

Ciber-Rato: Um Ambiente de Simulação de Robots Móveis e Autónomos

Nuno Lau, Artur Pereira, Andreia Melo, António Neves e João Figueiredo

Resumo - O Ciber-Rato é uma modalidade do concurso Micro-Rato destinada a concorrentes que apenas se queiram envolver nos aspectos algorítmicos do controlo de robots móveis e autónomos. Esta modalidade é suportada por um ambiente de simulação de robots e labirintos. Neste artigo dá-se uma visão geral deste ambiente de simulação e apresentam-se os resultados e experiências das duas edições já realizadas da modalidade Ciber-Rato.

Abstract - Ciber-Rato is a modality included in the Micro-Rato competition, directed to teams interested in the algorithmic issues and software control of mobile autonomous robots. This modality is supported by a software environment, which simulates both robots and a labyrinth. This paper gives an overview of the simulation environment and presents the results and experiences gathered from the two competitions already carried out.

I. INTRODUÇÃO

O Ciber-Rato é uma modalidade do Concurso Micro-Rato da Universidade de Aveiro. O Concurso Micro-Rato é uma competição entre robots pequenos e autónomos que tentam resolver um labirinto.

O Concurso Micro-Rato foi lançado em 1995 apenas com a modalidade Micro-Rato, tendo alcançado um sucesso assinalável ao longo das suas várias edições, conforme demonstra a quantidade crescente de equipas concorrentes [1,2]. Na modalidade Micro-Rato, a tarefa dos concorrentes envolve quer o desenvolvimento do *hardware*, quer o desenvolvimento do *software* do robot. O desenvolvimento do *hardware*, ou seja, a estrutura mecânica, os motores, os sensores, o processador, a electrónica de interface, etc, é uma tarefa que exige conhecimentos em várias áreas e para a qual nem todas as pessoas estão vocacionadas.

Sendo assim, o Concurso Micro-Rato foi enriquecido a partir da sua edição de 2001 com uma nova Modalidade, o Ciber-Rato. O objectivo desta modalidade é permitir que as equipas que nela participam se possam concentrar nas componentes algorítmica e de *software* da construção de um robot, eliminando os problemas associados ao projecto, montagem e teste de um robot real. No Ciber-Rato a competição é suportada por um ambiente de simulação, que modela todos os componentes de hardware dos robots e do ambiente da prova. Os concorrentes desenvolvem agentes de *software* que comandam os robots simulados.

Na primeira edição do Ciber-Rato, em 2001, participaram 5 equipas. Em Abril de 2002 decorreu a 2ª edição com a participação de 13 equipas.

II. REGRAS DA COMPETIÇÃO

As regras da Modalidade Ciber-Rato [3] são semelhantes às regras iniciais da Modalidade Micro-Rato. Três robots competem simultaneamente no mesmo labirinto, tentando resolvê-lo. Colocados num determinado ponto de partida, os robots devem percorrer um labirinto com o objectivo de atingir e imobilizar-se numa “zona de chegada”. Durante o percurso os robots encontram obstáculos fixos - paredes - e obstáculos móveis - outros robots - com os quais não podem chocar, sob pena de serem penalizados.

O labirinto é um quadrado de dimensões pré-definidas, onde foram colocados diversos obstáculos, e onde está definida a “grelha de partida” e a “zona de chegada”. Inicialmente os robots são colocados em posições sorteadas na “grelha de partida”. A “zona de chegada” é um círculo de côr diferente, no centro do qual está colocado um farol. O objectivo dos robots é atingir a “zona de chegada” o mais rapidamente possível, sem incorrer em penalizações. A classificação dos robots é obtida por ordem crescente das pontuações obtidas.

No fim de cada prova é atribuída a cada concorrente uma pontuação resultante da soma de duas parcelas. Uma parcela corresponde ao tempo que o concorrente levou até se imobilizar na área de chegada e a outra ao total de penalizações sofridas.

