

# Robótica Evolutiva

uma População de Robôs Reais que  
Fazem Amor e Geram Descendentes  
Enquanto Aprendem a Realizar Tarefas

**Eduardo Simões**

Mestrado em Microeletrônica – UFRGS

Doutorado em Robótica – University of Kent at Canterbury, UK

Prof. Dr. Instituto de Ciências Matemáticas e de Computação – ICMC

<http://www.icmc.usp.br/~simoese/seminars>

email: [simoese@icmc.usp.br](mailto:simoese@icmc.usp.br)

# Pensamento:

“Quem acreditaria em uma Formiga em teoria?”

S. J. Gould, 1950

... Como se projeta uma Girafa?

... O que é Vida Artificial?

...”

# Sumário

## 1- Computação Evolutiva

1.1 – Conceito

1.2 – Inspiração na Natureza

1.3 – O Papel da Computação Evolutiva

1.4 – Redes Neurais Artificiais

## 2- Aplicações da Computação Evolutiva na Robótica

2.1 – Robótica Evolutiva

2.2 – Implementação

2.3 – Primeiros Experimentos

2.4 – Análise dos Dados Experimentais

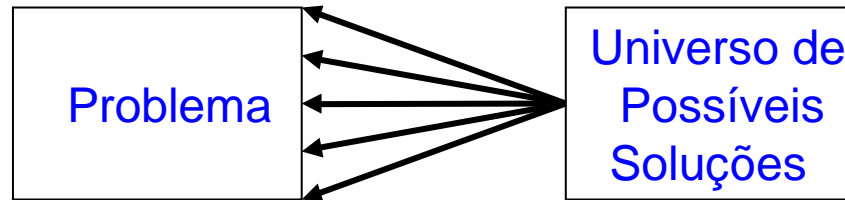
2.5 – Predação

## 3- Conclusões

# 1- Computação Evolutiva

# 1.1- Computação Evolutiva: Conceito

## ■ Sistemas de Computação Tradicionais:

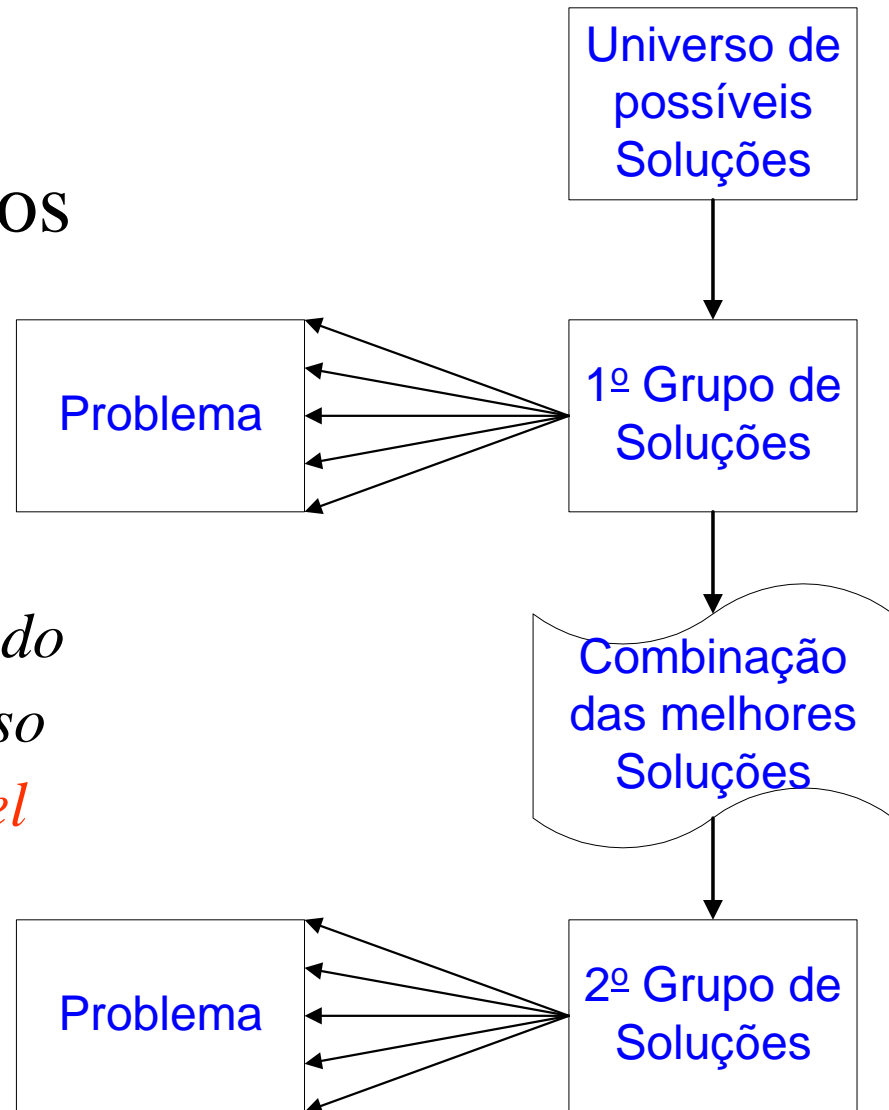


→ Tentar exaustivamente *todas as possíveis soluções* e escolher a mais adequada

# 1.1- Computação Evolutiva: Conceito

## ■ Algoritmos Genéticos

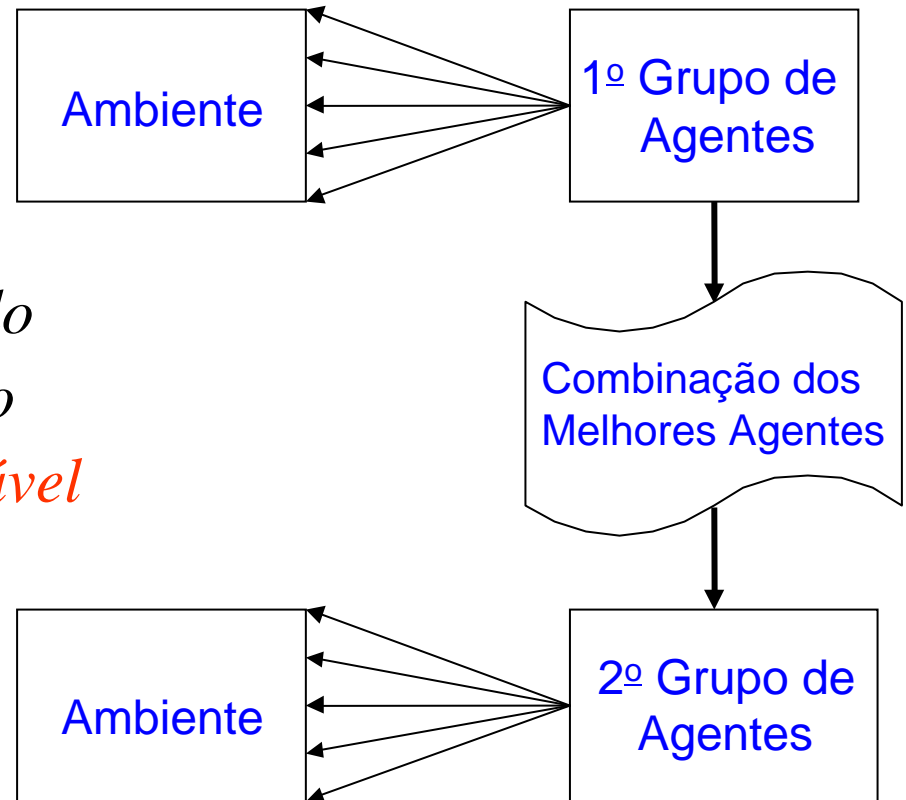
→ *Teste de um critério definido e inter rompimento do processo quando uma **solução aceitável** é produzida.*



# 1.1- Computação Evolutiva: Conceito

## ■ Computação Evolutiva

→ *Teste de um critério definido e interrompimento do processo quando um **desempenho aceitável** é produzido.*



# 1.1- Computação Evolutiva: Conceito

## ■ Computação Evolutiva:

→ *Uma Seleção Natural artificial dos mais adequados agentes ou soluções*

## ■ Premissa mais importante:

→ Especificar *o que* é desejado do robô, sem definir *como* ele deve fazer para obter esse comportamento



# 1.2- Inspiração na Natureza

## **Busca de inspiração na natureza:**

- **Nível Microscópico (Molecular):**
  - Algoritmos Genéticos
  
- **Nível Macroscópico (Comportamental):**
  - Complementaridade entre o ambiente natural e os organismos

## 1.2- Inspiração na Natureza

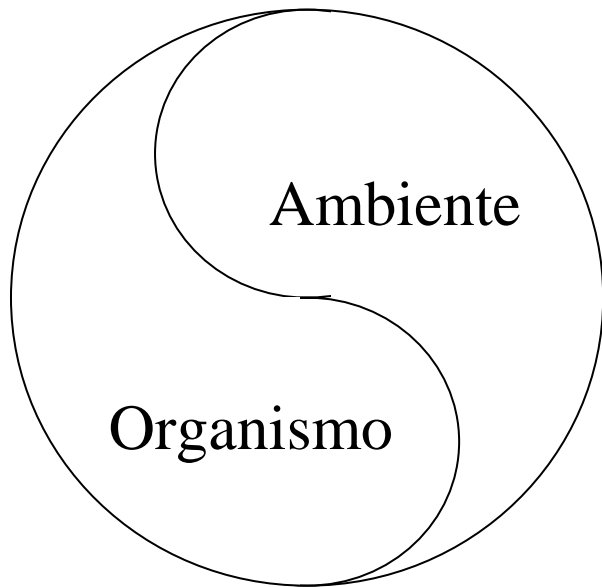
### **Interação entre Organismo e Ambiente:**

- Comportamento: propriedade emergente da interação entre organismo e meio ambiente
- *“O ambiente não é apenas uma entidade complexa e variável, mas um mundo de oportunidades”*

por J.J. Gibson (1950)

# 1.3- O Papel da Computação Evolutiva

## Evolução através da Seleção Natural



Características:

- Tamanho;
- Cor da Pele...

