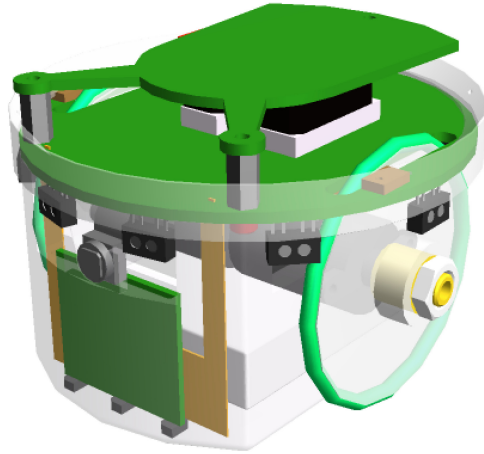
Pour avoir un robot simple d'utilisation, une approche est de le faire de petite taille, pouvant naviguer sur une table, ce qui permet de travailler à côté de l'ordinateur dans une salle de cours standard.

La mécanique doit être simple et robuste, limitant l'entretien. Ainsi des moteurs pas-à-pas avec réducteur incorporé et un axe de sortie pouvant supporter directement une roue ont été choisis pour leur robustesse et leur faible coût. L'e-puck est également pourvu d'une batterie Li-Ion interchangeable, garantissant ainsi une bonne autonomie et une utilisation en continu lors d'une séance de cours.

Le robot est également équipé d'un système radio, permettant de facilement interagir avec un ordinateur. Ainsi, une liaison standard Bluetooth<sup>®</sup> à été choisie.

Pour permettre une grande diversité éducative, un panel de capteurs a été incorporé au robot. Ce panel comporte un choix de capteurs de types différents, avec des bandes passantes variées et un flux de données plus ou moins important.

Le choix du microcontrôleur devait permettre une utilisation variée et être facilement programmable. Le choix s'est porté sur un dsPIC30 muni de deux coeurs : un microprocesseur normal et un coeur DSP. Ce microcontrôleur est doté d'un large choix de périphériques facilement utilisables par le coeur normal. Le coeur DSP permet de faire du traitement de signal poussé, comme par exemple une FFT. Le dsPIC30 est muni d'un



**Figure 2.1** Dessin CAD de l'e-puck. (3D)

set d'instruction assembleur simple à comprendre et peut également être programmé en C grâce au compilateur gcc qui est très répandu.

Pour finir, certains laboratoires voulaient avoir des caractéristiques très spécifiques pour leurs cours. Ainsi, l'e-puck a été, dès le départ, équipé de connecteurs d'extension permettant l'ajout de tourelles additionnelles.

### 2.1.1 La mécanique

Le design mécanique est simple et robuste (figure 2.1). Il est facile à monter et adapté pour une production de masse. Le robot se compose d'un châssis principal en plastique (figure 2.2) sur lequel le reste des composants est fixé. Deux moteurs pas à pas, avec réducteurs, sont vissés sur ce châssis. Les roues sont fixées directement sur l'arbre de sortie du moteur. Le châssis comporte une ouverture en-dessous pour la batterie. A l'avant du robot, le châssis comporte une fente où l'on peut glisser une extension comme des capteurs de sol.

Le circuit électronique principal vient se fixer sur le châssis grâce à trois vis. Des contacts pour la batterie sont directement soudés sur ce circuit. Un cercle diffuseur pour les LED est enchâssé sur ce circuit électronique et le

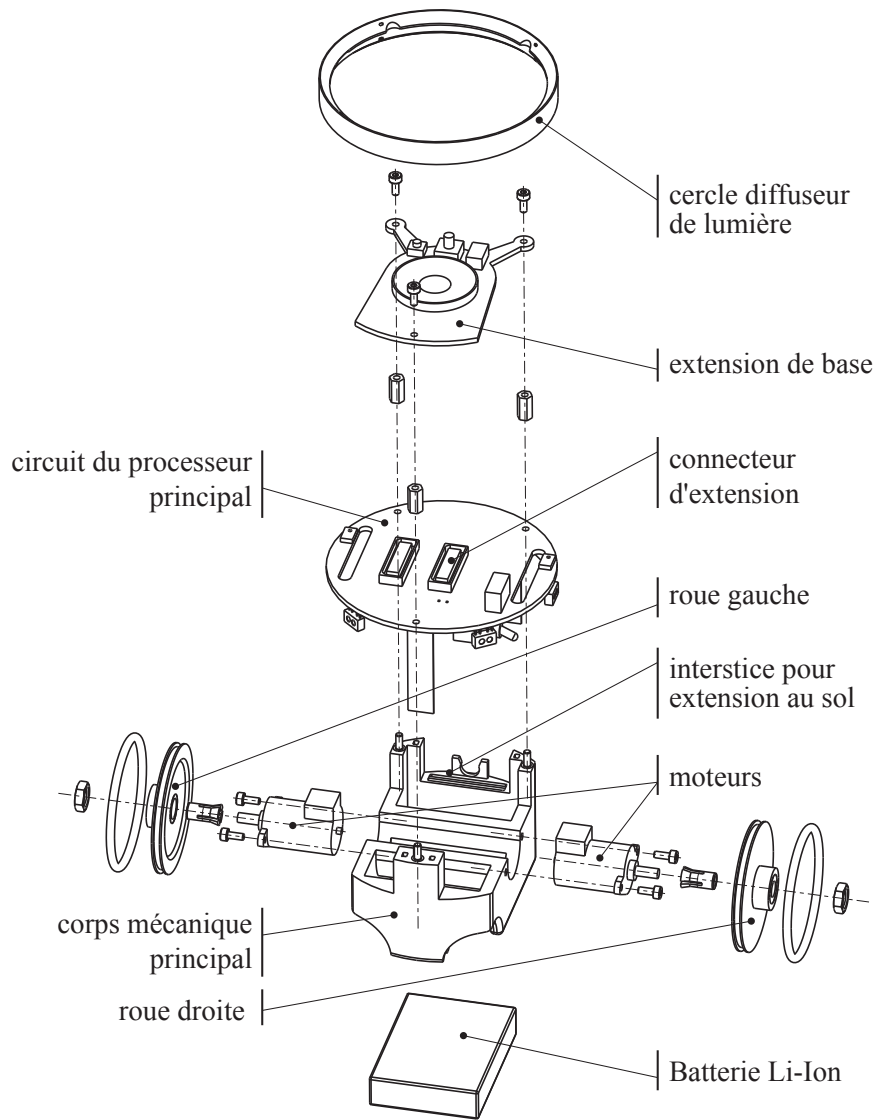


Figure 2.2 Explosé d'un e-puck

protège. Au-dessus de ce circuit principal, différentes extensions peuvent s'emboîter. La fixation mécanique est assurée par trois entretoises.

### La mobilité

L'e-puck est un robot avec deux roues différentielles et un point d'appui à l'arrière du robot. Sa garde au sol est de 0.8mm. L'e-puck ne peut que se déplacer sur une surface lisse et plane, dépourvue d'aspérité (pas de moquette, bois, carton). Un bloc de 5 feuilles peut déjà bloquer le robot. Le robot est propulsé par deux moteurs pas à pas avec réducteur à 1000 pas/tour. La vitesse maximale est 1000 pas/s ce qui correspond à 12.7 cm/s. Après une calibration appropriée, l'odométrie peut être caractérisée très précisément [50].

### 2.1.2 L'électronique

L'électronique principale de l'e-puck contient les principaux besoins pour ce type de robot. L'architecture est structurée autour d'un microcontrôleur DSPIC30 (figure 2.3). Ce microcontrôleur gère ainsi :

- 2 moteurs pas à pas
- 10 LED (8 rouges en périphérie, 1 frontale, 1 verte illuminant le corps)
- 8 capteurs de proximité infra-rouges
- 1 accéléromètre 3 axes
- 3 microphones
- 1 haut-parleur
- 1 caméra couleur CMOS VGA
- 1 sélecteur de programme 16 positions
- 1 module récepteur IR pour télécommande

L'e-puck contient aussi une liaison Bluetooth<sup>®</sup>, une connexion port série, un connecteur de programmation et de débogage. Deux connecteurs d'extension sont placés sur l'e-puck (figure 2.4). Le premier transmet ainsi certains signaux, comme ceux de la caméra et l'alimentation. Le second connecteur est un relais entre certains signaux et le microcontrôleur. On peut ainsi choisir en changeant l'extension de couper ou non ces signaux et faire ainsi différentes utilisations des signaux ou des entrées du microcontrôleur. Ces connecteurs permettent le design d'un grand choix de tourelles d'extension, ce qui donne un atout intéressant pour le collectif à l'e-puck.

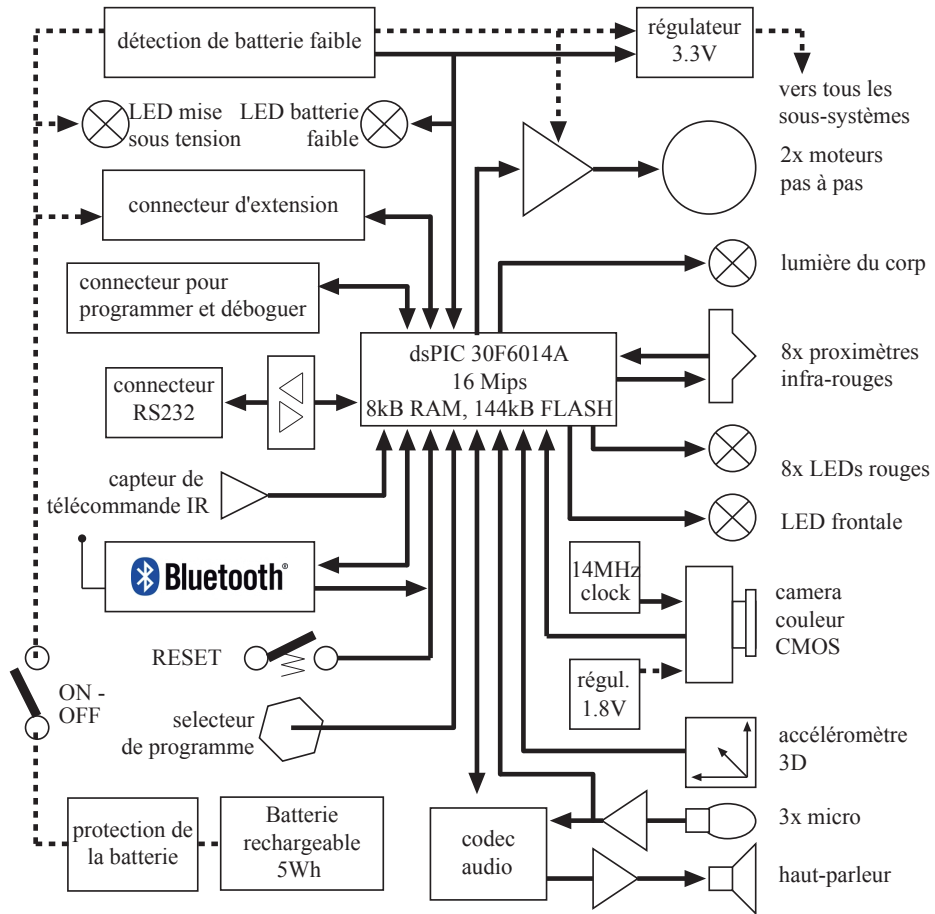


Figure 2.3 Schéma de l'électronique d'un e-puck

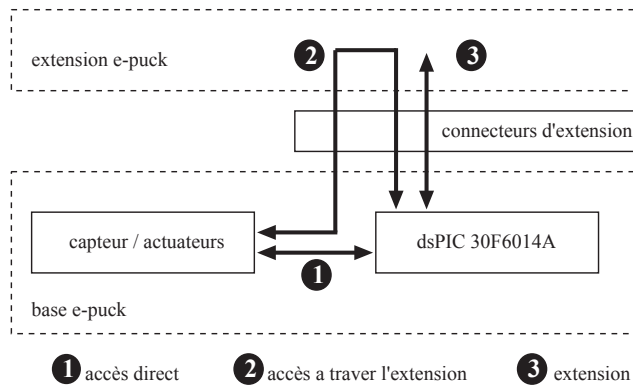
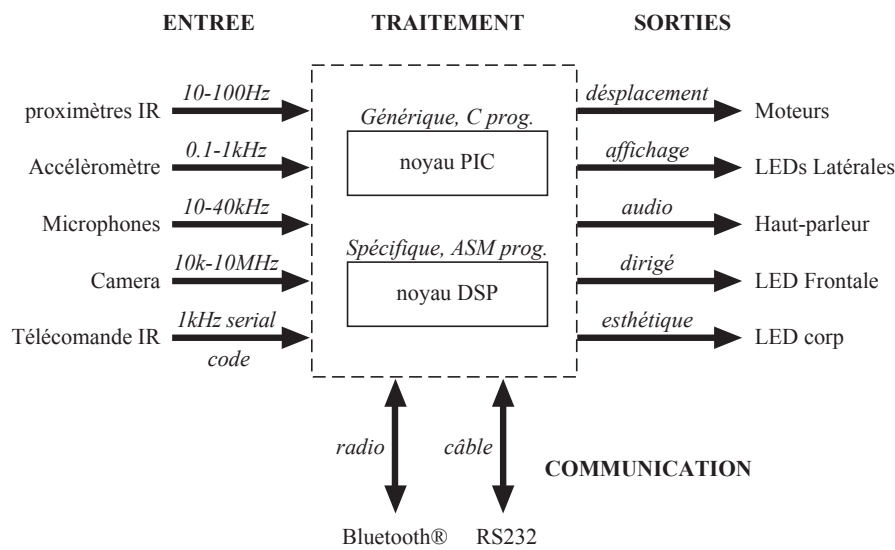


Figure 2.4 Schéma des signaux et extensions





**Figure 2.5** Les capteurs et actionneurs de l'e-puck classés par leurs caractéristiques

### L'énergie

Les caractéristiques de la batterie de l'e-puck sont présentées dans la table ci-dessous :

Type	No Référence	Capacité	Autonomie	T. Charge
1×Li-Ion	HLP103448	1400mAh 3.6V 5Wh	2-4h	2h

La batterie de l'e-puck est interchangeable. Elle permet de faire des expériences tout au long de la journée si le nombre de batteries est suffisant.

### Les capteurs

La figure 2.5 résume les entrées sensorielles de l'e-puck en fonction de leur bande passante.

Les 8 capteurs de proximité infra-rouges permettent de mesurer la distance jusqu'aux obstacles autour du robot de 0 à  $\approx 4$  cm. Ils peuvent aussi mesurer l'intensité de la lumière infra-rouge ambiante. Ce sont des capteurs typiques pour la navigation en robotique dans des environnements circonscrits. En collectif, ils permettent de situer les coéquipiers et même de communiquer.

Un accéléromètre 3 axes fournit le vecteur accélération de l'e-puck. Dans les situations statiques, ce vecteur représente la gravité et peut être utilisé

pour détecter l'inclinaison de l'e-puck. En lisant l'accélération du plan horizontal, l'e-puck peut détecter les collisions ou l'accélération produite par son propre mouvement. Si le vecteur accélération est nul, l'e-puck peut détecter qu'il est en chute libre. En collectif, l'accéléromètre a été utilisé comme boussole virtuelle [58]; les e-pucks se déplaçaient sur une surface verticale métallique et étaient dotés de roues magnétiques, la gravité fournissait une direction de référence commune au groupe de robot.

Les trois microphones permettent à l'e-puck de localiser la source du son par triangulation. La partie DSP de l'e-puck permet facilement de faire une transformée de Fourier et ainsi traiter les données produites par ces trois micros. Les systèmes sonores sont utilisés en robotique collective pour la communication et la localisation.

La caméra CMOS couleur avec une résolution de  $640 \times 480$  pixels à l'avant de l'e-puck permet l'expérimentation de la vision. Seule une sous-partie de l'image peut être saisie : la taille de l'acquisition est limitée par la taille de la mémoire du dsPIC et le taux limité par sa puissance de calcul. Tous les formats de sous-images sont acceptés tant que ces deux contraintes sont remplies. Par exemple, l'e-puck peut obtenir une image couleur  $40 \times 40$  pixels à 4.3 fps ("frame" ou image par seconde); la fréquence est doublée en niveau de gris.

### 2.1.3 Les techniques de production

Le design de l'e-puck a été pensé pour réaliser facilement une petite production avec des moyens limités dans les laboratoires de EPFL.

Le choix de faire un corps en plastique injecté fut assez novateur dans le domaine; non pas parce que la technique soit nouvelle mais plutôt de par son utilisation pour une petite série de quelques centaines de pièces. Aussi la pièce plastique a été utilisée comme corps et support de toutes les autres, ce qui est différent des châssis habituels avec capot de protection. L'injection permet ainsi de faire une pièce relativement complexe mais qui, par la suite, limite le nombre total de pièces et facilite le montage. De plus, la technique de surinjection d'insert métallique a été utilisée. Le corps a ainsi trois vis M2 prises dans le plastique qui font office de support pour le circuit principal et les extensions. Le choix de vis métriques permet une meilleure utilisation des extensions et contrairement aux vis auto-taraudeuses pour plastique, elles sont démontables plusieurs fois. Le moyeu des roues est aussi une pièce surinjectée. Pour le cercle diffuseur, le sablage a été directement inclus au moule, cependant, cela augmente les dépouilles (angles pour le démoulage).

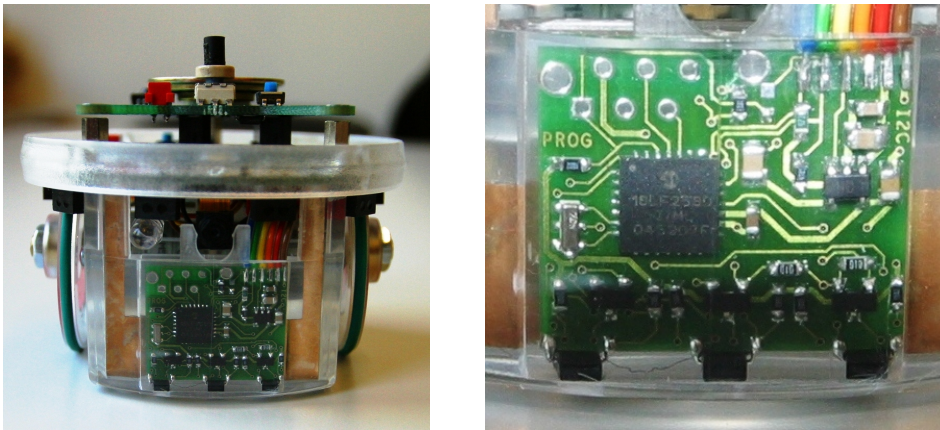


Figure 2.6 L'extension capteur de sol

L'électronique de l'e-puck n'a rien de révolutionnaire, si ce n'est le nombre élevé de capteurs et d'actionneurs inclus dans le circuit principal. Ce circuit est assez chargé en composants et a nécessité la technologie 4 couches. Pour la fabrication en série, les composants SMD ont été privilégiés, ainsi l'électronique peut être montée par n'importe quel assembleur conventionnel.

La batterie de l'e-puck est une Li-Ion sortie d'usine sans protection. Ce n'est donc pas un pack commercial. L'électronique de protection a donc été incluse dans le circuit de base, ce qui réduit fortement le coût d'un pack batterie et est donc avantageux pour une utilisation intense par les étudiants. Cependant, il faut concevoir un chargeur spécifique.

#### 2.1.4 Les extensions

Dans cette partie, les différentes extensions utiles pour la robotique collective sont présentées.

Une extension de l'e-puck qui est très utilisée dans divers travaux scientifiques présentés dans la section 2.3 est le capteur au sol (figure 2.6)<sup>1</sup>. A l'avant du robot, il y a une interstice dans la pièce principale. On peut y glisser un circuit électronique comportant des capteurs infra-rouges dirigés vers le sol. Cela permet de faire, par exemple, de la détection de bord de table ou de la détection de ligne. En collectif, on l'utilise aussi pour démarquer simplement différentes régions d'une arène avec des zones à niveaux de gris différents.

---

1. Développé au LIS par Adam Klaptocz

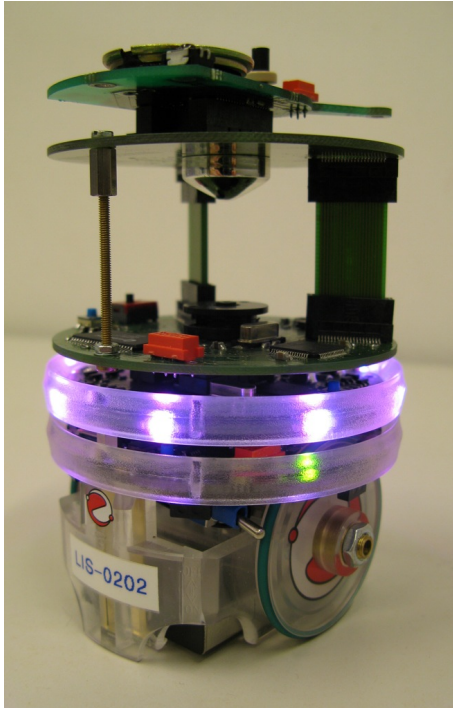


Figure 2.7 La tourelle omnivision

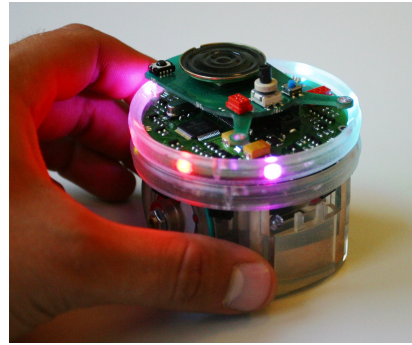


Figure 2.8 La tourelle LED RGB

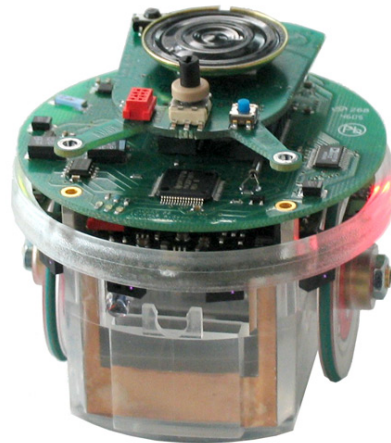


Figure 2.9 La tourelle communication

En robotique collective, savoir où sont ses pairs est souvent nécessaire. Ainsi, l'extension vision omnidirectionnelle<sup>2</sup> (figure 2.7) permet facilement de détecter d'autres e-pucks avec leurs LED allumées. En addition, l'extension LED RGB<sup>3</sup> (figure 2.8) permet également de transmettre des messages colorés. La communication en robotique collective est un besoin prépondérant. Dans la section 2.2, les différents canaux de communication de base sont présentés. En addition à ces canaux, une extension communication radio compatible IEEE 802.15.4 (Zigbee)<sup>4</sup> a été développée (figure 2.9). Elle a la particularité de pouvoir diminuer la puissance de la radio par une électronique spécifique. Cela permet d'expérimenter en réel

2. Développée au LIS par Adam Klaptocz <http://www.e-puck.org> → Extensions → Omnidirectional Vision Turret

3. Développée au LIS par Adam Klaptocz <http://www.e-puck.org> → Extensions → Colour LED Communication Turret

4. Développée au SWIS par Xavier Raemy

divers comportements de "swarm-robotic" liés à l'échange d'informations en réseau local [19].

## 2.2 Caractéristiques de l'e-puck

Dans cette section, les caractéristiques de l'e-puck sont présentées. Il s'agit d'avoir un point de comparaison avec d'autres robots et particulièrement ceux présentés dans cette thèse.

### Les degrés de liberté (ddl)

Dans sa configuration de base, l'e-puck a deux roues et donc deux degrés de liberté. Cela permet de se déplacer mais pas d'interagir physiquement avec ses coéquipiers ou avec l'environnement. D'autres ddl peuvent être ajoutés dans les extensions.

### Les informations sensorielles

Voici un tableau résumant les débits de données de chaque type de capteur :

Types	Données brutes	Signal utile	Fréquence	Data/s
8×IR	8×12bits	10bits	64Hz	5.1kbit/s
8×IR ambient	8×12bits	10bits	64Hz	5.1kbit/s
3D accéléromètre	3×12bits	10bits	256Hz	7.6kbit/s
3×micros	3×12bits	9bits	16.4kHz	443kbit/s
caméra couleur	2028p 16bits RGB565	13bits	4.3Hz	(113kbit/s)
caméra n. gris	4056p 8bits	6bits	8.6Hz	209kbit/s
Total				715kbit/s

L'unique microcontrôleur dsPIC30 tourne à 15MIPS. C'est un processeur 16bit avec un coeur DSP.

La caméra de l'e-puck peut normalement fournir une image de 640×480 pixels à 30fps. Cependant, le microcontrôleur dsPIC30 n'est pas du tout adapté pour acquérir tant de données. Le convertisseur A/D du DSPIC30 est cadencé à 100kHz. Ainsi, si l'on ne prend que les micros, sans les capteurs de proximité IR et l'accéléromètre, on peut les échantillonner à 33kHz. Avec son coeur DSP, il est adapté pour ce type de débit de données.

### La communication

L'e-puck de base a différents canaux de communication qui peuvent être utilisés. Premièrement avec Bluetooth<sup>®</sup>, un e-puck peut se connecter à l'un de ses pairs. Ceci demande, cependant, une communication "maître-esclave", non optimale en robotique collective. De plus, un maître ne peut se connecter qu'à 7 esclaves. La connectivité Bluetooth<sup>®</sup> a surtout été ajoutée pour interfacier l'e-puck avec un ordinateur. Deuxièmement, l'e-puck peut utiliser ses capteurs de proximité comme émetteurs-récepteurs infra-rouges. Ainsi en modulant des données sur deux fréquences, un e-puck peut transmettre des données localement à ses semblables<sup>5</sup>. On peut même avoir une idée de la position et de l'orientation de l'e-puck émetteur [36]. Suivant le même principe, une tourelle avec une électronique dédiée a été développée [35]. Troisièmement, l'e-puck a de base trois microphones et un haut-parleur. Ils peuvent être aussi utilisés comme canal de communication [1] et la direction de l'émetteur peut aussi être calculée [12]. Cependant, l'haut-parleur de base n'a pas une bonne dispersion omnidirectionnelle et manque un peu de puissance. Une cage de résonance améliorerait la distance de communication.

### Le prix

Le prix d'achat d'un e-puck est aujourd'hui de 850 Frs. Le tableau suivant recense les prix de revient lors de la première production de 400 pièces. Probablement avec les mises à jours des derniers circuits, le coût est légèrement plus faible.

	Coûts en Frs
Mécanique	16.-
Moteurs	26.-
Électronique	311.-
<b>Total</b>	<b>353.-Frs</b>

L'électronique représente 90% des coûts car :

- L'injection des pièces plastiques devient très rentable pour un grand nombre de pièces.
- Les moteurs pas à pas munis d'un réducteur de qualité moyenne et l'absence d'encodeur permettent d'avoir un prix très raisonnable.
- Les nombreux capteurs, le Bluetooth<sup>®</sup>, les connecteurs d'extension et le microcontrôleur représente près de 60% des coûts de l'électronique.
- L'électronique est fiable et adaptée aux étudiants.

---

5. <http://www.e-puck.org> → Software → IR Communication

Pour conclure, c'est la grande capacité sensorielle, ses moyens de communication et sa capacité d'extension qui augmentent le prix de l'e-puck. C'est aussi ces mêmes capacités qui le rendent intéressant pour la robotique collective.

### **La distribution**

Pour favoriser sa distribution et les développements futurs, les caractéristiques mécatroniques et les logiciels de l'e-puck sont publiés sous une licence libre du style "open-hardware". Cela a contribué à l'intérêt de différentes universités pour ce robot. Pour l'instant 400 e-pucks ont été fabriqués initialement à l'EPFL, 1250 vendus par GCtronic et  $\approx 900$  par la compagnie japonaise AAI. On estime à plus de 200 laboratoires, entreprises ou écoles qui ont des e-pucks (sans le Japon).

Sur les 1250 e-puck de l'entreprise GCtronic, 4 sont revenus pour des problèmes matériel et 5 ont dû être reprogrammés. Cela prouve sa grande fiabilité et robustesse.

### **2.3 L'e-puck en recherche sur la robotique collective**

Ce chapitre présente un résumé non exhaustif de l'utilisation du robot e-puck en recherche collective. Seuls les travaux utilisant les robots réels ont été pris en compte. Cependant, un grand nombre de travaux sont faits avec des e-pucks simulés grâce à la simplicité du robot qui favorise son intégration dans divers simulateurs [1, 3, 47, 54].

Dans [19], l'e-puck est équipé d'une tourelle de communication compatible IEEE 802.15.4 (Zigbee). Il est ainsi utilisé pour introduire différents concepts de "swarm robotic" aux étudiants. Dans [20], l'e-puck avec cette même tourelle est utilisé pour l'inspection avec des comportements collectifs. L'e-puck est utilisé dans [77] pour démontrer la faisabilité d'un service web qui facilite l'interaction d'un groupe hétérogène de robots. Dans [75], l'e-puck est utilisé dans deux expériences utilisant les techniques d'évolution dans le cadre de robotique collective. Les e-pucks coordonnent leur orientation en utilisant la communication locale et de l'infra-rouge [36]. Dans [58], les e-pucks sont utilisés pour des algorithmes de déplacement en troupeau, utilisant notamment l'accéléromètre comme boussole sur une surface verticale (des aimants sont fixés sous l'e-puck) et de la communication local infra-rouge. Dans [40], l'e-puck est utilisé dans une expérience de "multi-agent path planning". Les auteurs de l'article [49] utilisent l'e-puck pour démontrer l'efficacité d'un nouveau langage de script inter-microcontrôleur appelés ASEBA [48]. Dans [62], l'e-puck est utilisé dans une expérience de "multi-robot path planning for box-pushing".

L'e-puck est muni d'un scanner laser de marque Hokuyo pour des expériences de mouvements en groupe [73]; une preuve de la robustesse de la mécanique de l'e-puck car ce capteur est aussi lourd qu'un e-puck. Dans [18], les e-pucks sont munis d'une lampe et doivent se réorganiser sur une table pour créer des formes définies. L'impact sur la localisation de différentes contraintes sur la communication est étudié. Dans [44], l'e-puck est muni d'une extension reproduisant le dépôt de phéromones. Grâce à ce système, des expériences qui émulent la recherche de nourriture de fourmis sont étudiées.

Sur les  $\approx 70$  articles publiés à ce jour sur l'e-puck, 18% présentent l'utilisation de l'e-puck dans des expériences collectives.

### 2.4 Leçons apprises, limites et améliorations

L'expérience e-puck a été très enrichissante pour le laboratoire et moi-même. En premier lieu, nous avons exploré la fabrication à bas prix en Chine pour toute la mécanique. En effet la fabrication d'un moule d'injection y est près de 10 fois moins chère qu'en Europe. Cet avantage sur le coût a vraiment permis de limiter le prix de fabrication de l'e-puck. Cependant, il a fallu s'adapter à leur mode de fonctionnement et leur mentalité différente. Un plan normé n'y est pas forcément suivi ou compris et le travail se fait plutôt de la manière "essai et correction d'erreur". Leur réussite dans un projet tient à leur grande capacité à abattre du travail plutôt qu'à suivre une réflexion ou un plan. Néanmoins, après de nombreux essais, une production a pu y être maîtrisée<sup>6</sup>. Ainsi, le coût de fabrication peuvent être diminués. Par contre, cela demande un effort non négligeable au niveau de l'encadrement et de l'organisation.

Pour l'électronique, nous avons préféré garder la production en Suisse. En effet, l'électronique est une partie délicate et importante du robot; la proximité géographique permettait de mieux réagir aux aléas d'une mise en production. De plus, une fois montés, il est difficile de garantir que tous les composants électroniques sont conformes. Depuis la première production, l'électronique a également évolué; soit pour corriger des "footprint" qui ne facilitaient pas le montage, soit pour remplacer des composants obsolètes.

Le montage de l'e-puck n'est pas très long et nécessite que peu de main d'oeuvre. Cependant, ses nombreux capteurs et son électronique dense induisent qu'un certain nombre de circuits électroniques ne sont pas totalement fonctionnels. Il faut passer beaucoup de temps pour tester chaque e-puck et il faut prévoir un temps de débogage lors de la production.

---

6. Notre contact Omar Benjeloun de Invest Group a été une aide précieuse pour la production.



En général la fabrication de l'e-puck, nous a appris à gérer une production et à réagir à quantité d'imprévus.

L'e-puck n'a que peu de limites car il est modulaire et chaque manquement peut être corrigé par l'adjonction d'un module adapté. Les limites connues sont sa mobilité limitée à une surface totalement plane et propre. La garde au sol est minimale et une simple aspérité peut bloquer le robot. Les capteurs infra-rouges perçoivent également un peu le bord du cercle diffuseur des LED. Il lui est également difficile de distinguer un autre e-puck car le corps est transparent. Les contacts de la batterie ne sont parfaits ; les ressorts sont à peine comprimés et la force de contact n'est pas optimum. Nous avons fait le choix de prendre des moteurs pas à pas pour des raisons de coûts et pour ne pas avoir d'encodeur sur les moteurs. Cela se paie au point niveau de la consommation, qui est importante pour un robot de cette taille.

Les améliorations futures pour l'e-puck seront de s'adapter aux dernières technologies et d'essayer de diminuer encore le coût de l'électronique en choisissant des composants meilleurs marché.

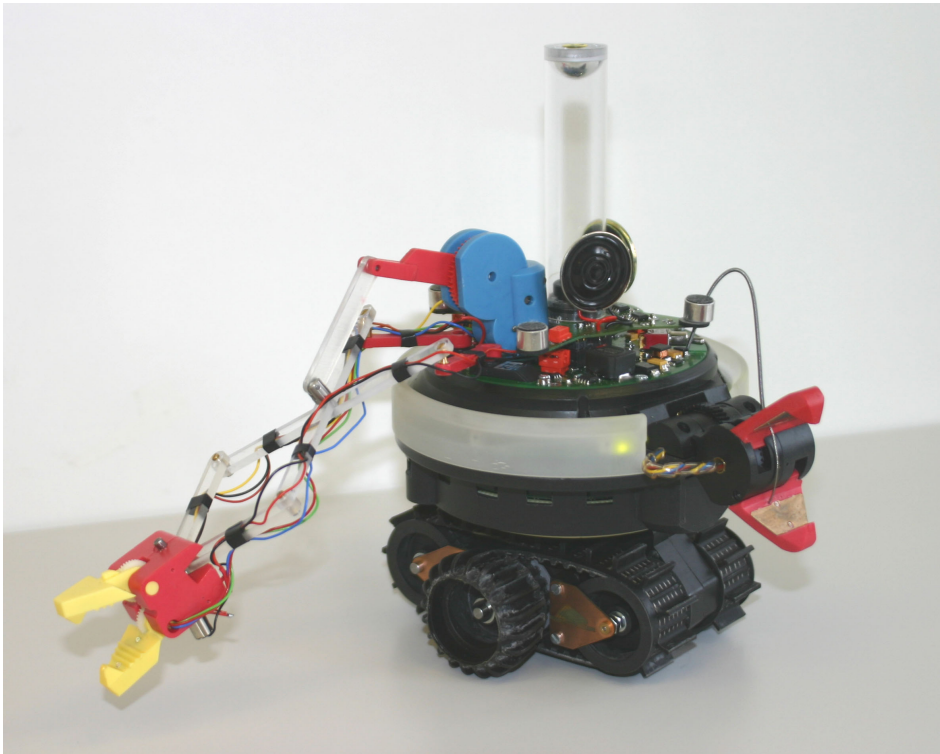
Dans un autre registre, le fait que l'e-puck soit en "open-hardware" suscite des réactions inattendues. Le produit gagne en sympathie et les extensions et améliorations suivent leur cours. Cependant, cela implique de répondre à une communauté qui s'attend à recevoir un soutien gratuit efficace. Le site internet doit être fonctionnel et exige rapidement une très grande disponibilité pour le maintenir. Il ne suffit pas de faire un site communautaire pour que tout fonctionne de manière autonome. Un autre aspect inattendu est que peu de personnes ont utilisé le gratuit des schémas de l'e-puck. Il n'y a que deux entreprises qui se sont lancées dans la production des e-pucks et elles l'ont surtout fait parce qu'elles étaient proches du laboratoire et pouvaient facilement profiter de notre expérience. Lancer et gérer une production n'est pas facile et beaucoup préfèrent acheter le produit fini et fonctionnel. Contrairement au logiciel, le matériel n'est pas reproductible sans coûts. Ainsi, faire de l'"open-hardware" permet de gagner en sympathie et ne signifie pas la perte de contrôle du produit, car pour le reproduire les entreprises ont besoin de nos conseils et de notre expérience.

### **2.5 Conclusion**

L'e-puck, grâce à son faible coût, la richesse de ses capteurs, ses moyens de communication variés est un outil pouvant être utilisé en robotique collective. Le design adapté à des techniques de production de masse permet d'avoir un grand nombre d'e-pucks pour un faible prix. La modularité possible grâce à ces connecteurs d'extension et la licence libre permettent

facilement d'ajouter du matériel pour augmenter les capacités de l'e-puck et pour en faire un robot collectif complet.

### 3 Le s-bot



Développé dans cadre du projet Swarm-bots, le robot s-bot est l'un des robots collectifs pouvant s'auto-assembler les plus aboutis. Le design mécanique est l'oeuvre d'André Guignard et de Francesco Mondada. J'ai surtout collaboré au design électronique et à la programmation de bas niveau, puis géré la production de 35 pièces. Le s-bot m'a surtout permis d'acquérir une grande expérience dans la conception mécatronique de robots.

La richesse de ses capteurs, son intégration et son système d'accrochage permettent aux utilisateurs du s-bot de concevoir une grande variété d'expériences en robotique collective. Dans ce chapitre, le design du s-bot sera présenté, ainsi qu'une analyse de ses atouts comme robot collectif. Une expérience d'auto-accrochage est décrite.

Le robot s-bot a été conçu dans le cadre du projet Européen Swarm-bots, projet en partenariat entre l'École Polytechnique Fédérale de Lausanne (Suisse), l'Université Libre de Bruxelles (Belgique), l'Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (Suisse) et le Consiglio Nazionale delle Ricerche de Rome (Italie).

### 3.1 Design du s-bot

#### Le cahier des charges

Le cahier des charges du robot était élémentaire : permettre des expérimentations avec des robots réels dans le projet Swarm-bots. Le but de ce projet était d'étudier l'impact de l'auto-assemblage en robotique collective. L'auto-assemblage est un comportement qui peut s'observer en nature chez des insectes sociaux comme les fourmis [2,45]. Au début du projet Swarm-bots, il n'y avait pas d'indications précises sur comment réaliser l'assemblage et la coopération robotique. Le cahier des charges s'est donc défini au fur et à mesure que le projet progressait.

#### L'approche

Le design du s-bot a commencé par une étude des méthodes d'accrochage, les formes géométriques permettant des formations de robot et le nombre de connections utiles à la création de ces configurations (figure 3.1). Les choix ont été faits au niveau du consortium, essayant de tenir compte des différents aspects de contrôle, d'utilité scientifique et surtout de mise en oeuvre.

Dans les méthodes d'accrochage, plusieurs approches ont été évaluées : VELCRO, magnétique, crochets mécaniques, pincement mécanique. C'est finalement ce dernier qui a été choisi pour des raisons de simplification du contrôle, d'économie d'énergie et de symétrie entre l'accrochage et le décrochage. Au niveau morphologique, le choix s'est porté sur un cylindre, qui laisse une forte marge de manoeuvre et permet des configurations très variées. Le nombre et le type de connections a été largement discuté au niveau du projet. Un nombre faible de connections simplifiait le robot, déjà très complexe, mais limitait les possibilités. Un grand nombre de connections rendait le robot très complexe et difficilement gérable, tout en ouvrant plus de possibilités. Après discussion il fallait que le type de connexion soit : ou élastique et faiblement contrôlable, ou rigide et bien contrôlable. Pour finir, le choix a été un compromis : deux connections au total, une rigide avec un degré de liberté vertical et l'autre un peu plus flexible, avec trois degrés de liberté sous forme de bras.

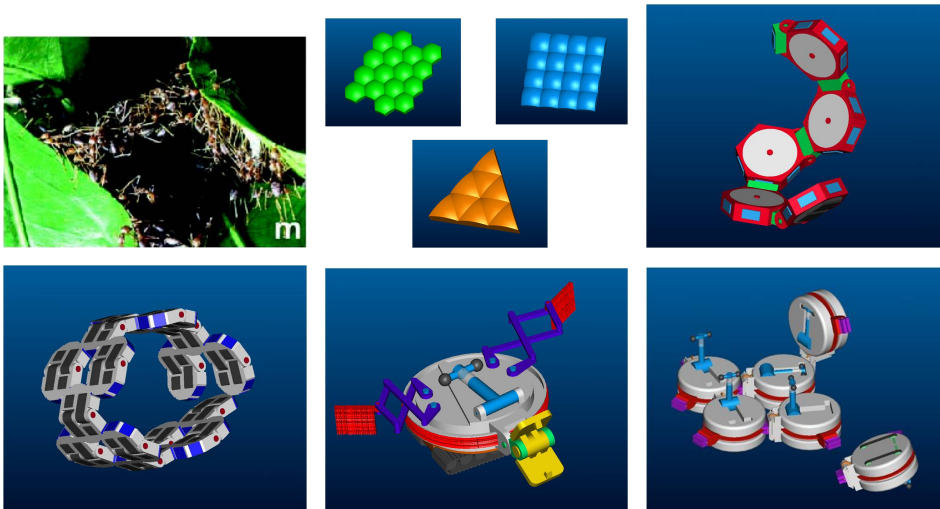


Figure 3.1 Divers concepts imaginés au début du projet Swarm-bots

Un choix délicat concernait la mobilité. Il semblait évident de pouvoir évoluer dans un terrain naturel d'une part, mais une bonne maniabilité était importante. Des chenilles étaient avantageuses pour le tout-terrain, des roues étaient avantageuses pour la précision de mouvement. Le résultat, après différents essais, a été une combinaison des deux, décrite en détail dans la section 3.1.1.

La taille des robots a été voulue la plus petite possible car cela permettait d'avoir des zones d'expérimentations petites. A l'époque également, le consortium expérimentait avec les robots Khepera d'un diamètre de 6cm et faire un robot plus grand était peu envisageable. En outre, plus le robot était petit, plus l'analogie avec les fourmis prenait sens. Ainsi, pour incorporer toutes les fonctions, l'intégration et la miniaturisation ont été très poussées.

Finalement un choix fondamental devait se faire quant à l'architecture électronique du robot. Le s-bot a un nombre important de capteurs et d'actionneurs. Certaines fonctionnalités, comme la vision ou la gestion du son, nécessitent une puissance de calcul suffisante. Pour les besoins du projet, le système devait pouvoir se programmer par des informaticiens dans des environnements de programmation connus et confortables. Somme toute, le robot devait se prêter à une expérimentation collective, ce qui implique de pouvoir suivre l'évolution de chaque robot en temps-réel, de pouvoir enregistrer les données expérimentales et de pouvoir interagir avec le groupe de robots. Les derniers points ont clairement orienté le choix d'avoir un ordinateur avec Linux embarqué et équipé de Wifi. Les capteurs et actionneurs ne pouvaient pas être tous connectés à cet ordina-

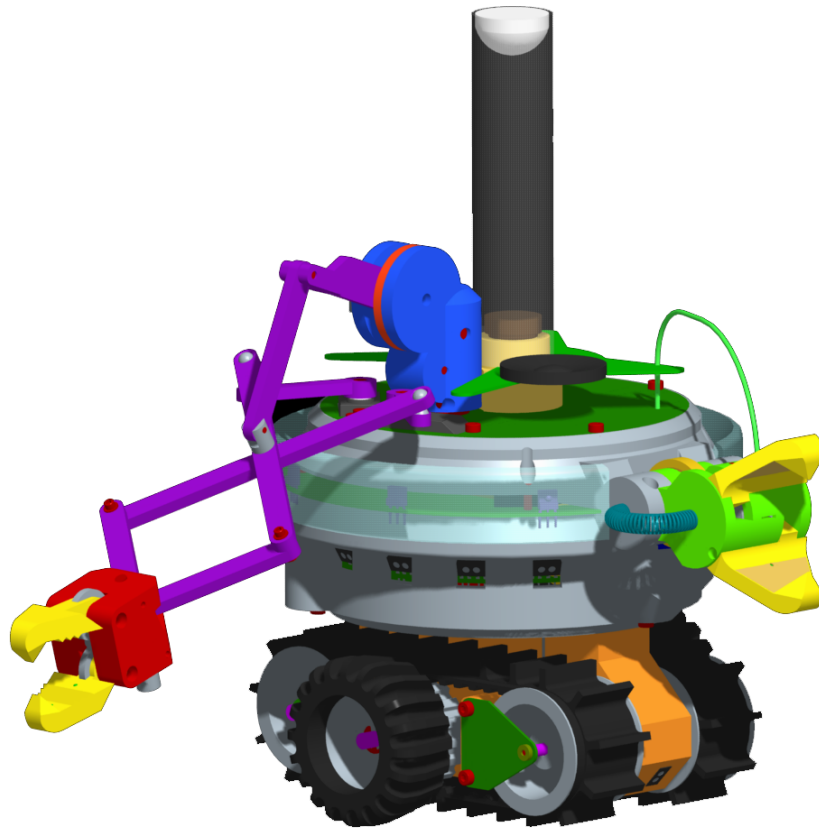


Figure 3.2 Dessin CAD du s-bot.(3D)

teur central. Le choix a donc été fait de contrôler capteurs et actionneurs par des microcontrôleurs dédiés et reliés au processeur central par un bus I<sup>2</sup>C. Un système de modules de ce type existait déjà à l'EPFL, ce qui a eu une influence sur le choix des microcontrôleurs et du bus de communication.

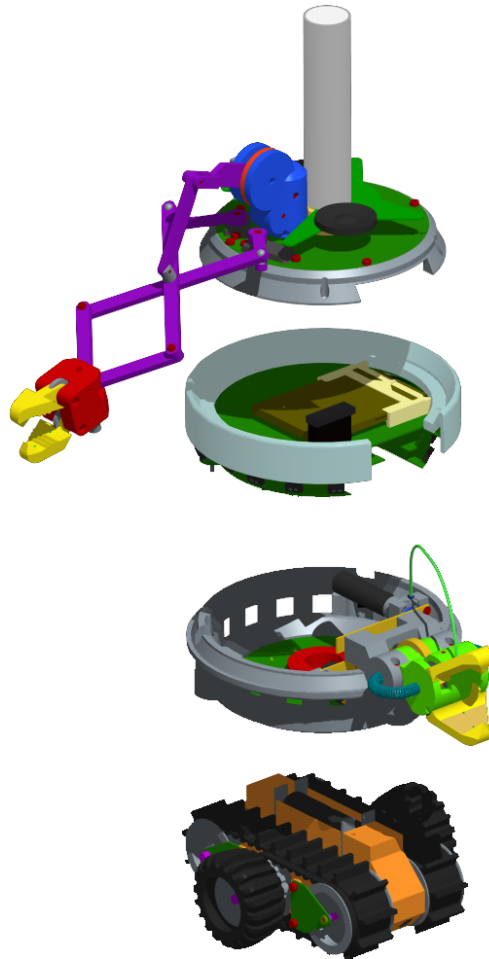
### 3.1.1 La mécanique<sup>1</sup>

Le robot a un corps cylindrique avec un diamètre de 12cm (figure 3.2). Les chenilles et les roues sont plus grandes que le corps, permettant l'approche des parois verticales.

Le robot s-bot a deux pinces. L'une rigide avec un degré de liberté ayant suffisamment de couple pour soulever un autre robot, l'autre plus mobile, avec trois degrés de liberté et une architecture parallèle. Un anneau

---

1. Développée en grande partie par André Guignard.



**Figure 3.3** Le s-bot divisé en quatre sous-parties : la base, la tourelle, l'électronique, le haut

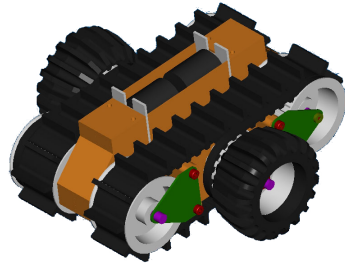


Figure 3.4 La base du s-bot

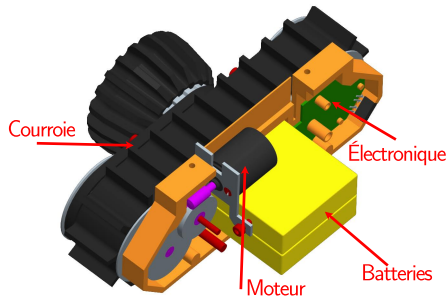


Figure 3.5 Le demi-châssis du s-bot

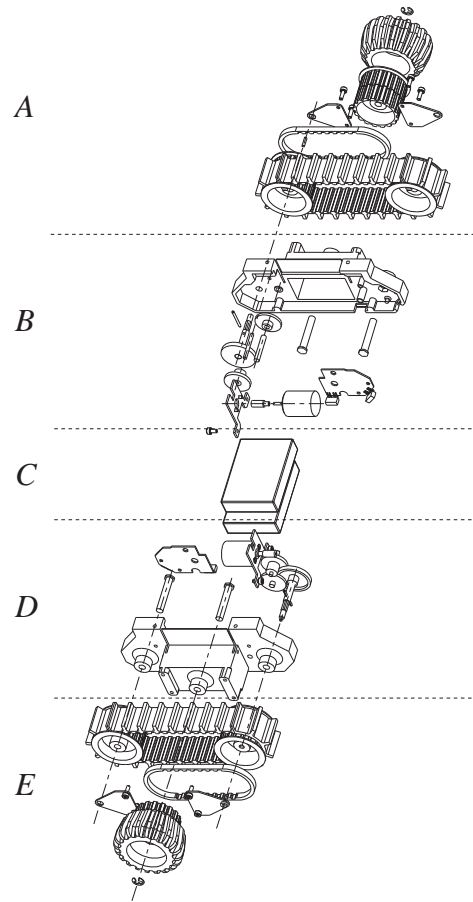


Figure 3.6 Plan éclaté de la base

d'attache lumineux permet au s-bot de s'accrocher à un autre robot grâce à l'une de ses pinces. La plus mobile pouvant être utilisée pour prendre des objets.

Le corps peut tourner par rapport au châssis, ce qui permet au s-bot d'orienter les chenilles une fois les robots assemblés. Pour simplifier l'explication du robot, on va le diviser en 4 parties (figure 3.3) :

- la base, avec les chenilles et les batteries
- la tourelle et la pince rigide
- le coeur de électronique
- le haut, avec la pince mobile et la caméra

### La base et la mobilité

Le châssis comporte deux chenilles et deux roues (figure 3.4). La combinaison des roues et des chenilles permet d'avoir une bonne mobilité. En



effet, les chenilles sont très utiles pour avoir une traction et une mobilité dans les terrains accidentés. Par contre, elles ne permettent pas au robot de tourner sur lui-même. Aussi des roues latérales ont été rajoutées sur l'axe central, ce qui facilite la rotation du robot. Cette combinaison garde aussi une forme proche du cylindre, ce qui s'harmonise bien avec le reste du robot. Cela contribue aussi à avoir une zone de contact vaste et augmente la stabilité.

Le châssis est composé de deux sous-parties identiques (figure 3.5), les batteries étant placées entre deux. Chenilles et roues sont propulsées par deux moteurs électriques à courant continu. Un réducteur à trois étages est placé dans le châssis. Le premier étage est fait avec une vis sans fin miniature à double filets, ce qui permet une grande réduction dans un espace très confiné.

La figure 3.6 montre un plan éclaté de la base. Sous *A* et *E*, il y a les chenilles, les roues et la courroie de transmission. Les deux triangles sont des renforts à la structure plastique pour reprendre les forces de tension des chenilles. Sous *B* et *D* la structure plastique, les moteurs et leur réducteur sont représentés. À côté des moteurs, il y a les circuits électroniques des 4 capteurs infra-rouges. Malgré sa hauteur, le robot dispose d'un centre de masse relativement bas, en raison de la mise en place des batteries entre les chenilles (*C*).

