

Aplicação da Teoria de Controle Supervisório em um Robô de Sumô Autônomo

Kallil M. C. Zielinski¹, César R. C. Torrico¹, Fábio Favarim¹

¹Universidade Tecnológica Federal do Paraná - Câmpus Pato Branco

kallil@alunos.utfpr.edu.br, {cesartorrico, favarim}@utfpr.edu.br

Abstract. *This article describes a method of automation based on the supervisory control theory of discrete event systems (DES) developed by Ramadge & Wonham, using as application a 3 kg sumo robot. The modeling was elaborated in two stages, the first one encompasses the free behavior of the robot, and the second one represents the specifications or restrictions of the robot behavior. In the sequence, the supervisory control theory was applied to obtain the minimally restrictive behavior of the robot that meets the desired specifications.*

Resumo. *Este artigo descreve uma metodologia de automação baseada na teoria de controle supervisório de sistemas a eventos discretos (SED) desenvolvida por Ramadge e Wonham, utilizando como aplicação um robô de sumô de 3 kg. A modelagem foi elaborada em duas etapas, a primeira engloba o comportamento livre do robô, e a segunda representa as especificações ou restrições de comportamento do robô. Na sequência foi aplicada a teoria de controle supervisório para obter o comportamento do robô minimamente restritivo que atenda às especificações desejadas.*

1. Introdução

A robótica é um ramo da tecnologia que engloba muitas áreas de conhecimento, entre elas a mecânica, a eletrônica e a computação, com a finalidade de desenvolver máquinas capazes de tomar suas próprias decisões, tornando-as autônomas, os robôs. É comum hoje em dia ver robôs industriais que soldam, pintam, e movimentam grandes peças. Há também robôs que desativam minas, procuram por sobreviventes em zonas devastadas e que realizam tarefas agrícolas. Isso tudo permite à robótica se instalar no entorno trazendo segurança e qualidade.

Com o passar dos anos foi introduzido na indústria, de modo significativo, os robôs manipuladores, cujo trabalho consiste frequentemente em tarefas repetitivas sem que os mesmos se movessem além da área máxima de extensão de suas articulações. De modo a assumir outra linha de atuação alternativa à dos robôs manipuladores, foram criados os primeiros Veículos Guiados Automaticamente (*Auto Guided Vehicles* - AGVs), responsáveis pelo transporte de carga no interior de fábricas. Desse modo, havia vários veículos navegando o ambiente estrutural, com capacidade sensorial e raciocínio mínimos, fazendo com que fosse executada uma sequência de ações que, após concluída, o veículo supõe que alcançou o objetivo a qual estava programado [Secchi 2008].

Assim, surgem os robôs móveis, caracterizados como capazes de determinar a relação com seu ambiente de trabalho através do sistema sensorial que possui. A capacidade de percepção do robô se baseia em toda a informação que os sensores o oferecem,

com o objetivo de gerar um mapa do ambiente na qual há uma sequência de ações para cada diferente informação sensorial. Existem vários tipos de robôs móveis com diferentes funções, entre elas se encontra a de lutador de sumô, na qual se baseia este projeto.

O sumô de robôs possui tal nomeação por se assemelhar com o sumô humano, na qual o objetivo da partida é desenvolver estratégias para conseguir empurrar o adversário para fora da arena denominada Dojô. Essa arena possui um formato circular e está quase totalmente revestida da cor preta, exceto por sua borda, que possui a cor branca para facilitar a percepção da borda pelos sensores do robô, conforme apresenta-se na Figura 1.

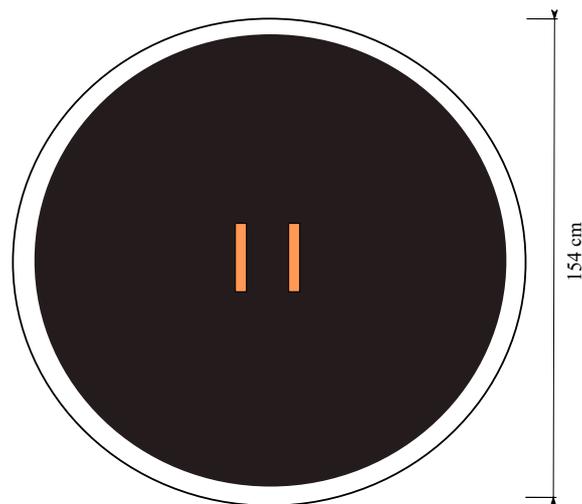


Figura 1. Área de combate dos robôs(Dojô)