Mecanismos de Estímulo-Resposta:



# 1.3- O Papel da Computação Evolutiva

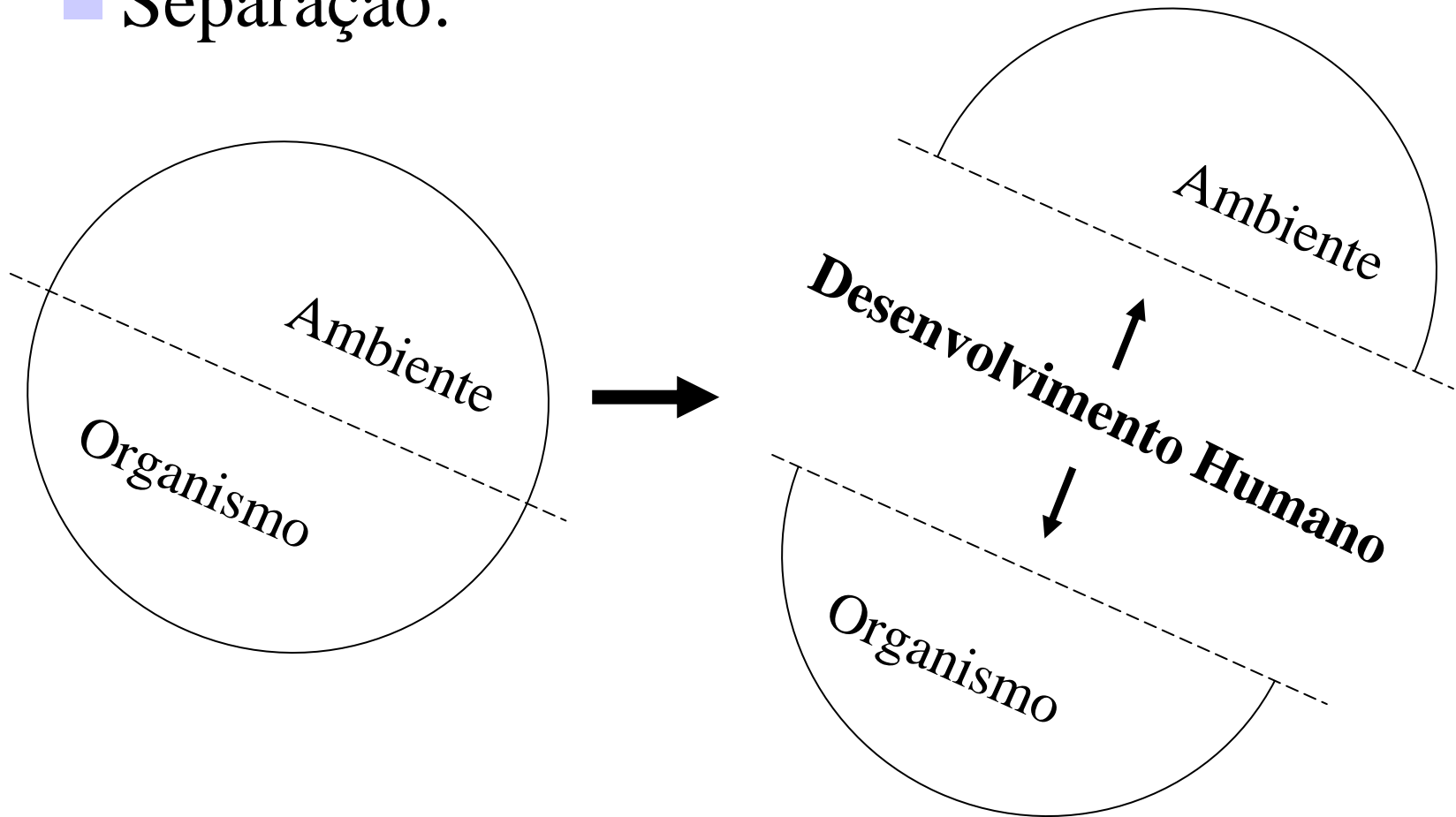
## ■ O Aparecimento da Inteligência:

- Biologia: Mecanismos → neurônios, cérebro...
- Psicologia: Mecanismos → cognição
- Informática: Modelos de Inteligência Artificial

**→ *Suprema Ferramenta para a Sobrevivência***

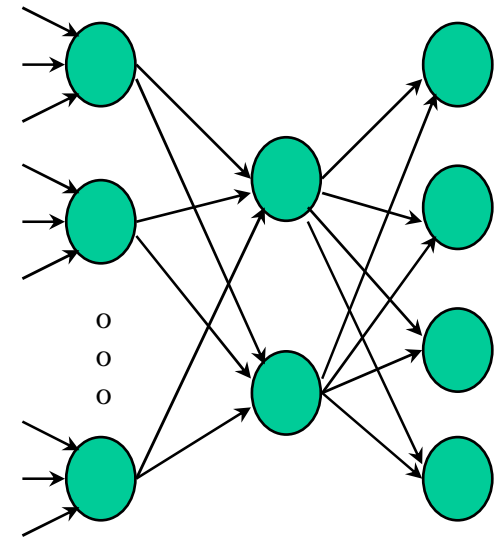
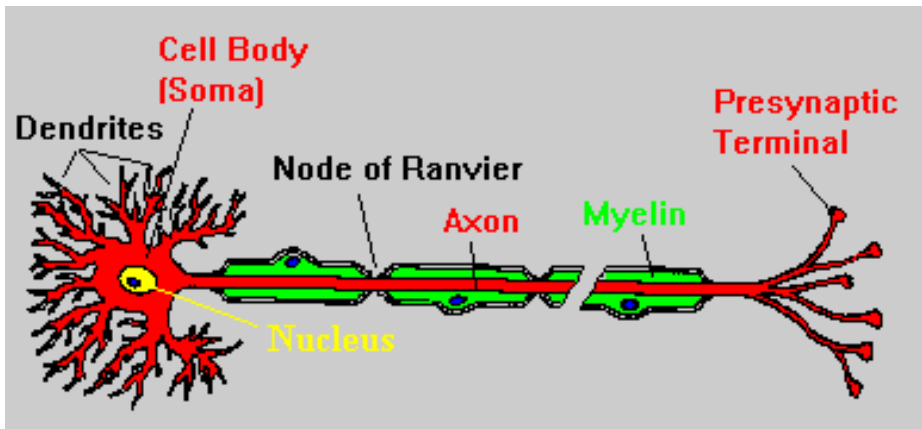
# 1.3- O Papel da Computação Evolutiva

## ■ Separação:



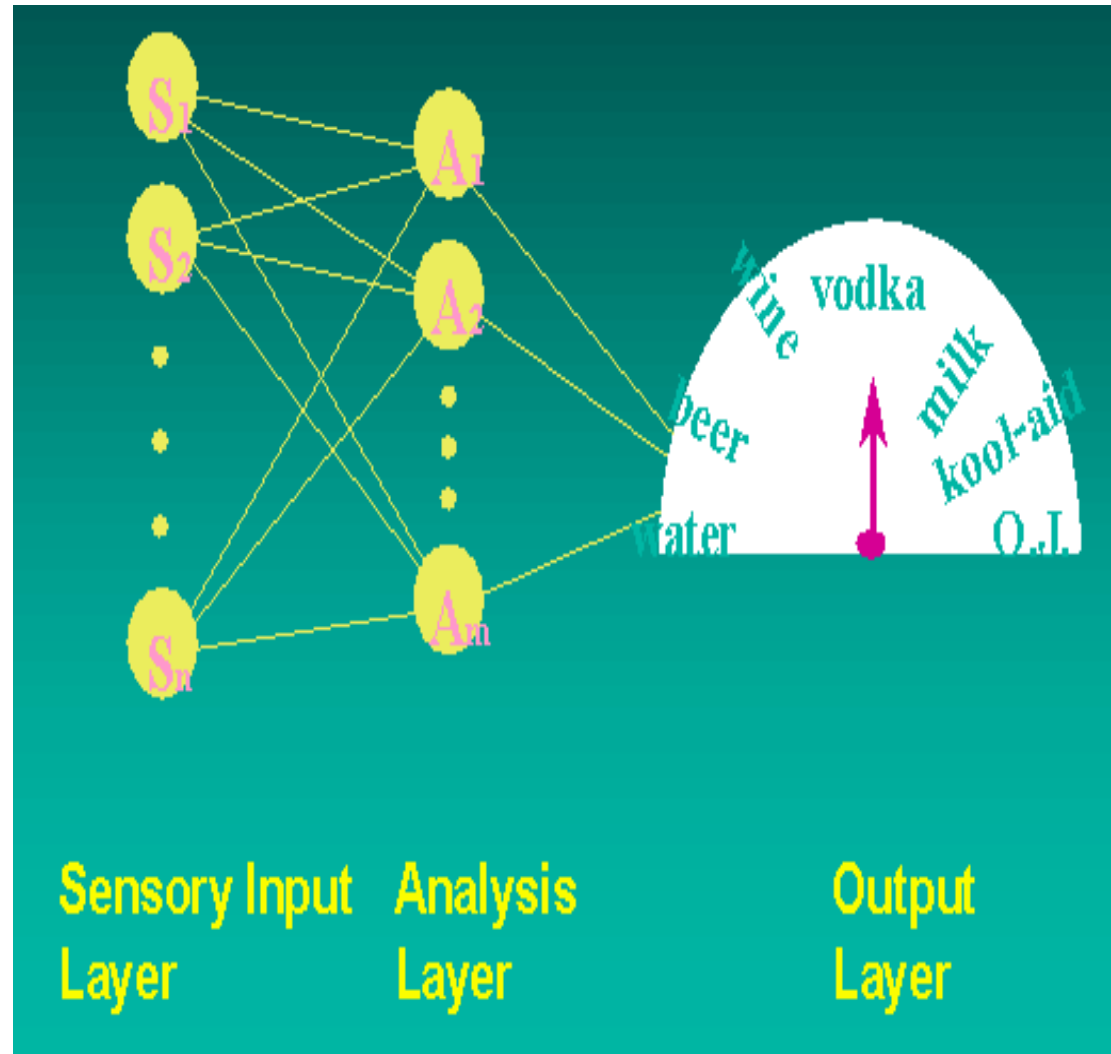
# 1.4- Redes Neurais Artificiais

- Redes Neurais Artificiais  
(cérebro dos Robôs):



