UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ARTES, CIÊNCIAS E HUMANIDADES
PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO

IGOR OLIVEIRA BORGES

**Estratégias para otimização do algoritmo de Iteração de Valor Sensível a
Risco**

São Paulo

2018

IGOR OLIVEIRA BORGES

**Estratégias para otimização do algoritmo de Iteração de Valor Sensível a
Risco**

Dissertação apresentada à Escola de Artes, Ciências e Humanidades da Universidade de São Paulo para obtenção do título de Mestre em Ciências pelo Programa de Pós-graduação em Sistemas de Informação.

Área de concentração: Metodologia e Técnicas da Computação

Versão corrigida contendo as alterações solicitadas pela comissão julgadora em 11 de outubro de 2018. A versão original encontra-se em acervo reservado na Biblioteca da EACH-USP e na Biblioteca Digital de Teses e Dissertações da USP (BDTD), de acordo com a Resolução CoPGr 6018, de 13 de outubro de 2011.

Orientador: Profa. Dra. Karina Valdivia Delgado

São Paulo

2018

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

CATALOGAÇÃO-NA-PUBLICAÇÃO

(Universidade de São Paulo. Escola de Artes, Ciências e Humanidades. Biblioteca)
CRB-8 4936

Borges, Igor Oliveira
Estratégias para otimização do algoritmo de Iteração de Valor
Sensível a Risco / Igor Oliveira Borges ; orientadora, Karina Valdivia
Delgado. – 2018.
60 f. : il.

Dissertação (Mestrado em Ciências) - Programa de Pós-
Graduação em Sistemas de Informação, Escola de Artes,
Ciências e Humanidades, Universidade de São Paulo.
Versão corrigida

1. Inteligência artificial. 2. Processos de Markov.
3. Processos estocásticos. 4. Programação dinâmica.
5. Heurística. I. Delgado, Karina Valdivia, orient. II. Título.

CDD 22.ed.— 006.3

Dissertação de autoria de Igor Oliveira Borges, sob o título “**Estratégias para otimização do algoritmo de Iteração de Valor Sensível a Risco**”, apresentada à Escola de Artes, Ciências e Humanidades da Universidade de São Paulo, para obtenção do título de Mestre em Ciências pelo Programa de Pós-graduação em Sistemas de Informação, na área de concentração Metodologia e Técnicas da Computação, aprovada em 11 de outubro de 2018 pela comissão julgadora constituída pelos doutores:

Profa. Dra. Karina Valdivia Delgado

Instituição: Universidade de São Paulo

Presidente

Prof. Dr. Carlos Roberto Lopes

Instituição: Universidade Federal de Uberlândia

Profa. Dra. Leliane Nunes de Barros

Instituição: Universidade de São Paulo

Prof. Dr. Masayuki Oka Hase

Instituição: Universidade de São Paulo

Para minha mãe Margareth e meu pai Wilson.

À minha querida vó Rosa e ao meu esforçado avô Anísio,
à querida vó Carmen e ao meu respeitado avô C^{el}. Enemêncio,
por todos seus ensinamentos.

Agradecimentos

Agradeço, primeiramente ao criador do universo e natureza, pela vida.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Agradeço fortemente a minha orientadora Dra. *Karina Valdivia Delgado* e ao professor Dr. *Valdinei Freire da Silva*, por todo apoio e dedicação que tiveram com a minha pesquisa durante o mestrado.

Agradeço ainda a todos meus familiares, pelo suporte.

E aos meus amigos e colegas, pela sinergia.

"The policy of being too cautious is the greatest risk of all."

(Jawaharlal Nehru)

Resumo

BORGES, Igor Oliveira. **Estratégias para otimização do algoritmo de Iteração de Valor Sensível a Risco**. 2018. 60 f. Dissertação (Mestrado em Ciências) – Escola de Artes, Ciências e Humanidades, Universidade de São Paulo, São Paulo, 2018.

Processos de decisão markovianos sensíveis a risco (*Risk Sensitive Markov Decision Process* – RS-MDP) permitem modelar atitudes de aversão e propensão ao risco no processo de tomada de decisão usando um fator de risco para representar a atitude ao risco. Para esse modelo, existem operadores que são baseados em funções de transformação linear por partes que incluem fator de risco e fator de desconto. Nesta dissertação são formulados dois algoritmos de Iteração de Valor Sensível a Risco baseados em um desses operadores, esses algoritmos são chamados de Iteração de Valor Sensível a Risco Síncrono (*Risk Sensitive Value Iteration* – RSVI) e Iteração de Valor Sensível a Risco Assíncrono (*Asynchronous Risk Sensitive Value Iteration* – A-RSVI). Também são propostas duas heurísticas que podem ser utilizadas para inicializar os valores dos algoritmos de forma a torná-los mais eficientes. Os resultados dos experimentos no domínio de Travessia do Rio em dois cenários de recompensas distintos mostram que: (i) o custo de processamento de políticas extremas a risco, tanto de aversão quanto de propensão, é elevado; (ii) um desconto elevado aumenta o tempo de convergência do algoritmo e reforça a sensibilidade ao risco adotada; (iii) políticas com valores para o fator de risco intermediários possuem custo computacional baixo e já possuem certa sensibilidade ao risco dependendo do fator de desconto utilizado; e (iv) o algoritmo A-RSVI com a heurística baseada no fator de risco pode reduzir o tempo para o algoritmo convergir, especialmente para valores extremos do fator de risco.

Palavras-chaves: Processo de Decisão Markoviano Sensível a Risco. Planejamento Estocástico. Política Sensível a Risco.

Abstract

BORGES, Igor Oliveira. **Strategies for optimization of Risk Sensitive Value Iteration algorithm**. 2018. 60 p. Dissertation (Master of Science) – School of Arts, Sciences and Humanities, University of São Paulo, São Paulo, 2018.

Risk Sensitive Markov Decision Process (RS-MDP) allows modeling risk-averse and risk-prone attitudes in decision-making process using a risk factor to represent the risk-attitude. For this model, there are operators that are based on a piecewise linear transformation function that includes a risk factor and a discount factor. In this dissertation we formulate two Risk Sensitive Value Iteration algorithms based on one of these operators, these algorithms are called Synchronous Risk Sensitive Value Iteration (RSVI) and Asynchronous Risk Sensitive Value Iteration (A-RSVI). We also propose two heuristics that can be used to initialize the value of the RSVI or A-RSVI algorithms in order to make them more efficient. The results of experiments with the River domain in two distinct rewards scenarios show that: (i) the processing cost in extreme risk policies, for both risk-averse and risk-prone, is high; (ii) a high discount value increases the convergence time and reinforces the chosen risk attitude; (iii) policies with intermediate risk factor values have a low computational cost and show a certain sensitivity to risk based on the discount factor; and (iv) the A-RSVI algorithm with the heuristic based on the risk factor can decrease the convergence time of the algorithm, especially when we need a solution for extreme values of the risk factor.

Keywords: Risk Sensitive Markov Decision Process. Stochastic Planning. Risk Sensitive Policy.

Lista de figuras

Figura 1 – Modelo de tomada de decisão em MDP.	23
Figura 2 – Função utilidade $u(R) = -\text{sgn}(\lambda)\exp(-\lambda R)$ para $\lambda = 0.99$ e $\lambda = -0.99$; e função utilidade $u(R) = R$	28
Figura 3 – Função $\mathcal{X}^k(x)$ para $k = -0.99$, $k = 0$ e $k = 0.99$	30
Figura 4 – Exemplo de um MDP com dois estados.	31
Figura 5 – Algoritmo de Iteração de Valor Sensível a Risco com heurística para inicializar os valores de $Q_h(s, a)$	39
Figura 6 – Instância do domínio de travessia do rio.	42
Figura 7 – Tempo de convergência por configuração do algoritmo de Iteração de Valor Sensível a Risco na instância do rio de tamanho 10×7 no cenário de recompensas (+) para os fatores de risco k ($k = -0.99$, $k = -0.8$, $k = -0.5$, $k = 0$, $k = 0.5$, $k = 0.8$ e $k = 0.99$) e fator de desconto γ ($\gamma = 0.6, \gamma = 0.7, \gamma = 0.8, \gamma = 0.9$ e $\gamma = 0.99$).	48
Figura 8 – Tempo de convergência por configuração do algoritmo de Iteração de Valor Sensível a Risco na instância do rio de tamanho 10×7 no cenário de recompensas (−) para os fatores de risco k ($k = -0.99$, $k = -0.8$, $k = -0.5$, $k = 0$, $k = 0.5$, $k = 0.8$ e $k = 0.99$) e fator de desconto γ ($\gamma = 0.6, \gamma = 0.7, \gamma = 0.8, \gamma = 0.9$ e $\gamma = 0.99$).	49
Figura 9 – Tempo de processamento nas configurações RSVI- $R\alpha$, RSVI- $R\alpha Hk$ e A-RSVI- $R\alpha Hk$ por tamanho de <i>grid</i> e cenário de recompensas (−) com o fator de desconto $\gamma = 0.99$, para dois valores extremos do fator de risco k ($k = -0.99$ e $k = 0.99$).	50
Figura 10 – Valor do estado inicial durante as iterações nas configurações RSVI- $R\alpha$, RSVI- $R\alpha Hk$ e A-RSVI- $R\alpha Hk$ no cenário de recompensas (−) para a instância do rio de tamanho 10×7 , para dois valores extremos do fator de risco k ($k = -0.99$ e $k = 0.99$) e fator de desconto γ ($\gamma = 0.6, \gamma = 0.7, \gamma = 0.8, \gamma = 0.9$ e $\gamma = 0.99$).	52

Figura 11 – Valor do residual relativo durante as iterações nas configurações RSVI-
 $R\alpha$, RSVI- $R\alpha Hk$ e A-RSVI- $R\alpha Hk$ no cenário de recompensas $(-)$
 para a instância do rio de tamanho 10×7 , para dois valores extremos
 do fator de risco k ($k = -0.99$ e $k = 0.99$) e fator de desconto γ
 ($\gamma = 0.6, \gamma = 0.7, \gamma = 0.8, \gamma = 0.9$ e $\gamma = 0.99$). 53

Lista de algoritmos

Algoritmo 1 – Algoritmo de Iteração de Valor	25
Algoritmo 2 – Algoritmo de Iteração de Valor Sensível a Risco Síncrono (RSVI)	36
Algoritmo 3 – Algoritmo de Iteração de Valor Sensível a Risco Assíncrono (A-RSVI)	37
Algoritmo 4 – Heurística α	40

Lista de tabelas

Tabela 1 – Valores de α baseados no valor de k	43
Tabela 2 – Políticas obtidas pelo algoritmo de Iteração de Valor Sensível a Risco em um <i>grid</i> 10×7 conforme cenário de recompensa adotado.	45
Tabela 3 – Políticas obtidas pelo algoritmo de Iteração de Valor Sensível a Risco em um <i>grid</i> 3×7 conforme cenário de recompensa adotado.	47

Lista de abreviaturas e siglas

A-RSVI	<i>Asynchronous Risk Sensitive Value Iteration</i>
H_α	<i>Heuristic α</i>
H_k	<i>Heuristic k</i>
MDP	<i>Markov Decision Process</i>
RL	<i>Reinforcement Learning</i>
R_α	<i>Relative α</i>
RSMDP	<i>Risk Sensitive Markov Decision Process</i>
RSVI	<i>Risk Sensitive Value Iteration</i>
SARSA	<i>State-Action-Reward-State-Action</i>
S_α	<i>Static α</i>
TD	<i>Temporal Difference</i>
VaR	<i>Value-at-Risk</i>
VI	<i>Value Iteration</i>