Para que o robô cumpra seu objetivo de empurrar o adversário para fora da arena, a Teoria de Controle Supervisório (TCS) de Sistemas a Eventos Discretos, desenvolvida por [Ramadge and Wonham 1989] será utilizada. Na literatura, existem trabalhos relacionados como em [Torricco et al. 2016], no qual foi desenvolvido um aplicação para robôs de sumô utilizando a TCS, porém esse trabalho trata um número menor de sensores, consequentemente uma lógica diferente e menos elaborada para o desenvolvimento da dinâmica do robô.

Este documento está estruturado como segue: na Seção 2 encontra-se uma apresentação da TCS; na Seção 3 apresentam-se as etapas de modelagem do robô; por fim na Seção 4 apresentam-se as conclusões.

2. Teoria de Controle Supervisório

De um ponto de vista formal, um Sistema a Eventos Discretos (SED) pode ser pensado como um sistema dinâmico que evolui de acordo com a ocorrência abrupta de eventos físicos, em intervalos de tempo em geral irregulares e desconhecidos [Ramadge and Wonham 1982]. Ao contrário de sistemas que possuem eventos ocorrendo de maneira contínua no tempo.

É possível representar o modelo de um SED utilizando autômatos, na qual podem ser representados formalmente por uma quintupla $G = (X, \Sigma, f, x_0, X_m)$, onde X é o conjunto de estados do autômato, Σ é o conjunto de símbolos que formam o alfabeto,

$f : X \times \Sigma \longrightarrow X$ é a função parcial de transição, x_0 é o estado inicial do autômato e X_m é o conjunto de estados marcados ou finais, $X_m \subseteq X$. Para a integração de vários autômatos (subsistemas), é necessário fazer a composição síncrona dos autômatos, a qual permite observar o comportamento global de todos os subsistemas trabalhando simultaneamente [Ramadge and Wonham 1989].

Para associar a um SED estruturas de controle, segundo [Ramadge and Wonham 1989], particiona-se o alfabeto de eventos Σ em $\Sigma_c \cup \Sigma_u$, em que Σ_c é o alfabeto de eventos controláveis e Σ_u é o alfabeto de eventos não controláveis. Os eventos controláveis podem ser inibidos a qualquer momento, ao contrário dos não controláveis. Alguns exemplos de eventos não controláveis são: a quebra de uma máquina, perda de um pacote em um canal de comunicação, ruídos externos em um circuito elétrico, etc. Dessa forma pode-se caracterizar os eventos não controláveis como sendo eventos nas quais não se pode evitar sua ocorrência, enquanto que os eventos controláveis são possíveis de serem evitados pelo agente de controle.

A Teoria de Controle Supervisório realiza a síntese da composição de todos os subsistemas a fim de garantir uma linguagem controlável minimamente restritiva e não bloqueante. Um autômato S que reconheça essa linguagem denota-se de supervisor e pode ser utilizado para implementação da lógica de controle. A ação de controle que S realiza é de tal forma a restringir o comportamento livre do robô de acordo com as especificações modeladas.

3. Modelagem e Controle do Robô Móvel

Nesta seção é apresentado todo o procedimento de modelagem do comportamento livre do robô e as especificações desejadas. O software utilizado para modelagem e síntese do supervisor foi o “Supremica” [Malik et al. 2017].

3.1. Descrição dos Eventos

Na Figura 2, apresenta-se o esquema do robô a ser automatizado. Para percepção do ambiente o robô dispõe de 2 sensores de faixa (s1 e s2), 5 sensores de presença (p1, p2, p3, p4 e p5) e uma botoeira BI para dar início ao confronto. A locomoção do robô é efetuada por 2 motores de corrente contínua. Toda a lógica de funcionamento foi embarcada em um microcontrolador (stm32F407VETx).