#### **La tourelle rotative et la pince rigide**

La tourelle rotative et la pince rigide (figure 3.7) sont les sous-parties les plus compliquées d'un point de vue mécanique, car elles donnent au robot ses capacités coopératives d'auto-assemblage.

La tourelle rotative se compose des éléments suivants :

- un capteur de traction entre la base et la tourelle
- un moteur et la mécanique assurant la rotation de la tourelle par rapport à la base
- deux moteurs contrôlant les deux degrés de liberté de la pince rigide
- un circuit vertical servant de support aux moteurs et d'un accéléromètre

Le capteur de traction a été ajouté durant le projet suite aux résultats des simulations des collègues informaticiens responsables de contrôle. En effet, pour simplifier la coordination des robots une fois assemblés, un capteur de force de traction donne directement la direction voulue par le groupe assemblé. Le capteur se compose de quatre plaques en acier ressort (figure 3.8 F) sur lesquels quatre jauges de contrainte sont collées. Les

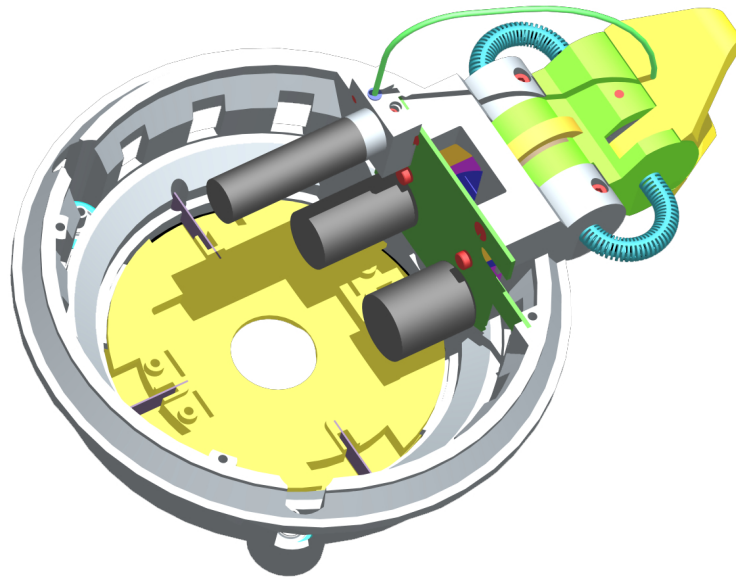


Figure 3.7 La tourelle rotative et la pince rigide

plaques sont placées verticalement et sont fixées sur la partie centrale fixe au châssis. Une couronne dentée, sur laquelle la tourelle s'engrène, coulisse par rapport à cette partie centrale. Les forces verticales sont reprises directement par ces deux pièces. Par contre, les forces horizontales sont reprises en X et Y par les plaques métalliques et mesurées par les jauges. Celles-ci sont placées en opposé, pour ne mesurer que les contraintes en X ou Y et annulent la mesure du couple.

Dans la figure 3.8 sous E, la coque externe du robot est représentée. Sur celle-ci vient se fixer toute la partie supérieure du robot. Des galets assurent le guidage de la tourelle rotative. Sous D, deux moteurs sont représentés ; l'un entraîne la rotation de la tourelle à travers une vis sans fin et un pignon s'engrenant sur la couronne dentée, l'autre actionne le levage de la pince rigide à travers une double réduction à vis sans fin et roues. Ce système fournit suffisamment de couple pour soulever un autre s-bot.

Dans la partie C de la figure 3.8, on peut voir les détails des pièces composant la pince rigide. Celle-ci est très complexe, car elle comporte un capteur de couple sur la rotation ainsi qu'une barrière optique entre la mâchoire supérieure et inférieure. Un ressort de torsion et un différentiel assurent aussi l'ouverture symétrique de celle-ci. Le capteur de couple est fabriqué à l'aide d'une barrière optique analogique et de deux filtres

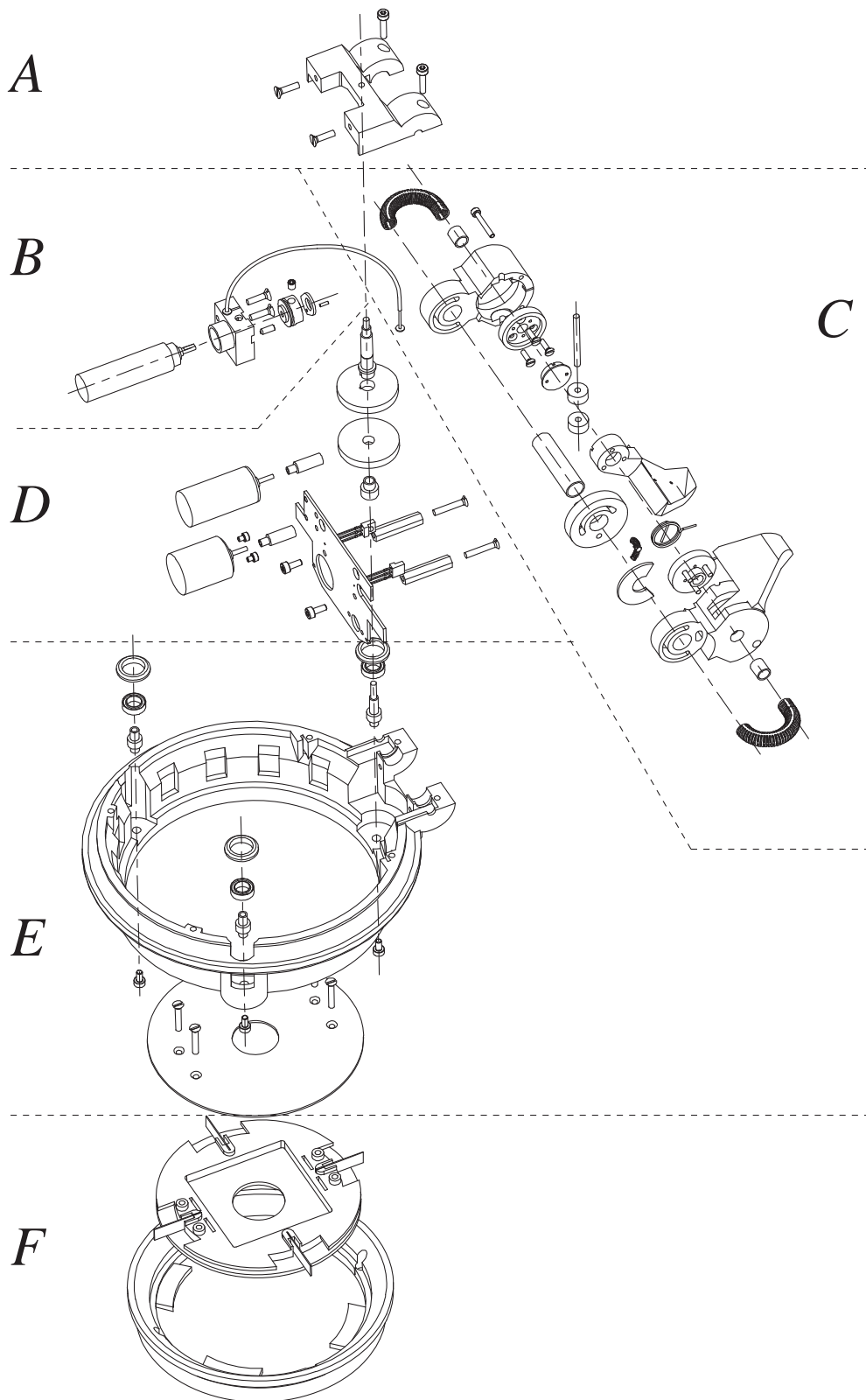
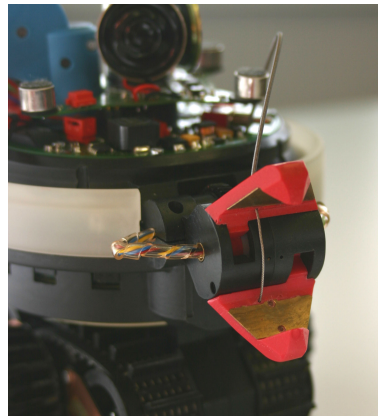


Figure 3.8 Plan éclaté de la tourelle rotative et de la pince rigide



**Figure 3.9** La pince rigide et son système de fermeture par câble

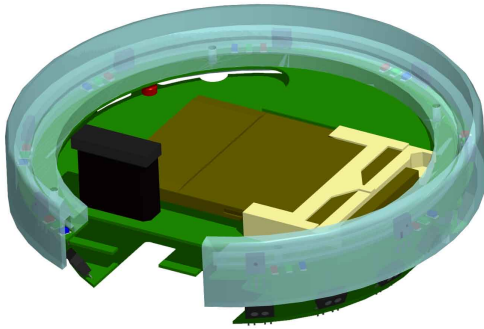
polarisants. Suivant le couple effectif sur la pince, les filtres deviennent plus ou moins opaques.

La fermeture de la pince est assurée par un entraînement par câble (figure 3.9). Un moteur enroule un câble (figure 3.8 B), qui à travers une gaine comme un frein de vélo, ressert les deux mâchoires de la pince avec une grande force.

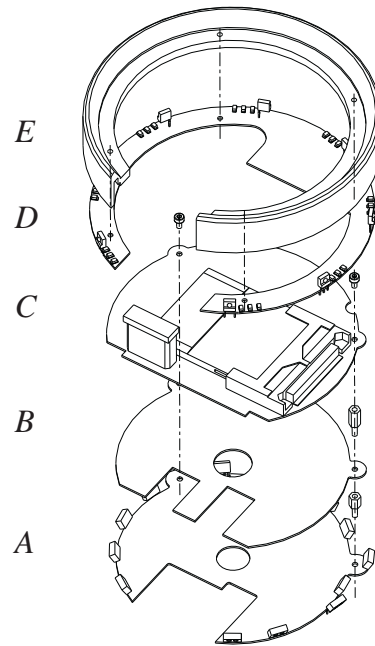
### **Le coeur de l'électronique**

Le coeur de l'électronique est placé dans la tourelle rotative (figure 3.10). Il s'agit d'une superposition de circuits imprimés assurant les différentes fonctions du s-bot décrites plus amplement dans la section 3.1.2.

La figure 3.11 représente les différentes pièces. Sous A, le circuit des capteurs de proximité est présenté. Ceux-ci mesurent les distances jusqu'aux objets à travers des ouvertures dans la pièce principale de la tourelle. Le circuit représenté sous B comporte les microcontrôleurs qui commandent les moteurs de la pince rigide et ses capteurs. Sous C, c'est le circuit mère principal du robot. Il contient le processeur PXA255 tournant sous Linux à 400MHz. Sous D, c'est le circuit de l'anneau lumineux comportant 8X3 LED RGB. Ce circuit comporte aussi 8 capteurs de lumière visible. Sous E, la pièce semi-transparente sert de diffuseur pour l'anneau d'attache lumineux. Grâce à sa forme inversée par rapport à la pince, cette pièce permet au s-bot de s'y accrocher. Mécaniquement parlant, les circuits sont fixés par 2 tourelles d'entretoises M2. Les circuits électroniques sont reliés par des connecteurs "print-to-print" de différentes hauteurs. Le placement des composants électroniques de ces circuits fut assez laborieux. En effet, la distance verticale entre les circuits et les composants des autres circuits



**Figure 3.10** Le coeur de l'électronique dans la tourelle



**Figure 3.11** Plan éclaté des circuits électroniques

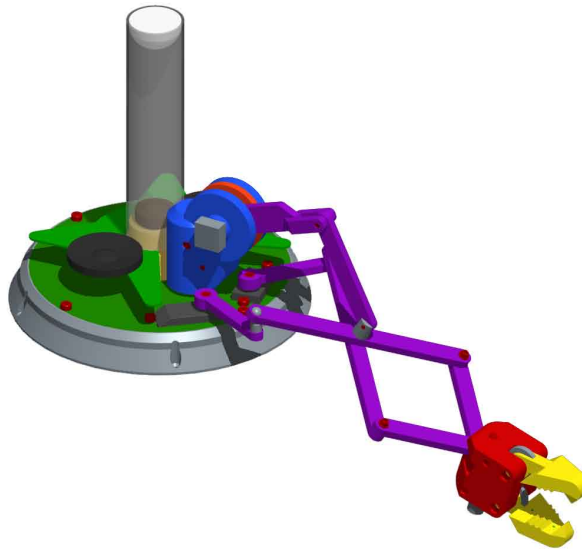
est par endroit très faible. Plusieurs itérations de placement de composants ont été nécessaires avant de pouvoir effectivement entrer les circuits dans le s-bot. Le circuit principal avec le Linux comporte aussi deux connecteurs compact flash, dont l'un deux pour une carte Wifi. Celle-ci rentre tout juste dans le corps du robot.

### La pince mobile et la caméra

La partie supérieure du s-bot (figure 3.12) se compose d'un couvercle électronique, de la pince mobile, de la caméra omnidirectionnelle et du système sonore. Le circuit électronique (figure 3.13 A) gère essentiellement la pince mobile, le moteur de la pince rigide fermant les mâchoires et la caméra. Sur ce circuit se trouve aussi les connecteurs permettant une connexion au bus I<sup>2</sup>C, au PXA255 ou à recharger le s-bot. Il sert aussi de support aux servomoteurs de la pince mobile.

La caméra est constituée d'un chip VGA, d'un support, d'une lentille et d'un miroir sphérique placé au sommet d'un tube en plexiglas (figure 3.13 E).

La pince mobile est constituée d'une structure parallèle. Le mouvement horizontal est assuré par deux servomoteurs actionnant un bras avec une structure en croisillon (figure 3.13 C). Cette structure permet ainsi un



**Figure 3.12** La pince mobile et la caméra

déplacement latéral, mais surtout une extension du bras. Sous *B*, on peut voir tout le mécanisme de levage du bras. Il se compose d'un moteur placé verticalement servant en même temps d'axe grâce à un roulement. Ce moteur actionne le troisième bras de la structure parallèle à travers une réduction à trois étages ; une vis sans fin et un engrenage droit.

Le circuit en étoile représenté dans la figure 3.13 sous *D*, supporte la partie son du s-bot. Elle est composée de deux hauts-parleurs et de quatre microphones placés à chaque pointe du circuit étoile.

### 3.1.2 L'électronique

L'électronique du robot s-bot est très complète et intégrée. Celle-ci gère au total 7 moteurs, 2 servomoteurs et 45 capteurs. Elle embarque aussi 13 microprocesseurs PIC16, un processeur XScale PXA255 tournant sous Linux. Le tout est réparti sur 14 circuits imprimés représentés dans la figure 3.14. Dans cette section, nous allons décrire l'électronique du robot niveau par niveau.

#### L'alimentation électrique

Au bas de la figure 3.14, on peut voir les batteries. Il s'agit de deux batteries Li-Ion montées en série, avec les caractéristiques suivantes :

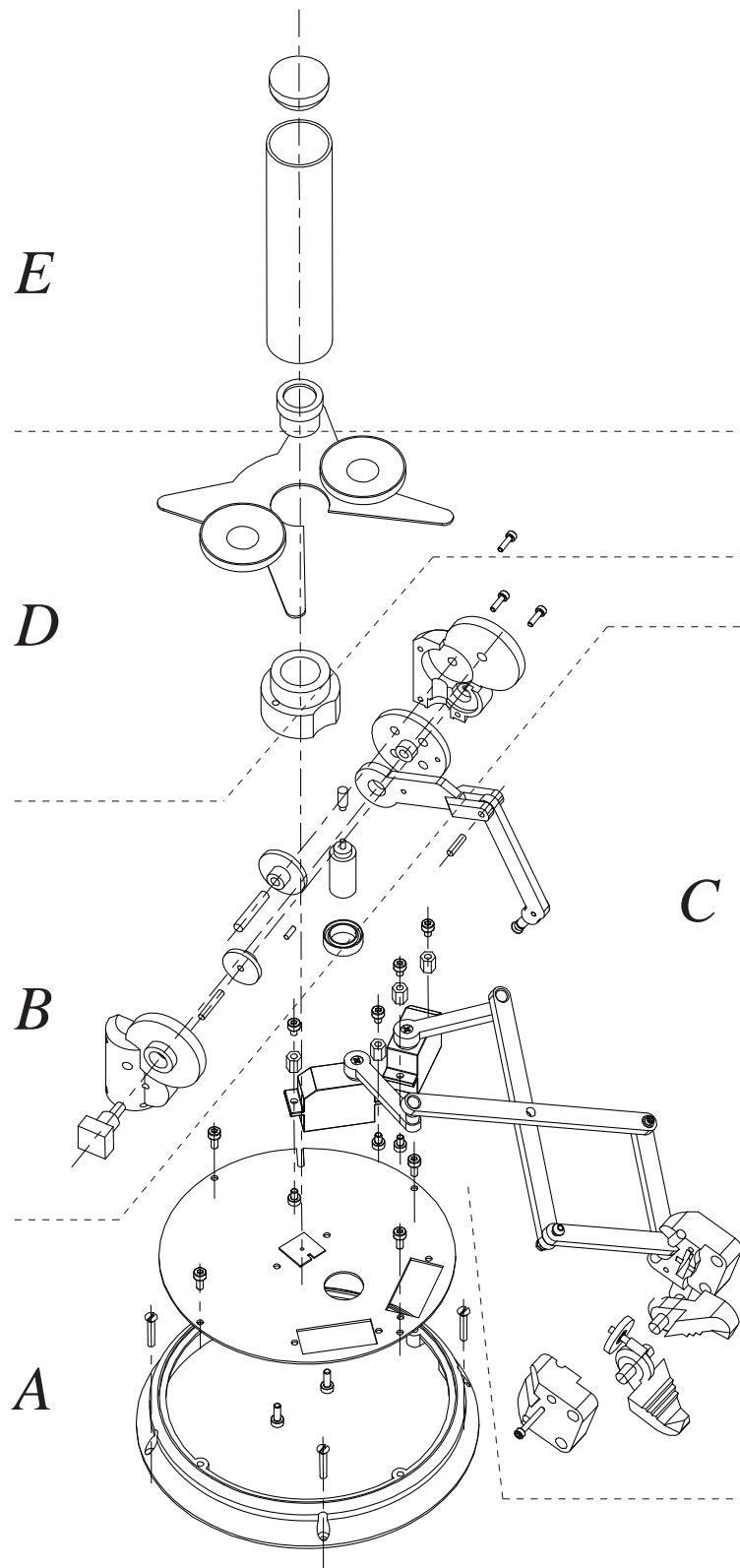
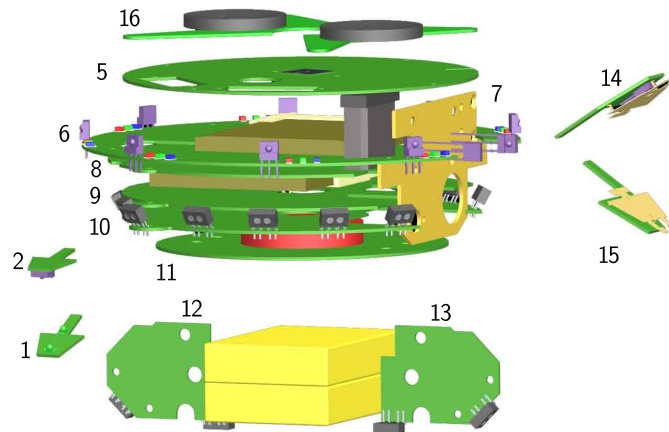
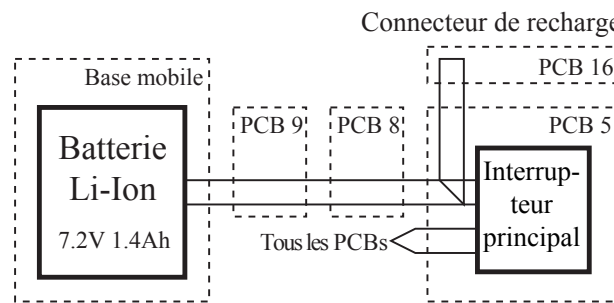


Figure 3.13 Plan éclaté la pince mobile et la caméra



**Figure 3.14** Les 14 circuits imprimés du s-bot représentés selon leur position dans le robot.



**Figure 3.15** Schéma de l'alimentation.

Type	No Référence	Capacité	Autonomie	T. Charge
2×Li-Ion	HLP103448	1400mAh 7.2V 10Wh	1-2h	2h

La tension de la batterie est répartie à travers le robot. Sur chaque sous-partie, un convertisseur DC/DC assure la mise à tension de 5V pour les microcontrôleurs et la plupart des composants électroniques, ce qui permet un meilleur rendement et moins de bruit. Les moteurs sont, quant à eux, directement alimentés à la tension des batteries.

La figure 3.15 représente la distribution de la tension batterie. Un câble, venant directement de la batterie, se connecte sur le PCB 9, puis à travers le PCB 8 arrive au PCB 5, où se trouve l'interrupteur principal du robot. L'alimentation redescend ainsi dans le robot pour alimenter tous les cir-



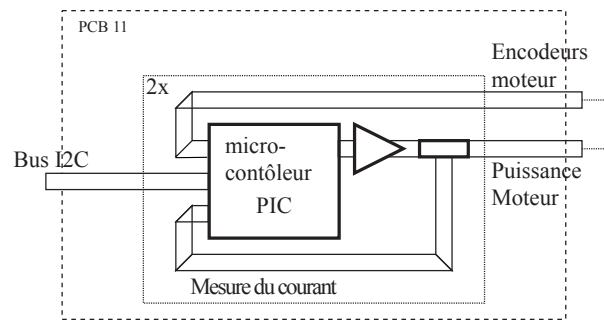


Figure 3.16 Schéma électronique des moteurs des chenilles

cuits. Un connecteur de recharge se trouve au sommet du robot sur le PCB 16.

### Les modules moteurs<sup>2</sup>

Chaque moteur dans le s-bot est contrôlé par son propre microcontrôleur. Sur chaque moteur, une mesure en courant a été implémentée. Tous les moteurs ont un encodeur hormis le moteur de levage de la pince mobile. Celui-ci a un potentiomètre alors que le moteur des mâchoires de la pince flexible n'est contrôlé qu'en couple. L'électronique est ainsi quasiment identique pour tous les moteurs. Quant au code, il est identique à l'exception de quelques paramètres liés aux valeurs des PID et les limites.

La figure 3.16 représente le schéma d'un des modules moteurs pilotant les chenilles. Le moteur est alimenté à travers un pont H à la tension de la batterie. Grâce à l'encodeur, le microcontrôleur mesure la position et la vitesse du moteur. Un amplificateur mesure le courant du moteur grâce à une résistance placée en série. Ainsi, avec ces trois mesures, on peut réguler le moteur, selon le couple, la vitesse ou la position. Il s'agit d'un régulateur PID, avec ses paramètres stockés dans la mémoire EEPROM du microcontrôleur. En régulation de position ou de vitesse, une limite de courant peut être rajoutée. De plus, la vitesse maximum peut être aussi limitée en asservissement de position. Ces deux modules moteurs sont placés sur le PCB11 juste au-dessus des moteurs.

Sur le PCB9 se trouvent deux modules moteurs (figure 3.17). Le premier commande la rotation de la tourelle. La particularité de ce module est qu'il peut être alimenté même après la coupure de l'interrupteur princi-

2. Les modules moteurs ont été développés à l'ASL par Michel Lauria et Ralph Piguet. Le code microcontrôleur a été amélioré par Daniel Baer lors de son diplôme "Actuateurs du robot mobile miniature s-bot" <http://lis/education/studentsProjects/abstracts/8.pdf>.

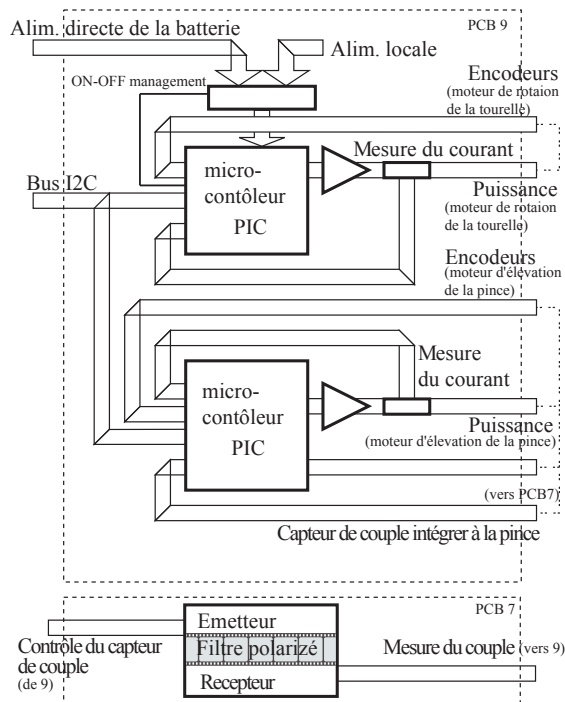


Figure 3.17 Schéma électronique des moteurs

pal. La tourelle n'a pas de butée mécanique, mais ne peut pas tourner indéfiniment, car les câbles entre le châssis et la tourelle se déconnecteraient. Ainsi, à chaque extinction du robot, le microcontrôleur mémorise la position absolue de la tourelle dans son EEPROM. Tout de suite après, le microcontrôleur coupe sa propre alimentation de back-up et s'éteint comme le reste du robot. La position de la tourelle est ainsi mémorisée et évite une mise à zéro manuelle à chaque démarrage du robot.

Le deuxième module moteur commande le levage de la pince rigide. Sa particularité est de commander aussi un capteur de couple sur cet axe vertical. En effet, le réducteur est irréversible. Ainsi, le courant injecté dans le moteur n'est pas proportionnel au couple réel appliqué à la pince. C'est pourquoi un capteur additionnel a été ajouté. Sur le PCB7, une barrière optique mesure l'intensité du couple à travers deux filtres polarisants reliés par un ressort. Par conséquent, suivant le couple, les filtres laissent passer une certaine quantité de lumière et ceci proportionnellement au couple.

La figure 3.18 représente les modules moteurs placés sur le PCB5 au sommet du robot. Il s'agit de modules moteurs conventionnels, sauf pour le moteur du bras vertical dont la position est mesurée à l'aide d'un

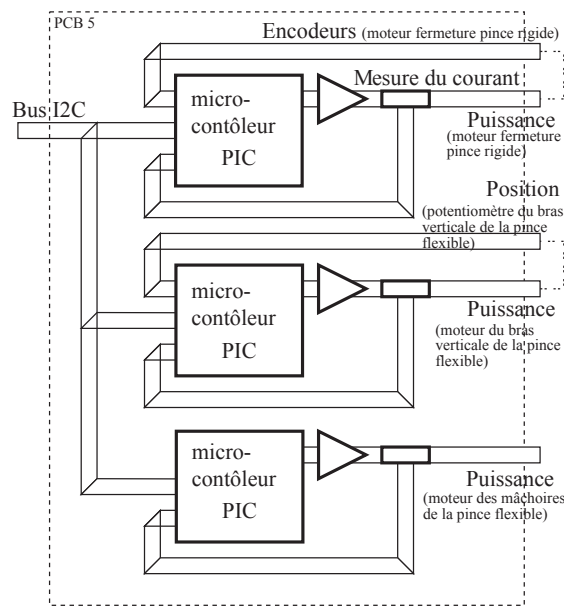


Figure 3.18 Schéma électronique des moteurs

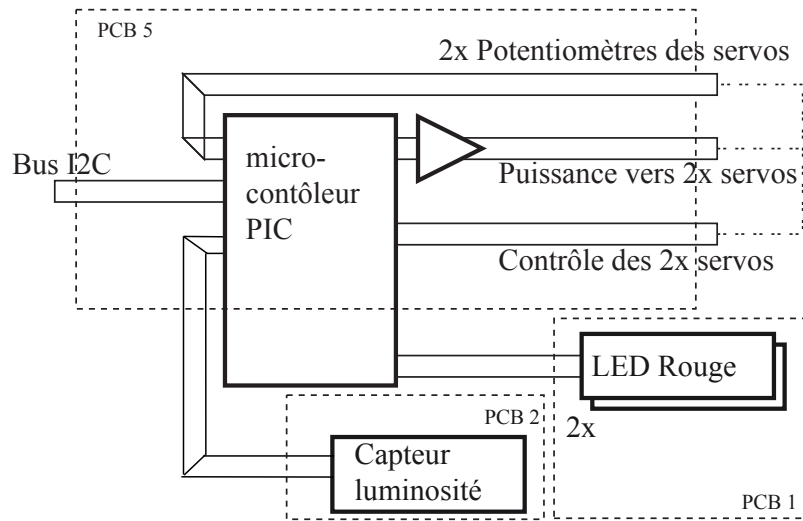
potentiomètre placé sur l'axe du bras. En ce qui concerne la fermeture de la pince flexible, seule la mesure du courant est utilisée.

### Les servomoteurs et les barrières optiques de la pince flexible

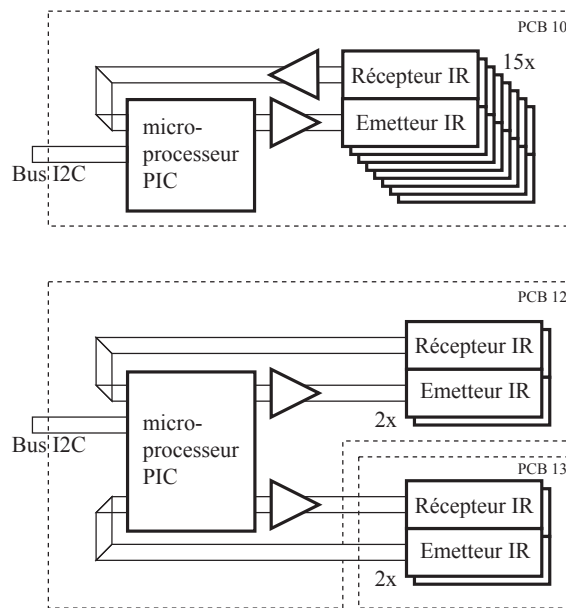
La figure 3.19 représente l'électronique de contrôle des servomoteurs et des barrières optiques dans la pince flexible. Depuis le microcontrôleur, la commande est envoyée pour chaque servomoteur. La puissance de ceux-ci peut aussi être coupée. Les servos ont aussi été démontés et modifiés, afin de sortir la valeur de leur potentiomètre interne pour avoir un feedback au sein du microcontrôleur. Celui-ci commande également la barrière optique incluse dans la mâchoire de la pince flexible. Cette barrière est composée de deux LED de couleur rouge et d'un capteur d'intensité lumineuse muni d'un filtre rouge.

### Les capteurs de proximité

Les capteurs de proximité sont essentiels à la navigation du s-bot. Il y en a 19 dans le robot (figure 3.20). Quatre sont situés sous le robot, afin de détecter d'éventuels trous. Les 15 autres sont disposés tout autour de la tourelle sur le PCB10. Ceux-ci ont la particularité d'avoir deux étages d'amplification avec un système de "sample-hold". En effet, on peut garder temporairement dans des capacités une image de la tension du



**Figure 3.19** Schéma électronique servomoteurs et barrière optique de la pince flexible



**Figure 3.20** Schéma électronique des capteurs de proximité

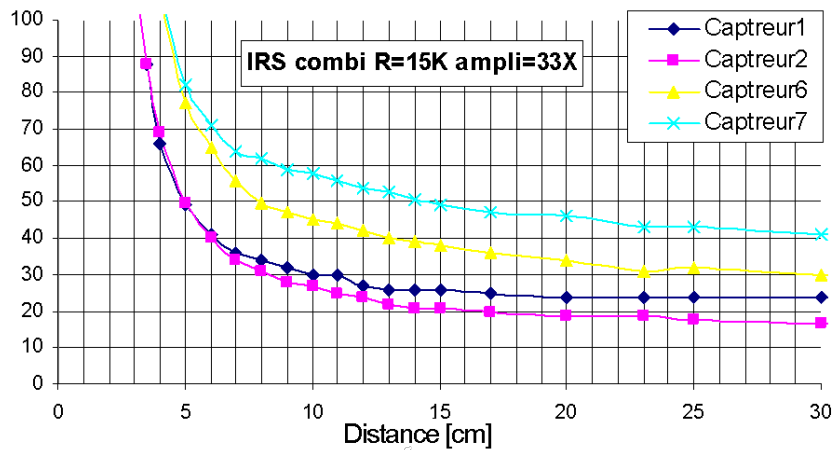


Figure 3.21 Réponse des capteurs de proximité

récepteur quand l'émetteur est éteint afin d'amplifier la différence quand l'émetteur est éteint. On amplifie ainsi électroniquement les deltas, qui sont des valeurs normalement déduites en logiciel. Ces capteurs peuvent ainsi mesurer jusqu'à 15cm des objets qui réfléchissent l'infrarouge (figure 3.21).

### Le capteur de force

Sur le PCB11 se trouve l'électronique du capteur de force (figure 3.22). Quatre jauges sont placées deux par deux dans des demi-ponts de Wheatstone. Ceux-ci sont complétés par des résistances de précision sur le circuit. Le pont peut être équilibré grâce à un potentiomètre. Le gain de l'amplificateur est aussi réglable et ainsi la sensibilité du capteur peut-être ajustée.

### Les accéléromètres et les capteurs d'humidité

La figure 3.23 représente les capteurs additionnels du s-bot. Deux accéléromètres identiques mesurent les accélérations en X et Y. L'un d'eux est placé sur le circuit vertical (PCB7). Ainsi, le s-bot peut mesurer l'accélération en 3 dimensions. Deux capteurs d'humidité et de température sont placés sur le PCB9. Le microcontrôleur gère également une barrière optique lumineuse identique à celle dans la mâchoire de la pince flexible.

### L'anneau d'attache lumineux

L'anneau lumineux est un actionneur essentiel au s-bot. En effet, grâce à celui-ci, il peut communiquer aux autres robots différents messages

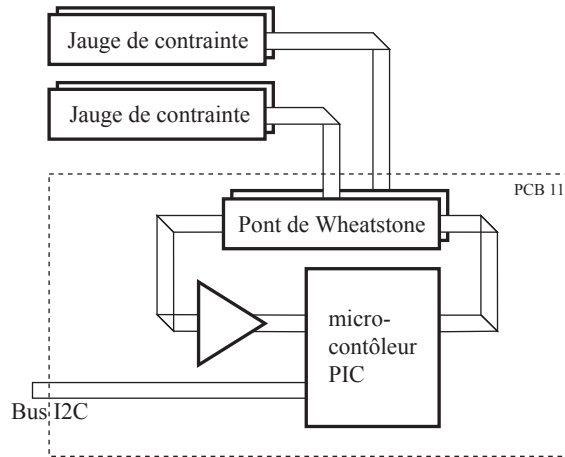


Figure 3.22 Schéma électronique des capteurs de forces

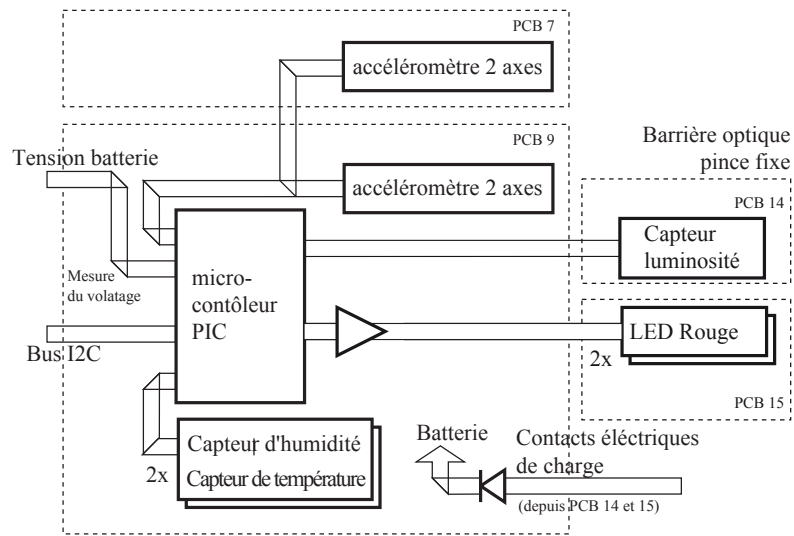


Figure 3.23 Schéma électronique des capteurs sur le PCB9

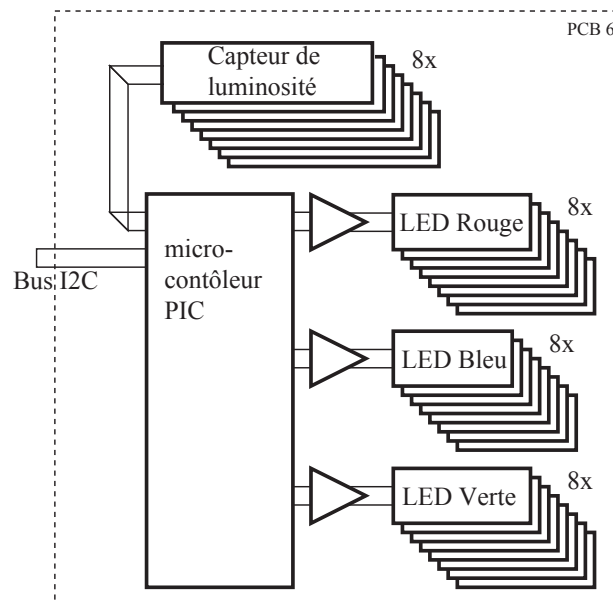


Figure 3.24 Schéma électronique de l'anneau d'attache lumineux

lumineux. Il est aussi utilisé pour détecter ses coéquipiers grâce à la caméra. L'anneau est composé de 24 LED ; 8 rouges, 8 vertes et 8 bleues, placées trois par trois pour former 8 pixels RGB (figure 3.24). Le circuit comporte aussi 3 convertisseurs DC/DC, l'un à 5V pour le microcontrôleur, un autre à 3.6V pour les LED de couleurs bleues et le dernier à 1.9V pour les LED vertes et rouges. On a ainsi un convertisseur optimisé pour chaque chute de tension des LED. Le circuit contient également 8 capteurs d'intensité lumineuse. Le s-bot peut ainsi détecter une source lumineuse.

### La caméra et le système sonore

Le système caméra du s-bot se compose d'un capteur image CMOS VGA de  $640 \times 480$  pixels. Il est interfacé avec une puce d'acquisition qui transforme les trames en image jpeg et les transmet au processeur principal par USB. Ces deux composants sont placés sur le PCB5. Ainsi, seules deux lignes pour l'USB sont nécessaires pour transmettre le flux vidéo au circuit principal du s-bot, le PCB8 (figure 3.25). Le système audio, composé de quatre microphones électrets et de deux haut-parleurs, est placé sur le PCB16 au sommet du s-bot. Les micros sont amplifiés localement et le signal est transmis directement au PCB8 où un CODEC numérise les signaux. Le son en sortie est aussi amplifié sur le PCB proche de la sortie. Le choix de mettre le CODEC et la partie analogique loin des microphones est discutable, mais le contraire posait des problèmes avec le bus I2S.

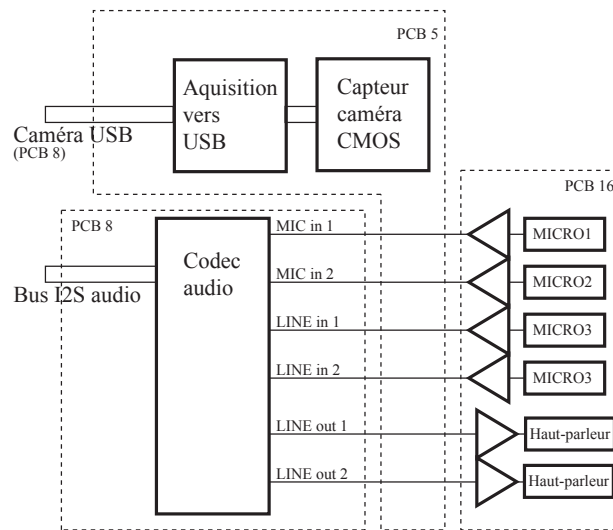


Figure 3.25 Schéma électronique de la caméra et du système sonore

### La carte principale de contrôle<sup>3</sup>

La figure 3.26 représente la carte principale de contrôle du s-bot. Celle-ci est principalement composée d'un processeur XScale PXA255 cadencé à 400Mhz, de 64Mo de mémoire RAM, 32Mo de mémoire FLASH, un contrôleur USB et de deux connecteurs CompactFlash. Le PXA255 gère un bus I<sup>2</sup>C commun à tout le robot, un codec audio, un USB slave, un USB master et une connections WIFI. Tous les microcontrôleurs PIC du s-bot sont connectés par le bus I<sup>2</sup>C. La connections RS232 nous permet d'avoir une console Linux. Le système audio est branché à travers un CODEC et un bus I2S standard. La caméra omnidirectionnelle est interfacée au processeur par USB. Un des ports CompactFlash est utilisé par une carte Wifi. L'antenne, ne fonctionnant par sur cette carte à cause des plans de masse du PCB, a été déportée sur le PCB5 par un câble coaxial. Ceci permet aussi une diminution du bruit électronique dû aux communications WiFi.

#### 3.1.3 Les logiciels

Les logiciels dans le s-bot sont dispersés dans le processeur principal et les 13 microcontrôleurs PIC répartis dans le robot comme décrit dans le sous-chapitre précédent. La figure 3.27 représente la structure globale logicielle du robot.

3. Développée avec Pierre Bureau de K-Team SA sur la base du travail de diplôme de Stéphane Magnenat et Julien Pilet <http://stephane.magnenat.net/data/rapport-diplome.pdf>.



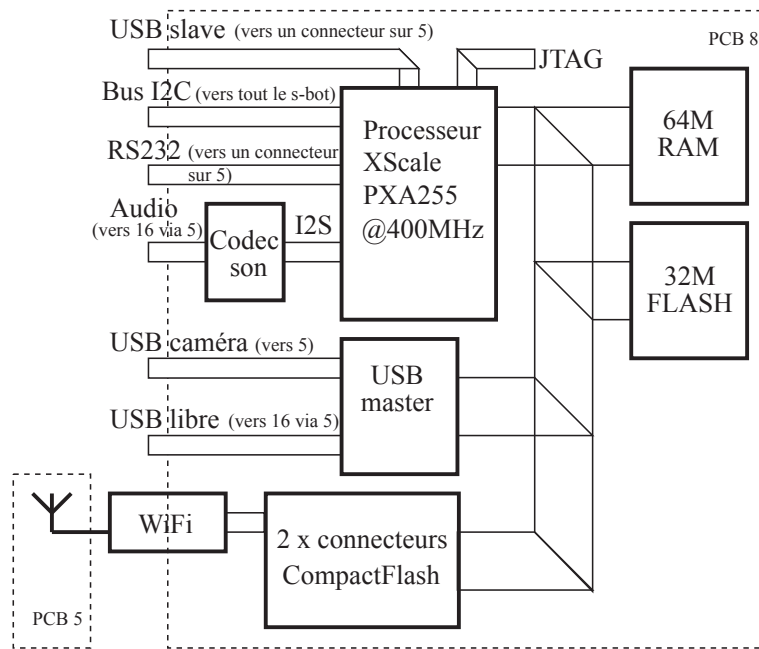


Figure 3.26 Schéma électronique de la carte principale de contrôle

#### Le logiciel utilisateur de haut niveau<sup>4</sup>

L'utilisateur final du s-bot va essentiellement programmer le robot sous le système exploitation Linux. Ce système tourne sur le processeur central PXA255 et interagit avec le reste des sous-parties du robot par différents bus de communication (USB, I<sup>2</sup>C, I2S). L'utilisateur ne peut pas modifier le code dans les microcontrôleurs PIC et les firmwares des différents sous-systèmes comme la caméra. Le PXA accède en tant que maître du bus I<sup>2</sup>C aux registres implémentés dans les microcontrôleurs PIC. Ces registres donnent accès au bas niveau du robot. Une interface de programmation d'application (API) enveloppe les accès I<sup>2</sup>C bas niveau aux différents registres pour donner aux utilisateurs des accès aux fonctionnalités par des appels en langage C.

Un driver Linux donne accès aux images de la caméra. Celles-ci sont au format jpeg et il faut les décompresser. Pour augmenter la fréquence de traitement, seulement la sous-partie de l'image correspondante au miroir est décompressée et traitée.

Un driver a été développé pour le Codec audio spécifique à la carte principale du s-bot ainsi que pour accéder aux fonctionnalités compact-flash.

4. Linux a été porté par Stéphane Magnenat.

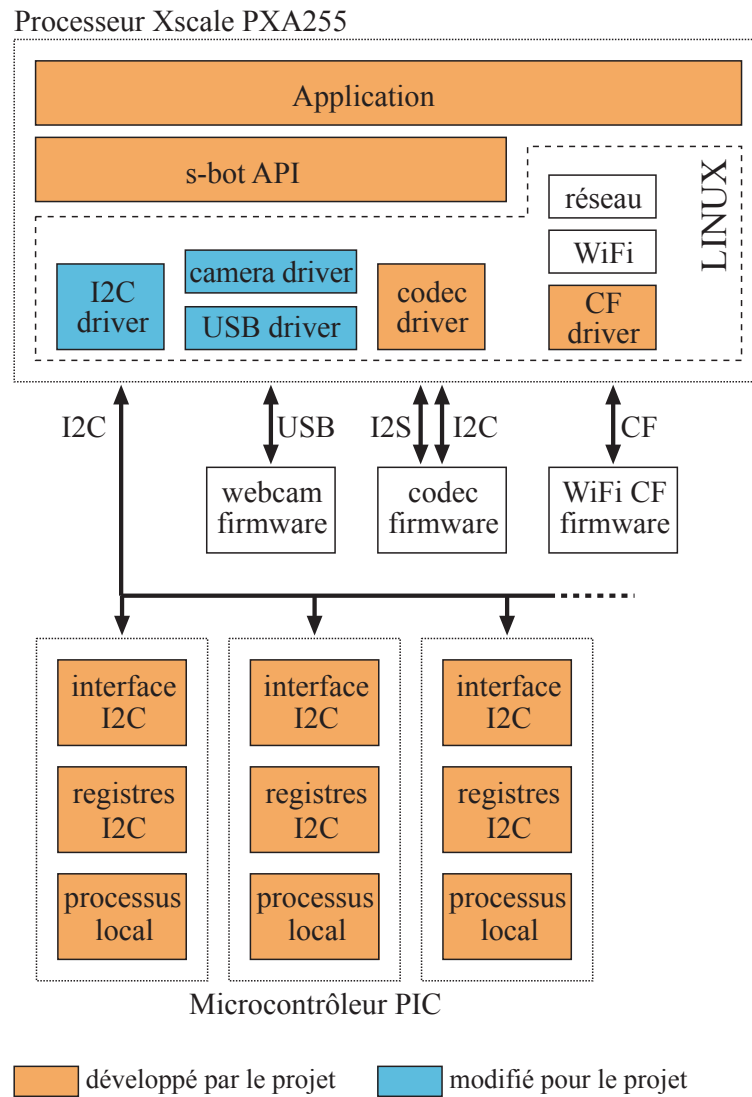


Figure 3.27 La structure logicielle

L'utilisateur final peut également utiliser les fonctionnalités standard au système d'exploitation Linux comme l'accès au réseau, le multitâche et la gestion de fichier.

### Les logiciels de bas niveau

Les détails de tous les logiciels embarqués dans les 13 microcontrôleurs ne vont pas être décrits ici. Il s'agit juste de donner une vue d'ensemble.

Il y a deux groupes de microcontrôleurs :

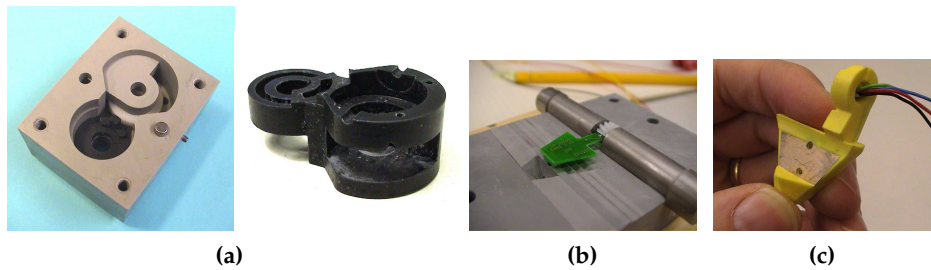
- Les 7 modules moteurs. Tous les microcontrôleurs ont le même code et seule la configuration de ceux-ci change par rapport aux moteurs et à leur fonction. Le code est écrit en assembleur et optimisé pour un contrôle de moteur DC.
- Les 6 autres microcontrôleurs gèrent le reste des capteurs, LED et servomoteurs. Le code est écrit en C, mais gère en temps réel les fonctionnalités spécifiques à chaque sous-système décrit dans la section 3.1.2.

Par le bus I<sup>2</sup>C, les différents registres des modules peuvent être lus ou écrits par un maître. Le microcontrôleur va réagir au changement de ces registres et va le plus souvent mettre à disposition les résultats dans d'autres registres. Si certains registres sont communs à tous les PIC, la majorité ont des significations différentes dépendantes des sous-systèmes.

#### 3.1.4 Les techniques de production

Sachant dès le départ que plusieurs s-bot seraient nécessaires (35 pièces), le design et les techniques de fabrication ont été adaptés.

Ainsi, pour la plupart des pièces non métalliques, la technique de moulage a été utilisée. Il s'agit de fabriquer un moule, le plus souvent en POM ou en PVC, qui est le négatif de la pièce (figure 3.28a). Ensuite une résine polyuréthane (Neukadur MultiCast 2) est coulée dans le moule. La résine est un bi-composant qui reste liquide pendant 2-3 minutes après mélange et durcit en environ 1 heure. Avec ce type de moule, environ 50 à 100 pièces peuvent être fabriquées. Les moules de fabrication artisanale, peuvent aussi comporter divers inserts comme une vis, qui après un démoulage manuel approprié, laissera directement un filetage dans la pièce. Avec un moule adapté, on peut aussi surmouler différentes pièces : des boulons, des engrenages et même des circuits imprimés (figure 3.28b). Par exemple, les deux coques de la base ont des boulons noyés dans le plastique qui servent de filetage pour les vis de montage. Les mâchoires des pinces ont, quant à elles, des circuits imprimés et des fils électriques à l'intérieur de celles-ci (figure 3.28c). Ces circuits servent même de renfort



**Figure 3.28** (a) Moule et pièce correspondante (b) Moule de la mâchoire de la pince mobile incluant un PCB et un engrenage (c) Mâchoire de la pince fixe avec les LED incluses

mécanique aux mâchoires. Les composants électroniques simples, comme les LED et photo-transistors, supportent très bien ce genre de surmoulage car les températures restent faibles, par rapport à de l'injection plastique conventionnelle.

Les moteurs-réducteurs conventionnels du marché sont souvent onéreux et allongés. Pour intégrer tous les mécanismes du s-bot, les étages de réductions ont été conçus spécifiquement et déportés afin de s'adapter à la morphologie du robot. De plus, des systèmes de réduction à vis sans fin ont été utilisés. Ces systèmes, malgré un rendement plus faible, permettent une grande réduction pour un espace confiné. Cependant, sur le marché, on ne trouve pas ce type d'engrenage d'assez petite taille. Grâce à l'expérience horlogère d'André Guignard, des vis sans fin de petit diamètre ont été fabriquées à l'EPFL.

Les circuits électroniques du s-bot sont tous des doubles couches, à l'exception de celui du processeur central qui lui en comporte 8. La plupart des composants des circuits sont des SMD. Ainsi, les circuits sont adaptés pour un montage industriel. Leur particularité se trouve plutôt dans leur forme non conventionnelle, mais cela ne pose que peu de problèmes pour les techniques de fabrication actuelles.

Le s-bot ne comporte pour finir que peu de pièces normalisées à part les vis et certains engrenages. Pour fabriquer un s-bot, il faut environ 1 mois, même en optimisant des opérations. De plus, les pièces moulées ont des tolérances très larges, leur variabilité oblige de nombreux ajustements. Il faut presque une semaine de montage après la fabrication des pièces. A cause de la forte intégration du robot, il faut par exemple deux heures de démontage et montage pour changer les batteries se trouvant au centre de la base.

