



INSTITUTO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DA PARAÍBA
DIRETORIA DE DESENVOLVIMENTO DE ENSINO
COORDENAÇÃO DO CURSO SUPERIOR DE BACHARELADO EM
ENGENHARIA DE COMPUTAÇÃO

ARTHUR V. F. FIALHO

MOZART LIMA DO NASCIMENTO

MODELAGEM DE UM SISTEMA EMBARCADO
COMO UM AUTÔMATO FINITO DETERMINÍSTICO
QUE AUXILIE NA VALIDAÇÃO DO BOI EM VAQUEJADAS

CAMPINA GRANDE - PB

2024

ARTHUR V. F. FIALHO
MOZART LIMA DO NASCIMENTO

MODELAGEM DE UM SISTEMA EMBARCADO
COMO UM AUTÔMATO FINITO DETERMINÍSTICO
QUE AUXILIE NA VALIDAÇÃO DO BOI EM VAQUEJADAS

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Engenharia de Computação, do Instituto Federal da Paraíba – Campus Campina Grande, em cumprimento às exigências parciais para a obtenção do título bacharel em Engenharia de Computação.

ORIENTADOR (A):
JOSÉ ANTONIO C. B. DA SILVA

CAMPINA GRANDE - PB

2024

F438m

Fialho, Arthur V. F.

Modelagem de um sistema embarcado como um autômato finito determinístico que auxilie na validação do boi em vaquejadas / Arthur V. F. Fialho, Mozart Lima do Nascimento - Campina Grande, 2024.

28 f. : il.

Trabalho de Conclusão de Curso (Curso Superior de Bacharelado em Engenharia de Computação.) - Instituto Federal da Paraíba, 2024.

Orientador: José Antonio C. B. da Silva

1. Desenvolvimento de sistema - sistema embarcado 2. Vaquejada 3. Autômato Finito Determinístico - AFD I. Nascimento, Mozart Lima do II. Silva, José Antônio C. B. da III. Título.

CDU 004.4

ARTHUR V. F. FIALHO
MOZART LIMA DO NASCIMENTO

MODELAGEM DE UM SISTEMA EMBARCADO
COMO UM AUTÔMATO FINITO DETERMINÍSTICO
QUE AUXILIE NA VALIDAÇÃO DO BOI EM VAQUEJADAS

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Engenharia de Computação, do Instituto Federal da Paraíba – Campus Campina Grande, em cumprimento às exigências parciais para a obtenção do título bacharel em Engenharia de Computação.

Aprovada em ____ / ____ / _____

Banca Examinadora

Prof(a). José Antonio C. B. da Silva , DSc. - IFPB

Orientador (IFPB)

Prof(a). Fagner de Araújo Pereira , DSc. - IFPB

Examinador

Prof(a). Moacy Pereira da Silva , DSc. - IFPB

Examinador

Abstract. In a "vaquejada," the bull can be validated or invalidated. A validated bull is one that has been brought down by the "puxador" (rider) between the two scoring lines. In this study, an embedded system is modeled as a deterministic finite automaton (DFA), which serves as an auxiliary tool in the decision-making process of the vaquejada judge regarding the validation of the bull's fall, whether it occurred within or outside the scoring lines (the objective of the sport). This is particularly useful since the vaquejada judge may face challenges in judging the bull in certain situations. The modeled DFA includes states such as "Wait," "Validated," and "Invalidated," and it employs the alphabet "Validation," "Invalidation," and "Reset".

Keywords: theory of computing, deterministic finite automaton, model, vaquejada

Resumo. Em uma vaquejada o boi pode ser validado ou invalidado, um boi validado é aquele que foi derrubado pelo puxador entre as duas faixas de pontuação seguindo todas as regras da vaquejada. No presente trabalho é modelado um sistema embarcado como um autômato finito determinístico (AFD), que tem como função auxiliar a decisão do juiz da vaquejada na validação da queda do boi, dentro ou fora da faixa (objetivo do esporte). Uma vez que, o juiz de vaquejada pode ter dificuldade para julgar o boi em algumas situações. O AFD modelado conta com estados como "Espera", "Validado" e "Invalidado", e utiliza o alfabeto "Validação", "Invalidação" e "Reset".

Palavras chaves: teoria da computação, autômato finito determinístico, modelo, vaquejada

*À Deus.
À nossos pais, Maurílio Furtado Fialho, Germinia Venâncio Fialho,
Moisés Lima do Nascimento e Maria Alves de Lima
por todo apoio e carinho!*

Dedicamos!

AGRADECIMENTOS

Ao nosso orientador José Antonio C. B. da Silva e membros da Banca, representados pelos Professores Fagner de Araújo Pereira e Moacyr Pereira da Silva, que estiveram sempre disponíveis a atender às nossas solicitações, nos apontando os caminhos que deveriam ser traçados para desenvolver este trabalho.

À todos os professores que nos guiaram durante o processo de aprendizagem ao longo da graduação.

Aos demais funcionários do IFPB, por toda atenção e profissionalismo empregados para que a instituição seja para sempre guardada com carinho e respeito em nossas vidas.

E por fim, às nossas companheiras Cynthia Ribeiro Guimarães e Deise de Figueiredo Araújo, obrigado por todo apoio.

Saga de um vaqueiro

*Vou pedir licença pra contar a minha história
Como um vaqueiro tem suas perdas e suas glórias
Mesmo sendo forte, o coração é um menino
Que ama e chora e por dentro e segue seu destino
Desde cedo assumi minha paixão, de ser vaqueiro
E ser um campeão, nas vaquejadas sempre fui batalhador
Consegui respeito por ser um vencedor..
Quando me preparava pra entrar na pista
Quando olhei de lado quase escureci a vista
Quando vi uma mulher, aquela que foi a minha vida
Segurei no meu cavalo para não cair
Tremi, fiquei nervoso quando eu a vi.
Enxugando e abraçando o vaqueiro bem ali..
Entrei na pista como um louco
O bate-esteira percebeu andei foi longe do boi
Isso nunca aconteceu.
O vaqueiro entrou na pista e eu fiquei a observar
Ela acenava, ela aplaudia e ele boi a derrubar
Derrubou o boi na faixa Ganhou o primeiro lugar.
Fiquei desconsolado envergonhado eu fiquei
Perdi o grande prêmio, isso até eu nem liguei
Mas perder aquele amor, ah eu não me conformei! ...
Ela veio sorridente em minha direção
E trouxe o vaqueiro pegado em sua mão
Olhou-me nos meus olhos
Falou com atenção:
Esse é o nosso filho que você não conheceu
Sempre quis ser um vaqueiro como você um campeão
E pela primeira vez, quer a sua benção..
Eu chorava de feliz, abraçado com meu filho
Um vaqueiro como eu, Eu nunca tinha visto
Posso confessar o maior prêmio que Deus me deu ! ...*

Rita de Cássia

LISTA DE FIGURAS

Figura 1. A derrubada do boi em vaquejadas.

