Projet Neubot : Framework pour la simulation de robots modulaires et auto-organisation de la locomotion sous-marine

Barthélémy von Haller

Travail de Master

Le 7 mars 2005

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Buts du projet

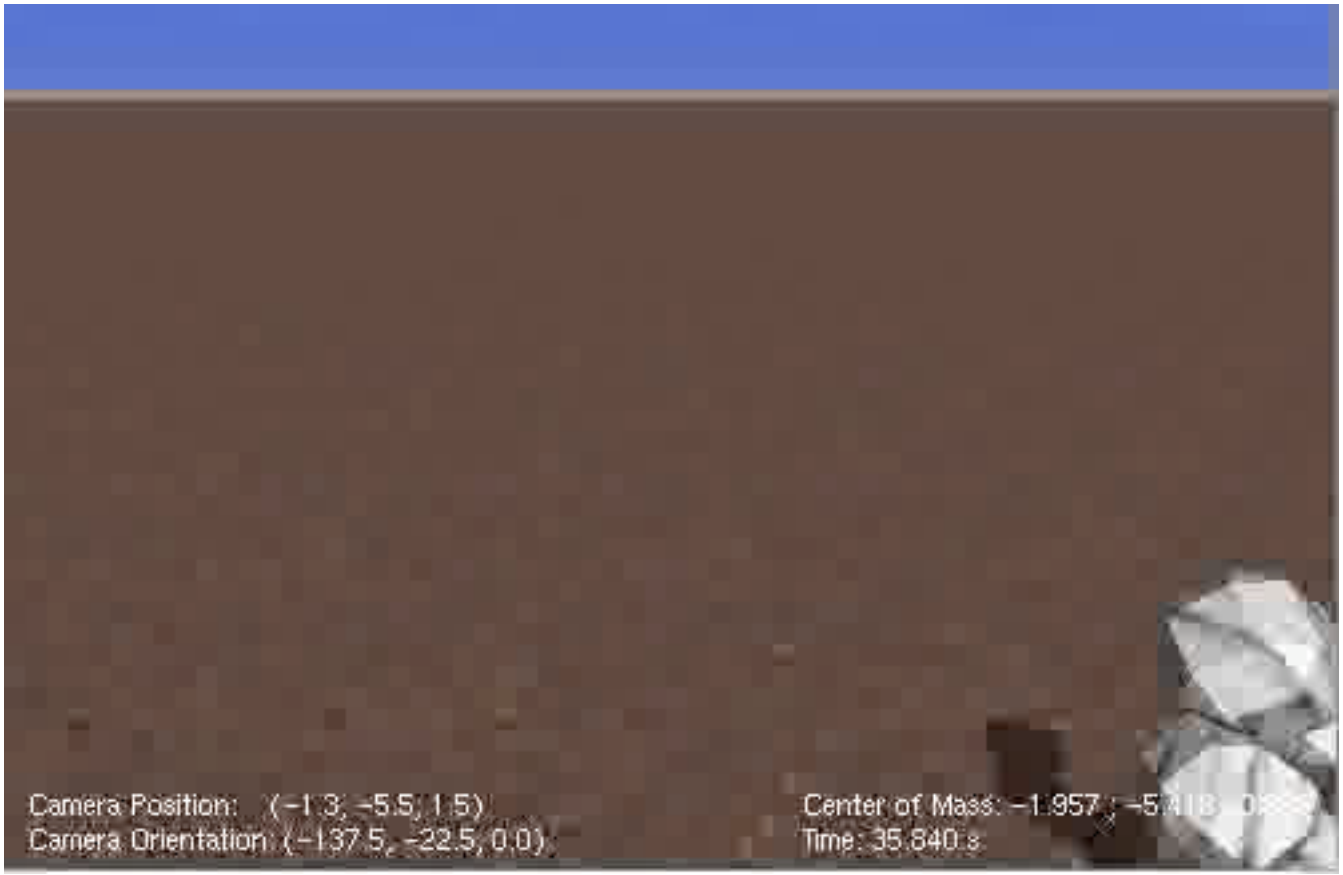
BIOLOGICALLY INSPIRED ROBOTICS GROUP (BIRG)

- Developper un framework pour la simulation de robots modulaires
- Utiliser ce framework pour faire un simulateur pour les modules Neubot
- Auto-organisation de la locomotion
  - Evolution d'un RN générant des oscillations
  - Utilisation de ce RN dans un CPG
  - Vérification de la validité de cette approche pour le cas particulier Neubot
  - Co-évolution du corps et du contrôleur

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

But à atteindre

BIOLOGICALLY INSPIRED ROBOTICS GROUP (BIRG)



Plan

- Introduction
  - Robotique modulaire
  - Projet Neubot
- Partie 1 : simulateur
  - Framework
  - Simulation des robots modulaires Neubot
- Partie 2 : auto-organisation de la locomotion
  - CPG constitué d'oscillateurs faits d'un réseau de neurones
  - Evolution du contrôleur de robots dont la forme est fixe
  - Co-évolution du contrôleur et du corps
- Conclusion et questions

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

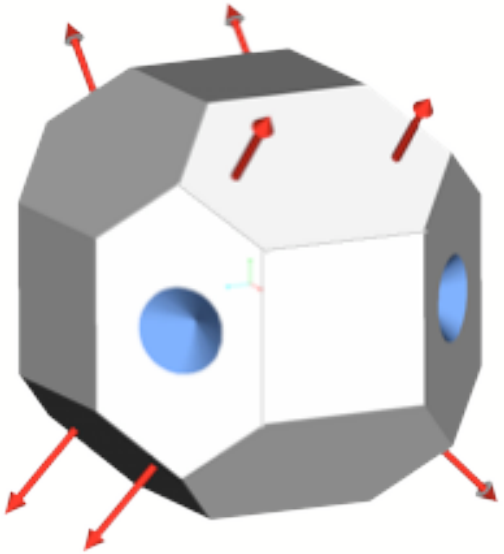
Robotique modulaire I

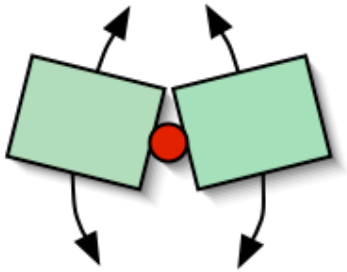
BIOLOGICALLY INSPIRED ROBOTICS GROUP (BIRG)

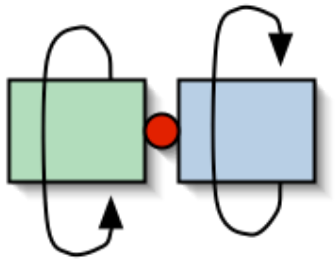
- Robots faits de multiples unités simples, souvent identiques
- Buts :
  - Polyvalence et adaptivité (grâce à l'auto-reconfiguration)
  - Robusteté (grâce à l'auto-réparation)
- La plupart des projets est terrestre.

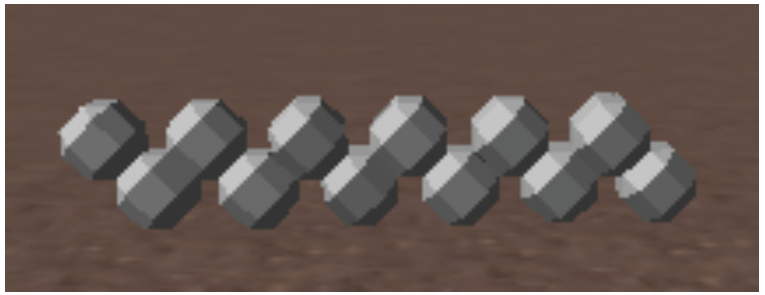
Le projet Neubot

- Collaboration LSL-ASL
- Von Neumann Robot -> hétérogénéité
- Milieu aquatique
- Auto-organisation
- Auto-reconfiguration
- Propulsion par jets
- Joints magnétiques
- Axe de rotations !





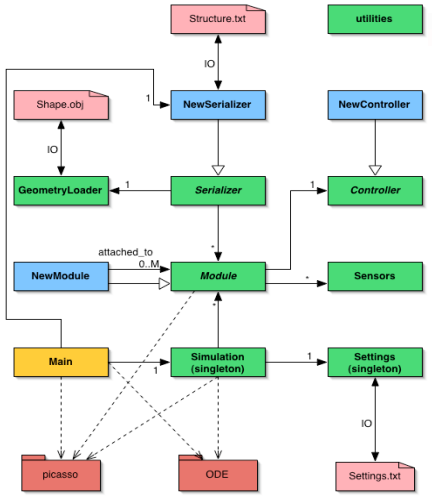




Plan

- Introduction
  - Robotique modulaire
  - Projet Neubot
- **Partie 1 : simulateur**
  - Framework
  - Simulation des robots modulaires Neubot
- Partie 2 : auto-organisation de la locomotion
  - CPG constitué d'oscillateurs faits d'un réseau de neurones
  - Evolution du contrôleur de robots dont la forme est fixe
  - Co-évolution du contrôleur et du corps
- Conclusion et questions

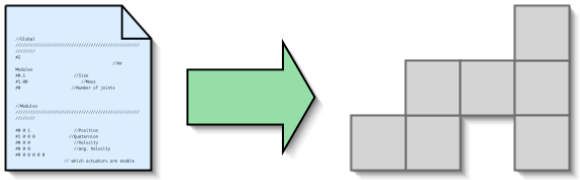
Framework



- Implémenté avec ODE
  - dynamique de corps solides
  - Frictions, collisions
- Le plus général possible
  - Cycles autorisés
  - Auto-reconfiguration / auto-organisation autorisées
  - Implémentation aisée de nouveaux modules et hétérogénéité
- Facilités pour stocker la forme des modules, la structure des robots et les paramètres dans des fichiers lisibles pour un humain