O concurso Ciber-Rato desenrola-se em várias mangas, sendo cada manga, por sua vez, dividida em provas de 3 concorrentes. A distribuição dos concorrentes pelas provas de cada manga é aleatória. Para cada manga são definidos um labirinto e um tempo máximo de prova. A passagem à manga seguinte depende dos resultados obtidos nas mangas anteriores.

III. ARQUITECTURA DO SISTEMA DE SIMULAÇÃO

O sistema de simulação possui uma arquitectura distribuída em que vários tipos de aplicações comunicam entre si, nomeadamente, um simulador, uma aplicação por cada concorrente em prova e ainda uma aplicação de visualização (figura 1). A arquitectura é do tipo cliente-servidor, funcionando o simulador como servidor e os agentes e o visualizador como clientes. Esta arquitectura é

semelhante à do simulador usado na Liga de Simulação do RoboCup [4,5].

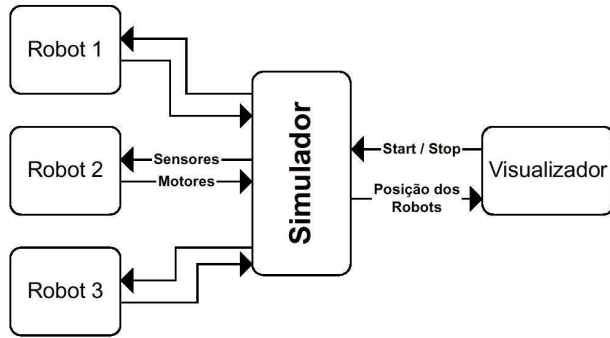


Figura 1: arquitetura do sistema de simulação.

As várias aplicações comunicam através de um protocolo comum, utilizando *sockets* UDP [3]. A utilização de *sockets* permite que as aplicações possam residir em computadores distintos, ou seja, cada agente pode estar a correr no seu computador, o simulador pode estar a correr noutra e o visualizador ainda num diferente.

Os concorrentes podem escolher qual a Linguagem de Programação e o Sistema Operativo que pretendem usar para os seus agentes, desde que respeitem o protocolo de comunicação definido.

Todas as ferramentas do sistema de simulação: simulador, visualizador, biblioteca de comunicações e agente robótico exemplo estão disponíveis no Arquivo Técnico Ciber-Rato [6].

A. O simulador

O simulador modela todos os componentes de hardware dos robots e do labirinto. É ainda responsável por fazer cumprir as regras do concurso aplicando penalizações sempre que necessário. A simulação é executada em tempo discreto, ciclo a ciclo.

No início de cada ciclo de simulação o simulador envia a todos os agentes robóticos em prova as medidas dos seus sensores, e a todos os visualizadores as posições e pontuações dos robots. Os agentes podem responder com os valores da potência a aplicar aos motores que comandam as rodas (ver secção B). No fim do ciclo o simulador actualiza as posições de todos os robots em prova, considerando os comandos recebidos, e calcula os valores dos sensores para as novas posições iniciando um novo ciclo (figura 2).

As posições dos robots não são alteradas de forma contínua. Durante o processo de actualização das posições que decorre no fim de cada ciclo de simulação, o robot “salta” da sua posição no ciclo anterior para a sua posição no ciclo seguinte.

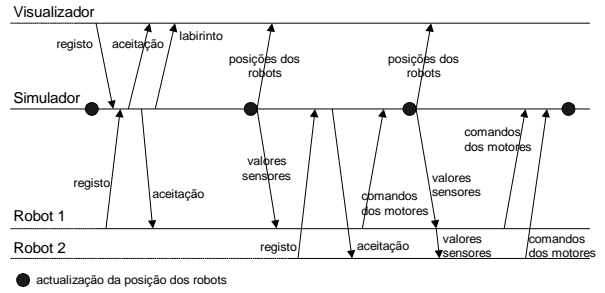


Figura 2: Mensagens trocadas com o simulador pelos agentes robóticos e visualizador

O simulador é uma aplicação Linux desenvolvida em C++. Um excerto do diagrama de classes do simulador pode ser visto na Figura 3. As classes CRobot e CViewer são independentes do protocolo de comunicação entre as aplicações do sistema, sendo os pormenores da implementação do protocolo tratados em CSockRobot e CSockViewer.