Figura 2. Dúvida do toque da parte superior do boi na primeira faixa (exemplo 1).

Figura 3. Dúvida do toque da parte superior do boi na primeira faixa (exemplo 2).

Figura 4. Dúvida do toque da parte superior do boi na primeira faixa (exemplo 3).

Figura 5. Diagrama de estados do autômato M

Figura 6. Algoritmo 1 - parte 1.

Figura 7. Algoritmo 1 - parte 2.

Figura 8. Algoritmo 1 - parte 3.

Figura 9. Algoritmo 1 - parte 4.

Figura 10. Métrica para considerar a derrubada do boi.

LISTA DE ABREVIATURA E SIGLAS

ABVAQ	Associação Brasileira de Vaquejada
AF	Autômato finito
AFD	Autômato Finito Determinístico
AFND	Autômatos Finitos Não-Determinísticos
ANC	Autômato Não Completo
ER	Expressões Regulares
VAR	Vídeo Assistant Referee

SUMÁRIO

1. Introdução	11
2. Fundamentação Teórica	12
2.1 Vaquejada	12
2.2 Autômatos Finitos Determinísticos	15
2.2.1 Definição Formal	15
2.2.2 Determinismo	15
2.2.3 Computação de strings	16
2.2.4 Completude	16
2.3 Modelagem	17
3. Metodologia	17
4. Resultados	19
4.1 Interpretação da definição do autômato M	20
4.1.1 Conjunto de Estados	20
4.1.1.1 Estado “Espera” (e)	20
4.1.1.2 Estado “Validado” (v)	20
4.1.1.3 Estado “Invalidado” (i)	20
4.1.2 Alfabeto	20
4.1.2.1 Símbolo “Validação” (1).....	20
4.1.2.2 Símbolo “Invalidação” (0)	20
4.1.2.3 Símbolo “Reset” ()... ..	21
4.1.3 Estado Inicial	21
4.1.3.1 Estado “Espera” (e)	21
4.1.4 Conjunto de estados de aceitação	21
4.1.4.1 Estado “Espera” (e)	21
4.1.5 Conjunto de Transições	21
4.1.5.1 δ_1 : “Espera” X “Validação” \rightarrow “Validado”	21
4.1.5.2 δ_2 : “Espera” X “Invalidação” \rightarrow “Invalidado”	21
4.1.5.3 δ_3 : “Validado” X “Reset” \rightarrow “Espera”	21
4.1.5.4 δ_4 : “Invalidado” X “Reset” \rightarrow “Espera”	21
4.2 Computação de algumas strings arbitrárias como entradas do autômato M	22
4.2.1 1ω	22
4.2.2 0ω	22
4.2.3 $0\omega 1\omega$	22
4.2.4 $1\omega 0\omega$	22
4.2.5 $0\omega\omega 1$	22
4.3 Algoritmo de validação do boi	23
4.3.1 Tradução do algoritmo	25
5. Conclusão.....	26
6. Referências bibliográficas	27
Anexo	28

1. Introdução

O impacto da tecnologia no esporte é objeto de estudo, sob a perspectiva dos atletas, treinadores, espectadores, fãs, pais, oficiais, mídia, indústria e indivíduos [Katz, 2002]. De acordo com Katz(2002), o uso de ferramentas tecnológicas na arbitragem é um fator primordial para oferecer aos juízes a oportunidade de arbitrar com a ajuda de evidências relevantes.

Considerando a visão, de acordo com a posição física, as ramificações de localização e a habilidade de processar dados, de forma precisa e eficaz, a tecnologia abrange o desenvolvimento de simulações que possibilitem aos juízes tornarem suas decisões mais assertivas. Para tal fim, é primordial o uso de ferramentas como animação gráfica, modelos tridimensionais e o uso de sequências de vídeo reais da competição, usando ângulos múltiplos de câmara, além de todo um arcabouço de possíveis tecnologias utilizadas a favor do esporte. Conforme relatado na literatura [Gantois, 2015][Neto, 2023], o uso de tecnologias trouxe impactos positivos em relação a várias métricas em esportes como base, esgrima, taekwondo e futebol, por exemplo.

A utilização da tecnologia no esporte é realidade desde a adoção do photo finish nas corridas de atletismo, automobilismo e hipismo. Atualmente, faz parte do cotidiano dos envolvidos no esporte (atletas, espectadores, juízes, técnicos, dentre outros), o auxílio tecnológico, seja com o Vídeo Assistant Referee (VAR), os desafios no vôlei ou mesmo a vaquejada que atualmente utiliza filmagens para auxiliar na arbitragem, o boi TV. Sabendo que a vaquejada é um esporte dinâmico, cuja fração de segundo no lance não pode ser analisada de forma precisa, o juiz muitas vezes tem dificuldade em julgar a validade do boi. Assim, surge a seguinte problemática: O juiz de vaquejada tem dificuldade em definir a validade da derrubada do boi em alguns casos, nos quais o boi cai em cima de uma das faixas de pontuação.

No Manual de Julgamento de boi, da Associação Brasileira de Vaquejada (ABVAQ) (2022) são definidas diversas regras, que definem como o juiz de vaquejada deve julgar o boi. As páginas 5 e 6 do manual em questão, abordam como deve ser julgada a validação do boi, sendo fundamentais para entender como ocorre a validação do boi em vaquejadas. De maneira simplificada, o boi deve cair entre as duas faixas de pontuação, para que seja validado, seguindo todas as regras da vaquejada.