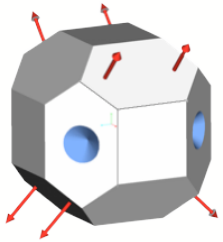
Simulateur pour Neubot

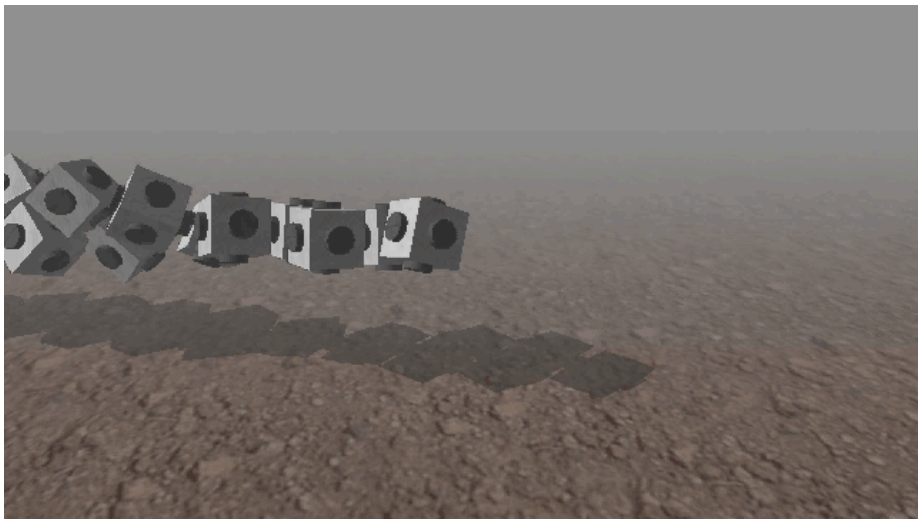
- Niveau de détails choisi
  - Ni trop précis (-> lent) ni pas assez (-> transfert vers le monde réel impossible)
  - Pas de prototype réel du module -> tous les coefficients sont estimés mais devront être déterminés expérimentalement par la suite
- Scripts décrivant la structure du robots facilement réalisables à la main
  - basé sur le module et non sur la structure



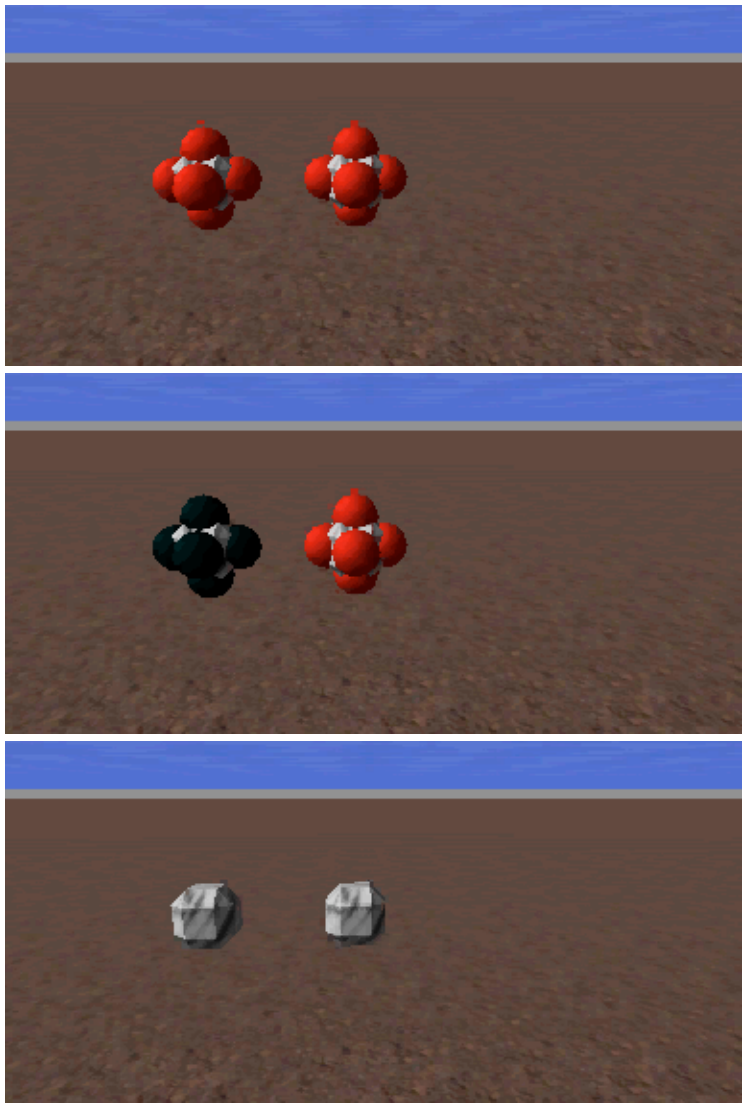
Simulateur pour Neubot

- Modèle physique de l'eau
  - Simplifications fortes
  - 3 types de force : poussée (archimède), inertielle et visqueuse
  - Forces appliquées lors de translations et de rotations
- Forme des modules
  - D'abord : cube
  - Ensuite : vraie forme





Simulateur pour Neubot



- Joints, aimants
  - Chaque aimant est simplifié par une sphère
  - Pour limiter les calculs, nous ne calculons les forces magnétiques que quand deux sphères se touchent

$$F_{magnet} = \frac{1}{\mu}\frac{C_1 C_2}{d^2}$$

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

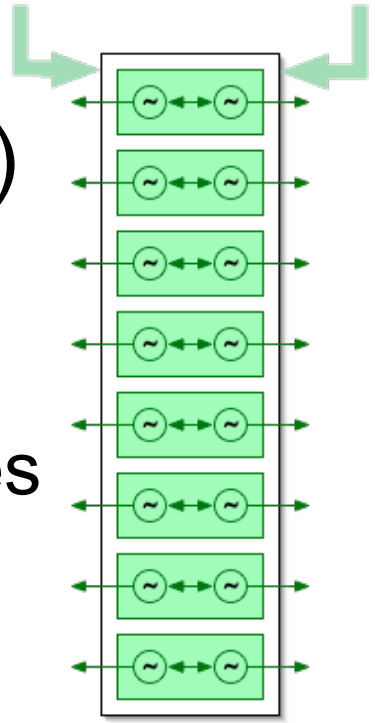
Plan

BIOLOGICALLY INSPIRED ROBOTICS GROUP (BIRG)

- Introduction
  - Robotique modulaire
  - Projet Neubot
- Partie 1 : simulateur
  - Framework
  - Simulation des robots modulaires Neubot
- **Partie 2 : auto-organisation de la locomotion**
  - CPG constitué d'oscillateurs faits d'un réseau de neurones
  - Evolution du contrôleur de robots dont la forme est fixe
  - Co-évolution du contrôleur et du corps
- Conclusion et questions

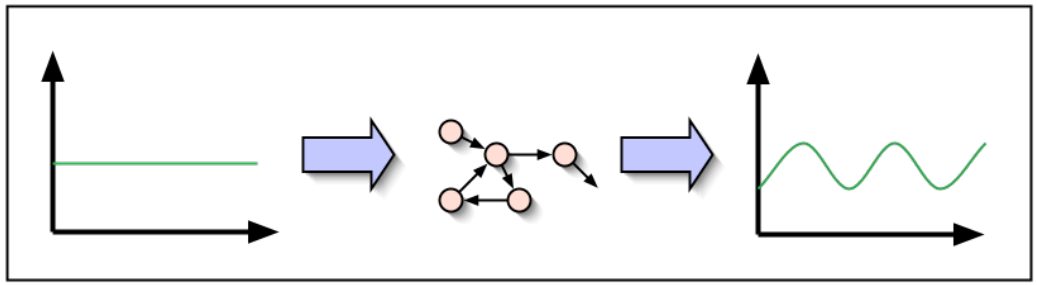
Auto-Organisation de la locomotion

- Approche bio-inspirée:
  - Central Pattern Generator (CPG)
    - réseaux de neurones comme oscillateurs
    - Synaptic spreading pour coupler les oscillateurs
  - Algorithme génétique (GA)
  - Co-évolution



Rappel CPG

- Réseaux de neurones générant des signaux oscillatoires à partir d'une simple entrée tonique venant du cerveau (BS = brainstem), ces oscillateurs sont couplés -> déphasage



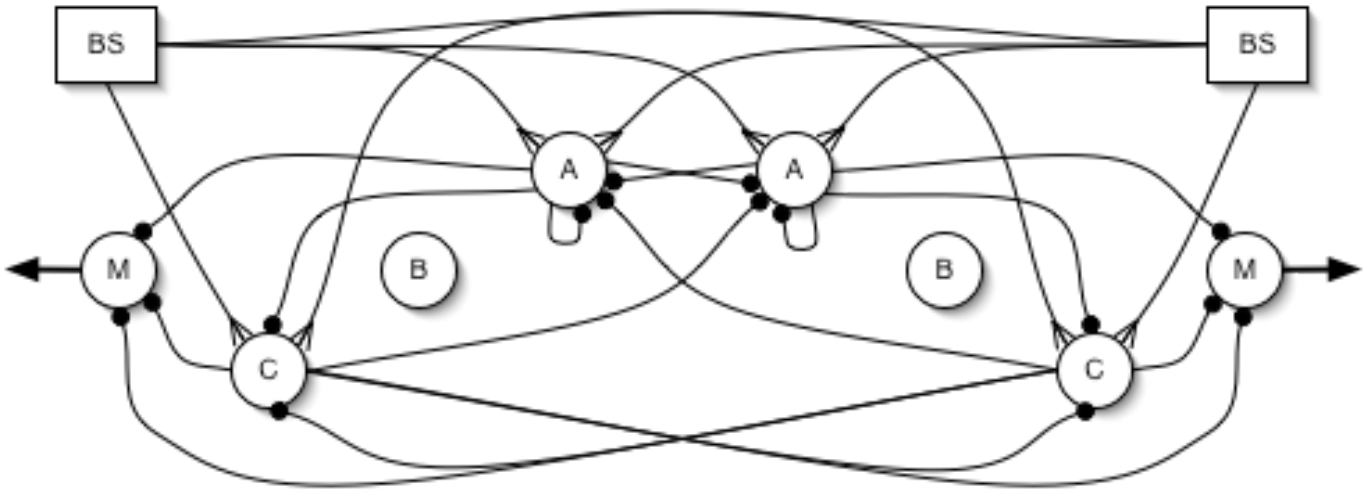
- Capacité à générer des "motifs" différents en fonction de l'entrée
- Garantit des transitions douces entre les démarches (marche, trot, gallop) et une prise en compte des senseurs dans le signal de sortie

Oscillateurs : description

$$\tau \frac{dm_i}{dt} = -m_i + \sum_j \omega_{i,j} x_j$$

$$x_i = (1 + e^{-(m_i + b_i)})^{-1}$$

Neurones de complexité moyenne modélisés comme des "Leaky Integrators"



Encodage : 43 gènes (grâce à la symétrie)

Oscillateurs : description : GA

GA : steady-state

Operateurs : crossover, mutation (gaussienne), pruning

Fonction de fitness : produit de 6 facteurs

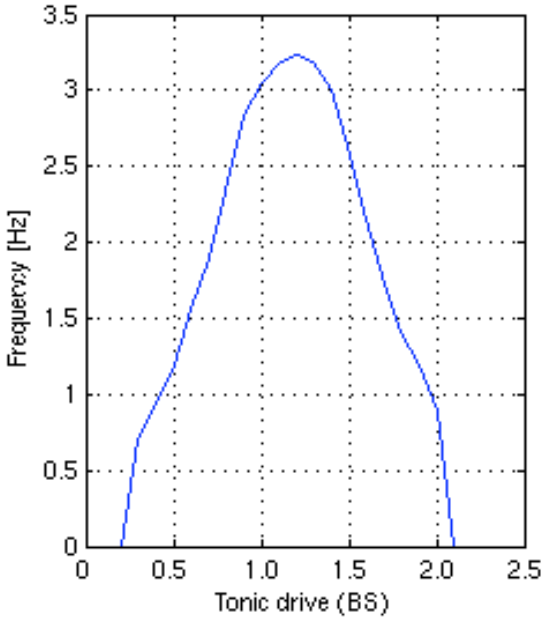
But : produire des oscillations régulières, dont la fréquence et l'amplitude peuvent être variées avec le BS et dont le nombre de connections internes est aussi faible que possible.