Lista de símbolos

a	Ação
$\mathcal{A}$	Conjunto de todas ações
$\mathcal{S}$	Conjunto de todos estados
Π	Conjunto de todas políticas
$\mathbb{R}$	Conjunto dos números reais
i	Contador da iteração
x	Entrada da função
ϵ	Erro mínimo
s	Estado
E	Expectativa do retorno
γ	Fator de desconto
λ	Fator de risco em utilidade exponencial
k	Fator de risco na função linear por partes
$\mathcal{T}$	Função de transição dos estados
$\mathcal{X}$	Função linear por partes
$\mathcal{R}$	Função recompensa
V	Função valor
Q	Função valor estado-ação
Q^*	Função valor estado-ação ótima
V^*	Função valor ótima
$\mathcal{N}$	Operador de ponto-fixo
π	Política

π^*	Política ótima
r	Recompensa imediata
R	Retorno observado
α	Tamanho do passo
u	Utilidade esperada

Sumário

1	Introdução	18
1.1	<i>Objetivo principal</i>	19
1.2	<i>Objetivo secundário</i>	20
1.3	<i>Organização</i>	20
2	Processos de decisão markovianos neutros a risco	22
2.1	<i>Definição de MDP</i>	22
2.2	<i>Política</i>	23
2.3	<i>Avaliação da política</i>	23
2.4	<i>Política ótima</i>	24
2.5	<i>Iteração de Valor</i>	24
3	Processos de decisão markovianos seguros	26
3.1	<i>Classificação de processos de decisão markovianos seguros</i>	26
3.2	<i>Função de utilidade exponencial</i>	27
4	Abordagem de Mihatsch e Neuneier para RS-MDPs	29
4.1	<i>Avaliação da política</i>	30
4.2	<i>Política ótima</i>	32
4.3	<i>Operador de ponto-fixo $\mathcal{N}_{\alpha,k}$</i>	33
5	Algoritmo de Iteração de Valor Sensível a Risco	35
5.1	<i>Algoritmo de Iteração de Valor Sensível a Risco Síncrono</i>	35
5.2	<i>Algoritmo de Iteração de Valor Sensível a Risco Assíncrono</i>	37
6	Heurísticas para o algoritmo de Iteração de Valor Sensível a Risco	39
6.1	<i>Heurística baseada no tamanho do passo α</i>	39
6.2	<i>Heurística baseada no fator de risco k</i>	41
7	Experimentos	42
7.1	<i>Domínio de Travessia do Rio</i>	42
7.2	<i>Configurações do experimento</i>	43

7.3	<i>Políticas Obtidas</i>	44
7.4	<i>Tempo de processamento</i>	46
7.5	<i>Valor do estado inicial e valor do residual</i>	51
7.6	<i>Considerações finais</i>	51
8	Conclusão	55
8.1	<i>Resumo das contribuições</i>	56
8.2	<i>Publicações geradas</i>	56
8.3	<i>Trabalhos futuros</i>	57
	Referências¹	58

¹ De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023.

1 Introdução

Problemas de tomada de decisão sempre foram objeto de estudo em pesquisas científicas em diferentes campos do conhecimento e para resolver esses problemas é necessário planejar. Planejar é organizar ações de modo que, quando elas sejam executadas, seja possível atingir metas pré-definidas. Na Computação, a área de Planejamento em Inteligência Artificial estuda esse processo. Existem diversas sub-áreas em Planejamento em Inteligência Artificial, por exemplo: (i) planejamento clássico, que trabalha com ações que têm efeitos determinísticos; e (ii) planejamento probabilístico que trabalha com problemas com ações que têm efeitos probabilísticos.

Um modelo comumente utilizado em planejamento probabilístico é o processo de decisão markoviano (*Markov Decision Process* – MDP), que permite representar problemas que têm estados, ações, transições entre estados e recompensas. Nesse modelo, o agente deve encontrar uma política ótima que maximize a recompensa acumulada total esperada (PUTERMAN, 1994). Uma política pode ser classificada como sendo estacionária ou não-estacionária. Uma política estacionária escolhe uma ação para cada estado independentemente da época de decisão. Em contrapartida uma política não-estacionária, escolhe uma ação que também depende da época de decisão.

MDPs podem ser utilizados para modelar problemas complexos, por exemplo: (i) em simulações de modelos baseados no mundo real, a fim de alcançar alguma previsibilidade; (ii) na robótica móvel, em que é definida uma sequência de ações para atingir uma meta (e.g. chegar em uma sala, pegar um objeto e carregar a bateria); e (iii) no controle de ações na bolsa de valores, almejando o acúmulo financeiro.

Um ponto a ser considerado durante a tomada de decisão em planejamento probabilístico é como levar em conta o risco. O risco é inerente à própria tomada de decisão que surge a partir das incertezas associadas a eventos futuros, e é inevitável dada a natureza probabilística do próprio problema. Porém, esse risco não necessariamente é otimizado pelo critério do tomador de decisão, i.e., o agente. Um agente que minimiza o custo esperado pode ser considerado um agente neutro ao risco, enquanto um agente sensível ao risco deve escolher entre duas atitudes: aversão ou propensão ao risco (SHEN *et al.*, 2014).

O desenvolvimento de algoritmos sensíveis a risco, i.e. que consideram a sensibilidade ao risco na tomada de decisão, é um tema pouco explorado na literatura (GARCÍA;

FERNÁNDEZ, 2015). Existem diferentes abordagens para quantificar o risco como: utilidade exponencial esperada (HOWARD; MATHESON, 1972; JAQUETTE, 1976; DENARDO; ROTHBLUM, 1979; ROTHBLUM, 1984; PATEK, 2001), o uso de uma função de transformação linear por partes com fator de desconto (MIHATSCH; NEUNEIER, 2002), ponderação entre esperança e variância (SOBEL, 1982; FILAR; KALLENBERG; LEE, 1989) e estimação de desempenho em um intervalo de confiança (FILAR *et al.*, 1995; YU; LIN; YAN, 1998; HOU; YEOH; VARAKANTHAM, 2014; HOU; YEOH; VARAKANTHAM, 2016). Tais abordagens sensíveis a risco são consideradas mais difíceis de serem aplicadas aos problemas e mais custosas computacionalmente, se comparadas às modelagens neutras ao risco (GARCÍA; FERNÁNDEZ, 2015).

Nos trabalhos baseados em utilidade exponencial esperada, deve-se especificar um fator de risco, sendo que os valores factíveis para esse fator de risco dependem do problema de decisão em questão (PATEK, 2001), uma alternativa é considerar fator de desconto, mas nesse caso a política ótima torna-se não estacionária (CHUNG; SOBEL, 1987).

Outra abordagem que permite encontrar uma política ótima estacionária como solução, assim como uma escolha arbitrária para o fator de risco é introduzida em (MIHATSCH; NEUNEIER, 2002). Embora o foco do artigo (MIHATSCH; NEUNEIER, 2002) seja sobre aprendizado por reforço, os autores propõem um conjunto de operadores de ponto-fixo com propriedades de contração, baseado em uma função de transformação linear por partes com fator de desconto, que também pode ser aplicada ao planejamento probabilístico. Existe uma avaliação prévia do papel que o fator de desconto desempenha em diferentes modelos sensíveis ao risco em (FREIRE, 2016), em que mostra que o fator de desconto por si só apresenta uma característica propensa ao risco sob o critério de minimização de custos. Contudo, de acordo com o nosso conhecimento, não existe avaliação empírica dos operadores proposto por Mihatsch e Neuneier (2002).

1.1 *Objetivo principal*

O objetivo principal deste trabalho é formular algoritmos de Iteração de Valor para resolver processos de decisão markovianos sensíveis a risco que usam o operador proposto por Mihatsch e Neuneier (2002). Além disso, o objetivo é propor heurísticas que permitam melhorar o tempo de convergência desses algoritmos, i.e. o custo computacional.

1.2 *Objetivo secundário*

O objetivo secundário deste trabalho é analisar de forma empírica o impacto da escolha do fator de risco e do fator de desconto nos algoritmos de Iteração de Valor Sensível a Risco formulados e que usam o operador proposto por Mihatsch e Neuneier (2002) sob dois aspectos:

- (i) quão sensível a risco (aversa ou propensa) é a política obtida sob tais parâmetros, e
- (ii) como o tempo de convergência do algoritmo de planejamento é afetado por tais parâmetros.

1.3 *Organização*

O restante desta dissertação está organizado da seguinte forma:

- O Capítulo 2 apresenta a definição de processos de decisão markovianos, o conceito de política ótima e os algoritmos existentes para resolver MDPs.
- O Capítulo 3 expõe uma classificação de processos de decisão markovianos seguros e apresenta uma das abordagens sensíveis a risco mais conhecidas, a de função de utilidade exponencial.
- O Capítulo 4 explora a abordagem de Mihatsch e Neuneier (2002) para MDPs sensíveis a risco, mostra como encontrar políticas ótimas e descreve um operador de ponto-fixe e sensível a risco proposto pelo autor.
- O Capítulo 5 descreve os algoritmos de Iteração de Valor Sensíveis a Risco formulados que utilizam o operador proposto por Mihatsch e Neuneier (2002).
- O Capítulo 6 projeta duas heurísticas para os algoritmos de Iteração de Valor Sensível a Risco.
- O Capítulo 7 descreve o domínio utilizado nos experimentos e apresenta os resultados da execução dos algoritmos propostos.

- O Capítulo 8 apresenta a conclusão da dissertação, publicações geradas e trabalhos futuros.

2 Processos de decisão markovianos neutros a risco

Processos de decisão markovianos (MDPs) permitem modelar problemas da área de planejamento probabilístico e de aprendizado por reforço. Em MDPs as transições entre estados são definidas probabilisticamente (PUTERMAN, 1994) e o processo é chamado de markoviano pois o efeito de uma ação em um dado estado depende somente da ação escolhida naquele estado, não levando em conta o histórico de tomadas de decisão sequenciais já realizadas (BELLMAN, 1957).

2.1 Definição de MDP

Formalmente um MDP é uma tupla: $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R} \rangle$, no qual:

- $\mathcal{S}$ é o conjunto finito de estados pertencentes ao processo;
- $\mathcal{A}$ é o conjunto finito de ações que podem ser executadas durante as épocas de decisão;
- $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$ é uma função que define a probabilidade de transição dos estados no sistema, sendo que $\mathcal{T}(s'|s, a)$ representa a probabilidade de chegar no estado $s' \in \mathcal{S}$, dado que o agente está no estado $s \in \mathcal{S}$ e foi escolhida a ação $a \in \mathcal{A}$;
- $\mathcal{R} : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ é uma função recompensa que define a recompensa recebida no estado $s \in \mathcal{S}$ ao tomar uma ação $a \in \mathcal{A}$.

Um processo de decisão markoviano modela a interação entre um agente e seu ambiente e o processo é composto por estágios. Em cada estágio, o agente sabe qual é o estado atual (s_i) e quais ações podem ser tomadas. Essas ações possuem efeitos probabilísticos. O agente decide então executar uma ação a_i que leva o agente a um estado futuro s_{i+1} , e ganha uma recompensa r_{i+1} (Figura 1).

A quantidade de épocas de decisão é chamada de horizonte e pode ser finito (definido por um número fixo), infinito (repetido seguidamente sem parada) ou ainda indeterminado (repetido seguidamente com possibilidade de parada, por exemplo, quando o agente atinge um estado meta ou um beco-sem-saída).

Figura 1 – Modelo de tomada de decisão em MDP.

Fonte: Sutton e Barto (1998).

2.2 Política

A solução de um MDP é uma política π . A política é chamada de total quando o mapeamento é definido para todos os estados do MDP e parcial quando o mapeamento é definido para apenas alguns estados do MDP.