Assim como em outras áreas da Engenharia, há as fases de projeto e execução, seja em edificações, estações de tratamento de água, conservação ou secagem de alimentos. Este trabalho foi centrado na fase de construção do modelo de autômato, que terá o papel de modelo base, para a futura implementação de um protótipo que auxilie na validação do boi em vaquejadas. Deixa-se claro, que não haverá implementação de um protótipo. A modelagem do presente trabalho se destina a responder o que o sistema deve fazer, modelando o comportamento do sistema, não entrando em detalhes sobre como deve ser feita a implementação de um protótipo.

Com o intuito de solucionar a problemática mencionada, o objetivo geral deste trabalho é modelar um sistema embarcado por meio de um Autômato Finito Determinístico (AFD) que pode vir a apoiar juízes em suas decisões durante as vaquejadas.

2. Fundamentação Teórica

2.1. Vaquejada

A Vaquejada é um patrimônio cultural imaterial do Brasil, de acordo com a ementa da Lei 13.364/2016, cumprindo os requisitos para ser enquadrada como esporte e podendo, portanto, gozar de toda a homologação esportiva, sofrendo os ditames elencados no artigo 217 da Constituição da República Federativa do Brasil, na lei geral do Desporto (Lei nº 10.671).

Hoje, a vaquejada reproduz uma técnica centenária, usada por vaqueiros, que tentavam arrebanhar touros e vacas, nos quais fugiam de criadouros que não possuíam cercas adequadas para cercá-los em grandes áreas do rebanho das fazendas [Veiga, 2020] [Rodrigues; Cavalcanti, 2017]. Atualmente este esporte está conquistando regiões diversificadas do país, passando das fronteiras do Nordeste do Brasil, chegando ao Norte e ao Sudeste [Veiga, 2020].

A Associação Brasileira de Vaquejada (ABVAQ) normatiza a prática profissional do esporte no país, elaborando manuais e chancelando corridas de vaquejada por todo o país. Dessa forma, a vaquejada obedece regras para sua prática de forma profissional e/ou amadora [ABVAQ, 2022, 2023]. A compreensão das regras é de suma importância, para que o autômato possa vir a modelar o sistema embarcado, que auxilie na validação do boi em vaquejadas. As regras sobre validação do julgamento, da derrubada do boi na faixa estão apresentadas no Anexo A, podendo ser encontradas também, no Manual de Julgamento de

Boi [ABVAQ, 2022].

Em uma vaquejada, o boi pode ser validado ou invalidado, um boi validado é aquele que foi derrubado pelo puxador entre as duas faixas de pontuação seguindo todas as regras da vaquejada [ABVAQ, 2022].

O Regulamento Geral da Vaquejada [ABVAQ, 2023] define na segunda página (Seção 1, art.3, inciso 6) que “6. Valeu o boi – Expressão que caracteriza o êxito do competidor;”. Assim, a expressão “valeu o boi” é uma expressão comum no meio e explicada no regulamento, ressaltando que o boi pode valer ou não, ser validado ou invalidado.

A competição, de forma simplificada e de acordo com a regulamentação da ABVAQ, é formada pela disputa entre duplas de vaqueiros (vaqueiro-puxador e vaqueiro-esteireiro) a fim de obter a maior pontuação, através da derrubada do boi. Cada dupla tem o direito de correr, em média, três bois por senha. Para conquistar a derrubada do boi (conhecida como “valeu boi”), o vaqueiro puxador deve derrubar o boi, em uma faixa 9 m de extensão, demarcada por duas linhas feitas com cal, sob o colchão de areia (pista de corrida), conforme representado na Figura 1.



Figura 1. A derrubada do boi em vaquejadas.

De acordo com o manual de julgamento de boi (Anexo A), as partes superiores do boi (linha imaginária onde se localiza o jarrete - joelho ou parte seca; para cima) não podem tocar a primeira faixa de pontuação. Assim, a problemática encontrada na derrubada do boi, ocorre quando parte do corpo do animal fica para fora das faixas demarcadas. Para isso, a ABVAQ faz uso de uma comissão alternativa (Boi TV), porém essa alternativa pode apresentar erros de julgamento, pois o ângulo da filmagem pode comprometer o julgamento.

Nas Figuras 1, 2, 3 e 4 têm-se casos duvidosos sobre o toque ou não das partes superiores do boi na primeira faixa de pontuação.



Figura 2. Dúvida do toque da parte superior do boi na primeira faixa (exemplo 1).



Figura 3. Dúvida do toque da parte superior do boi na primeira faixa (exemplo 2).



Figura 4. Dúvida do toque da parte superior do boi na primeira faixa (exemplo 3).

2.2. Autômatos Finitos Determinísticos

Máquinas de Estados e autômatos podem ser representadas por programas, que procuram por padrões e, frequentemente, possuem uma estrutura especial. É possível identificar posições no código, em que se conhece algo particular sobre o progresso do programa, a fim de reconhecer uma instância de um padrão. Estas posições são chamadas de estados. O comportamento do programa, pode ser visto como a transição de um estado para o outro, conforme a entrada é lida [Araújo, 2015].

A teoria dos autômatos, fornece definições e propriedades para se trabalhar com modelos de computação [Sipser, 2007]. Existem diversos tipos de autômatos, contudo serão utilizados apenas os autômatos finitos ou máquinas de estados finitos. Estes são usados para processamento de texto, compiladores, e projetos de hardware [Sipser, 2007]. O projeto de hardware desenvolvido neste trabalho é a modelagem de um sistema embarcado como um autômato

2.2.1. Definição Formal

Um autômato finito (AF) é um diagrama de transição de estados, que permite modelar, de maneira intuitiva, diversos problemas computacionais [Lewis; Papadimitriou, 1998], contendo um conjunto de estados nos quais a transição de um estado para um outro é realizada na ocorrência de um evento [Hopcroft; Ullman, 2001].