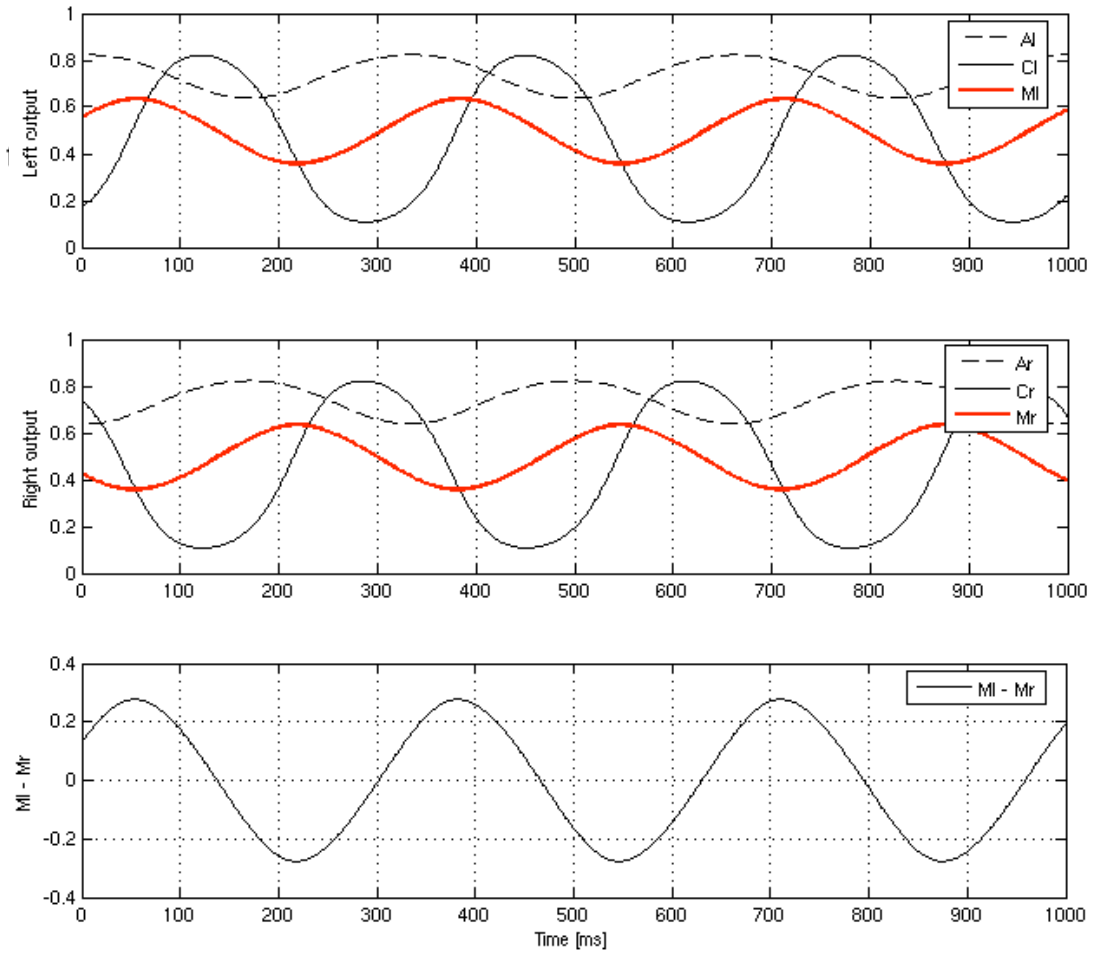
Function	Variable	Bounds $[bad, good]$
fit_nb_connect	number of connections	[24, 8]
fit_oscil	number of optima	[0, 20]
fit_regularity	SD of periods	[0.25, 0.0]
fit_oscil_phase	$\lvert\theta_{oscil} - 0.5\rvert$	[0.5, 0.0]
fit_freq_range	max_freq / min_freq	[1.0, 15.0]
fit_ampl	mean of amplitude	[0, 0.1]

Oscillateurs : résultats

- 15 évolutions
  - Toutes ont convergés vers des solutions correctes
- 1 très bonne solution très proche de celle trouvée par Auke Ijspeert
  - 1 - 9.3 Hz
- 1 moins bonne mais qui ne nécessite que 8 connexions internes au lieu de 14
  - 0.8 - 3.3 Hz
  - Le neurone de type B devient inutile

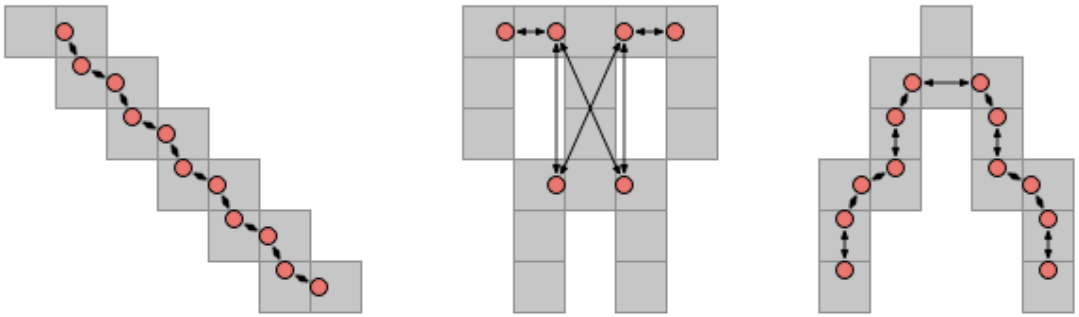
Oscillateurs : résultats





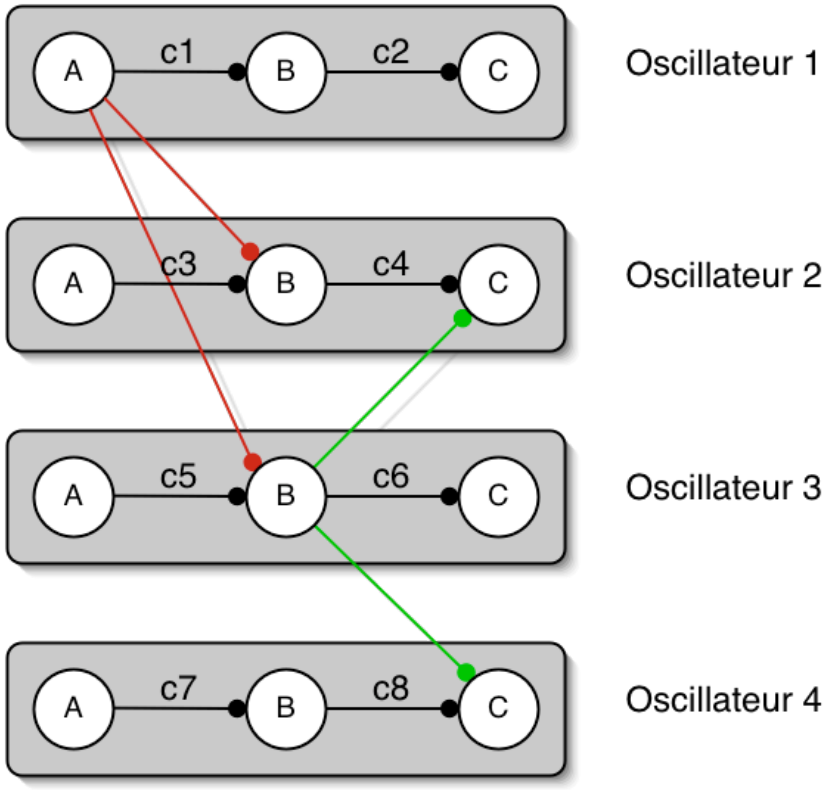
Locomotion

- Evolution uniquement du contrôleur
- 3 formes choisies



Locomotion : Méthode

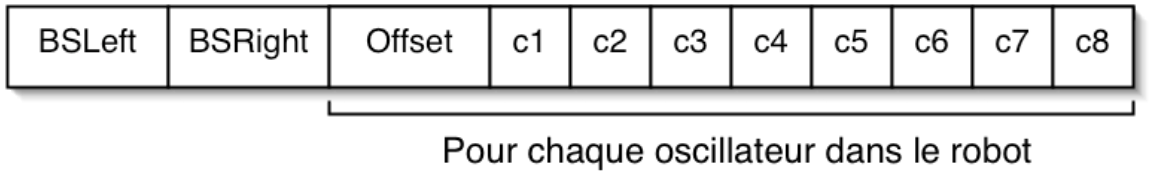
- Méthode : couplage des oscillateurs par **synaptic spreading**

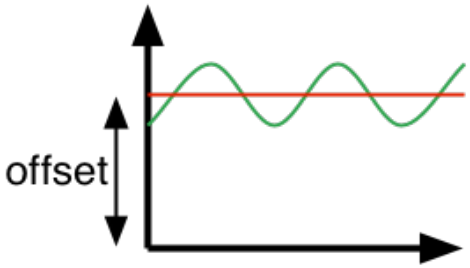


Locomotion : GA

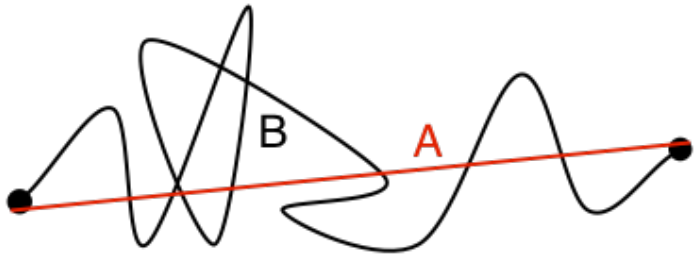
- Genetic algorithm :
  - Le même que pour l'oscillateur