Pode-se também classificar uma política em relação às épocas de decisão como sendo estacionária ou não-estacionária. Uma política estacionária escolhe uma ação independentemente da época de decisão. Em contrapartida uma política não-estacionária, escolhe uma ação que depende da época de decisão.

Uma outra classificação das políticas é quanto ao mapeamento de estados como sendo determinística ou estocástica. Na política determinística, para cada estado existe apenas uma ação. Na política estocástica é definida uma distribuição de probabilidade no espaço de ações, em que cada ação possui uma probabilidade de ser selecionada.

O tema desta dissertação de mestrado está relacionado com o uso de políticas totais, estacionárias e determinísticas, i.e, uma função que mapeia estados em ações ($\pi : \mathcal{S} \rightarrow \mathcal{A}$).

2.3 Avaliação da política

O valor $V^\pi(s)$ de uma política em um estado $s \in \mathcal{S}$ é determinado por:

$$V^\pi(s) = \mathbb{E} \left[\sum_{t=0}^{\infty} \gamma^t r_t \middle| \pi, s_0 = s \right], \quad (1)$$

A recompensa em cada estágio de decisão t é representada por r_t e o fator de desconto $\gamma \in [0, 1)$ é utilizado para garantir a convergência do valor da recompensa total esperada em horizontes infinitos.

2.4 Política ótima

Seja Π o conjunto de políticas estacionárias, a função valor ótima $V^*(s) = \max_{\pi \in \Pi} V^\pi(s)$ é a solução da equação de Bellman (BELLMAN; KALABA, 1965):

$$V^*(s) = \left[\max_{a \in A} \sum_{s' \in \mathcal{S}} \mathcal{T}(s'|s, a) [\mathcal{R}(s, a) + \gamma V^*(s')] \right], \forall s \in \mathcal{S}. \quad (2)$$

A política π^* ótima pode ser obtida com base na função valor ótima por:

$$\pi^*(s) = \left[\arg \max_{a \in A} \sum_{s' \in \mathcal{S}} \mathcal{T}(s'|s, a) [\mathcal{R}(s, a) + \gamma V^*(s')] \right], \forall s \in \mathcal{S}. \quad (3)$$

Um resultado importante da teoria de MDPs de horizonte infinito é que existe uma política ótima determinística e estacionária (PUTERMAN, 1994).

Em MDPs, o critério de otimalidade adotado é baseado na maximização da recompensa esperada acumulada, sendo por isto considerados neutros ao risco pois não estimam o risco inerente existente no próprio ambiente.

2.5 Iteração de Valor

O algoritmo de Iteração de Valor (*Value Iteration* – VI) é um algoritmo de programação dinâmica. Em cada iteração i é calculado o valor $V^i(s)$ baseado no valor $V^{i-1}(s)$ para cada estado $s \in \mathcal{S}$ do MDP, isto é:

$$V^i(s) \leftarrow \max_{a \in A} \sum_{s' \in \mathcal{S}} \mathcal{T}(s'|s, a) [R(s, a) + \gamma V^{i-1}(s')], \forall s \in \mathcal{S}. \quad (4)$$

Ao longo das iterações o valor de V^i converge para a função valor ótima V^* , mesmo partindo de qualquer estimativa inicial V^0 .

Um possível critério de parada no algoritmo de Iteração de Valor é considerar a diferença entre o valor na iteração i e $i - 1$, conhecido como erro residual, i.e., $\max_{s \in \mathcal{S}} |V^i(s) - V^{i-1}(s)|$ e iterar enquanto o residual for maior que um erro mínimo desejado ϵ .

O algoritmo de Iteração de Valor (Algoritmo 1) recebe como entrada um MDP, um fator desconto e um erro mínimo ϵ que é usado no critério de parada; e devolve como saída o valor ótimo para todo estado do problema com a precisão esperada. Nas Linhas 1 a 3 é inicializado o valor V^0 com a maior recompensa para cada estado. Em cada iteração i o

algoritmo atualiza os valores (Linha 7) e calcula o residual (Linha 8) de todos os estados. Por fim, devolve V na Linha 12.

Algoritmo 1 Algoritmo de Iteração de Valor

Entrada: MDP $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R})$, γ , ϵ

Saída : Função valor V para o MDP de entrada

```

1 para cada  $s \in \mathcal{S}$  faça
2    $V^0(s) \leftarrow \max_{a \in \mathcal{A}} \mathcal{R}(s, a)$ 
3 fim
4  $i \leftarrow 1$ 
5 faça
6   para cada  $s \in \mathcal{S}$  faça
7      $V^i(s) \leftarrow \max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} T(s'|s, a)[R(s, a) + \gamma V^{i-1}(s')]$ 
8      $\text{residual}(s) \leftarrow |V^i(s) - V^{i-1}(s)|$ 
9   fim
10   $i \leftarrow i + 1$ 
11 enquanto  $\max_{s \in \mathcal{S}} \{\text{residual}(s)\} > \epsilon$ ;
12 retorna  $V$ 
```

Fonte: Adaptado de (PUTERMAN, 1994).

3 Processos de decisão markovianos seguros

Dada a limitação da neutralidade ao risco que caracteriza a função utilidade em MDPs clássicos, desenvolveu-se pesquisas na área de MDPs para aplicações que necessitem lidar com outro tipo de critérios.

3.1 Classificação de processos de decisão markovianos seguros

García e Fernández (2015) apresentam uma revisão da literatura da área de processos de decisão markovianos chamados pelos autores de seguros, em que os algoritmos existentes são classificados pelo critério de otimização e pelo processo de exploração usado. O tema desta dissertação de mestrado está relacionado com o critério de otimização usado. Os MDPs seguros de acordo com o critério de otimização adotado podem ser classificados em:

- **Critério do pior caso:** a política ótima é aquela que maximiza o retorno esperado no pior caso, por isto é considerada aversa ao risco (HEGER, 1994; GASKETT, 2003; NILIM; GHAOUI, 2005; TAMAR; XU; MANNOR, 2014).
- **Critério sensível a risco:** incorpora o risco na tomada de decisão. Existem várias abordagens para este critério, dentre elas destacam-se utilidade exponencial esperada (HOWARD; MATHESON, 1972; JAQUETTE, 1976; DENARDO; ROTHBLUM, 1979; ROTHBLUM, 1984; PATEK, 2001), função de transformação linear por partes com fator de desconto (MIHATSCH; NEUNEIER, 2002), ponderação entre esperança e variância (SOBEL, 1982; FILAR; KALLENBERG; LEE, 1989) e estimação de desempenho em um intervalo de confiança (FILAR *et al.*, 1995; YU; LIN; YAN, 1998; HOU; YEOH; VARAKANTHAM, 2014; HOU; YEOH; VARAKANTHAM, 2016).
- **Critério com restrições:** consiste em maximizar o retorno sujeito a um conjunto de restrições. (MOLDOVAN; ABBEEL, 2012; CASTRO; TAMAR; MANNOR, 2012; KADOTA; KURANO; YASUDA, 2006). Tais restrições permitem estabelecer um conjunto de medidas esperadas a serem consideradas na otimização do problema.
- **Outros critérios:** baseados na construção de políticas focadas a determinadas aplicações, em especial a otimização financeira, como o do coeficiente de determinação (*r-squared*), valor do risco (*Value-at-Risk* – VaR) e densidade do retorno

(MORIMURA *et al.*, 2010; LUENBERGER, 2013; CASTRO; TAMAR; MANNOR, 2012).

3.2 Função de utilidade exponencial

Uma das abordagens clássicas para quantificar o risco é a utilidade exponencial (HOWARD; MATHESON, 1972; JAQUETTE, 1976; DENARDO; ROTHBLUM, 1979; ROTHBLUM, 1984; PATEK, 2001). Entretanto, nessa abordagem é preciso definir um fator de risco, e os valores factíveis para esse fator de risco dependem do problema de decisão (PATEK, 2001). Uma solução para tornar todo problema factível é considerar um fator de desconto; embora, neste caso, a política ótima é não-estacionária (CHUNG; SOBEL, 1987).

Essa abordagem que utiliza função de utilidade exponencial é baseada na teoria da utilidade e a função utilidade u utilizada nessa abordagem é:

$$u(R) = -\text{sgn}(\lambda) \exp(-\lambda R),$$

em que λ é o fator de risco e R é o retorno (i.e., $R = \sum_{t=0}^{\infty} \gamma^t r_t$). Se $\lambda < 0$, tem-se a propensão ao risco, em contrapartida se $\lambda > 0$, tem-se aversão ao risco, e quando $\lambda \rightarrow 0$, tem-se a neutralidade no risco (HOWARD; MATHESON, 1972). A função objetivo desta abordagem maximiza a expectativa do retorno exponencial, i.e.:

$$\pi^* = \arg \max_{\pi \in \Pi} \lambda^{-1} \log E_{\pi}[\exp(\lambda R)].$$

Na Figura 2 são mostradas as curvas da função utilidade $u(R) = -\text{sgn}(\lambda) \exp(-\lambda R)$ para $\lambda = 0.99$ e $\lambda = -0.99$. Veja que a curva verde que representa a atitude do agente com aversão ao risco ($\lambda = 0.99$) é côncava e a curva vermelha que representa a atitude do agente propenso ($\lambda = -0.99$) é convexa. Já a função identidade $u(R) = R$ (linha azul) representa uma atitude neutra a risco.

Nesta abordagem, o custo de se calcular o exponencial do retorno, i.e., $\exp(\lambda R)$ pode ser proibitivo a muitas aplicações, em especial quando R é um número muito grande, neste caso pode ocorrer o problema de *overflow* (estouro) (GOSAVI, 2009). Esse estouro da representação numérica do dado em memória pode comprometer o cálculo completo e por tanto o algoritmo pode devolver uma política errada. Assim, uma implementação cuidadosa precisa atentar-se à capacidade de representação do tipo de dado utilizado e

Figura 2 – Função utilidade $u(R) = -\text{sgn}(\lambda)\exp(-\lambda R)$ para $\lambda = 0.99$ e $\lambda = -0.99$; e função utilidade $u(R) = R$.

Fonte: Baseado em Howard e Matheson (1972).

da operação para o cálculo em cada etapa, evitando ultrapassar os respectivos limites numéricos que existem tanto para valores positivos quanto negativos. Esse problema é mais acentuado nos extremos de risco, tanto para aversão quanto para propensão.

4 Abordagem de Mihatsch e Neuneier para RS-MDPs

Outra abordagem para lidar com o risco é a proposta em (MIHATSCH; NEUNEIER, 2002) que é baseada em uma função linear por partes. Essa abordagem permite encontrar uma política estacionária como solução, bem como uma escolha arbitrária para o fator de risco. A abordagem de Mihatsch e Neuneier se concentra principalmente em Aprendizado por Reforço (RL) (SUTTON; BARTO, 1998) e, em vez de transformar a recompensa acumulada, como é feito na abordagem de utilidade exponencial, um operador de ponto fixo é definido (MIHATSCH; NEUNEIER, 2002). Duas versões de risco dos algoritmos *Q-Learning* e *Temporal Difference* (TD) são propostas em (MIHATSCH; NEUNEIER, 2002). Além disso, os autores demonstram que esses algoritmos propostos convergem e, para isso, definem operadores diferentes.

A seguir, é descrita a abordagem de Mihatsch e Neuneier que é chamada de Processos de Decisão Markovianos Sensíveis a Risco (RS-MDP) no restante desta dissertação. Segundo Mihatsch e Neuneier (2002), essa abordagem é considerada mais simples de ser implementada do que a abordagem que usa a função utilidade exponencial. No próximo capítulo, usamos um desses operadores para formular os algoritmos de Iteração de Valor Sensível a Risco, síncrono (RSVI) e assíncrono (A-RSVI), bem como heurísticas a fim de reduzir o tempo de convergência destes algoritmos.