De acordo com Sipser (2007), um autômato finito pode ser definido como uma 5-upla $(Q, \Sigma, \delta, q_0, F)$ onde

Q é um conjunto finito chamado de os estados,

Σ é um conjunto finito chamado de o alfabeto,

δ é a função de transição $\delta : Q \times \Sigma \rightarrow Q$,

$q_0 \in Q$ é o estado inicial, e

$F \subseteq Q$ é o conjunto de estados finais.

2.2.2. Determinismo

Existem dois tipos de Autômatos Finitos: Autômatos Finitos Determinísticos (AFD) e Autômatos Finitos Não-Determinísticos (AFND). Os Determinísticos possuem exatamente, para cada estado e para cada símbolo do alfabeto da entrada, uma transição com este símbolo,

deixando este estado. Os Não-Determinísticos, não possuem restrições nos valores de suas transições. Um símbolo pode ser o valor de várias transições, que saem do mesmo estado, e ϵ , a palavra vazia, é um valor possível.

Autômatos Finitos são bons modelos para descrever dispositivos eletromecânicos [Sipser, 2017]. Os AFDs conseguem, com naturalidade, representar de forma concisa, qualquer sistema que mantenha uma definição interna de estado – capacidade do sistema de ser representado, utilizando a metodologia e a terminologia de autômatos finitos determinísticos [Gribkof, 2013]. Quanto aos AFNDs, do ponto de vista prático, não são modelos realistas, indicados para representar algo concreto, do mundo real [Silva, 2023].

O autômato finito usado neste trabalho é determinístico, pois sempre que está num dado estado e lê o próximo símbolo de entrada, sabe-se qual estado será o próximo [Sipser, 2007]. A computação com AFDs é completamente determinística, ou seja, para cada par (q, a) , sendo q um estado e a um símbolo, temos exatamente uma transição definida no ponto (q, a) [Silva, 2023]. Em contraposição, num autômato finito, não determinístico (AFND), várias escolhas podem existir, para cada próximo estado, em qualquer ponto [Sipser, 2007].

2.2.3. Computação de strings

Autômatos Finitos são reconhedores (aceitam ou rejeitam) de strings. Eles podem dizer “sim” (aceita) ou “não” (rejeita) sobre cada possível símbolo da entrada. Se todos os símbolos de uma string forem aceitos, a string é aceita; se um dos símbolos de uma string for rejeitado, a string é rejeitada. Pode-se definir formalmente a aceitação e rejeição de strings da seguinte forma: “Seja $D = (Q, \Sigma, \delta, q_0, F)$ um AFD e $w \in \Sigma^*$. Se $\delta(q_0, w) \in F$, então dizemos que D aceita a string w . Se $\delta(q_0, w) \notin F$, então dizemos que D rejeita a string w ” [Silva, 2023].

Dado um alfabeto, uma linguagem é um conjunto qualquer de strings, de símbolos sobre este alfabeto. Expressões regulares (ERs) são expressões matemáticas, que representam linguagens. Essas ERs têm a capacidade de expressar com exatidão, os problemas que podem ser resolvidos pelos algoritmos, expressos na forma de um AFD [Silva, 2023].

2.2.4. Completude

Em um autômato completo para cada estado, tem-se uma indicação para mudar de estado, para todo o "input" recebido. Há casos onde isto não acontece, e aí estamos perante um Autômato Não Completo (ANC). [IST LEIC-A, 2024]

Em um ANC, não está especificado o que acontece, quando o autômato está num determinado estado e recebe um símbolo do alfabeto, para o qual não existe uma transição. Não é sem querer, que isso acontece nesses casos, de fato o que acontece nessas situações é que o autômato rejeita (“não aceita”) essa essa transição.

2.3. Modelagem

Um sistema embarcado é um sistema computacional aplicado [Noergaard, 2005]. Visto que pode-se pensar em computadores como “instâncias” de objetos abstratos e seus possíveis relacionamentos matemáticos em objetos físicos e seu conjunto de possíveis graus de movimento [Silva, 2023], podemos considerar o AFD modelado neste trabalho como um sistema computacional. Além disso, o projeto tem uma aplicação prática, bem definida, que é a validação da derrubada do boi em vaquejadas.

Define-se “modelo” como “representação, em escala reduzida, de objeto, obra de arquitetura etc. a ser reproduzida em dimensões normais” [Oxford, 2023]. Em acordo com a definição, a modelagem de um AFD é uma representação reduzida de um sistema embarcado a ser implementado.

Um modelo, antes de mais nada, é uma representação de um recorte da realidade. Funcionando como um instrumento de abstração, destinado à aquisição de novos conhecimentos, representação e compreensão da realidade [Sayão,2001].

Qualquer projeto, pode se beneficiar com o uso de algum tipo de modelagem, pois modelos auxiliam, para ter uma visão mais abrangente, do funcionamento de um sistema. Tornando possível desenvolver os projetos, de forma mais rápida e correta, pois, quanto mais complexo for o sistema, maior será a probabilidade de ocorrência de erros, caso tenha sido elaborado sem uma modelagem prévia [Chwif et al., 2010].

Somado a todas essas vantagens, é válido dizer que um AFD é adequado para representar de forma concisa, natural e ao mesmo tempo formal, qualquer sistema determinístico [Gribkoff, 2013].

3. Metodologia

Foi realizado um levantamento das regras da vaquejada, coletando informações das regras a que o vaqueiro, o boi e a pista estariam sujeitos. Para isso, foram realizadas consultas ao Regulamento Geral da Vaquejada [ABVAQ,2023] e ao Manual de Julgamento de Boi

[ABVAQ, 2022]. Assim, todo o material elaborado, está de acordo com o regulamento da ABVAQ mais atual até o momento da construção deste projeto. As principais regras, referentes a como julgar a validação do boi, podem ser consultadas no Anexo A.

Ao pesquisar sobre a construção de um AFD, foi estudado, como defini-lo formalmente. Tal conhecimento serviu como base para a construção do modelo esboçado neste trabalho. Desta forma, o modelo segue todas as regras às quais um AFD está sujeito segundo Sipser(2007).

Foi adotada a abordagem determinística, para modelar o sistema, com o intuito de evidenciar qual seria o comportamento do sistema, quando ele estivesse num determinado estado e fizesse a leitura de um símbolo.

