

Atividade 3 – Operadores para recarga de recursos

Nesta atividade foi proposto o desenvolvimento de operadores que permitissem que os agentes buscassem recarregar sua energia e saúde utilizando o conhecimento da posição dos carregadores destes recursos obtidos pela navegação no mapa.

Para isto, foi definido o operador (e sub estado) 'recharge' que deve ser aplicado quando um dos recursos estiver com baixa quantidade. O primeiro passo foi criar uma elaboração que definisse no estado superior quais recursos se encontram escassos no agente. Abaixo, o código desta elaboração:

sp {elaborate*state*resource*low	
(state <s> ^name tanksoar	
^io.input-link. { << energy	se houver recurso
health >> <resource> } <= 200)	<resource> dentre
	(energia, health) com
	baixo valor
-->	
(<s> ^low-resource <resource>)}	Augumenta o super estado
	com o valor do recurso
	escasso

A seguir, foram criadas as regras de proposição e seleção do operador dentre as outras regras:

#####

sp {propose*recharge	
(state <s> ^name tanksoar	
^low-resource <res>)	se foi identificado que o
	agente possui um recurso
	escasso
-->	
(<s> ^operator <o> + =)	Cria um operador de nome
(<o> ^name recharge)	'recharge'
}	

sp {select*recharge*wander*chase	Determina a preferência de
	'recharge' sobre 'wander' e
	'chase'
(state <s> ^name tanksoar	
^operator <o1> +	
^operator <o2> +)	
(<o1> ^name recharge)	
(<o2> ^name << wander chase >>)	
-->	
(<s> ^operator <o1> > <o2>)}	

sp {select*attack*retreat*recharge	Determina a preferência
	de 'attack' e 'retreat'
	sobre 'recharge'
(state <s> ^name tanksoar	

<code>^operator <o1> +</code>	
<code>^operator <o2> +)</code>	
<code>(<o1> ^name recharge)</code>	
<code>(<o2> ^name << retreat attack >>)</code>	
<code>--></code>	
<code>(<s> ^operator <o2> > <o1>)} </code></code>	

Se um recurso é escasso e foi determinado que ele deve ser recarregado, é necessário encontrar o carregador ou se dirigir a ele caso já tenha sido encontrado. Para estas funções, a abordagem utilizada foi aplicar o operador *wander* já existente enquanto o carregador do recurso necessário não for encontrado e um método de busca de caminho no mapa para o carregador já encontrado.

Antes de mais nada, é preciso manter através de uma elaboração a informação de quais carregadores já foram encontrados e suas localizações. A elaboração abaixo serve a esta função:

<code>sp {recharge*elaborate*knows*resource*recharger</code>	
<code>(state <s> ^name tanksoar</code>	
<code>^map.square <square>)</code>	Para todo quadrado do mapa
<code>(<square> ^ { << health energy >> <resource> } *yes*)</code>	Determina que recurso pode ser carregado neste quadrado
<code>--></code>	
<code>(<s> ^recharger <r>)</code>	Adiciona uma aumentação ' <i>recharger</i> ' no estado
<code>(<r> ^square <square></code>	com o quadrado onde se encontra o carregador
<code>^resource <resource>)</code>	E o tipo do recurso que recarrega
<code>}</code>	

O próximo passo é utilizar esta informação para selecionar um operador de vagueio ou de deslocamento até o carregador se este já foi encontrado.

<code>sp {recharge*propose*wander</code>	Propõe um operador de vagueio se há um recurso escasso sem recarregador conhecido
<code>(state <s> ^name recharge</code>	se o estado é recharge
<code>^top-state <ts>)</code>	seja <ts> o estado superior
<code>(<ts> ^low-resource <resource></code>	Seja <resource> um recurso escasso
<code>-^recharger.resource <resource>)</code>	Se não é conhecido um carregador para este recurso
<code>--></code>	
<code>(<s> ^operator <op> + =)</code>	

(<op> ^name wander)	Propõe um operador de nome wander
}	

sp {recharge*propose*goto-square	Propõe um operador que leva o agente de um quadrado a outro no mapa com um caminho conhecido
(state <s> ^name recharge ^top-state <ts>)	
(<ts> ^recharger <recharger> ^low-resource <resource>)	Seja <recharger> um carregador conhecido Seja <resource> um recurso escasso
(<recharger> ^resource <resource> ^square <square>)	se <recharger> é um carregador para este recurso Guarda o quadrado onde se encontra o carregador
-->	
(<s> ^operator <op> + =)	
(<op> ^name goto-square ^target <square>)	Propõe um operador que deve levar o agente de onde quer que esteja para <square> utilizando seu mapa interno
}	

No cenário de TankSoar, se um carregador é conhecido então necessariamente um caminho aberto é conhecido de onde o agente se encontra até o quadrado do carregador. Ao desenvolvermos um comportamento que tem a capacidade de computar um caminho de um quadrado a outro dado um mapa, podemos fazer o agente efetuar a tarefa. Este operador poderia ser usado em diversas outras tarefas e portanto deve ser genérico. O desenvolvimento de um operador como este foge do escopo desta atividade, por isso não foi desenvolvido. Em atividades futuras serão exploradas formas de busca que cobrirão este aprendizado.