### 3.2 Caractéristiques du s-bot

#### Les degrés de liberté (ddl)

Le s-bot comporte 9 degrés de liberté : deux chenilles, la tourelle rotative, 2 pour la pince rigide et 4 pour la pince flexible. Hormis les chenilles, les degrés de liberté sont pour les fonctions collectives d'auto-assemblage du robot. La pince rigide et la tourelle sont d'ailleurs les plus utilisées du robot. La pince flexible manque de fiabilité et n'a pour finir pas été montée sur tous les s-bots. Elle n'apporte d'ailleurs pas d'avantage pour former des structures et n'est pas assez précise pour attraper des objets de façon fiable. C'est ainsi que dans les prochaines expériences décrites dans ce chapitre et le chapitre 6, les s-bots ne sont pas munis de leur pince flexible. Par souci de comparaison avec les autres robots, le s-bot sera considéré avec 5 degrés de liberté.

#### Les informations sensorielles extrinsèques

Voici un tableau résumant les débits de données de chaque type de capteur :

	Types	Données brutes	Signal utile	Fréquence	Data/s
microcontrôleurs	4×IR au sol	4×8bits	7bits	50Hz	1.4kbit/s
	16×IR proximité	16×12bits	11bits	50Hz	8.8kbit/s
	2×Forces	2×12bits	9bits	50Hz	0.9kbit/s
	3D accéléromètre	3×8bits	8 bits	50Hz	1.2kbit/s
	2×barrière	2×10bits	9bits	50Hz	0.9kbit/s
	optique				
	couple pince rigide	8bits	7bits	50Hz	0.35kbit/s
	2×humidité	2×12bits	12bits	1Hz	24bit/s
	2×température	2×14bits	14bits	1Hz	28bit/s
	8×lumière	8×10bits	9bits	50Hz	3.6kbit/s
	Total				20.8kbit/s
PXA255	4×micros	4×16bits	13bits	44.1kHz	2.3Mbit/s
	caméra couleur	640×480p 8bits	8bit	8Hz	19.7Mbit/s
	Total				22Mbit/s

	Coûts en Frs
Mécanique	124.-
Moteurs	647.-
Électronique	2184.-
<b>Total</b>	<b>2955.-Frs</b>

**Tableau 3.1** Coût du sbot

Les 12 microcontrôleurs PIC16F tournent à 5 MIPS et sont des processeurs 8bits. Cela fait un total de 60 MIPS : Le processeur 32 bits Xscale est cadencé à 400Mhz, et 371-533 MIPS @400 MHz<sup>5</sup>

### La communication

Le s-bot a trois canaux de communication. Premièrement le Wifi, communication radio haut débit, est utilisé comme interface utilisateur. Cette communication peut aussi être utilisée entre les robots, mais n'a été que très peu employée par souci idéologique propre au projet. Deuxièmement le système sonore est utilisé pour des communications simples et globales. Le principe est d'émettre des sons sinusoïdaux sur différentes fréquences et de détecter par une transformée de Fourier. La direction de l'émetteur peut aussi être déterminée, cependant, le bruit électronique ne permettait pas de le faire d'une façon fiable. Le troisième moyen est la caméra et l'anneau lumineux. Grâce à la caméra omnidirectionnelle, les robots peuvent se détecter et transmettre leur état ou besoin grâce à leur cercle lumineux. En allumant les LED de différentes couleurs, une direction peut même être indiquée [16].

### Le prix

Le budget matériel lors du projet pour les s-bot était de 3000 frs. Ce montant n'était pas suffisant pour sous-traiter toutes les pièces mécaniques. Ainsi beaucoup de pièces ont été fabriquées par nos propres moyens. Il faut environ 1 mois-homme pour fabriquer un s-bot, dont presque une semaine de montage et d'ajustage. Les grands PCB ont été montés par des monteurs industriels, sauf pour le PCB5, qui faute de budget et temps à été monté à la main. Tous les petits PCB ont été montés aussi par nous-même. Le tableau 3.1 résume la répartition des coûts matériels.

L'électronique représente la plus grande partie du prix du robot (73%), en tout cas d'un point de vue matériel. Une grande partie du prix est

---

5. Source : [http://en.wikipedia.org/wiki/ARM\\_architecture](http://en.wikipedia.org/wiki/ARM_architecture)



(a) Le terrain accidenté (b) Commencement et résultat d'une expérience avec 6 s-bots

**Figure 3.29** Illustrations des expériences d'auto-assemblage"

ici omis par le fait que beaucoup de travail a dû être fourni par les collaborateurs du projet. La part mécanique se rapprocherait plus des 1000.-Frs si la fabrication des pièces avait été sous-traitée. Ainsi la mécanique représenterait 25% du prix de revient du robot qui au total coûterait prêt de 4000.-Frs.

Le prix de la mécanique est élevé car la miniaturisation et l'intégration ont été très poussées. Cependant, en fabricant à plus grande échelle, la part mécanique diminuerait fortement contrairement à la part électronique moins compressible.

#### 3.2.1 La capacité d'auto-assemblage<sup>6</sup>

La plus grande caractéristique du s-bot est de pouvoir s'auto-assembler avec ses pairs. Les comportements collectifs du s-bot dépendent énormément du succès de cette phase délicate d'accrochage. Nous allons présenter succinctement les résultats présentés dans le papier "Autonomous self-assembly in swarm-bots" [31].

Dans les expériences présentées, les s-bot doivent s'assembler à leurs pairs ou à une proie ayant le même profil d'accroche. Les expériences ont été effectuées sur différents terrains (figure 4.39a) à des conditions initiales différentes avec un ou plusieurs s-bots (figure 3.29b).

Les résultats peuvent être résumés dans la table 3.2. Les taux de réussite dans les différents terrains sont très élevés. Ceci est dû à différents points :

- La morphologie de la pince et de l'anneau d'accrochage permettent une connexion même si l'alignement et le positionnement n'est pas précis.
- L'assemblage peut se faire sur plus des 2/3 de la périphérie du s-bot.

6. Le comportement en simulation a été développé par Roerich Gross et transposé sur les robots réels avec mon aide.

N	D(cm)	T. plat		T. peu accidenté		T. très accidenté	
		R(%)	T(s)	R(%)	T(s)	R(%)	T(s)
1	25	100.0	18.0	100.0	29.7	95.0	24.9
1	50	100.0	28.7	100.0	36.9	95.0	71.3
6	70	97.6	86.7	98.33	121.2	90.83	115.4

**Tableau 3.2** Résultats des expériences d’auto-assemblage sur les trois types de terrain. N=nombre de s-bot qui se déplace, R=taux de réussite, T=moyenne du temps d’exécution

- Les capteurs dans la pince et les informations intrinsèques des moteurs permettent de déterminer le succès de l’assemblage, sinon une marche arrière et une nouvelle approche sont effectuées.
- Les chenilles et les roues fournissent une mobilité permettant une approche et un positionnement adéquat même en terrain accidenté.

Les temps d’exécution montrent qu’il faut logiquement plus de temps pour effectuer un assemblage en terrain accidenté. Le terrain très accidenté est particulièrement difficile pour un robot seul sur la distance de 50cm. En effet, lors de l’approche, le robot perd de temps en temps la cible à cause des irrégularités du terrain et du coup, la recherche en effectuant un tour sur lui-même, augmentant significativement le temps d’approche.

### 3.3 Leçons apprises, limites et améliorations

Le s-bot nous a d’abord appris que faire petit et intégré n’est pas facile et complique bien des aspects. Par exemple, la première version des circuits électroniques ne pouvait pas être montée dans le robot. Il y avait à chaque fois un composant mal placé qui touchait soit une pièce mécanique ou le composant d’une autre carte. Quand on a dû changer de moteur pour une version plus grande et plus puissante, il a fallu à nouveau détourner et router deux PCB.

Nous avons aussi totalement sous-estimé le coût, le temps de fabrication et de montage de 35 s-bot. Quand il faut fabriquer sur mesure plus de 100 pièces par robot, soit par moulage ou par usinage la tâche devient vite difficile. Inclure l’électronique dans une pièce moulée et très intéressant et pratique, mais demande un effort de fabrication non négligeable.

Avec le s-bot, nous avons bien amélioré la technique de moulage et réaliser des pièces très complexes, irréalisables par des techniques conventionnelles. Nous avons aussi vu ses limites, les pièces n’ont pas une tolérance suffisante pour un robot aussi petit et intégré que le s-bot et de nombreux ajustements sont nécessaires au montage. Aussi, quand il



faut 1 à 2h pour que la pièce durcisse dans le moule, fabriquer 35 pièces ne se fait pas rapidement. De plus, il faut compter 5 à 10 minutes pour préparer un moule et presque autant pour démouler une pièce. Ainsi, une personne ne peut que s'occuper de 3 à 4 moules en parallèle, ce qui fait que le temps de fabrication avec cette technique est très long.

Une des limitations du s-bot est tout d'abord énergétique. Leur autonomie est de 1 à 2h et les batteries ne sont pas amovibles. Il faut ainsi laisser charger le robot pendant 2h après chaque utilisation. Cela fait qu'ils peuvent être au mieux disponibles que 50% du temps. De plus, les deux éléments batteries Li-Ion ont été mis en série et n'ont pas de circuit de protection. Deux robots ont d'ailleurs brûlé lors d'une charge. Il a fallu rajouter à chaque robot un PCB contrôlant la charge et changer tous les packs batteries. Cela a représenté un travail considérable car il faut au minimum 2h de démontage et remontage pour accéder aux batteries. Par conséquent, mettre deux éléments en série n'est pas conseillé sans avoir un circuit de protection adéquat et un chargeur accédant au point milieu. De ce fait, nous en tirons comme conclusion qu'il est indispensable de pouvoir changer les batteries pour un robot de recherche.

Les capteurs de proximité du s-bot sont également perfectibles, ils mesurent légèrement leur propre bord du corps. Pour limiter cet effet, il a été décidé de faire toutes les parties plastiques externes du corps du s-bot en noir. Du coup, les s-bots ne se distinguaient plus entre eux et il a fallu rajouter une bande réfléchissante. Je conseille plutôt pour des robots collectifs de faire un corps blanc avec des capteurs libres de toute arête ou bord qui pourrait réfléchir légèrement des infra-rouges.

Le capteur de traction du s-bot est très utile et presque indispensable pour un robot collectif pouvant s'auto-assembler. Cependant, l'électronique des jauges de contrainte demande un soin particulier et doit pouvoir être calibrée facilement.

Les deux capteurs d'humidité et de température sont quasiment inutiles. Ils n'ont été utilisés qu'une fois pour trouver la position d'un humidificateur.

La pince rigide du s-bot comprend un capteur de couple dans son degré de liberté verticale. Le design intégré a demandé de la place et a affaibli la partie mécanique. Ainsi, la pince se casse souvent et c'est fort embêtant car elle est indispensable au robot. Ce capteur n'est pas vraiment nécessaire ou ne devrait pas affaiblir autant la partie mécanique.

Un autre point limitant du s-bot est sa caméra omnidirectionnelle. L'utilisation d'un miroir en demie-sphère fait qu'un tiers de l'image ne comporte que la réflexion du robot, ce qui est inutile. De plus, il s'agit d'une caméra type webcam passant par un bus USB. Le processeur doit donc encore décoder le Jpeg avant de pouvoir faire le traitement d'image.

Finalement, le bus principal I<sup>2</sup>C est vite saturé et ne permet pas d'avoir une communication multi-mâtres efficace. Ce bus est plutôt fait pour connecter un capteur ou une puce à un microcontrôleur, mais ne permet pas de communiquer entre de nombreux microcontrôleurs. Les microcontrôleurs du s-bot ne sont pas programmables par le bus I<sup>2</sup>C, ce qui demande de démonter le robot pour programmer ceux-ci.

Toutes ces limitations ci-dessus ont été améliorées pour le robot marXbot qui est décrit dans le prochain chapitre.

### 3.4 Conclusion

Le s-bot est un robot novateur par son intégration et la multitude de ses fonctions. Sa petite taille permet de faire des expériences dans des endroits restreints. La mobilité fournie par les chenilles associées aux roues est novatrice et permet une bonne mobilité sur du terrain accidenté tout en garantissant un contrôle simple du robot. Les richesses de ses capteurs, le système d'accrochage accommodant relativement simple et la puissance de calcul embarqué permettent au s-bot de facilement s'auto-assembler avec ses coéquipiers. Le tout permet d'entrevoir des capacités collectives nouvelles et super-linéaires (voire chapitre 6).

Néanmoins, l'intégration système et la quantité des fonctions du s-bot en font un robot complexe et pas vraiment bon marché.

## 4 Le MarXbot



**marXbot** est l'acronyme de "**modular all terrain Linux based robot**"

Il est conçu pour faire de nombreuses expériences en robotique mobile. Que ce soit en vision, en collectif ou en construction, le marXbot est un outil robuste et complet, incluant des technologies récentes et adaptées aux systèmes embarqués. Le marXbot peut être considéré à ce jour comme l'état de l'art des robots de recherche de cette taille.

J'ai développé la plus grande partie de la mécatronique du robot avec l'aide de plusieurs personnes du laboratoire. C'est un outil abouti qui découle des expertises de ces personnes en mécanique, électronique et logiciel embarqué. Il se base, notamment, sur toutes les expériences acquises après le développement du s-bot et de l'e-puck.

Le marXbot a la particularité d'être modulaire. Ainsi, suivant le type d'expérience à réaliser, un ou plusieurs modules peuvent y être ajoutés. Dans ce chapitre, le marXbot ainsi qu'une expérience d'auto-assemblage comparable au s-bot sont présentés.

### 4.1 Design du marXbot

#### Le cahier des charges

Le robot marXbot a été développé dans le laboratoire de systèmes robotiques (LSRO) de l'EPFL. Sa tâche est de remplacer le s-bot pour le projet "Swarmanoid" qui est la suite du projet "Swarm-bots", mais aussi pour d'autres projets du laboratoire. En effet, les différents pôles de recherche du laboratoire avaient besoin d'une plate-forme robotique mobile ayant les caractéristiques suivantes :

- Avoir une bonne autonomie énergétique.
- Disposer d'une grande gamme de capteurs permettant une navigation avancée (réalisation de SLAM).
- Posséder un système de vision efficace.
- Avoir une bonne mobilité.
- Disposer de suffisamment de puissance de calcul pour effectuer de la recherche algorithmique.
- Être orienté pour des tâches collectives.

Le but recherché était également d'uniformiser les technologies développées au laboratoire, afin de diminuer le temps de développement et d'augmenter la collaboration au niveau mécatronique et logiciel.

#### L'approche

Au vu des besoins du laboratoire, il a été tout de suite souhaité de faire un robot modulaire. L'idée initiale était d'avoir une base commune pour la mobilité et d'une partie supérieure responsable de la vision et munie de capteurs longues distances. Un module spécifique à chaque tâche pouvait être ajouté entre ces deux parties (figure 4.1). Dans la base, il a été décidé d'y mettre également les batteries, des capteurs de proximité périphérique et des capteurs de sol. Dans la partie supérieure étaient réunis les capteurs longues distance, une caméra, des LED RGB et le processeur principal. En bref, on avait séparé les capteurs et actionneurs bas niveau rassemblés dans la base et les capteurs plus évolués dans la partie supérieure. Par la suite, les capteurs longue distance ont été séparés de cette partie supérieure, devenant le module scanner décrit dans la section 4.1.1

Lors du design du marXbot, on voulait corriger les faiblesses du s-bot :

- son intégration et sa miniaturisation trop poussées qui compliquent le design et la maintenance et limitaient sa flexibilité
- sa faible autonomie et ses batteries prises au centre de la base
- les problèmes rencontrés avec le bus I<sup>2</sup>C et sa programmation centrale, comme la surcharge ou la paralysie du bus

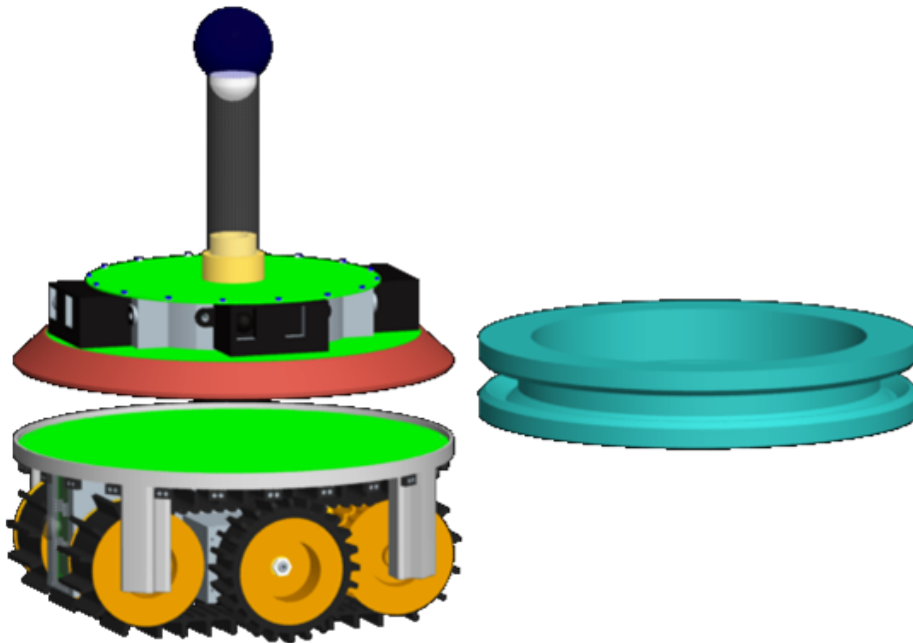


Figure 4.1 Le premier concept du marXbot

En se permettant de faire un robot légèrement plus grand, on pouvait éviter pas mal de problèmes liés à la miniaturisation du s-bot. On pouvait également se permettre des batteries beaucoup plus grandes et donc plus d'autonomie au robot. De plus, il a vite été décidé que les batteries soient interchangeables, ne bloquant pas le robot lors de la charge. Le concept de changement de batterie à chaud et automatique permettrait même des expériences en continu.

L'architecture électronique du marXbot est comparable à celle du s-bot. Un processeur puissant et central tournant sous Linux s'occupe de la vision et du contrôle haut-niveau tandis que les moteurs et les capteurs sont gérés par des microcontrôleurs. Cependant, les évolutions technologiques nous permettent d'avoir des microcontrôleurs beaucoup plus puissants. Ils sont interconnectés grâce à un bus CAN multi-maître et un nouveau concept d'échange d'informations par événement a été développé (voir section 4.1.3).

Pour la morphologie du marXbot, la forme cylindrique du s-bot convenait également. En outre, le concept des roues et des chenilles était une réussite et donc a aussi été utilisé pour le marXbot.

Pour le projet Swarmanoid, les robots n'ont pas l'utilité d'avoir un système d'accrochage flexible ou permettant de soulever un autre robot. Ainsi le système d'auto-assemblage a été simplifié.

Le marXbot et les modules relatifs au projet Swarmanoid seront présentés dans la partie suivante. Dans cette configuration, le marXbot est également appelé "footbot " dans le projet.

### 4.1.1 La mécanique

La figure 4.2 représente le marXbot. Le robot est de forme cylindrique d'un diamètre de 17cm, d'une hauteur de 29cm et pèse 1.8kg. Il se compose :

- D'une base mobile avec des capteurs de proximité et d'une batterie.
- D'un module d'accrochage lui permettant de s'auto-assembler avec ses pairs.
- D'un module de distance et de relèvement lui permettant de détecter, d'identifier et de communiquer avec ses pairs.
- D'un scanner rotatif longue distance lui permettant de mesurer la distance d'objet jusqu'à 1.5 m
- D'un module CPU principal et vision qui est la base du contrôle logiciel haut niveau.

La figure 4.3 présente tous ces modules.

### La base mobile

Le design de la base (figure 4.4) a été conçu autour de la batterie. Le cahier des charges voulait qu'on puisse aisément changer la batterie. Le pack batterie a même été conçu pour être changé automatiquement. Dans cette optique, la batterie s'extrait de façon horizontale, ce qui permet de simplifier le mécanisme d'une station d'échange de batteries.

Comme le s-bot, le marXbot a un système roue-chenille pour se mouvoir. Cette combinaison fonctionne bien et seuls quelques défauts ont dû être corrigés. La tension dans la chenille peut ainsi être réglée. Un moteur-réducteur standard entraîne directement la roue extérieure. Celui-ci est placé en partie à l'intérieur de la roue pour gagner de la place.

La figure 4.5 représente un plan éclaté de la base. Toutes les pièces se fixent sur une pièce principale (figure 4.6). Deux circuits verticaux servent de contact pour la batterie et de support pour l'électronique des moteurs. Un circuit comportant 24 capteurs de proximité IR se fixe au sommet de la base.

Les roues de la chenille sont faites de deux pièces identiques. Selon la manière dont on les imbrique, on peut choisir la largeur des roues afin de tenir compte de la courroie d'entraînement : à l'arrière du robot, elles sont plus larges à cause de la courroie reliant la chenille à la roue centrale.

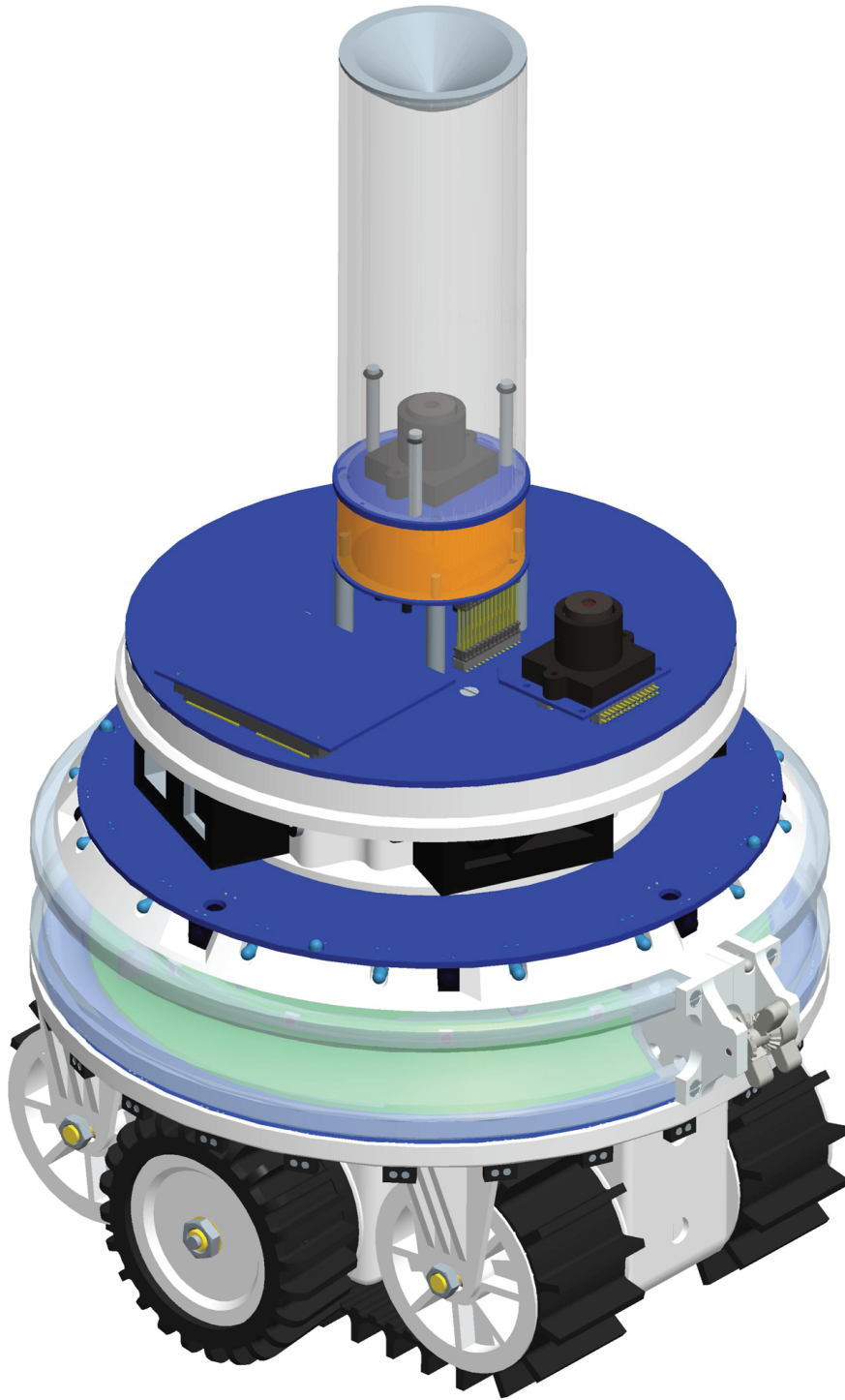


Figure 4.2 Dessin CAD du marXbot.(3D)

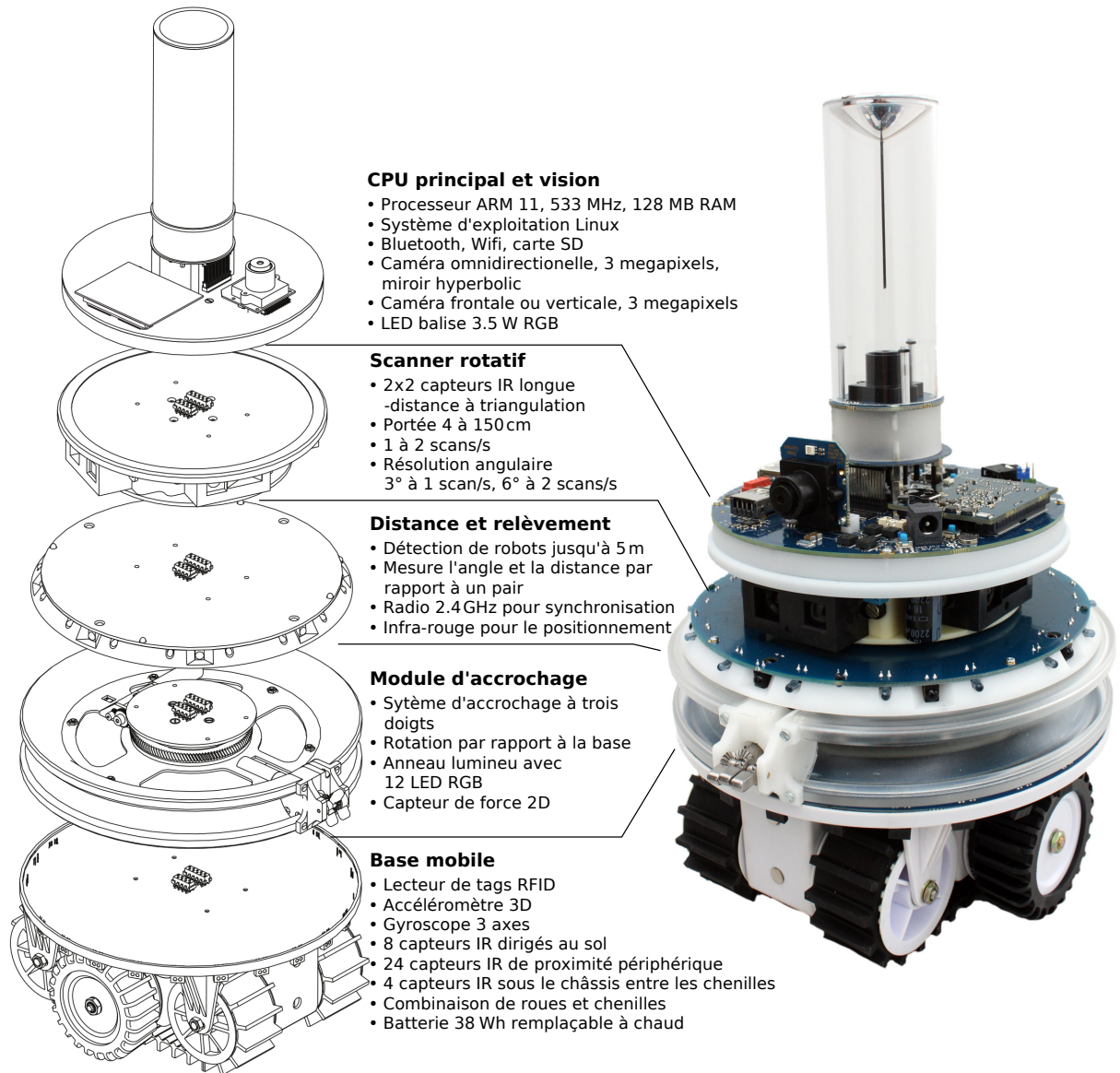


Figure 4.3 Le marXbot et ses modules



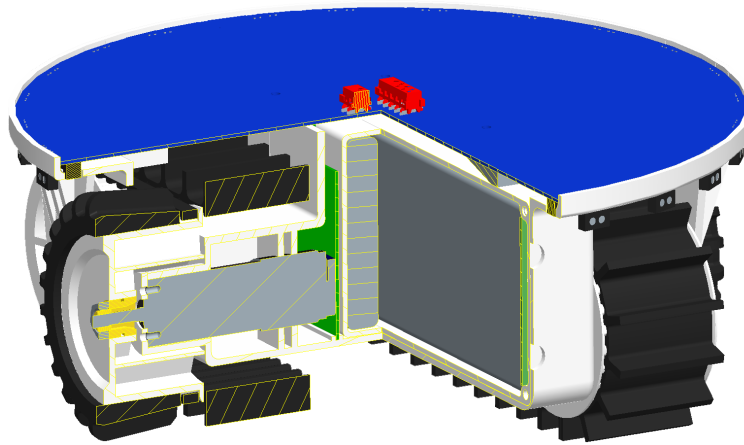


Figure 4.4 Le vue 3/4 de la base du marXbot

### Module d'accrochage<sup>1</sup>

Le module de connexion du marXbot lui permet de s'auto-assembler avec d'autres marXbots, mais aussi avec des handbots présentés dans le chapitre 5. Par rapport au s-bot, le système est plus simple mais reprend les principes de sa fiabilité. La connexion peut se faire sur la plus grande partie de la circonférence. Le mécanisme se compose de trois doigts pivotants (figure 4.7). Les deux doigts extérieurs pivotent à l'opposé du doigt central. La circonférence de la couronne pivotante a un profil s'adaptant au système de doigts. Ainsi pour se connecter, un marXbot doit seulement s'avancer contre l'un de ses pairs en essayant de se centrer, puis une fois qu'il le touche, ouvrir le mécanisme pour établir une connexion rigide.

La figure 4.8 montre un plan éclaté du module d'attache. C'est l'un des modules les plus compliqués d'un point de vue mécanique. **A** est le circuit imprimé (PCB) contrôlant le capteur de force. Au dessus de celui-ci, une pièce de support **B** assure l'intégrité structurelle et d'attache des autres parties. Cette pièce de support est attachée à la base du marXbot et traverse l'entier du module. La pièce **B** a également un trou au centre, permettant le passage des connecteurs **K** qui contiennent la puissance et les bus de données. La partie **C** fournit le support pour le système de rotation. Elle est reliée à **B** par quatre plaquettes métalliques en acier ressort. Sur chaque plaquette, une jauge de contrainte est collée, ce qui permet au robot de mesurer la force appliquée par d'autres marXbots sur le plan horizontal. La partie tournante se compose des sous-parties **E** à **H**. Un câble flexible **D** assure la liaison entre la partie fixe et la couronne tournante. Il permet une rotation de  $360^\circ$  à  $-220^\circ$ . La couronne tournante

1. La mécanique de ce module a été développée avec Tarek Baaboura

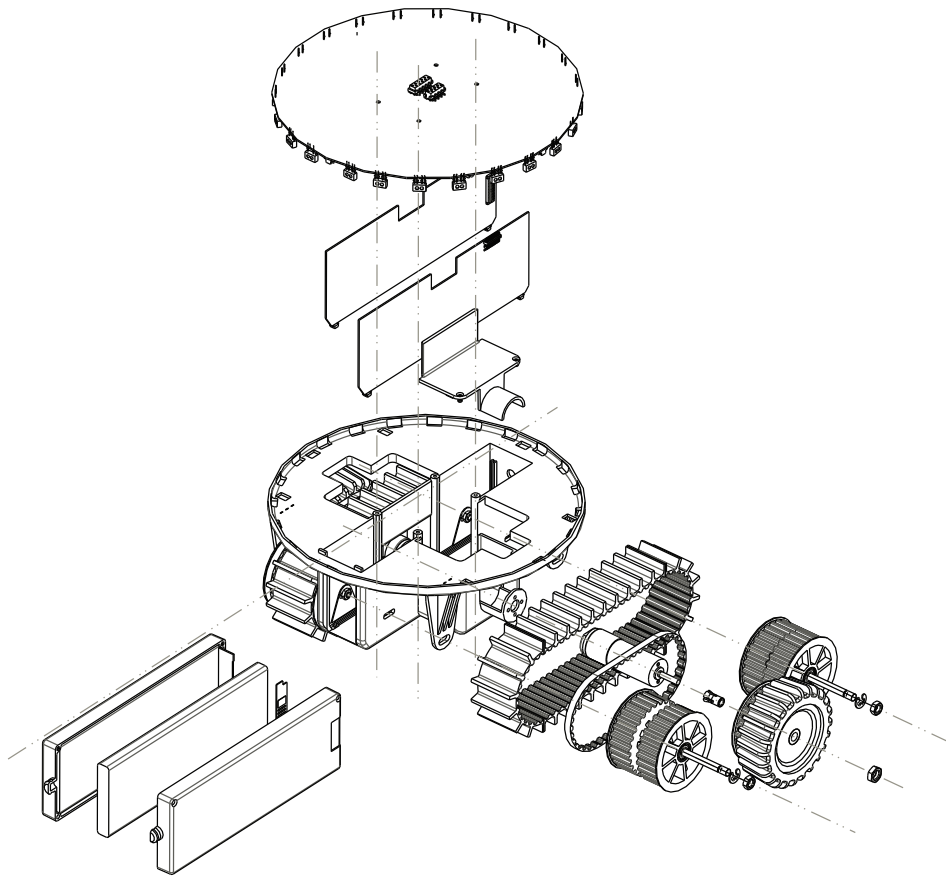


Figure 4.5 Le plan éclaté de la base mobile

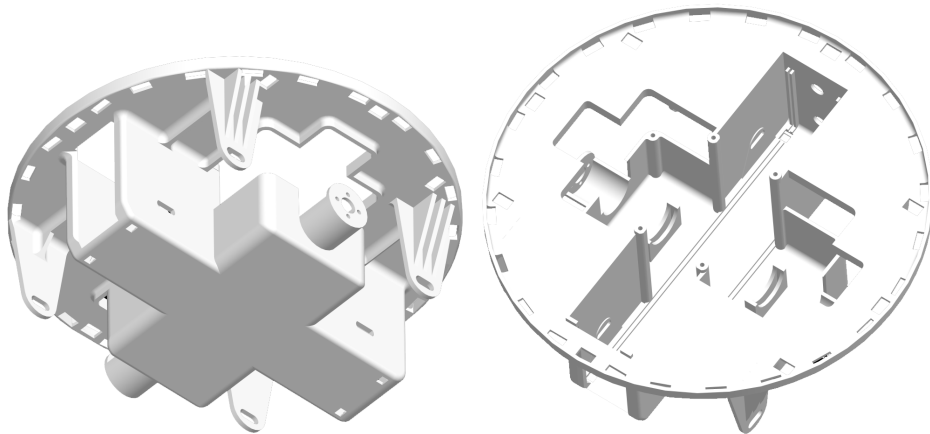


Figure 4.6 La pièce principale de la base

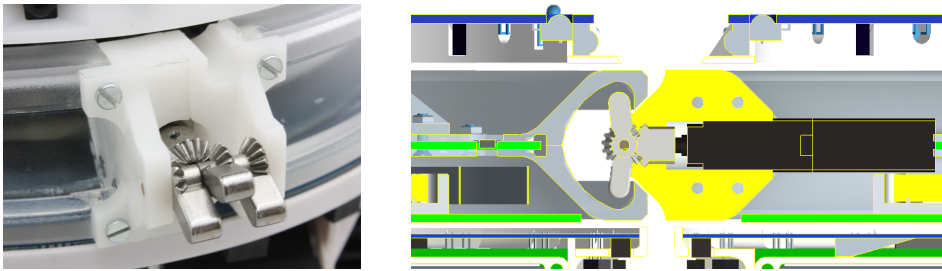


Figure 4.7 Système d'accrochage

est principalement composée de deux pièces en plastique **F** enfermant un PCB **H**. Ce PCB contient un microcontrôleur qui contrôle le moteur de rotation et le moteur du système d'accrochage, ainsi que 12 LED RGB. Le moteur de rotation entraîne une vis sans fin à trois filets, rendant la rotation réversible. Ainsi, une fois connecté, la tourelle de rotation pivote automatiquement en suivant le mouvement des chenilles du robot. Le système de connexion **E** est basé sur trois doigts entraînés par des engrenages coniques (figure 4.7). Au sommet du module, un PCB **L** assure la liaison électrique et mécanique à un autre module. Cette connection est identique pour chaque module, ce qui permet la modularité du robot.

### Le module distance et relèvement (D&R)<sup>2</sup>

Le module "distance et relèvement" (figure 4.9) sert au marXbot à détecter les autres robots et à mesurer la distance et la direction de ceux-ci. Grâce à ce module, il peut aussi échanger des messages ciblés ou envoyer des messages globaux aux robots environnants qui ne répondent qu'en fonction de leur positionnement géographique. Ces particularités sont utiles lors de développements de comportements collectifs.

Le module se compose de deux pièces : un circuit électronique et un cache plastique. Pour la localisation, le module émet un pulse infra-rouge le plus uniforme possible. Il y a 20 LED infra-rouges, 16 réparties sur le pourtour (figure 4.10) et 4 émettant à la verticale du robot, afin de communiquer avec les robots volants du projet. Les récepteurs infra-rouges sont au nombre de 12. 8 sont répartis sur le périmètre et 4 sont placés verticalement. La partie électronique et le principe de fonctionnement sont expliqués dans la section 4.1.2.

<sup>2</sup> Ce module a été développé sur la base d'un design développé au DISAL par Jim Pugh et Xavier Reamy [68]

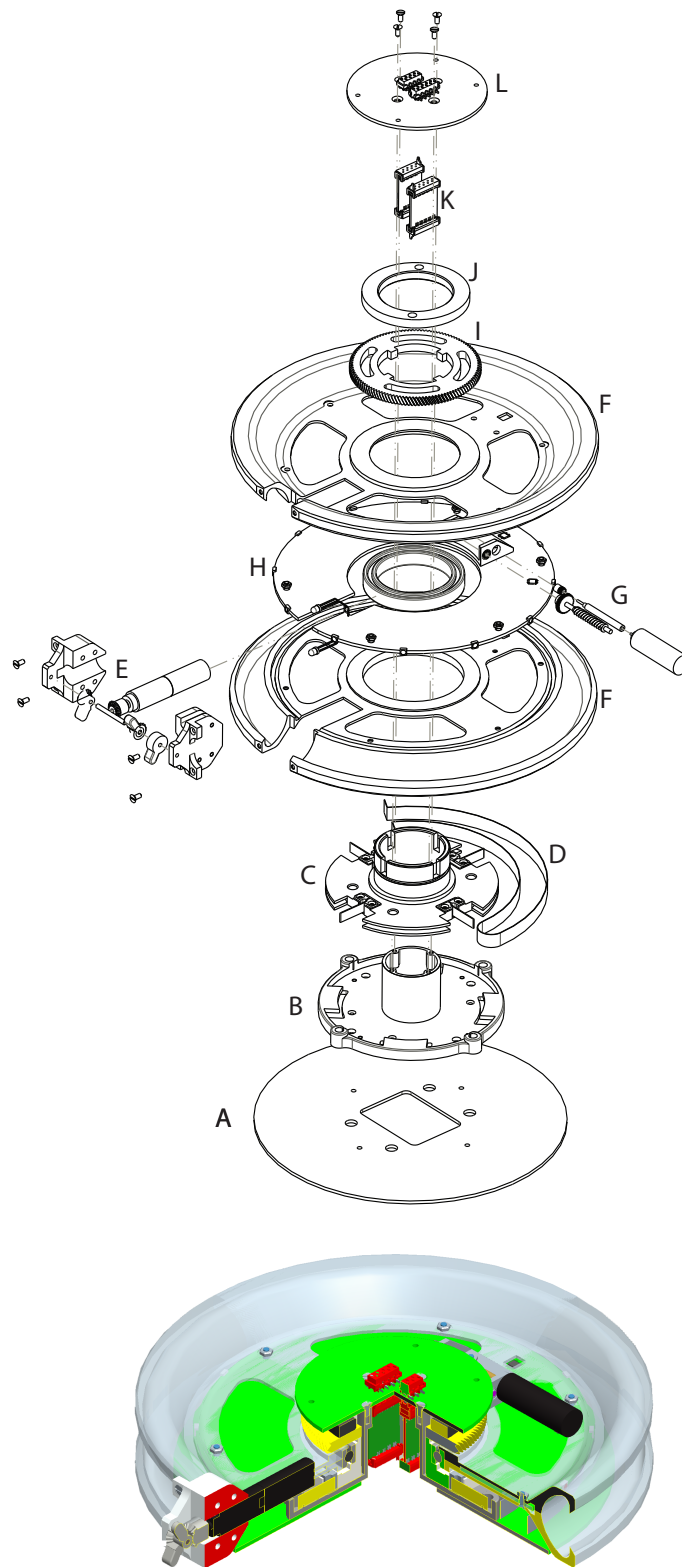


Figure 4.8 Plan éclaté et vue 3/4 du module d'accrochage

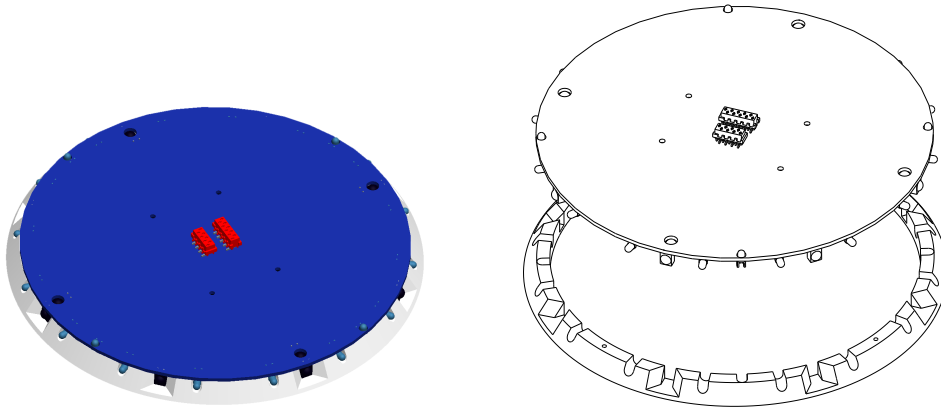


Figure 4.9 Le module distance et relèvement

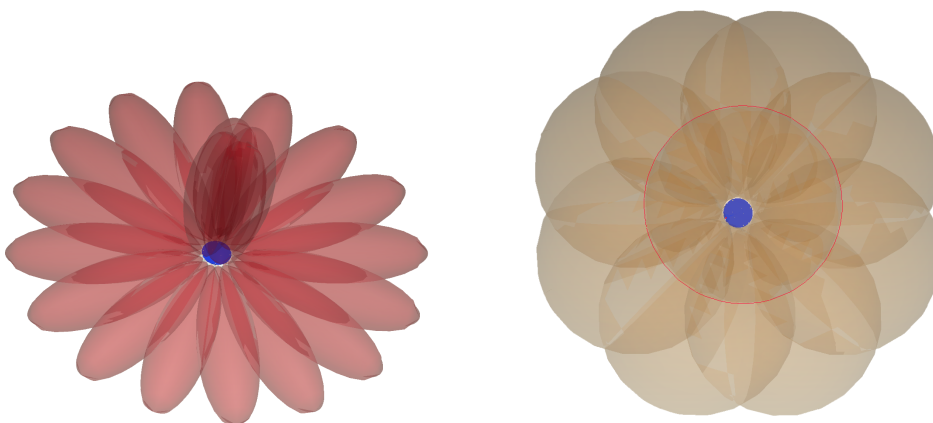
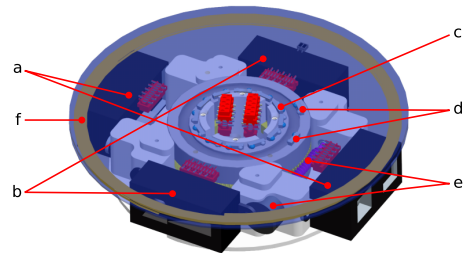
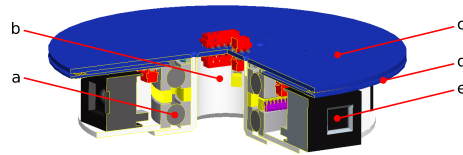


Figure 4.10 Les faisceaux d'émission et réception du module distance et relèvement

**Figure 4.11** Dessin CAD du scanner, avec les PCB en semi-transparent. **a:** capteur SHARP longue distance, **b:** capteur SHARP courte distance, **c:** communication IRDA pour la transmission de données (12×fixes), **d:** communication IRDA pour la transmission de données (2×rotatifs), **e:** moteur de rotation avec réducteur à vis sans fin, **f:** bobines d'induction directement dessinées sur les PCBs



**Figure 4.12** Dessin CAD du scanner, avec une vue 3/4. **a:** roulement à billes en plastique, **b:** trou au centre pour passer la puissance et les bus de données, **c:** PCB fixe (self primaire), **d:** PCB rotatif (self secondaire), **e:** capteurs de distance



### Le scanner longue distance<sup>3</sup>

Le module scanner (Figures 4.11 et 4.12) est composé de 2 paires de capteurs longue distance de marque SHARP. Ceux-ci tournent en permanence pour scanner le périmètre du robot. Une paire mesure par triangulation les distances de 4 à 30cm, l'autre de 20 à 150cm. La table 4.1 résume ces caractéristiques.

Spécification	Valeur
diamètre	130 mm
hauteur	29 mm
poids	220 g
consommation	2 W
coût	390 USD
capteur de distance IR	2× GP2Y3A001K0F (4–30 cm) 2× GP2Y3A002K0F (20–150 cm)
distance de fonctionnement	4 à 150 cm
vitesse scan max.	2 scans/s
résolution angulaire	3° at 1 scan/s, 6° at 2 scan/s

**Tableau 4.1** Caractéristiques du scanner de distance

La figure 4.13 montre un plan éclaté du module. Un moteur muni d'une vis sans fin assure la rotation. Deux roulements à billes en plastique

3. La mécanique de ce module a été développée avec Tarek Baaboura

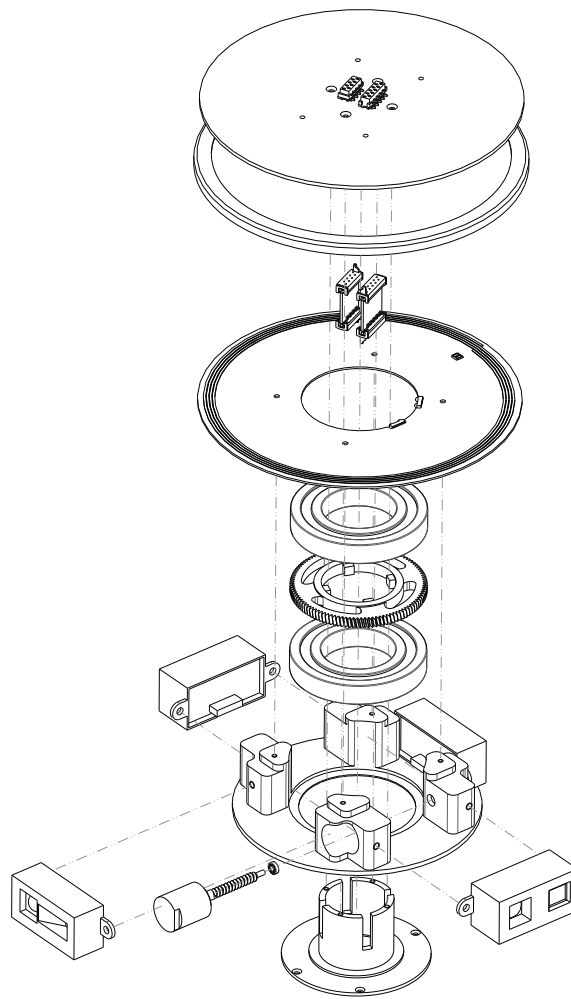


Figure 4.13 Plan éclaté du module scanner longue distance

assurent le guidage. Une transmission par induction est utilisée pour transférer la puissance sur la partie tournante. Les données sont transférées grâce à des émetteurs et récepteurs infra-rouges (IRDA).

Des contacts glissants ont été étudiés et testés comme contacts électriques. Cependant, la physionomie du module ne permettait pas d'installer ceux qui auraient été efficaces. De plus, selon nos tests, la fiabilité à long terme n'était pas assurée. Ainsi, la méthode sans contact choisie assure une transmission fiable et indépendante de l'usure. L'électronique est décrite plus amplement dans la section 4.1.2.

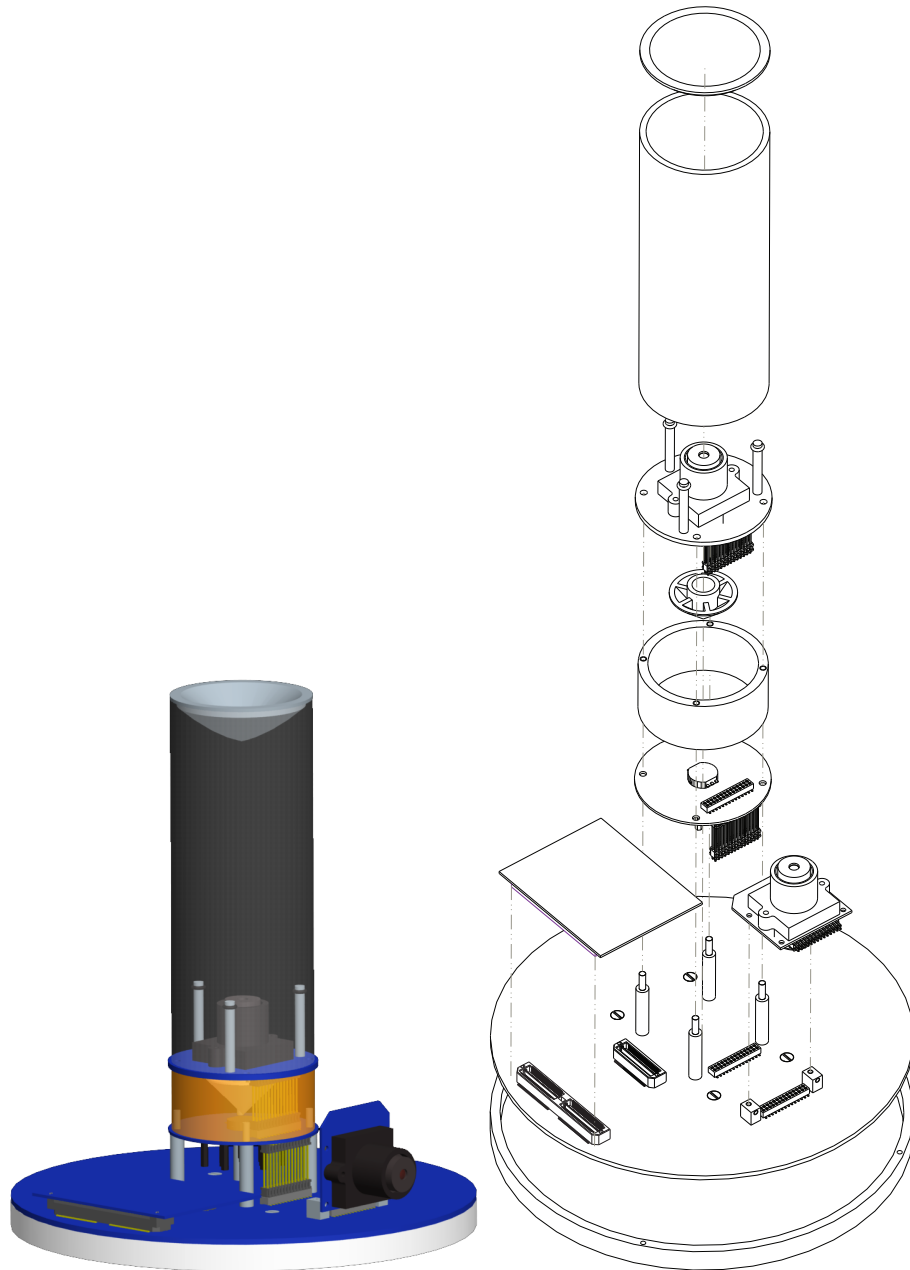


Figure 4.14 Le CPU principal et la vision



### CPU principal et vision

Le CPU principal se compose d'un processeur i.MX31 de Freescale, de deux caméras et d'une LED balise puissante.