O projeto foi modelado como um AFD e não utilizando outras metodologias determinísticas, como redes de Petri, pois os desenvolvedores deste trabalho possuíam familiaridade com a modelagem de AFDs, o que reduziu a curva de aprendizagem necessária para modelar o projeto.

Não existe resultado final duvidoso no julgamento da validação do boi em uma vaquejada, existem apenas dois possíveis resultados, validado (êxito do competidor) ou invalidado (ausência de êxito do competidor). O resultado é discretizado pelos juízes, mesmo nas situações em que o julgamento do boi é difícil (situações de dúvida), como apresentado na problemática.

Como uma implementação futura, da modelagem proposta, vai julgar se o boi está 50% dentro da faixa ou não (por exemplo), é um problema futuro. Seja qual for a abordagem adotada para solucionar esse problema, ela precisará, em análise final, ser discretizada, se quiser validar ou invalidar o boi. E essa discretização precisará ser detalhada, informando como foi realizada.

Este trabalho se propõe apenas a modelar o sistema embarcado, como um AFD e não, a entrar em detalhes de implementação. A modelagem reflete o resultado final do julgamento do boi de uma vaquejada, que é um resultado discreto.

4. Resultados

Na Figura 5 está apresentado o autômato M que modela o sistema.

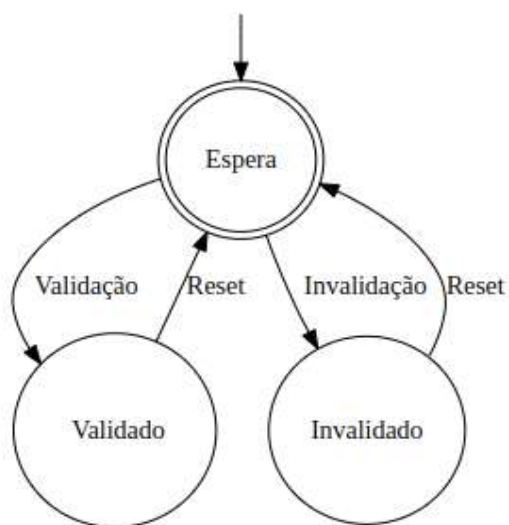


Figura 5. Diagrama de estados do autômato M

O AFD M pode ser formalmente descrito escrevendo $M = (Q, \Sigma, \delta, q_0, F)$, com $e =$ “Espera”, $v =$ “Validado”, $i =$ “Invalidado”, $\omega =$ “Reset”, $1 =$ “Validação”, $0 =$ “Invalidação”, onde:

- $Q = \{e, v, i\}$;
- $\Sigma = \{\omega, 0, 1\}$;
- $\delta = \delta_1: e \times 1 \rightarrow v, \delta_2: e \times 0 \rightarrow i, \delta_3: v \times \omega \rightarrow e, \delta_4: i \times \omega \rightarrow e$;

Tabela 1. Conjunto de transições do autômato M

	ω	0	1
e	–	i	v
v	e	–	–
i	e	–	–

- $q_0 = e$;
- $F = \{e\}$;

A linguagem L, que representa o autômato M é:

$L = \{w \mid w \text{ é uma sequência contendo qualquer quantidade das strings } 0\omega \text{ ou } 1\omega\}$ (1).

L pode ser representada pela expressão regular $((0\omega) \cup (1\omega))^*$ (2).

4.1. Interpretação da definição do autômato M

4.1.1. Conjunto de Estados

4.1.1.1. Estado “Espera” (e)

Indica ao sistema, que ele deverá apresentar ao usuário, por meio de atuadores, que o sistema está no estado “Espera”.

O estado “Espera” indica ao usuário, que o sistema está pronto para realizar novas leituras e fazer o julgamento da validação do boi.

4.1.1.2. Estado “Validado” (v)

Indica ao sistema, que ele deverá apresentar ao usuário, por meio de atuadores, que o sistema está no estado “Validado”.

O estado “Validado” indica ao usuário que o sistema realizou o julgamento do boi e confirmou a validação.

4.1.1.3. Estado “Invalidado” (i)

Indica ao sistema, que ele deverá apresentar ao usuário, por meio de atuadores, que o sistema está no estado “Invalidado”.

O estado “Invalidado” indica ao usuário que o sistema realizou o julgamento do boi e confirmou a invalidação.

4.1.2. Alfabeto

4.1.2.1. Símbolo “Validação” (1)

O sistema recebeu um conjunto de entradas indicando que o boi foi validado.

4.1.2.2. Símbolo “Invalidação” (0)

O sistema recebeu um conjunto de entradas indicando que o boi foi invalidado..

4.1.2.3. Símbolo “Reset” (ω)

O sistema recebeu um conjunto de entradas indicando que ele deve voltar ao estado inicial.

4.1.3. Estado Inicial

4.1.3.1. Estado “Espera” (e)

Indica que, ao ser inicializado, o sistema deve iniciar no estado “Espera”, com o intuito de aguardar instruções do usuário.

4.1.4. Conjunto de estados de aceitação

4.1.4.1. Estado “Espera” (e)

Indica que, ao ser finalizado, o sistema deve finalizar no estado “Espera”. Com o intuito de garantir que quaisquer leituras tenham sido capturadas pelo usuário.

4.1.5. Conjunto de Transições

4.1.5.1. δ_1 : “Espera” X “Validação” \rightarrow “Validado”

Ocorre quando o sistema está no estado “Espera” e lê uma entrada de “Validação”. O sistema transita para o estado “Validado”, com o intuito de indicar a validação do boi.

4.1.5.2. δ_2 : “Espera” X “Invalidação” \rightarrow “Invalidado”

Ocorre quando o sistema está no estado “Espera” e lê uma entrada de “Invalidação”. O sistema transita para o estado “Invalidado”, com o intuito de indicar a invalidação do boi.

4.1.5.3. δ_3 : “Validado” X “Reset” \rightarrow “Espera”

Ocorre quando o sistema está no estado “Validado” e lê uma entrada de “Reset”. O sistema transita para o estado “Espera”, com o intuito de resetar o sistema.