- Encodage :





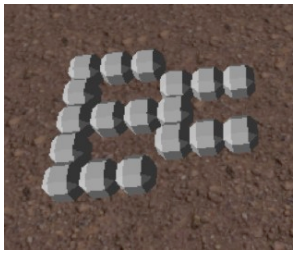
- Fonction de fitness :



$$F = A + \beta \cdot B$$

Locomotion : Résultats I

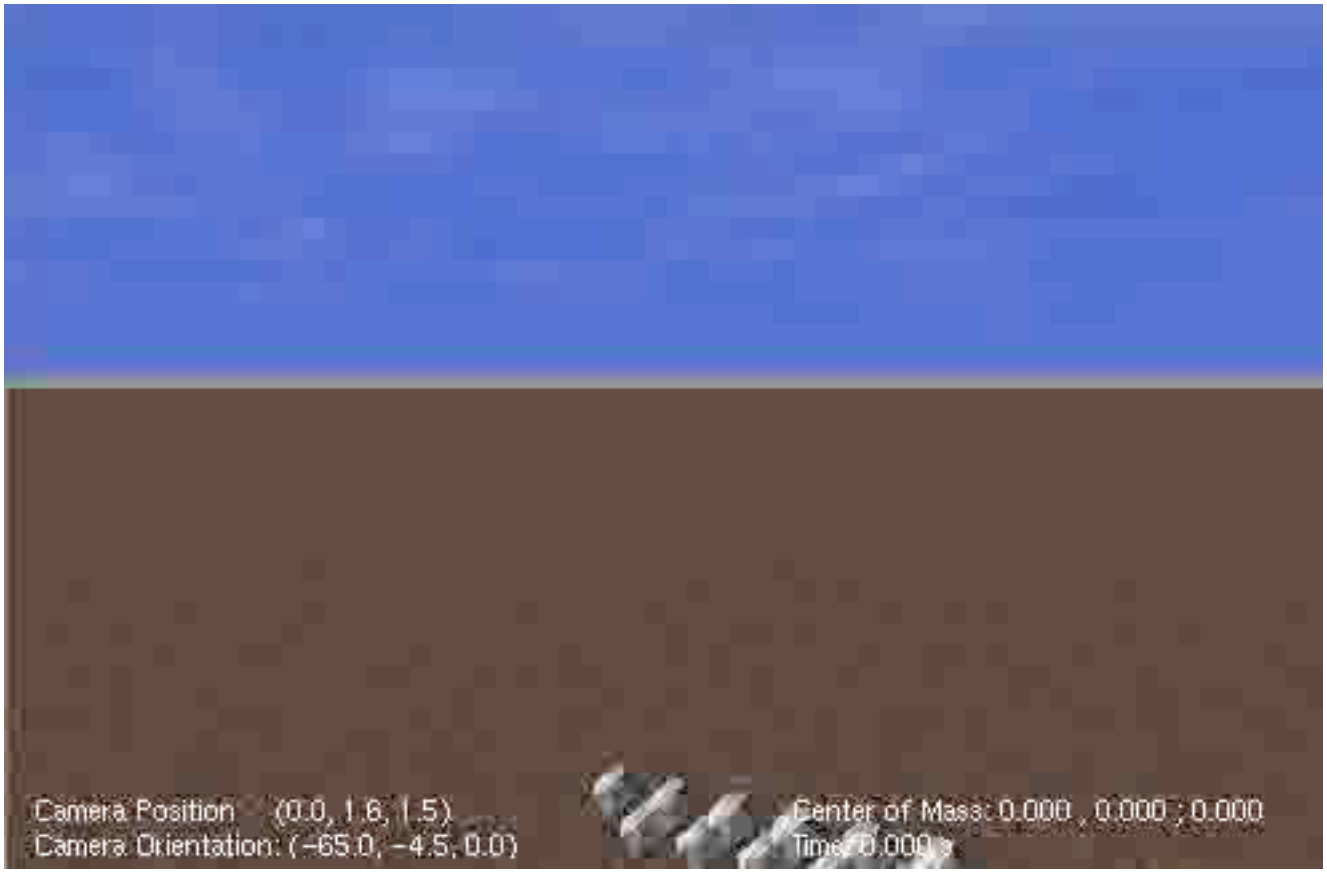
- 3-4 runs pour *fleche* et *tetrapode*, 12 pour le *serpent*
- Tetrapode : résultats décevants en terme de vitesse (dûs à la forme ?)
- Le serpent n'a pas retrouvé la locomotion trouvée manuellement mais une efficiente, plié en deux



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Serpent

BIOLOGICALLY INSPIRED
ROBOTICS GROUP (BIRG)

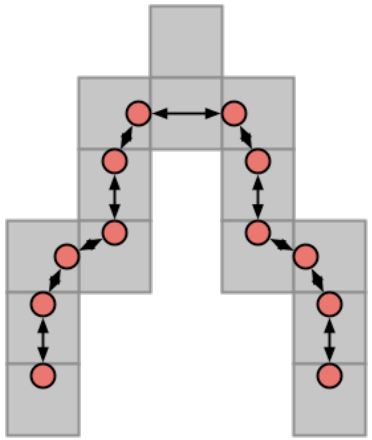


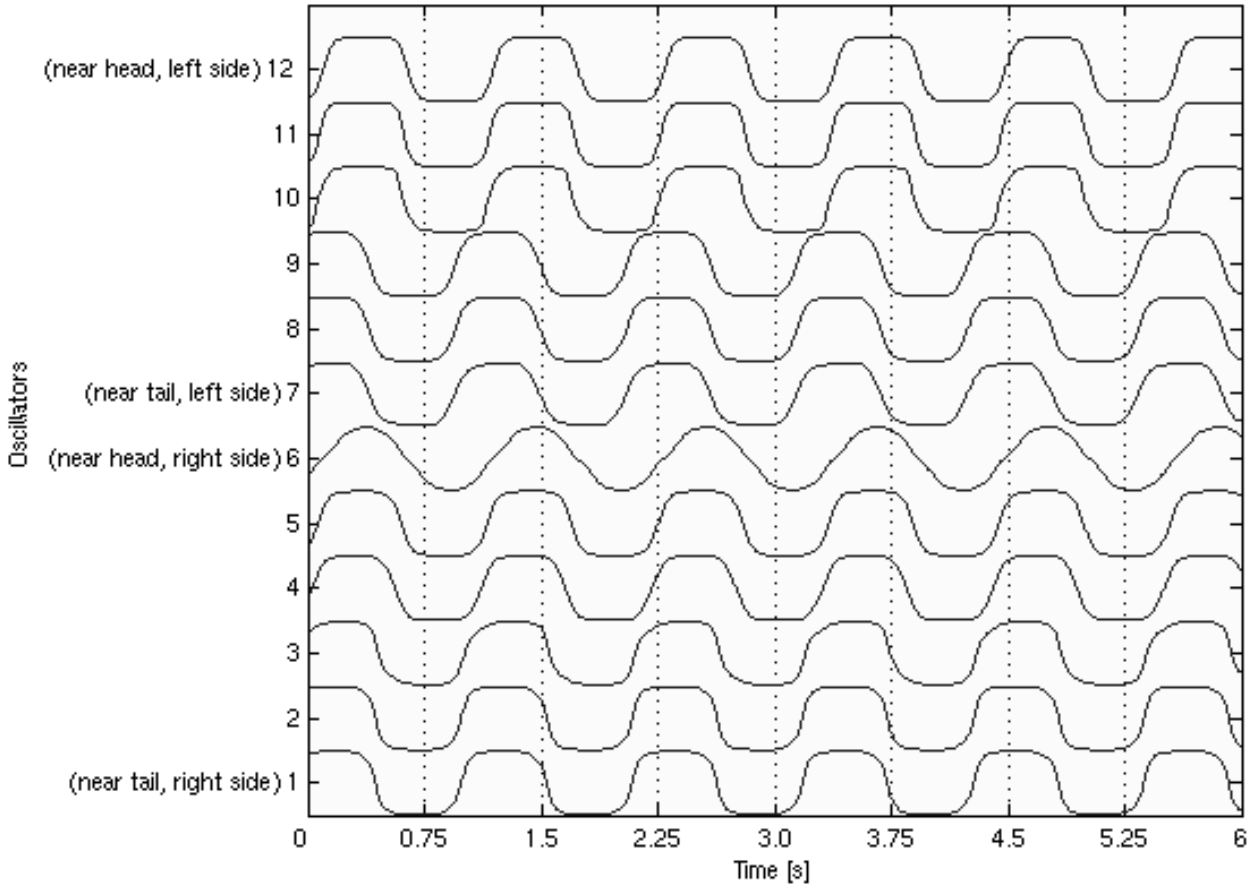
Fleche


Camera Position (-0.3, 1.8, 2.1)
Camera Orientation: (-93.0, -39.5, 0.0)
Center of Mass: 0.000 , 0.000 , 0.000
Time: 0.000 s

fleche

- La *fleche* (~9 m/min)