# 1.4- Redes Neurais Artificiais

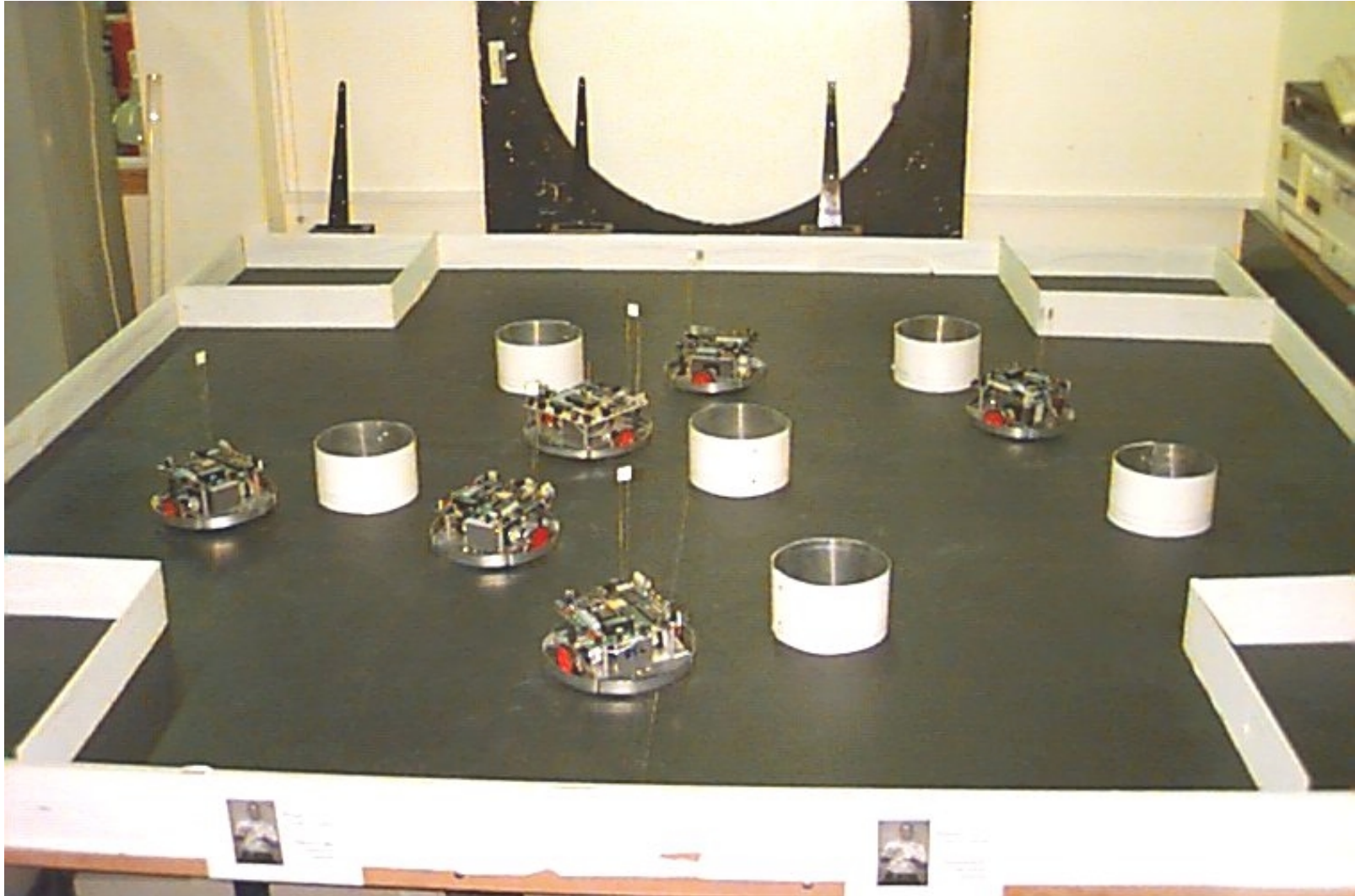
- Modelos inspirados na organização do cérebro
- Aprendizado e Teste
- Generalização e Classificação