4.1.5.4. δ_4 : “Invalidado” X “Reset” \rightarrow “Espera”

Ocorre quando o sistema está no estado “Invalidado” e lê uma entrada de “Reset”. O sistema transita para o estado “Espera”, com o intuito de resetar o sistema.

4.2. Computação de algumas strings arbitrárias como entradas do autômato M

4.2.1. 1ω

1. e
2. $e \times 1 \rightarrow v$
3. $v \times \omega \rightarrow e$

Nesse caso, o estado final é o estado de aceitação, logo, a string é aceita.

4.2.2. 0ω

1. e
2. $e \times 0 \rightarrow i$
3. $i \times \omega \rightarrow e$

Nesse caso, o estado final é o estado de aceitação, logo, a cadeia é aceita.

4.2.3. $0\omega 1\omega$

1. e
2. $e \times 0 \rightarrow i$
3. $i \times \omega \rightarrow e$
4. $e \times 1 \rightarrow v$
5. $v \times \omega \rightarrow e$

Nesse caso, o estado final é o estado de aceitação, logo, a cadeia é aceita.

4.2.4. $1\omega 0\omega$

1. e
2. $e \times 1 \rightarrow v$
3. $v \times \omega \rightarrow e$
4. $e \times 0 \rightarrow i$
5. $i \times \omega \rightarrow e$

Nesse caso, o estado final é o estado de aceitação, logo, a string é aceita.

4.2.5. $0\omega\omega 1$

1. e
2. $e \times 0 \rightarrow i$
3. $i \times \omega \rightarrow e$

4. $e \times \omega \rightarrow$ rejeição

O autômato não reconhece esse string, pois não existe uma transição, partindo do estado de espera, lendo o símbolo de reset.

4.3. Algoritmo de validação do boi

O algoritmo 1 (disposto nas figuras 6, 7, 8, e 9), representa o processo de validação do boi, quando o boi está, de alguma forma, entre as duas faixas de pontuação.

Para construção do algoritmo, foram feitas algumas perguntas baseadas nas regras. As respostas a essas perguntas, de forma estruturada num algoritmo, permitem a inferência do resultado do julgamento do boi (validado ou invalidado). Essas perguntas estão numeradas de 1 à 11, dentro do algoritmo, no formato de comentário (linha 1 à linha 13).

No algoritmo também está explicado, no formato de comentário, o que é a variável `rj` (linha 16 à linha 22).

```

1 /*Perguntas a serem respondidas
2 p1: O puxador, montado no seu cavalo, derrubou o boi, puxando-o pelo protetor
   de cauda do bovino?
3 p2: Após adentrar as faixas de pontuação, o boi saiu total ou parcialmente
   (qualquer parte do corpo)? (Se p1 == F)
4 p3: A primeira faixa de pontuação permanece intocada pelas partes superiores
   do boi (do jarrete para cima)? (Se p1==V)
5 p4: O boi está próximo à segunda faixa de pontuação, de forma que ele possa
   se levantar com mais de 50% do seu corpo após a segunda faixa de pontuação?
   (Se p3==V)
6 p5: Faz mais de 30s que o boi foi deitado entre as faixas? (Se p4==V)
7 p6: O boi se levantou? (Se p4==V)
8 p7: Mais de 50% do boi está entre as duas faixas de pontuação? (Se p4==V)
9 p8: Após 30 segundos do boi ser deitado entre as faixas. Alguma parte do boi
   permanece dentro da faixa. O boi ainda está sendo reposicionado entre as
   faixas de pontuação pelos vaqueiros? (Se p6 ==F)
10 p9: O estereiro passou pelo boi e voltou para reposicionar o boi entre as
    faixas de pontuação?
11 p10: Algum vaqueiro não montado teve contato físico com o boi ao tentar
    reposicioná-lo?
12 p11: O puxador agiu pelo lado errado?
13 */
14
15
16 // rj: resultado do julgamento
17 // rj pode assumir os valores -1, 0 e 1
18 // -1 representa que o que resultado ainda não foi definido, está em espera
19 // 0 representa que o boi não valeu, foi invalidado
20 // 1 representa que o boi valeu, foi validado
21 // rj é inicializado com o valor -1, representando que ainda não existe um
   resultado final
22 // rj finaliza com o valor 0 ou 1, representando o resultado final do
   julgamento

```

Figura 6. Algoritmo 1 - parte 1.

```

23 // funções booleanas podem retornar verdadeiro (V) ou falso (F)
24 //As função fp1, ..., fp11 são funções booleanas (retornam V ou F)
25 fp1()
26     se ("foi verificado que o puxador, montado no seu cavalo, derrubou o boi,
    puxando-o pelo protetor de cauda do bovino")
27         retorne V
28     se não
29         retorne F
30 fp2()
31     se ("foi verificado que após adentrar as faixas de pontuação, o boi saiu
    total ou parcialmente")
32         retorne V
33     se não
34         retorne F
35 fp3()
36     se ("foi verificado que a primeira faixa de pontuação permanece intocada
    pelas partes superiores do boi")
37         retorne V
38     se não
39         retorne F
40 fp4()
41     se ("foi verificado que o boi está próximo à segunda faixa de pontuação,
    de forma que ele possa se levantar com mais de 50% do seu corpo após a
    segunda faixa de pontuação")
42         retorne V
43     se não
44         retorne F
45 fp5()
46     se ("foi verificado que faz mais de 30s que o boi foi deitado entre as
    faixas")
47         retorne V
48     se não
49         retorne F

```

Figura 7. Algoritmo 1 - parte 2.

```

50 fp6()
51     se ("foi verificado que o boi se levantou")
52         retorne V
53     se não
54         retorne F
55 fp7()
56     se ("foi verificado que mais de 50% do boi está entre as duas faixas de
    pontuação")
57         retorne V
58     se não
59         retorne F
60 fp8()
61     se ("foi verificado que após 30 segundos do boi ser deitado entre as
    faixas, alguma parte do boi permanece dentro da faixa, e o boi ainda está
    sendo reposicionado entre as faixas de pontuação pelos vaqueiros")
62         retorne V
63     se não
64         retorne F
65 fp9()
66     se ("foi verificado que o estereiro passou pelo boi e voltou para
    reposicionar o boi entre as faixas de pontuação")
67         retorne V
68     se não
69         retorne F
70 fp10()
71     se ("foi verificado que algum vaqueiro não montado teve contato físico
    com o boi ao tentar reposicioná-lo")
72         retorne V
73     se não
74         retorne F
75 fp11()
76     se ("foi verificado que o puxador agiu pelo lado errado")
77         retorne V
78     se não
79         retorne F