Formalmente um RS-MDP (MIHATSCH; NEUNEIER, 2002) é definido por uma tupla $\langle \text{MDP}, k, \gamma \rangle$, em que $-1 < k < 1$ é o fator de risco, e $\gamma \in [0, 1]$ é o fator de desconto. A abordagem proposta em (MIHATSCH; NEUNEIER, 2002) usa a função de transformação linear por partes $\mathcal{X}^k$ que depende da entrada x (a diferença temporal) ser positiva ou não e do fator de risco k . Esta função é definida por:

$$\mathcal{X}^k(x) = \begin{cases} (1 - k)x & \text{se } x > 0, \\ (1 + k)x & \text{caso contrário.} \end{cases} \quad (5)$$

A Figura 3 mostra a função de transformação $\mathcal{X}^k$ para $k = -0.99$, $k = 0$ e $k = 0.99$. Note que a forma da função para $k = 0.99$ também é côncava e para $k = -0.99$ é convexa como acontece com a função exponencial (Figura 2). Além disso, diferente da função exponencial, a função $\mathcal{X}^k$ está definida para $k = 0$ que representa a atitude neutra ao risco.

Figura 3 – Função $\mathcal{X}^k(x)$ para $k = -0,99$, $k = 0$ e $k = 0,99$.

Fonte: Baseado em Mihatsch e Neuneier (2002).

4.1 Avaliação da política

Dada uma política estacionária π , a função valor $V_k^\pi(s)$ correspondente pode ser obtida resolvendo o seguinte sistema de equações para todo $s \in S$:

$$\sum_{s' \in \mathcal{S}} \mathcal{T}(s'|s, \pi(s)) \mathcal{X}^k \left(\mathcal{R}(s, \pi(s)) + \gamma V_k^\pi(s') - V_k^\pi(s) \right) = 0. \quad (6)$$

Note que, diferente de MDPs, não é possível colocar em evidência V_k^π , pois V_k^π é parte do parâmetro da função $\mathcal{X}^k$.

Seja $x = \mathcal{R}(s, \pi(s)) + \gamma V_k^\pi(s') - V_k^\pi(s)$ a diferença temporal, se k for positivo, então diferenças temporais negativas tem peso de ponderação maior que as positivas. Em outras palavras, se: $\mathcal{R}(s, \pi(s)) + \gamma V_k^\pi(s') - V_k^\pi(s) < 0$, transições para estados sucessores em que a recompensa imediata passou a ser menor que a média recebem um peso extra maior. Enquanto, as transições para estados que prometem um retorno maior do que a média, recebem uma ponderação menor. Ou seja, se $k > 0$, a função objetivo $V_k^\pi(s)$ é aversa ao risco; e é propensa ao risco, se $k < 0$; em $k = 0$ é neutra a risco e existe a equivalência com o critério de MDPs clássicos. No limite de extrema aversão ao risco, quando $k \rightarrow 1$, a

função objetivo resolve um problema equivalente a otimização no pior dos casos. No limite de extrema propensão a risco, quando $k \rightarrow -1$, o agente é muito otimista assumindo que para todos os possíveis próximos estados, aquele que acontece é sempre o melhor.

Teorema 1 (*Solução única e casos limites*) (MIHATSCH; NEUNEIER, 2002). *Para cada $k \in (-1, 1)$ existe uma solução única $V_k^\pi(s)$ obtida pela resolução da Equação 6. Para $k=0$, $k \rightarrow 1$ e $k \rightarrow -1$, temos:*

$$V_0^\pi(s) = E \left(\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, \pi(s_t)) \middle| s_0 = s \right), \quad (7)$$

$$\lim_{k \rightarrow 1} V_k^\pi(s) = \inf \left(\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, \pi(s_t)) \middle| s_0 = s \right), \quad (8)$$

$$\lim_{k \rightarrow -1} V_k^\pi(s) = \sup \left(\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, \pi(s_t)) \middle| s_0 = s \right). \quad (9)$$

O Teorema 1 mostra que a abordagem permite interpolar entre os critérios de melhor caso, neutro e pior caso. Uma política estacionária π^* é ótima para um dado k , se $V_k^{\pi^*}(s) \geq V_k^\pi(s) \forall \pi \in \Pi, s \in \mathcal{S}$.

Figura 4 – Exemplo de um MDP com dois estados.

Fonte: Adaptado de Mihatsch e Neuneier (2002).

A Figura 4 apresenta um RS-MDP com $k = 0.99$, dois estados, duas ações no estado s_0 ($a_1 = \text{mover}$ e $a_2 = \text{ficar}$) e apenas uma ação no estado s_1 (a_3). Sendo p a probabilidade, r a recompensa associada a transição e a a ação. Note que no estado s_0 as duas ações são determinísticas e no estado s_1 a ação é probabilística. No estado s_1 , a probabilidade

de ficar no próprio estado é 0.999 com uma recompensa 1 e a probabilidade de ir para o estado s_0 é 0,001 com uma recompensa de $-\rho$, sendo $\rho \geq 0$.

É possível calcular a função valor sensível ao risco para este exemplo de RS-MDP conforme exposto em (MIHATSCH; NEUNEIER, 2002), para as políticas π_1 e π_2 (tal que $\pi_1(s_0) = \text{ficar}$ e $\pi_2(s_0) = \text{mover}$), da seguinte forma:

$$V_k^{\pi_1}(s_0) = 0,$$

$$V_k^{\pi_2}(s_0) = \frac{\gamma}{1-\gamma} \cdot \frac{(1-\theta)(1-k) - \rho\theta(1+k)}{(1-\theta)(1-k) + (1+\gamma)\theta(1+k)}.$$

Assim a política π_2 é ótima se $V_k^{\pi_2}(s_0) \geq 0$, i.e., se:

$$\rho \leq \frac{1-\theta}{\theta} \cdot \frac{1-k}{1+k}.$$

Para o exemplo, o agente escolhe a ação *mover* se $\rho \leq 5,02$, caso contrário o agente escolhe a ação *ficar* no estado s_0 a fim de evitar possíveis perdas futuras, uma vez que é extremamente averso ao risco ($k = 0.99$). Por exemplo, considerando $\rho = 4$, $\gamma = 0,99$, o valor $V_k^{\pi_1}(s_0) = 0$. Enquanto que o valor $V_k^{\pi_2}(s_0) = 14,40$. Por tanto, neste caso, o agente prefere a política π_2 .

4.2 Política ótima

Analogamente a MDPs, existem políticas ótimas determinísticas e estacionárias para RS-MDPs e a função valor ótima correspondente é única.

Teorema 2 (*Políticas ótimas*) (MIHATSCH; NEUNEIER, 2002). *Para cada $k \in (1, -1)$ existe uma única função valor ótima, $V_k^*(s) = \max_{\pi \in \Pi} V_k^\pi(s), \forall s \in S$, que satisfaz a seguinte equação de otimalidade:*

$$\max_{a \in A} \sum_{s' \in S} \mathcal{T}(s'|s, a) \mathcal{X}^k \left(\mathcal{R}(s, a) + \gamma V_k^*(s') - V_k^*(s) \right) = 0, \forall s \in S.$$

Uma política π^* é ótima se e somente se:

$$\pi^*(s) = \arg \max_{a \in A} \sum_{s' \in S} \mathcal{T}(s'|s, a) \mathcal{X}^k \left(\mathcal{R}(s, a) + \gamma V_k^*(s') - V_k^*(s) \right).$$

Além disso, também pode ser definida para RS-MDPs a função ótima $Q_k^*(s, a)$.

Teorema 3 (*Função ótima Q^**) (MIHATSCH; NEUNEIER, 2002). *A função ótima Q_k^* é a única solução da equação de otimalidade a seguir:*

$$\sum_{s' \in \mathcal{S}} \mathcal{T}(s'|s, a) \mathcal{X}^k \left(\mathcal{R}(s, a) + \gamma \max_{u \in A} Q_k^*(s', u) - Q_k^*(s, a) \right) = 0 \quad \forall s \in \mathcal{S}, u \in A.$$

Além disso, uma política π^ é ótima se e somente se:*

$$\pi^*(s) = \arg \max_{a \in A} Q_k^*(s, a). \quad (10)$$

Analogamente à teoria clássica de programação dinâmica, as provas desses teoremas dependem da propriedade de contração de alguns operadores que atuam na função valor e na função Q . A seguir é descrito um desses operadores.

4.3 Operador de ponto-fixo $\mathcal{N}_{\alpha, k}$

O operador $\mathcal{N}_{\alpha, k}$ (MIHATSCH; NEUNEIER, 2002), que será considerado neste trabalho, utiliza a função $\mathcal{X}^k$ e é definido sobre o espaço de funções $Q(s, a)$, é:

$$\mathcal{N}_{\alpha, k}[Q](s, a) := Q(s, a) + \alpha \sum_{s' \in \mathcal{S}} \mathcal{T}(s'|s, a) \mathcal{X}^k \left(\mathcal{R}(s, a) + \gamma \max_{u \in A(s)} Q(s', u) - Q(s, a) \right), \quad (11)$$

com α sendo um tamanho de passo positivo (ou seja, α não é o fator de aprendizado geralmente usado em algoritmos de aprendizado por reforço).

O operador $\mathcal{N}_{\alpha, k}$ é um mapeamento de contração relacionado com a norma máxima $|Q| := \max_{s \in \mathcal{S}, a \in A} |Q(s, a)|$, desde que α seja pequeno o suficiente, conforme especificado no Lema 1 (adaptado de (MIHATSCH; NEUNEIER, 2002)).

Lema 1 (MIHATSCH; NEUNEIER, 2002) *Seja $k \in (-1, 1)$, $0 \leq \gamma < 1$ e $0 < \alpha \leq (1 + |k|)^{-1}$ para todas as funções Q_1 e Q_2 , é verdade que:*

$$|\mathcal{N}_{\alpha, k}[Q_1] - \mathcal{N}_{\alpha, k}[Q_2]| \leq \rho |Q_1 - Q_2|,$$

em que $\rho = (1 - \alpha(1 - |k|)(1 - \gamma)) \in (0, 1)$. Assim o operador $\mathcal{N}_{\alpha, k}$ é um mapeamento de contração.

Note que, diferentemente de (MIHATSCH; NEUNEIER, 2002), o Lema 1 afirma que α pode ser igual a $(1 + |k|)^{-1}$ se γ for menor que 1.

Intuitivamente, o tamanho de passo α assegura que os valores de Q não cresçam muito, principalmente quando o fator de desconto γ está próximo de 1 e o fator de risco k é negativo.

5 Algoritmo de Iteração de Valor Sensível a Risco

O processo de atualização durante o algoritmo de Iteração de Valor para MDPs neutros a risco pode ocorrer de duas formas: síncrona ou assíncrona. Nos algoritmos síncronos todos os estados são atualizados em cada iteração e para computar o valor de V na iteração i , são utilizados os valores da iteração $i - 1$. Nos algoritmos assíncronos (i) os estados não são atualizados em um ordem predefinida e/ou (ii) apenas um subconjunto de S é atualizado em cada iteração.

Nesta seção são formulados dois algoritmos de Iteração de Valor Sensível a Risco baseados no operador $\mathcal{N}_{\alpha,k}$ (Equação 11). Os algoritmos são chamados de Iteração de Valor Sensível a Risco Síncrono (*Risk Sensitive Value Iteration* – RSVI) e Iteração de Valor Sensível a Risco Assíncrono (*Asynchronous Risk Sensitive Value Iteration* – A-RSVI). Ambos algoritmos convergem se satisfazem o Lema 1 ($0 < \alpha \leq (1 + |k|)^{-1}$).