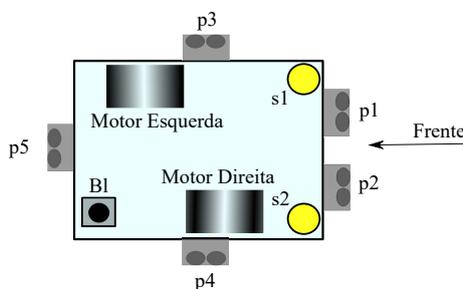


Figura 2. Nomenclatura e posicionamento dos sensores e motores do robô

A classificação de controlabilidade dos eventos foram definidos a partir do ponto de vista do microcontrolador, desta forma todos os eventos vindos dos sensores foram considerados não controláveis, pois o microcontrolador não consegue evitar a ocorrência

deles. Por outro lado, os eventos de comando dos motores foram considerados controláveis, pois é o microcontrolador quem habilita esses eventos.

A tabela 1 mostra a nomenclatura adotada para os eventos da modelagem dos autômatos.

Tabela 1. Nome e descrição dos eventos utilizados na modelagem

Eventos controláveis		Eventos não controláveis	
Evento	Descrição do Evento	Evento	Descrição do Evento
<i>f</i>	Anda para a frente	<i>Bl</i>	Botão de Início
<i>ft</i>	Anda para a frente com velocidade turbo	<i>p1a</i>	Borda de subida do sensor p1
<i>ge1</i>	Gira à esquerda travando o motor da direita	<i>p1p</i>	Borda de descida do sensor p1
<i>ge2</i>	Gira à esquerda no eixo por 90 graus	<i>p2a</i>	Borda de subida do sensor p2
<i>ge3</i>	Gira à esquerda no eixo por 120 graus	<i>p2p</i>	Borda de descida do sensor p2
<i>ge4</i>	Gira à esquerda no eixo por 180 graus	<i>p3a</i>	Borda de subida do sensor p3
<i>gd1</i>	Gira à direita travando o motor da esquerda	<i>p4a</i>	Borda de subida do sensor p4
<i>gd2</i>	Gira à direita no eixo por 90 graus	<i>p5a</i>	Borda de subida do sensor p5
<i>gd3</i>	Gira à direita no eixo por 120 graus	<i>s1</i>	Borda de subida do sensor s1
<i>gd4</i>	Gira à direita no eixo por 180 graus	<i>s2</i>	Borda de subida do sensor s2
<i>r</i>	Anda para trás		

Uma vez definido os eventos para modelagem, na sequência apresentam-se os autômatos das plantas e das especificações de forma que o sistema se comporte da maneira desejada. Para isso cada sensor do sistema teve de ser modelado de forma individual, sendo os autômatos de cada sensor plantas do sistema.

3.2. Modelagem de inicialização do processo

O autômato da Figura 3 representa o modelo de inicialização adotado para tratamento de todos os eventos envolvidos no processo. No estado q_0 todos os eventos precisam ser ignorados enquanto o início do confronto (evento *Bl*) não seja autorizado. Após o disparo de *Bl* os outros eventos do processo poderão ser tratados normalmente.

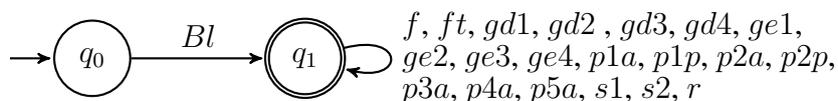


Figura 3. Modelagem de inicialização do processo

3.3. Modelagem dos Sensores de Presença

Cada sensor de presença possui dois eventos próprios, que são suas bordas de subida ou descida, indicando a presença ou ausência, respectivamente, do oponente. Esses eventos são nomeados p_{xa} para caso o sensor capte o oponente, ou p_{xp} caso o perca, na qual x indica o respectivo sensor.

Para os sensores $p1$ e $p2$, a modelagem é representada pela Figura 4. No estado q_0 , todos os eventos dos sensores envolvidos no modelo são ignorados até que a botoeira *Bl* seja pressionada. Após disparado o evento *Bl*, a transição leva ao estado q_1 . Nesse estado, os eventos de borda de subida dos sensores envolvidos em cada subsistema podem ocorrer,

e caso algum evento $p1a$ ou $p2a$ ocorrer, no caso achou o seu adversário, o autômato migra para o estado $q2$. Enquanto estiver nesse estado, os eventos $p3a$, $p4a$ e $p5a$, estão fisicamente impossibilitados de ocorrer visto que um dos sensores frontais já o encontrou. No estado $q2$, caso dispare uma borda de descida do sensor em questão ($p1p$ ou $p2p$) indicando a perda do oponente, o autômato volta ao estado $q1$, possibilitando novamente a ocorrência dos sensores envolvidos em cada subsistema.