La carte processeur se connecte sur la carte mère principale comportant toutes les interfaces nécessaires au robot. Une description plus détaillée se trouve dans la section électronique 4.1.2.

La première caméra, située à l'avant du robot peut être soit placée horizontalement, soit verticalement. La seconde caméra au sommet du robot regarde un miroir conique, ce qui permet au robot d'avoir une vue omnidirectionnelle. Le profil du miroir<sup>4</sup> a été optimisé pour avoir un champ de vision le plus utile possible. Ainsi le robot voit d'un angle de 30° à 95° par rapport à l'axe verticale (figure 4.15). Il évite ainsi de se voir lui-même et perçoit 5° au dessus de l'horizon. Ce miroir est placé au sommet d'un tube en verre. En son centre, une tige en carbone a été ajoutée pour limiter les reflets dans le tube.

Ce module comporte aussi une LED RGB de puissance servant de balise au robot. Il peut ainsi, grâce au miroir, détecter ses coéquipiers et transmettre des messages colorés. Cette LED se trouve sous la caméra omnidirectionnelle. La lumière de la LED est réfléchiée par un miroir conique, puis diffusée par un tube en plexiglas sablé. Un radiateur est placé sous le PCB de la LED, pour diffuser les 3.5W de puissance de celle-ci.

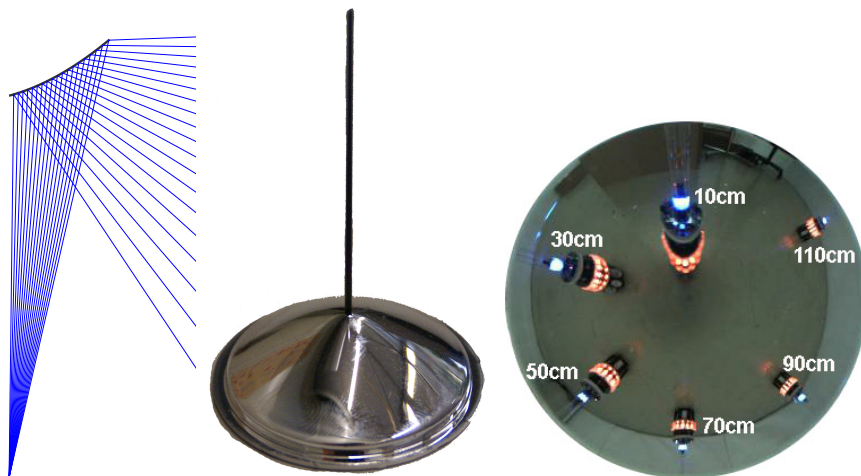


Figure 4.15 Le profil du miroir

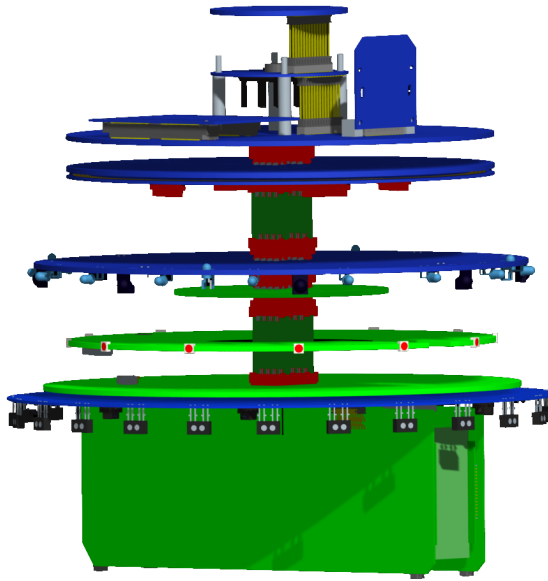


Figure 4.16 Les circuits électroniques du marXbot

#### 4.1.2 L'électronique

L'électronique du robot marXbot est très complète et répartie dans chaque module. Celle-ci gère au total 5 moteurs, 41 LED et 80 capteurs. Elle embarque 10 microcontrôleurs dsPIC33 et un processeur i.MX31 de Freescale tournant sous Linux. Le tout est reparti sur 21 circuits imprimés représentés dans la figure 4.16. Dans cette section, nous allons décrire partie par partie l'électronique du robot.

La figure 4.17 représente une vue d'ensemble du schéma électronique du marXbot. Au sommet du robot, il y a l'ordinateur principal i.MX31 contrôlant les deux caméras, le Wifi et un système sonore. Il y a encore un dsPIC33 servant de translateur CAN entre l'IMX et le reste du robot. Un dspic33 gère aussi la LED balise.

Le scanner est composé de deux dsPIC33, l'un sur la partie rotative gérant les capteurs et l'autre fixe gérant l'induction et l'interface avec le robot. Le module "distance et relèvement" est géré par un dsPIC33. Deux dsPIC33 sont nécessaires pour le module d'accrochage. L'un pour la gestion des moteurs et des LED de la couronne et l'autre pour gérer les capteurs de force.

Dans la base, un dsPIC33 gère les 24 capteurs de proximité et les 8 capteurs au sol périphérique. Dans la partie gauche, un microcontrôleur

4. Le programme d'optimisation a été fait par Stéphane Magnenat

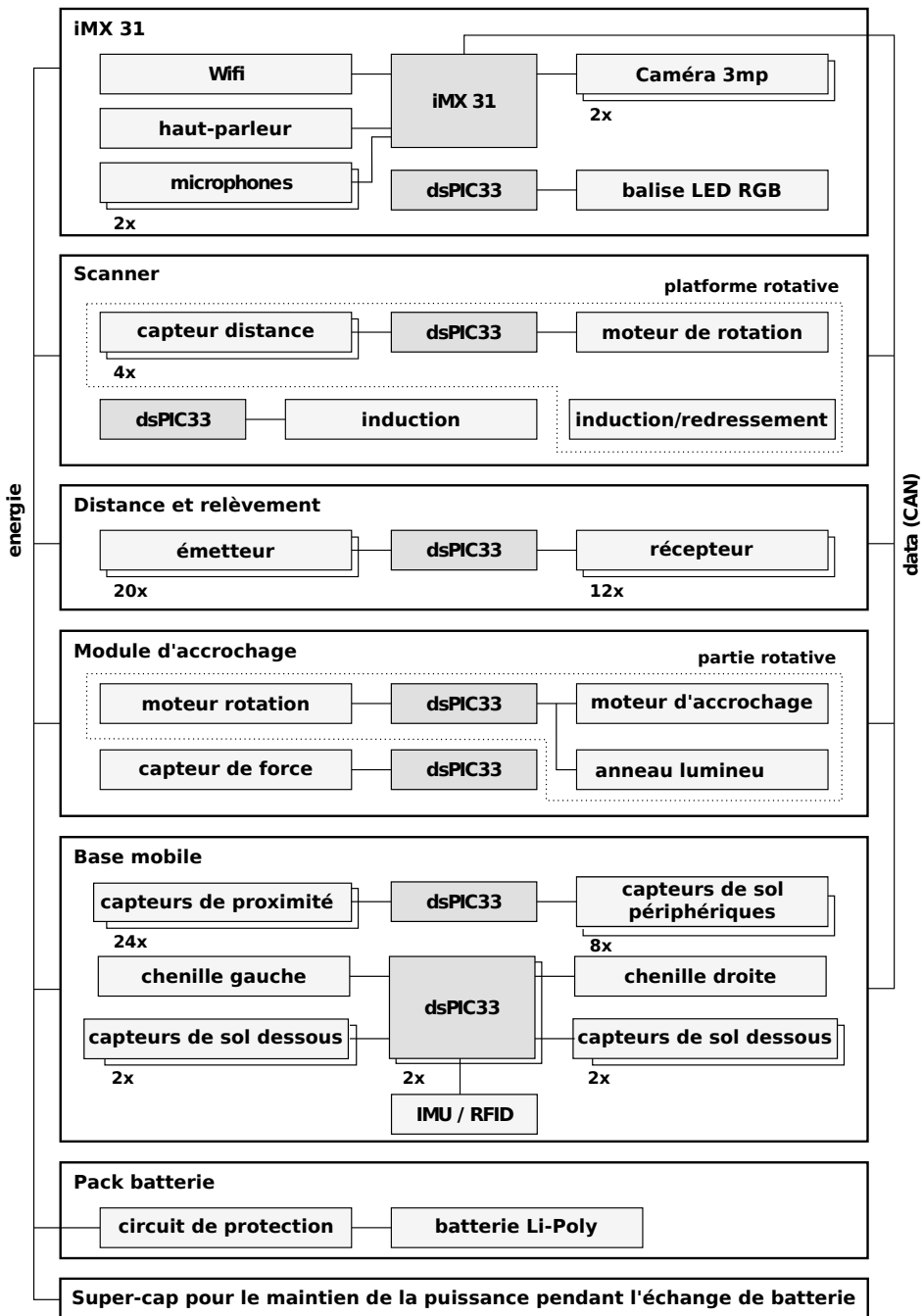


Figure 4.17 Schéma électronique complet du marXbot



**Figure 4.18** L'intérieur du pack batterie et les contacts à l'intérieur de la base

gère le moteur de la chenille, deux capteurs IR au sol, un lecteur RFID et une IMU. La chenille droite est contrôlée par un dsPIC33 qui gère également deux capteurs IR au sol.

### La batterie et la gestion d'énergie <sup>5</sup>

Le pack batterie du marXbot (figure 4.18) est composé de deux éléments Lithium-Polymère mis en parallèle :

Type	No Référence	Capacité	Autonomie	T. Charge
2×LiPo	GSP0845131	2×5300mAh 3.6V 38Wh	4-8h	4h

Un circuit de protection coupe la batterie en cas de sur ou sous-tension, de sur-courant et de court-circuit. La tension et le courant sont mesurés en continu et accessibles par un bus I<sup>2</sup>C. Un intégrateur sur le courant sert de jauge.

Deux PCB servant de contact extérieur sont soudés à 90° sur le PCB de protection. Des contacts sont placés dans les PCB verticaux de la base (figure 4.18), assurant la transmission de la puissance et du bus I<sup>2</sup>C.

Pour permettre un échange automatique de batterie à chaud, une alimentation de secours a été ajoutée dans la base du marXbot. Celle-ci est faite grâce à deux "super-caps" de 10F chacune. Mises en série, contrôlées par un circuit dédié, elles emmagasinent suffisamment d'énergie pour maintenir le robot éveillé pendant 15 sec. Au retrait de la batterie, un capteur détecte son absence et commute l'alimentation sur les super-caps.

Pour tenir 15 sec, l'alimentation du robot est un peu spéciale. La figure 4.19 présente la structure de celle-ci. Ainsi il y a deux alimentations

<sup>5</sup>. Le design électronique de la batterie a été développé avec Florian Vaussard lors de son travail de diplôme

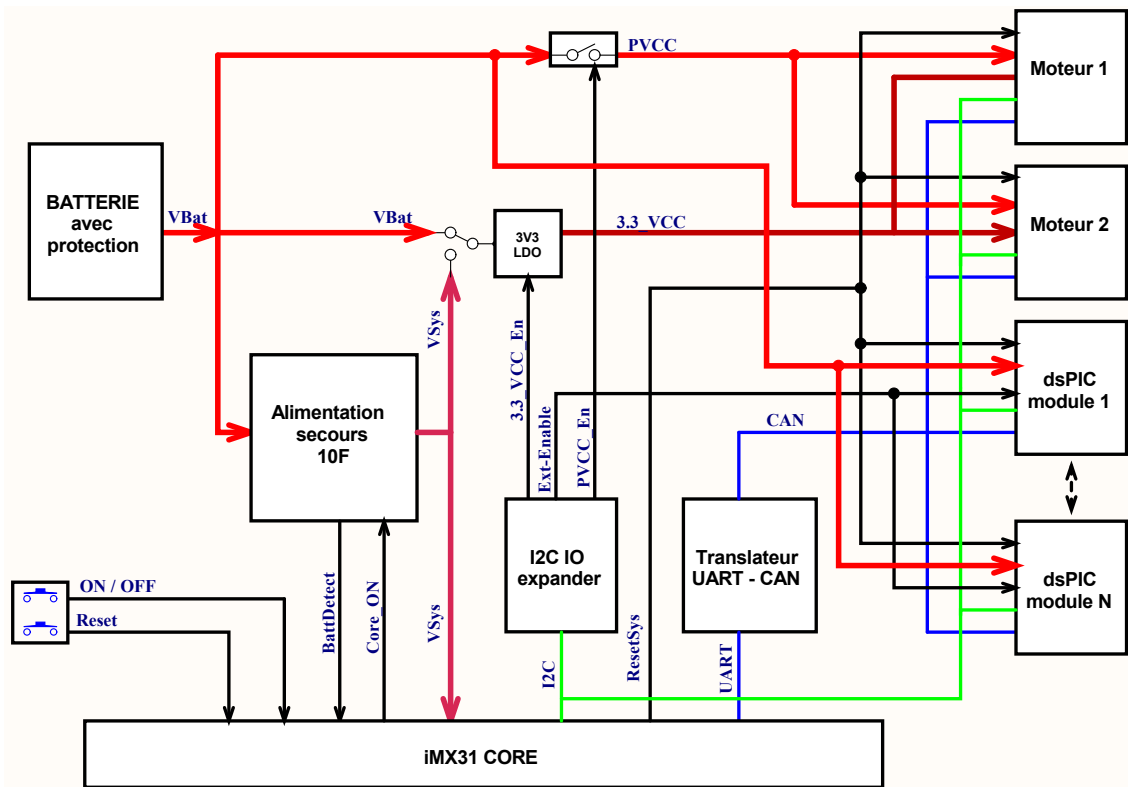


Figure 4.19 Schéma électronique de la puissance, des données et des signaux de contrôle.

dans le marXbot, l'une est "VBat" étant la tension venant directement de la batterie, l'autre "Vsys" est l'alimentation batterie couplée à l'alimentation de secours. "VBat" alimente directement toutes les parties électroniques de puissance, comme les moteurs, les LED et autres circuits non vitaux au robot. Quand on enlève la batterie, ces parties sont donc directement coupées. "Vsys" alimente essentiellement le CPU principal, la communication Wifi et quelques circuits choisis. On peut par exemple choisir "VBat ou Vsys" avec un "solder-jump" pour alimenter certains microcontrôleurs.

Le robot est muni d'un bus I<sup>2</sup>C, celui-ci est utilisé pour la gestion de l'alimentation. Le marXbot ne possédant pas d'interrupteur mécanique, toutes les sous-parties du robot peuvent être coupées électroniquement grâce au bus I<sup>2</sup>C. Un "IO expander" est utilisé pour allumer la puissance (PVCC En) ou l'électronique des moteurs (3.3 VCC En). Il fournit aussi un signal pour couper ou non les modules (Ext-Enable). Le bus I<sup>2</sup>C permet aussi de mettre en veille profonde les microcontrôleurs et de les réveiller. De ce fait, si pendant un certain temps un module n'est pas nécessaire au comportement, celui-ci peut être mis en veille, puis réveillé, retrou-

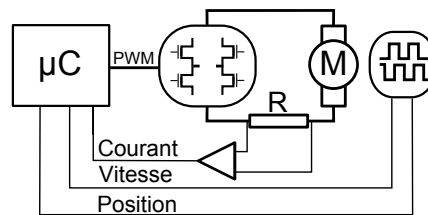


Figure 4.20 Schéma électronique d'un contrôle moteur

avant son état précédent. En veille, un dsPIC33 ne consomme que  $300\mu A$ , contrairement au 50 à 90mA en fonctionnement.

Le processeur i.MX31 n'est pas équipé avec le périphérique CAN. Ainsi, un translateur est nécessaire entre l'i.MX31 et les dsPIC33 pour le bus CAN où toutes les données transitent.

### Le contrôle des moteurs <sup>6</sup>

Le contrôle des moteurs dans le marXbot est proche de celui des s-bot. Électroniquement, les composants sont équivalents, même pont-H, même type de mesure du courant. Tous les moteurs ont un encodeur. La figure 4.20 résume le schéma. La seule différence vient du microcontrôleur qui, grâce aux évolutions technologiques, peut gérer jusqu'à 4 moteurs sans compter d'autres périphériques. Ceci est possible grâce à l'utilisation de dsPIC33 pouvant effectuer près de 40MIPS (Méga Instruction Par Seconde); bien plus qu'il n'en faut pour un contrôle de moteur DC.

Le code de contrôle <sup>7</sup> a aussi changé par rapport au s-bot. Il s'agit d'un régulateur complet de courant, vitesse et position. Comme le résume la figure 4.21, pour effectuer un contrôle en position, un régulateur PD donne la consigne au régulateur PID de vitesse qui lui même donne la consigne au régulateur PI de courant, qui finalement actionne le PWM du ponts-H du moteur. Ces régulateurs en cascade permettent à tout moment d'avoir des limitations en vitesse ou en courant. Ces limites sont réellement régulées, contrairement au s-bot, où lorsqu'elles étaient atteintes, cela limitait le PWM de sortie.

Le contrôle du courant a aussi été amélioré. Le filtre de l'amplificateur a été adapté pour faire un contrôle à 1kHz sur le courant du moteur. En intégrant celui-ci, on mesure aussi le courant moyen et on peut limiter celui-ci par rapport à la constante thermique du moteur. Ce qui permet

6. Développé avec Florian Vaussard lors de son travail de diplôme sur une base de travail de Daniel Baer.

7. Le code a été complètement réécrit par Philippe Rétornaz.

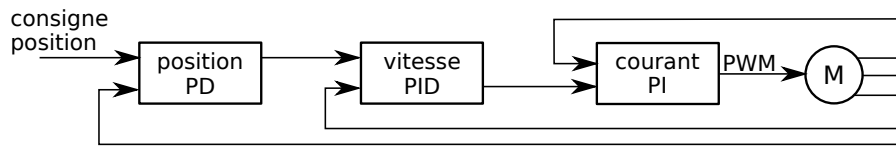


Figure 4.21 Contrôleur PID des modules moteurs

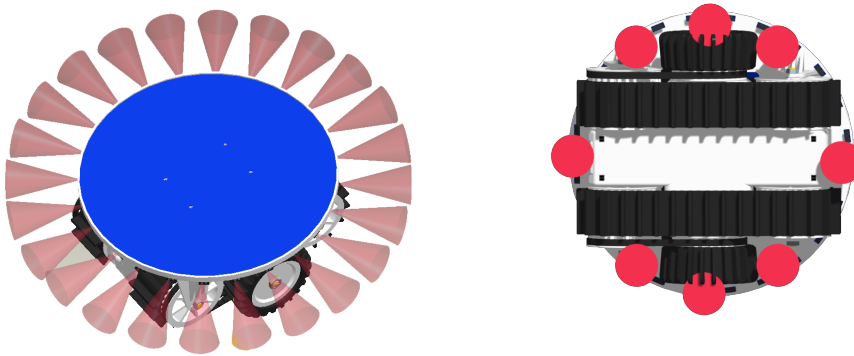


Figure 4.22 Les capteurs infra-rouges de proximité

pendant de courts instants de survolter le moteur tout en conservant son intégrité.

### L'électronique de la base<sup>8</sup>

La base comporte tous les capteurs nécessaires à la navigation locale. Elle comporte, en effet, 24 capteurs infra-rouges TCRT1000, comparables à l'e-puck et au s-bot, tout autour de son périmètre (figure 4.22) qui permettent au robot d'éviter les obstacles. Ces capteurs permettent de détecter des objets jusqu'à 5cm. Pour détecter les trous, le marXbot est muni de 8 capteurs infra-rouges TCND5000 dirigés au sol tout autour de son périmètre. Ces capteurs permettent de détecter convenablement des objets jusqu'à 12cm. Il y a 4 capteurs (figure 4.23) qui sont placés sous la base (2 à l'avant, 2 à l'arrière), permettant également de faire du suivi de ligne ou de la détection de zone à niveaux de gris variables.

Par rapport au s-bot, les capteurs de proximité périphériques n'ont pas d'amplificateur différentiel "sample-hold". Par contre, leur résistance de charge sur le photo-transistor est grande, ce qui implique une grande disparité des capteurs. C'est pourquoi une méthode de calibration et de

8. Développée avec Florian Vaussard lors de son travail de diplôme, Simon Fivat (IMU,RFID) lors d'un projet de semestre et Philippe Rétornaz (RFID).

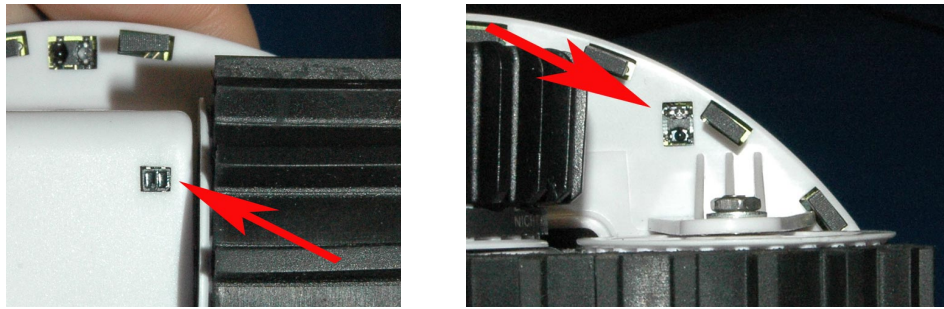


Figure 4.23 Les capteurs infra-rouges orientés vers le sol

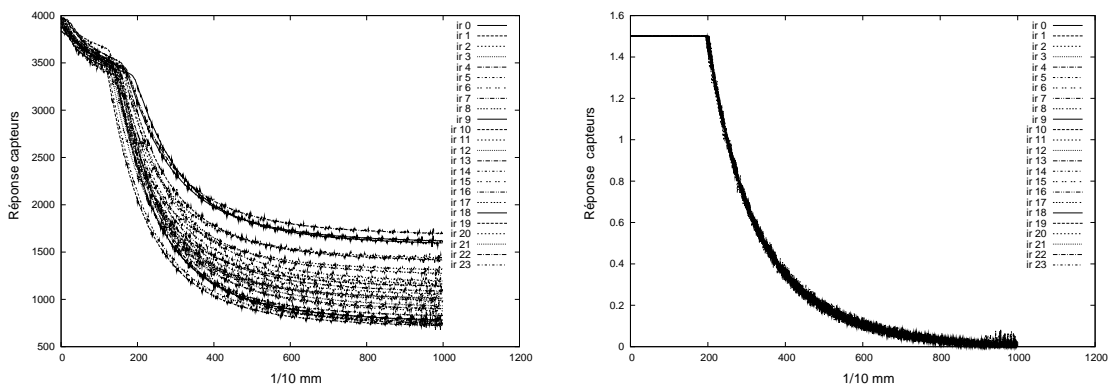


Figure 4.24 Réponse des capteur de proximité avant (gauche) et après calibration (droite)

normalisation de ces capteurs a été développée. La figure 4.24 montre les graphiques des réponses avant et après calibration de ces capteurs.

Dans la partie gauche de la base se trouve une antenne pour lire les tags RFID (figure 4.25). Une électronique spécifique est interfacée avec le dsPIC33 qui gère également le moteur de la chenille gauche et l'IMU. L'IMU est composé de trois gyroscopes et d'un accéléromètre à trois axes. Les gyroscopes sont amplifiés avec un filtre passe bas.

### L'électronique du module d'accrochage

Le module d'accrochage est composé de 4 circuits, 2 servent de connecteurs pour faire passer l'alimentation et les données du bas jusqu'au sommet du module. Le troisième contient l'électronique de mesure des jauges de contrainte. La figure 4.26 présente son principe. Deux jauges sont montées en demi-pont de Wheatstone, l'autre demi-pont est réalisé avec des résistances normales de précision. Un potentiomètre permet d'équilibrer le pont pour la mise à zéro. Le pont est ensuite amplifié par



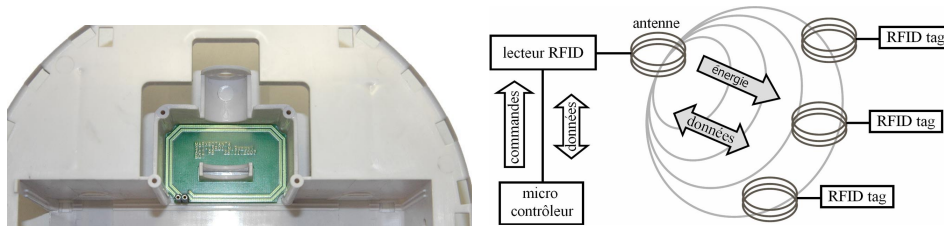


Figure 4.25 L'antenne et le schéma RFID

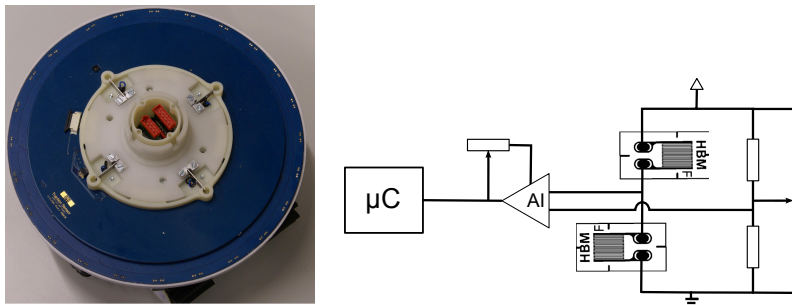


Figure 4.26 Schéma des jauges de contraintes

un amplificateur d'instrumentation. Un potentiomètre permet de régler le gain de celui-ci et donc la sensibilité du système.

Des potentiomètres numériques ont été utilisés, le capteur peut ainsi être calibré automatiquement. Par rapport au s-bot, le circuit d'amplification et les potentiomètres ont été correctement dimensionnés. Une attention particulière a été accordée au routage, afin de diminuer fortement le bruit et d'améliorer l'information utile de ce capteur.

Le quatrième circuit du module est sur la partie tournante dans l'anneau lumineux. L'électronique de l'anneau n'a rien de spécial, elle se compose d'un dsPIC33 gérant deux contrôles moteurs et 12 LED RGB. La figure 4.27 représente l'électronique et les différentes couleurs que peut prendre l'accroche. Les LED RGB sont placées sur le flanc du PCB, les disques servent de guide d'ondes et de diffuseur. Contrairement au s-bot, il s'agit de LED RGB d'un bloc de dernière génération dont le rendement est bien meilleur (20 à 100 fois plus lumineuses suivant la couleur). De plus, elles sont, cette fois-ci contrôlées en courant par un chip dédié. Ce chip peut contrôler jusqu'à 18 LED en courant et également augmenter la tension d'alimentation si nécessaire. Il est interfacé au microcontrôleur par I<sup>2</sup>C.

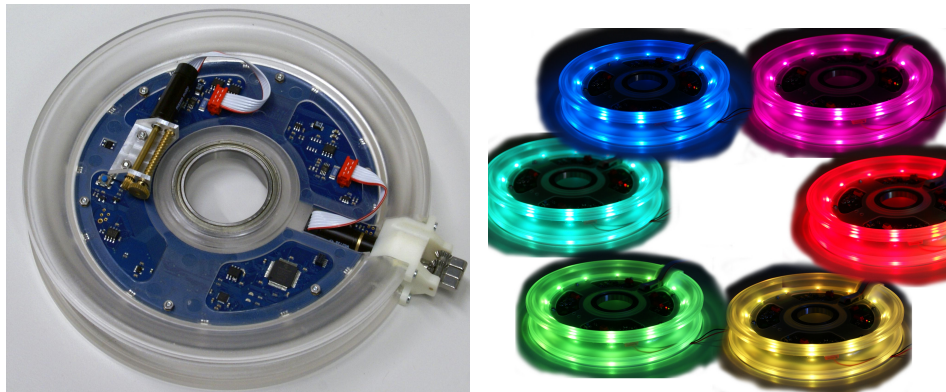


Figure 4.27 Électronique de la couronne lumineuse

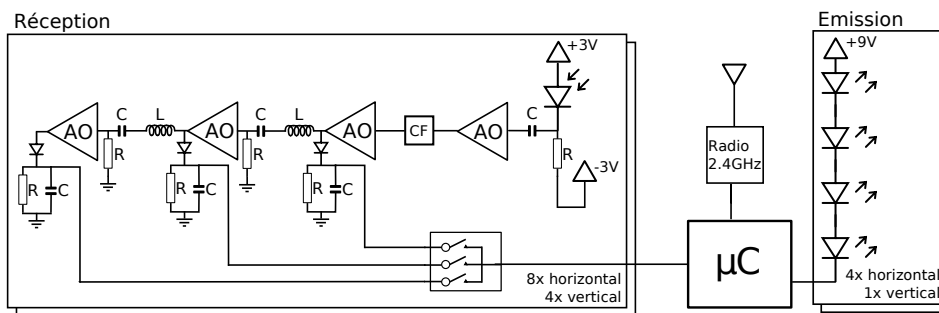


Figure 4.28 Électronique du module distance et relèvement

### L'électronique du module distance et relèvement<sup>9</sup>

Pour mesurer la distance à d'autres marXbots, le module mesure l'intensité du signal infra-rouge émis par ceux-ci. C'est un signal modulé à 455Khz. Pour se synchroniser, il utilise une radio à 2.4GHz. Cette radio sert aussi à s'échanger un identifiant et des données. La figure 4.28 montre le schéma de l'électronique du module. La partie réceptrice est composée de 8 photodiodes horizontales et 4 verticales. Le signal de la photodiode est ensuite amplifié, puis passe par un filtre céramique passe-bande 455kHz. Le signal est alors encore amplifié pour avoir la première valeur mesurée par le microcontrôleur. Deux étages d'amplification sont encore nécessaires pour avoir une bonne distance de mesure. Des filtres passe-bande RLC sont suffisants pour garder un bon signal sur les deux étages d'amplification. Pour avoir une mesure du niveau du signal, celui-ci passe par une diode d'écrêtage. Un multiplexeur analogique est nécessaire

9. L'électronique a été développée avec Philippe Réternaz et James Roberts du LIS sur la base d'une électronique développée par le DISAL par Jim Pugh et Xavier Reamy [68].

pour limiter les entrées analogiques nécessaires sur le microcontrôleur. La partie analogique est alimentée en +3.3V et -3.3V. Un LDO fournit le 3.3V et un step-down inverseur fournit le -3.3V

Pour l'émission, les LED sont mises en série par groupe de 4. Une step-up 9V sert de tension de base aux LED. En tout, 16 LED émettent sur le plan horizontal et 4 verticalement.

Tout le circuit est assez complexe. Il comporte près de 580 composants. Le routage a demandé beaucoup de soins et la partie analogique est très sensible aux perturbations. De plus, les photodiodes ont une dispersion dans leurs caractéristiques et cela a des conséquences non négligeables dues aux trois étages d'amplification. Ainsi, une méthode de calibration a été développée pour linéariser chaque circuit de réception.

Les trois photodiodes qui mesurent les plus grands signaux sont combinées pour connaître la direction de l'émetteur. Les performances permettent de distinguer un robot jusqu'à 6m et de mesurer un angle jusqu'à 5m. Les distances sont mesurées avec une erreur inférieure à 20% et l'angle avec une erreur maximale de  $\pm 10^\circ$ .

### **L'électronique du module scanner<sup>10</sup>**

L'électronique du scanner est répartie sur 2 PCB principaux et de 2 PCB au centre servant de remplacement de câble. Le PCB fixe gère la transmission d'énergie et le PCB rotatif acquiert les données des capteurs et les transmet par infra-rouge au PCB fixe. L'électronique peut être divisée en deux sous-systèmes : la transmission de la puissance et la transmission des données. Sur chaque PCB un microcontrôleur synchronise les opérations.

La batterie du marXbot fournit une tension entre 3.5V et 4.2V. Les capteurs de distance SHARP ont besoin d'une tension d'entrée de 5V. La tension de sortie du transfert inductif doit donc être supérieure à 5V additionnée de la chute de tension dans le redresseur. Un transfert inductif avec une tension nominale avant rectification de 8V et une puissance en pointe de 3.5W a été conçue. La bobine primaire fixe contient deux spires et la bobine secondaire rotative 8 spires, la fréquence de résonance est de 228kHz. Le rendement est de  $\eta = 0.78$  pour la partie inductive avec une tension d'entrée de 3.8V et une tension de sortie de 5.51V. Le rendement global est de  $\eta = 0.69$ , si l'on considère les pertes dans le pont H, le redresseur et le microcontrôleur de la partie fixe. C'est un rendement très satisfaisant étant donné que l'induction sert aussi d'élévateur de tension.

L'induction génère beaucoup de bruit dans les plans d'alimentation. Il en résulte une perturbation dans la transmission de données à cause

---

10. Développée avec Philippe Rétornaz et l'aide de Paolo Germano du LAI pour la partie induction.

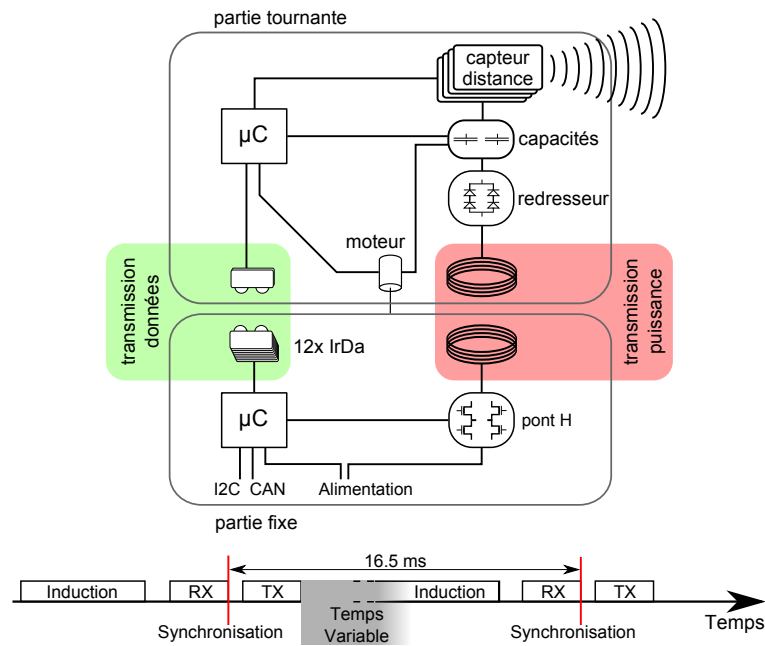


Figure 4.29 Électronique du scanner et chronogramme de synchronisation

de leur fréquence de 115kHz proche de celle de l'induction. Ainsi, pour communiquer d'une manière fiable, l'induction est coupée pendant la transmission de données. L'énergie est stockée pendant la coupure dans de grandes capacités ( $4400\mu F$ ) qui lissent également la sortie du redresseur. La communication IrDA est bi-directionnelle et en simple-duplex avec un code de correction d'erreurs à 16 bits. Le microcontrôleur commande l'émission des émetteurs-récepteurs en parallèle. Leur réception commande des transistors en collecteur ouvert mis en parallèle sur la patte de réception de microcontrôleur. Le PCB rotatif envoie pour chaque acquisition des capteurs (60 Hz) la valeur des 4 capteurs, la position de la tourelle et la tension de sortie du redresseur. Cette dernière permet de réguler la puissance de la bobine primaire pour économiser de l'énergie. Le PCB fixe renvoie un message en retour contenant la vitesse ou la position de consigne pour le moteur.

Lors de la synchronisation de l'induction et de la communication IrDA, un temps variable permet de moduler la puissance transmise et de réguler la tension sur le PCB secondaire (figure 4.29 en bas). La tension cible est de 6.5V, ce qui entraîne une consommation électrique de 500mA à 3.8V pour tout le module.

### L'électronique du CPU principal et la vision<sup>11</sup>

Le CPU principal (figure 4.31) se compose de 5 PCB. Deux sont des caméras, un contient la LED balise, un autre le coeur du processeur i.MX31, c'est-à-dire le processeur, la mémoire et l'alimentation. Le tout est fixé sur le dernier PCB qui est la carte-mère du robot. Le coeur processeur (figure 4.30) a été designé par le laboratoire pour bénéficier d'un processeur de dernière génération. Le choix de faire un petit coeur distinct a été fait pour des soucis de fabrication. En effet, ce genre de processeur miniature demande un routage très fin ainsi qu'un montage très précis, effectué en salle grise. De plus, il faut presque obligatoirement un circuit 10 couches pour pouvoir router la densité des pattes du processeur. Un circuit de trop grande taille serait trop onéreux.

La figure 4.32 présente tout le schéma de la carte mère du robot. Celle-ci relie le coeur processeur aux périphériques et microcontrôleurs du robot. Sans entrer dans les détails voici les principaux composants :

- deux connecteurs pour connecter le coeur processeur.
- un multiplexeur de signaux pour brancher les deux caméras sur l'unique port image de processeur.
- deux ports USB HOST, l'un servant à brancher un dongle Wifi, l'autre restant libre pour d'autres applications.
- un port USB OTG servant de connexion haute vitesse avec le processeur i.MX31.
- deux ports pour des cartes SD, l'un de taille micro avec de la mémoire flash, servant de disque dur, l'autre de taille mini restant libre.
- un système sonore, composé de deux micros et d'un haut-parleur.
- un sélecteur 16 positions pour permettre de choisir sans autre l'interface des options d'utilisations.
- un dsPIC33 servant de translateur UART-CAN pour interfacer le processeur à tous les microcontrôleurs du robot.
- divers connecteurs (I<sup>2</sup>C, CAN, SPI, UART, jack audio) pouvant être utiles pour des futurs développements ou pour le débogage.
- des boutons pour allumer ou éteindre le robot ainsi que pour le réinitialiser.

La caméra frontale peut être placée soit verticalement pour regarder à l'avant, soit horizontalement pour regarder le plafond. Pour faire cela deux connecteurs adéquats sont montés sur le PCB.

---

11. Développée par Valentin Longchamp et Daniel Burnier sur la base d'un travail de diplôme de Samuel Progin





**Figure 4.31** Photo du module CPU principal et vision

Au centre de la carte mère se trouve la tourelle composée du PCB de la LED balise et du PCB pour la caméra omnidirectionnelle.

#### 4.1.3 Les logiciels

Comme pour le s-bot, les logiciels dans le marXbot sont également dispersés dans le processeur central et dans les 10 microcontrôleurs dsPIC33. Cependant, le grand nombre de capteurs et la modularité du robot rend une programmation centralisée non optimum. La programmation centralisée basée sur l'interrogation ("polling") des microcontrôleurs pose des problèmes de temps de réaction comme expliqué dans la figure 4.33. De surcroît, dans le s-bot le bus I<sup>2</sup>C était saturé et nécessitait quelques astuces pour que le robot ne plante pas.

Les microcontrôleurs dsPIC33 ont une puissance de calcul utilisable pour des comportements de bas niveau. En effet, ceux-ci peuvent très bien faire du traitement sur les données des capteurs de proximité et éviter les obstacles très proches, comme par réflexe, sans que le processeur principal n'intervienne. Ainsi, une architecture logicielle basée sur l'envoi d'événements a été développée pour les microcontrôleurs appelée ASEBA<sup>12</sup>. Cette architecture est aussi possible grâce à l'utilisation du bus CAN, un bus multi-mâtres.

12. Développée par Stéphane Magnenat

# CHAPITRE 4. LE MARXBOT

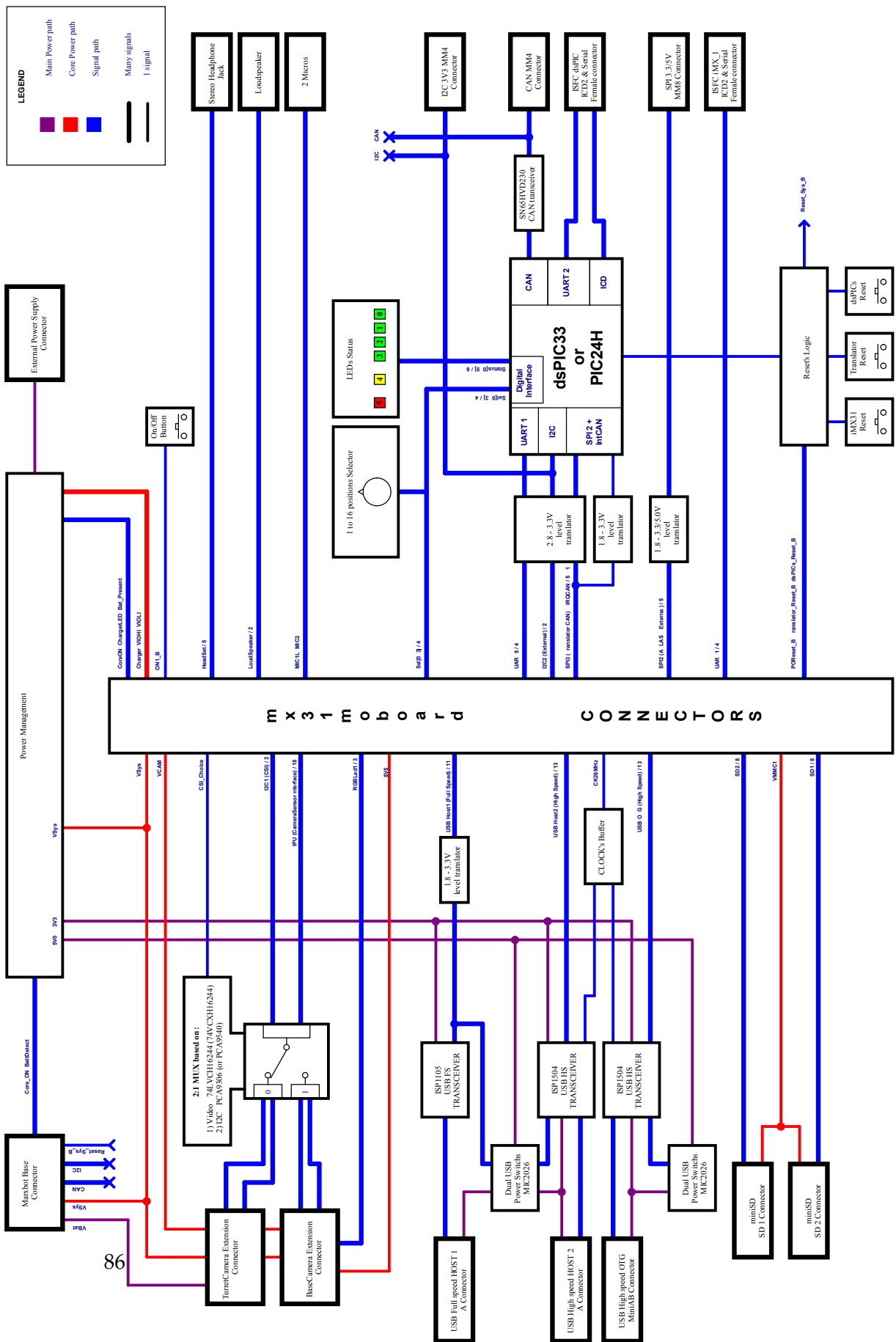
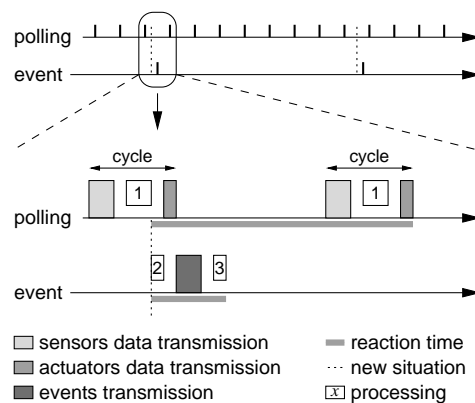


Figure 4.32 Le schéma de la carte mère du marXbot





**Figure 4.33** Différence de séquence temporelle entre une structure par "polling" (comme le s-bot) et une approche par événement comme dans le marXbot avec ASEBA

### Les logiciels bas-niveau <sup>13</sup>

Le code bas niveau des microcontrôleurs se base sur la librairie open-source Molole <sup>14</sup> développée par le laboratoire pour standardiser l'utilisation des dsPIC33. Cette librairie fournit toute l'architecture pour utiliser les périphériques des dsPIC33 d'une façon efficace par interruption. Elle fournit également le support pour divers capteurs et le contrôle de moteur.

La figure 4.34 montre la structure logicielle ASEBA dans le microcontrôleur. Dans chaque dsPIC33, une machine virtuelle exécute un code utilisateur pouvant réagir à des événements internes et externes. Celle-ci interagit avec le contrôle bas niveau. La machine virtuelle peut également appeler des fonctions natives optimisées pouvant effectuer des calculs en peu de temps. Le tout communique avec le reste du robot en envoyant des événements au gré du "bytecode" exécuté par la machine virtuelle. Une explication plus poussée d' ASEBA se trouve dans l'article [48].

Ainsi, grâce à ASEBA, l'utilisateur final peut programmer les comportements des microcontrôleurs sans danger pour l'électronique. En effet, la machine virtuelle donne accès au bas niveau sans interférer dans sa structure fonctionnelle. Pour ceci, l'utilisateur programme les machines virtuelles grâce à un IDE développé spécialement (figure 4.35). Il peut gérer les échanges des événements des microcontrôleurs. Par exemple, l'utilisateur peut programmer de façon à ce que le dsPIC33 des capteurs de proximité ne se manifeste que quand il détecte un obstacle et reste muet la plupart du temps. Sur le s-bot, le processeur principal doit constamment

<sup>13</sup>. Le code des microcontrôleurs a été programmé par Philippe Rétornaz.

<sup>14</sup>. Développée par Philippe Rétornaz et Stéphane Magnenat <http://www.gna.org/projects/molole>

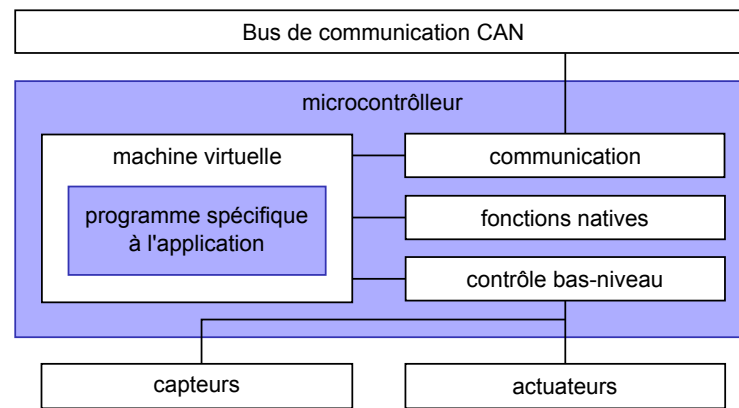


Figure 4.34 La structure logicielle dans les microcontrôleurs

sonder les données aux microcontrôleurs, puis les traiter et éventuellement agir. Sur le marXbot, le processeur et les microcontrôleurs réagissent directement aux événements quand ils interviennent, selon le code choisi dans chaque microcontrôleur. La figure 4.33 décrit le gain de temps que l'on peut avoir par la programmation par événements. Un second avantage est que l'on ne surcharge pas le bus de données CAN, ce qui devenait critique sur le s-bot avec son grand nombre de capteurs. Avec ASEBA on accède à plus de puissance de calcul et à une programmation basée sur les événements, ce qui en fait une structure de programmation plus efficace et flexible.

### Les logiciels haut-niveau<sup>15</sup>

Le marXbot contient, comme le s-bot, un processeur exécutant Linux. Celui-ci est en charge du comportement haut-niveau du robot. Il traite les images de la caméra, fusionne les données des capteurs, réagit si nécessaire aux événements envoyés par les microcontrôleurs et envoie également des événements ASEBA en fonction de ses programmes. Il sert aussi d'interface avec le reste du robot pour l'utilisateur final (figure 4.36). Un programme appelé "Medulla"<sup>16</sup> gère les événements ASEBA et les distribue sur le réseau ou les transmet par D-Bus au système d'exploitation Linux. Les événements ASEBA ont été développés pour être transmis sur différents canaux comme le TCP, CAN, UART ou Bluetooth<sup>®</sup>.

L'acquisition des images des caméras se fait par un port spécifique sur le processeur i.MX31. Les images sont directement transférées en mémoire

15. Linux a été porté par Valentin Longchamp

16. Développé par Stéphane Magnenat

## 4.1. DESIGN DU MARXBOT

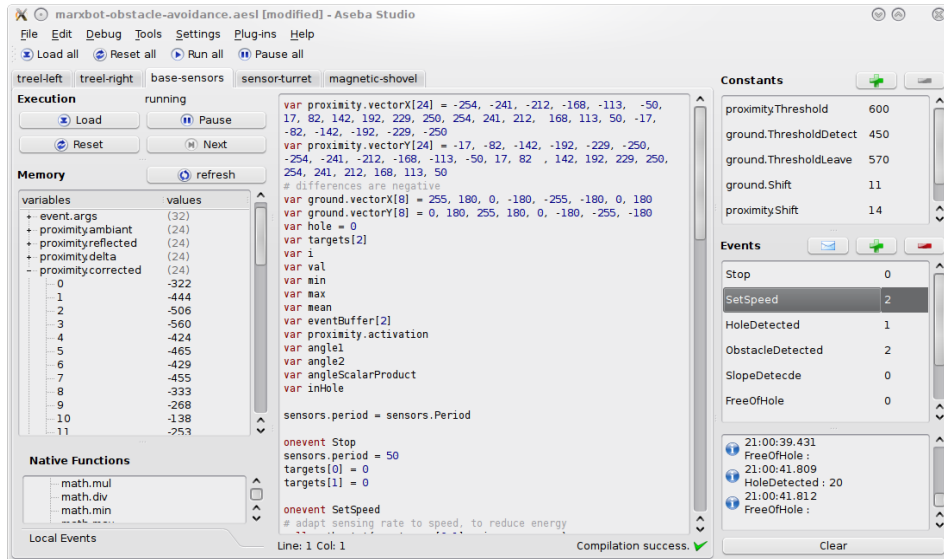


Figure 4.35 ASEBA studio

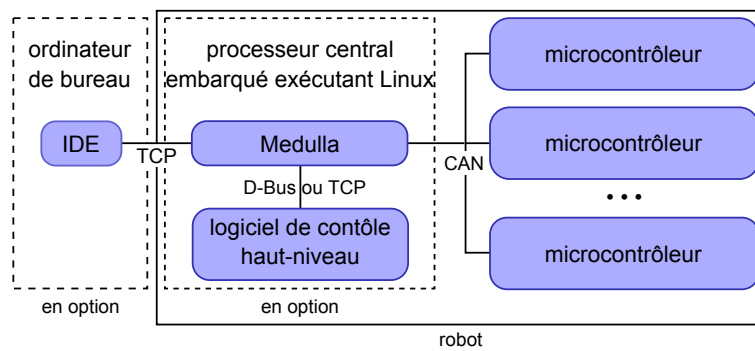


Figure 4.36 Le schéma du coeur processeur marxbot



**Figure 4.37** Les pièces du système d'accrochage fabriquées en MIM

par DMA. Ainsi, le processeur peut directement traiter les images sans devoir les acquérir et les décompresser contrairement au s-bot.

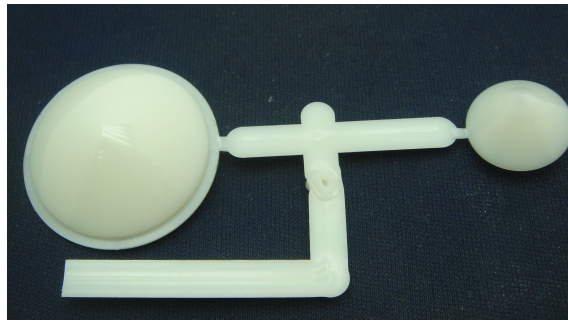
#### 4.1.4 Les techniques de production

Le marXbot a été conçu pour faire de la robotique collective, mais aussi pour être une plate-forme de recherche polyvalente. Le robot est modulaire, tout en se reposant sur des éléments de base fiables. Ainsi, les modules du marXbot ont été développés et fabriqués en essayant de garantir une bonne fiabilité tout en gardant un coût de production moyen. Les techniques de production ont été adaptées pour une première série de 70 pièces.