5.1 Algoritmo de Iteração de Valor Sensível a Risco Síncrono

Dado um RS-MDP definido pela tupla $\langle \text{MDP}, k, \gamma \rangle$, no qual $-1 < k < 1$ e α satisfaz o Lema 1, considerando a Equação 11 é possível definir a função de atualização de Q a seguir, a qual é utilizada no algoritmo de Iteração de Valor Sensível a Risco Síncrono (RSVI):

$$Q^i(s, a) \leftarrow Q^{i-1}(s, a) + \alpha \sum_{s' \in \mathcal{S}} \mathcal{T}(s'|s, a) \mathcal{X}^k \left(\mathcal{R}(s, a) + \gamma \max_a Q^{i-1}(s', a) - Q^{i-1}(s, a) \right). \quad (12)$$

Nessa equação, a função escalar $\mathcal{X}^k$ é também aplicada diretamente nas diferenças temporais dos valores. Dado $Q^i(s, a)$, é possível obter a função valor na iteração i :

$$V^i(s) = \max_a \{Q^i(s, a)\},$$

e uma política gulosa:

$$\pi(s) = \arg \max_a \{Q^i(s, a)\}.$$

O critério de parada usado no algoritmo de Iteração de Valor Sensível a Risco é baseado no seguinte residual relativo:

$$\text{residual}(s) = \left| \frac{V^i(s) - V^{i-1}(s)}{V^{i-1}(s)} \right|.$$

Dado um erro mínimo desejado ϵ , se $\max_{s \in \mathcal{S}} \{\text{residual}(s)\} \leq \epsilon$, o algoritmo para. Note que a utilização do residual relativo é melhor para o algoritmo de Iteração de Valor Sensível a Risco do que o residual absoluto uma vez que o algoritmo pode ter valores muito diferentes para $V^i(s)$ e $V^{i-1}(s)$ dependendo do fator de risco e fator de desconto utilizados.

Algoritmo 2 Algoritmo de Iteração de Valor Sensível a Risco Síncrono (RSVI)

Entrada: RSMDP $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, k, \gamma)$, ϵ , α , $Q_h(s, a)$

Saída: Função valor V para o RSMDP de entrada

```

1 para cada  $s \in \mathcal{S}$  faça
2    $V^0(s) \leftarrow \max_{a \in \mathcal{A}} \{\mathcal{R}(s, a)\}$ 
3   para cada  $a \in \mathcal{A}$  faça
4      $Q^0(s, a) = Q_h(s, a)$ 
5   fim
6 fim
7  $i \leftarrow 1$ 
8 faça
9   para cada  $s \in \mathcal{S}$  faça
10    para cada  $a \in \mathcal{A}$  faça
11       $Q^i(s, a) =$ 
12         $Q^{i-1}(s, a) + \alpha \sum_{s' \in \mathcal{S}} \mathcal{T}(s'|s, a) \mathcal{X}^k[\mathcal{R}(s, a) + \gamma \max_{a \in \mathcal{A}} Q^{i-1}(s', a) - Q^{i-1}(s, a)]$ 
13    fim
14     $V^i(s) = \max_{a \in \mathcal{A}} \{Q^i(s, a)\}$ 
15     $\text{residual}(s) = \left| \frac{V^i(s) - V^{i-1}(s)}{V^{i-1}(s)} \right|$ 
16  fim
17   $i \leftarrow i + 1$ 
18 enquanto  $\max_{s \in \mathcal{S}} \{\text{residual}(s)\} > \epsilon;$ 
19 retorna  $V$ 

```

O algoritmo de Iteração de Valor Sensível a Risco Síncrono (Algoritmo 2) recebe como entrada um RSMDP, o erro mínimo desejado ϵ , o fator α e o valor inicial de $Q_h(s, a)$; e devolve como saída a função valor V ótima sensível a risco. Nas Linhas 1 a 6 são inicializados o valor V^0 com a maior recompensa para cada estado e o valor Q_0 é inicializado com $Q_h(s, a)$, por exemplo 0 para todo par estado-ação. Em cada iteração i o algoritmo atualiza os valores usando a Equação 12 (Linha 11). Na Linha 13, V^i é calculada com base em Q^i e na Linha 14 é calculado o residual relativo entre as iterações i e $i - 1$ de cada estado. Por fim, na Linha 18 o algoritmo devolve a função valor V .

5.2 Algoritmo de Iteração de Valor Sensível a Risco Assíncrono

Algumas vezes algoritmos assíncronos podem resolver problemas grandes com um menor custo computacional que algoritmos síncronos. Considerando isso, nesta seção é proposto um algoritmo de Iteração de Valor Sensível a Risco com atualização assíncrona (A-RSVI).

O algoritmo de Iteração de Valor Sensível a Risco Síncrono (Algoritmo 2) é síncrono pois todos os pares estado-ação são atualizados em cada iteração e os valores $Q(s, a)$ na iteração i dependem somente de $Q(s, a)$ da iteração $i - 1$. O algoritmo de Iteração de Valor Sensível a Risco Assíncrono (Algoritmo 3) permite calcular $Q(s, a)$ na iteração i usando os valores $Q(s', a)$ já calculados na própria iteração i , isto é usando os valores dos estados sucessores já calculados. Note que a única diferença entre o Algoritmo 2 e o Algoritmo 3 é a Linha 11.

Algoritmo 3 Algoritmo de Iteração de Valor Sensível a Risco Assíncrono (A-RSVI)

Entrada: RSMDP $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, k, \gamma), \epsilon, \alpha, Q_h(s, a)$

Saída: Função valor V para o RSMDP de entrada

```

1 para cada  $s \in \mathcal{S}$  faça
2    $V^0(s) \leftarrow \max_{a \in \mathcal{A}} \{\mathcal{R}(s, a)\}$ 
3   para cada  $a \in \mathcal{A}$  faça
4      $Q^0(s, a) = Q_h(s, a)$ 
5   fim
6 fim
7  $i \leftarrow 1$ 
8 faça
9   para cada  $s \in \mathcal{S}$  faça
10    para cada  $a \in \mathcal{A}$  faça
11       $Q(s, a) \leftarrow$ 
12         $Q(s, a) + \alpha \sum_{s' \in \mathcal{S}} \mathcal{T}(s'|s, a) \mathcal{X}^k[\mathcal{R}(s, a) + \gamma \max_{a \in \mathcal{A}} Q(s', a) - Q(s, a)]$ 
13    fim
14     $V^i(s) = \max_{a \in \mathcal{A}} \{Q^i(s, a)\}$ 
15     $\text{residual}(s) = \left| \frac{V^i(s) - V^{i-1}(s)}{V^{i-1}(s)} \right|$ 
16  fim
17   $i \leftarrow i + 1$ 
18 enquanto  $\max_{s \in \mathcal{S}} \{\text{residual}(s)\} > \epsilon;$ 
19 retorna  $V$ 

```

Por exemplo, suponha um problema com cinco estados s_1, s_2, s_3, s_4, s_5 , em que as atualizações são feitas nessa ordem. Suponha também que os sucessores do estado s_4 aplicando a ação a são s_2 e s_3 . Quando o Algoritmo 3 está calculando o valor $Q(s_4, a)$ na

iteração i , o algoritmo assíncrono já usará os valores de $Q(s_2, a)$ e $Q(s_3, a)$ calculados na própria iteração i .

Atualizar os valores dos estados de forma assíncrona durante as iterações pode trazer mais agilidade para a convergência do algoritmo de Iteração de Valor Sensível a Risco. Ambos algoritmos de Iteração de Valor Sensível a Risco propostos neste capítulo podem ser utilizados com as heurísticas que serão apresentadas no próximo capítulo a fim de melhorar ainda mais o seu desempenho.

6 Heurísticas para o algoritmo de Iteração de Valor Sensível a Risco

Foi realizada uma análise prévia sobre o impacto do tamanho do passo α e o fator de risco no tempo de convergência dos algoritmos de Iteração de Valor Sensível a Risco. A partir desta análise, descobriu-se que:

- (i) o custo de processamento de políticas extremas ao risco é alto;
- (ii) valores intermeditários de fatores de risco tem baixo custo computacional; e
- (iii) o uso de α próximo de 1 tem menor custo computacional do que o uso de outros valores.

Desta forma, são propostas duas heurísticas para os algoritmos de Iteração de Valor Sensível ao Risco, a fim de torná-los mais eficientes para valores extremos de risco. Uma heurística é baseada no tamanho do passo α e outra baseada no fator de risco k . Ambas as heurísticas são usadas para inicializar o valor de $Q_h(s, a)$ em ambos os algoritmos (Figura 5).

Figura 5 – Algoritmo de Iteração de Valor Sensível a Risco com heurística para inicializar os valores de $Q_h(s, a)$.

Fonte: Igor Oliveira Borges, 2018.

6.1 Heurística baseada no tamanho do passo α

A heurística baseada no tamanho do passo α chamada de H_α (Algoritmo 4), executa o algoritmo de Iteração de Valor Sensível a Risco com valor de α_{inicial} maior do que α a

fim de que o algoritmo possa ser executado mais rapidamente e obter um bom valor inicial para $Q(s, a)$. O valor de α_{inicial} pode não satisfazer o Lema 1; por exemplo, pode ser usado $\alpha_{\text{inicial}} = 1$. Note que com esse valor de α_{inicial} a convergência com k diferente de zero não é garantida.

Algoritmo 4 Heurística α

Entrada: RSMDP $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, k, \gamma)$, α_{inicial}

Saída: Função Q_h

```

1 para cada  $s \in \mathcal{S}$  faça
2    $V^0(s) \leftarrow \max_{a \in \mathcal{A}} \{\mathcal{R}(s, a)\}$ 
3   para cada  $a \in \mathcal{A}$  faça
4      $Q_h^0(s, a) = 0$ 
5   fim
6    $\text{residual}^0(s) = \infty$ 
7 fim
8  $i \leftarrow 0$ 
9 faça
10   $i \leftarrow i + 1$ 
11  para cada  $s \in \mathcal{S}$  faça
12    para cada  $a \in \mathcal{A}$  faça
13       $Q_h^i(s, a) =$ 
14         $Q_h^{i-1}(s, a) + \alpha_{\text{inicial}} \sum_{s' \in \mathcal{S}} \mathcal{T}(s'|s, a) \mathcal{X}^k[\mathcal{R}(s, a) + \gamma \max_{a \in \mathcal{A}} Q_h^{i-1}(s', a) -$ 
15         $Q_h^{i-1}(s, a)]$ 
16    fim
17     $V^i(s) = \max_{a \in \mathcal{A}} \{Q_h^i(s, a)\}$ 
18     $\text{residual}^i(s) = \left| \frac{V^i(s) - V^{i-1}(s)}{V^{i-1}(s)} \right|$ 
19  fim
20 enquanto  $\max_{s \in \mathcal{S}} \{\text{residual}^i(s)\} < \max_{s \in \mathcal{S}} \{\text{residual}^{i-1}(s)\};$ 
21 retorna  $Q_h$ 

```

O algoritmo 4 recebe como entrada um RSMDP e o tamanho do passo α_{inicial} ; e retorna um bom valor inicial para a função $Q(s, a)$. Este algoritmo realiza atualizações enquanto o máximo residual até a iteração i for menor que o máximo residual até a iteração $i - 1$ (Algoritmo 4, Linhas 9 – 18). Em outras palavras, atualizações são realizadas até o momento que o máximo residual começa a aumentar, ao invés de diminuir, entre uma iteração e outra. Isto acontece porque é usado um valor α_{inicial} que não satisfaz o Lema 1. Por fim, na Linha 19 o algoritmo retorna a função Q_h . Note que as principais diferenças entre o Algoritmo 2 e Algoritmo 4 são (i) o uso de α_{inicial} ; (ii) a condição do laço (Linha 18); e (iii) o valor retornado (Linha 19).