## 2- Aplicações da Computação Evolutiva na Robótica

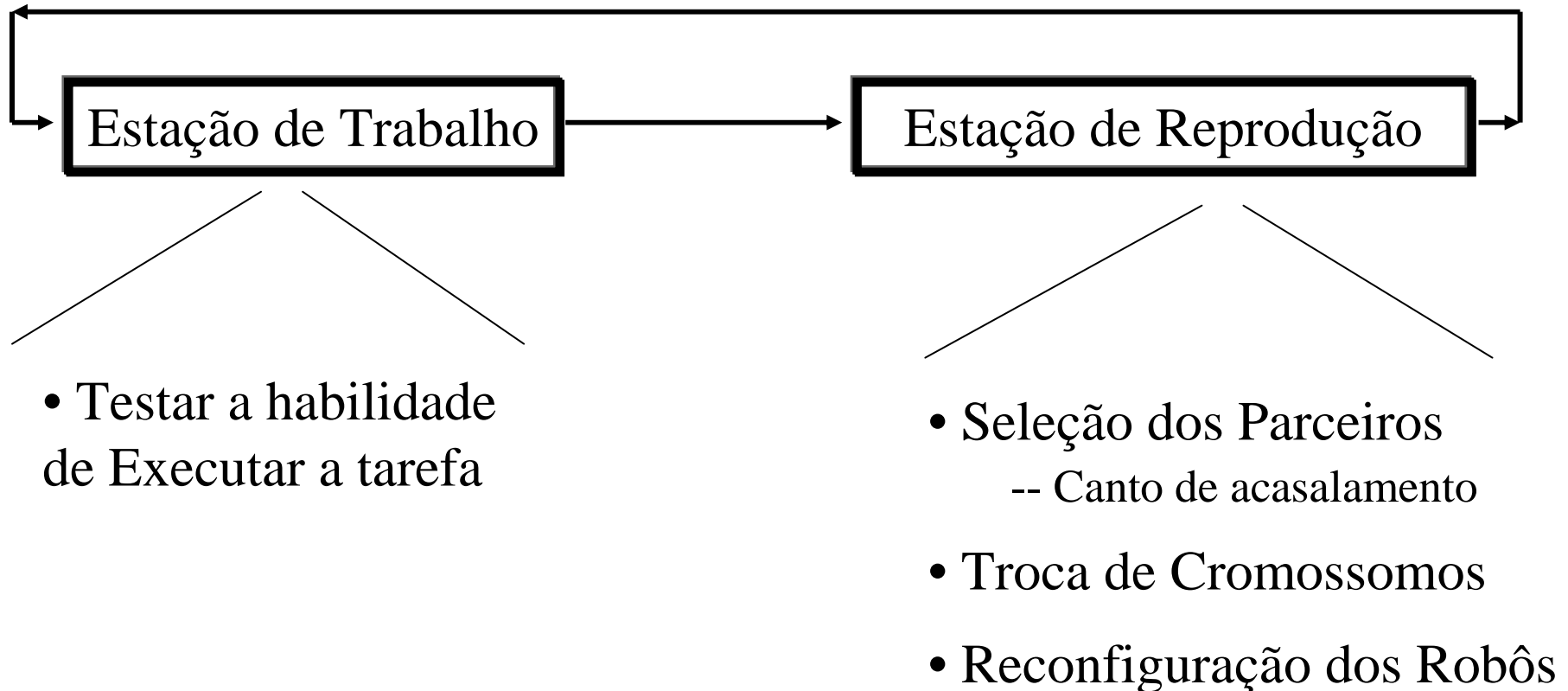


# 2.1- Robótica Evolutiva

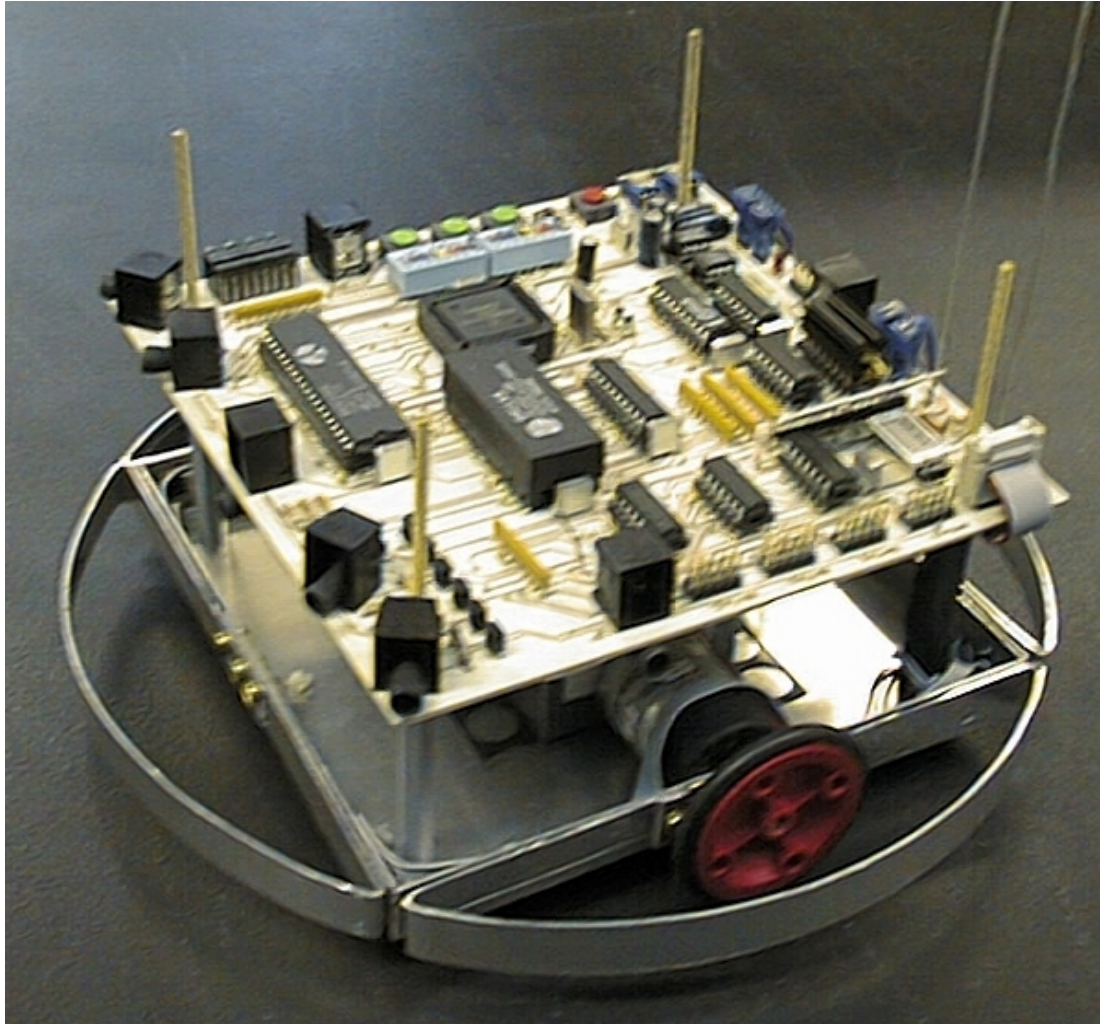


# 2.1- Robótica Evolutiva

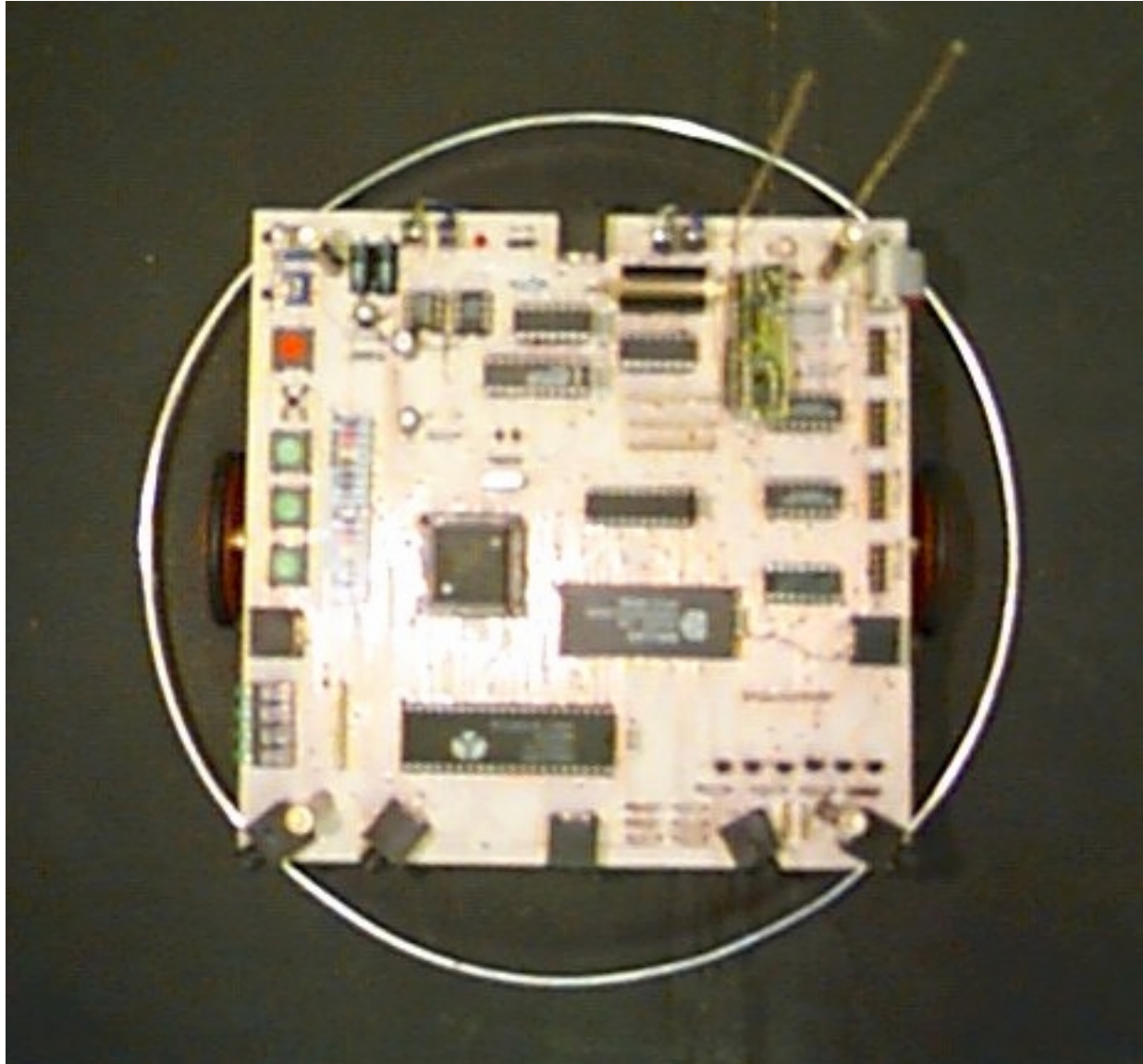
## Processo Evolucionário:



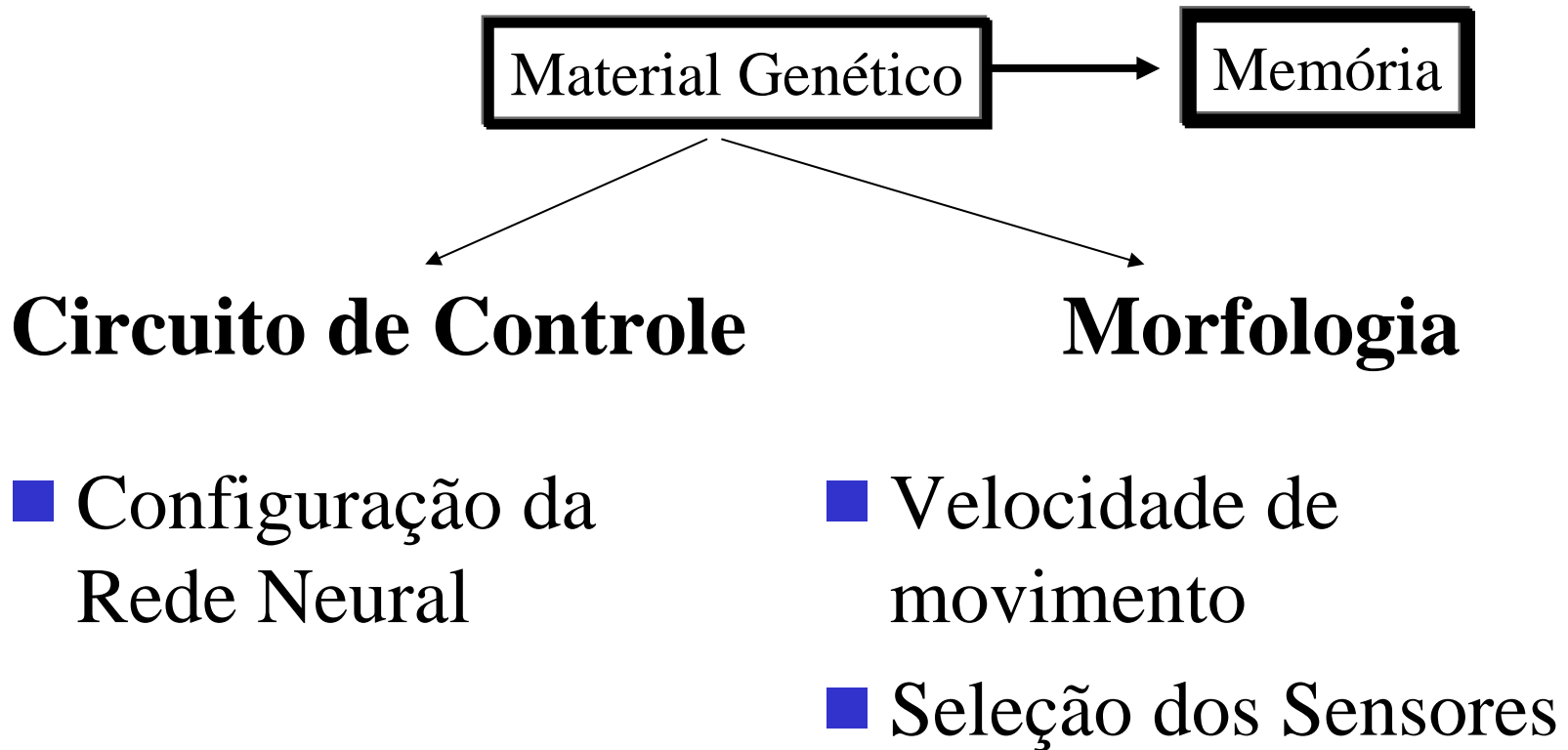
## 2.1- Robótica Evolutiva



## 2.1- Robótica Evolutiva

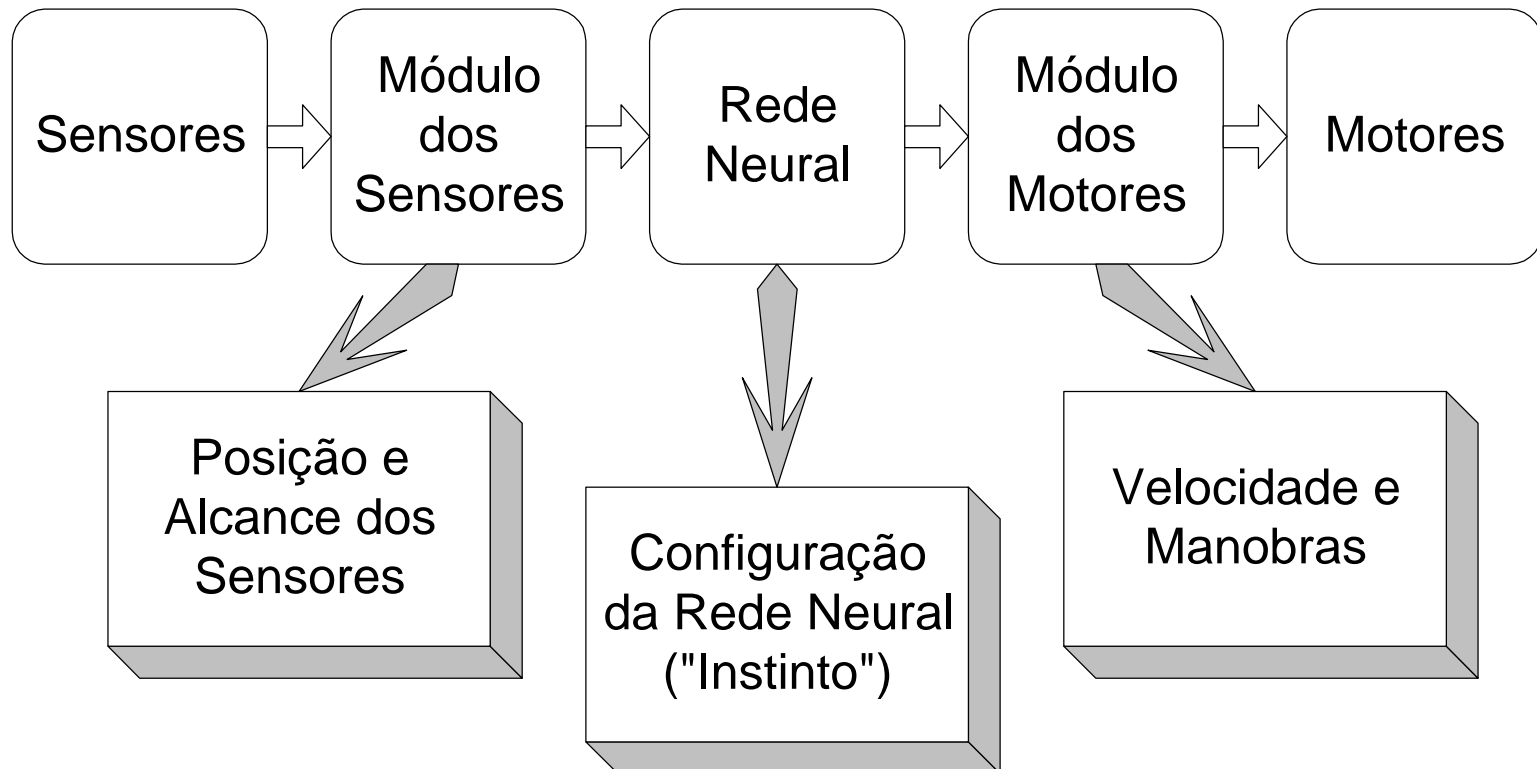


# 2.1- Robótica Evolutiva



# 2.2- Implementação

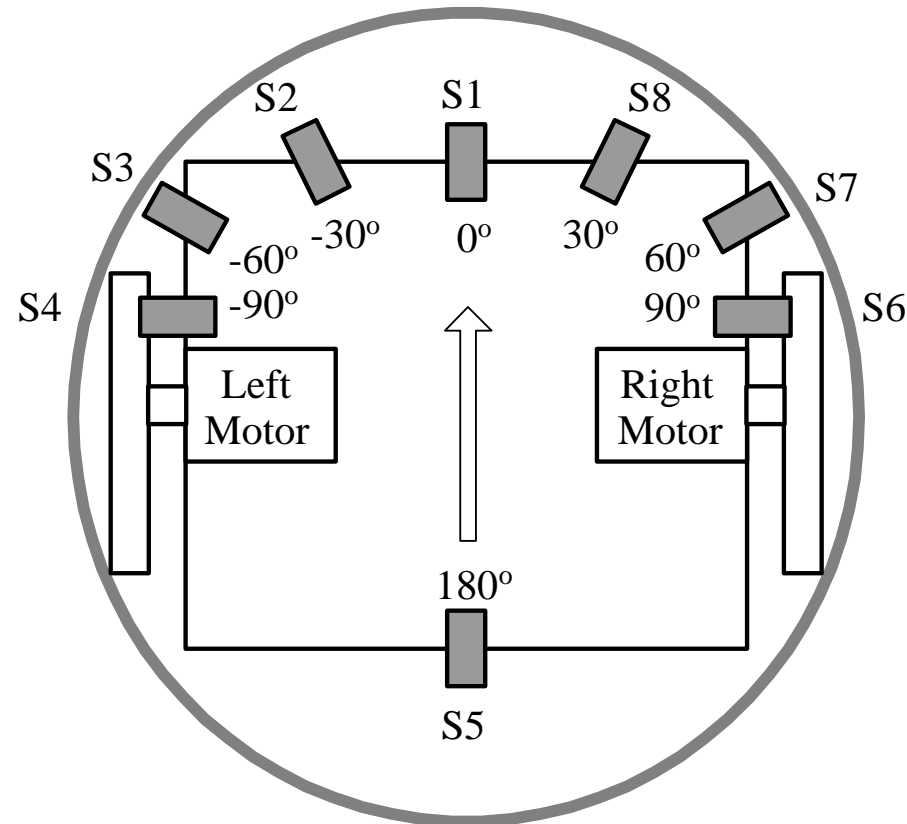
## ■ Arquitetura do Robô



# 2.2- Implementação

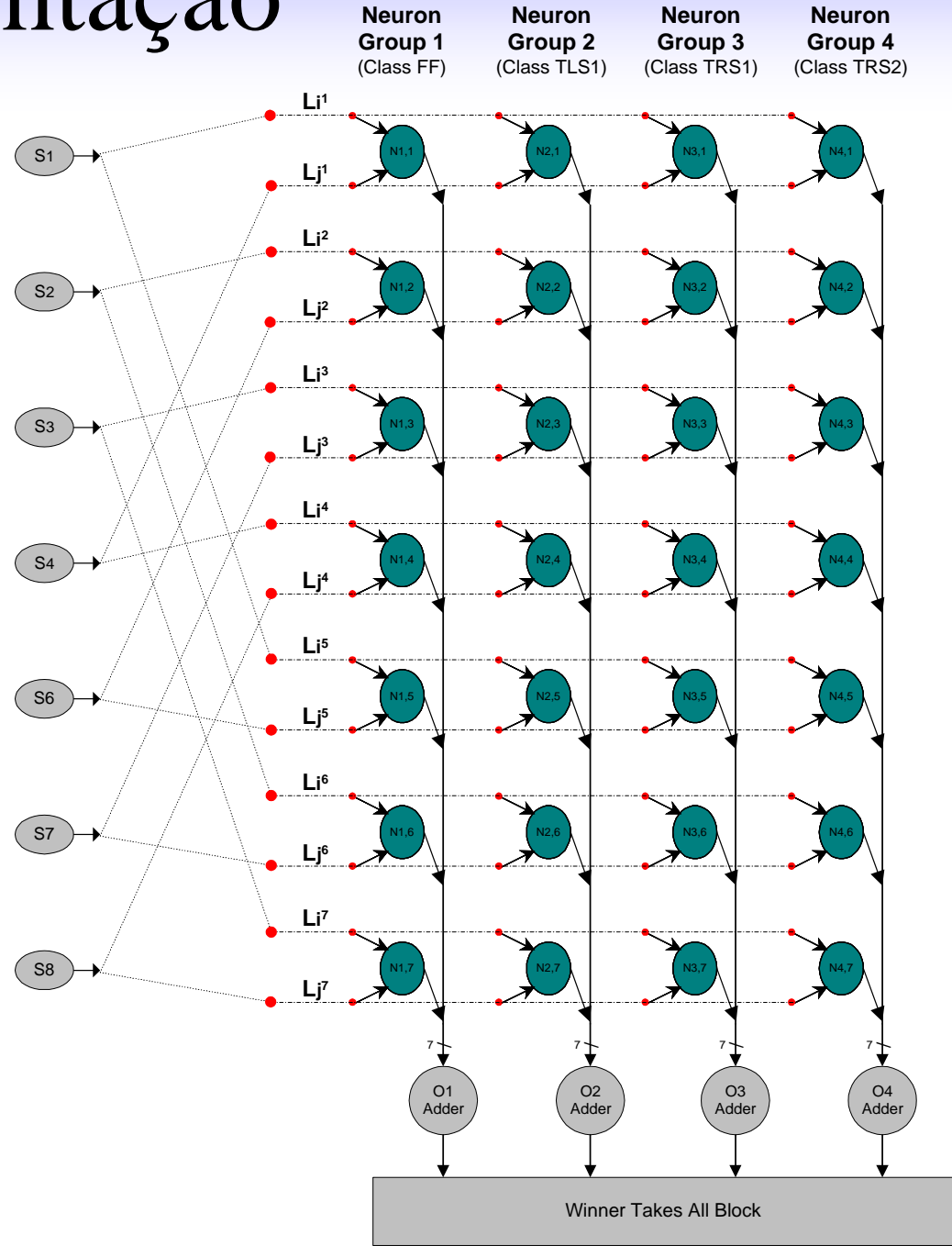
## ■ Controle por Solução Tradicional

*Left = Right = 0;*  
*If (Sensor4=1) then Left = Left + 1;*  
*If (Sensor3=1) then Left = Left + 1;*  
*If (Sensor2=1) then Left = Left + 1;*  
*If (Sensor6=1) then Right = Right + 1;*  
*If (Sensor7=1) then Right = Right + 1;*  
*If (Sensor8=1) then Right = Right + 1;*  
*If (Left > Right) then Command = TRS1;*  
*If (Left = Right) then Command = FF;*  
*If (Left < Right) then Command = TLS1;*  
*If (Sensor1=1) then Command = TRS2;*