```

Figura 8. Algoritmo 1 - parte 3.

```

75 fp11()
76 se ("foi verificado que o puxador agiu pelo lado errado")
77     retorne V
78 se não
79     retorne F
80 inicio
81 rj = -1
82 enquanto rj == -1
83     se fp1() == V // "boi caiu"
84         se fp3() == V // "não encostou na primeira faixa"
85             se fp4() == F // "não está próximo a segunda faixa de pontuação"
86                 rj = 1 // boi validado
87             enquanto rj == -1 // "está próximo a segunda faixa de pontuação"
88                 se fp9() == V ou fp10() == V ou fp11() == V // "boi foi
            invalidado por descumprimento de alguma regra"
89                     rj = 0 // boi invalidado
90                 se fp5() == V // "após 30s da queda"
91                     se fp7() == V // "mais de 50% entre as faixas"
92                         rj = 1 // boi validado
93                     se não se fp8() == V // "vaqueiro reposicionando o boi"
94                         rj = 1 // boi validado
95                     se não
96                         rj = 0 // boi invalidado
97                     se não se fp6() == V e fp7() == V // "boi levantou com mais
            de 50% entre as faixas"
98                         rj = 1 // boi validado
99                     se não // "encostou na primeira faixa"
100                         rj = 0 // boi invalidado
101                 se não se fp2() == V // boi saiu das faixas
102                     rj = 0 // boi invalidado
103                 se fp9() == V ou fp10() == V ou fp11() == V // "boi foi invalidado por
            descumprimento de alguma regra"
104                     rj = 0 // boi invalidado
105 imprimir rj
106 fim

```

Figura 9. Algoritmo 1 - parte 4.

4.3.1. Tradução do algoritmo

Enfatiza-se que o algoritmo 1, pode ser traduzido para alguma linguagem de programação, que permita o desenvolvimento do sistema embarcado, como a linguagem de programação C, por exemplo.

Contudo, o programador precisa entender que as funções (fp1, ..., fp11 apresentadas nas figuras 8 e 9) precisam ser implementadas após a tradução, aproveitando-se apenas a sua estrutura, essas funções precisam ser implementadas de acordo com o enunciado de cada uma delas.

Destaca-se também que o algoritmo 1, representa o processo de validação. Não necessariamente, um dispositivo que auxilie o juiz, precisa detectar todas as informações, de forma autônoma. O dispositivo pode interagir com o juiz e solicitar algumas informações. Por exemplo, a função fp11, que se propõe a identificar se o puxador agiu pelo lado errado, poderia haver um botão, para o juiz pressionar, quando o juiz identificasse essa irregularidade.

A tradução de um algoritmo, em uma linguagem de programação, pode ser feita manualmente ou de forma automática, por softwares dedicados a isso, como por exemplo o ChatGPT (do inglês: Chat Generative Pre-Trained Transformer) – um chatbot online de inteligência artificial desenvolvido pela OpenAI.

5. Conclusão

A vaquejada é um esporte bastante praticado no Nordeste brasileiro, e está enraizado de forma profunda na cultura desta região. Utilizar os conhecimentos adquiridos ao longo do curso, em prol da comunidade e abordar essa temática, no meio acadêmico, faz com que a universidade se torne ainda mais presente na sociedade, cumpre, deste modo, um de seus principais papéis, que é propor soluções para demandas da comunidade.

Em síntese, as várias regras da vaquejada, que determinam a validade do boi, são abstraídas pelos símbolos 1 (“validação”) e 0 (“invalidação”), de forma que esses dois símbolos são suficientes em uma abstração de alto nível do sistema.

Portanto, este trabalho cumpriu seu objetivo: modelar um sistema embarcado, como um AFD, que auxilie na validação do boi em vaquejadas, iniciando no meio acadêmico, a discussão sobre a dificuldade em julgar a validação do boi em uma vaquejada e projetando uma possível solução para o problema.

6. Referências bibliográficas

- ABVAQ. Manual de Julgamento de Boi da ABVAQ 2022. João Pessoa, Brasil: ABVAQ, 2022. Disponível em: <https://www.abvaq.com.br/regulamento>. Acesso em: 13 de novembro de 2023.
- ABVAQ. Regulamento Geral da Vaquejada 2023. João Pessoa, Brasil: ABVAQ, 2023. Disponível em: <https://www.abvaq.com.br/regulamento>. Acesso em: 13 de novembro de 2023.
- Chwif, L., Medina, A. C. Modelagem e simulação de eventos discretos: teoria e aplicações, 3 ed. São Paulo: Ed. do Autor, 2010
- Gantois, Rodrigo Amaral. Fair Play na arbitragem: a tecnologia no futebol. a importância do auxílio da tecnologia nas partidas do esporte mais popular do mundo. 2015.
- Gribkoff, Eric. "Applications of deterministic finite automata." UC Davis (2013): 1-9.
- Hopcroft, R. M. J. E.; Ullman, J. D. Introduction to Automata Theory, Languages and Computation. 2. ed. [S.l.]: ser. Series in Computer Science. Addison-Wesley, 2001.
- IST LEIC-A. Elementos da Matemática Discreta. Disponível em: <https://resumos.leic.pt/emd/archive/automatos-finitos/#aut%C3%B3mato-completo>. Acesso em: 26 de janeiro de 2024.
- Katz, Larry. Inovações na Tecnologia Esportiva: Implicações para o Futuro. Revista Educação Física, n. 3, p. 27-32, 2002.
- Lewis, H. R.; Papadimitriou, C. H. Elements of the Theory of Computation. 2. ed. New Jersey: Prentice-Hall, 1998.
- MODELO. In: OXFORD LANGUAGES. Oxford, UK: Copyright, 2023. Disponível em: <https://www.google.com/search?q=defini%C3%A7%C3%A3o+de+modelo&oq=defini%C3%A7%C3%A3o+de&aqs=chrome.0.69i59j69i57j0i512l8.4925j0j7&sourceid=chrome&ie=UTF-8>. Acesso em: 16 de novembro de 2023.
- Neto, J. dos Santos, & Silva, A. A. Estudo do Impacto da Adoção de Tecnologias Digitais no Futebol. RECIMA21 - Revista Científica Multidisciplinar, 2023 - ISSN 2675-6218
- Noergaard, T. Embedded Systems Architecture: a Comprehensive Guide for Engineers and Programmers. Burlington, MA 01803, USA: Elsevier, 2005. cap.1, p. 5.
- Rodrigues, F. L. F.; Cavalcante, C. H. L. Desenvolvimento Do Jogo Vaquejada Para Dispositivos Android, Utilizando Como Motor Gráfico A Godot Engine. Canindé: Instituto Federal de Educação, Ciência e Tecnologia do Ceará - IFCE, 2017.
- Sayão, Luís Fernando. "Modelos Teóricos em Ciência da Informação-abstração e Método Científico." Ciência da informação, v. 30, p. 82-91, 2001.
- Silva, M. V. G. Autômatos, Computabilidade e Complexidade Computacional, 2023.
- Sipser, M. Introdução à Teoria da Computação. São Paulo: Cengage Learning, 2007. Original: Introduction to the Theory of Computation, 2ed.
- Veiga, Maurício de Figueiredo Corrêa da. Da tourada em Portugal à Vaquejada no Brasil: Aspectos Jurídicos. Lisboa, Portugal: Universidade Autónoma de Lisboa, 2020.