Ainsi, les éléments plastiques de la base sont tous des pièces injectées, ce qui assure une certaine homogénéité. Par conséquent, au montage, il y a moins d'ajustements à faire que pour le s-bot. La technique d'injection permet aussi de faire des pièces de formes très complexes, limitant au final le nombre total de pièces (figure 4.6). Comme pour l'e-puck, l'injection est déjà rentable pour un nombre faible de robots (les pièces ont été injectées à 100 exemplaires).

Pour limiter le temps de montage et les pièces, la plupart des moteurs du robot sont des ensembles moteurs-réducteurs commerciaux. Ceci permet d'avoir une grande fiabilité, cependant, l'intégration mécanique est plus difficile et le prix légèrement plus élevé. Deux moteurs n'ont pas de réducteur intégré dans le robot (scanner et rotation du gripper). Ils entraînent tous deux une vis sans fin s'engrenant sur un roue à grand diamètre. La réduction est ainsi grande et un réducteur additionnel n'est pas nécessaire.

Pour le système d'accrochage (figure 4.37), nous avons besoin de pièces à grande résistance, mais quasiment impossible à usiner par des techniques conventionnelles. En effet, il n'est pas possible d'usiner un engrenage conique muni de l'appendice d'accrochage. Un essai de fabrication de la



**Figure 4.38** Les miroirs de la caméra (gauche) et de la LED balise (droite) avant dépôt de peinture réfléchissante

pièce en plastique a d'abord été fait, mais la résistance des dents n'était pas suffisante. Pour finir, la technique de moulage par injection de métal (MIM) a été utilisée. Cette technique permet d'injecter, comme pour le plastique, un mélange métal-plastique dans un moule. Puis, la pièce est frittée, ce qui lui fait prendre son aspect final et sa dureté. La précision de la technique peut aller jusqu'à 0.01mm.

Le miroir du s-bot est une simple demi-sphère de verre où une fine couche d'aluminium a été déposée en phase vapeur. Pour le marXbot, un miroir de plus grande taille a été souhaité avec un profil bien défini permettant d'avoir un meilleur champ de vision. Une fois le profil défini, un premier miroir a été réalisé en laiton et poli, puis une couche de chrome a été déposée. Le profil a pu être validé, mais la technique de production n'était pas adéquate pour une petite série de 70 pièces. De ce fait, les miroirs ont été fabriqués par injection plastique (figure 4.38). Les techniques actuelles permettent d'avoir des pièces polies. Une peinture a par la suite été déposée pour leur fonction réfléchissante.

### 4.2 Caractéristiques du marXbot

#### Les degrés de liberté (ddl)

Le marXbot a 4 degrés de liberté, 2 chenilles, un anneau d'attache rotatif et un système d'accrochage. Un 5<sup>ème</sup> degré est utilisé pour la rotation du scanner, mais ne peut être comptabilisé car il est intrinsèque. Le système d'accrochage relatif au module est spécifiquement un besoin pour les comportements collectifs permettant l'auto-assemblage.

**Les informations sensorielles extrinsèques**

Voici un tableau résumant les débits de données de chaque type de capteurs :

	Types	Données brutes	Signal utile	Fréquence	Data/s
microcontrôleurs	4×IR au sol	4×12bits	8bits	100Hz	3.2kbit/s
	8×IR au sol	8×12bits	10bits	100Hz	8kbit/s
	24×IR	24×12bits	10bits	100Hz	24kbit/s
	proximité				
	24×IR	24×12bits	10bits	100Hz	24kbit/s
	ambient				
	3D accéléromètre	3×12bits	10bits	200Hz	6kbit/s
	3×gyroscopes	3×15bits	14bits	200Hz	8.4kbit/s
	2×force	2×12bits	12bits	1kHz	24kbit/s
	D&R	12×3×10bits	12bits	250Hz	36kbit/s
scanner	4×12bits	10bits	60Hz	2.4kbit/s	
	Total				136kb/s
i.MX31	2×micros	2×13bits	13bits	16kHz	416kbit/s
	2×caméras couleur	1600×1200p 8bits bayer	8bits	20Hz	307Mbit/s par caméra
	Total				307.5Mbit/s

Les 10 microcontrôleurs dsPIC33 tournent à 40 MIPS et sont des processeurs 16bits. Ils ont également un coeur DSP et un bus DMA. Les microcontrôleurs totalisent ainsi 400 MIPS à 16bits, près de 13 fois plus que les microcontrôleurs du s-bot. Le processeur 32 bits i.MX31 est cadencé à 533 Mhz, et 740 MIPS @532–665 MHz<sup>17</sup>

**La communication**

Le marXbot a différents moyens de communication. Premièrement, une connexion Wifi type 802.11n bi-bande à 2.4Ghz et 5Ghz. Comme pour le s-bot, le Wifi est utilisé comme interface utilisateur. Malgré la possibilité technique, le réseau Wifi n'est pas utilisé pour communiquer entre les robots. Cette connexion peut aussi permettre de faire de gros calculs sur un ordinateur de bureau et de retransmettre le résultat au robot. Cette solution est surtout utilisée lors du développement, par la suite le code peut être optimisé pour le processeur embarqué i.MX31. Deuxièmement, le

17. Source : [http://en.wikipedia.org/wiki/ARM\\_architecture](http://en.wikipedia.org/wiki/ARM_architecture)

marXbot est muni d'un haut-parleur et de deux microphones, qui peuvent être utilisés comme moyen de communication. Un prototype d'un module sonore muni d'un réseau de 16 microphones a été testé. Avec un si grand nombre de microphones, les positions de plusieurs émetteurs peuvent même être déterminées.

Le troisième moyen est la caméra et les LED lumineuses. Comme pour le s-bot, cela permet de communiquer et de déterminer la position des robots avoisinants. Le marXbot a un avantage, avec sa LED balise de 3.5W et son miroir optimisé, car il peut être distingué à plus d'un mètre (figure 4.15), contrairement au s-bot qui l'était jusqu'à 60cm. Le dernier moyen de communication est le module "distance et relèvement". Ce module a été spécialement développé pour la communication collective. Les collègues du projet responsables du contrôle ont demandé un moyen de communication qui soit global et qu'on puisse limiter la distance dynamiquement. La communication doit également permettre d'identifier l'émetteur du message. Ainsi le module "distance et relèvement" répond indirectement à ces demandes. La communication radio 2.4Ghz associée est globale et peut aller jusqu'à 30 m, le système infra-rouge permet de mesurer la distance et la direction de l'émetteur jusqu'à 5 m. On peut répondre de ce fait aux exigences des personnes responsables du contrôle.

### Le prix

Le coût de fabrication totale d'un marXbot s'élève à 3610 frs. Il faut ajouter 1 semaine-homme de fabrication et de montage des pièces spécifiques réalisées par nos moyens. Une journée de test et calibration est aussi nécessaire.

Le tableau suivant résume le prix par modules du marXbot.

Modules	Mécanique	Moteurs	Électronique	Total
Base	105.-	293.-	527.-	925.-
M. d'accrochage	264.-	267.-	353.-	884.-
Distance et relèvement	20.-		300.-	320.-
Scanner	30.-	69.-	480.-	578.-
CPU et vision	202.-		700.-	902.-
<b>Totaux</b>	<b>621.-</b>	<b>629.-</b>	<b>2360.-</b>	<b>3610.-Frs</b>

Tableau 4.2 Coûts du marXbot

L'électronique représente 65% du prix total du robot : c'est le prix à payer pour avoir une grande capacité sensorielle. Le reste du prix est partagé entre les moteurs et la mécanique. Les 5 moteurs sont des ensembles moteurs réducteurs de haute qualité et donc assez onéreux. Toutes les

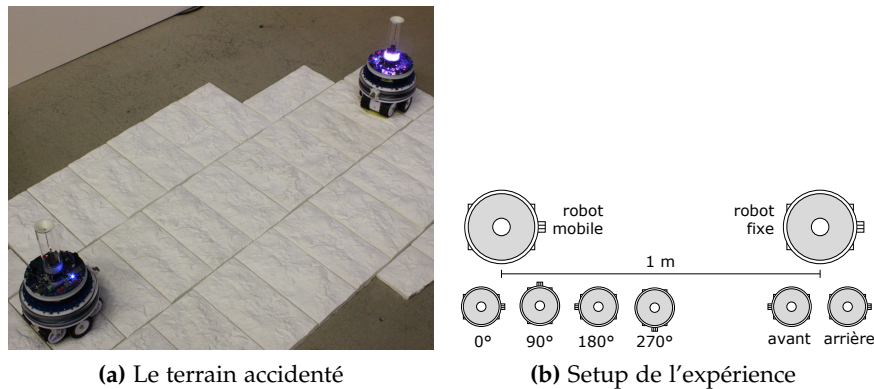


Figure 4.39 Expérience d'auto-assemblage

pièces mécaniques ne représentent que 17% du coût. Ce prix est encore assez élevé car le nombre de pièces ne permet pas encore de réduire les frais fixes dus à la fabrication des moules pour les pièces injectées. Fabriquées à plus grande échelle, les pièces mécaniques ne représenteraient que 10% du prix de revient comme pour l'e-puck.

Si on regarde d'un point de vue modulaire, on remarque que les deux modules spécifiques à la robotique collective, le module d'accrochage et le module de "distance et relèvement", représentent 1/3 du coût total. Ainsi, malgré que le marXbot de base est déjà un robot très complet, le coût pour en faire un robot collectif pouvant interagir physiquement avec ces coéquipiers a un prix non négligeable.

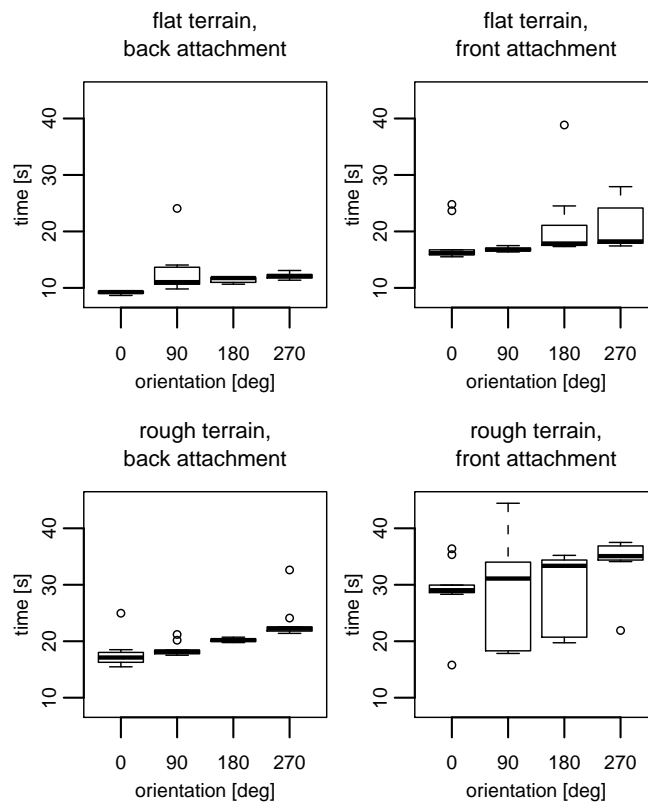
#### 4.2.1 La capacité d'auto-assemblage

Comme le s-bot, le marXbot a la capacité de s'auto-assembler avec ses pairs. Il peut aussi s'assembler avec des handbots. Nous allons présenter ici les résultats du papier "The MarXbot, a Miniature Mobile Robot Opening new Perspectives for the Collective-robotic Research" [7, Chapitre VI]<sup>18</sup>.

Le marXbot devait s'assembler avec un autre marXbot fixe distant de 1m qui avait sa LED balise allumée en blanc. L'expérience a été faite sur un terrain plat et un terrain accidenté (figure 4.39a) qui est identique au terrain très accidenté utilisé pour tester le s-bot. Le marXbot mobile partait avec 4 orientations différentes et le marXbot fixe était présenté de face comme de dos (figure 4.39b). Chaque combinaison a été testée 10 fois. Pour toutes les tentatives le marXbot a réussi à s'accrocher à son égal. Ce qui nous mène à 100% de réussite dans tous les cas de figure. Ainsi le

18. Le code de l'expérience a été fait avec Stéphane Magnenat et les résultats rassemblés avec la collaboration de Valentin Longchamp





**Figure 4.40** Les résultats d’auto-assemblage mesurés en temps nécessaire en fonction de l’orientation et du type de terrain. La distance d’approche est fixée à 1 m.

marXbot réussit parfaitement là où le s-bot avait quelques difficultés en terrain accidenté. Ceci est dû à différents points :

- La morphologie du système d’accrochage et de l’anneau d’accrochage permettent une connexion même si l’alignement et le positionnement ne sont pas précis. L’approche est plus simple car il suffit d’avancer jusqu’à toucher le cercle et s’accrocher. Avec la pince du s-bot, il fallait être plus précis sur la distance de fermeture.
- L’assemblage peut se faire sur plus de 320° de la périphérie du marXbot (sur le s-bot que 240°).
- Les capteurs et les informations intrinsèques des moteurs permettent de déterminer le succès de l’assemblage, sinon une marche arrière et une nouvelle approche sont effectuées.
- Les chenilles et les roues fournissent une mobilité permettant une approche et un positionnement adéquat même en terrain accidenté.

A l'approche d'un robot à l'arrière d'une distance de 50 cm, le S-bot a pris en moyenne 28.7 s pour s'attacher sur le terrain plat, et 71.3 sur le terrain très accidenté(3.2). Le marXbot est beaucoup plus rapide, à l'approche d'une distance de 1 m et ne prend en moyenne que 11.4s pour s'assembler sur le terrain plat et 19.9 sur le terrain accidenté. Dans la plus difficile des conditions, l'approche du robot fixe à partir de l'avant, le marXbot prend 20.0 s dans le terrain plat et 30.8 s sur l'accidenté. La figure 4.40 montre le détail des mesures pour les différentes orientations de départ.

Le marXbot est donc plus efficace que le s-bot et ceci pour différents points :

- Le robot est plus rapide, plus grand et donc le terrain accidenté est moins contraignant pour sa taille.
- Le traitement de la caméra permettant l'approche est plus efficace, car la caméra a une meilleure résolution ; le traitement est effectué sur un processeur plus puissant et la décompression jpeg n'est plus nécessaire.
- Le système ASEBA permet de se déplacer plus vite, sans collision et l'approche finale est directement traitée par la base et l'échange d'événements entre les microcontrôleurs.

### 4.2.2 SLAM<sup>19</sup>

En robotique mobile, la localisation est un aspect très important pour de nombreux comportements. Elle peut se faire de différentes manières en utilisant des moyens externes ou pour les robots d'extérieur avec un GPS. Une autre approche est de se localiser simultanément sur une carte tout en la construisant (SLAM). Pour se faire, en robotique, on utilise très souvent des scanners laser. Cependant, pour un robot de la taille d'un marXbot les systèmes laser sont chers, encombrants et consomment beaucoup d'énergie. C'est pourquoi le module scanner du marXbot a été développé.

Nous allons présenter succinctement les résultats du papier "Affordable SLAM through the Co-Design of Hardware, Software, and Methodology" [46]. Dans cet article, nous présentons une solution au problème de SLAM basée sur la co-conception du scanner rotatif, d'un algorithme logiciel SLAM léger et d'une méthodologie d'optimisation. L'algorithme SLAM est une adaptation de Fast SLAM 2.0 qui fonctionne en temps réel sur un robot miniature.

---

19. Le codage et l'article en résultant sont pour l'essentiel l'oeuvre de Stéphane Magnenat et Valentin Longchamp.

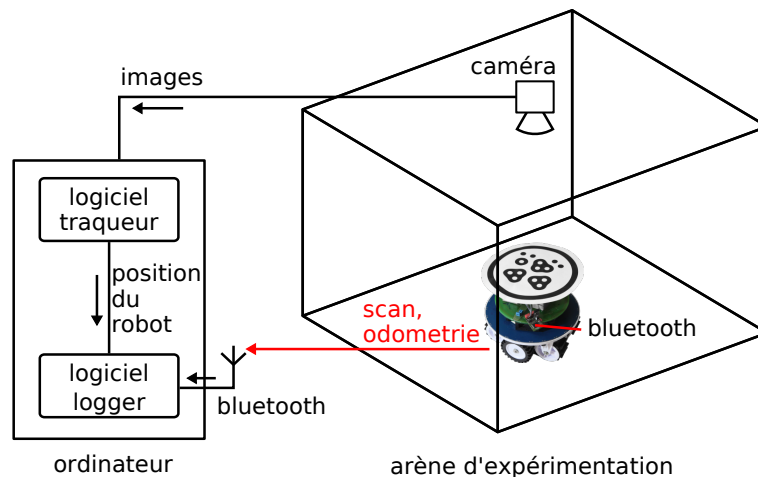


Figure 4.41 .

Grâce à un traqueur (figure 4.41), différents parcours de marXbot ont été enregistrés. Ensuite grâce à un algorithme d'optimisation globale évolutif, les paramètres pour l'algorithme Fast SLAM 2.0 ont pu être optimisés. Les paramètres ont été optimisés pour différents pourcentages de la puissance de calcul de l'i.MX31. Le marXbot utilise les informations venant de l'odométrie des chenilles, d'un gyroscope et des balayages du scanner pour effectuer la localisation.

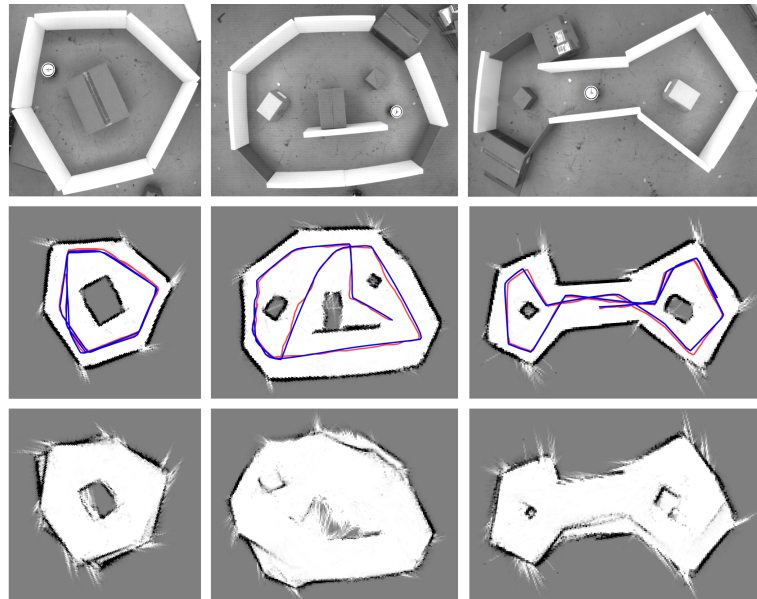
La figure 4.42 montre différentes cartes de SLAM effectuées dans différents environnements. L'erreur de positionnement est très faible (différence entre la trace rouge et bleue), ainsi, le marXbot a la capacité de construire une carte et de se localiser dans celle-ci quand on le place dans un environnement inconnu. L'algorithme de SLAM est embarqué dans le robot et n'a pas besoin de puissance de calcul additionnelle. Cette capacité est novatrice pour un robot de la taille du marXbot.

### 4.2.3 La station d'échange de la batterie<sup>20</sup>

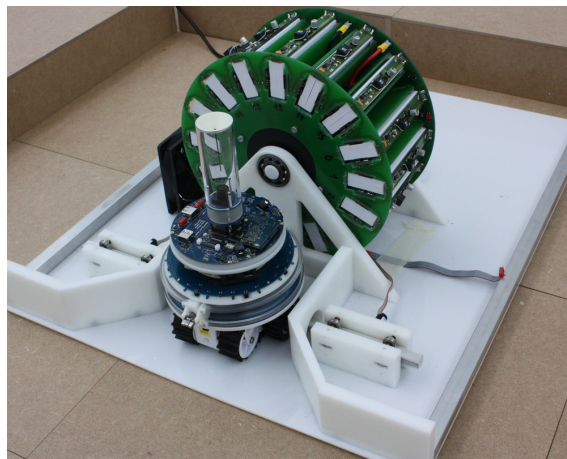
Le marXbot a la fonction d'échange de batterie à chaud. En outre, une station d'échange de batterie (figure 4.43) a été conçue en parallèle à la base. Cette station accueille le marXbot dans une ouverture adaptée à une approche autonome. Deux vérins centrent le robot et le maintiennent en place durant l'échange. Un troisième vérin munit d'un embout approprié vient prendre la batterie déchargée et la remplace par une pleine. Une vidéo<sup>21</sup> montre le processus complet et en détails.

20. Développée avec Tarek Baaboura, François Rey et Philippe Rétornaz.

21. marXbot automatically exchange its battery : <http://www.youtube.com/watch?v=zEeLjeJs1H4>



**Figure 4.42** Les différents environnements expérimentaux (en haut), et la carte construite par le SLAM (au centre), en utilisant un budget de 65'000 rayons. La trajectoire SLAM est en rouge clair alors que la trajectoire réelle (tracker) est en bleu foncé. En bas, la carte de reconstruction sans la phase B de l'algorithme [46, Section V]



**Figure 4.43**

La station peut contenir 15 batteries et les recharger. Ainsi des expériences en continu peuvent être envisagées, où les robots gèrent leur énergie de façon autonome.

#### 4.3 Leçons apprises, limites et améliorations

Le marXbot est probablement un des robots les plus complexes et aboutis de cette taille. La modularité est un avantage indéniable, bien qu'elle limite ou impose des choix mécaniques et électroniques afin de garantir la compatibilité.

Le marXbot a encore été peu utilisé et toutes ses limites ne sont pas encore connues. Toutefois, certains défauts peuvent encore être facilement corrigés.

Un des problèmes connus vient des packs batteries. Le fait d'avoir mis des éléments en parallèle pose quelques problèmes de charge. Ainsi, on ne peut pas charger le pack à 100%, on est obligé de couper avant à 85%-90% pour garantir son intégrité. Néanmoins, au vu de l'autonomie de 4h-8h et la possibilité de changer automatiquement la batterie, ce n'est pas trop contraignant. Cependant, il reste un doute sur la durée de vie des packs. Le remplacement à chaud des batterie ne fonctionne par moments pas sur certains robots, un petit ajustement de l'électronique de contrôle est nécessaire.

Un autre problème vient du système d'accrochage, les trois têtes ne garantissent pas toujours une connection très rigide surtout en torsion. Le cercle lumineux ne tourne pas en continu et est limité à cause d'un câble plat de  $+360^\circ$  à  $-220^\circ$ . Il n'y a, par contre, pas de butée mécanique et si le robot est mal initialisé ou qu'un dysfonctionnement surgit, il se peut qu'on arrache le câble, ce qui demande un temps de réparation de 10-20 minutes.

Le module D&R et le module scanner interfèrent entre eux. Malgré la différence de modulations des deux capteurs, il arrive parfois que les mesures soient faussées. Ces deux modules injectent aussi du bruit sur l'alimentation ce qui peut aussi expliquer ce phénomène.

La scanner est aussi limité et n'est pas comparable à la précision qu'on peut avoir avec les scanners laser traditionnels. La fréquence de rafraîchissement devrait être plus rapide et le fait qu'il y a des ambiguïtés entre les capteurs SHARP de courte et longue distance complique son utilisation. Sa portée limitée à 150cm limite aussi la précision du SLAM dans des espaces trop grands. Ainsi, on peut très bien faire du SLAM pour se positionner dans une carte mais difficilement avoir la précision nécessaire pour faire de la construction, par exemple.

Les gyroscopes utilisés dans la base ne sont pas les meilleurs du marché. Ils sont plus chers et moins précis que ceux de l'handbot. De plus ils dérivent plus rapidement. Cependant, la base a été conçue avant que ces nouveaux gyroscopes existent.

Précisons qu'il s'agit d'un problème récurant en électronique. Un design est à peine fini et la production lancée que certains composants sont déjà obsolètes. Cependant, par expérience je dirai que le proverbe "le mieux est l'ennemi du bien" prend tout son sens en robotique. Lors d'un design d'un tel robot, il faut savoir arrêter le développement pour lancer la production. Le fait d'avoir un robot modulaire peut aussi permettre de corriger plus facilement certains défauts et garantira sans doute son évolution au gré des progrès.

La conception du marXbot et sa production ont aussi demandé un effort de gestion d'une équipe de plusieurs ingénieurs et techniciens. Il m'est arrivé parfois de passer la majorité de mon temps à gérer l'équipe et de ne plus pouvoir avancer dans mes propres tâches.

### 4.4 Conclusion

Grâce à l'expérience acquise avec le s-bot et le e-puck, le marXbot est un robot de recherche beaucoup plus abouti. Son premier intérêt est d'être modulaire, permettant d'y ajouter le matériel spécifique à de nombreuses applications différentes. Néanmoins, le robot dispose d'une base solide et d'une partie supérieure avec une grande puissance de calcul et un système de vision évolué. La gestion d'énergie améliorée couplée à une batterie de grande capacité permet au marXbot d'avoir une autonomie de plus de 8h. Cette autonomie est bien supérieure au standard pour un robot de cette taille. En outre, la batterie peut être échangée automatiquement.

La structure électronique et l'architecture logiciel ASEBA permettent de facilement ajouter les modules nécessaires. ASEBA permet également d'accéder à la puissance de calcul des microcontrôleurs et de changer la façon de programmer, passant d'une architecture logicielle séquentielle à une architecture événementielle distribuée.

Le système d'accrochage simplifié et la mise à niveau des technologies embarquées permettent au marXbot d'être plus efficace que le s-bot pour s'auto-assembler. Le module scanner permet également des expériences où le robot découvre et se positionne dans son environnement de façon autonome, ce que peu de robots de cette taille peuvent faire.

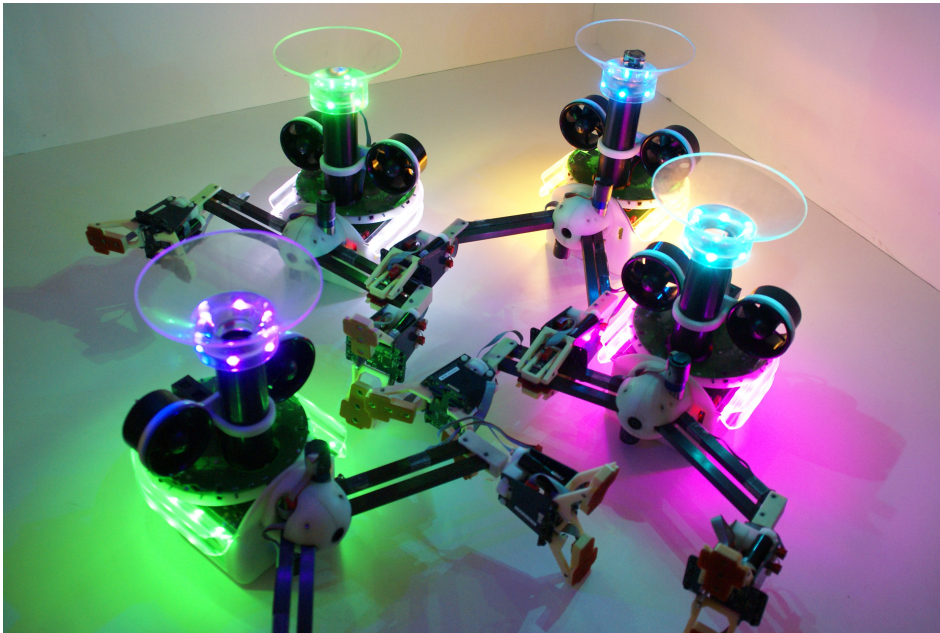
Le design du marXbot permet d'utiliser des techniques de production de masse, limitant le temps de main d'oeuvre et permettant d'en construire une grande quantité. Cependant, sa grande complexité, la richesse de ses

capteurs et la grande puissance de calcul impliquent une électronique onéreuse.





## 5 Le Handbot



Le handbot est un robot grimpeur, pouvant manipuler des objets. Pour sa tâche, le handbot est muni d'un lanceur lui permettant de jeter un système d'accrochage magnétique au plafond accroché à une ficelle. Puis s'aidant de la ficelle pour soutenir son poids, il utilise ses deux pinces pour grimper le long d'une étagère. Il est aussi muni de deux ventilateurs qui lui permettent de s'orienter s'il est uniquement suspendu. Le handbot est muni du même anneau d'accrochage pour que le marXbot puisse s'auto-assembler.

J'ai conçu la mécanique du handbot avec l'aide de notre prototypiste Tarek Baaboura. L'électronique est en grande partie inspirée par celle du marXbot et des standards du laboratoire. Une aide des collègues experts en système magnétique a été fournie pour développer le système d'accrochage. Ma connaissance du robot a fortement aidé au développement des comportements de grimpe.

Dans ce chapitre, nous présenterons le design du handbot ainsi qu'une expérience de prise d'un livre sur une étagère.

Le robot handbot a été conçu dans le cadre du projet Européen Swarmanoid, projet en partenariat entre l'École Polytechnique Fédérale de Lausanne (Suisse), l'Université Libre de Bruxelles (Belgique), l'Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (Suisse) et le Consiglio Nazionale delle Ricerche de Rome (Italie).

### 5.1 Design du handbot

#### Le cahier des charges

Dans le projet "Swarmanoid" un essaim composé de robots de trois types différents est utilisé pour remplacer un robot humanoïde. Premièrement, il y a les footbots (marXbots), ou littéralement les robots pieds qui déplacent les robots mains, les handbots. Les handbots sont donc les manipulateurs, mais doivent également pouvoir grimper pour prendre des objets dans un environnement bureautique. Ils fournissent ainsi au groupe la possibilité d'intervenir dans la dimension verticale. Le troisième type de robot, les eyebots, sont des robots volants munis d'une caméra. Ils doivent fournir au groupe une vue d'ensemble et guider les footbots dans un environnement inconnu à l'avance. Les eyebots ont été développés par nos partenaires du laboratoire des systèmes intelligents (LIS) de l'EPFL et ne sont pas décrits ici.

Le cahier des charges du handbot était très vague au début du projet. Les seules contraintes étaient de pouvoir manipuler des objets et de pouvoir grimper pour les prendre. Il devait également être compatible avec le footbot au niveau auto-assemblage.

#### État de l'art

Un panorama des différents robots grimpeurs existants a été faite cette année par Baeksuk et al [17]. Ils résument dans leur article les différentes techniques utilisées pour l'escalade, et les applications cibles. Les applications potentielles sont répertoriées comme suit : inspection, nettoyage des murs, entretien, peinture, diagnostic, soudage, reconnaissance, consolidation de parois rocheuses, transport de matière et manipulation d'objets. Dans la manipulation d'objets, nous sommes cités à titre d'exemple seulement. Aucune des autres applications n'implique la manipulation d'objets.

Dans le domaine de la soudure, les robots sont guidés par rail [43] ou sur la base adhérence magnétique [67]. Dans cette dernière catégorie, nous trouvons les mises en oeuvre les plus intéressantes. Par exemple, Bridge et al. [10] a développé un ensemble de quatre robots avec une adhérence magnétique pour des tâches de soudage sur les navires. Leurs grands robots, entre 35 et 110 kg, collaborent pour effectuer l'opération

de soudage et d'inspection sur les surfaces verticales. Cela est possible parce que la surface est plane et possède des propriétés ferromagnétiques. Leur choix de la distribution de l'outillage sur plusieurs robots rend leur approche similaire au projet Swarmanoid.

Sur les surfaces ferromagnétiques, plusieurs autres systèmes démontrent une mobilité très intéressante. Un exemple est le robot à quatre pattes de Peters et al. [66]. Ce robot peut passer des obstacles sur les surfaces verticales grâce à un système d'adhésion magnétique au bout des pattes, mais il est extrêmement complexe dans sa structure. A l'opposé, le Cy-mag3D est un robot extrêmement simple qui est capable de se déplacer dans les structures ferromagnétiques avec des obstacles [70].

Lors de l'examen des surfaces non ferromagnétiques, la plupart des applications utilisent comme méthode adhérence l'aspiration. Dans cette catégorie on peut trouver un travail intéressant de Kim et al. [42]. Leur robot est basé sur un ensemble de sous-modules coopérant pour passer les obstacles sur des parois verticales à l'aide d'aspiration. Plusieurs autres chercheurs travaillent sur l'amélioration des méthodes d'aspiration [6,69]. Une combinaison intéressante d'un mécanisme de déplacement et d'aspiration a été proposée par Hayakawa et al. [37]. Leur système d'aspiration basé sur un ventilateur est couplé à un mécanisme qui génère une onde se propageant sur le fond du robot, permettant d'avancer sans perdre l'adhérence. Cette solution leur permet de se déplacer sur des surfaces verticales de types très différents, mais le prototype a une alimentation externe et n'a pas d'électronique embarquée.

Alon Capoue et al. [13] ont développé un robot appelé Spiderbot capable de se fixer à un plafond en utilisant plusieurs fils. Le robot peut tirer de nouveaux points d'attache et récupérer les câbles d'une manière similaire au handbot.

### **L'approche**

L'ascension des surfaces verticales est un problème difficile : il faut un robot léger et résistant avec des actionneurs puissants, ce qui entraîne une conception mécatronique sensible. Pour cette raison, peu de systèmes permettent de combiner les capacités d'escalade et de manipulation.

Différentes solutions ont été explorées notamment dans des projets de semestre par des étudiants (figure 5.1). Leurs recherches se sont portées sur les systèmes d'accrochage et la morphologie du robot. Pour l'adhésion, un système muni de ventouse était le plus prometteur. La morphologie, quant à elle, s'est toujours rapprochée d'un bras à 5 degrés de liberté. Ces recherches ont fait ressortir deux grands problèmes : la force d'adhésion doit être très grande et surtout pouvoir soutenir le moment de la masse du

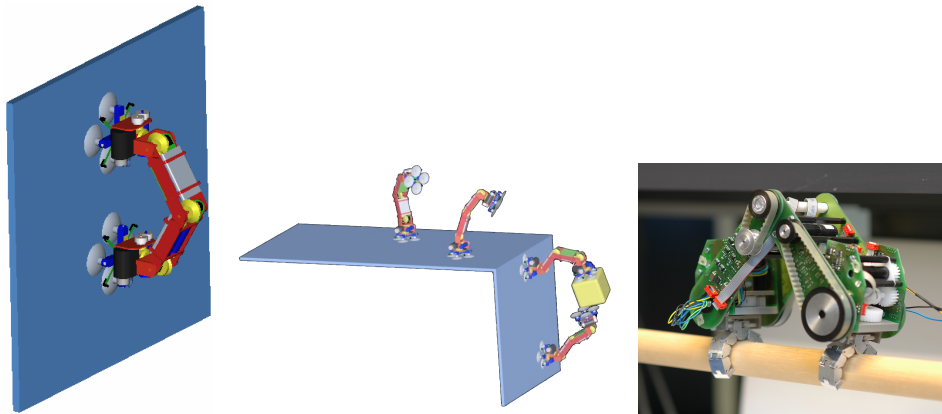


Figure 5.1 Divers concepts imaginés pour le handbot

robot lorsqu'il est fixé au mur. Les couples des moteurs dans les poignets doivent également soutenir ce couple.

De plus, leur approche ne facilite pas la manipulation des objets. On est soit obligé de rajouter un troisième bras articulé ce qui complique beaucoup la morphologie, soit obligé de collaborer en se passant l'objet. Cette dernière opportunité semble d'un point de vue du contrôle irréalisable avec les capacités sensorielles actuelles.

Pour grimper, il faut avant tout compenser la force de pesanteur verticale. Si on sépare cet aspect de celui de s'accrocher, le moyen le plus simple est d'avoir une corde ou une ficelle et de l'enrouler pour s'élever. Cependant la ficelle ne peut pas fournir un positionnement précis. L'utilisation de bras et de pinces pour s'accrocher aux structures verticales trouvées dans l'environnement est complémentaire et permet de garder un point de référence. De plus, ces mêmes bras peuvent être utilisés pour manipuler les objets. Ainsi, la force nécessaire dans ces bras est moins grande car la force de pesanteur est soutenue par la ficelle.

Pour que le robot soit autonome, il doit pouvoir lancer la ficelle pour se fixer à l'endroit souhaité. Il doit également pouvoir la récupérer et recharger son système.

On a ainsi muni le handbot d'un lanceur pouvant propulser un système d'accrochage magnétique au plafond. Ce système peut en temps voulu se décrocher et ainsi être récupéré en enroulant rapidement la ficelle. Bien que cette solution ne marche que dans des environnements avec un plafond ferro-magnétique (beaucoup de faux plafonds le sont), le système d'accrochage peut être adapté.

Le handbot n'a pas besoin de se déplacer au sol, il peut se faire transporter. C'est pourquoi il est muni du même anneau d'accrochage que le marXbot.

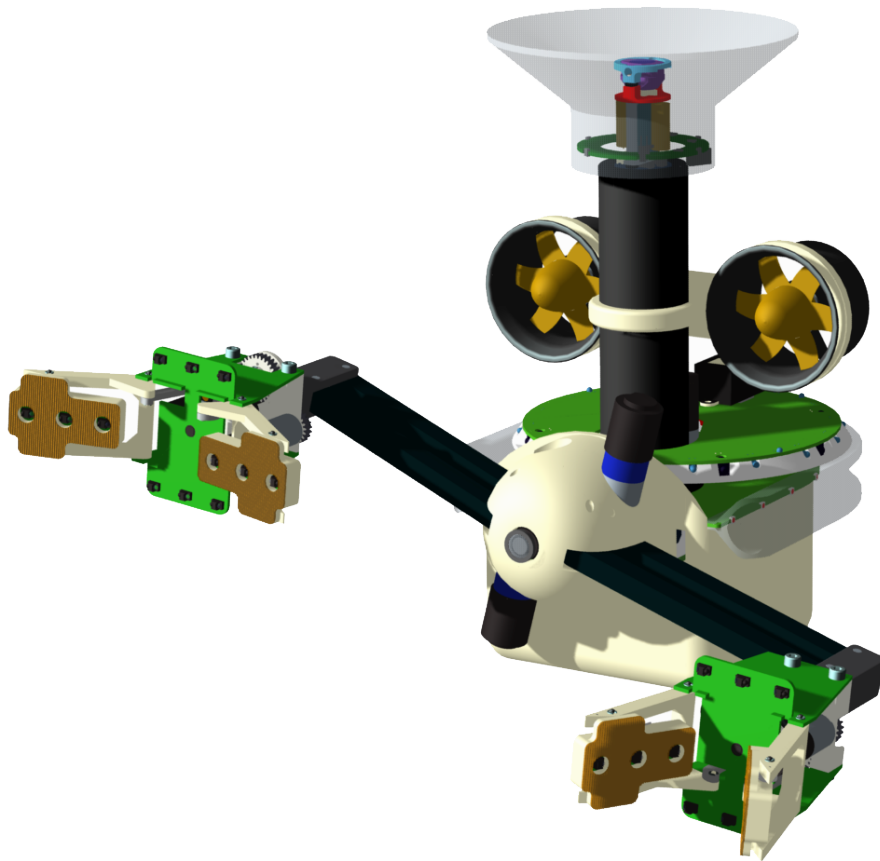


Figure 5.2 Dessin CAD du handbot.(3D)

### 5.1.1 La mécanique

La handbot, présenté dans la figure 5.2, est composé de deux pinces ayant chacune deux degrés de liberté. L'un pour fermer les mâchoires, l'autre pour tourner la pince. Les pinces sont au bout de deux bras ayant une structure parallèle. Celles-ci sont montées sur une tête qui peut tourner par rapport au reste du corps. Dans le corps, se trouve tout le système du lanceur et les moteurs permettant d'enrouler la ficelle. Sur le tube du lanceur se trouvent deux ventilateurs permettant au robot de contrôler son angle de lacet quand le handbot est suspendu.

La figure 5.3 présente les 8 degrés de liberté (ddl) du handbot. La tête peut tourner de  $-90^\circ$  à  $+180^\circ$  par rapport au corps. Les poignets des pinces sont aussi limités, mais dans le sens opposé, laissant la possibilité au handbot de garder la même orientation des pinces pendant que la tête

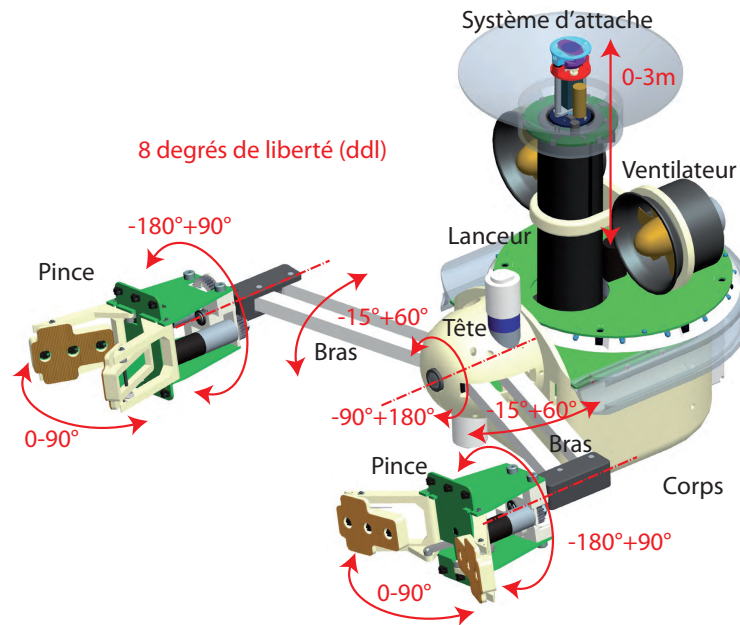


Figure 5.3 Les degrés de liberté du handbot

tourne. Les mâchoires s'ouvrent de 0 à 90°. La ficelle permet de s'élever jusqu'à une hauteur de 3m.

La figure 5.4 représente un plan éclaté du handbot. Pour simplifier sa description, le chapitre consacré au handbot a été découpé en cinq sous-parties en fonction de ses composants :

- les deux pinces
- la tête et les deux bras
- le corps principal
- le tube lanceur et les ventilateurs
- le système d'attache magnétique

### Les pinces<sup>1</sup>

Les pinces du handbot peuvent s'ouvrir totalement à 90° et se fermer complètement. Elles ont la capacité de s'adapter à différentes épaisseurs d'objets (figure 5.5). Un mécanisme avec un ressort de torsion garde les mâchoires ouvertes. Quand les mâchoires touchent un objet, les ressorts de torsion permettent une inclinaison de celles-ci. Ceci permet à la pince de s'adapter à l'épaisseur de l'objet. Les pinces peuvent exercer une force de 24N et prendre des objets épais de 40mm. Le mécanisme de fermeture se compose d'un moteur réducteur actionnant une vis sans fin qui déplace un

1. Développées avec Grégoire Terrien et Ralph Piguet de BlueBotics SA.

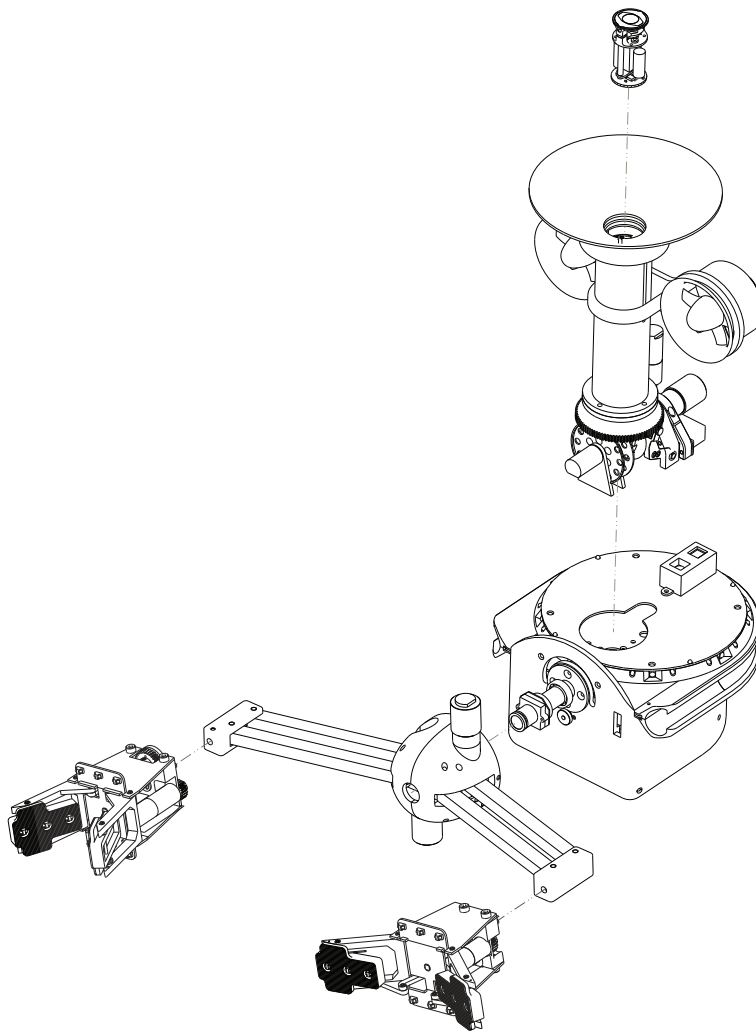


Figure 5.4 Le plan éclaté du handbot en cinq sous-parties

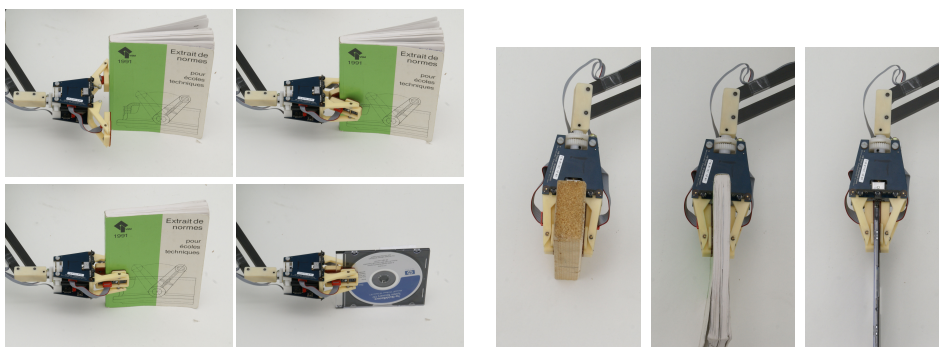
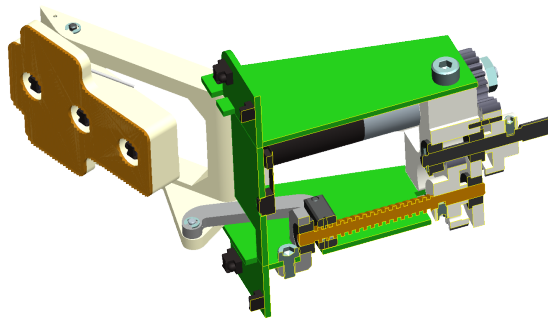


Figure 5.5 L'adaptabilité de la pince du handbot





**Figure 5.6** Une coupe de la pince

chriot (figure 5.6). Ce chariot actionne les deux bras fermant les mâchoires de la pince. Les mâchoires à l'avant sont en plastique pour limiter le poids. Une membrane en caoutchouc a été surmoulée sur les mâchoires pour augmenter leur adhérence.

La figure 5.7 montre un plan éclaté de la pince. La structure mécanique se compose d'une pièce en plastique et de trois circuits imprimés. Les PCB ont une épaisseur de 1.6mm, ce qui assure une rigidité à la structure. L'électronique de contrôle y est naturellement incluse limitant le poids et le nombre pièces de l'ensemble.

Les deux moteurs actionnant la pince sont fixés sur la pièce plastique entre les PCB. Le moteur de la rotation entraîne des engrenages droits dont un est fixe sur l'axe. Ainsi, toute la structure tourne autour de l'axe. La rotation des pinces est limitée par une goupille.

### La tête et les bras

La tête du handbot supporte les deux bras. Celle-ci peut tourner par rapport au corps de  $-90^\circ$  à  $180^\circ$  (figure 5.8 a,b). Le  $0^\circ$  étant la position horizontale. La limitation de la rotation est due aux câbles conduisant l'énergie et les données vers la tête et les pinces. Le moteur permettant la rotation de la tête se trouve dans le corps. La tête est munie d'un engrenage intérieur. L'électronique de contrôle des deux moteurs se trouve dans la tête.

La tête est formée de deux parties symétriques en plastique ressemblant à des coques (figure 5.9). Les moteurs de rotation des bras sont fixés sur celles-ci. Les bras sont composés d'une structure parallèle avec des tiges en carbone. Ils peuvent tourner autour de leur axe de pivotement avec un angle allant de  $-15^\circ$  à  $60^\circ$  (figure 5.8 c,d).



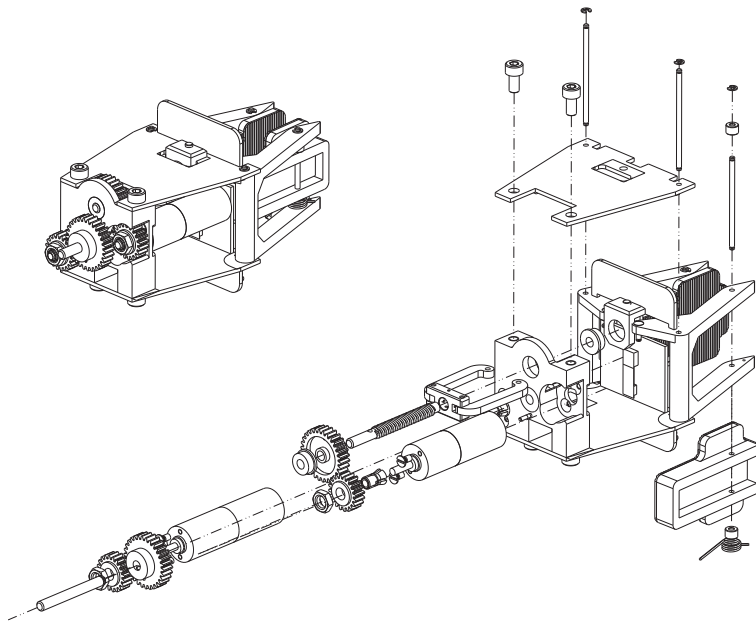


Figure 5.7 L'éclaté des pinces

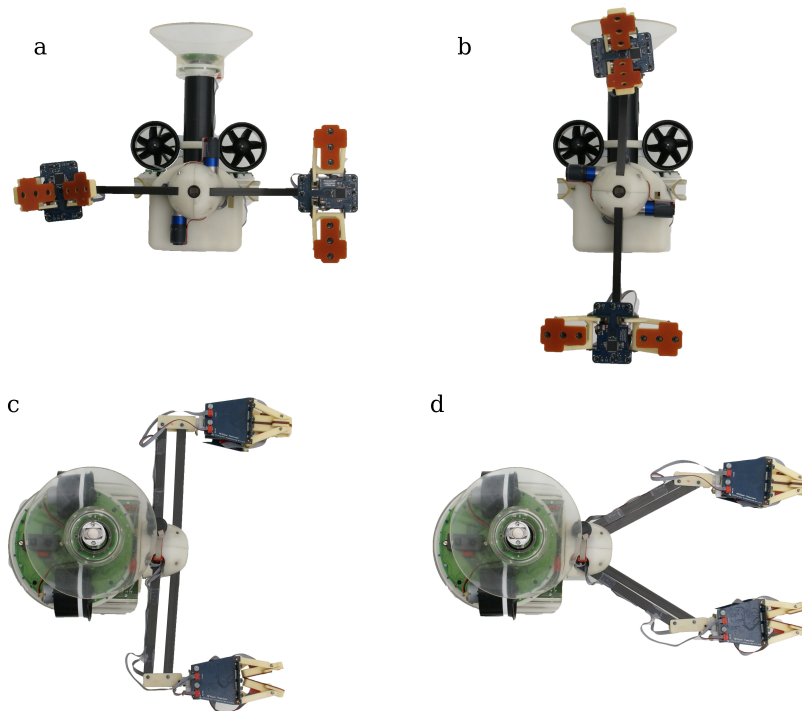


Figure 5.8 Les positions extrêmes de la tête et des bras du handbot.