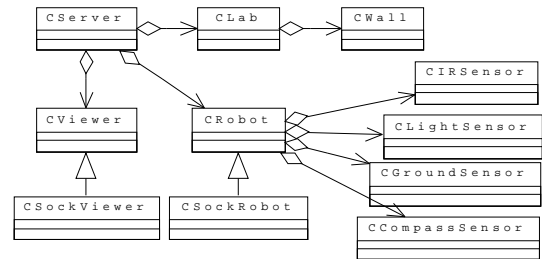


Figura 3: Diagrama de classes simplificado do simulador.

B. O robot virtual

Todos os robots em prova tem exactamente as mesmas características físicas [3]. Todos têm os mesmos sensores e os motores são iguais para todos.

Cada robot (figura 4) está equipado com os seguintes sensores:

- 3 sensores de proximidade orientados para a frente e a 60° para cada lado.
- Sensor de farol que lhe indica qual a diferença entre a sua direcção e a direcção do farol.
- Sensor de chão, activo quando o robot entra na zona de chegada
- Bússola, que permite ao robot saber qual a sua orientação absoluta no labirinto.
- Sensor de colisão, activo no caso de o robot colidir.

As medidas dos sensores incluem algum ruído adicionado pelo simulador de modo a modelar um sensor real.

De modo a detectar o início da prova e possíveis interrupções da prova, cada robot tem 2 botões:

- Start, activo quando é iniciada a prova.
- Stop, activo quando existe uma interrupção de prova.

Em termos de actuadores os robots virtuais são constituídos por:

- 2 rodas actuadas por 2 motores independentes, uma à esquerda e uma à direita;
- um LED de finalização, a acender quando atingida a zona de chegada.

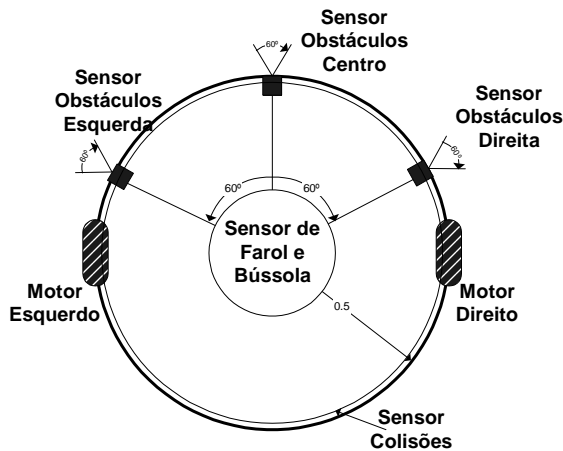


Figura 4: Constituição do robot virtual

Também os motores têm algum ruído, executando os comandos do agente com precisão limitada.

C. O visualizador

O visualizador desempenha três funções: mostra o labirinto e movimento dos robots no seu interior; apresenta um *placard* com a evolução no tempo da pontuação dos concorrentes; finalmente disponibiliza um painel de controlo que permite dar início à prova, interrompê-la, ou remover um robot mal comportado.

Nas 2 edições do Ciber-Rato foram usados 2 visualizadores distintos. No Ciber-Rato 2001 foi usado um visualizador bastante simples. No Ciber-Rato 2002 foi usado um visualizador desenvolvido no âmbito de um projecto de fim-de-curso bastante mais atractivo [7].

D. Os agentes robóticos

Em cada ciclo de simulação os agentes recebem os valores medidos em todos os seus sensores e devem decidir qual a potência a aplicar em cada motor e se devem terminar a prova acendendo o LED de finalização.