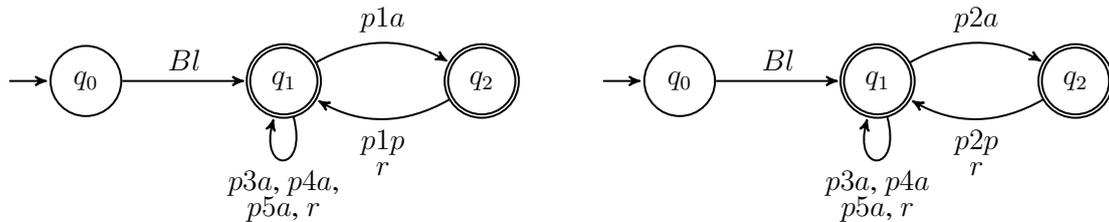


Figura 4. Modelagem dos sensor p1 e p2

O evento r é habilitado em todos os estados em que o robô já está se movimentando, pois esse evento deve ser priorizado, de forma que se algum dos sensores de faixa forem acionados, esse evento não pode estar impossibilitado de ocorrer por nenhuma outra planta, só assim o robô poderá tratar o sensor de faixa e evitar que saia dos limites do Dojô. Após disparado o evento r , o autômato migrará para o estado $q1$.

A modelagem dos autômatos dos sensores $p3$, $p4$ e $p5$ é mostrada na Figura 5, e seguem uma lógica semelhante à dos sensores $p1$ e $p2$. Enquanto no estado $q1$, a transição para $q2$ será feita após o evento $p3a$, $p4a$ ou $p5a$ ocorrer respectivamente, indicando que um dos sensores laterais ou traseiro achou o oponente. No estado $q2$, os demais sensores ficam impossibilitados de encontrar o oponente devido à disposição geométrica dos sensores. Para os sensores $p3$, $p4$ e $p5$ não foi tratado a borda de descida do sinal, diferentemente de $p1$ e $p2$, nesse caso a transição do estado $q2$ para o estado $q1$ é feita a partir de um evento controlável, que será disparado automaticamente pelo microcontrolador. Para o modelo do sensor $p3$, esse evento controlável é o $ge2$, significando um giro de 90° à esquerda de forma que o robô se posicione de frente ao adversário. No caso do sensor $p4$, será realizado um giro de 90° para a direita com $gd2$ e para o sensor $p5$, será escolhido um evento aleatório entre $gd4$ ou $ge4$ para que o robô faça um giro de 180° . O evento r continua habilitado em todos os estados em que o robô se movimenta, devido à prioridade do sensor de faixa.

3.4. Modelagem dos sensores de faixa

A planta dos sensores $s1$ e $s2$ são necessárias para que o robô lide com os limites do Dojô. Dessa forma pode ser especificada uma sequência de ações que façam com que ele volte para dentro da arena ao captar uma faixa.

Os autômatos dos sensores $s1$ e $s2$, representados na Figura 6, assim como os autômatos vistos até então, possuem um estado $q0$ em que todos os eventos dos sensores envolvidos são ignorados, com exceção de Bl . Ao fazer a transição para $q1$, há algumas outras transições em forma de auto laço que indicam a possibilidade de ocorrência desses eventos nesse estado. No estado $q1$, caso o evento relacionado ao sensor da planta dispare, o autômato migra para o estado $q2$. Em $q2$, há um auto laço com o evento do outro sensor