6.2 Heurística baseada no fator de risco k

A heurística baseada no fator de risco, chamada H_k executa o algoritmo de Iteração de Valor Sensível a Risco considerando um fator de risco k_{inicial} que possui um custo computacional menor, i.e., um fator de risco mais distante dos valores extremos do fator de risco -1 e 1 . Nesta heurística o algoritmo executa atualizações em $Q_h(s, a)$ com k_{inicial} (em que $|k_{\text{inicial}}| < |k|$ e k_{inicial} está mais longe dos extremos do fator de risco -1 e 1) até convergir considerando $\epsilon_{\text{inicial}}$ maior do que o erro final desejado ϵ (Linhas 8 – 17), isto porque o objetivo é obter bons valores iniciais para a função Q . Por exemplo, dado $k = 0.99$ e $\epsilon = 0.00001$, pode-se escolher $k_{\text{inicial}} = 0$ e $\epsilon_{\text{inicial}} = 0.001$.

Assim, para calcular a heurística k , precisamos apenas modificar o algoritmo de Iteração de Valor Sensível a Risco, tanto síncrono (Algoritmo 2) quanto assíncrono (Algoritmo 3), para retornar $Q(s, a)$ em vez de $V(s)$ e chamá-lo passando o parâmetro k_{inicial} , $\epsilon_{\text{inicial}}$, o tamanho do passo α tal que $0 < \alpha \leq (1 + |k|)^{-1}$ (conforme com o Lema 1), e o valor inicial de Q igual a 0, por exemplo:

$$Q_h = \text{RSVI}(\text{RSMDP}(\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, k_{\text{inicial}}, \gamma), \epsilon_{\text{inicial}}, \alpha, 0).$$

A função Q_h , com os valores obtidos pela execução do próprio algoritmo de Iteração de Valor Sensível a Risco com esses parâmetros, fornece uma boa heurística inicial para ser usada a fim de acelerar a convergência em valores de k extremos que são custosos computacionalmente.

7 Experimentos

Nesta seção analisamos o algoritmo de Iteração de Valor Sensível a Risco e algumas combinações das variantes propostas no domínio de Travessia do Rio.

O código-fonte¹ que inclui a modelagem do domínio do rio, o algoritmo de iteração de valor sensível a risco e a interface gráfica foi desenvolvido em Octave/Matlab. Os experimentos foram realizados no Matlab R2015a, na versão acadêmica de 64 bits no Windows 10, em um processador Intel Core i7-3537U CPU @3.10 GHz, 8 GB de memória RAM @1600 MHz, 256 GB de armazenamento SSD. A interface gráfica foi desabilitada para o experimento.

7.1 Domínio de Travessia do Rio

O problema da Travessia do Rio (FREIRE; DELGADO, 2017) é representado como um *grid* de tamanho $N_x \times N_y$. Um exemplo pode ser visualizado na Figura 6. Nesta tabela está em azul claro, o rio; em azul escuro, a cachoeira; em vermelho, a ponte; em verde, o solo; em cinza, o agente propriamente dito no estado inicial; e por fim a meta é destacada em amarelo.

O domínio consiste em apenas um agente no extremo do canto inferior esquerdo da matriz, o qual tem como objetivo chegar ao canto inferior direito do *grid*, sendo que apenas 4 ações são possíveis: ir para o norte ($\uparrow$), sul ($\downarrow$), oeste ($\leftarrow$) e leste ($\rightarrow$). Assim o agente pode chegar no objetivo de duas formas:

- (i) nadando a partir de qualquer ponto do rio; ou
- (ii) subindo o *grid* até a ponte que está na posição extrema superior.

Figura 6 – Instância do domínio de travessia do rio.

Fonte: Igor Oliveira Borges, 2018.

¹ Disponível em: <https://github.com/RSVI>

7.2 Configurações do experimento

A probabilidade de transição fora do rio é de 99% da ação ter o efeito desejado e 1% de ficar parado. No rio, o agente tem 80% de chance de ser arrastado pela correnteza e 20% de sucesso na execução da ação escolhida. A ação sul dentro do rio, também tem 80% de ser arrastado pela correnteza, porém tem 20% de ficar parado no mesmo estado. A probabilidade de ir para o estado inicial dado que o agente está na cachoeira é 1, isto é, sempre que o agente cai na cachoeira ele retorna para o estado inicial.

Os experimentos foram realizados em dois cenários de recompensa distintos, o primeiro com recompensa 0 em cada estado e +1 no estado meta, o qual chamaremos de cenário de recompensas acumuladas positivas (+); e o segundo com recompensa negativa -1 em cada estado e 0 na meta, o qual chamaremos de cenário de recompensas acumuladas negativas (-).

Nos experimentos, o fator α baseado no valor de k (como mostra a Tabela 1) é chamado de α relativo. Além disso, o fator de risco k é chamado de estático quando ele não muda durante o algoritmo.

Tabela 1 – Valores de α baseados no valor de k .

k	α
-0.99 e 0.99	0.50
-0.8 e 0.8	0.56
-0.5 e 0.5	0.67
0	1

Para avaliar os algoritmos RSVI e A-RSVI com diferentes parâmetros e heurísticas, as configurações são nomeadas da seguinte forma:

- RSVI- $S\alpha$: algoritmo RSVI com $\epsilon = 0.00001$ e tamanho de passo estático ($\alpha = 0.5$);
- RSVI- $R\alpha$: algoritmo RSVI com $\epsilon = 0.00001$ e tamanho de passo relativo (Tabela 1);
- RSVI- $R\alpha H\alpha$: algoritmo RSVI com $\epsilon = 0.00001$, tamanho de passo relativo (Tabela 1) e heurística H_α (usando $\alpha_{\text{inicial}}=1$);

- RSVI- $R\alpha Hk$: algoritmo RSVI com $\epsilon = 0.00001$, tamanho de passo relativo (Tabela 1) e heurística H_k (usando $k_{\text{inicial}} = 0$, $\epsilon_{\text{inicial}} = 0.001$, e também usando tamanho de passo relativo);
- A-RSVI- $R\alpha Hk$: algoritmo A-RSVI com $\epsilon = 0.00001$, tamanho de passo relativo (Tabela 1), e heurística H_k (usando $k_{\text{inicial}} = 0$, $\epsilon_{\text{inicial}} = 0.001$, e também usando tamanho de passo relativo).

Para o experimento adotou-se:

- fator de desconto $\gamma \in \{0.6; 0.7; 0.8; 0.9; 0.99\}$; e
- fator de risco $k \in \{-0.99; -0.8; -0.5; 0; 0.5; 0.8; 0.99\}$;

7.3 Políticas Obtidas

Foram realizados experimentos com diferentes tamanhos de *grid*, entre eles 3×7 , 5×7 , 7×7 e 10×7 . Nesta seção primeiro são avaliados os resultados obtidos para o *grid* 10×7 pois ele apresenta uma variedade maior de políticas. A Tabela 2 mostra as políticas obtidas para esse *grid*, variando k e γ no algoritmo de Iteração de Valor Sensível a Risco nos cenários de recompensa (+) e recompensa (-), parte superior e inferior da tabela, respectivamente. Em ambas as configurações (RSVI- $S\alpha$ e RSVI- $R\alpha$) foram obtidas as mesmas políticas relativas ao cenário de recompensa adotado, isto ocorre pois para ambos os valores de α utilizados, é garantida a convergência pelo Lema 1.

Em ambos cenários se observa um comportamento de risco esperado para os parâmetros de propensão, neutralidade e aversão ao risco, sendo que para $k = -0.8$, $k = -0.5$, $k = 0$, $k = 0.5$ e $k = 0.8$ as políticas encontradas variando γ são as mesmas. A diferença entre ambos cenários acontece nos extremos, com $k = -0.99$ e $k = 0.99$.

Para $k = -0.99$, no cenário (-) há algumas políticas com atitudes mais propensas ao risco que não aparecem no cenário de recompensa (+), em especial saltar da ponte a fim de chegar mais rápido na meta (veja no cenário de recompensa acumulada negativa $k = -0.99$ e $\gamma \geq 0.7$). Para $k = 0.99$ aparecem duas políticas diferentes no cenário (-) e no cenário positivo três políticas diferentes. Nessas políticas o agente tenta atravessar pela ponte e caso caia no rio tenta voltar a borda mais perto ou se está perto da ponte tenta subir para ter uma travessia mais segura evitando cair na cachoeira.

Reforço do fator de risco k [illegible]

Propensão Aversão

Fonte: Igor Oliveira Borges, 2018.

Em valores de $k \geq -0.5$, observasse políticas mais conservadoras que tentam atravessar pela ponte e evitam entrar no rio para assegurar uma transição mais segura. Ao tomar uma política mais propensa ($k = -0.99$ e $k = -0.8$), o agente tende a se arrisca mais atravessando pelo rio e saltando da ponte em direção a meta.

Nota-se ainda que nos extremos de valor de risco $k = -0.99$ para propensão e $k = 0.99$ para aversão obtém-se políticas mais reforçadas para o respectivo tipo de atitude esperado, em especial quando o γ é elevado ($\gamma \geq 0.9$). Note que o fator de desconto tem o papel de atenuar as atitudes se o desconto for pequeno ou realçar as atitudes se o desconto for grande. O parâmetro de risco por sua vez infere na política obtida um comportamento de propensão, neutralidade ou aversão conforme o valor do parâmetro escolhido.

A Tabela 3 mostra as políticas obtidas para o *grid* 3×7 , variando k e γ no algoritmo de Iteração de Valor Sensível a Risco nos cenários de recompensa (+) e recompensa (-), parte esquerda e direita da tabela, respectivamente. Nesta instância com menos estados observa-se também um comportamento de risco esperado. Porém, essa instância apresenta uma variedade menor de políticas que o *grid* 10×7 .

7.4 Tempo de processamento

As Figuras 7 e 8 apresentam o tempo de processamento para o cenário de recompensas (+) e (-) no *grid* 10×7 . As políticas obtidas nas cinco configurações para cada cenário de recompensa adotado foram iguais entre si. Porém, o tempo de processamento necessário para a convergência foi diferente.

Para o cenário (+), a configuração RSVI- $R\alpha$ é mais eficiente que a RSVI- $S\alpha$ (especialmente para $k = 0$) e também é mais eficiente que usar RSVI- $R\alpha H\alpha$ (especialmente para $k = -0.99$). Para o cenário (-), a configuração de RSVI- $R\alpha$ é também mais eficiente do que a RSVI- $S\alpha$, mas RSVI- $R\alpha H\alpha$ é ligeiramente melhor para $\gamma \leq 0.8$.

O problema da heurística α , considerando $\alpha_{\text{inicial}} = 1$, no cenário (+) consiste em não se saber ao certo em qual momento os valores de $Q(s, a)$ já estão muito distantes do valor ótimo.

Para o cenário (+), as configurações RSVI- $R\alpha$ e RSVI- $R\alpha Hk$ têm um comportamento similar. Enquanto que para o cenário (-), as configurações RSVI- $R\alpha Hk$ e A-RSVI- $R\alpha Hk$ são os melhores para $k \leq -0.5$.

Tabela 3 – Políticas obtidas pelo algoritmo de Iteração de Valor Sensível a Risco em um *grid* 3×7 conforme cenário de recompensa adotado.