# 2.2- Implementação

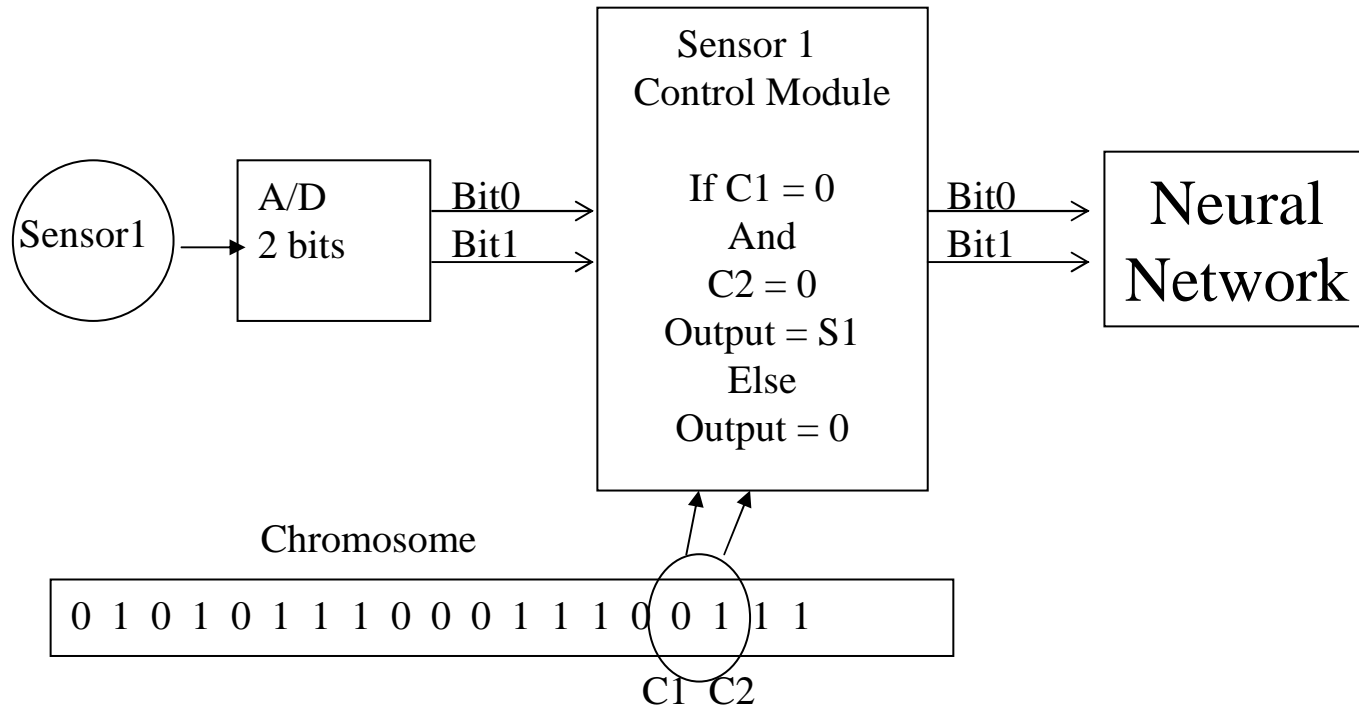
## ■ Controle por Rede Neural





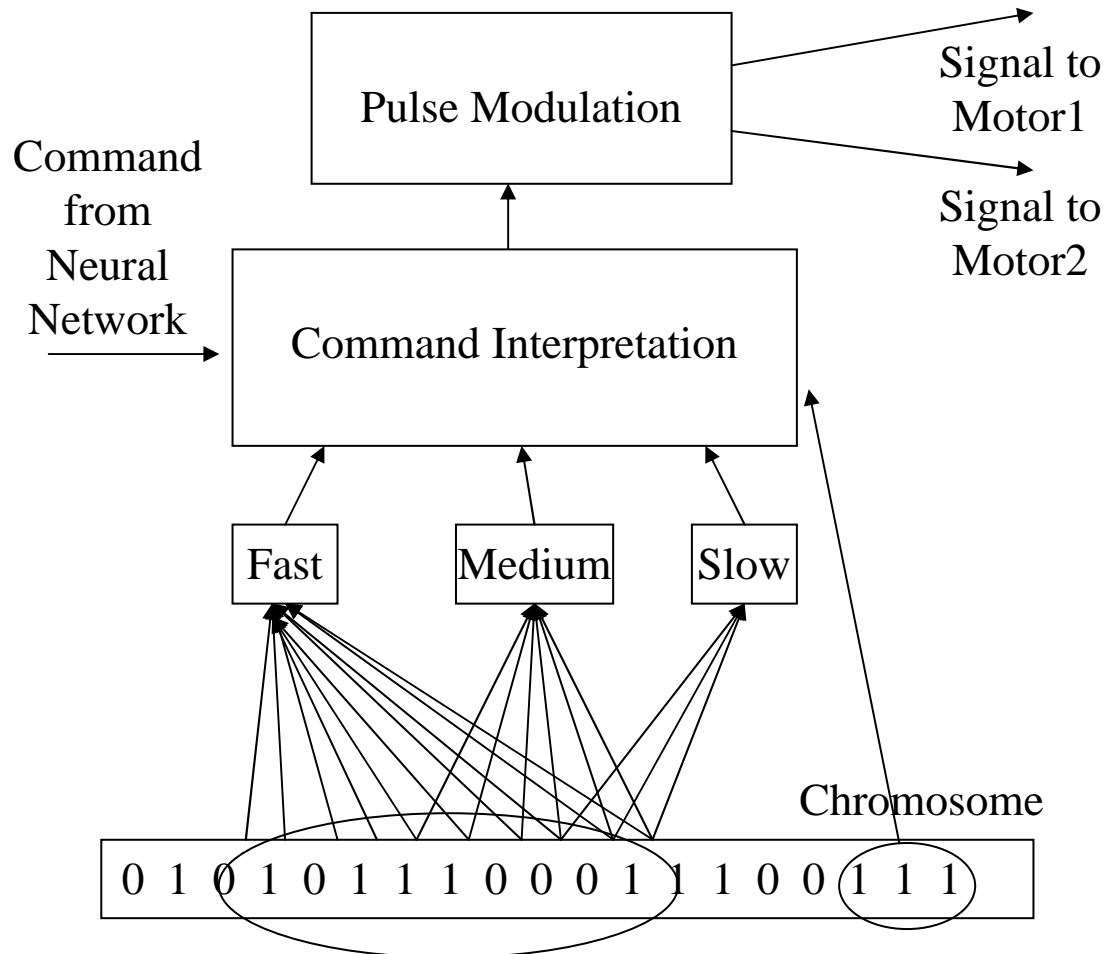
# 2.2- Implementação

## Sensor Control Module



# 2.2- Implementação

## Motor Control Module



## 2.2- Implementação

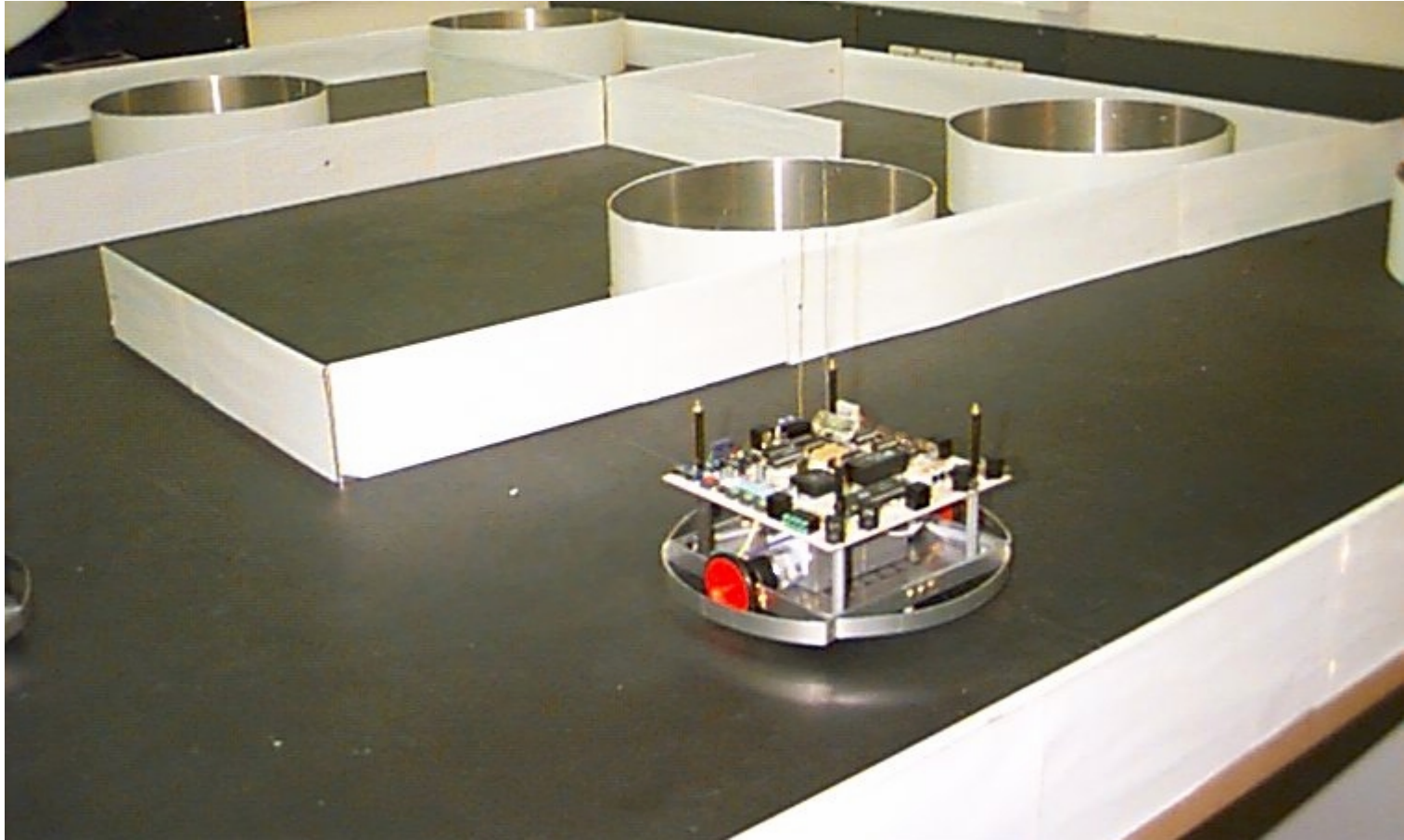
### ■ Função de *Fitness*

1- *Começa com 5000 pontos;*

2- *Recompensa: + 1 pontos para cada 1 seg. de movimento à Frente;*

3- *Punição: - 10 pontos a cada colisão.*

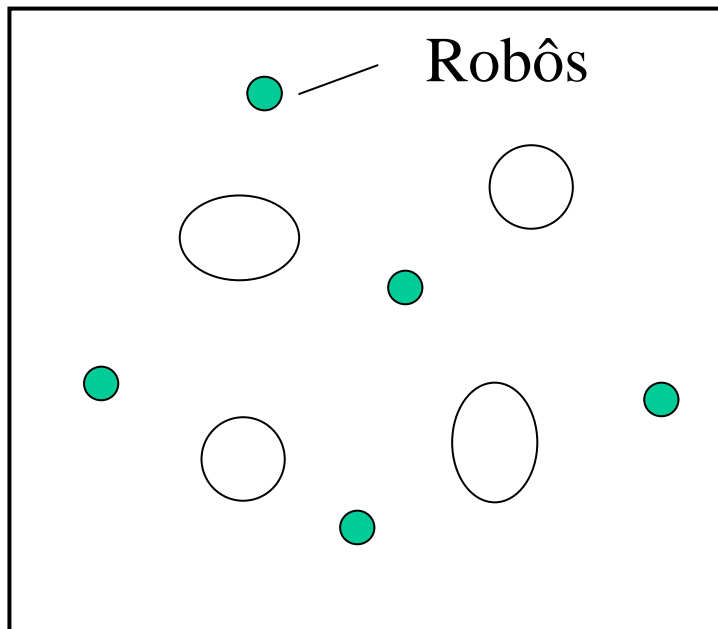
## 2.3- Primeiros Experimentos



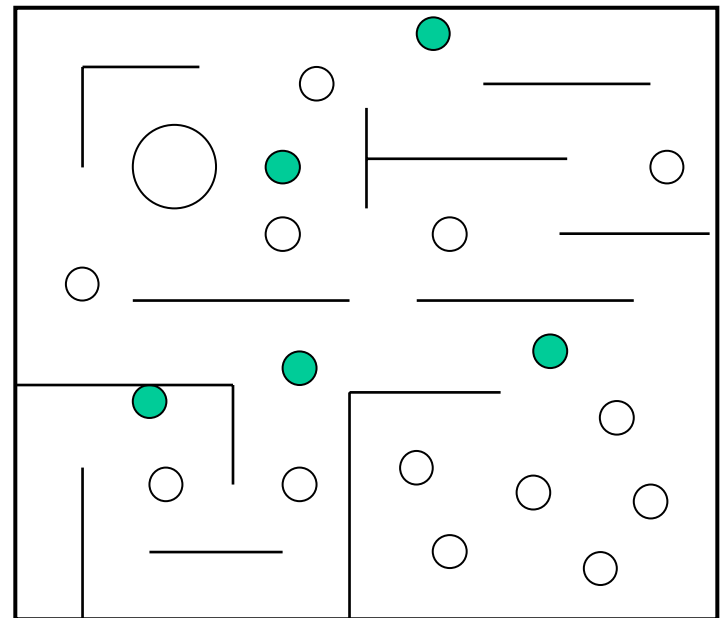