A percepção que um agente robótico tem do ambiente exterior é relativamente limitada. Na figura 5 está representada a zona livre, resultante da interpretação dos valores dos sensores de proximidade, através de arcos, e as direcções do farol e absoluta, através de setas. Nesta situação o robot sente um obstáculo à sua direita, tendo também o farol à sua direita, no entanto quer a zona livre quer a direcção do farol e da bússola podem ter erros resultantes do ruído.

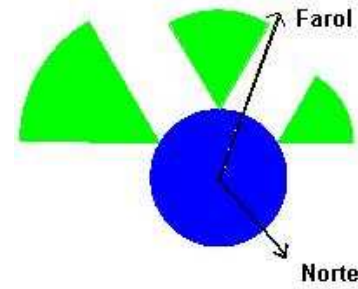


Figura 5: percepção do exterior de um robot

Os agentes podem ser programados em qualquer Linguagem de programação/Sistema Operativo. Existem neste momento bibliotecas de comunicação com o simulador em C/Linux, C/Windows, Delphi/Windows e Prolog/Linux [6]. Estas bibliotecas incluem várias primitivas [3] do tipo:

```
void InitRobot(char *rob_name, char *host);
float GetObstacleSensor(int pos);
float GetBeaconSensor(void);
void DriveMotors(float LeftPow, float RightPow);
```

Que tornam a tarefa dos concorrentes bastante mais simples, escondendo os pormenores do protocolo de comunicação com o simulador.

IV. CIBER-RATO 2001

O Ciber-Rato 2001 decorreu em 23 de Maio de 2001, concorreram 5 equipas, 1 da Universidade de Aveiro e 4 da Universidade Fernando Pessoa. Em termos de Linguagens de Programação/Sistema Operativo, a equipa de Aveiro utilizou C/Linux, enquanto as restantes 4 utilizaram Delphi/Windows. Os resultados são apresentados na tabela 1.

	Equipa	Origem	Linguagem/S.O.
1.	Ciber-Pessoa	UFP	Delphi/Windows
2.	Mauzz	UA	C++/Linux
3.	Midalp	UFP	Delphi/Windows
4.	Lupama	UFP	Delphi/Windows
5.	Mocho	UFP	Delphi/Windows

Tabela 1: Resultados Ciber-Rato 2001.

V. CIBER-RATO 2002

O Ciber-Rato 2002 decorreu em 26 de Abril de 2002, concorreram 13 equipas de vários pontos do país. Nesta edição participaram 2 equipas provenientes de escolas secundárias, o que aconteceu pela primeira vez, tendo participado ainda 6 equipas de politécnicos e 5 equipas de universidades (ver tabela 2). Nesta tabela são apresentados as Linguagens de Programação e os Sistemas Operativos utilizados pelos concorrentes. A tabela está ordenada pela classificação final obtida no concurso.

	Equipa	Origem	Linguagem/S.O.
1.	YAM	FCUP	C/Linux
2.	Ciber-FEUP	FEUP	Delphi/Windows
3.	R_Zero	ISEP	Delphi/Windows
4.	Nai	FCUP	C/Linux
5.	Hamster	ESTCB	C/Linux
6.	Ciber-Pessoa1	UFP	Delphi/Windows
7.	Encurralado	ESTCB	C/Linux
8.	Ciber-Pessoa2	UFP	Delphi/Windows
9.	12 C	Ancorensis	Delphi/Windows
10.	Fuinha	ESTGL	C/Linux
11.	11 C	Ancorensis	Delphi/Windows
12.	Estraquinas	ESTCB	C/Linux
13.	Speedy Gonzalez	FEUP	C/Linux

Tabela 2: Resultados Ciber-Rato 2002.

Nesta edição foi usado um novo visualizador [7], tendo sido também alterado o protocolo de comunicações que passou a ser baseado em mensagens XML. Na figura 6 são apresentados os labirintos usados nas 4 mangas do Ciber-Rato 2002, usando o visualizador de 2002.