Locomotion : Discussion

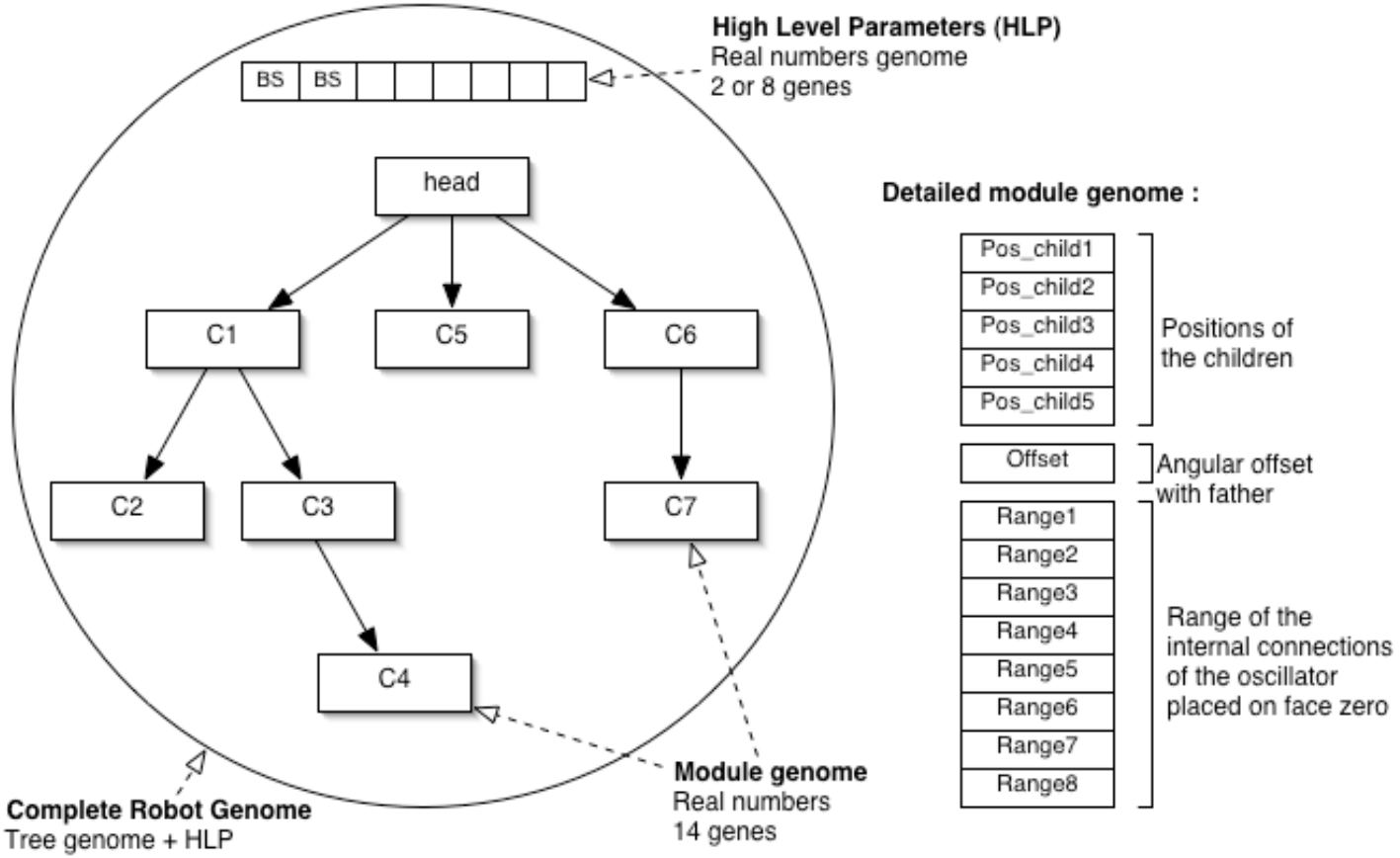
- Les CPG basés sur des réseaux de neurones oscillatoires couplés par synaptic spreading sont efficaces dans notre cas
- Nous n'avons utilisés que des mouvements ne tirant pas parti des joints particuliers du module neubot
- Les formes ne sont pas forcément les mieux adaptées

➡ **co-évolution**

Co-évolution : motivation

- Co-évolution de la structure et du contrôleur
  - Eviter de péjorer des résultats avec des formes inappropriées (cf axes de rotation des modules neubot)
  - Bio-inspiré
  - Bien adapté à la robotique modulaire
- Autres recherches
  - Daniel Marbach
  - M-TRAN, recherches de Hornby et Pollack
  - Sims block creatures, framestick

Co-Ev : Encoding

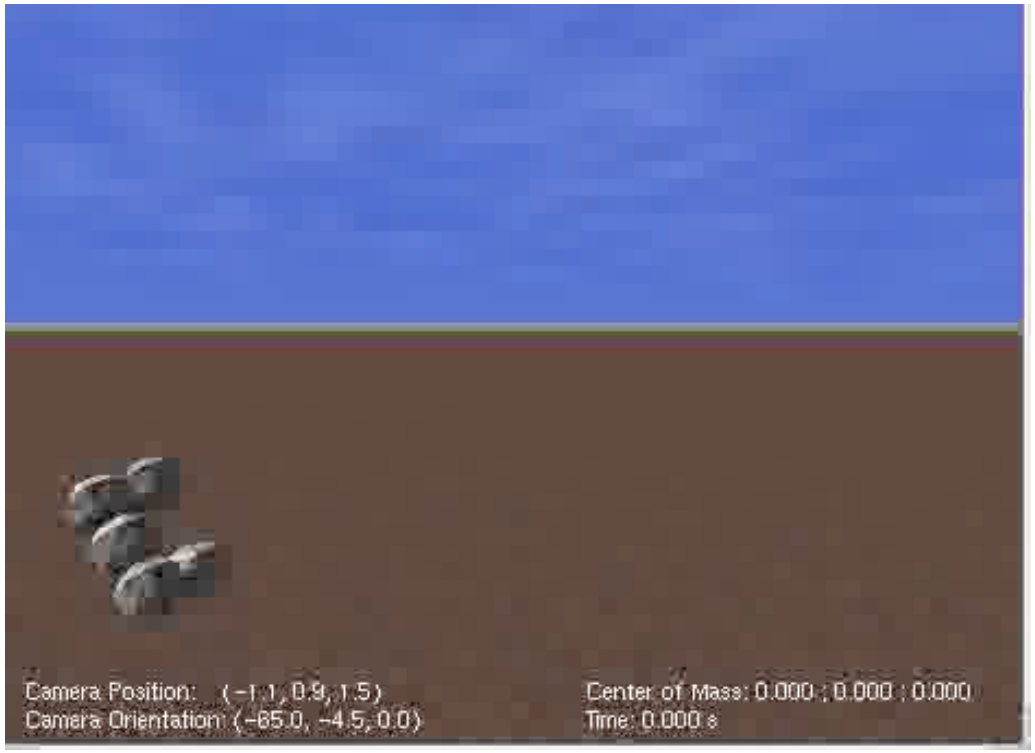


Detailed High Level Genome :

BSleft	BSRight	P_Mut	STD_Dev	P_Dest	PSwapTree	PSwapNode	P_Cross

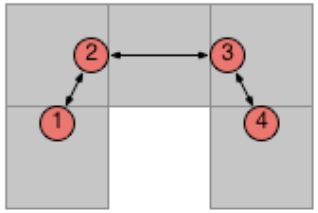
CE : Resultats 1ère série

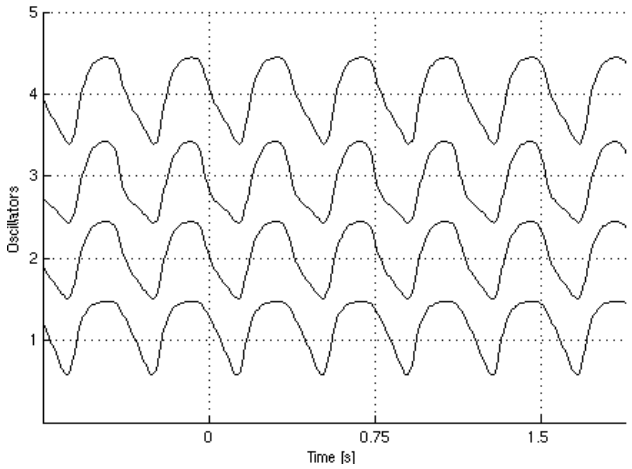
- Nombre initial de modules variable
- Fonction de Fitness générique
- Problèmes
  - Trajectoires en spirales
  - "Petits" robots

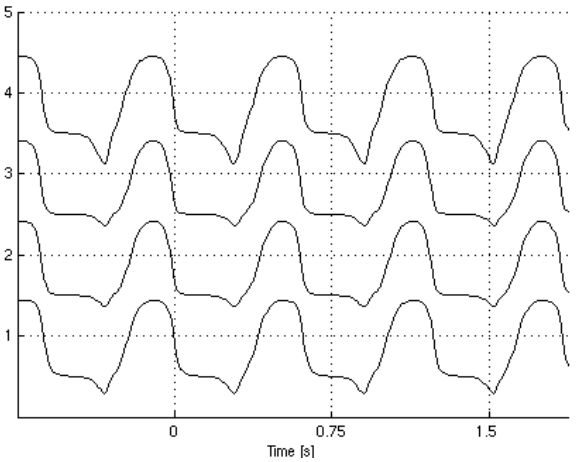


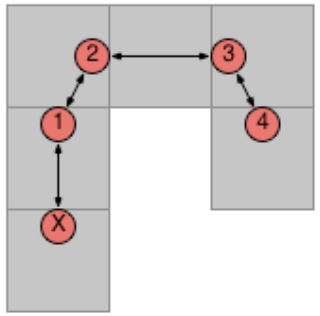
CE : Resultats 4ème série

- Changements séries 2 à 4
  - Nombre initial de module plus élevé
  - Les modules peuvent être activés ou non : auparavant, seuls certains couplages pouvaient faire cela
  - Oscillateur d'un nouveau module n'est plus connecté aléatoirement aux autres