## 2.3- Primeiros Experimentos

- Objetivo: Navegação sem Colisões

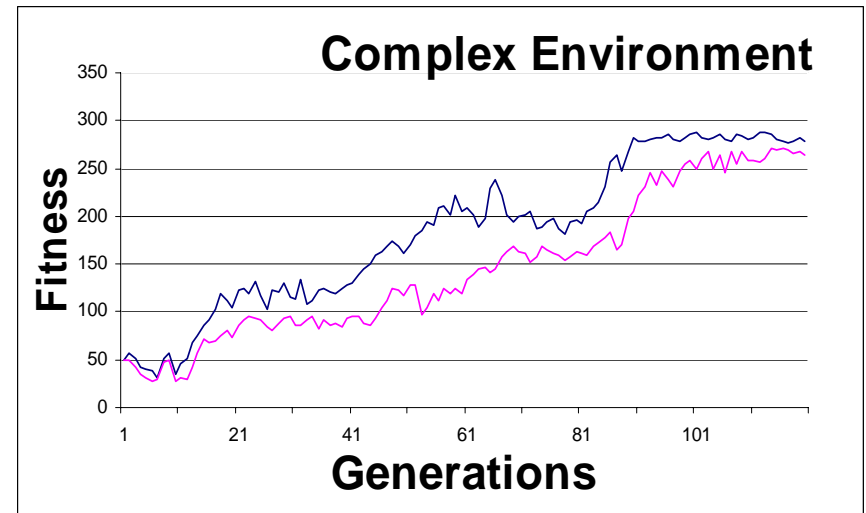
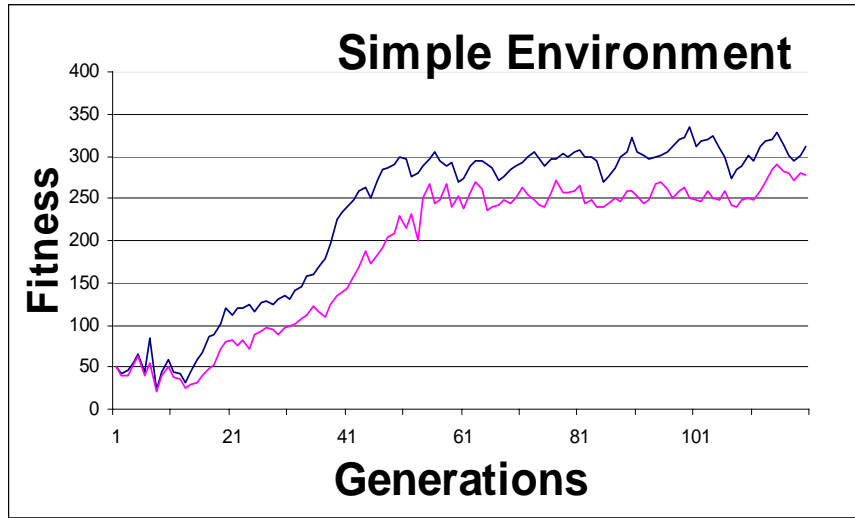
Simple



Complexo



## 2.3- Primeiros Experimentos

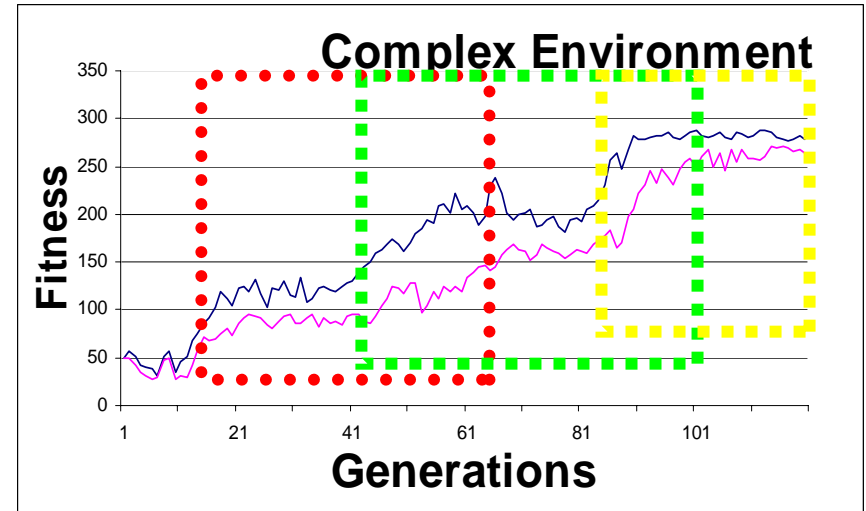
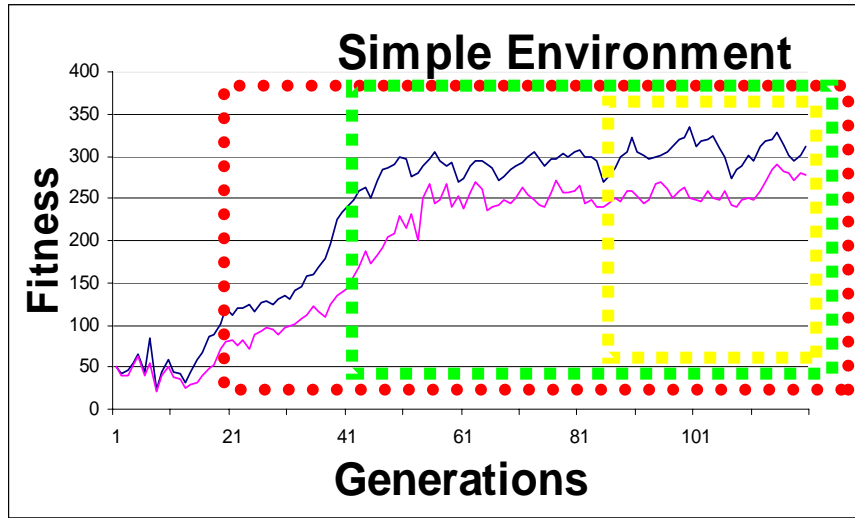


■ 120 Gerações: (1 min.)

■ Pontuação do — Melhor Robô

— Média da População

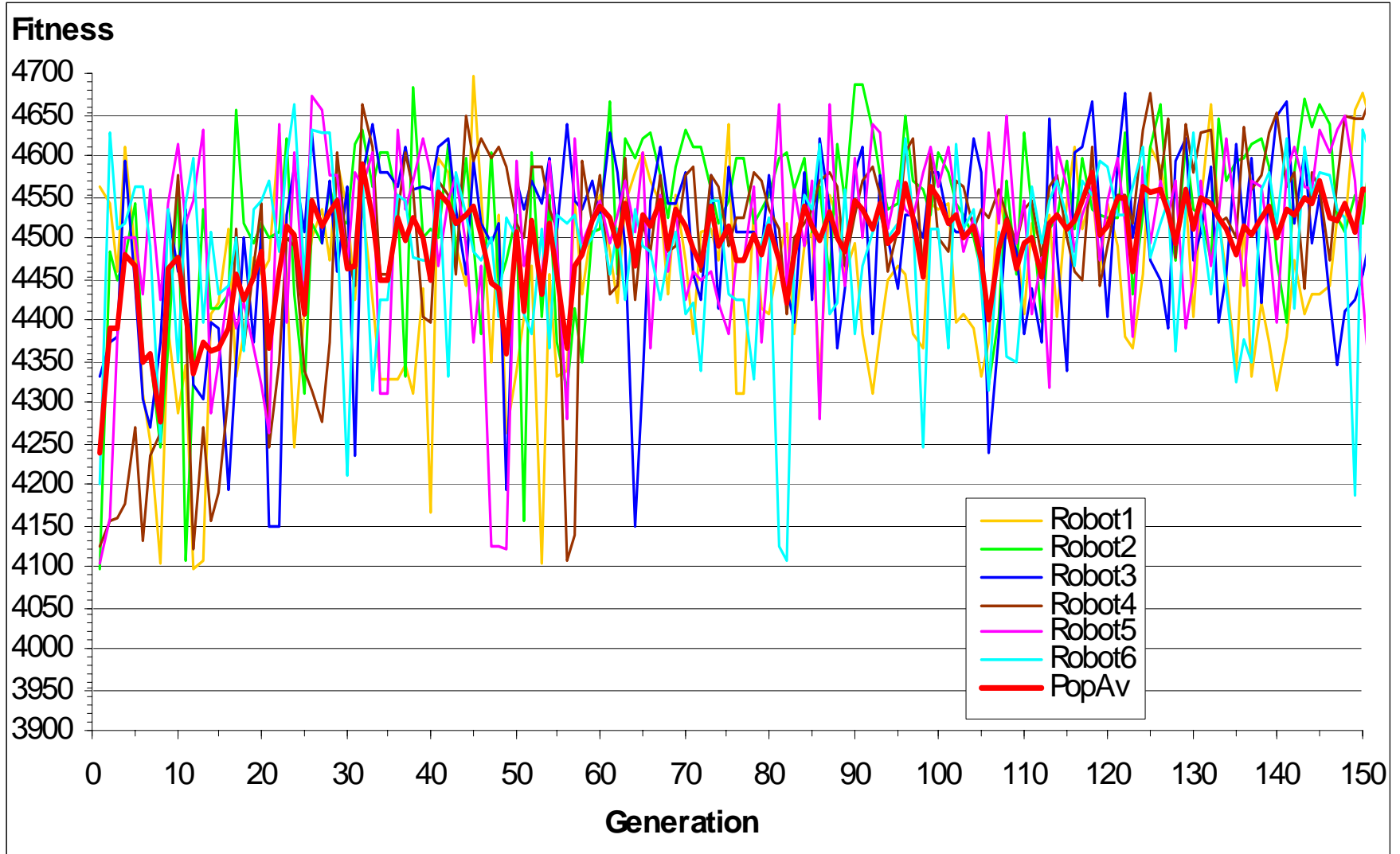
## 2.3- Primeiros Experimentos



- Espécie 1** – Um sensor frontal
- Espécie 2** – Dois sensores, um frontal e outro lateral
- Espécie 3** – Três sensores, um frontal e dois laterais