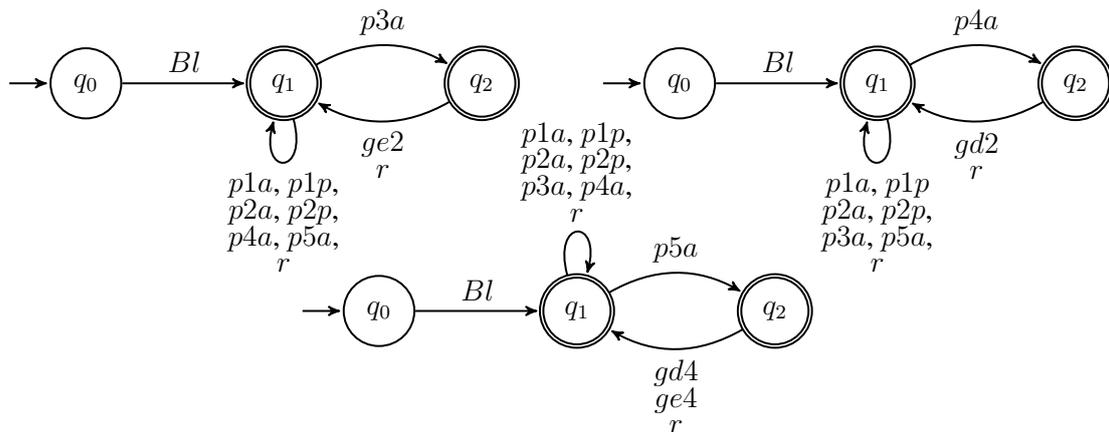


Figura 5. Modelagem dos sensores p3, p4 e p5

de faixa, indicando que esse evento não pode ser ignorado, seguido por uma transição que leva ao estado q_3 gerado pelo evento controlável r , isso significa que sempre que o microcontrolador receba uma borda de subida de algum dos sensores de faixa, o robô deverá ignorar os sensores de presença e andar de ré para retornar ao centro do Dojô. No estado q_3 , qualquer sensor de presença que receber uma borda de subida indicando a presença do oponente fará o autômato migrar novamente para o estado q_1 , e se caso nenhum sensor receber esse sinal, será gerado um evento controlável $gd3$ ou $ge3$, que faz com que o robô realize um giro contrário ao sensor disparado de 120° buscando se orientar ao centro do Dojô.

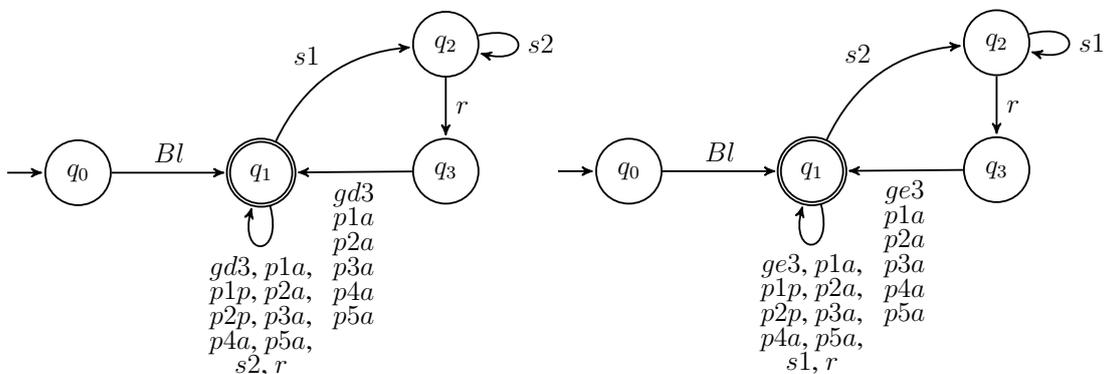


Figura 6. Modelagem dos sensores s1 e s2

3.5. Modelagem das especificações

A especificação de ataque demonstrada na Figura 7 foi modelada com o objetivo final de, após o oponente ser detectado por ambos os sensores frontais, o microcontrolador gere o evento controlável ft , na qual o robô anda para a frente com o dobro da velocidade que é utilizada em outros casos. Todos os estados possuem o evento r habilitado, uma vez que se o robô precisar se afastar da borda do Dojô após um movimento de ré, o robô volta para a situação inicial de ataque. Quando no estado q_0 , se algum dos sensores $p1$ ou $p2$ disparar uma borda de subida, migrando para os estados q_1 ou q_4 respectivamente, o robô realiza um giro através dos eventos $ge1$ ou $gd1$ de forma a se reposicionar para que o segundo

sensor encontre o oponente e assim ficar de frente ao adversário. No estado q_7 , o robô se encontra frente a frente com seu adversário, nesse caso é gerado o evento controlável ft para que os motores acionem para frente com máxima velocidade de forma a jogar o adversário para fora do Dojô.