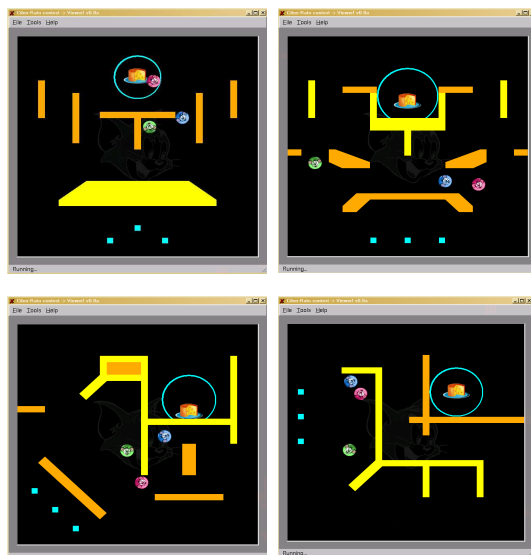


Figura 6: Labirintos usados no CiberRato 2002

VI. CONCLUSÃO

A utilidade de um ambiente de simulação tipo Ciber-Rato ultrapassa a sua utilização no âmbito de uma competição. Pode também ser útil como ferramenta pedagógica no âmbito de cadeiras curriculares do ensino universitário, politécnico e secundário. O ambiente de simulação Ciber-Rato foi já usado na Universidade de Aveiro, Faculdade de Engenharia da Universidade do Porto, Universidade

Fernando Pessoa e Escola Superior de Tecnologia de Castelo Branco no âmbito de cadeiras de Programação, Inteligência Artificial e Robótica.

Por outro lado, a diversidade das origens e o aumento, relativamente a 2001, do número dos concorrentes participantes no Ciber-Rato 2002 mostra que é uma iniciativa bem sucedida. Os comentários formulados pelos participantes no fim do concurso foram, em geral, bastante positivos.

À primeira vista, a programação de um agente robótico para o Ciber-Rato parece tarefa simples. No entanto, foram apresentados pelos concorrentes estratégias de navegação diversas e bastante elaboradas, o que reforça a ideia de que este ambiente é uma boa plataforma de trabalho para fins pedagógicos, ensaios de algoritmos a ser implementados em robots reais e, claro, como suporte a futuras edições do Concurso Ciber-Rato.

A qualidade das estratégias apresentadas pelos concorrentes (ver outros artigos nesta edição da Revista), funciona como um incentivo à redefinição das regras do Concurso, de forma a tornar o desafio proposto às equipas cada vez mais aliciante.

AGRADECIMENTOS

Cabe aqui deixar um voto de agradecimento a todas as individualidades e instituições que de uma forma ou de outra apoiaram esta iniciativa. Aos participantes porque sem eles não faria sentido tudo isto. Ao público pelo seu entusiasmo. Em particular, gostaríamos de agradecer aos restantes elementos da Organização do Concurso Micro-Rato e aos membros dos Jurís do Concurso.

REFERÊNCIAS

- [1] Luís Almeida, Pedro Fonseca, José Luís Azevedo, Bernardo Cunha, "The Micro-Rato Contest: Mobile Robotics for All", Proceedings of CONTROLO'2000, The Portuguese Control Conference, Guimarães, Outubro 2000.
- [2] Concurso Micro-Rato, <http://microrato.ua.pt/>
- [3] Regras e Especificações Técnicas da Modalidade Ciber-Rato, <http://microrato.ua.pt/main/Docs/RegrasCiberRato2002.pdf>
- [4] Noda, I.; Matsubara, H.; Hiraki, K.; Frank, I.; "SoccerServer: A Tool for Research in Multiagent Systems", Applied Artificial Intelligence, 12(3): 233-250.
- [5] RoboCup Official Site, <http://www.robocup.org/>
- [6] Arquivo Técnico Ciber-Rato, <http://microrato.ua.pt/main/documentos.htm#ArqTecCR>
- [7] António José Neves, João Pedro Figueiredo, "Ferramentas de Apoio ao Concurso Ciber-Rato", Relatório da disciplina de Projecto da Licenciatura em Engenharia Electrónica e de Telecomunicações da Universidade de Aveiro, 2002.