# 2.4- Analise dos Dados Experimentais

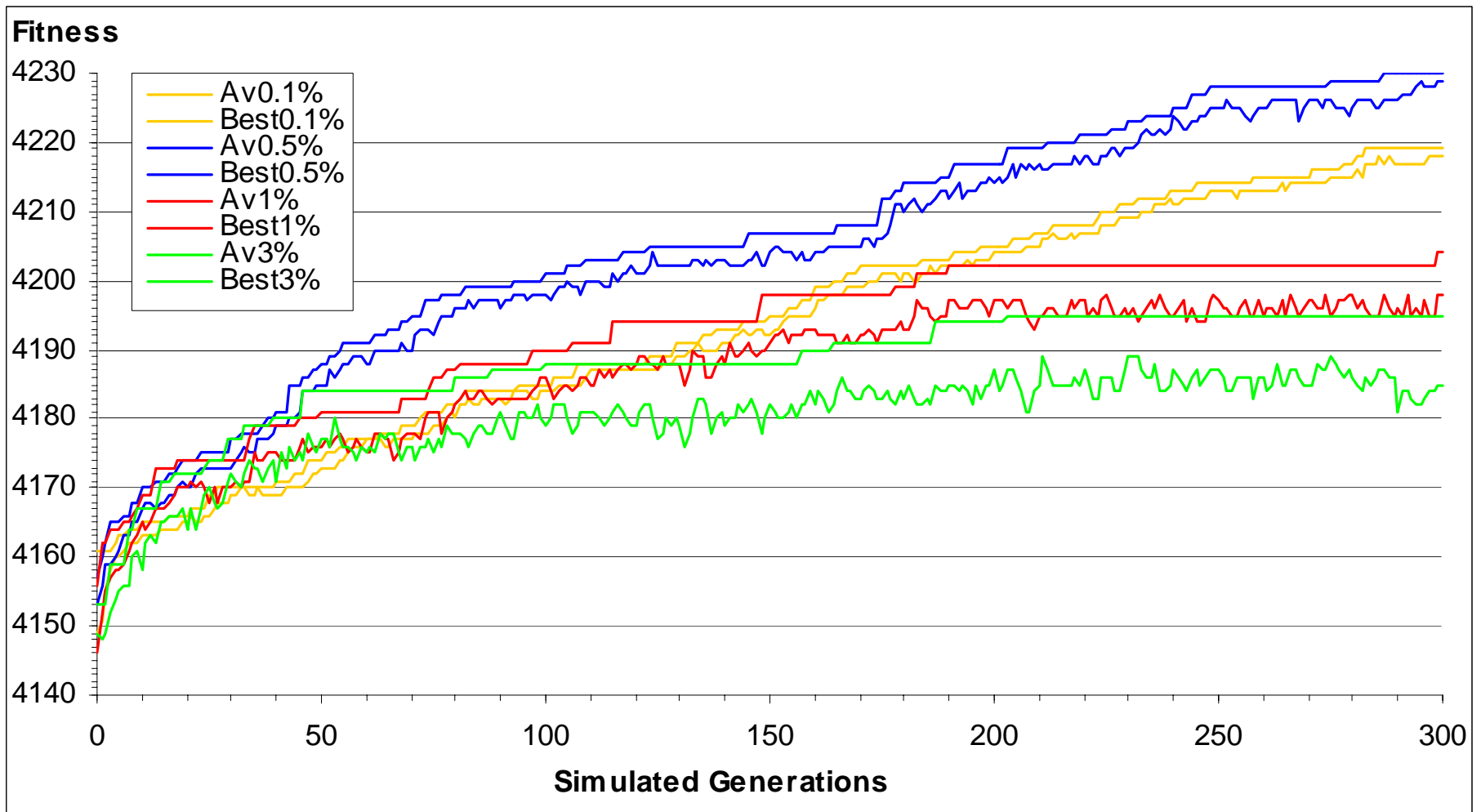
## ■ Robôs Reais





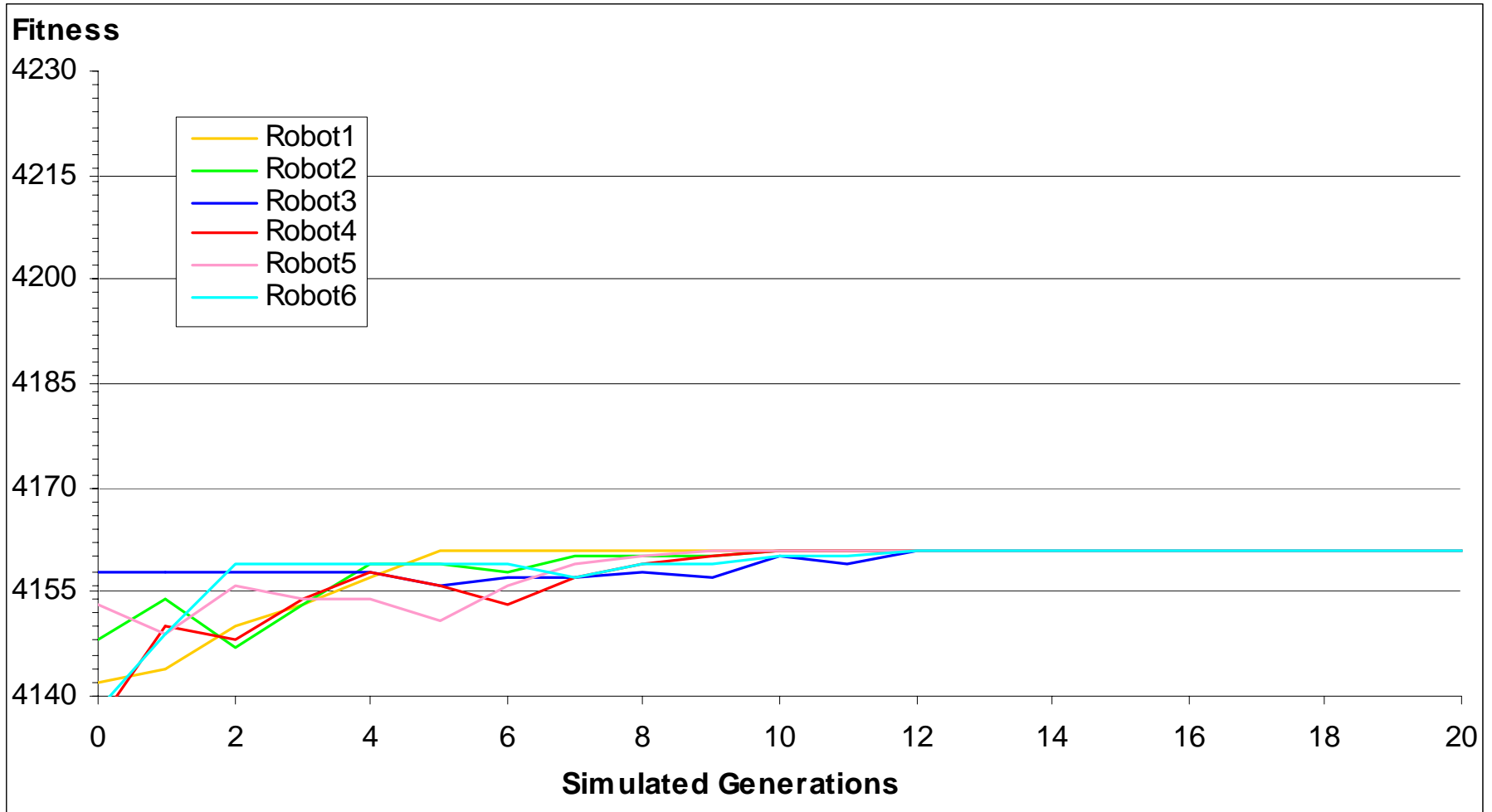
# 2.4- Analise dos Dados Experimentais

## ■ Efeito da Mutação:



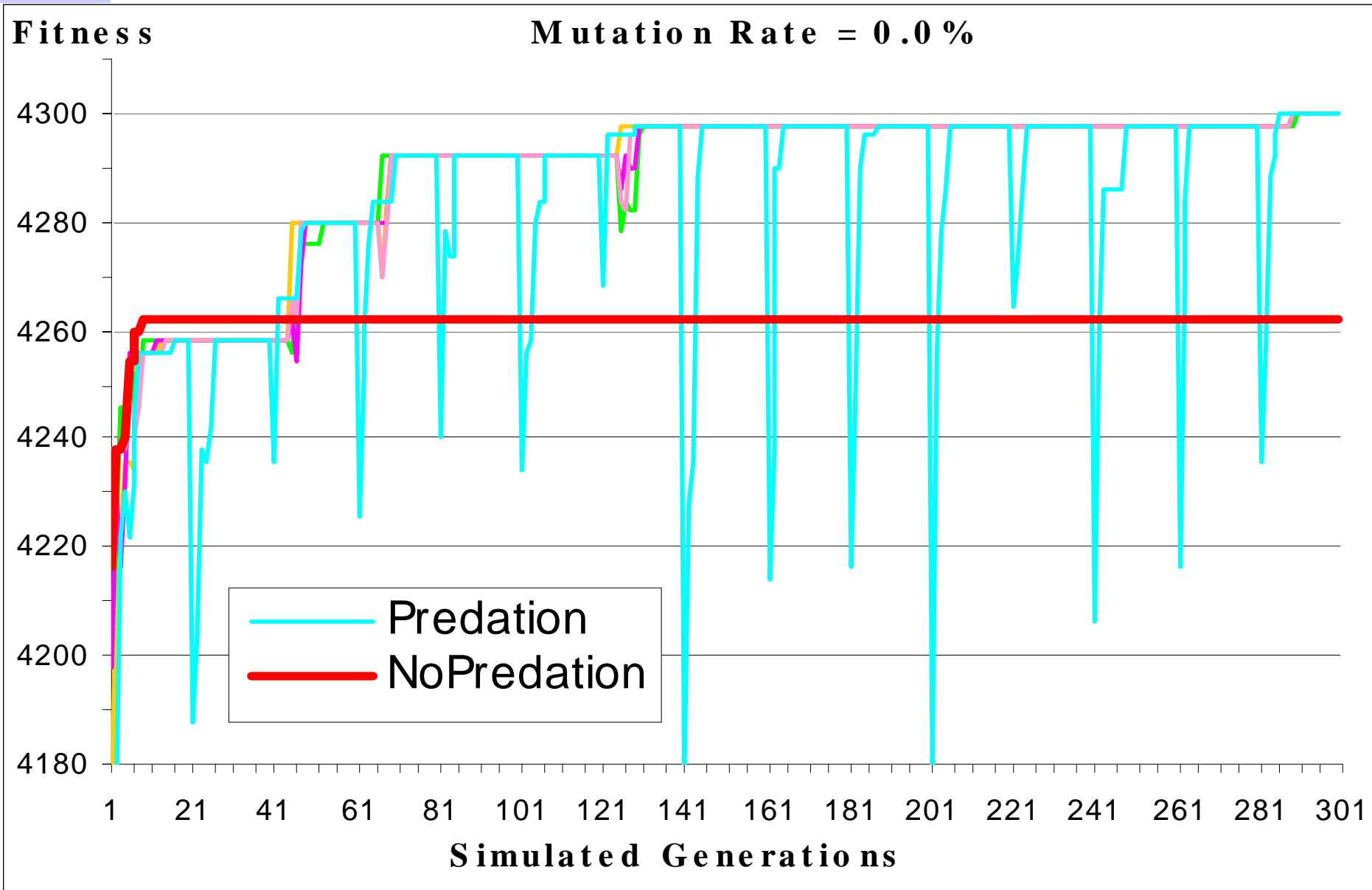
# 2.4- Analise dos Dados Experimentais

## ■ Efeito da Mutação:





# 2.5- Predação



# 3- Conclusão

# 3- Conclusão

- A Computação Evolutiva pode contribuir muito com a Robótica
  - Evolução Contínua X Busca de Solução
  - Produz soluções aceitáveis para problemas de navegação e desvio de obstáculos
  - Possibilita auto-programação de sistemas complexos
  - Construção de infraestrutura para realização de experimentos com 40 robôs móveis autônomos
  - Construção de um time de futebol contendo 5 robôs

# FIM

*Cópia das transparências e referências bibliográficas podem ser obtidas no site:*

<http://www.icmc.usp.br/~simoese/seminars>

email: [simoese@icmc.usp.br](mailto:simoese@icmc.usp.br)