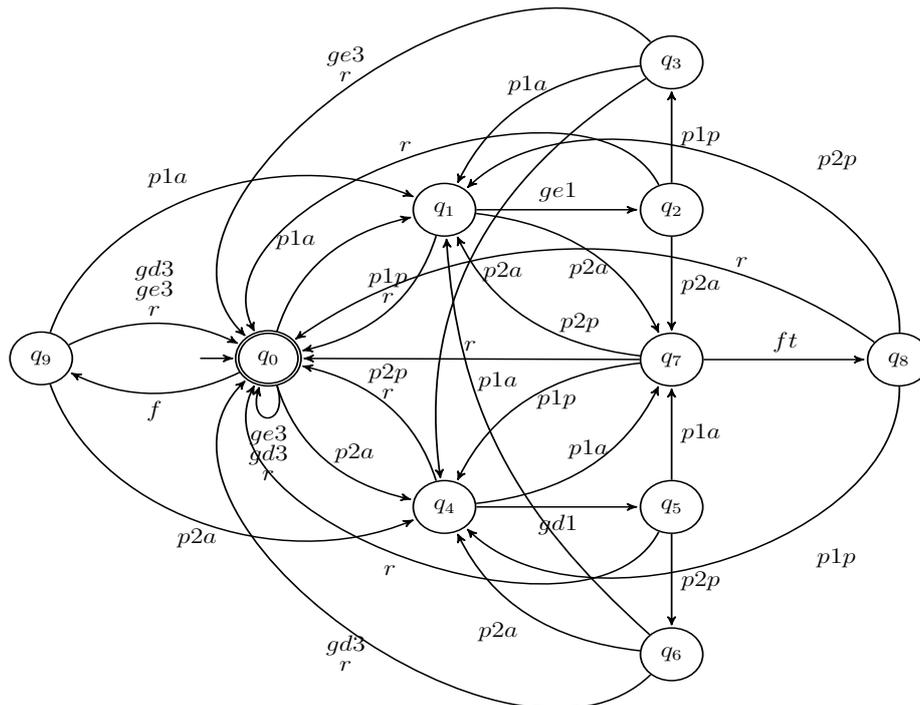


Figura 7. Modelagem da especificação de ataque frontal

A Figura 8 apresenta duas especificações do sistema. A especificação da esquerda representa a modelagem de afastamento da borda do Dojô. Quando os sinais emitidos pelos sensores de faixa s_1 e s_2 são disparados, todos os sinais de borda de subida dos sensores de presença são desabilitados até que o robô consiga dar ré para poder voltar ao centro da arena. Quando no estado q_0 , o autômato pode fazer a transição para q_1 através de s_1 ou para q_3 através de s_2 . Quando em um desses estados, habilitam-se apenas o evento controlável r e o outro sensor de faixa, fazendo que o robô desista do que estava fazendo anteriormente para poder dar ré de forma que não ultrapasse os limites da arena. Após o evento r ser gerado, habilita-se novamente os eventos destinados ao ataque.

Já a especificação da direita da Figura 8 mostra um autômato que faz o tratamento dos sinais emitidos pelos sensores de presença não-frontais. Caso algum desses sensores capte a presença do oponente, o robô fará um giro de acordo com o sensor que emitiu o sinal. Todos os estados possuem uma transição de retorno ao estado q_0 a partir do evento controlável r , isso significa que após um afastamento da borda do Dojô dando ré, o robô volte para a condição inicial de ataque.