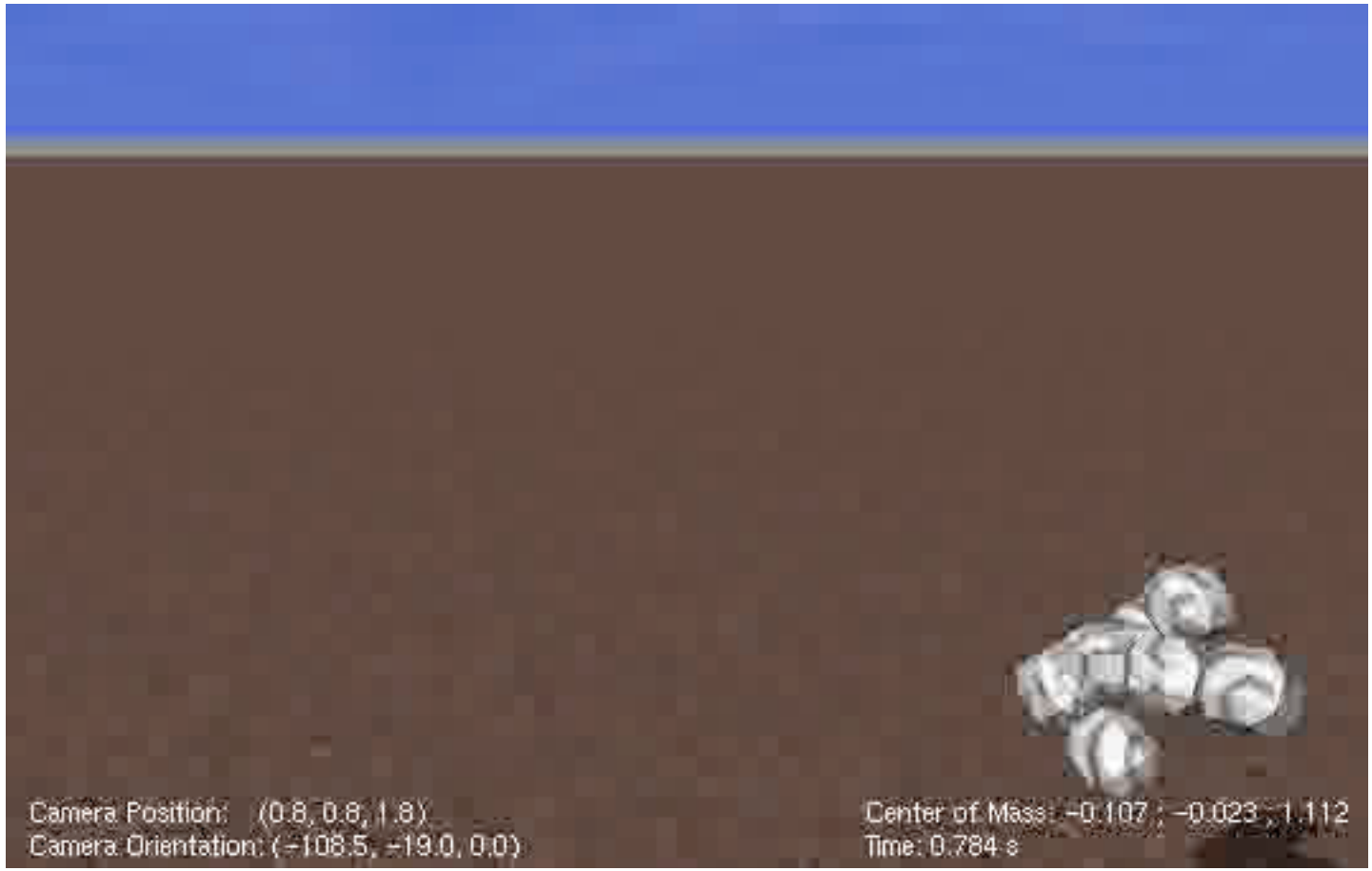




Films

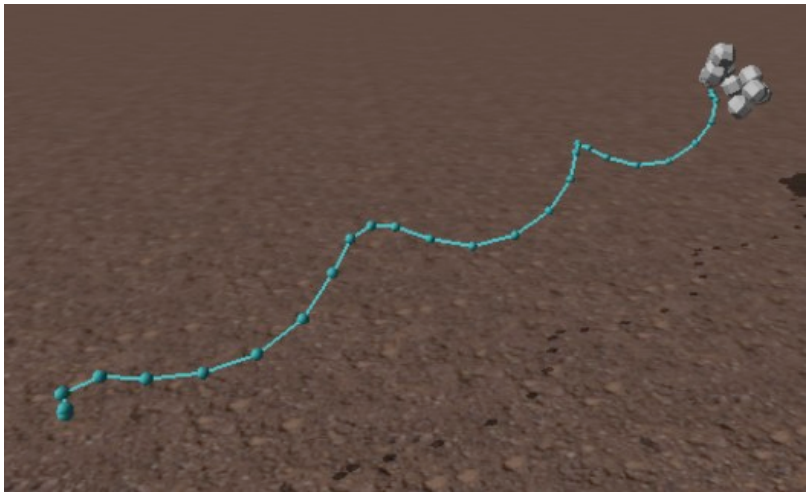

Camera Position: (-1.2, 1.4, 1.5)
Camera Orientation: (-91.0, -12.5, 0.0)
Center of Mass: -0.581 , 0.119 , 1.058
Time: 5.880 s

Films


Camera Position: (0.8, 0.8, 1.8)
Camera Orientation: (-108.5, -19.0, 0.0)
Center of Mass: -0.107 ; -0.023 ; 1.112
Time: 0.784 s

Co-Ev : Discussion

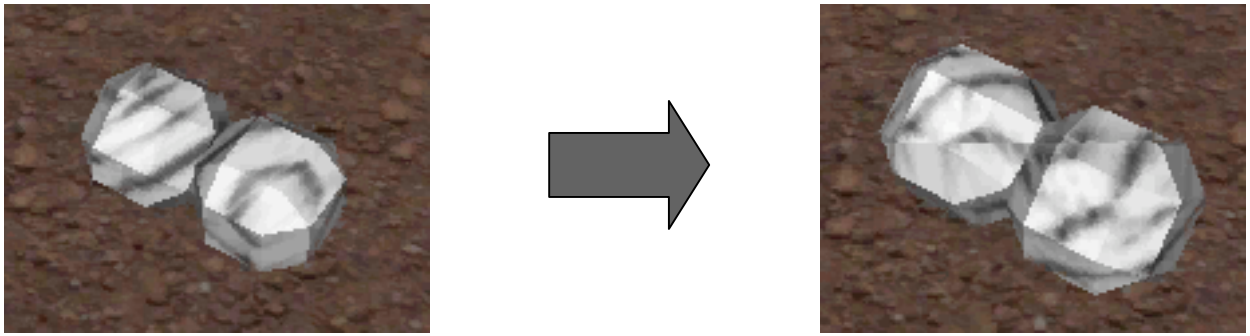
- Symétrie : souvent (toujours?) le cas dans la nature. Ici, pas obligatoire
  - Certains deviennent symétriques (difficulté à grandir…)
  - Les autres ont des trajectoires en spirale
- Construction de l'arbre en profondeur
- Les robots sont bloqués dans optima locaux



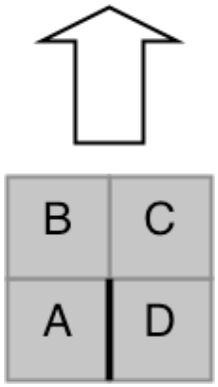
=> autres méthodes d'optimisation (PSO50, autre GA)

Co-Ev : Autre contrôleur

- Jusqu'ici utilisation d'oscillations
- Les moteurs des modules Neubot peuvent être utilisés pour tourner de manière continue (comme une hélice)



Co-Ev : Autre contrôleur



Conclusion

- Framework & simulateur
  - Fonctionne bien mais le transfert vers la réalité risque d'être problématique (simplifications très fortes)
- Locomotion
  - Les robots co-évolués sont plus efficaces que ceux avec une formes fixées
  - L'absence de symétrie est compensée par des trajectoires en spirale
  - Oscillations > rotation continue
- Projet Neubot
  - L'axe de rotation parallèle à la normale est peu utilisé

Travail futur

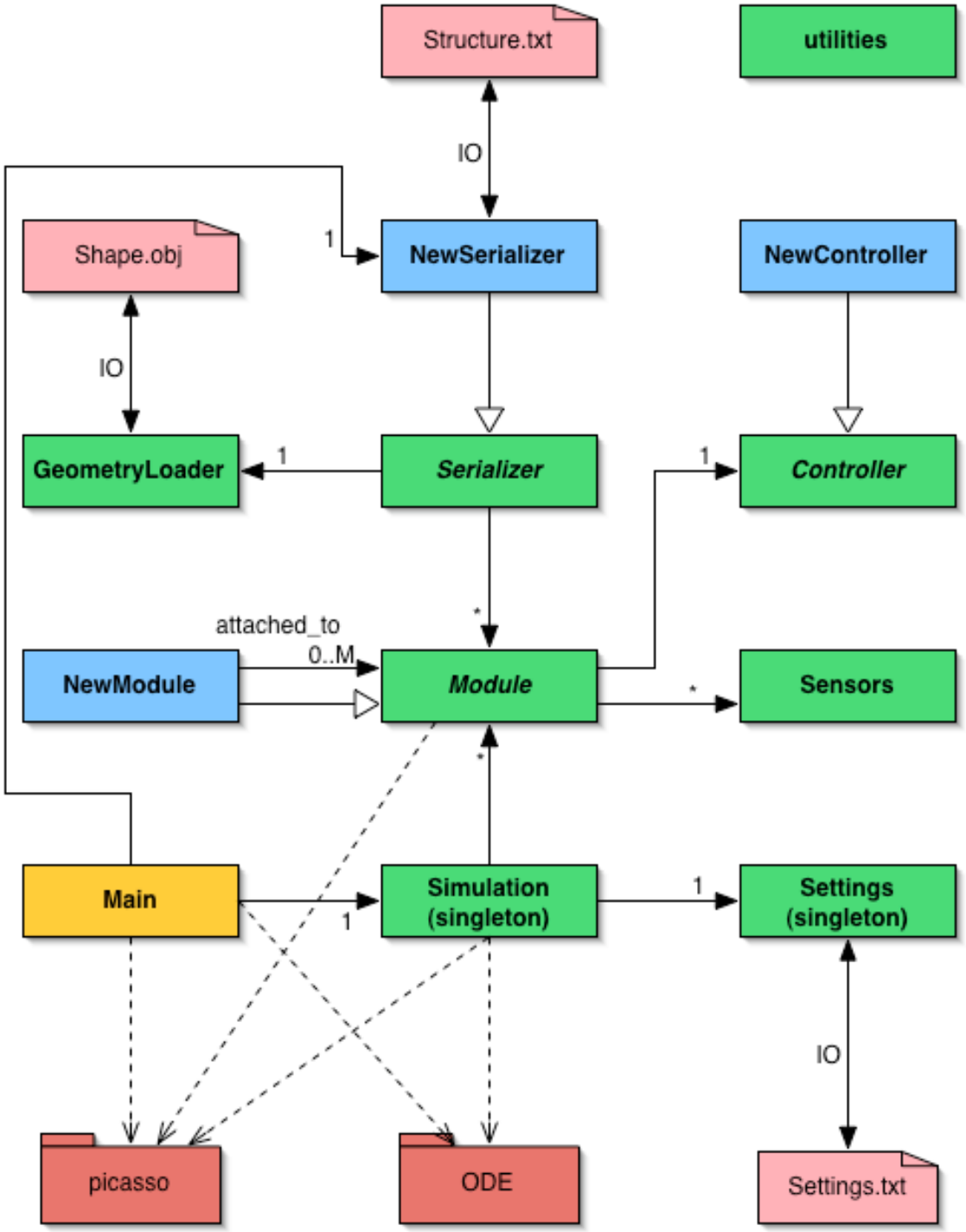
- Affiner le simulateur en fonction de résultats expérimentaux
- Fonction de fitness qui favorise des robots capables de contrôler leur direction et leur vitesse
- Tester des solutions mixtes : oscillations - rotation continue
- Tester d'autres GA et d'autres techniques d'optimisation