3x7 (+)	k=-0.99	k=-0.8	k=-0.5	k=0	k=0.5	k=0.8	k=0.99	3x7 (-)	k=-0.99	k=-0.8	k=-0.5	k=0	k=0.5	k=0.8	k=0.99
$\gamma=0.6$	↓↓↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	$\gamma=0.6$	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o
$\gamma=0.7$	↓↓↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	$\gamma=0.7$	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o
$\gamma=0.8$	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	$\gamma=0.8$	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o
$\gamma=0.9$	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	$\gamma=0.9$	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o
$\gamma=0.99$	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	$\gamma=0.99$	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ →↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o	→↓ ↑↓ →↓ →↓ →↓ →↓ ↑+o

Fonte: Igor Oliveira Borges, 2018.

Tanto no cenário (+) quanto no cenário (−), as configurações RSVI- $R\alpha Hk$ e A-RSVI- $R\alpha Hk$ têm um comportamento similar.

No intervalo de risco $-0.5 \leq k \leq 0.5$, em ambos cenários e para todas as configurações, notou-se uma execução rápida, independente do fator de desconto adotado. Para valores extremos de k ($k = -0.99$ e $k = 0.99$) têm-se uma maior demanda de tempo para processamento quando comparado com outros valores de k .

O fator de desconto por sua vez tem o papel de garantir que a convergência ocorra e tem influência no tempo necessário para convergência, porém de forma mais tênue, se

Figura 7 – Tempo de convergência por configuração do algoritmo de Iteração de Valor Sensível a Risco na instância do rio de tamanho 10×7 no cenário de recompensas (+) para os fatores de risco k ($k = -0.99, k = -0.8, k = -0.5, k = 0, k = 0.5, k = 0.8$ e $k = 0.99$) e fator de desconto γ ($\gamma = 0.6, \gamma = 0.7, \gamma = 0.8, \gamma = 0.9$ e $\gamma = 0.99$).

Fonte: Igor Oliveira Borges, 2018.

comparado ao impacto do fator de risco no tempo. Quanto maior o desconto, maior o tempo necessário para convergência.

Na Figura 9 é mostrado o tempo para diferentes tamanhos de *grid* considerando $\gamma = 0.99$ nos cenários de recompensas (+) e (-) para dois valores de k extremos ($k = -0.99$ no lado esquerdo da figura e $k = 0.99$ no lado direito da figura). Nessa figura são mostrados os resultados para as três melhores configurações da seção anterior: (i) RSVI- R_α (parte superior); (ii) RSVI- $R_\alpha Hk$ (parte central); e (iii) A-RSVI- $R_\alpha Hk$ (parte inferior). Como

Figura 8 – Tempo de convergência por configuração do algoritmo de Iteração de Valor Sensível a Risco na instância do rio de tamanho 10×7 no cenário de recompensas $(-)$ para os fatores de risco k ($k = -0.99, k = -0.8, k = -0.5, k = 0, k = 0.5, k = 0.8$ e $k = 0.99$) e fator de desconto γ ($\gamma = 0.6, \gamma = 0.7, \gamma = 0.8, \gamma = 0.9$ e $\gamma = 0.99$).

Fonte: Igor Oliveira Borges, 2018.

esperado, o tempo de processamento depende do tamanho do problema, quanto maior a quantidade de estados, maior o tempo necessário para convergência.

Note que para a configuração RSVI- R_α para $k = -0.99$, a adoção de um cenário $(+)$ pode reduzir o tempo de convergência do algoritmo quando comparado com o cenário $(-)$, o contrário ocorre para $k = 0.99$.

Figura 9 – Tempo de processamento nas configurações RSVI- $R\alpha$, RSVI- $R\alpha Hk$ e A-RSVI- $R\alpha Hk$ por tamanho de *grid* e cenário de recompensas (–) com o fator de desconto $\gamma = 0.99$, para dois valores extremos do fator de risco k ($k = -0.99$ e $k = 0.99$).

Fonte: Igor Oliveira Borges, 2018.

Para o cenário (–) e para $k = -0.99$, RSVI- $R\alpha Hk$ obteve uma expressiva redução de tempo para convergência quando comparado com a RSVI- $R\alpha$. Já para o cenário (+) essa configuração obteve um desempenho similar.

Para o cenário (–) e para $k = 0.99$, RSVI- $R\alpha Hk$ obteve um desempenho similar quando comparado com RSVI- $R\alpha$. Já para o cenário (+) e $k = 0.99$, RSVI- $R\alpha Hk$ foi um pouco melhor em termos de tempo.

Para o cenário (+), tanto para $k = -0.99$ quanto para $k = 0.99$, A-RSVI- $R\alpha Hk$ foi melhor para o *grid* 10×7 e similar para os outros tamanhos de *grid*.

Podemos concluir que para esse domínio, *RSVI- $R\alpha Hk$* pode reduzir o tempo de convergência para os extremos de risco independente do tamanho do problema, embora vale ressaltar que tal redução pode ser maior caso seja escolhido um cenário de recompensas que permita uma execução mais eficiente para o risco desejado. A atualização assíncrona por sua vez torna essa configuração ainda mais eficiente quando usada com a heurística k , e sua vantagem é percebida conforme o tamanho do problema aumenta.

7.5 Valor do estado inicial e valor do residual

As Figuras 10 e 11 apresentam a convergência do valor de estado inicial e o valor do residual encontrados durante as iterações do algoritmo de Iteração de Valor Sensível a Risco para três configurações: (i) *RSVI- $R\alpha$* (parte superior); (ii) *RSVI- $R\alpha Hk$* (parte central); e (iii) *A-RSVI- $R\alpha Hk$* (parte inferior). Escolheu-se para isto o cenário $(-)$ e dois valores extremos do fator de risco k ($k = -0.99$ e $k = 0.99$).

Percebe-se que o fator de desconto tem impacto no número de iterações, quanto maior o fator de desconto maior o número de iterações. Além disso, para $k = -0.99$, *RSVI- $R\alpha Hk$* obteve uma expressiva redução no número de iterações quando comparado com *RSVI- $R\alpha$* . Já para $k = 0.99$ não houve ganhos nem prejuízos.

Para o cenário $(-)$, as curvas do valor do estado inicial nas configurações *RSVI- $R\alpha Hk$* e *A-RSVI- $R\alpha Hk$* têm um comportamento diferente quando comparadas às curvas das outras configurações: elas decrescem enquanto o algoritmo trabalha com o valor de $k_{inicial} = 0$ e depois crescem quando o algoritmo *RSVI* é executado com k . As curvas do valor residual desses duas configurações têm também um comportamento diferente quando comparadas às curvas das outras configurações. Note ainda que em todas as configurações o valor final do estado inicial é o mesmo.

7.6 Considerações finais

Nesta seção, foi analisado de forma empírica o impacto da escolha do fator de risco e o fator de desconto na política e no tempo de convergência dos algoritmos de Iteração de Valor formulados. Os algoritmos de Iteração de Valor Sensíveis a Risco conseguem encontrar políticas sensíveis ao risco conforme o fator de risco escolhido, tanto para aversão ($k > 0$) quanto propensão ($k < 0$) ao risco. Além disso, a característica de neutralidade é

Figura 10 – Valor do estado inicial durante as iterações nas configurações RSVI- R_α , RSVI- $R_\alpha Hk$ e A-RSVI- $R_\alpha Hk$ no cenário de recompensas (–) para a instância do rio de tamanho 10×7 , para dois valores extremos do fator de risco k ($k = -0.99$ e $k = 0.99$) e fator de desconto γ ($\gamma = 0.6, \gamma = 0.7, \gamma = 0.8, \gamma = 0.9$ e $\gamma = 0.99$).

Fonte: Igor Oliveira Borges, 2018.

mantida quando $k = 0$ sendo equivalente ao algoritmo de Iteração de Valor clássico. Nesta abordagem é necessário a escolha de um valor de risco k arbitrário coerente ao domínio de aplicação, para garantir que ocorra a ponderação esperada que permita a construção de uma política π com a sensibilidade ao risco desejada.

Foram realizados experimentos no domínio de Travessia do Rio com diferentes tamanhos de *grid* em dois cenários de recompensas distintos: recompensas positivas acumuladas e recompensas negativas acumuladas. O fator de desconto tem o papel de

Figura 11 – Valor do residual relativo durante as iterações nas configurações RSVI- R_α , RSVI- $R_\alpha Hk$ e A-RSVI- $R_\alpha Hk$ no cenário de recompensas (–) para a instância do rio de tamanho 10×7 , para dois valores extremos do fator de risco k ($k = -0.99$ e $k = 0.99$) e fator de desconto γ ($\gamma = 0.6, \gamma = 0.7, \gamma = 0.8, \gamma = 0.9$ e $\gamma = 0.99$).

Fonte: Igor Oliveira Borges, 2018.

atenuar as atitudes se o desconto for pequeno ou realçar as atitudes se o desconto for grande. No limite de k ($k = -0.99$ para propensão extrema ao risco e $k = 0.99$ para aversão extrema ao risco), embora os algoritmos encontrem políticas com as respectivas atitudes de risco esperadas, os algoritmos demoram mais tempo para convergir.

Os experimentos mostram que o tempo de processamento está diretamente relacionado ao fator de risco escolhido, o fator de desconto γ , o fator α e o tamanho da instância. Enquanto o fator de desconto tem uma influência fraca no tempo de convergência, o uso de

um α dependente de k traz bastante ganho no desempenho do algoritmo reduzindo o tempo necessário para o algoritmo convergir. Além disso, o uso da configuração A-RSVI- $R\alpha Hk$ pode reduzir o número de iterações necessárias e o tempo para o algoritmo convergir especialmente para $k = -0.99$.

8 Conclusão

Nesta dissertação, são formulados e analisados dois novos algoritmos baseados em iteração de valor para MDPs sensíveis a risco, o RSVI e o A-RSVI, no qual o operador de atualização é baseado no operador $\mathcal{N}_{\alpha,k}$ proposto em (MIHATSCH; NEUNEIER, 2002). Na versão assíncrona, os valores entre estados podem ser calculados com base nos valores obtidos na iteração mais recente. Além disso, propomos duas heurísticas para melhorar seu desempenho, especialmente para valores extremos do fator de risco (extremo averso ou extremo propenso). Estas duas heurísticas, inicializam os valores de Q executando uma versão ligeiramente modificada do algoritmo RSVI proposto, mas com um fator α (heurística H_α) e k (heurística H_k) que tenham um baixo custo computacional.

Foram realizados experimentos no domínio de travessia do rio em dois cenários de recompensas (+) e (−) e foi feita uma análise da influência do fator de desconto e do fator de risco na política estacionária obtida e no tempo de processamento necessário para a convergência nesses cenários. Os algoritmos de Iteração do Valor Sensível ao Risco podem encontrar políticas sensíveis ao risco de acordo com o fator de risco escolhido, tanto para aversão ($k > 0$) quanto para propensão ($k < 0$) ao risco. Além disso, a característica de neutralidade é mantida quando $k = 0$, sendo equivalente ao algoritmo clássico de Iteração de Valor.

Os resultados dos experimentos mostram que:

- (i) o custo de processamento de políticas extremas a risco, tanto de aversão quanto de propensão é elevado;
- (ii) um desconto elevado aumenta o tempo de convergência do algoritmo e reforça a sensibilidade ao risco adotada, embora tenha uma influência fraca no tempo de convergência se comparada a influência do fator de risco k ;
- (iii) políticas com valores para o fator de risco intermediários possuem custo computacional baixo e já possuem certa sensibilidade ao risco dependendo do fator de desconto utilizado; e
- (iv) o algoritmo A-RSVI com a heurística H_k (baseada no fator de risco) pode reduzir o número de iterações e o tempo de convergência, especialmente para $k = -0.99$.