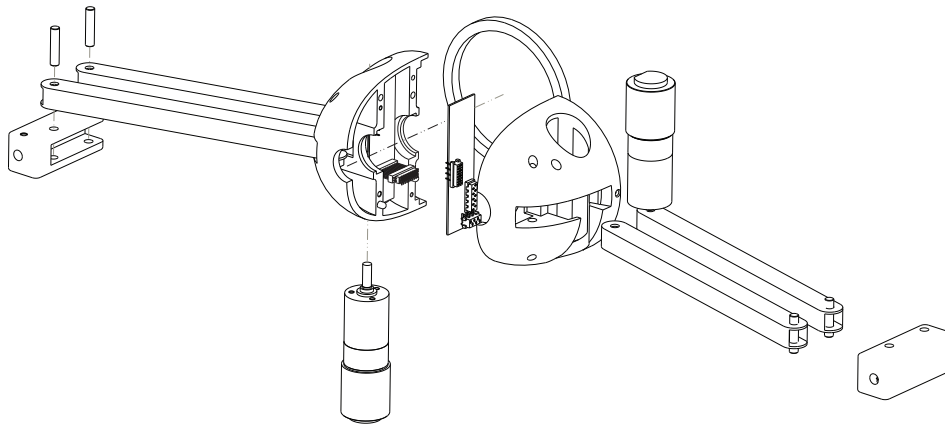


Figure 5.9 Le plan éclaté de la tête et des bras

### Le corps

Dans le corps du handbot se trouvent :

- La batterie qui est identique à celle du marXbot.
- Deux PCB servant de contact pour la batterie.
- La carte mère du processeur central i.MX31.
- Le moteur de rotation de la tête.
- L'axe de rotation de la tête.
- Une caméra frontale placée dans l'axe de la tête.
- Une couronne lumineuse d'accrochage ayant le même profil que celle du marXbot.
- Un PCB, contenant l'électronique de contrôle des moteurs du lanceur, les ventilateurs et les LED RGB de la couronne lumineuse.
- Un module "distance et relèvement" presque identique à celui du marXbot.
- Les moteurs du lanceur qui sont mieux détaillés dans la prochaine section.
- Trois capteurs de proximité au sol.

Le plan éclaté figure 5.10 présente ce corps. Mécaniquement, la majorité des pièces se fixent sur la grande pièce centrale. Cette pièce sert de structure mécanique et de coque au robot. C'est une pièce en plastique épaisse de 4 mm.

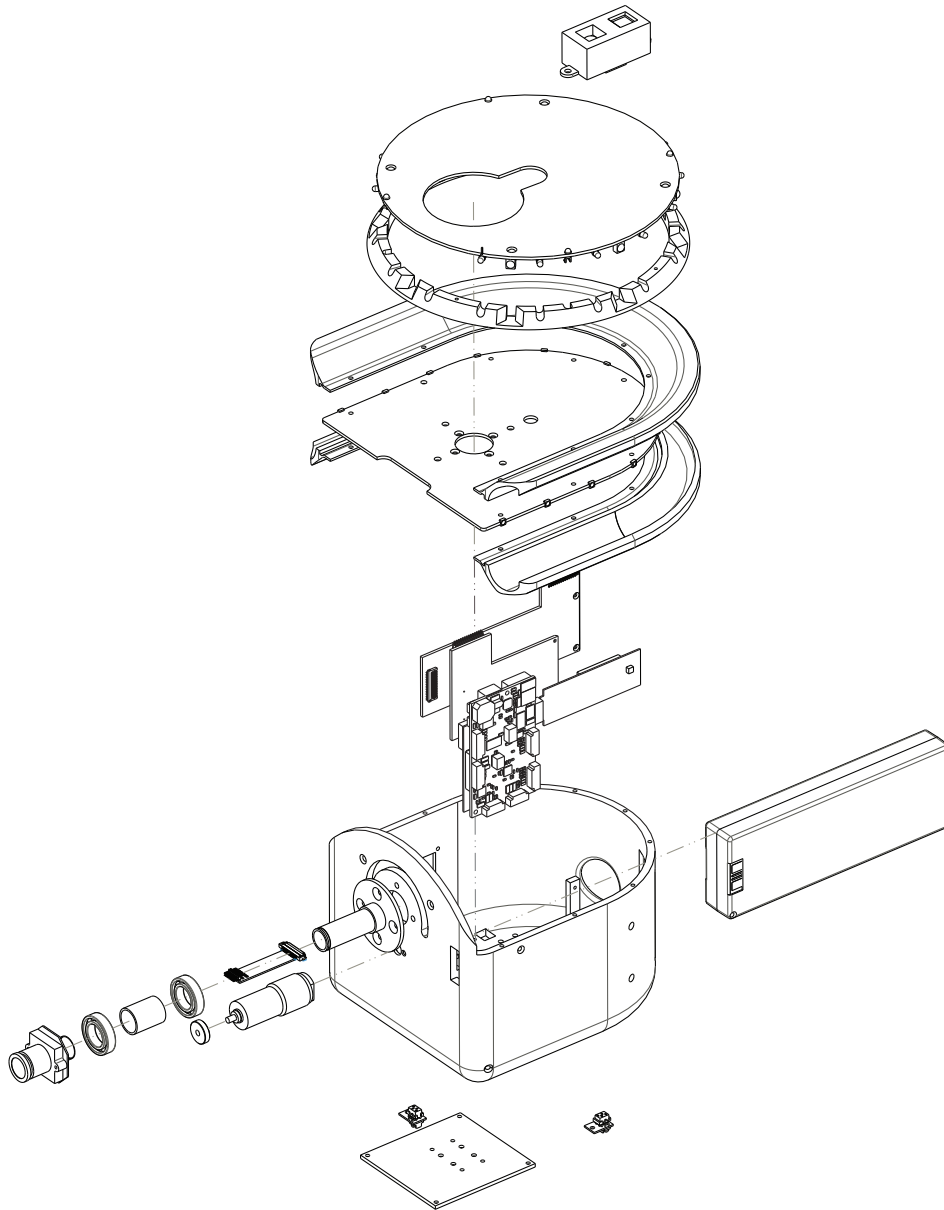
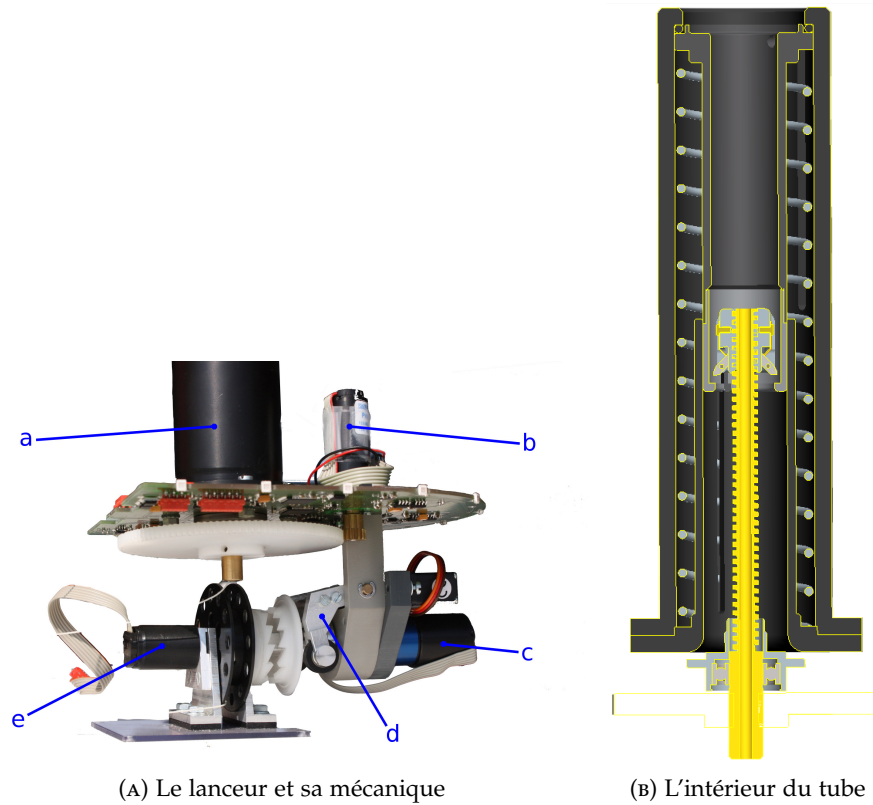
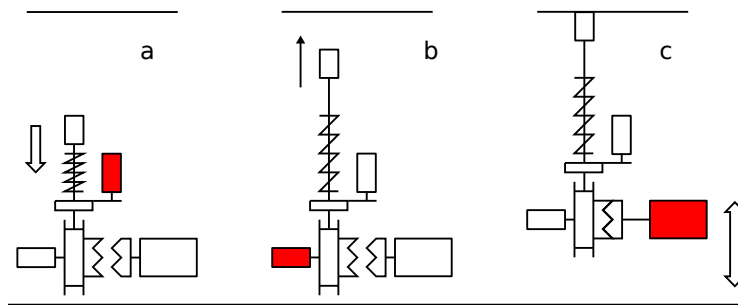


Figure 5.10 Le plan éclaté du corps du handbot



**Figure 5.11** (A) **a**: Le tube du lanceur, contenant un ressort. **b**: Le moteur chargeur compressant le ressort. **c**: Le moteur d'élévation. **d**: Le servomoteur de l'embrayage. **e**: Le moteur rapide directement branché à la poulie



**Figure 5.12** Séquence de lancement, les moteurs actifs sont en rouge. **a:** Un moteur comprime le ressort **b:** Un moteur rapide contrôle le déroulement de la bobine et garde la ficelle tendue. **c:** L’embrayage est enclenché et un moteur d’élévation soulève le robot.

### Le lanceur<sup>2</sup>

Le lanceur du handbot se compose d’un tube contenant un ressort de compression, trois moteurs et un servomoteur (figure 5.11a). La ficelle est enroulée autour d’une bobine.

Lors d’un lancé, un moteur comprime le ressort. À l’intérieur du tube, le moteur fait tourner une vis sans fin (figure 5.11b). Celle-ci entraîne un chariot qui peut s’accrocher à un tube intérieur. La vis sans fin est creuse pour laisser passer la ficelle. Quand le chariot descend le ressort se comprime. Une fois que le chariot arrive en bas, un mécanisme relâche le ressort d’un coup qui peut, par conséquent, projeter le système d’accrochage magnétique à environ 270 cm.

Durant le lancé (figure 5.12), la bobine de la ficelle prend beaucoup de vitesse et son inertie déroulerait la ficelle trop rapidement sans contrôle. C’est pourquoi un moteur rapide a été ajouté directement sur la bobine. Celui-ci freine la bobine et contrôle le déroulement de la ficelle. Ce moteur va rembobiner la ficelle dès que la hauteur cible a été atteinte. Il est aussi utilisé pour récupérer le système d’accrochage. Il enroule suffisamment vite la ficelle pendant que le projectile retombe.

Pour soulever le robot, un moteur puissant de 3.9W muni d’un grand réducteur (246 :1) est utilisé. Un embrayage est actionné par un servomoteur pour permettre le couplage du moteur de levage à la bobine. L’embrayage est composé de deux disques dentés. Les dents sont asymétriques pour permettre de tenir plus de force dans le sens du poids du robot. En effet, même si par mégarde le robot se réinitialise ou s’il n’a plus d’énergie, l’embrayage reste enclenché et évite la chute du robot.

2. Développé avec Stephan Kopf lors d’un projet de semestre et Tarek Baaboura.

La figure 5.13 représente un plan éclaté du lanceur. Sur le tube se fixent également les deux ventilateurs du handbot. Au sommet du tube, une coupelle transparente aide à récupérer le système d'accrochage quand il retombe. A l'intérieur de celui-ci, un PCB en forme d'anneau contient 6 LED RGB servant de balise au robot.

### **Le système d'attache magnétique<sup>3</sup>**

Le système d'accrochage du handbot est magnétique. Le champ magnétique peut être enclenché ou déclenché. Ceci est possible grâce à un interrupteur magnétique formé de deux aimants permanents. Les aimants sont cylindriques et magnétisés radialement. Comme expliqué par la figure 5.14, lorsque les aimants sont opposés le champ est fermé dans le système. A l'inverse, si les aimants sont en concordance, le champ passe par l'extérieur. Ainsi, le système peut passer d'une force perpendiculaire à la surface de 140N à presque rien.

La mécanique est actionnée par un moteur réducteur de basse qualité mais, néanmoins, compact (figure 5.15). Un circuit électronique muni d'un microcontrôleur contrôle le système. Le handbot commande l'ouverture et la fermeture par infra-rouge. Le système est autonome énergiquement car il est pourvu de capacités haute densité de 3,3F. Quand le dispositif est sur le robot, il est rechargé. En l'air, le microcontrôleur ne consomme que 1.4 mA et peut rester en veille pendant 35 min. Des contacts électriques ressorts permettent le contact et la recharge. Il sont protégés par un cercle en plastique. Un cône en plastique aide au centrage du dispositif lors de son retour sur le tube lanceur. Des O-rings absorbent les chocs lors du lancé.

Le handbot enclenche ainsi le système d'accrochage magnétique juste avant son lancement. Lorsque le robot retourne au sol, il envoie la commande d'ouverture au système et rembobine la ficelle en même temps que la chute de l'aimant.

### **5.1.2 L'électronique**

L'électronique du handbot est très similaire à celle du marXbot. Ainsi, la plupart des schémas des circuits ont été repris. Le routage a, néanmoins, été refait pour s'adapter à la morphologie du handbot. Elle est, ainsi, répartie sur 25 PCB (figure 5.16). Nous donnerons une vue d'ensemble sur l'électronique tout en portant une attention particulière aux spécificités propres du handbot.

---

3. La mécanique a été développée avec Frédéric Rochat et l'électronique avec Florian Vaussar et François Rey.

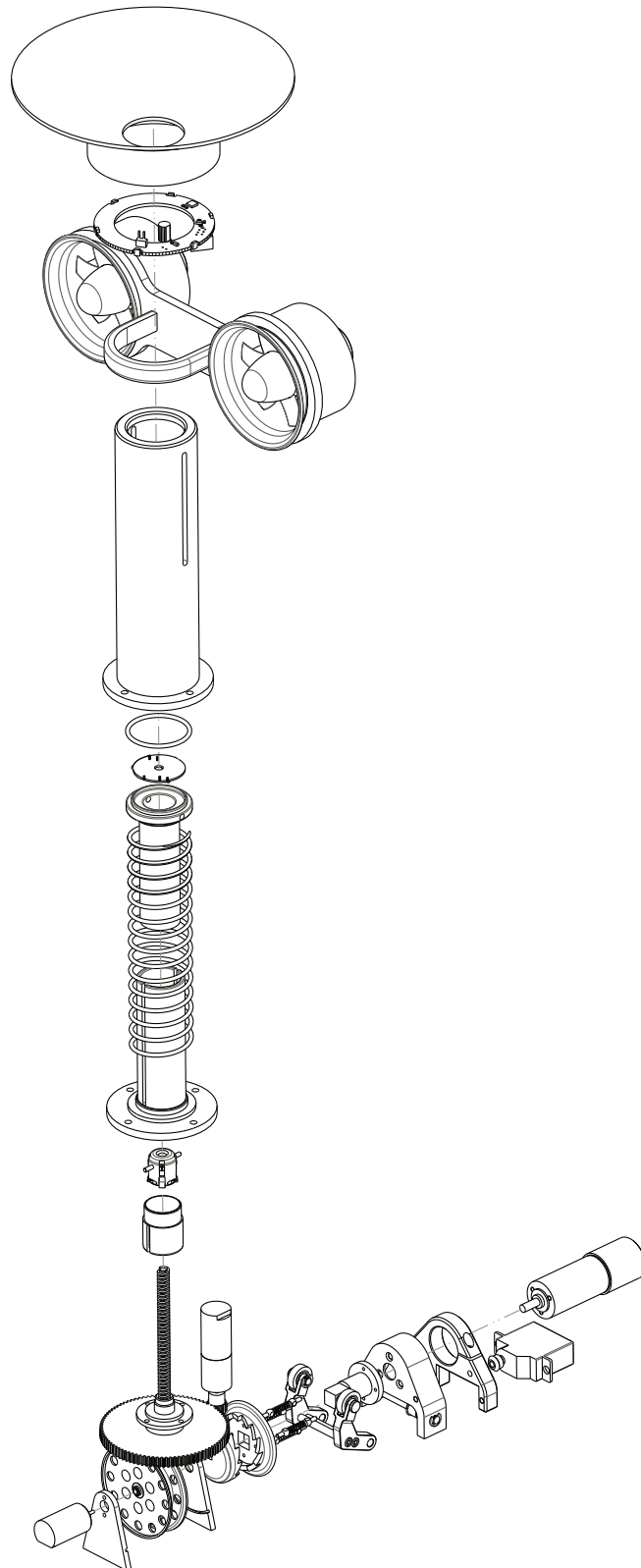
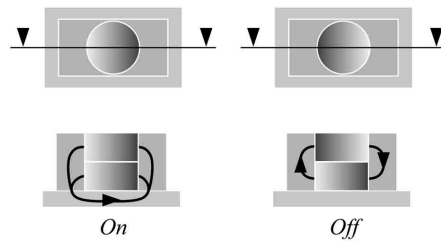
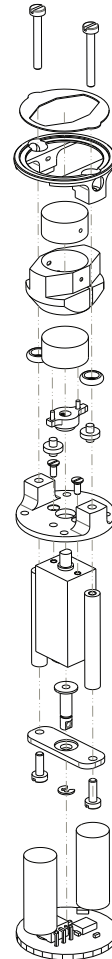


Figure 5.13 Le plan éclaté du lanceur

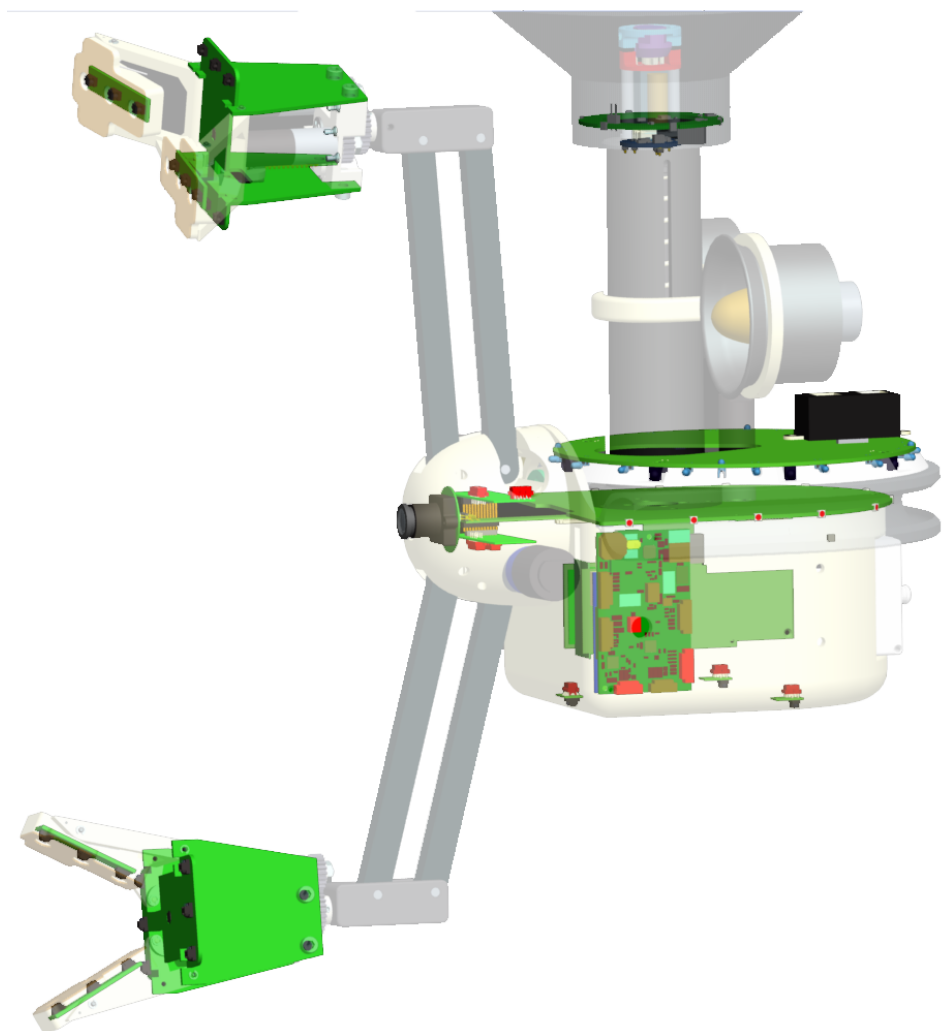


**Figure 5.14** Le principe d'interrupteur magnétique, à gauche : les aimants sont en concordance et le système magnétique est enclenché, à droite : les aimants sont opposés et le champs est confiné à l'intérieur du système



**Figure 5.15** Le plan éclaté du système d'accrochage magnétique





**Figure 5.16** Les 25 circuits imprimés du handbot représentés selon leur position dans le robot.

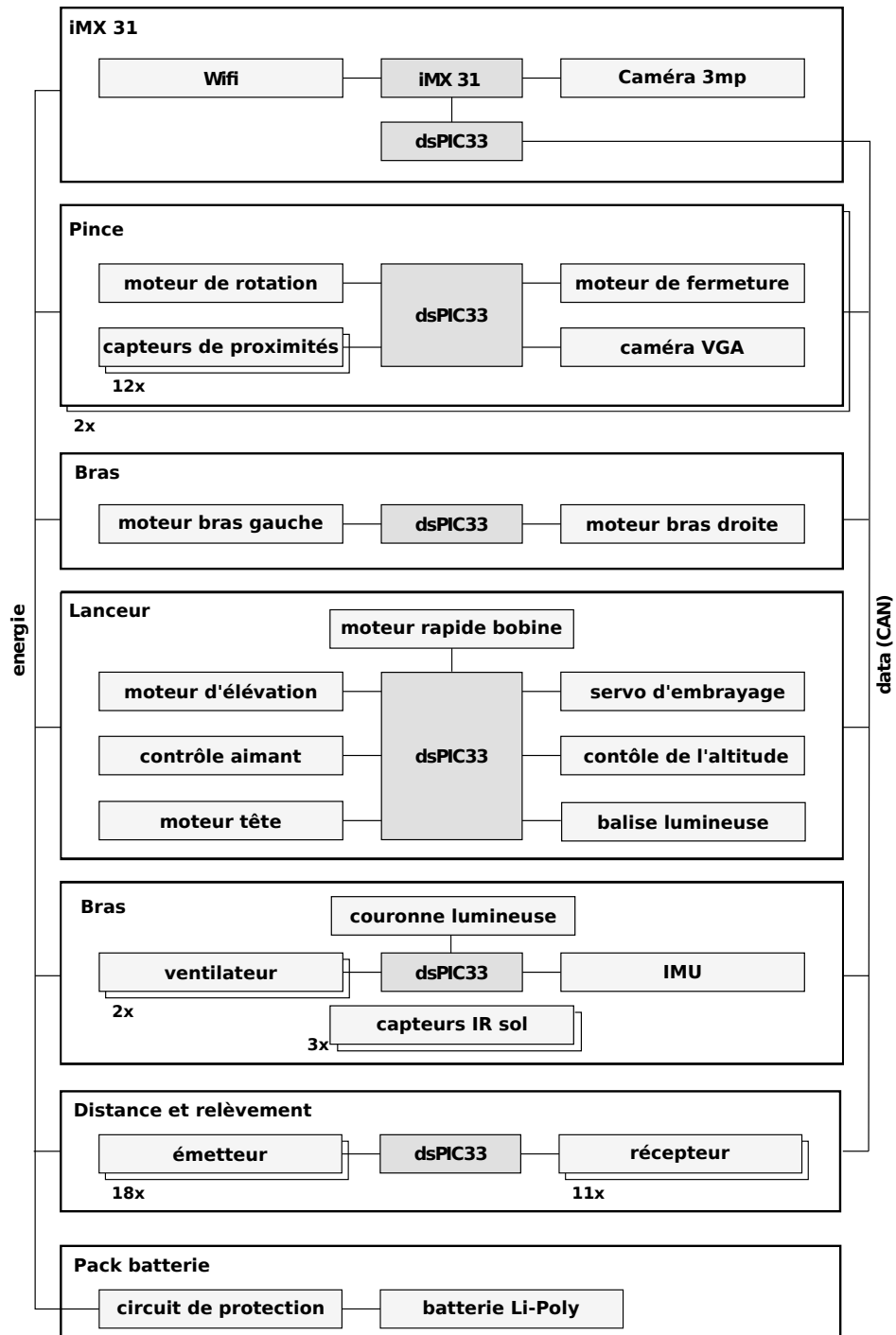


Figure 5.17 Schéma électronique complet du handbot

La figure 5.17 représente le schéma électronique global du handbot. Le processeur central est identique ; seule la carte mère a été changée et réduite pour entrer dans le robot. Les pinces comportent chacune deux contrôles moteurs identiques à ceux du marXbot, une caméra VGA, 12 capteurs IR de proximité, les capteurs au sol semblables à ceux du marXbot. Un microcontrôleur contrôle les deux moteurs des bras. Pour le lanceur, un dsPIC33 contrôle les trois moteurs et le servomoteur du lanceur, le moteur de rotation de la tête, ainsi que les LED balises au sommet du tube. Un microcontrôleur est en charge des ventilateurs et de l'IMU. Le module "distance et relèvement" est le même que celui du marXbot. Seul un récepteur et un émetteur ont été supprimés à l'avant du robot. Finalement la batterie est identique, cependant, le handbot n'est pas muni de l'alimentation de secours du marXbot.

### **La carte mère et la caméra principale <sup>4</sup>**

La carte mère du handbot est simplifiée et plus petite que celle du marXbot. Elle a été développée avec le laboratoire LIS pour permettre également d'embarquer un processeur i.MX31 sur les robots volants "eyebot". Ainsi, elle a été réduite au minimum. Elle comporte uniquement un connecteur USB host, un connecteur micro-SD, 3 UART, 1 I<sup>2</sup>C, 1 SPI et un connecteur flex-print pour la caméra. Son alimentation peut accepter du 3.6V à 10.8V afin d'être compatible au handbot et à l'eyebot.

Pour la caméra, l'électronique est semblable à celle du marXbot. Cependant, la longueur du circuit flexible provoque des perturbations sur les images. C'est pourquoi des amplificateurs de signaux ont été ajoutés pour chaque ligne de données de la caméra. Cette partie a été implantée dans le PCB (figure 5.19) qui se trouve à l'intérieur du tube servant d'axe à la tête.

### **L'électronique des pinces <sup>5</sup>**

L'électronique des pinces comporte deux contrôles moteurs identiques au marXbot. De plus, 12 capteurs IR de proximité sont montés sur les pinces. La figure 5.20 présente leur répartition. Ces capteurs permettent de détecter par exemple les bords de l'armoire et de se centrer dessus.

Une caméra VGA se trouve au centre de la pince. Elle est interfacée au dsPIC33. Ce microcontrôleur n'a pas la mémoire et la puissance de calcul pour acquérir des images complètes. Cependant des sous-échantillons de l'image peuvent être obtenus, comme par exemple une ligne, permettant

---

4. Développée avec Yannick Fournier du LIS, Valentin Longchamp et Daniel Burnier.

5. Développée avec Ralph Piguet de BlueBotics SA et Daniel Burnier.



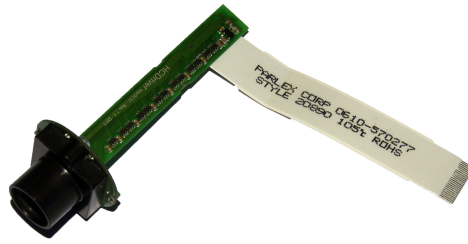


Figure 5.19 PCB de la caméra placés à l'intérieur de l'axe de la tête du handbot

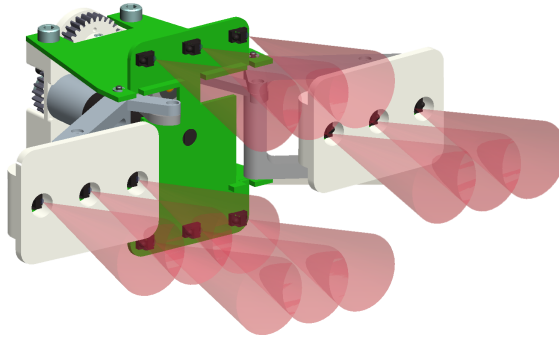


Figure 5.20 Répartition des capteurs IR de la pince

de détecter le bord de l'étagère. Cette caméra est un modèle semblable à celle montée sur l'e-puck.

### L'électronique dans le corps

Dans le corps, il y a deux microcontrôleurs. L'un gérant le lanceur et l'autre les ventilateurs ainsi que l'IMU. Le premier est en charge essentiellement de 4 contrôles de moteurs, le maximum pour un dsPIC33, un servomoteur et les LED balises au sommet du tube. Il gère aussi un émetteur IR pour communiquer au système d'accrochage. En bref, ce microcontrôleur gère tous les systèmes du handbot lui permettant de grimper.

Le deuxième dsPIC33 gère 2 ventilateurs, 3 gyroscopes et un accéléromètre à trois axes. Les gyroscopes sont de dernière génération, ils sont très précis et leur dérive est très faible. On peut ainsi très bien les intégrer et connaître l'angle d'orientation du handbot. Avec un contrôleur PID adéquat, les ventilateurs permettent d'orienter précisément le handbot quand il est suspendu à la ficelle.

### L'électronique du système d'accrochage magnétique<sup>6</sup>

L'électronique du système d'accrochage se trouve sur un petit PCB de 17mm de diamètre. Il comporte un microcontrôleur PIC12 gérant un moteur et un capteur IR démodulateur de très basse consommation. Deux super-capacités de 3.3F sont gérées par une électronique de charge et d'équilibrage spécifique. Elles fournissent l'énergie au système.

Tout le système a été conçu pour consommer le moins possible. Ainsi, il y a assez d'énergie pour que le système reste en veille pendant 35 minutes et puisse encore désenclencher l'interrupteur magnétique. Le moteur n'ayant pas d'encodeur, un pont résistif mesure la tension induite de celui-ci : ce qui permet de contrôler la vitesse du moteur et de mieux gérer la position des aimants. Pour économiser de l'énergie, le microcontrôleur diminue sa fréquence pendant qu'il est en attente des signaux venant du récepteur IR.

#### 5.1.3 Les techniques de production

Le handbot a été conçu uniquement pour le projet Swarmanoid. Bien que nous ayons prévu au départ de faire 20 exemplaires, nous n'avions pas les moyens financiers ni l'utilité d'en faire autant. C'est un robot complexe qui reste un prototype. Ainsi, seuls 8 exemplaires ont été fabriqués. Les techniques de production ont donc été adaptées à ce nombre.

La plupart des pièces plastiques ont été fabriquées par moulage. Une première pièce a été fabriquée par CNC ou par prototypage rapide. Puis un moule silicone a été utilisé pour faire les 8 pièces nécessaires. Aussi, une grande partie des pièces a été faite par usinage conventionnel. La technique des moules en POM utilisée pour le s-bot a été employée pour fabriquer les mâchoires du handbot. La partie en caoutchouc a même été surmoulée sur la partie solide en résine epoxy.

Les moteurs du handbot sont des ensembles moteurs-réducteurs. Seul le moteur de la bobine n'en comporte pas. Tous les moteurs ont aussi un encodeur, sauf le moteur de charge du ressort et le moteur du système d'accrochage magnétique. Le contrôle en courant suffit pour ces deux moteurs. Ils sont actionnés à couple constant jusqu'à ce qu'ils atteignent leurs butées.

---

6. L'électronique a été développée avec Florian Vaussard, François Rey et Philippe Rétornaz.

## 5.2 Caractéristiques du handbot

### Les degrés de liberté (ddl)

Le handbot a 10 degrés de liberté qui sont utilisés pour grimper et manipuler les objets. Les deux degrés de liberté contrôlés par les ventilateurs n'ont qu'un rôle de stabilisation lors de la descente du robot. Ainsi les 8 degrés de liberté restant sont vraiment nécessaires pour la grimpe.

Le degré de liberté gérant la ficelle et soulevant le poids du robot, permet de simplifier la mécanique des 7 restants. En effet, lors des études préliminaires, le minimum de degré de liberté imaginé était au nombre de 7 (2 d'accrochages et 5 pour le bras). Ainsi la séparation du problème de levage de la masse et le positionnement couplé à la manipulation des objets simplifie la mécanique et surtout le contrôle du robot. Avec une ficelle et un bras attaché à une structure, le robot est complètement positionné dans l'espace. Un bras grim pant aurait eu besoin de beaucoup plus de capteurs pour se déplacer.

La tâche complexe de grimpe et de manipulation requiert de ce fait tous ces degrés de liberté. Ces degrés de liberté ne sont donc pas rajoutés pour les besoins collectifs du handbot. Il lui manque même deux degrés de liberté pour être totalement autonome. En effet, le handbot est dépendant du marXbot pour se déplacer au sol.

### Les informations sensorielles extrinsèques

Voici un tableau résumant les débits de données de chaque type de capteurs :

	Types	Données brutes	Signal utile	Fréquence	Data/s
microcontrôleurs	3×IR au sol	4×12bits	10bits	1kHz	10kbit/s
	24×IR pinces	24×12bits	10	100Hz	24kbit/s
	2×caméra pince	5000p 16bits RGB565	13bits	25fps	3250kbit/s
	3D accéléromètre	3×12bits	10bits	1kHz	30kbit/s
	3×gyroscope	3×15bits	14bits	400Hz	16.8kbit/s
	D&R	11×3×10bits	12bits	250Hz	33kbit/s
	Total				3363.8kbit/s
i.MX31	caméra tête	1600×1200p 8bits bayer	8bits	20Hz	307Mbit/s

Le débit de données des caméras est celui que peut acquérir le dsPIC33 ; néanmoins, il n’a pas la puissance de calcul pour en traiter autant. On peut soit se limiter à une sous-partie de l’image, soit la transmettre par le bus CAN, mais alors, cela consommerait la moitié de la bande passante du bus.

Les 7 microcontrôleurs dsPIC33 tournent à 40 MIPS et sont des processeurs 16bits. Ce sont les mêmes types de microcontrôleurs que pour le marXbot. Les microcontrôleurs ont ainsi un total de 280 MIPS a 16bits. Le processeur 32 bits i.MX31 est cadencé à 533 Mhz, et 740 MIPS @532–665 MHz<sup>7</sup> identique au marXbot.

### La communication

Le handbot a les mêmes moyens de communication que le marXbot hormis les systèmes sonores. Il n’a qu’une caméra frontale, il est donc moins efficace que le marXbot pour détecter les robots avoisinants. Ce qui n’est pas vraiment un problème car il ne peut pas se déplacer sur le sol. Il a par contre le même système de LED, permettant au marXbot de le distinguer parfaitement. Pour d’autres détails, se référer au chapitre 4.2.

### Le prix

Le coût total du handbot est de 4033 frs. Il faut ajouter un mois-homme pour les pièces faites maison et le montage. 2 jours de tests et calibrations sont également nécessaires.

Le tableau suivant représente les coûts :

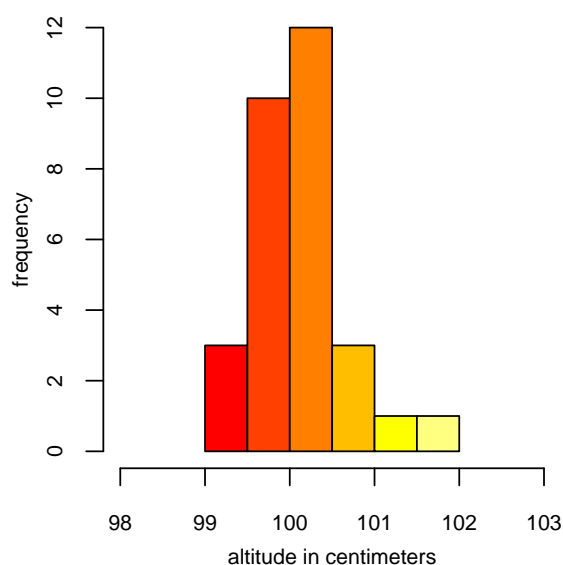
	Coûts en Frs
Mécanique	980.-
Moteurs	1343.-
Électronique	1710.-
<b>Total</b>	<b>4033.-Frs</b>

Contrairement aux autres robots, la part de l’électronique est plus faible (42%). Cela est dû aux capacités sensorielles extrinsèques qui sont moindres et à la complexité mécanique qui est élevée. Ainsi les nombreux moteurs de grande précision et de puissance élevée représentent 1/3 du prix de revient. Le coût permettant de grimper et ainsi d’accéder à la troisième dimension pour le groupe de robots a un prix non négligeable. La mécanique a encore un coût assez élevé par rapport aux autres robots (25%). Ceci est aussi dû aux handbots qui ont été fabriqués à très petite échelle, presque comme des prototypes.

---

7. Source : [http://en.wikipedia.org/wiki/ARM\\_architecture](http://en.wikipedia.org/wiki/ARM_architecture)





**Figure 5.21** Distribution de la hauteur atteinte par le handbot pour une consigne de 100 cm pour 30 essais. La moyenne empirique est de 100.2 cm la déviation standard de 0.6 cm.

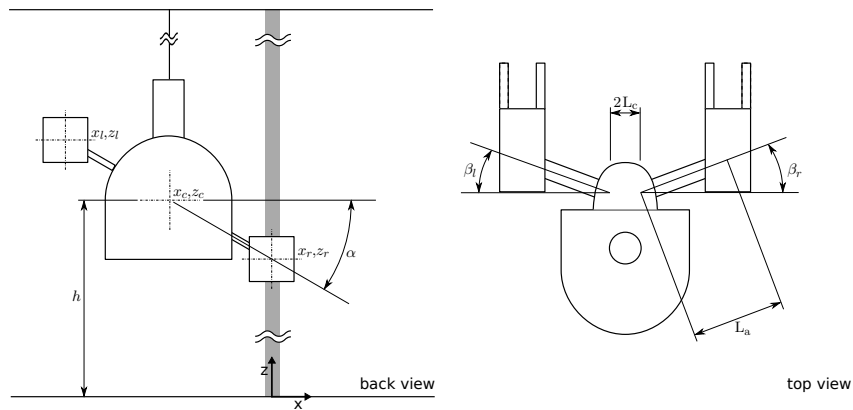
### 5.2.1 La prise d'un livre sur une étagère

Nous allons présenter les résultats d'une expérience de prise de livre décrite dans le papier "The Hand-bot, a Robot Design for Simultaneous Climbing and Manipulation" [8].

#### Le contrôle de l'altitude

L'expérience que nous présentons considère la récupération d'un livre situé à une altitude prédéfinie, ce qui signifie que la précision du contrôle de l'altitude est un facteur déterminant pour sa réussite. Le handbot compte les rotations de la bobine pour estimer la longueur de la corde enroulée. Toutefois, la corde peut se rembobiner dans différentes conditions qui pourraient affecter le rayon effectif de la bobine. En outre, la corde est légèrement élastique et sa longueur enroulée par rotation de la bobine peut changer de lancer en lancer.

Pour ces raisons, l'erreur sur l'altitude a été caractérisée. Le handbot lance le système d'accrochage magnétique, puis s'élève en enroulant de la ficelle jusqu'à ce qu'il atteigne une hauteur cible de 100 centimètres au-dessus du sol. Cette séquence a été répétée 35 fois. Dans cinq de ces essais, le lancement a échoué car l'aimant ne s'est pas fixé au plafond. Lors du lancer, le robot rembobine rapidement la ficelle après qu'elle ait été déroulée à la hauteur cible du plafond et le handbot peut ainsi mesurer le succès du lancer. Dans les 30 autres essais, le lancement a été un succès. Cela donne un taux de 86% de réussite. La figure 5.21 montre l'altitude que le handbot atteint dans ces 30 essais. L'altitude moyenne est de 100.2 cm



$$\begin{aligned}
 l_l &= L_c + L_a \cos(\beta_l) & l_r &= L_c + L_a \cos(\beta_r) & x_r &= 0 & z_r &= h - l_r \sin(\alpha) \\
 x_l &= x_r - (l_l + l_r) \cos(\alpha) & z_l &= z_r - (l_l + l_r) \sin(\alpha) & x_c &= l_r \cos(\alpha) & z_c &= h
 \end{aligned}$$

**Figure 5.22** Les équations de la position des pinces en fonction la longueur de la corde enroulée, de la rotation de la tête et de l'extensions des bras. Ces équations s'appliquent lorsque la pince droite est accrochée, elles sont symétriques lorsque la gauche est attachée. Les constantes sont les suivantes :  $L_a = 145$  mm la longueur des bras.  $L_c = 20$  mm est la moitié de la longueur entre les axes des bras dans la tête. Les variables mesurées par les capteurs intrinsèques du robot sont les suivantes :  $\beta_l, \beta_r$  sont les angles des bras gauche et droit.  $\alpha$  est l'angle de rotation de la tête.  $h$  est l'altitude du centre de la tête. Les variables recherchées sont  $x_c, z_c, x_l, z_l$  et  $x_r, z_r$  : la position du centre de la tête, de la pince gauche et de la pince droite.

et l'écart-type est de 0.6 cm. La plus grande erreur est de 2.0 cm. Le contrôle de l'altitude en utilisant la ficelle est donc suffisamment précis pour monter à une altitude cible et récupérer un objet.

Le handbot peut calculer la position d'une pince sachant la longueur de la ficelle enroulée, la position de la tête, et l'extension des bras.

### L'espace de travail

Le handbot grimpe sur les structures verticales en tournant ses bras avec une pince fixe, et commute de pince après chaque demi-tour (figure 5.23a). Dans le même temps, la ficelle enroulée par la bobine donne la principale force de levage. Comme le handbot ne nécessite que d'une seule pince attachée pour assurer sa stabilité, il peut utiliser l'autre pour manipuler des objets. Tout en gardant une pince attachée, le handbot peut atteindre des objets situés dans un disque vertical (rayon intérieur 18,2 cm, 32,0 cm de rayon externe) centré autour de cette pince. Pour accéder à des

objets hors de ce disque, il faut monter ou descendre, en changeant les extensions des bras, puis remonter à une position où il peut accéder à l'objet. De cette façon, le handbot peut atteindre tous les objets situés au maximum à 32,0 centimètres du montant vertical (figure 5.23a).

### L'escalade et la prise du livre

Dans cette expérience, le handbot commence au sol avec sa pince droite attachée au montant vertical de l'étagère. La position du livre est connue à l'avance. Le handbot lance en premier le système d'accrochage magnétique, puis monte par commutation de ses pinces, en gardant toujours au moins une pince jointe au montant. Pour ce faire, le handbot tourne la tête lentement et enroule la ficelle en conséquence en utilisant son moteur d'élévation. Il tourne activement les pinces pour les garder parallèles avec son corps, de sorte qu'il puisse saisir le montant vertical ou un livre. Le handbot utilise ses capteurs infra-rouges de la pince et l'orientation de la tête pour décider quand il peut saisir. lorsqu'il atteint une altitude déterminée (120 cm), il attrape le livre en utilisant les capteurs de proximité de la pince. Quand il a saisi le livre, il descend, librement suspendu à la corde. La vidéo 1<sup>8</sup> montre cette séquence. Cette prise prend deux minutes.

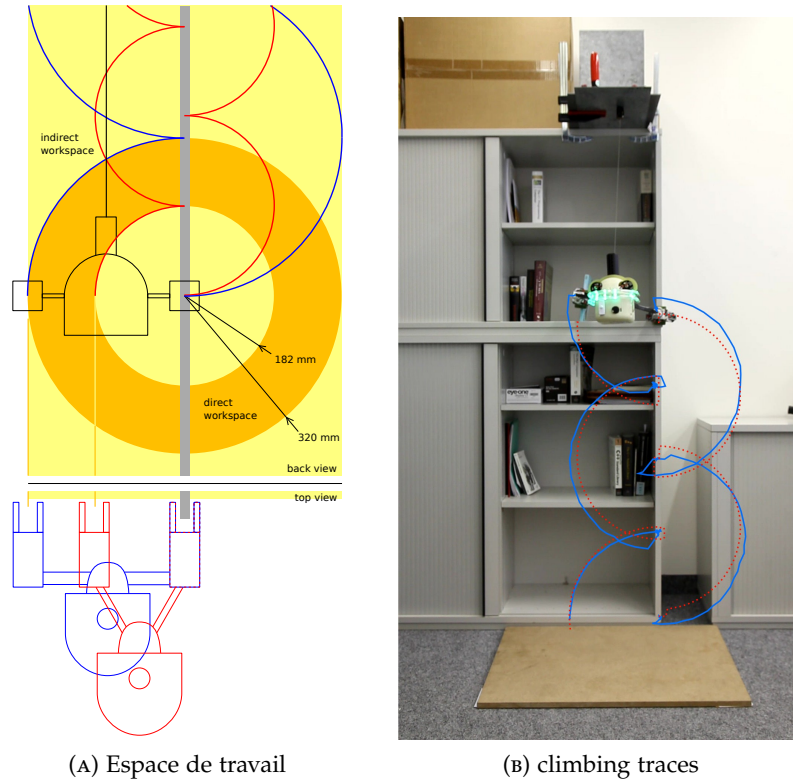
La figure 5.23 montre les trajectoires des pinces pendant que le handbot monte l'étagère et récupère le livre. La figure montre à la fois les trajectoires réelles extraites de la vidéo 2<sup>9</sup> et les trajectoires estimées par le robot. Le robot calcule en utilisant la longueur de corde enroulée, l'orientation de la tête, et les extensions des bras. Dans l'ensemble, les deux trajectoires correspondent bien. Tant la trace réelle et la trace estimée montrent un déplacement vertical, lorsque le robot attache (ou détache) l'une de ses pinces. La raison est due aux bras qui ont une structure parallèle, ainsi, lorsque le robot rétracte (ou étend) un bras, la position projetée de la pince sur le plan vertical s'éloigne (ou se rapproche) du robot.

Il y a plusieurs causes à ces différences entre les deux traces. Tout d'abord, l'escalade crée des couples élevés sur les articulations des pinces, qui affectent l'équilibre du robot qui n'est plus parfaitement horizontal. Deuxièmement, quand le robot monte, il se déplace sur l'axe horizontal, ce qui crée des écarts par rapport aux hypothèses de notre modèle cinématique qui considère une corde verticale. Toutefois, l'adaptation passive des pinces permet au robot de surmonter cette différence et de gravir avec succès de toutes façons. Troisièmement, nous avons manuellement marqué

---

8. hand-bot retrieves a book, video 1 : <http://www.youtube.com/watch?v=92bLgE6D02g>

9. hand-bot retrieves a book, video 2 : <http://www.youtube.com/watch?v=FFyqf051sik>



**Figure 5.23** (A) Espace de travail du handbot. Les trajectoires des pinces pour les deux positions extrêmes du bras sont dessinées en bleu et en rouge. Le disque orange foncé montre l'espace de travail direct, qui est le lieu des positions où le handbot peut saisir un objet avec une pince. La zone jaune indique l'espace de travail indirect, qui est le lieu de tous les endroits où le robot peut saisir un objet, après être éventuellement descendu et remonté. (B) Trace de l'escalade du handbot pour aller chercher un livre. Les lignes bleues indiquent les trajectoires réelles de la pince, extraites de la vidéo. Les pointillés rouges montrent les trajectoires des pinces, telles que calculées par le robot.

les centres des pinces toutes les 0,5 secondes sur la vidéo de l'expérience, ce qui introduit des erreurs sur les positions. De plus, la projection de la scène en trois dimensions sur le capteur à deux dimensions de l'appareil introduit également des erreurs.

#### 5.3 La prise d'un livre sans escalade

Le premier prototype du handbot disposait de deux ventilateurs plus petit qu'actuellement. Ils étaient juste suffisants pour contrer la vitesse de rotation du robot autour de la ficelle. De plus, les premiers gyroscopes utilisés avaient une grande dérive, empêchant ainsi un contrôle en position de l'angle d'orientation du handbot.

Sur le robot final, des ventilateurs plus gros et une nouvelle génération de gyroscopes permettent de stabiliser cette rotation de façon beaucoup plus efficace et permettent même de contrôler l'angle de rotation à 5° près. En conséquence, nous avons testé si le handbot pouvait prendre un livre sans garder constamment une référence fixe à l'étagère.

La figure 5.24 présente le déroulement d'un tel essai<sup>10</sup>. Le handbot est placé au bas de l'étagère face à celle-ci. Il tire son système d'accrochage magnétique comme d'habitude.

Juste avant de monter, il calibre ses gyroscopes et remet à zéro l'intégration de l'angle. Puis le robot s'élève à la recherche du tablar horizontal où le livre est posé. Le handbot s'accroche au tablar, puis s'élève légèrement pour prendre le livre. Par contre le handbot ne peut pas prendre le livre sans s'accrocher au tablar, son instabilité ferait tomber constamment le livre. Pour la descente, le robot tourne de 180° et déroule doucement la ficelle.

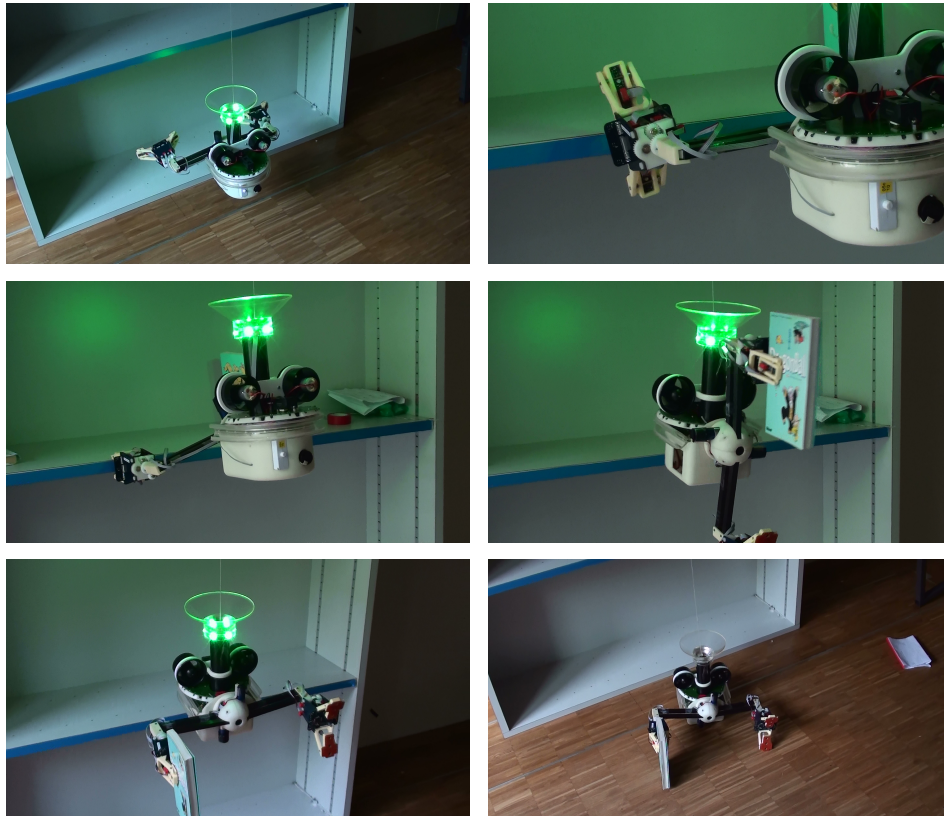
Ainsi le handbot est également capable de prendre un livre sans référence constante. Par contre, le positionnement de départ doit être précis horizontalement et angulairement par rapport à l'étagère, ce que ne peuvent pas vraiment faire les marXbot transportant le handbot.