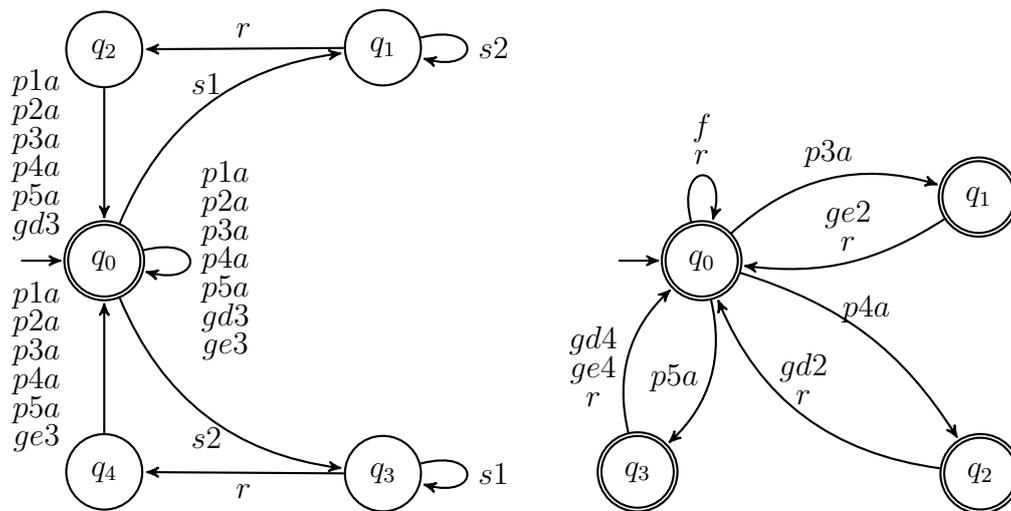


Figura 8. Modelagem das especificações de afastamento da borda do Dojô (esquerda) e tratamento dos sensores de presença não-frontais (direita)

Por último, o autômato da Figura 9 representa uma especificação de coordenação do sistema. Sua principal função é a de fazer o chaveamento entre as configurações representadas pelos autômatos das Figuras 7 e 8. Cada estado desse autômato representa uma das especificações anteriores. O alfabeto de eventos equivalente à especificação representada pelos estados estão em auto laço, e as transições entre um estado para o outro representam as alternâncias entre especificações.

3.6. Síntese do Supervisor

Para síntese do supervisor foi utilizada a TCS [Ramadge and Wonham 1989], percorrendo o seguinte procedimento:

1. Obtenção da planta global do sistema a partir do produto síncrono entre todos os subsistemas envolvidos, sendo esses subsistemas os modelos representados nas Figuras 3, 4 e 5. Esta composição resultou em um autômato com 32 estados e 226 transições.
2. Obtenção da especificação global desejada a partir do produto síncrono de todas as especificações parciais, sendo as especificações parciais os modelos representados nas Figuras 7, 8, e 9. O modelo global das especificações resultou num autômato com 159 estados e 574 transições.
3. O produto síncrono entre a planta global e especificação global representa o comportamento desejado do sistema, o qual não necessariamente será controlável. Este modelo resultou num autômato com 123 estados e 308 transições.
4. Finalmente, a partir da planta global e o comportamento desejado, usando um algoritmo descrito em [Ramadge and Wonham 1989] se obtém o comportamento controlável e minimamente restritivo que atende a todas as especificações simultaneamente que resultou em um autômato com 123 estados e 308 transições na qual foi minimizado, de tal forma a manter a linguagem reconhecida. O resultado foi um autômato com 29 estados e 114 transições e foi utilizado para implementação do supervisor S .

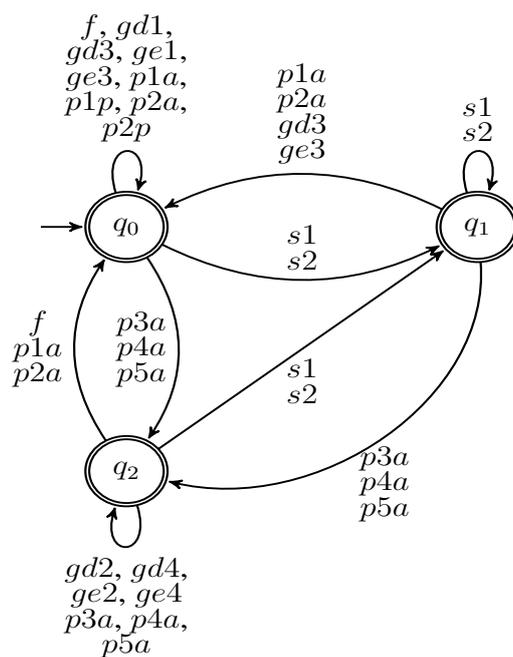


Figura 9. Modelagem da especificação de coordenação entre as demais especificações

3.7. Validação do Modelo

Com o supervisor obtido foi realizada uma simulação exaustiva de tal forma a verificar o atendimento das especificações propostas. Na sequência esse Supervisor foi embarcado em um microcontrolador (stm32F407VETx), utilizando como auxílio uma ferramenta para geração automática de código, e instalado no robô da Figura 10.