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Questions

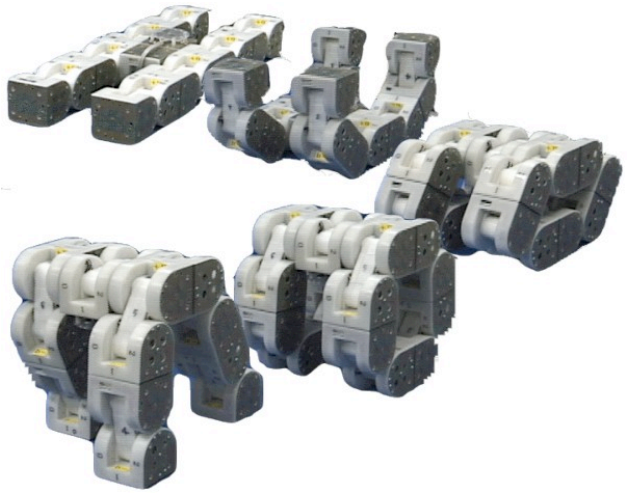
BIOLOGICALLY INSPIRED
ROBOTICS GROUP (BIRG)

?


Structure.txt
utilities
IO
1
Shape.obj
NewSerializer
NewController
IO
GeometryLoader
1
Serializer
1
Controller
attached_to
0..M
NewModule
Module
Sensors
Main
1
Simulation
(singleton)
1
Settings
(singleton)
IO
picasso
ODE
Settings.txt

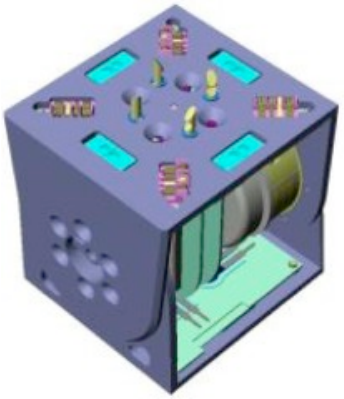
Robotique modulaire II

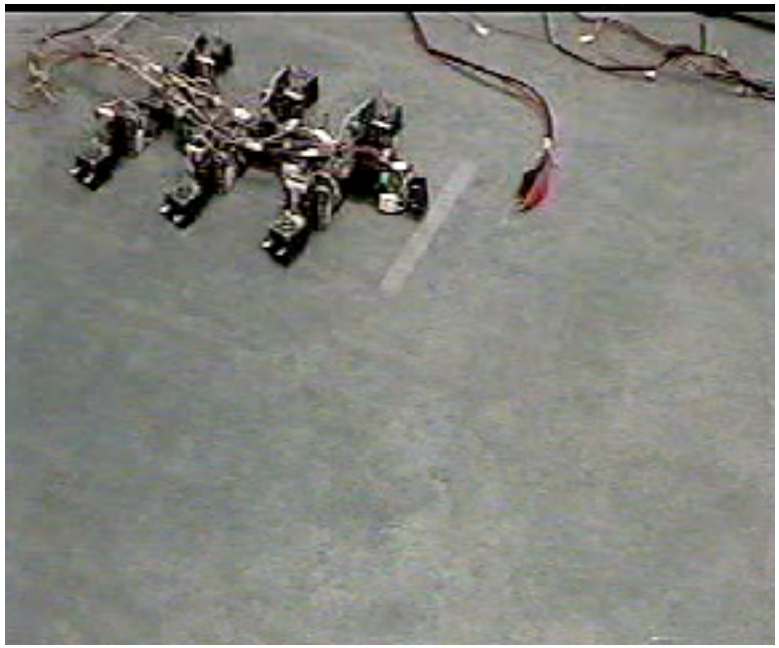
- Exemple 1 : M-TRAN II
  - Développé par AIST (Japon)
  - Auto-reconfiguration
  - Différentes façons de se mouvoir réalisées avec GA et CPG



Robotique modulaire III

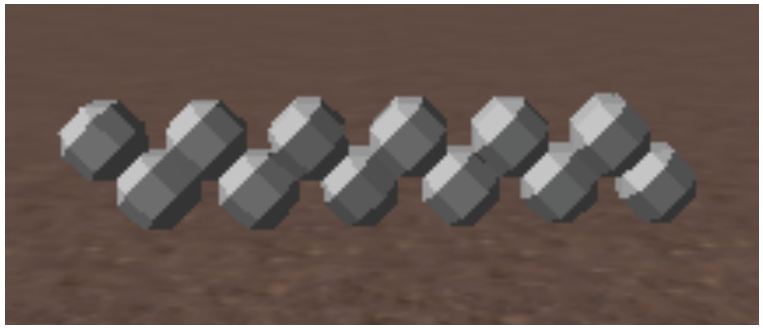
- Exemple 2 : PolyBot
  - Développé par le XEROX PARC
  - 3ème génération :
  - Auto-reconfiguration
  - Seulement 2 faces de connection mais éléments passifs





Le projet Neubot

- 6 faces pour s'attacher
- Joints magnétiques faits grâce à deux anneaux de dipôles dont la polarité peut être adaptée -> face attractive, **neutre** ou repulsive



Simulateur pour Neubot

- Changements effectués par la suite
  - Système de connection
  - Sérialisation suivant le type de contrôleur (avec oscillateurs non-linéaires, réseaux de neurones, etc…)
  - Sérialisation séquentielle : position et quaternion uniquement du premier module, puis les suivants sont placés sur l'une des faces des précédents
  - Modèle physique de l'eau

Dynamique des fluides

$$F_R = L \cdot C_T \cdot v^2$$

where L is the frontal area of the object in the fluid, CT the dimensionless rotation friction constant and v is the velocity of the module

$$F_T = -\frac{1}{2} \rho_{water} \cdot L \cdot C_D \cdot v^2$$

where L is the frontal area of the object in the fluid, ρ_{water} *water is the fluid density ,* C_D *the drag coefficient and v is the velocity of the module*

$$F_B = \rho_{water} \cdot g \cdot V$$

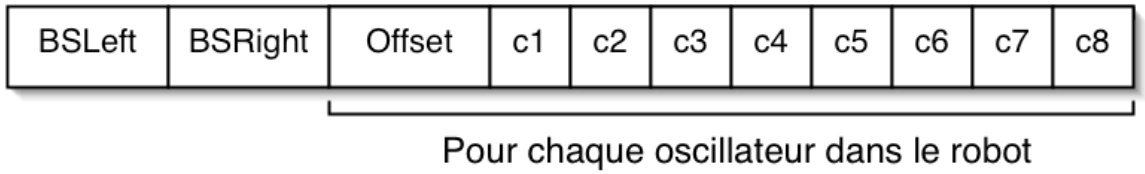
where water represents the density of the fluid, g is gravitation acceleration and V the volume of the module

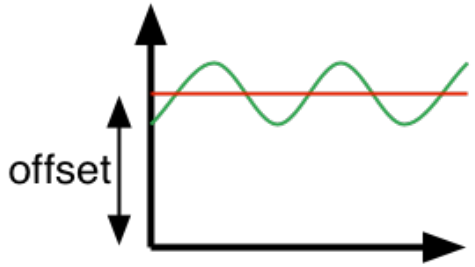
Oscillateurs : description

- **Réseaux de neurones** VS oscillateurs non linéaires
  - Cohérence avec les autres parties du contrôleurs
  - Oscillations plus variées
  - Plus proche de la nature
- Choix d'un réseau simple dont les connexions interneurales ainsi que les paramètres des neurones sont évolués (GA) dans le but d'avoir un réseau générant des oscillations

Locomotion : GA

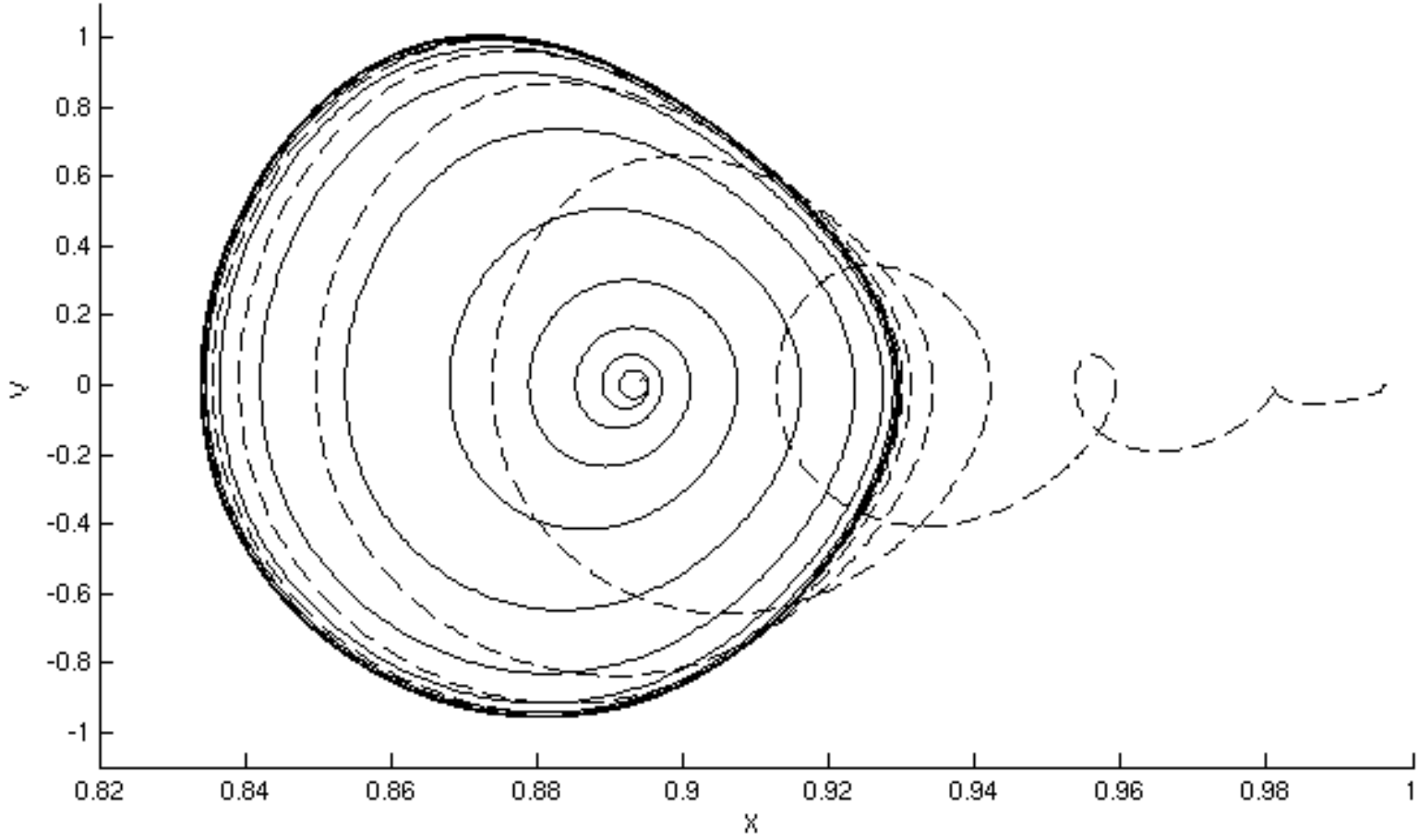
- Genetic algorithm :
  - Le même que pour l'oscillateur
  - Encoding :