#### 5.4 Leçons apprises, limites et améliorations

Le handbot est sans doute le robot le plus compliqué que j'ai développé d'un point de vue mécanique. Son développement a été laborieux et beaucoup d'aspects restent à améliorer. Par conséquent, la fiabilité du robot est encore faible. Le design est à sa deuxième version et les 8 exemplaires produits restent des prototypes. Il aurait fallu un prototype supplémentaire entre la première version et la production des 8 exemplaires afin

---

10. handbot retrieves automatically a book, video : <http://www.youtube.com/watch?v=wZtvVs6kHLg>



**Figure 5.24** Séquence du handbot prenant un livre en utilisant uniquement les ventilateurs pour se stabiliser. Une description détaillée de la démarche se trouve dans le texte.

d'améliorer les points suivants :

- Les pinces et l'avant du robot sont trop lourds. Pour équilibrer le robot et garantir qu'il reste à peu près à l'horizontal quand il est suspendu, il faut rajouter près de 800g de contrepoids. Ce poids est non négligeable par rapport à sa masse totale de 2.8kg. On pourrait également avancer le tube mais cela impliquerait trop de changements sur la disposition des moteurs à l'intérieur du corps.
- Le couple du moteur de rotation de la tête est juste insuffisant, un peu plus de puissance faciliterait la grimpe de l'étagère.
- La bobine du lanceur est un élément critique pour la réussite d'un lancer. Elle doit à la fois éviter que le ficelle s'emmêle et tourner le plus librement possible. Il faut régulièrement ajuster sa position sur l'axe et il serait judicieux d'adapter le design pour éviter ces soucis.
- Le système de recharge du ressort du lanceur s'use et le système se bloque par moments. Il est vrai que les forces en jeu sont très grandes et un petit ajustement est nécessaire pour garantir une meilleure fiabilité.
- L'embrayage pose quelques problèmes bien que la dernière version soit fonctionnelle. Avec un servomoteur ayant légèrement plus de couple ou avec une transmission plus directe on éviterait ces défauts.
- La structure des bras en carbone casse régulièrement. Il y a un point de rupture vers l'axe du moteur, ce qui fait que la structure de la barre de carbone se fissure à cet endroit.
- Une des pièces de la mâchoire a un point faible et se casse souvent. De plus, les mâchoires ne peuvent pas se fermer correctement si la pince est complètement à l'appui du bord de l'étagère.
- Le système d'accrochage magnétique subit de gros chocs, aussi, les contacts garantissant la recharge se cassent régulièrement.
- Il y a un problème général d'alimentation du robots, les gros moteurs des bras et les ventilateurs peuvent faire couper la batterie par le système de protection (il coupe à 9A) car ils peuvent au démarrage dépasser cette limite. Les câbles plats transportant le courants au jusqu'aux pinces ne pas pas adaptés. Il arrive parfois qu'il y a une chute de tension qui réinitialise le microcontrôleur.
- Le routage du PCB principal du corps est à refaire, il y a des boucles de masse et la plupart des amplis de mesure du courant des moteurs sont bruités, il a fallu retoucher à plusieurs reprise ce circuit. L'alimentation des capteurs de distance au sol est fortement bruitée par les moteurs et ils sont dès lors inutilisables.

### 5.5 Conclusion

Le robot handbot est capable de grimper et prendre un livre sur une étagère de façon autonome, ce qu'aucun autre robot de cette taille n'est capable de faire. Il réussit cela grâce à la synergie innovante entre ses capacités d'escalade et de manipulation. Le handbot est également compatible et dépendant du marXbot pour se déplacer au sol, ouvrant la perspective d'étudier les comportements collectifs d'auto-assemblage avec des robots hétérogènes.

Les dix degrés de liberté, les 13 moteurs et l'électronique de contrôle nécessaire rendent le handbot fort complexe et également coûteux.



## 6 Capacités collectives en auto-assemblage

Hormis l'e-puck, les robots présentés dans les précédents chapitres peuvent s'auto-assembler. Dans ce chapitre les performances des robots lorsqu'ils sont auto-assemblés vont être présentées et étudiées.

### 6.1 Super-linéarité des performances des robots

En robotique collective, on caractérise souvent les performances d'un groupe en utilisant la notion de facteur d'accélération collective ou "collective speedup factor (CS)" [51]. Ce facteur apporte une évaluation quantitative de la performance d'un groupe de robots. Il est calculé par la formule suivante :

$$CS(n) = \frac{mP(n)}{nP(m)} \quad (6.1)$$

$P(n)$  est la performance d'un groupe de  $n$  robots et  $m$  est le nombre minimal de robots nécessaires pour accomplir la tâche. Dans la plupart des cas, on peut ainsi comparer la performance du groupe à celle de l'individu. La performance est super-linéaire lorsque  $CS(n) > 1$ , est linéaire lorsque  $CS(n) = 1$  et est sous-linéaire lorsque  $CS(n) < 1$ . Si une tâche peut être réalisée avec un robot ( $m = 1$ ), une simple combinaison de  $n$  robots génère une performance linéaire si les tâches sont effectuées  $n$  fois mieux ou plus vite qu'un robot. Ce résultat n'est possible que si les robots n'ont pas d'influence les uns sur les autres ce qui est rarement le cas car ils partagent généralement une même ressource ou un même espace. Cependant, dans certaines situations, les interférences sont constructives et aident à mieux résoudre la tâche, engendrant des performances super-linéaires (voir la collaboration de prise de bâtonnets [39] et le temps de poussée d'une boîte [52]). Les tâches où nous pouvons observer les performances super-linéaires sont bien sûr les applications les plus intéressantes pour la robotique collective.

En plus de la caractérisation de la linéarité de la performance, il est important de vérifier l'extensibilité de cette propriété. Il est bien sûr plus intéressant lorsque les performances s'étendent à un grand nombre de robots.

Nous allons dans cette section analyser différentes propriétés super-linéaires des robots s-bot, marXbot et handbot et nous intéresser plus particulièrement aux interactions physiques de ces robots, liées à leur capacité d'auto-assemblage.

### 6.1.1 La super-linéarité des performances du s-bot

Nous allons présenter les résultats publiés dans l'article "Superlinear Physical Performances in a SWARM-BOT" [53]. L'accélération collective du s-bot a été mesurée dans trois expériences pour trois caractéristiques : la force de traction d'un objet, la capacité à franchir un interstice et la capacité à franchir une marche. Un groupe de s-bots, aussi appelé SWARM-BOT, a des performances super-linéaires pour ces trois tâches et pour un faible nombre de robots.

#### La force de traction d'un groupe de s-bot



**Figure 6.1** (a) 5 s-bots tirent en chaîne sur un dynamomètre. (b) un s-bot tire sur le même dynamomètre.

Les s-bots tirent en chaîne sur un dynamomètre comme présenté dans la figure 6.1a. La force de traction du groupe est ainsi mesurée à trois reprises pour  $n[1,5]$  sur quatre sols différents. La moyenne de ces trois mesures et l'accélération collective sont présentées dans le tableau 6.1.

Les groupes de s-bots ont des performances super-linéaires comparable au s-bot unique à l'exception de l'expérience sur le sol 4 où la performance est presque linéaire. Les CS sont en moyenne super-linéaires jusqu'à  $n = 5$  mais sont à leur optimum pour 2 robots  $CS(2) = 1.4$ . Ceci est dû au fait qu'entre un et deux s-bots, il y a une stabilisation physique. En effet, comme le montre la figure 6.1b, la pince du s-bot est située au-dessus du centre de masse et le robot bascule quand il tire sur le dynamomètre. Les chenilles ne sont alors pas en contact et la force de traction ne peut

Nombre de s-bot $n$ composant le groupe	1	2	3	4	5
Force moyenne sol 1 $P_1(n)$ [N]	2.65	6.75	11.4	15.1	18.5
$CS_1(n)$	1	1.27	1.43	1.42	1.4
Force moyenne sol 2 $P_2(n)$ [N]	2	8	11	12	17.5
$CS_2(n)$	1	2	1.8	1.5	1.75
Force moyenne sol 3 $P_3(n)$ [N]	3	7.5	12	13.5	15
$CS_3(n)$	1	1.25	1.33	1.12	1
Force moyenne sol 4 $P_4(n)$ [N]	4.7	10	11.6	19.2	23.5
$CS_4(n)$	1	1.06	0.82	1.02	1
Moyenne $CS(n)$	1	1.4	1.36	1.27	1.29

Tableau 6.1 Force de traction

pas être optimum. Avec deux s-bots la structure reste horizontale et les chenilles sont davantage en contact avec le sol.

Pour un SWARM-BOT composé de plus de deux s-bots il n'y a aucune amélioration structurelle supplémentaire, ce qui se manifeste par la perte de la valeur du  $CS(n)$  pour  $n > 2$ . Cette baisse de l'accélération collective lorsque  $n > 3$  est due à l'alignement des robots qui n'est plus parfait et ainsi la somme de la force n'est pas maximale. Pour un plus grand nombre de robots ( $n > 5$ ) les forces sont suffisamment fortes pour briser la pince du premier robot de la chaîne. Pour cette raison, les tests n'ont été faits que pour un maximum de 5 s-bots.

### Le passage d'un fossé

Dans cette tâche, le but est de franchir un fossé. La configuration des s-bots est une chaîne comme dans l'expérience précédente (figure 6.2), mais ici la rigidité de la chaîne est exploitée. Dans cette expérience, les robots sont assemblés à la main et contrôlés par un programme simple de détection du trou permettant aux s-bots de compenser la flexion en activant le levage de la pince. Dans cette expérience la performance était déterminée pas la taille maximale du fossé franchissable par les chaînes de s-bots.

Le tableau 6.2 présente les performances  $P(n)$  et les accélérations collectives qui en résultent  $CS(n)$ . Pour  $n \geq 4$ , la taille de l'interstice est constante car la pince ne peut supporter plus de deux s-bots suspendus horizontalement. En outre, lorsque deux s-bots sont suspendus horizontalement, le troisième robot qui les soutient a une très forte pression sur les chenilles. Cette pression peut bloquer les chenilles et immobiliser le s-bot, bloquant tout le groupe.



Figure 6.2 Trois s-bots passant un fossé.

Nombre de robots	1	2	3	4	5
$P(n) = \text{taille du fossé [cm]}$	4	9	18	22	22
$CS(n) = \frac{P(n)}{nP(1)}$	1	1.125	1.5	1.375	1.1

Tableau 6.2 Performance de franchissement de fossé

Malgré les limitations mentionnées ci-dessus pour  $n \geq 4$ , les mesures montrent une accélération collective distinctive pour 3 et 4 s-bots. La meilleure accélération est obtenue pour  $n = 3$ , qui s'explique par une raison structurelle, comme l'illustre la figure 6.3. Quand deux s-bots sont assemblés, leur structure devient plus de deux fois la longueur d'un s-bot en raison de la longueur du dispositif de connexion. Cette longueur supplémentaire n'est pas bien exploitée pour une chaîne composée de deux s-bots, car le centre de masse n'est pas placé sur les chenilles. Avec trois s-bots, le groupe peut exploiter pleinement sa longueur, car le centre de masse est situé sur les chenilles du deuxième s-bot.

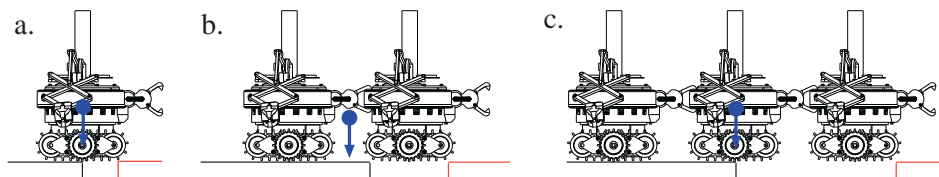
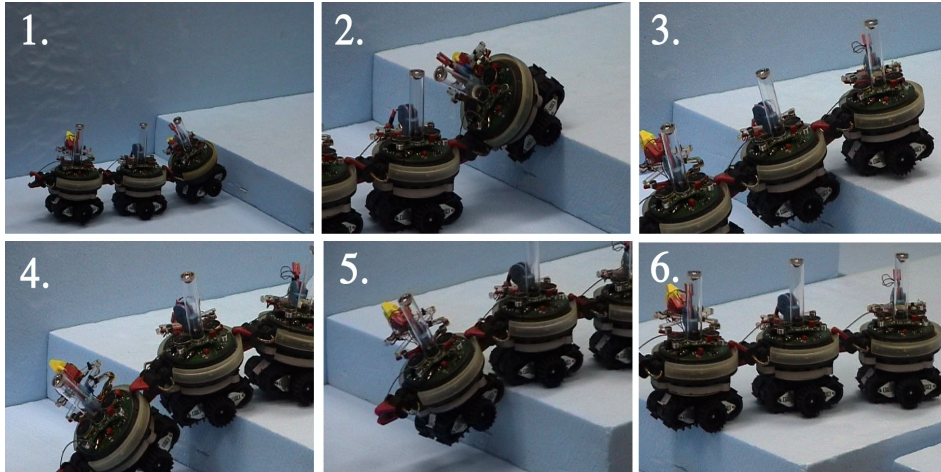


Figure 6.3 Centre des masses d'un SWARM-BOT face au fossé en fonction du nombre de s-bot connectés. La taille maximale que le groupe peut passer dépend de la position du centre de masse par rapport aux chenilles.

### Le passage d'une marche

Dans cette tâche, l'objectif du SWARM-BOT est de passer une marche. La configuration de la chaîne est également utilisée pour cette expérience. Comme l'illustre la figure 6.4, la particularité de cette tâche est que le SWARM-BOT doit se plier correctement pour passer la marche. Pour réaliser cette tâche, les s-bots ont été télécommandés à distance. La performance est quantifiée par la hauteur maximale de la marche que le groupe de robots peut passer.



**Figure 6.4** Séquence d'un SWARM-BOT composé de trois s-bots passant une marche de 10 cm.

Le tableau 6.3 présente les résultats. Le nombre de s-bots a été limité à cinq en raison de contraintes mécaniques : l'effort mécanique appliqué à la bague de connexion à travers le robot est très important lorsque vous pliez la structure. Avec le matériau utilisé et pour  $n > 5$ , la bague peut se briser. En raison de cette limitation, la hauteur de la marche ( $P(n)$ ) ne serait pas augmentée de manière significative pour  $n > 5$ . Cette tâche donne des résultats impressionnants du point de vue de la navigation tout-terrain. Les s-bots peuvent passer une marche de leur propre taille.

Nombre de robots	1	2	3	4	5
$P(n)$ = hauteur de la marche [cm]	1.5	4.5	10	14	16
$CS(n) = \frac{P(n)}{nP(1)}$	1	1.5	2.22	2.33	2.13

**Tableau 6.3** Performances de passage d'une marche

L'accélération collective mesurée dans cette tâche est la plus élevée parmi toutes les expériences effectuées. Le CS le plus haut est obtenu pour  $n = 4$

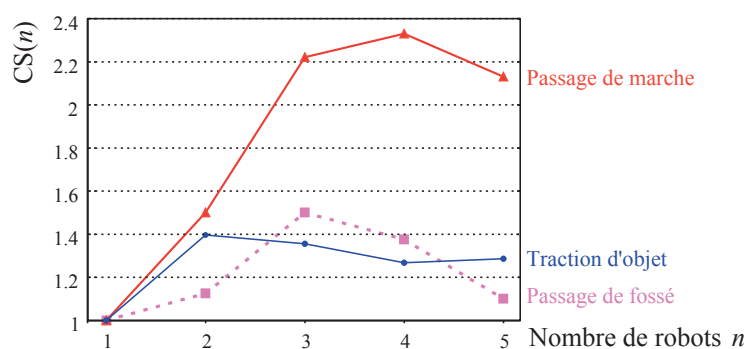
et les accélérations sont super-linéaires pour toutes les configurations. Les raisons de cette performance super-linéaires sont dues à la meilleure stabilité structurelle de la configuration du SWARM-BOT. Un seul s-bot, malgré ses chenilles, a une capacité très limitée pour la navigation tout-terrain, en raison de son centre de masse relativement élevé. C'est la raison de la très mauvaise performance d'un s-bot pour franchir une marche. Une configuration de SWARM-BOT composée de deux s-bots a une stabilité bien meilleure et peut traiter une grande variété de conditions de terrain. En plus de la stabilité de la structure, le SWARM-BOT peut exploiter la flexibilité offerte par l'articulation de la pince. En articulant la structure, le SWARM-BOT peut améliorer de manière significative sa mobilité tout-terrain. C'est la principale explication de la grande  $CS(n)$  dans le cas de  $n = 3, 4$  et  $5$ . Ces configurations ont plus d'articulations dans la chaîne et donc plus de souplesse permettant un meilleur franchissement d'obstacle.

### Analyse de la super-linéarité du s-bot

Trois expériences montrant les performances du SWARM-BOT en fonction du nombre de s-bots qui composent une chaîne ont été présentées. Nous pouvons observer deux propriétés principales :

- **Des performances super-linéaires** : Les trois expériences montrent des performances super-linéaires. Ceci est une indication claire qu'une connexion physique joue un rôle constructif dans la collaboration entre les robots. La super-linéarité est due à une interaction physique, modifiant la structure du groupe et améliorant ainsi ses performances physiques.
- **L'extensibilité limitée** : L'extensibilité des résultats est limitée à un petit nombre ( $2 < n < 5$ ) de la taille du groupe. Il s'agit d'une limitation claire de notre système. Néanmoins les limites supérieures sont nettement liées à des caractéristiques physiques et mécaniques des robots. Malgré les possibilités d'amélioration de la conception, les contraintes physiques seront toujours une limite supérieure à la super-linéarité des performances.

La figure 6.5 récapitule l'accélération collective présentée dans cette section. Nous pouvons voir que la plus petite accélération est obtenue par la plus simple des tâches (traction d'objet), où les caractéristiques du lien physique entre les s-bots sont les moins exploitées. Dans ce cas, le lien est utilisé seulement pour créer une connexion en traction mais n'exploite pas la rigidité ou la mobilité de la liaison. La deuxième tâche utilise la rigidité de la liaison et obtient une meilleure accélération avec une valeur maximale pour un plus grand nombre de robots. La dernière tâche, le passage de marche, exploite toutes les propriétés du lien physique (la



**Figure 6.5** Récapitulatif de l'accélération collective en fonction du nombre de s-bots pour chacune des trois tâches présentées dans cet article.

rigidité et la mobilité) et réalise les meilleures performances collectives pour un plus grand nombre de robots. Ces résultats démontrent la relation entre l'exploitation de la liaison physique et l'accélération collective.

Les expériences décrites ci-dessus donnent une indication de la taille optimale d'un SWARM-BOT pour aborder des tâches physiques. Par exemple, les performances mesurées montrent que les chaînes de quatre robots permettent d'exploiter de manière optimale leur physiologie. Par conséquent, le SWARM-BOT le plus efficace pour la navigation tout-terrain doit avoir un rayon de quatre s-bots.

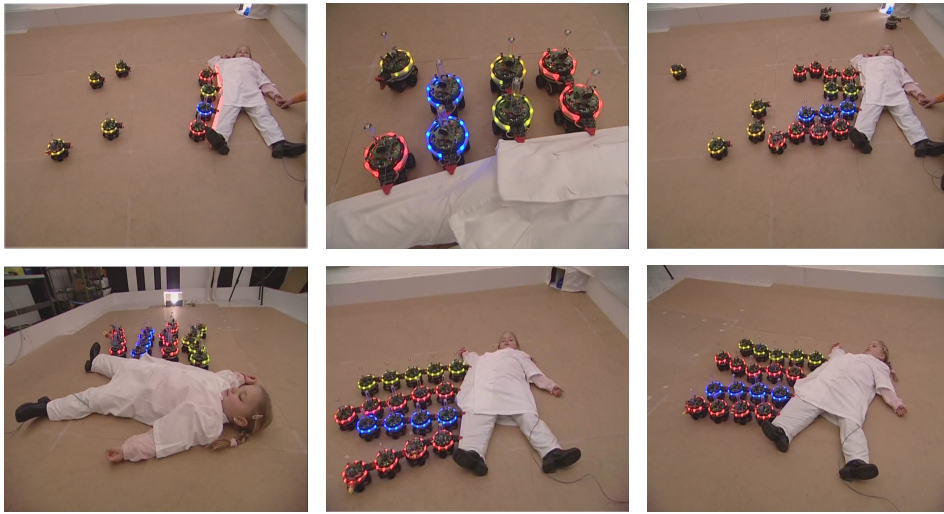
#### Application de la super-linéarité du s-bot : traction d'un enfant<sup>1</sup>

La bonne capacité pour tirer du SWARM-BOT a été exploitée dans une expérience de traction d'un enfant. Les s-bots s'accrochent à l'enfant de façon autonome en formant des chaînes (figure 6.6). Les chaînes sont distinguées par différentes couleurs. Ainsi, quand un s-bot s'approche, il choisit une couleur prédéfinie pour s'assembler. Puis une fois les chaînes terminées les s-bots tirent l'enfant en ayant pour cible une lumière pour garder un azimut commun.

Une application de recherche et de sauvetage est ainsi envisageable avec ce type de robots. Leur mobilité tout-terrain, leur force additionnée et leur grand nombre autorisant une recherche distribuée permettraient d'être efficace dans le genre de situation d'urgence imaginé dans la figure 6.7.

1. Cette expérience a été réalisée par Ivan Aloisio lors de son projet de diplôme sur la base du travail d'auto-assemblage de Roderich Gross [31].





**Figure 6.6** Séquence de la traction d'un enfant. Vidéo sur <http://www.youtube.com/watch?v=CJ0ubyiITsE>



**Figure 6.7** Exemple d'application, sauvetage de personne (photo-montage)



### 6.1.2 La super-linéarité des marXbots

Une fois les marXbots auto-assemblés grâce à leur module d'accrochage, on peut mesurer leur force de traction. Comme pour le s-bot, les marXbots ont été mis en chaîne pour tirer sur un dynamomètre. La force de traction du groupe est ainsi mesurée pour  $n[1,4]$  sur trois sols différents. La moyenne de ces trois mesures et l'accélération collective sont présentées dans le tableau 6.4.

Nombre de marXbots $n$ composant le groupe	1	2	3	4
Force moyenne sol 1 $P_1(n)$ [N]	10.7	26.7	46	57.3
$CS_1(n)$	1	1.21	1.39	1.3
Force moyenne sol 2 $P_2(n)$ [N]	8.2	24.7	46.7	56.5
$CS_2(n)$	1	1.51	1.9	1.73
Force moyenne sol 3 $P_3(n)$ [N]	10.7	23.7	43.2	47.5 <sup>1</sup>
$CS_3(n)$	1	1.11	1.35	1.11
Moyenne $CS(n)$	1	1.28	1.55	1.38

**Tableau 6.4** Force de traction et accélération collective en fonction du sol et du nombre de robots.

Les marXbots ont aussi une caractérisation super-linéaire lors de la traction. Comme le s-bot, un marXbot seul a tendance à se basculer lorsqu'il tire. Ainsi dès que l'on ajoute un robot, la structure reste plaquée au sol et augmente son efficacité. Quand il y a deux robots, le premier est plaqué au sol, et le deuxième patine légèrement. Ce qui introduit une vibration dans la structure et la rend moins efficace. L'accélération collective est encore plus grande avec  $n = 3$  robots car les deux premiers robots sont bien plaqués au sol et le troisième patine légèrement. Par la suite en ajoutant un robot en plus on ne peut qu'ajouter linéairement sa force. Il n'introduit plus d'effet mécanique super-linéaire. De plus, avec la force de quatre robots, le système d'accrochage n'est pas assez résistant pour retenir toute la force.

En comparaison, deux marXbots ont plus de force de traction que 5 s-bots. Leur plus grande taille, leurs chenilles plus larges et leur plus grande puissance expliquent ce gain en performance.

1. Le système d'attachement a lâché à ce moment-là, normalement la force devrait être plus élevée.

### 6.1.3 La super-linéarité des handbots

#### Les performances d'un handbot seul

Dans cette partie nous présentons les résultats de l'article "Physical interactions in swarm robotics : the hand-bot case study" [9].

En utilisant le système d'attachement magnétique, les ventilateurs et les bras, un handbot unique peut fonctionner suivant deux stratégies principales :

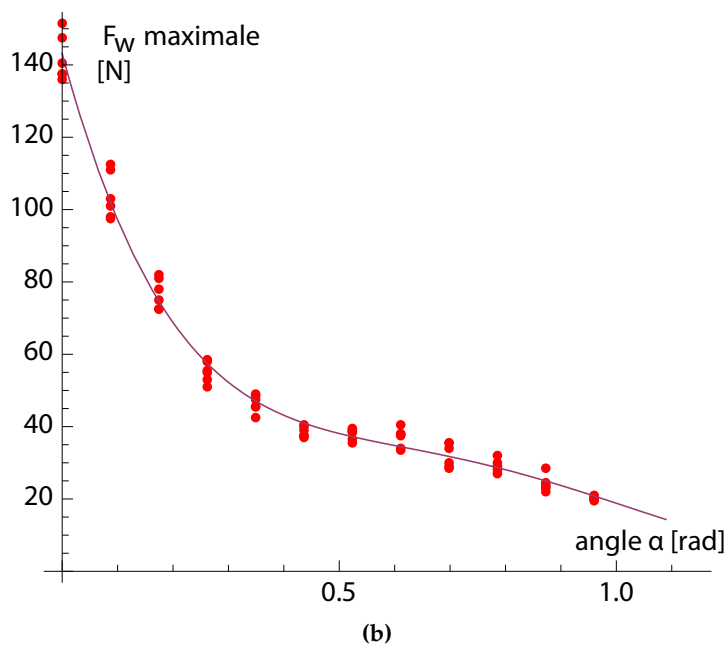
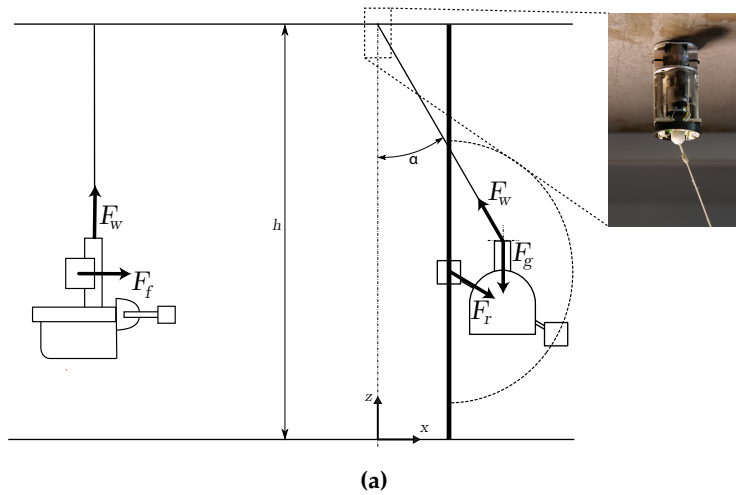
1. Le handbot s'élève dans l'air en utilisant sa ficelle. Il peut se stabiliser et contrôler sa rotation en utilisant les ventilateurs. Il peut également faire de très petits mouvements en avant et en arrière. Les deux pinces restent libres pour la manipulation d'objets, mais le positionnement et la stabilité sont médiocres.
2. Le handbot peut grimper en suivant une structure comme dans la partie 5.2.1, présentant la prise du livre. Le robot garde sa stabilité et a un positionnement précis. Les pinces et les bras lui permettent d'avoir également de la mobilité latérale. Cependant, il ne lui reste qu'une pince de libre et il ne peut pas grimper et manipuler un objet en même temps.

Dans les deux cas, le principe est d'utiliser la fixation au plafond pour fournir la force de levage verticale ( $F_w$  dans la figure 6.8a). En outre, dans la stratégie 1, le handbot peut se déplacer légèrement sur l'axe horizontal grâce aux ventilateurs qui fournissent une force horizontale ( $F_f$  à la figure 6.8a). Dans la stratégie 2, le robot peut pousser ou tirer sur la structure avec sa pince fixée. La résultante ( $F_r$  dans la figure 6.8a) est alignée avec les bras.

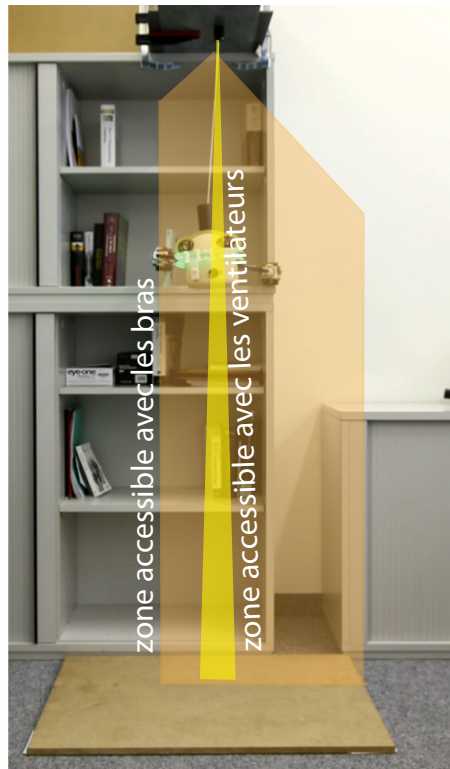
Comme le montre la figure 6.8b, la force maximale de fixation  $F_w$  dépend de l'orientation de la ficelle. En effet, comme le point d'attache est situé au bas du dispositif de fixation, la force  $F_w$  génère un moment qui tend à décoller l'aimant lorsque le câble n'est pas vertical. La valeur maximale de  $F_w$  en fonction de l'angle  $\alpha$  a été mesurée sur le dispositif réel, parce que la force d'un aimant en décollement est extrêmement difficile à calculer de manière analytique. Les valeurs obtenues sont illustrées à la figure 6.8b.

Il apparaît clairement que la principale limitation de la première stratégie (sur la base des ventilateurs) est la faible propulsion des ventilateurs, résultant à un volume accessible limité. Lorsque les ventilateurs soufflent à pleine puissance, ils peuvent générer un angle (mesuré expérimentalement) de seulement 0.026 rad (1.5°).

La principale limitation de la seconde approche est l'utilisation d'un bras et d'une pince pour le déplacement latéral, ce qui les rend indisponibles pour la manipulation. La zone accessible est beaucoup plus grande et



**Figure 6.8** (a) Schéma des forces et détails sur le système d'accrochage magnétique. (b) La force maximale magnétique en fonction de l'angle  $\alpha$ . Les mesures sont représentées par des points rouges, la ligne affiche une interpolation possible.



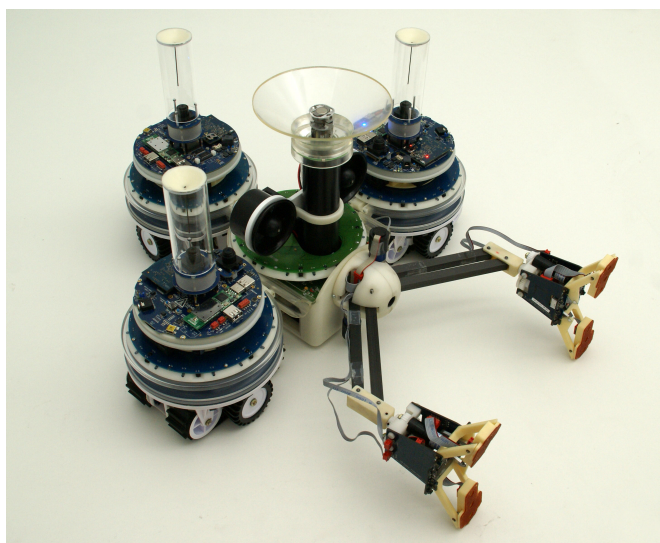
**Figure 6.9** Zones d'accès latérales par rapport aux forces fournies par les ventilateurs et les bras

et est limité par la morphologie. En outre, près du plafond l'angle de la ficelle augmente et le système magnétique ne peut plus soutenir le robot. Les deux zones d'accès sont illustrées dans la figure 6.9.

### L'auto-assemblage des marXbots aux handbots

La handbot est strictement spécialisé dans l'escalade et la manipulation. Il n'est pas équipé de roues ou de tout autre actionneur pour se déplacer sur le sol. Les déplacements se font grâce à l'auto-assemblage avec le marXbot, appelé aussi pour cette occasion footbot. Les footbots peuvent s'auto-assembler sur les deux côtés et à l'arrière du handbot comme illustré dans la figure 6.10.

Cet assemblage de robots peut fournir au handbot la mobilité nécessaire sur le sol. Ses performances en déplacement dépendent fortement du nombre de robots impliqués. Un seul footbot peut déplacer un handbot, mais ne peut guère le positionner correctement. Un footbot connecté à un handbot ne peut que le tirer ou le pousser. Il ne peut se déplacer



**Figure 6.10** Configuration de trois footbots pour transporter un handbot.

dans d'autres directions, car cela le ferait presque tourner sur place. Si le footbot tire le handbot, il est presque impossible de le déplacer en face d'une étagère. Si le footbot pousse le handbot, le contrôle est très instable et nécessite des manoeuvres très complexes.

Une meilleure configuration est réalisée avec deux footbots placés latéralement au handbot. Cela permet au groupe de robots de se déplacer dans n'importe quelle direction. Le positionnement sur le plan horizontal est beaucoup plus précis et facile à contrôler. La situation idéale se produit lorsque le handbot est relié à trois footbots, deux placés latéralement et l'un derrière. Cette configuration permet de pousser ou de tirer dans toutes les directions. L'auto-assemblage avec plus de trois robots n'a pas de sens du point de vue de la stabilité et de la mobilité et il est difficile en raison de la faible surface de fixation.

### **L'auto-assemblage des handbots**

Dans les sections précédentes nous avons vu qu'il est possible au handbot de se déplacer verticalement et d'utiliser l'auto-assemblage pour se positionner sur le terrain. La possibilité la plus intéressante est de s'auto-assembler à plusieurs handbots et d'étendre le volume accessible par les robots. Le handbot peut s'auto-assembler avec ses pinces aux bras d'autres handbots, comme l'illustre la figure 6.11.

Le scénario complet réalisable grâce à l'auto-assemblage entre les footbots et les handbots est le suivant : Les footbots déplacent les handbots au bord de la zone de travail. Dans les exemples suivants nous allons consi-

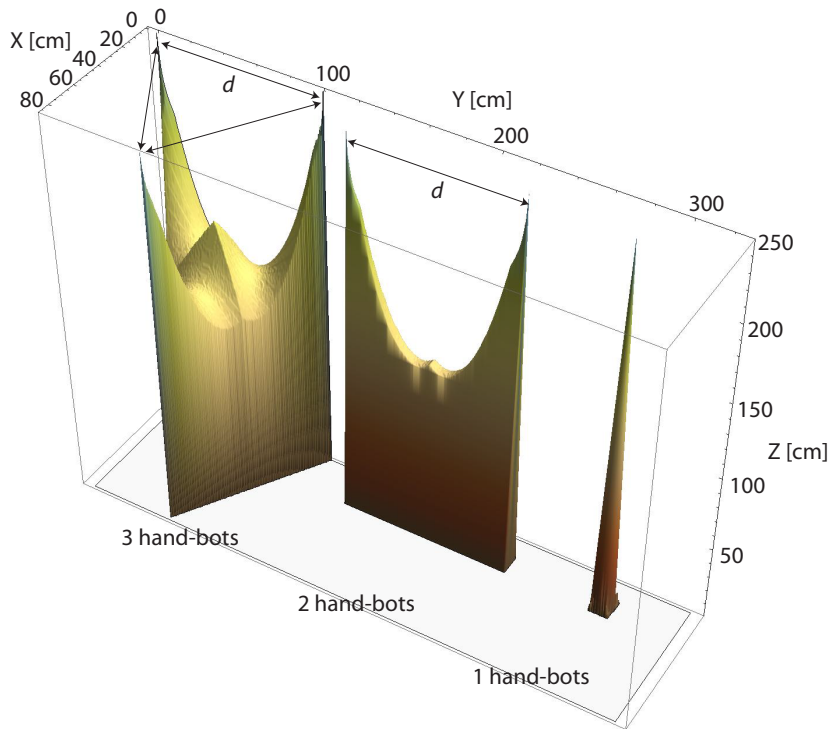


**Figure 6.11** Trois handbots assemblés et suspendus par leur câble.

dérer deux ou trois handbots placés à une distance  $d$  les uns des autres. Quand les handbots sont placés, ils lancent leur aimant pour s'accrocher au plafond. Puis les footbots rassemblent tous les handbots au centre de la zone de travail. Cela permet aux handbots de s'auto-assembler entre eux à l'aide de leurs pinces. Une fois réunis, ils peuvent se soulever dans les airs en enroulant en même temps leur ficelle. En tirant plus ou moins sur celles-ci, le groupe peut se déplacer dans l'espace.

Comme illustré dans la partie 6.1.3, le handbot peut utiliser deux actionneurs principaux pour les déplacements latéraux : les ventilateurs et les bras. Par l'auto-assemblage, nous pouvons ajouter un troisième actionneur, qui est le système de fixation au plafond d'un autre handbot. Si l'on considère la situation de l'élévation du handbot dans l'air en gardant les deux pinces disponibles pour la manipulation, nous pouvons distinguer trois situations principales :

1. Quand le handbot est seul, il peut contrôler ses mouvements latéraux et de rotation sur lui-même (trois degrés de liberté "DOF") en utilisant seulement les ventilateurs. Le volume accessible est limité.
2. Lorsque deux handbots sont assemblés, deux degrés de liberté sont contrôlés par cette connexion supplémentaire et un seul degré de liberté doit être contrôlé par les ventilateurs. Le volume est plus grand et suit un plan vertical entre les deux points de fixation.



**Figure 6.12** Volume accessible pour 3, 2 et 1 handbots auto-assemblés. Les volumes calculés sont pour une distance inter-robots de  $d = 100\text{cm}$ .

3. Quand les trois handbots sont assemblés, tous les degrés de liberté sont contrôlés par les cordes reliées au plafond. Le volume devient plus important et est inclus dans les trois plans verticaux passant par les points de fixation.

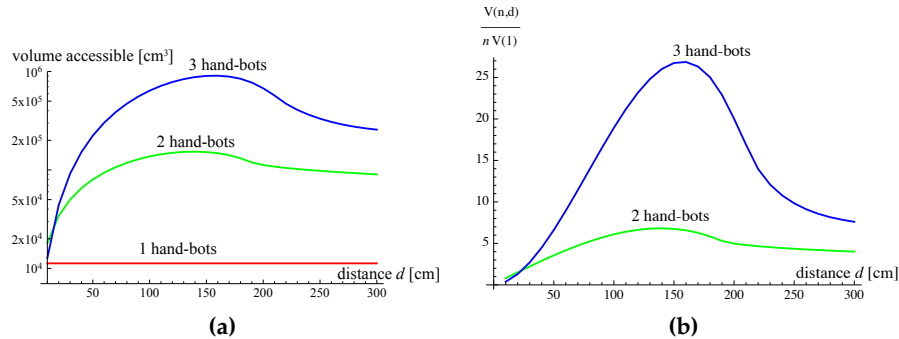
Dans les deuxième et troisième situations, les limites supérieures du volume est donnée par la force d'attachement par rapport à l'angle, comme illustré sur la figure 6.8b. Les volumes qui en résultent pour une distance  $d$  de 100 cm entre les robots sont illustrés dans le graphique 6.12.

Nous avons utilisé ces volumes pour mesurer l'accélération collective d'un groupe de handbots auto-assemblés suivant la formule :

$$CS_v(n, d) = \frac{V(n, d)}{nV(1)} \quad (6.2)$$

où  $V(n, d)$  est le volume accessible par  $n$  handbots avec leur système d'accrochage magnétique placé à une distance  $d$  l'un de l'autre et  $V(1)$  le volume accessible par un handbot utilisant ses ventilateurs pour les déplacements latéraux. Le graphique 6.13a présente le volume accessible  $V(n, d)$  en fonction de  $d$  pour  $n = 1, 2$  et 3 handbots. Le graphique 6.13b

représentant l'accélération collective  $CS_v(n, d)$  montre des performances hautement super-linéaires.



**Figure 6.13** (a) Volume accessible par 1,2 et 3 handbots utilisant l'auto-assemblage. (b) Accélération collective pour 2 et 3 handbots en fonction de la distance  $d$ .

#### 6.1.4 Analyse de la super-linéarité du handbot

L'exemple de l'auto-assemblage des footbots et handbots montre clairement que la collaboration physique peut générer une très forte multiplication de la performance. Cela avait déjà été observé avec le s-bot, mais le facteur de  $CS$  a été beaucoup plus faible que celui observé ici.

Dans la tâche de positionnement du handbot par les footbots, la performance est difficile à définir et à mesurer, mais l'amélioration des performances est claire lorsqu'on passe d'un à deux footbots. Trois footbots ont de meilleures performances, même si le  $CS$  entre deux et trois robots pourraient être sous-linéaire. En effet, le gain en performance entre deux et trois robots semble être mineur par rapport à l'augmentation de 50% du nombre. D'autres footbots peuvent améliorer la robustesse, la charge utile et la vitesse du système, mais n'améliorent pas la qualité de positionnement ou l'efficacité du contrôle.

Pour le volume de travail accessible, la quantification mesurée montre clairement les performances super-linéaires. L'auto-assemblage de deux handbots génère un facteur accélération collective de 6,8 dans le meilleur des cas ( $d = 140$  cm). L'ajout d'un autre handbot permet d'atteindre un  $CS$  de 26,8, ce qui représente une accélération supplémentaire d'un facteur quatre par rapport à une combinaison de deux robots. Ces performances sont obtenues par la géométrie du système, où l'ajout d'un robot permet d'ajouter une dimension à l'ensemble. Ceci est très spécifique à des systèmes qui collaborent par des interactions physiques. Ces performances ne sont également pas extensibles, car au-dessus de trois robots, que ce



soit pour les footbots ou pour les handbots, les performances du groupe restent stables et donc le CS diminue.

Les facteurs d'accélération énumérés ci-dessus ne considèrent pas le nombre total de robots, qui devrait tenir compte de l'hétérogénéité du système. En prenant en considération les deux types de robots, le rapport  $m/n$  n'augmenterait ainsi que le facteur d'accélération. Cela montre un effet très intéressant de l'hétérogénéité. Des robots à la mobilité différente pour les opérations au sol et verticales permettent d'optimiser chaque sous-système de façon indépendante et de combiner leurs fonctionnalités d'une manière orthogonale. Par exemple, une fois que trois footbots sont disponibles, le nombre de handbots peut être augmenté librement, car les footbots sont capables de positionner chaque handbot de manière séquentielle. Cela accroît également l'accélération collective.

Enfin, il y a une observation intéressante à faire sur la configuration matérielle requise pour l'auto-assemblage. Les marXbots, qui sont des robots mobiles avec une mécanique simple pour se déplacer —une tâche triviale— nécessitent un module matériel spécifique pour assurer la capacité d'auto-assemblage. L'ajout du système d'accrochage est l'un des composants matériels les plus complexes dans les footbots. Le handbot, au contraire, est équipé dans sa configuration de base avec des pinces pour accomplir cette tâche fondamentale de manipulation. En utilisant ces mêmes pinces pour l'auto-assemblage, le handbot ne nécessite pas de matériel supplémentaire pour effectuer une coopération physique. Ceci suggère que, dans des tâches plus complexes exigeant à la base des robots plus élaborés, la coopération basée sur l'auto-assemblage pourrait être plus accessible et nécessiterait moins de capacités spécifiques supplémentaires.

### 6.2 Exemples d'expériences collectives complètes

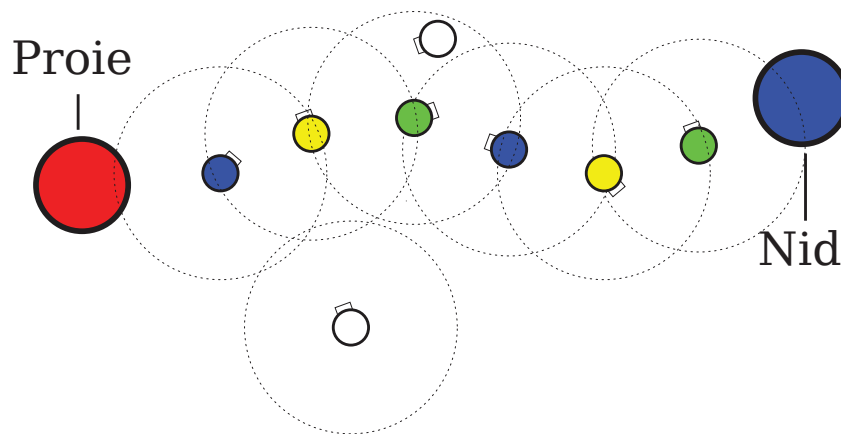
#### 6.2.1 Les s-bots à la recherche de nourriture<sup>2</sup>

Dans cette partie nous allons présenter succinctement les résultats de l'article "Teamwork in Self-Organized Robot Colonies" [59].

Dans l'expérience décrite dans cet article les s-bots doivent rapporter de la nourriture à leur nid. Il s'agit plus précisément de ramener un objet de masse importante ayant une couronne lumineuse semblable au s-bot à un second objet identique. La nourriture, ou la proie, est représentée par cet objet avec les LED allumées en rouge, le nid est en bleu. La masse de

---

2. La programmation du comportement et l'expérience ont été faites par Roderich Gross et Shervin Nouyan, j'ai surtout soutenu le travail en maintenant une flotte de robots disponible.



**Figure 6.14** Exemple de chaîne lumineuse entre la proie et le nid.

l'objet nécessite au minimum 2 s-bots pour le transporter, mais avec trois robots le déplacement est plus efficace.

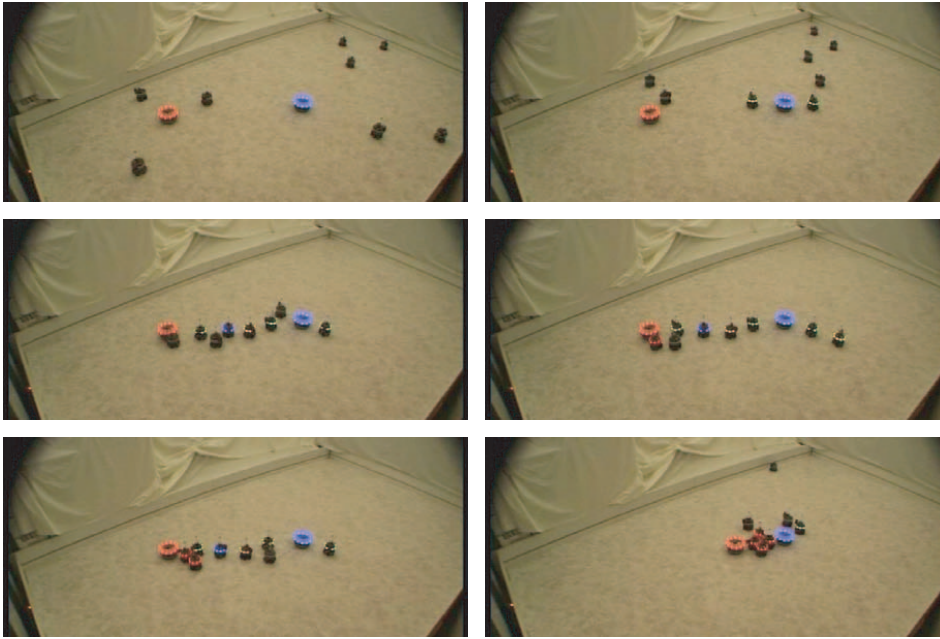
Les s-bots utilisent essentiellement leur caméra omnidirectionnelle pour distinguer les LED des autres robots et des objets, leur vision est limitée à 30-40 cm. Localement les capteurs infrarouges sont utilisés pour distinguer les murs et les objets.

Pour rallier le nid, les robots forment une chaîne lumineuse avec leur corps comme représenté dans le figure 6.14. Les s-bots peuvent prendre trois couleurs dans la chaîne : bleu, jaune ou vert. Grâce à ces trois couleurs, un s-bot qui découvrirait la chaîne peut savoir le sens de celle-ci. Il peut ainsi remonter la chaîne et voir si la proie se trouve au bout. Sinon le s-bot devient lui même membre de la chaîne et se place en bout de celle-ci activant ses LED de la couleur appropriée.

La figure 6.15 montre une séquence<sup>3</sup> de l'ensemble de l'expérience.

Au début, les s-bots sont disposés aléatoirement dans l'arène. Ils commencent à chercher soit une chaîne soit le nid. Le premier s-bot arrivé proche du nid va commencer une chaîne de façon aléatoire. Les s-bots suivants vont soit compléter la chaîne, soit en commencer une autre. Un s-bot qui est au milieu de la chaîne doit y rester, par contre celui qui est au bout, attend un certain temps légèrement aléatoire et a une certaine probabilité de la quitter. Ainsi, la chaîne peut se construire et se détruire de façon probabiliste. Cela permet au groupe d'explorer différentes directions depuis le nid.

3. Vidéo accessible sur : [http://www.youtube.com/watch?v=Xs\\_Y22N1r\\_A&feature=related](http://www.youtube.com/watch?v=Xs_Y22N1r_A&feature=related)



**Figure 6.15** Séquence de l'expérience de s-bots à la recherche de nourriture. Une description détaillée de la démarche se trouve dans le texte.

Si un s-bot arrive au bout d'une chaîne et aperçoit la proie, il va s'aligner sur celle-ci. Les robots qui sont au centre de la chaîne cherchent constamment à rester alignés avec leur voisin. Ainsi toute la chaîne s'aligne en direction de la proie.

Les s-bots suivants qui arrivent au bout de la chaîne, vont pouvoir s'accrocher à la proie. Une fois accrochés, ils commencent à ramener l'objet en suivant la chaîne lumineuse. Les membres de bout de chaîne voyant la proie revenir, vont quitter la chaîne et reprendre l'exploration de l'environnement. Ils peuvent également joindre le groupe pour tracter la proie. Ainsi le groupe ramène la proie au nid et réussit la tâche collective.

C'est l'un des exemples le plus abouti de tâches collectives faites avec des robots homogènes et autonomes. La tâche, comme le montre le tableau 6.5 [59, TABLE III] n'est faisable que par un nombre minimum de robots suivant la distance entre la proie et le nid. Un nombre supérieur de robots n'empêche pas la réussite de la tâche, mais n'augmente pas forcément l'efficacité. En effet, si on regarde le tableau 6.6 [59, TABLE V], le temps d'exécution de la tâche est essentiellement plus courte quand le nombre minimum de robots pour la réussite est disponible. En calculant l'accélération collective, elle est inférieure à 1 dans presque tous les cas. En moyenne, elle est d'une valeur de 0.54 et donc clairement sous-linéaire. Ainsi, les robots en surnombre ont plutôt tendance à gêner le groupe.

D/N	1	2	3	4	5	6	7	8	10	12
60	1	1	3	3	3	3	3	3	3	3
90	0	1	1	3	3	3	3	3	3	3
120	0	0	1	1	3	3	2	3	3	2
150	0	0	0	1	1	1	3	2	3	3
180			0	0	1	1	1	3	3	3
210				0	0	1	1	0	2	2
240						0	0	0	3	3

**Tableau 6.5** [59, TABLE III] Niveau global de réussite obtenu pour les configurations (N nombre de s-bot, D distance en cm) : sans succès (0), la formation d'une chaîne complète accomplie (1), l'accrochage au minimum de deux s-bots (2) et la retour au nid du groupe avec la proie (3).

D/N	1	2	3	4	5	6	7	8	10	12
60			317	247	327	395	301	194	113	245
90				123	394	255	271	223	616	178
120					411	766		791	276	
150							302		948	315
180								219	465	2606
210										
240									1314	915

**Tableau 6.6** [59, TABLE V] Temps (en s) pour accomplir la tâche complète suivant le distance (D en cm) et le nombre de robot (N)

### 6.2.2 Le scénario de Swarmanoid

- Dans le projet Swarmanoid<sup>4</sup>, trois groupes de robots vont être utilisés :
- Les marXbots munis du module d'accrochage et du module de communication "distance et relèvement", appelés pour l'occasion footbots.
  - Les handbots pouvant grimper et manipuler les objets.
  - Les eyebots qui sont des robots volants.