Anexo A - Regras da derrubada do boi, na vaquejada, de acordo com o manual de julgamento da ABVAQ 2022.

1) A primeira faixa de pontuação é intocável pelas partes superiores do boi.

a) É considerada, parte superior do boi, da linha imaginária (Figura 10) onde se localiza o jarrete (joelho ou parte seca) para cima.



Figura 10. Métrica para considerar a derrubada do boi.

2) Só será válido o boi, se o mesmo, em algum momento da ação do puxador, ao deitar-se no solo, soltar-se completamente e, ao levantar-se (considerando "levantar-se", o momento em que o boi retoma o contato das extremidades de suas 4 patas com o solo, ou seja, o casco de cada uma delas tocar o solo e se firmar completamente) estiver com as patas entre as duas faixas de pontuação.

a) O boi deverá ser julgado, quando o mesmo estiver em pé, depois das 4 (quatro) patas firmadas, jamais no momento em que o boi tirar a barriga do solo. Boi alavancado não pode ser considerado firmado e, portanto, não pode ser julgado.

b) Só será válido, o boi na faixa, se o puxador agir pelo lado certo (se de direita ou de esquerda).

c) O boi na faixa só terá validade se, e somente se, tiver a ação direta do puxador, ou seja, o vaqueiro puxador, desde que esteja montado no seu cavalo, deve ter puxado no protetor de cauda do bovino.

d) Se o puxador não conseguir validar o boi de imediato, o esteireiro pode auxiliá-lo com o seu animal, desde que venha na corrida.

e) Se o esteireiro passar pelo boi, não poderá mais voltar, para completar a pontuação do mesmo, salvo se ainda estiver ocorrendo ação da puxada ainda em movimento.

f) Se o boi cair sozinho dentro da faixa de pontuação, sem a ação direta do puxador, não vale para pontuação, será zero (0).

g) Em caso de segundo cal, em que se precise julgar, se o boi saiu ou não, mais de 50%,

deverá o juiz de pista proceder da seguinte forma:

g.1 Dentro dos 30 segundos previstos no regulamento geral da ABVAQ, caso o boi se levante fora do segundo cal, será julgado zero pelo juiz de pista;

g.2 Caso o competidor consiga reposicionar o boi, para que se levante entre as faixas, ou se o boi permanecer imóvel, no prazo de 30 segundos, o boi deverá ser remetido à comissão alternativa, sem ônus para o vaqueiro, para que seja julgado a questão dos 50%.


g.3 A comissão alternativa, verificando que o boi estava com menos 50% para fora, validará o boi, independentemente de onde o bovino se firme, caso contrário, permanece zero.

h) Após o boi ser deitado entre as faixas, o juiz terá no máximo 30 segundos, para julgá-lo. Caso o boi ainda esteja sendo trabalhado pelo vaqueiro, o juiz deverá julgá-lo em favor do competidor.

i) O vaqueiro, caso não esteja montado no seu cavalo, só poderá participar do reposicionamento do bovino, quer na faixa ou em qualquer outro local da pista de competição, se não houver contato físico com o boi. Caso haja contato físico, nesse caso, o boi será julgado zero.

3) Ao adentrar na faixa de pontuação, o boi não poderá mais sair da mesma total ou parcialmente antes de dar ponto.

a) Se, ao adentrar na faixa de pontuação, o boi tocar a parte externa da faixa com qualquer parte do corpo, antes de dar ponto, o mesmo será julgado zero, mesmo que o puxador, seguindo na puxada, consiga trazê-lo para dentro e deitá-lo.

	INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
	Campus Campina Grande
	R. Tranquílino Coelho Lemos, 671, Dinamérica, CEP 58432-300, Campina Grande (PB)
	CNPJ: 10.783.898/0003-37 - Telefone: (83) 2102.6200

Documento Digitalizado Ostensivo (Público)

entrega do TCC e termos de autorização de postagem em repositório digital

Assunto:	entrega do TCC e termos de autorização de postagem em repositório digital
Assinado por:	Mozart Nascimento
Tipo do Documento:	Dissertação
Situação:	Finalizado
Nível de Acesso:	Ostensivo (Público)
Tipo do Conferência:	Cópia Simples

Documento assinado eletronicamente por:

- **Mozart Lima do Nascimento, ALUNO (201721250034) DE BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO - CAMPINA GRANDE**, em 26/02/2024 13:12:26.

Este documento foi armazenado no SUAP em 26/02/2024. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 1093330

Código de Autenticação: c06389d747