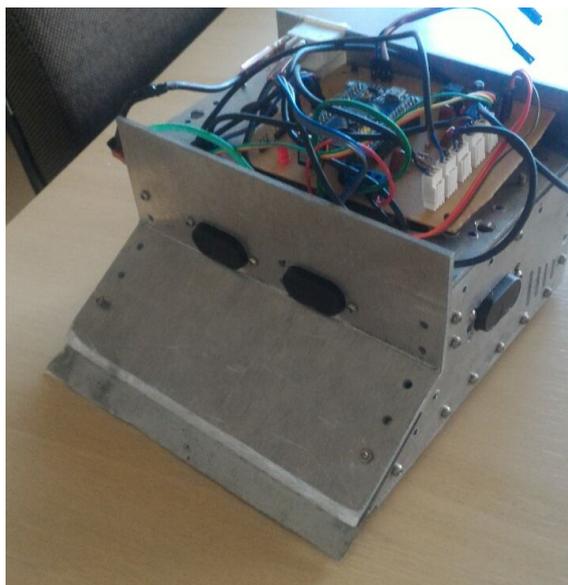


Figura 10. Robô de sumô 3 kg utilizado para a validação dos resultados

Da mesma forma que na simulação, o robô se comportou conforme o esperado, se

reposicionando de frente para o adversário de acordo com o sensor de presença disparado, e em seguida atacando com velocidade turbo. Nos casos em que não havia adversário na arena, o robô se locomovia para frente até achar uma borda da arena, acionando assim a ré nos motores por tempo determinado no código do microcontrolador e em seguida se reposicionando para o centro da arena.

4. Conclusão

Este artigo demonstrou o uso da Teoria de Controle Supervisório em um robô de sumô autônomo. O procedimento da TCS fez com que o espaço de estados do sistema fosse reduzido de acordo com as restrições impostas pelas especificações modeladas, garantindo que o robô obtivesse um comportamento minimamente restritivo e não bloqueante, e esse comportamento sendo representado pelo Supervisor Controlável S .

A metodologia adotada para a modelagem dos autômatos permite que cada sensor seja modelado separadamente de forma fácil e eficaz. Uma pequena mudança, como por exemplo, a adição de mais sensores ou atuadores, implicaria apenas no acréscimo de mais um subsistema para a síntese do comportamento global, sem alterar os demais subsistemas.

O procedimento sistemático da TCS, permitiu o desenvolvimento de ferramentas computacionais tal como o “Supremica”, o qual foi utilizado para manipulação do autômatos deste trabalho. Com a ajuda de outras ferramentas para geração automática de código em um microcontrolador a partir do Supervisor Controlável S , esse método se torna uma forma rápida e eficaz de realizar testes com as lógicas de controle modeladas, podendo ou não ser relativas à robótica. Dessa forma, o uso da TCS em um robô de sumô se torna apenas um exemplo de como essa estratégia de controle pode ser expandida em diversas outras aplicações na área de automação.

Referências

- Cassandras, C. G. and Lafortune, S. (2009). *Introduction to discrete event systems*. Springer Science & Business Media.
- Malik, R., Åkesson, K., Flordal, H., and Fabian, M. (2017). Supremica, an efficient tool for large-scale discrete event systems. *IFAC-PapersOnLine*, 50(1):5794–5799.
- Ramadge, P. and Wonham, W. (1989). The control of discrete event systems. *Proceedings of the IEEE*, 77(1):81–98.
- Ramadge, P. J. and Wonham, W. M. (1982). Supervision of discrete event processes. In *1982 21st IEEE Conference on Decision and Control*, pages 1228–1229. IEEE.
- Secchi, H. A. (2008). Uma introdução aos robôs móveis. page 81.
- Torrico, C. R., Leal, A. B., and Watanabe, A. T. (2016). Modeling and supervisory control of mobile robots: A case of a sumo robot. *IFAC-PapersOnLine*, 49(32):240 – 245.