  - Différentes fonction de fitness testées :
    - Générique : $f = \alpha \cdot \left\| \vec{p}_N - \vec{p}_1 \right\| + \beta \cdot \sum_{i=1}^{N-1} \left\| \vec{p}_{i+1} - \vec{p}_i \right\|$
    - Ou privilégiant un direction ou le plan XY

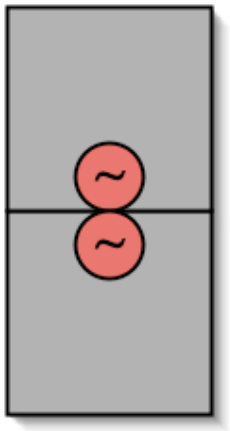
Oscillateurs : résultats

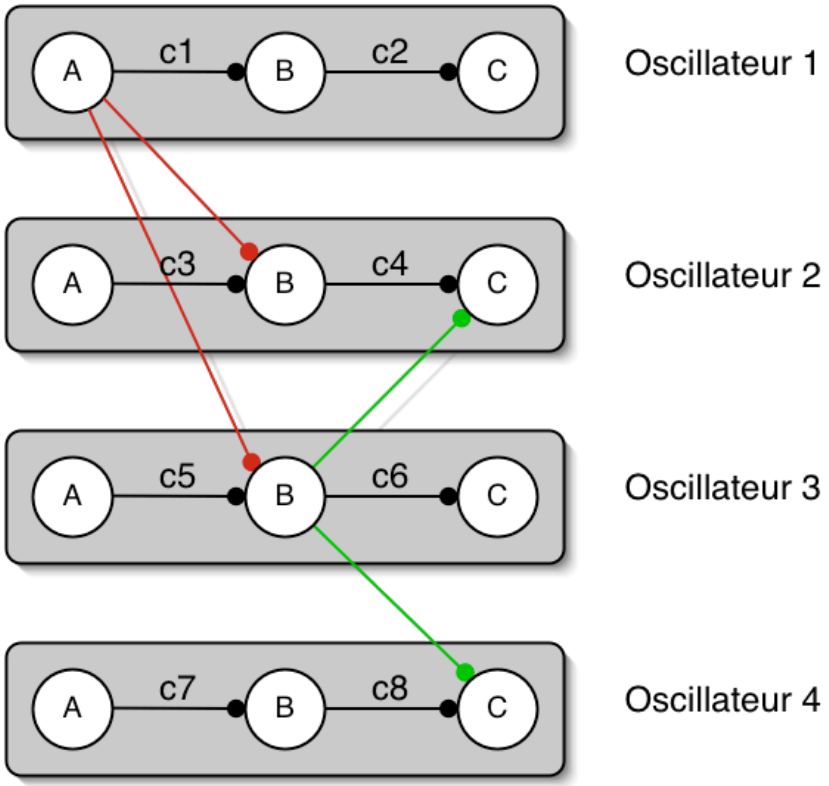


Cycle limite : affichage de MI selon la position et la vitesse

Locomotion : Méthode

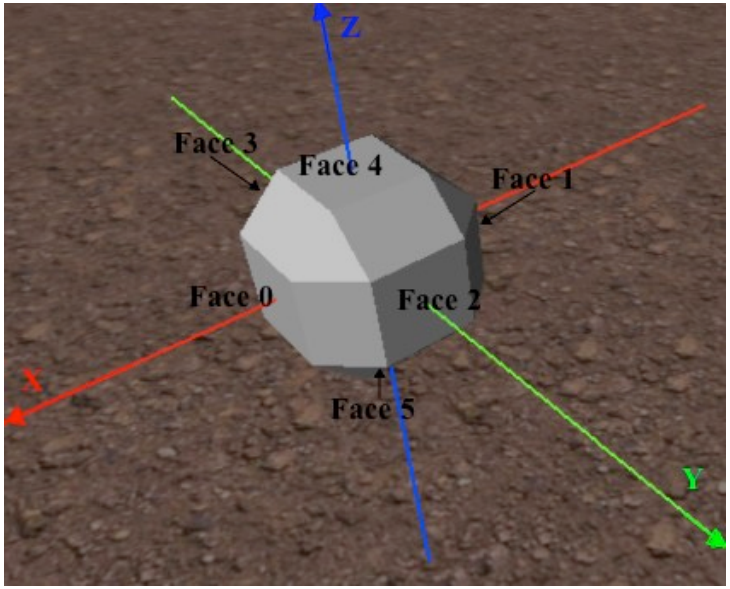
- Méthode : couplage des oscillateurs par **synaptic spreading**



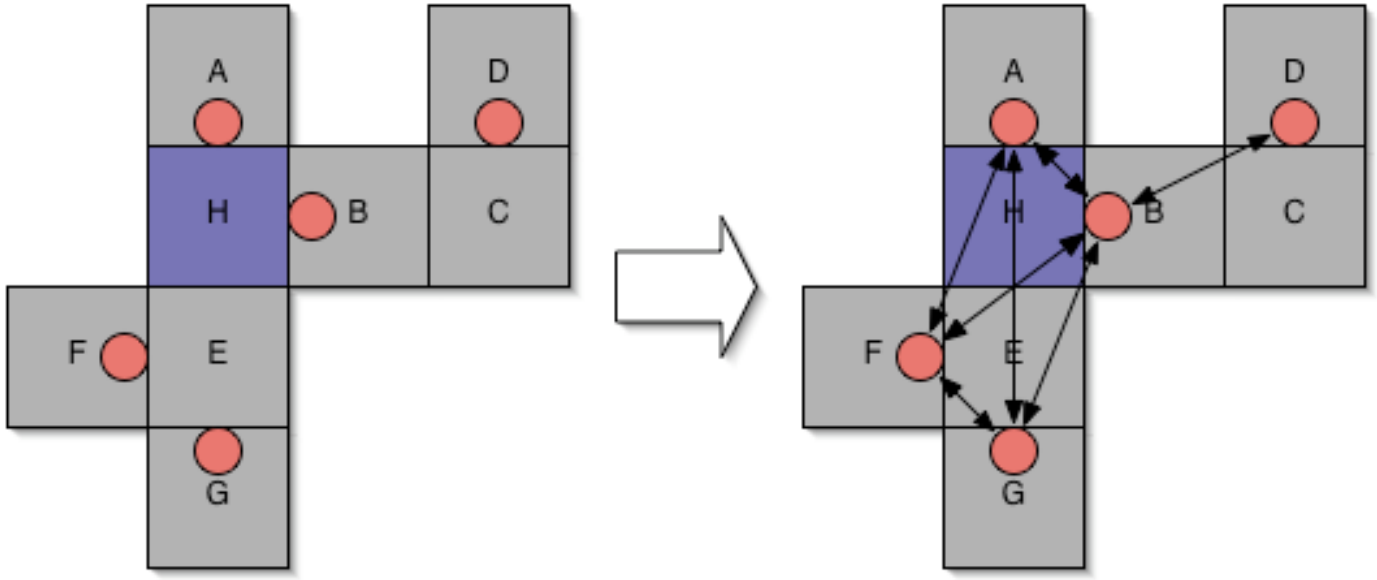


Co-Ev : Généralités

- Structure en arbre (plus de cycles)
- Toujours attaché au père par la face 0. Uniquement 1 moteur est activé dans chaque module -> un seul oscillateur
- Même oscillateur qu'avant
- Synaptic spreading
- Elimination des robots "impossibles"



Co-Ev : Encoding



Co-Ev : GA + fitness

- GA : steady-state, pop : 100, rank-proportional roulette wheel selection, même ff qu'avant
- Mutations
  - Au niveau de l'arbre
    - Destruction d'un nœud ou d'un sous-arbre
    - Ajout d'un nœud
    - Echange de 2 nœuds ou sous-arbres
  - Au niveau des nœuds et des paramètres généraux:
    - Ajout d'une petite valeur tirée d'une gaussienne
    - Pour les positions des enfants : échange de deux places
- Crossover
  - Au niveau de l'arbre
    - Echange de sous-arbre de deux robots

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Co-Ev : Results

BIOLOGICALLY INSPIRED ROBOTICS GROUP (BIRG)

- Nombreuses évolutions lancées mais principalement 4 séries
- Généralités
  - Diversité des robots trouvés
  - Optima locaux fortement attractifs
  - Le nombre initial de modules a une très forte influence, difficulté à augmenter ou réduire le nombre de modules

CE : Resultats 2ème série

- Changements
  - FF favorise le plan horizontal
  - Nombre initial de modules plus élevé
- Résultats relativement semblables
- Mêmes problèmes
- Film…

CE : Results 3ème série

- Changements
  - L'oscillateur d'un nouveau module n'est pas connecté initialement aux autres oscillateurs
  - But : ne plus perturber le comportement général à cause de l'adjonction d'un module
- Résultats relativement semblables
- Mêmes problèmes
- Film…

Co-Ev : Discussion

- Solutions ?
  - Forcer la symétrie
  - Tester la construction des arbres en largeur
  - Tester des méthodes d'optimisation différentes (PSO50?) ou des GA différents (populations multiples avec transfert)