Le rôle des footbots est de transporter le handbot dans un environnement de type bureautique. Les eyebots vont explorer cet environnement en volant à la recherche de l'étagère où se trouvent des livres à ramener. Une fois l'étagère trouvée, les eyebots guident les footbots en formant des chaînes lumineuses un peu à la manière des s-bots. Les robots peuvent aussi utiliser le module "distance et relèvement" pour communiquer et se diriger.

Deux footbots transportent un handbot en le tirant. Une fois proche de l'étagère, le handbot est retourné et le groupe cherche à accrocher une pince du handbot à un montant de l'étagère. Quand le handbot s'est accroché à l'étagère, les footbots le lâchent et celui-ci peut lancer son système d'accrochage magnétique. Le handbot grimpe et attrape le livre. Pour redescendre, il utilise les ventilateurs pour contrôler son angle d'orientation. Le handbot tourne d'abord de 180° pour redescendre, ainsi le livre ne peut pas s'accrocher aux tabliers de l'étagère. Juste avant de toucher le sol, il se réoriente face à l'étagère, pour que les footbots puissent s'auto-assembler. Le handbot récupère son système d'accrochage magnétique et le groupe peut repartir, guidé par les eyebots. La figure 6.16 présente le déroulement en images.

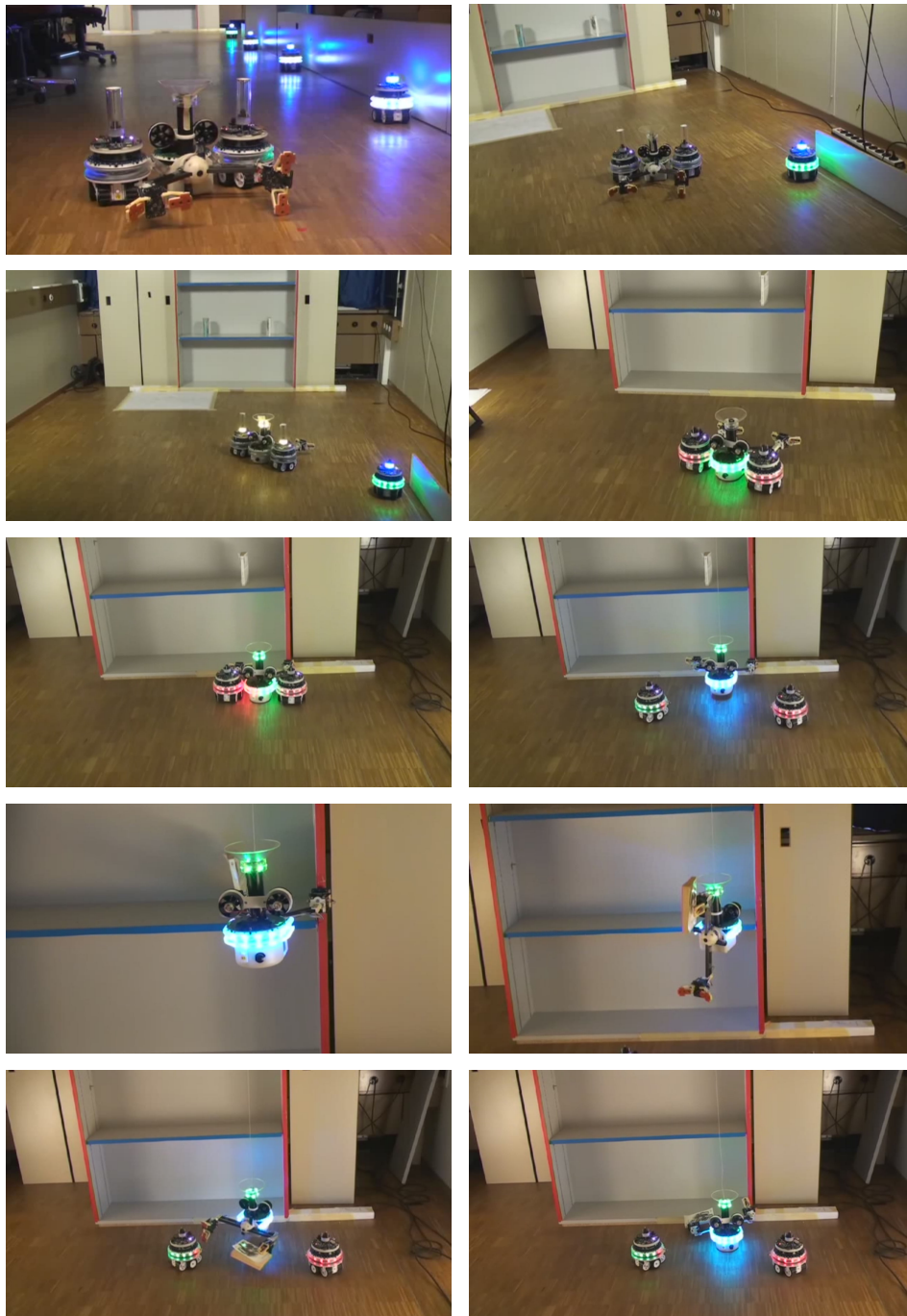
## 6.3 Conclusion

Si les performances super-linéaires sont un phénomène bien connu en robotique collective, notre expérience apporte un nouvel élément montrant ces propriétés dans des tâches physiques utilisant les capacités d'auto-assemblage des robots. Ceci est un aspect clé de notre approche et peut changer radicalement la façon de concevoir un système robotique pour des tâches telles que la navigation tout-terrain ou le transport d'un objet.

Les choix de conception du s-bot influencent clairement les performances individuelles et en groupe pour la navigation tout-terrain. Les petits détails de conception ont un impact encore plus grand sur la performance du système collectif, si ces performances sont amplifiées de façon super-linéaires.

---

4. <http://www.swarmanoid.org/>



**Figure 6.16** Séquence du scénario de Swarmanoid à la recherche d'un livre dans une étagère. Une description détaillée de la démarche se trouve dans le texte.

Le travail collectif n'induit pas forcément de super-linéarité. L'expérience de la section 6.2.1 du s-bot montre plutôt qu'il y a un optimum de nombre de robots pour effectuer une tâche, qui est également le minimum de robots nécessaires. Tout robot supplémentaire va gêner d'une certaine façon et diminuer la performance relative du groupe. La super-linéarité intervient dans nos expériences quand il y a "un changement de la nature du problème".<sup>5</sup> Ainsi, quand les s-bots et le marXbots tirent ensemble, la physionomie du groupe change le contact des chenilles au sol. Ce changement de la nature du problème est possible quand il y a une interaction physique et constructive entre les robots, ce qui se produit avec l'auto-assemblage.

Un autre moyen d'augmenter la performance du groupe est d'utiliser des robots hétérogènes. Ce type de collaboration permet une accélération très forte des performances. L'hétérogénéité améliore l'accélération et l'optimisation des sous-tâches comme pour le marXbot spécialisé au transport au sol et le handbot spécialisé dans la grimpe.

Avec l'hétérogénéité, on peut également avoir des performances super-linéaires sans auto-assemblage. Par exemple, un eyebot pourrait guider un marXbot dans un environnement totalement inconnu bien plus rapidement car sa position lui permet d'avoir une vue d'ensemble et lointaine du monde, ce que ne peut pas avoir le marXbot au sol.

Cependant, les interactions physiques produisant des performances super-linéaires ne sont pas extensibles ou "scalable". Lorsque le nombre optimum est dépassé, les performances diminuent car les individus supplémentaires n'apportent plus ou peu de changements à la nature du problème. En outre, les contraintes physiques des interactions ne sont pas extensibles. Il y aura toujours un moment où la force additionnée des robots va rompre un élément mécanique. En effet, on a souvent dû arrêter les expériences car les systèmes d'accrochage se cassaient simplement sous la force globale du groupe.

---

5. Philippe Rétornaz, discussion informelle





## 7 Comparaison des quatre robots

### 7.1 Comparatif des caractéristiques et coûts

Voici un tableau résumant les caractéristiques des quatre robots :

	e-puck	s-bot	marXbot	handbot
Degré de liberté	2	5 <sup>1</sup>	4	10
Flux de données des microcontrôleurs	715 kbit/s	20.8 kbit/s	136 kbit/s	3363.8 kbit/s
Nombre des microcontrôleurs	1	13	10	7
Puissance de calcul des microcontrôleurs <sup>2</sup>	15 MIPS	30 MIPS	400 MIPS	280 MIPS
Flux de données du processeur central	-	2.3Mbit/s	307.5Mbit/s	307Mbit/s
Puissance de calcul du processeur	-	533 MIPS	740 MIPS	740 MIPS
Communication	Bluetooth <sup>®</sup> Son LED IR	Wifi, Son, LED	Wifi, Son, LED, D&R <sup>3</sup>	Wifi, LED, D&R
Modulaire	oui	non	oui	non
Prix	353.-	2955.-	3610.-.-	4033.-
Part mécanique	5%	4%	17%	24%
Part moteur	7%	22%	17%	33%
Part électronique	88%	74%	66%	43%
Main d'œuvre	1/2 jour	5 sem.	1 sem.	4 sem.

#### Degrés de liberté

Les degrés de liberté sont très différents d'un robot à l'autre. L'e-puck n'a que deux degrés et il ne peut pas s'auto-assembler sous cette forme. Il faudrait y ajouter deux à trois degrés de liberté avec une extension pour

1. Les degrés de la pince mobile ne sont pas comptabilisés car elle a été peu utilisée

2. Rapporté à 16 bits

3. Module Distance et Relèvement

que l'assemblage soit réalisable. Le s-bot et le marXbot peuvent s'auto-assembler et ont ainsi les degrés de liberté nécessaires en plus. Le s-bot a un degré de liberté supplémentaire ce qui lui confère un avantage pour franchir les obstacles, comme présenté dans la section 6.1.1. Finalement, le handbot a 10 degrés de liberté qui sont utilisés pour la grimpe et la manipulation d'objets. Néanmoins, cette complexité accrue lui permet également de s'auto-assembler et ainsi d'accroître ses performances physiques. Par conséquent, on peut observer que la complexité des individus permet de faciliter la collaboration en groupe.

### **Flux d'informations**

Le flux de données pour le seul microcontrôleur de l'e-puck est important, cependant, avec 15 MIPS il ne peut pas traiter toute l'information en même temps. Néanmoins, la grande variété des capteurs peut déjà rendre l'e-puck de base intéressant pour différentes expériences collectives. De plus l'e-puck peut accepter une extension pour accroître ses performances.

Les architectures de l'électronique du s-bot, du marXbot et du handbot sont très semblables. Ils ont chacun des microcontrôleurs dispersés en charge des capteurs et actionneurs de bas niveau. Le s-bot par contre ne dispose pas de la puissance de calcul pour traiter l'information dans les microcontrôleurs. Ainsi le processeur principal doit traiter l'information lui-même. Les avancées technologiques ont permis de considérablement augmenter la puissance de calcul des microcontrôleurs et par conséquent le marXbot et le handbot bénéficient du décuple de la puissance. Ce regain de puissance est également utilisé pour distribuer le comportement dans tout le robot. Le marXbot a plus de microcontrôleurs que le handbot car il est modulaire et certains microcontrôleurs ne sont pas utilisés de façon optimale contrairement au handbot où l'électronique est moins répartie. Le grand flux de données du handbot est essentiellement dû aux caméras incluses dans les pinces.

Le processeur central des marXbots et handbots et leurs caméras ont été améliorés par rapport au s-bot. Ces améliorations sont surtout techniques mais elles ont été effectuées car elles limitaient les capacités du s-bot. Dès lors, les marXbots et handbots peuvent faire des tâches beaucoup plus compliquées et dans des environnements moins maîtrisés.

### **Communication**

Les capacités de communication des quatre robots sont assez comparables. Chacun a une communication radio de haut débit (Bluetooth<sup>®</sup> et Wifi) essentiellement utilisée comme interface utilisateur. Ils ont également

une communication sonore, qui est une communication globale pouvant transmettre très peu d'informations (quelques fréquences). Le e-puck et le s-bot peuvent éventuellement calculer la direction de l'émetteur. Les robots peuvent également utiliser leurs LED pour signaler leur position et transmettre de l'information (différentes couleurs et des informations spatiales). Sur le s-bot, footbot et handbot le disque lumineux peut aussi indiquer au partenaire où s'auto-assembler. Cependant, le e-puck et le handbot n'ont pas de caméra omnidirectionnelle<sup>1</sup>, ainsi la communication est limitée à leur champ de vision. Le s-bot ne peut que voir ses pairs jusqu'à 60cm et le marXbot jusqu'à 110cm, ainsi la communication lumineuse est locale et unidirectionnelle. Malgré tout, cette façon de communiquer est très utile et spécifique à des robots collectifs.

Le dernier moyen de communication est le module distance et relèvement du marXbot et handbot. L'e-puck peut utiliser ses capteurs infrarouges mais seulement jusqu'à 30cm et le débit de données est limité. Le module du marXbot et du handbot peut permettre de distinguer un autre robot jusqu'à 6m, et de le positionner dans le plan jusqu'à 5m. Ce module peut également transmettre des informations (16 bytes). Le plus intéressant est que l'on peut identifier la position et l'identité de l'émetteur et ainsi cibler la communication spatialement ou nominativement.

La communication est primordiale pour des robots collectifs. Il ne s'agit pas tant de la quantité de données qui est importante mais plutôt de la possibilité de la transmettre localement, ou de façon spatiale et même nominative. Ces moyens de communication nécessitent le plus souvent du matériel spécifique et donc un coût supplémentaire.

### Coûts du collectif

Le coût de fabrication des divers robots suscite les observations suivantes. Avoir des robots collectifs pouvant avoir une interaction physique, a un coût important. Même si l'e-puck fabriqué en masse est 10%<sup>2</sup> moins cher à la fabrication, si on lui ajoute la capacité d'auto-assemblage pour le rendre collectif il faudrait presque doubler son prix. En effet, on aurait besoin de deux moteurs supplémentaires, d'un autre microcontrôleur, de quelques capteurs et d'une mécanique assez complexe. La fabrication en masse pour du collectif avec auto-assemblage le rendrait donc plus cher.

L'exemple modulaire du marXbot est également intéressant car les deux modules typiquement conçus pour les tâches collectives que sont le module d'accrochage et le module distance et relèvement, représentent 1/3

---

1. Il existe une extension caméra omnidirectionnelle et un anneau LED RGB pour l'e-puck

2. Différence entre 100 et 1000 pièces avec amortissement des moules et des frais fixes de l'électronique

des coûts (voir tableau 4.2). En ajoutant le prix de la caméra omnidirectionnelle et de la LED balise, qui n'ont que peu d'intérêt pour un robot unique, le coût de ces éléments représente le 50% du prix de revient. Les différents ajouts pour rendre le robot collectif font donc doubler son coût de fabrication.

### **Part de la mécanique, des moteurs et de l'électronique**

Une autre observation à faire sont les parts respectives de la mécanique, des moteurs et de l'électronique. La part de la mécanique de l'e-puck représente seulement 5% des coûts de revient. La mécanique de l'e-puck est peu chère parce qu'elle est simple et adaptée à la production de masse. Le marXbot a été également fabriqué avec les mêmes techniques de production que pour l'e-puck, mais il n'a été fabriqué qu'à 70 exemplaires et les coûts fixes ont encore une part importante. S'il avait été fabriqué à plus grande échelle, ces coûts seraient mieux amortis et ainsi la mécanique ne représenterait que 10% du prix total du marXbot.

On remarque que l'électronique est une part très importante pour tous les robots, pour l'e-puck elle est même de 88%. Ces coûts importants s'expliquent par le fait que par rapport à la mécanique, la production de l'électronique ne peut pas être adaptée en fonction du nombre de pièces. Le seul rabais qui peut être obtenu concerne le nombre de composants électroniques commandés. Ceux-ci restent, néanmoins, passablement coûteux, surtout en robotique, car les différents capteurs extrinsèques ne sont pas forcément des produits de grande consommation.

Les parts du s-bot sont ici légèrement faussées. En effet, le prix présenté pour la mécanique ne compte que la matière première et non pas les heures de fabrication effectuées par le consortium. La part de la mécanique doit plutôt s'approcher de 20-25%. Les pourcentages sont alors semblables à ceux du marXbot.

Pour le handbot, les temps de travail ne sont pas pris en compte ni pour l'électronique ni pour la mécanique, ainsi les parts peuvent être considérées comme correctes, sauf celle pour les moteurs, qui devrait donc être légèrement plus basse. Pour un robot aussi complexe mécaniquement que le handbot, il est clair que la part de l'électronique est plus basse. Aussi, les 13 moteurs nécessaires pour son fonctionnement ont un coût très peu compressible. Fabriquée en grande quantité, la part de la mécanique peut diminuer, en utilisant des techniques de production appropriées.

## 7.2 Conclusion

Les robots collectifs pouvant interagir physiquement, qui sont d'ailleurs les plus intéressants par leur performance super-linéaire, ont besoin de degrés de liberté supplémentaires. Leurs besoins en capteurs ne sont pas inférieurs à un robot unique et demandent même des capteurs spécifiques pour localiser leurs pairs. Pour interagir, ils ont également besoin de système de communication non standard. Finalement, un robot collectif pouvant s'auto-assembler n'est pas simple, au contraire il aura plutôt tendance à être complexe.

Somme tout, plus le robot est complexe, plus il demande de l'électronique et des moteurs pour le contrôler. Ce sont des parts des robots qui représentent de 85% à 90% des coûts. Même en grande quantité, on ne peut pas baisser considérablement leur prix. On peut le faire seulement sur la mécanique en investissant dans des techniques de production adaptées. Ainsi, même produit en très grande masse, on ne peut pas espérer diminuer fortement le coût d'un robot collectif évolué.



# 8 Conclusion

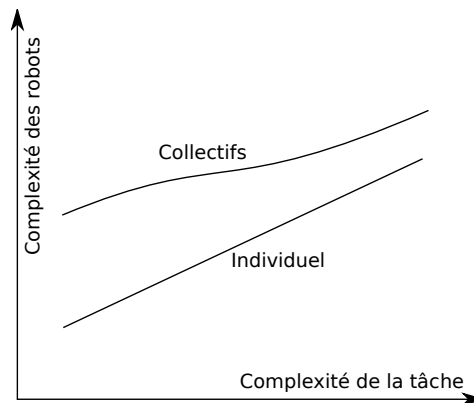
## 8.1 Conclusion

En robotique collective, il est très intéressant d'avoir des performances super-linéaires, car c'est ici que le collectif peut montrer un avantage par rapport à une approche avec un robot unique. Cependant, les performances super-linéaires adviennent plus facilement dans des cas impliquant des interactions physiques telles que l'auto-assemblage ou une collaboration hétérogène, changeant la nature du problème. Ces interactions physiques et la collaboration inter-individus demandent du matériel spécifique que ce soit au niveau mécanique ou électronique. L'auto-assemblage exige des degrés de liberté supplémentaires donc des pièces mécaniques additionnelles et des moteurs. La collaboration requiert des moyens de communication évolués et spécifiques, pouvant par exemple localiser et identifier l'émetteur, ce qui nécessite une électronique additionnelle. Tout ce matériel a un coût pouvant facilement doubler le prix d'un individu. Même produit en grande quantité, on ne peut pas réduire considérablement les coûts. En effet, les seuls coûts réductibles sont ceux liés à la mécanique qui a une part faible par rapport à l'électronique. Cela va à l'encontre des affirmations de certains spécialistes du domaine qui donnent un avantage aux robots collectifs pour leur faible coût car produits en masse.

Ces personnes prétendent également que les individus sont "simples", par rapport à un robot unique. Nous pensons au contraire qu'ils sont compliqués, avec probablement plus de degrés de liberté et de besoins matériels.

Cependant, un robot comme le handbot qui de base est complexe peut accroître ses performances sans adjonction de matériel spécifique en s'auto-assemblant avec ses pairs. Ceci n'était pas prévu à la base, néanmoins ses nombreux degrés de liberté ont permis de le faire. Ainsi un robot complexe aura plus facilement la capacité d'auto-assemblage, une aptitude apportant un grand gain en performance. La figure 8.1 exprime ce pressentiment, où avec l'augmentation de la complexité des tâches, la différence entre un robot individuel et des robots collectifs diminue.

Les gains en performance des robots collectifs ne sont pas extensibles. Il y aura toujours un moment où les propriétés physiques seront insuffisantes, ou alors que l'environnement ne peut contenir plus de robots.



**Figure 8.1** Comparaison des robots collectifs et individuels

Finalement, ces résultats devraient être une motivation supplémentaire pour développer des techniques de conception afin de mieux exploiter l'accélération collective depuis le début de la phase de conception en prédisant la performance du système collectif.

Nous sommes également convaincus que l'avantage des robots collectifs réside dans leur capacité à facilement s'adapter à une nouvelle tâche et à avoir plus de robustesse pour l'accomplir. Encore faut-il que les robots défaillants ne perturbent pas le groupe. Ces capacités ne sont pas encore démontrées et ouvrent des nouvelles questions pour la recherche en robotique collective.

## 8.2 Mes contributions à l'état de l'art

Ce travail apporte des contributions suivantes à l'état de l'art :

- J'ai développé l'e-puck qui a été un des premiers robots "open hardware" et un des plus diffusés dans le milieu académique.
- J'ai contribué au développement du s-bot et participé au projet Swarm-bots. Le s-bot a été le premier robot permettant des expériences d'auto-assemblages d'une certaine envergure ( $> 20$  robots).
- J'ai conçu le marXbot qui est l'un des robots modulaires de recherche les plus aboutis pour sa taille. Il a permis par des expériences d'auto-assemblage, de SLAM, d'interaction avec des poussins et d'expérience de construction.
- J'ai inventé le robot handbot qui est le premier robot capable de grimper et de rapporter un livre d'une étagère.



- Nous avons fait avec le marXbot et le handbot une des premières expériences de grande envergure avec des robots collectifs hétérogènes avec des interactions physiques.
- Dans cette thèse j'ai apporté des exemples qui permettent de supposer que les robots collectifs pouvant s'auto-assembler sont complexes et chers, contrairement aux prétentions du domaine.



# Bibliographie

- [1] A. ACERBI, D. MAROCCO et S. NOLFI : Social facilitation on the development of foraging behaviors in a population of autonomous robots. *LECTURE NOTES IN COMPUTER SCIENCE*, 4648:625, 2007.
- [2] C. ANDERSON, G. THERAULAZ et J. DENEUBOURG : Self-assemblages in insect societies. *Insectes Sociaux*, 49(2):99–110, 2002.
- [3] E. ANTONELLO, B. SCHRAUWEN et D. STROOBANDT : Event detection and localization for small mobile robots using reservoir computing. *Neural Networks*, 21(6):862–871, 2008.
- [4] E. BONABEAU, M. DORIGO et G. THERAULAZ : *Swarm Intelligence : From Natural to Artificial Systems*. Oxford University Press, 1999.
- [5] E. BONABEAU, M. DORIGO et G. THERAULAZ : Inspiration for optimization from social insect behaviour. *Nature*, 406(6791):39–42, 2000.
- [6] F. BONACCORSO, D. LONGO et G. MUSCATO : modelling of an innovative actuator for climbing robot adhesion. In O. TOSUN, H. L. AKIN, M. O. TOKHI et G. S. VIRK, édés : *Proceedings of 12th International conference on climbing and walking robots and the support technologies for mobile machines (CLAWAR)*, p. 891–898, Istanbul, 2009. World Scientific.
- [7] M. BONANI, V. LONGCHAMP, S. MAGNENAT, P. RÉTORNAZ, D. BURNIER, G. ROULET, F. VAUSSARD, H. BLEULER et F. MONDADA : The MarXbot, a Miniature Mobile Robot Opening new Perspectives for the Collective-robotic Research. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2010)*, 2010.
- [8] M. BONANI, S. MAGNENAT, P. RÉTORNAZ et F. MONDADA : The hand-bot, a robot design for simultaneous climbing and manipulation. *Intelligent Robotics and Applications*, p. 11–22, 2009.
- [9] M. BONANI, P. RÉTORNAZ, S. MAGNENAT, H. BLEULER et F. MONDADA : Physical interactions in swarm robotics : the hand-bot case study. In *Proceedings CFP of the 10th International Symposium on Distributed Autonomous Robotics Systems*. Springer, 2010.
- [10] B. BRIDGE, H. RODRIGUEZ, S. MONDAL et T. SATTAR : field trials of a cell of climbing cooperating robots for fast and flexible manufacturing of large scale engineering structures. In O. TOSUN, H. L. AKIN, M. O. TOKHI et G. S. VIRK, édés : *Proceedings of 12th International conference on climbing and walking robots and the support technologies for mobile machines (CLAWAR)*, p. 801–810, Istanbul, 2009. World Scientific.
- [11] A. CAI, T. FUKUDA, F. ARAI, T. UYAMA et A. SAKAI : Hierarchical control architecture for cellular robotic system-simulations and experiments. In *1995 IEEE International Conference on Robotics and Automation, 1995. Proceedings.*, p. 1191–1196, 1995.

- [12] G. CAPI, G. POJANI et S. KANEKO : Evolution of Task Switching Behaviors in Real Mobile Robots. In *Innovative Computing Information and Control, 2008. ICICIC'08. 3rd International Conference on*, p. 495–495, 2008.
- [13] A. CAPUA : Spiderbot- an underconstrained cable suspended robot. <http://www.youtube.com/watch?v=uBikHgnt16E>.
- [14] A. CASTANO, A. BEHAR et P. WILL : The Conro modules for reconfigurable robots. *IEEE/ASME transactions on mechatronics*, 7(4):403–409, 2002.
- [15] A. CASTANO, W. SHEN et P. WILL : CONRO : Towards deployable robots with inter-robots metamorphic capabilities. *Autonomous Robots*, 8(3):309–324, 2000.
- [16] A. L. CHRISTENSEN, R. O'GRADY et M. DORIGO : Morphology control in multirobot system. *IEEE Robotics and Automation Magazine*, 14(4):18–25, 2007.
- [17] B. CHU, K. JUNG, C.-S. HAN et D. HONG : A survey of climbing robots : Locomotion and adhesion. *International Journal of Precision Engineering and Manufacturing*, 11:633–647, 2010. 10.1007/s12541-010-0075-3.
- [18] C. CIANCI, J. NEMBRINI, A. PROROK et A. MARTINOLI : Assembly of Configurations in a Networked Robotic System : A Case Study on a Reconfigurable Interactive Table Lamp. In *IEEE Swarm Intelligence Symposium, 2008. SIS 2008*, p. 1–7, 2008.
- [19] C. CIANCI, X. RAEMY, J. PUGH et A. MARTINOLI : Communication in a swarm of miniature robots : The e-puck as an educational tool for swarm robotics. In *Swarm Robotics : Second Sab 2006 International Workshop, Rome, Italy, September 30-October 1, 2006 Revised Selected Papers*, p. 103. Springer Verlag, 2007.
- [20] N. CORRELL, C. CIANCI, X. RAEMY et A. MARTINOLI : Self-Organized Embedded Sensor/Actuator Networks for Smart Turbines. In *IROS 2006 Workshop : Network Robot System : Toward intelligent robotic systems integrated with environments*. Citeseer, 2006.
- [21] A. DAHBI, A. HEFETZ, X. CERDA et A. LENIOR : Trophallaxis Mediates Uniformity of Colony Odor in *Cataglyphis iberica* Ants (Hymenoptera, Formicidae). *Journal of Insect Behavior*, 12:559–567, 1999.
- [22] R. DAMOTO, A. KAWAKAMI et S. HIROSE : Study of super-mechano colony : concept and basic experimental set-up. *Advanced Robotics*, 15(4):391–408, 2001.
- [23] M. DORIGO : The swarmanoid project, 2007.
- [24] T. FUKUDA, M. BUSS, H. HOSOKAI et Y. KAWAUCHI : Cell structured robotic system CEBOT : control, planning and communication methods. *Robotics and autonomous systems*, 7(2-3):239–248, 1991.
- [25] T. FUKUDA et S. NAKAGAWA : A dynamically reconfigurable robotic system (Concept of a system and optimal configurations). In *1987 International Conference on Industrial Electronics, Control, and Instrumentation*, p. 588–595, 1987.
- [26] T. FUKUDA et S. NAKAGAWA : Approach to the dynamically reconfigurable robotic system. *Journal of Intelligent & Robotic Systems*, 1:55–72, 1988. 10.1007/BF00437320.

- 
- [27] T. FUKUDA et S. NAKAGAWA : Method of autonomous approach, docking and detaching between cells for dynamically reconfigurable robotic system CEBOT. *JSME INT. J.*, 33(2):263–268, 1990.
- [28] T. FUKUDA, S. NAKAGAWA, Y. KAWAUCHI et M. BUSS : Self organizing robots based on cell structures - cebot. *Intelligent Robots, 1988., IEEE International Workshop on*, p. 145–150, 31 Oct-2 Nov 1988.
- [29] T. FUKUDA et T. UEYAMA : *Cellular robotics and micro robotic systems*. World Scientific Pub Co Inc, 1994.
- [30] T. FUKUDA, T. UEYAMA et Y. KAWAUCHI : Self-organization in cellular robotic system(CEBOT) for space application with knowledge allocation method. *i-SAIRAS'90*, p. 101–104, 1990.
- [31] R. GROSS, M. BONANI, F. MONDADA et M. DORIGO : Autonomous self-assembly in swarm-bots. *IEEE Transactions on Robotics*, 22(6):1115–1130, 2006.
- [32] R. GROSS et M. DORIGO : Cooperative transport of objects of different shapes and sizes. In *Ant Colony, Optimization and Swarm Intelligence*, num. 3172/2004 de Lecture Notes in Computer Science, p. 106–117. Springer Berlin / Heidelberg, 2004.
- [33] R. GROSS et M. DORIGO : Self-assembly at the macroscopic scale. *Proceedings of the IEEE*, 96(9):1490–1508, sep. 2008.
- [34] R. GROSS, M. DORIGO et M. YAMAKITA : Self-assembly of mobile robots—from swarm-bot to super-mechano colony. In *Proceedings of the 9th International Conference on Intelligent Autonomous Systems*, p. 487–496. Citeseer, 2006.
- [35] A. GUTIERREZ, A. CAMPO, M. DORIGO, D. AMOR, L. MAGDALENA, F. MONASTERIO-HUELIN et S. MADRID : An Open Localisation and Local Communication Embodied Sensor. *Sensors*, 8:7545–7563, 2008.
- [36] Á. GUTIÉRREZ, E. TUCI et A. CAMPO : Evolution of neuro-controllers for robots' alignment using local communication. *International Journal of Advanced Robotic Systems*, 6(1), 2009.
- [37] T. HAYAKAWA, T. NAKAMURA et H. SUZUKI : development of a wave propagation type wall-climbing robot using a fan and slider cranks. In O. TOSUN, H. L. AKIN, M. O. TOKHI et G. S. VIRK, édés : *Proceedings of 12th International conference on climbing and walking robots and the support technologies for mobile machines (CLAWAR)*, p. 439–446, Istanbul, 2009. World Scientific.
- [38] S. HIROSE : Super mechano-system : New perspective for versatile robotic system. *Experimental Robotics VII*, p. 249–258, 2001.
- [39] A. J. IJSPEERT, A. MARTINOLI, A. BILLARD et L. M. GAMBARELLA : Collaboration through the exploitation of local interactions in autonomous collective robotics : the stick pulling experiment. *Autonomous Robots*, 11:149–171, 2001.
- [40] K. IOANNIDIS, G. SIRAKOULIS et I. ANDREADIS : A Cellular Automaton Collision-Free Path Planner Suitable for Cooperative Robots. In *Informatics, 2008. PCI'08. Panhellenic Conference on*, p. 256–260, 2008.
- [41] A. KAMIMURA, H. KUROKAWA, E. YOSHIDA, S. MURATA, K. TOMITA et S. KOKAJI : Automatic locomotion design and experiments for a modular robotic system. *IEEE/ASME TRANSACTIONS ON MECHATRONICS*, 10(3), 2005.

- [42] H. KIM, K. SEO, K. LEE, J. KIM et H. S. KIM : Development of a multi-body wall climbing robot with tracked wheel mechanism. In H. FUJIMOTO, M. O. TOKHI, H. MOCHIYAMA et G. S. VIRK, édés : *Proceedings of 13th International conference on climbing and walking robots and the support technologies for mobile machines (CLAWAR)*, p. 439–446, Istanbul, 2010. World Scientific.
- [43] J. KIM, K. LEE, T. KIM, D. LEE, S. LEE, C. LIM et S. KANG : Rail Running Mobile Welding Robot ‘RRX3’ for Double Hull Ship Structure. In *Proc. of the 17th World Congress the International Federation of Automatic Control*, p. 4292–4297, 2008.
- [44] N. KITAMURA, Y. NAKAMICHI et K. FUKUDA : Development of a desktop swarm robot system based on pheromone communication. *Artificial Life and Robotics*, 14(3):329–331, 2009.
- [45] A. LIONI, C. SAUWENS, G. THERAULAZ et J. DENEUBOURG : Chain formation in *Oecophylla longinoda*. *Journal of Insect Behavior*, 14(5):679–696, 2001.
- [46] S. MAGNENAT, V. LONGCHAMP, M. BONANI, P. RÉTORNAZ, P. GERMANO, H. BLEULER et F. MONDADA : Affordable SLAM through the Co-Design of Hardware and Methodology. In *Proceedings of the 2010 IEEE International Conference on Robotics and Automation*, p. 5395–5401. IEEE Press, 2010.
- [47] S. MAGNENAT, B. NORIS et F. MONDADA : Aseba-challenge : an open-source multiplayer introduction to mobile robots programming. In *Proceedings of International conference on Fun and Games*. Springer, 2008.
- [48] S. MAGNENAT, P. RÉTORNAZ, M. BONANI, V. LONGCHAMP et F. MONDADA : Aseba : a modular architecture for event-based control of complex robots. *IEEE Transactions on Mechatronics*, 2010. Accepted for publication.
- [49] S. MAGNENAT, P. RÉTORNAZ, B. NORIS et F. MONDADA : Scripting the swarm : event-based control of microcontroller-based robots. In *International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAN) 2008*, 2008. Workshop Proceedings ISBN : 978-88-95872-01-8.
- [50] A. MARTINELLI : Using the Distribution Theory to Simultaneously Calibrate the Sensors of a Mobile Robot. Research Report RR-6796, INRIA, 2009.
- [51] A. MARTINOLI : *Swarm Intelligence in Autonomous Collective Robotics : From Tools to the Analysis and Synthesis of Distributed Collective Strategies*. Thèse de doctorat, Swiss Federal Institute of Technology in Lausanne (EPFL), Lausanne, Switzerland, 1999.
- [52] M. MATARIC, M. NILSSON et K. SIMSARIAN : Cooperative multi-robot box-pushing. In *Proceedings of the 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems*, p. 556–561. IEEE Computer Society, 1995.
- [53] F. MONDADA, M. BONANI, A. GUIGNARD, S. MAGNENAT, C. STUDER et D. FLOREANO : Superlinear physical performances in a SWARM-BOT. *Advances in Artificial Life*, p. 282–291, 2005.
- [54] F. MONDADA, M. BONANI, X. RAEMY, J. PUGH, C. CIANCI, A. KLAPTOCZ, S. MAGNENAT, J. ZUFFEREY, D. FLOREANO et A. MARTINOLI : The e-puck, a robot designed for education in engineering. In *Proceedings of the 9th conference on autonomous robot systems and competitions*, vol. 1, p. 59–65, 2009.

- 
- [55] M. MONDADA et M. BONANI : e-puck the educational robot, 2006.
- [56] S. MURATA, K. KAKOMURA et H. KUROKAWA : Docking experiments of a modular robot by visual feedback. *In 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, p. 625–630, 2006.
- [57] S. MURATA, E. YOSHIDA, A. KAMIMURA, H. KUROKAWA, K. TOMITA et S. KOKAJI : M-TRAN : Self-reconfigurable modular robotic system. *IEEE/ASME transactions on mechatronics*, 7(4):431–441, 2002.
- [58] I. NAVARRO, Á. GUTIÉRREZ, F. MATÍA et F. MONASTERIO-HUELIN : An Approach to Flocking of Robots Using Minimal Local Sensing and Common Orientation. *In Proceedings of the 3rd international workshop on Hybrid Artificial Intelligence Systems*, p. 616–624. Springer, 2008.
- [59] S. NOUYAN, R. GROSS, M. BONANI, F. MONDADA et M. DORIGO : Teamwork in self-organized robot colonies. *Evolutionary Computation, IEEE Transactions on*, 13(4):695–711, aug. 2009.
- [60] S. NOUYAN, R. GROSS, M. DORIGO, M. BONANI et F. MONDADA : Group Transport Along a Robot Chain in a Self-Organised Robot Colony. *In 9th Conference on Intelligent Autonomous Systems (IAS)*, p. 433–442. IOS Press, 2006.
- [61] R. O’GRADY, R. GROSS, F. MONDADA, M. BONANI et M. DORIGO : Self-assembly on Demand in a Group of Physical Autonomous Mobile Robots Navigating Rough Terrain. *In 8th European Conference on Artificial Life (ECAL)*, p. 272–281. Springer, 2005.
- [62] T. OTANI et M. KOSHINO : Applying a path planner based on RRT to cooperative multirobot box-pushing. *Artificial Life and Robotics*, 13(2):418–422, 2009.
- [63] R. OUNG, F. BOURGAULT, M. DONOVAN et R. D’ANDREA : The Distributed Flight Array. *In IEEE Conference on Robotics and Automation (ICRA)*, p. 601–607, 2010.
- [64] R. OUNG, A. RAMEZANI et R. D’ANDREA : Feasibility of a Distributed Flight Array. *In Joint 48th IEEE Conference on Decision and Control (CDC) and 28th Chinese Control Conference*, p. 3028–3034, 2009.
- [65] D. PELEG : Distributed algorithms for systems of autonomous mobile robots. *In Principles of Distributed Systems*, num. 3974/2006 de Lecture Notes in Computer Science, p. 1. Springer Berlin / Heidelberg, 2006.
- [66] D. PETERS, G. and Pagano, D. LIU et K. WALDRON : A prototype climbing robot for inspection of complex ferrous structures. *In H. FUJIMOTO, M. O. TOKHI, H. MOCHIYAMA et G. S. VIRK, édés : Proceedings of 13th International conference on climbing and walking robots and the support technologies for mobile machines (CLAWAR)*, p. 150–156, Istanbul, 2010. World Scientific.
- [67] M. PRIETO, P. GONZÁLEZ, R. PONTICELLI, G. FERNÁNDEZ, J. SARRIA, J. GRIECO, R. FERNÁNDEZ, H. MONTES COTO, S. NABULSI, S. ROS *et al.* : On the design and development of climbing and walking robots for the maritime industries. *Journal of maritime research*, 2(1):9, 2005.
- [68] J. PUGH, X. RAEMY, C. FAVRE, R. FALCONI et A. MARTINOLI : A fast onboard relative positioning module for multirobot systems. *Mechatronics, IEEE/ASME Transactions on*, 14(2):151–162, 2009.

- [69] S. RAUT, A. PATWARDHAN et R. HENRY : Design and construction of glossy surface climbing biped robot. In H. FUJIMOTO, M. O. TOKHI, H. MOCHIYAMA et G. S. VIRK, édés : *Proceedings of 13th International conference on climbing and walking robots and the support technologies for mobile machines (CLAWAR)*, p. 423–430, Istanbul, 2010. World Scientific.
- [70] F. ROCHAT, P. SCHOENEICH, B. LÜTHI, F. MONDADA et H. BLEULER : Cymag3D : a simple and miniature climbing robot with advance mobility in ferromagnetic environment. In H. FUJIMOTO, M. O. TOKHI, H. MOCHIYAMA et G. S. VIRK, édés : *Emerging Trends in Mobile Robotics*, p. 383–391, Singapour, 2010. World Scientific. Best paper award "innovation award 2010 - for practical innovation in the field of robotics".
- [71] M. RUBENSTEIN, K. PAYNE, P. WILL et W. SHEN : Docking among independent and autonomous CONRO self-reconfigurable robots. In *IEEE International Conference on Robotics and Automation*, vol. 3, p. 2877–2882. Citeseer, 2004.
- [72] J. SEYFRIED, M. SZYMANSKI, N. BENDER, R. E. NA, M. THIEL, et H. WÖRN : The i-swarm project : Intelligent small world autonomous robots for micro-manipulation. In *Swarm Robotics*, num. 3342/2005 de Lecture Notes in Computer Science, p. 70–83. Springer Berlin / Heidelberg, 2005.
- [73] T. SHIMONE, D. KURABAYASHI, K. OKITA et T. FUNATO : Implementation of Formation Transition System using Synchronization in a Mobile Robot Group. In *Field and Service Robotics : Results of the 6th International Conference (STAR : Springer Tracts in Advanced Robotics Series Volume 42)*, vol. 42, p. 423–432. Springer, 2008.
- [74] W. M. SPEARS, D. F. SPEARS, J. C. HAMANN et R. HEIL : Distributed, physics-based control of swarms of vehicles. *Autonomous Robots*, 17:137–162, 2004.
- [75] V. SPERATI, V. TRIANNI et S. NOLFI : Evolving coordinated group behaviours through maximisation of mean mutual information. *Swarm Intelligence*, 2(2):73–95, 2008.
- [76] M. TARSITANO : Route selection by a jumping spider (*Portia labiata*) during the locomotory phase of a detour. *Animal Behaviour*, 72:1437–1442, 2006.
- [77] V. TRIFA, C. CIANCI et D. GUINARD : Dynamic Control of a Robotic Swarm using a Service-Oriented Architecture. In *Proceedings of International Symposium on Artificial Life and Robotics, Beppu, Japan*, 2008.
- [78] R. WEHNER : Desert ant navigation : how miniature brains solve complex tasks. *Journal of Comparative Physiology A : Neuroethology, Sensory, Neural, and Behavioral Physiology*, 189:579–588, 2003.
- [79] M. YIM, S. WEI-MIN, B. SALEMI, D. RUS, M. MOLL, H. LIPSON, E. KLAVINS et G. CHIRIKJIAN : Modular self-reconfigurable robot systems. *IEEE robotics and automation magazine*, 14:43–52, March 2007.
- [80] M. YIM, Y. ZHANG et D. DUFF : Modular robots. *Spectrum, IEEE*, 39(2):30–34, Feb 2002.
- [81] M. YIM, Y. ZHANG, K. ROUFAS, D. DUFF et C. ELDERSHAW : Connecting and disconnecting for chain self-reconfiguration with PolyBot. *IEEE/ASME transactions on mechatronics*, 7(4), 2002.



# Curriculum vitæ

Michael Bonani, né le 19 Juin 1980, nationalité Suisse et Italienne

## Formation

- 11.2006–2010 **PhD en robotique mobile** (prévu, octobre 2010)  
École Polytechnique Fédérale de Lausanne, Suisse.
- 2007–2008 Cours école doctorale
- Venture challenge@EPFL (2010)
  - Embedded systems (2010)
  - Tutoring Skills (2008)
  - Autonomous robots (2007)
- École Polytechnique Fédérale de Lausanne, Suisse.
- 1998–2003 **Master en microtechnique**
- Approfondissement : Technique de production
- École Polytechnique Fédérale de Lausanne, Suisse.
- 1995–1998 **Maturité scientifique**
- Prix d'excellence en mathématique
  - Prix de géométrie
- Gymnase de la Cité, Lausanne, Suisse.

## Expériences en recherche et professionnel

- 06.2006–2010 **Laboratoire des systèmes robotiques, EPFL, Suisse**  
Doctorat, Travail de recherche au sein du projet «Swarmanoid»
- 05.2005–05.2006 **Laboratoire de système autonome 2, EPFL, Suisse**  
Développement d'un robot éducatif et pluridisciplinaire pour divers cours à l'EPFL
- 09.2004–03.2005 **Université Libre de Bruxelles IRIDIA, Belgique**  
Travail de recherche au sein du projet «Swarmbots»
- 04.2003–12.2003 **Service civil au Laboratoire de système autonome 2, EPFL, Suisse**  
Travail de recherche au sein du projet «Swarmbots»
- Hiver 2002–2003 **Laboratoire de système autonome 2, EPFL, Suisse**  
Conception électronique de la perception du robot mobile «S-bot»

été 2002–2003	<b>Laboratoire de système autonome 2, EPFL, Suisse</b> Communication numérique sans fil pour «Slow indoor Flyer» avec technologie Bluetooth
hiver 2001–2002	<b>Laboratoire de système autonome, EPFL, Suisse</b> Conception d’un module pour le mini robot «Alice» permettant des interactions avec des insectes
2001	Participation au concours de robots de l’EPFL

### Publications

- [1] M. Bonani, P. Réornaz, S. Magnenat, H. Bleuler, and F. Mondada. Physical interactions in swarm robotics : the hand-bot case study. In *Proceedings of 10th International Symposium on Distributed Autonomous Robotics Systems (DARS 2010)*. 2010. To appear.
- [2] M. Bonani, V. Longchamp, S. Magnenat, P. Rétornaz, D. Burnier, G. Roulet, H. Bleuler, and F. Mondada. The marxbot, a miniature mobile robot opening new perspectives for the collective-robotic research. In *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2010)*. 2010. To appear.
- [3] F. Rochat, P. Schoeneich, M. Bonani, S. Magnenat, and F. Mondada. Design of magnetic switchable device and applications in climbing robots. In *Climbing And Walking Robots : Proceedings of the 13th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines*. 2010. To appear.
- [4] S. Magnenat, P. Schoeneich, F. Rochat, P. Rétornaz, M. Bonani, V. Longchamp, M. Voelkle, T. Barras, D. Burnier, P. Noirat, T. Baaboura, F. Vaussard, and F. Mondada. Autonomous Construction by a Mobile Robot in Unknown Environments with Scarce Resources. Presented at CogSys 2010, ETH Zurich, Switzerland, 2010.
- [5] S. Magnenat, P. Rétornaz, M. Bonani, V. Longchamp, and F. Mondada. ASEBA : A Modular Architecture for Event-Based Control of Complex Robots. *IEEE/ASME Transactions on Mechatronics*, 2010. To appear.
- [6] S. Magnenat, V. Longchamp, M. Bonani, P. Rétornaz, P. Germano, Bleuler, Hannes,, and F. Mondada. Affordable SLAM through the Co-Design of Hardware and Methodology. In *Proceedings of the 2010 IEEE International Conference on Robotics and Automation*. IEEE Press, 2010. To appear.
- [7] M. Bonani, S. Magnenat, P. Rétornaz, and F. Mondada. The Hand-bot, a Robot Design for Simultaneous Climbing and Manipulation. In M. Xie et al., editor, *Proceedings of the Second International Conference on Intelligent Robotics and Applications*, volume 5928 of *Lecture Notes in Computer Science*, pages 11–22. Springer, Berlin / Heidelberg, 2009.
- [8] R. O’Grady, C. Pinciroli, G. Roderich, A. L. Christensen, F. Mondada, M. Bonani, and M. Dorigo. Swarm-bots to the Rescue. In *Proceedings of the 10th European Conf. on Artificial Life, ECAL 2009*, *Lecture Notes in Computer Science*. Springer Verlag, Berlin, Germany, 2009.

- 
- [9] R. Groß, S. Magnenat, L. Kächler, V. Massaras, M. Bonani, and F. Mondada. Towards an Autonomous Evolution of Non-Biological Physical Organisms. In *Proceedings of the 10th European Conference on Artificial Life*, Lecture Notes in Computer Science. Springer, Berlin / Heidelberg, 2009.
- [10] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J.-C. Zufferey, D. Floreano, and A. Martinoli. The e-puck, a Robot Designed for Education in Engineering. In *Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions*, volume 1, pages 59–65. IPCB : Instituto Politécnico de Castelo Branco, 2009.
- [11] S. Nouyan, R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Teamwork in Self-Organized Robot Colonies. *IEEE Transactions on Evolutionary Computation*, volume 13, pages 695–711, 2009.
- [12] R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Autonomous Self-assembly in a Swarm-bot. In *Proc. of the 3rd Int. Symp. on Autonomous Minirobots for Research and Edutainment, AMiRE 2005*, pages 314–322. Springer Verlag, Berlin, Germany, 2006.
- [13] R. Groß, S. Nouyan, M. Bonani, F. Mondada, and M. Dorigo. Division of Labour in Self-Organised Groups. In *Proc. of the 10th Int. Conf. on Simulation of Adaptive Behavior*, number 426-436 in Lecture Notes in Artificial Intelligence. Springer Verlag, Berlin, Germany, 2008.
- [14] R. O’Grady, R. Groß, A. L. Christensen, F. Mondada, M. Bonani, and M. Dorigo. Performance Benefits of Self-Assembly in a Swarm-Bot. In *Proc. of the 2007 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 2381–2387. IEEE Computer Society Press, Los Alamitos, 2007.
- [15] S. Nouyan, R. Groß, M. Dorigo, M. Bonani, and F. Mondada. Group Transport Along a Robot Chain in a Self-Organised Robot Colony. In *Proc. of the 9th Int. Conf. on Intelligent Autonomous Systems*, pages 433–442. IOS Press, Amsterdam, The Netherlands, 2006.
- [16] G. Baldassarre, V. Trianni, M. Bonani, F. Mondada, M. Dorigo, and S. Nolfi. Self-Organised Coordinated Motion in Groups of Physically Connected Robots. *IEEE Transactions on Systems, Man and Cybernetics Part B : Cybernetics*, volume 37, pages 224–239, 2007. Sponsor : swarm-bots, OFES 01-0012-1.
- [17] E. Tuci, R. Groß, V. Trianni, F. Mondada, M. Bonani, and M. Dorigo. Cooperation through self-assembly in multi-robot. *ACM Transactions on Autonomous and Adaptive Systems*, volume 1, pages 115–150, 2006.
- [18] R. Groß, M. Bonani, F. Mondada, and M. Dorigo. Autonomous Self-assembly in Swarm-Bots. *IEEE Transactions on Robotics*, volume 22, pages 1115–1130, 2006. Sponsor : swarm-bots, OFES 01-0012-1.
- [19] M. Dorigo, E. Tuci, V. Trianni, R. Groß, S. Nouyan, C. Ampatzis, T. H. Labella, R. O’Grady, M. Bonani, and F. Mondada. SWARM-BOT : Design and Implementation of Colonies of Self-Assembling Robots. In *Computational Intelligence : Principles and Practice*. IEEE Computational Intelligence Society, New York, 2006.
- [20] N. Scheidegger, F. Mondada, M. Bonani, and R. Siegwart. Bi-pedal Robot for Rescue Operations. In *CLAWAR 2006*, pages 425–430. 2006.

- [21] R. Groß, E. Tuci, M. Dorigo, M. Bonani, and F. Mondada. Object Transport by Modular Robots that Self-assemble. In *Proc. of the 2006 IEEE Int. Conf. on Robotics and Automation*, pages 2558–2564. IEEE Computer Society Pres, Los Alamitos, CA, 2006.
- [22] R. O’Grady, R. Groß, F. Mondada, M. Bonani, and M. Dorigo. Self-assembly on Demand in a Group of Physical Autonomous Mobile Robots Navigating Rough Terrain. In *Proc. of the 8th European Conf. on Artificial Life, ECAL 2005*, Lecture Notes in Artificial Intelligence, pages 272–281. Springer Verlag, Berlin, Germany, 2005. Lecture Notes in Artificial Intelligence LNAI3630 Sponsor : swarm-bots, OFES 01-0012-1.
- [23] F. Mondada, M. Bonani, A. Guignard, S. Magnenat, C. Studer, and D. Floreano. Superlinear Physical Performances in a SWARM-BOT. In *Proceedings of the VIIIth European Conference on Artificial Life*, Lecture Notes in Artificial Intelligence, pages 282–291. Springer, Berlin / Heidelberg, 2005.
- [24] F. Mondada, M. Bonani, S. Magnenat, A. Guignard, D. Floreano, F. Groen, N. Amato, A. Bonari, E. Yoshida, and B. Kräse. Physical connections and cooperation in swarm robotics. In *8th Conference on Intelligent Autonomous Systems (IAS8)*, pages 53–60. 2004.
- [25] F. Mondada, A. Guignard, M. Bonani, D. Bär, M. Lauria, and D. Floreano. SWARM-BOT : From Concept to Implementation. In *International Conference on Intelligent Robots and Systems (IROS’2003)*, pages 1626–1631. IEEE Press, 2003.