8.1 *Resumo das contribuições*

Entre as principais contribuições do trabalho estão:

- Formulação de dois algoritmos Iteração de Valor Sensível a risco, síncrono (RSVI) e assíncrono (A-RSVI) baseado no operador proposto por Mihatsch e Neuneier (2002) e duas heurísticas (baseada em α e k).
- Análise da influência do fator de desconto γ , fator de risco k e tamanho do passo α na política obtida e no tempo de convergência dos algoritmos de Iteração de Valor Sensíveis a Risco.
- Análise do desempenho e convergência de diferentes configurações dos algoritmos de Iteração de Valor Sensíveis a Risco formulados junto com as heurísticas propostas.

8.2 *Publicações geradas*

No período do mestrado houve três publicações:

1. Em (BORGES; DELGADO, 2017) foi definido o plano de pesquisa inicial que consistia em propor um algoritmo de aprendizado por reforço sensível a risco. Esse trabalho foi publicado no X Workshop de Teses e Dissertações do Simpósio Brasileiro de Sistemas de Informação.
2. Em (BORGES; DELGADO; FREIRE, 2017) foi proposto um novo simulador do jogo de futebol visando fornecer um ambiente de mapeamento discreto, de fácil utilização. O simulador permite a configuração de diversas variáveis de ambiente que possibilitam a simulação das características de um jogo real. Ainda foram implementadas estratégias para os agentes no simulador, baseada em regras estáticas com busca local A*, aleatória ou com as técnicas de Aprendizado por Reforço, *Q-Learning* e SARSA. Esse trabalho foi publicado no XIV Encontro Nacional de Inteligência Artificial.
3. Em (BORGES; DELGADO; FREIRE, 2018) foi feita a análise da influência do fator de desconto e de risco no algoritmo de Iteração de Valor Sensível a Risco formulado. Esse trabalho foi aceito para publicação no XV Encontro Nacional de Inteligência Artificial.

8.3 *Trabalhos futuros*

Em trabalhos futuros pretende-se:

- Aplicar os algoritmos de Iteração de Valor Sensíveis a Risco e heurísticas propostas em problemas mais realísticos que possuam um conjunto de estados e ações maiores, a fim de avaliar a escalabilidade do algoritmo em problemas mais complexos e reais de tomada de decisão.
- Aplicar as heurísticas propostas em outras abordagens sensíveis a risco, por exemplo na abordagem que usa função utilidade exponencial.
- Elaborar métricas capazes de avaliar políticas sensíveis a risco de forma menos dependente do domínio de aplicação.

Referências¹

- BELLMAN, R. A Markovian decision process. *Indiana University Mathematics Journal*, v. 6, p. 679–684, 1957. Citado na página 22.
- BELLMAN, R.; KALABA, R. *Dynamic programming and modern control theory*. New York, USA: Academic Press, 1965. 112 p. Citado na página 24.
- BORGES, I. O.; DELGADO, K. V. Algoritmo de aprendizado por reforço para resolução de processos de decisão markovianos sensíveis ao risco. In: *Anais do X Workshop de Teses e Dissertações em Sistemas de Informação (WTDSI) do Simpósio Brasileiro de Sistemas de Informação*. Lavras, Minas Gerais, Brasil: Sociedade Brasileira de Computação, 2017. p. 13–16. Citado na página 56.
- BORGES, I. O.; DELGADO, K. V.; FREIRE, V. D2DSS : Simulador discreto 2D do jogo de futebol. In: *Anais do XIV Encontro Nacional de Inteligência Artificial e Computacional (ENIAC)*. Uberlândia, Minas Gerais, Brasil: Sociedade Brasileira de Computação, 2017. p. 972–983. Citado na página 56.
- BORGES, I. O.; DELGADO, K. V.; FREIRE, V. Análise do algoritmo de iteração de valor sensível a risco. In: *Anais do XV Encontro Nacional de Inteligência Artificial e Computacional (ENIAC)*. São Paulo, Brasil: Sociedade Brasileira de Computação, 2018. p. 365–376. Citado na página 56.
- CASTRO, D. D.; TAMAR, A.; MANNOR, S. Policy gradients with variance related risk criteria. In: *Proceedings of the 29th International Conference on Machine Learning*. Edinburgh, Scotland, UK: International Machine Learning Society, 2012. p. 387–396. Citado 2 vezes nas páginas 26 e 27.
- CHUNG, K.-J.; SOBEL, M. J. Discounted MDP's: distribution functions and exponential utility maximization. *Journal on Control and Optimization*, Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania, USA, v. 25, p. 49–62, 1987. Citado 2 vezes nas páginas 19 e 27.
- DENARDO, E. V.; ROTHBLUM, U. G. Optimal stopping, exponential utility, and linear programming. *Mathematical Programming*, v. 16, n. 1, p. 228–244, 1979. Citado 3 vezes nas páginas 19, 26 e 27.
- FILAR, J. A.; KALLENBERG, L. C. M.; LEE, H.-M. Variance-penalized Markov decision processes. *Mathematics of Operations Research*, v. 14, n. 1, p. 147–161, 1989. Citado 2 vezes nas páginas 19 e 26.
- FILAR, J. A.; KRASS, D.; ROSS, K. W.; ROSS, K. W. Percentile performance criteria for limiting average Markov decision processes. *IEEE Transactions on Automatic Control*, v. 40, n. 1, p. 2–10, 1995. Citado 2 vezes nas páginas 19 e 26.
- FREIRE, V. The role of discount factor in Risk Sensitive Markov Decision Processes. In: *Proceedings of the 5th Brazilian Conference on Intelligent Systems (BRACIS)*. Recife, Pernambuco, Brasil: IEEE Computer Society, 2016. p. 480–485. Citado na página 19.

¹ De acordo com a Associação Brasileira de Normas Técnicas. NBR 6023.

- FREIRE, V.; DELGADO, K. V. GUBS: a utility-based semantic for Goal-Directed Markov Decision Processes. In: *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems*. São Paulo, Brasil: International Foundation for Autonomous Agents and Multiagent Systems, 2017. p. 741–749. Citado na página 42.
- GARCÍA, J.; FERNÁNDEZ, F. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, Microtome Publishing, v. 16, n. 1, p. 1437–1480, 2015. Citado 2 vezes nas páginas 19 e 26.
- GASKETT, C. Reinforcement learning under circumstances beyond its control. In: *Proceedings of the International Conference on Computational Intelligence, Robotics and Autonomous Systems*. Vienna, Austria: Computational Intelligence for Modelling Control and Automation, 2003. p. 1–12. Citado na página 26.
- GOSAVI, A. Reinforcement learning: A tutorial survey and recent advances. *INFORMS Journal on Computing*, INFORMS, Linthicum, Maryland, USA, v. 21, n. 2, p. 178–192, 2009. Citado na página 27.
- HEGER, M. Consideration of risk in reinforcement learning. In: *Proceedings of the 11th International Machine Learning Conference*. San Francisco, California, USA: Elsevier, 1994. p. 105–111. Citado na página 26.
- HOU, P.; YEOH, W.; VARAKANTHAM, P. Revisiting risk-sensitive MDPs: New algorithms and results. In: *Proceedings of the 24th International Conference on Automated Planning and Scheduling (ICAPS)*. Portsmouth, New Hampshire, USA: Association for the Advancement of Artificial Intelligence, 2014. Citado 2 vezes nas páginas 19 e 26.
- HOU, P.; YEOH, W.; VARAKANTHAM, P. Solving risk-sensitive POMDPs with and without cost observations. In: *Proceedings of the 30th AAAI Conference on Artificial Intelligence*. Phoenix, Arizona, USA: Association for the Advancement of Artificial Intelligence, 2016. p. 3138–3144. Citado 2 vezes nas páginas 19 e 26.
- HOWARD, R. A.; MATHESON, J. E. Risk-sensitive Markov decision processes. *Management science*, INFORMS, v. 18, n. 7, p. 356–369, 1972. Citado 4 vezes nas páginas 19, 26, 27 e 28.
- JAQUETTE, S. C. A utility criterion for Markov decision processes. *Management Science*, INFORMS, v. 23, n. 1, p. 43–49, 1976. Citado 3 vezes nas páginas 19, 26 e 27.
- KADOTA, Y.; KURANO, M.; YASUDA, M. Discounted Markov decision processes with utility constraints. *Computers and Mathematics with Applications*, Pergamon Press, Tarrytown, New York, USA, v. 51, n. 2, p. 279–284, 2006. Citado na página 26.
- LUENBERGER, D. G. *Investment science*. 2. ed. New York, USA: Oxford University Press, 2013. Citado na página 27.
- MIHATSCH, O.; NEUNEIER, R. Risk-sensitive reinforcement learning. *Machine Learning*, Kluwer Academic Publishers, v. 49, n. 2, p. 267–290, 2002. Citado 10 vezes nas páginas 19, 20, 26, 29, 30, 31, 32, 33, 55 e 56.
- MOLDOVAN, T. M.; ABBEEL, P. Safe exploration in Markov decision processes. In: *Proceedings of the 29th International Conference on Machine Learning*. Edinburgh, Scotland, UK: International Machine Learning Society, 2012. p. 1451–1458. Citado na página 26.

- MORIMURA, T.; SUGIYAMA, M.; KASHIMA, H.; HACHIYA, H.; TANAKA, T. Nonparametric return distribution approximation for reinforcement learning. In: *Proceedings of the 27th International Conference on Machine Learning*. Haifa, Israel: International Machine Learning Society, 2010. p. 799–806. Citado na página 27.
- NILIM, A.; GHAOUI, L. E. Robust control of Markov decision processes with uncertain transition matrices. *Operations Research*, INFORMS, Linthicum, Maryland, USA, v. 53, n. 5, p. 780–798, 2005. Citado na página 26.
- PATEK, S. D. On terminating Markov decision processes with a risk-averse objective function. *Automatica*, Elsevier, v. 37, n. 9, p. 1379–1386, 2001. Citado 3 vezes nas páginas 19, 26 e 27.
- PUTERMAN, M. L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. 1. ed. New York, NY, USA: John Wiley and Sons, 1994. Citado 4 vezes nas páginas 18, 22, 24 e 25.
- ROTHBLUM, U. G. Multiplicative Markov decision chains. *Mathematics of Operations Research*, INFORMS, v. 9, n. 1, p. 6–24, 1984. Citado 3 vezes nas páginas 19, 26 e 27.
- SHEN, Y.; TOBIA, M. J.; SOMMER, T.; OBERMAYER, K. Risk-sensitive reinforcement learning. *Neural computation*, MIT Press, v. 26, n. 7, p. 1298–1328, 2014. Citado na página 18.
- SOBEL, M. J. The variance of discounted Markov decision processes. *Journal of Applied Probability*, Applied Probability Trust, v. 19, n. 4, p. 794–802, 1982. Citado 2 vezes nas páginas 19 e 26.
- SUTTON, R.; BARTO, A. *Reinforcement learning: An introduction*. Cambridge, Massachusetts, USA: MIT Press, 1998. v. 116. Citado 2 vezes nas páginas 23 e 29.
- TAMAR, A.; XU, H.; MANNOR, S. Scaling up robust MDPs by reinforcement learning. In: *Proceedings of the 31st International Conference on Machine Learning*. Beijing, China: International Machine Learning Society, 2014. v. 32, p. 181–189. Citado na página 26.
- YU, S. X.; LIN, Y.; YAN, P. Optimization models for the first arrival target distribution function in discrete time. *Journal of Mathematical Analysis and Applications*, Academic Press, v. 225, n. 1, p. 193 – 223, 1998. Citado 2 vezes nas páginas 19 e 26.