



ALGORITMO E PROGRAMA PARA ENCONTRAR EQUILÍBRIOS PUROS DE NASH EM JOGOS COM “N” JOGADORES E “M” ESTRATÉGIAS: NASH EQUILIBRIUM FINDER - NEFINDER

RENAN HENRIQUE CAVICCHIOLI SUGIYAMA - rhcsugiyama@fearp.usp.br
UNIVERSIDADE DE SÃO PAULO - USP - RIBEIRÃO PRETO

ALEXANDRE BEVILACQUA LEONETI - ableoneti@usp.br
UNIVERSIDADE DE SÃO PAULO - USP - RIBEIRÃO PRETO

Resumo: *DADA A COMPLEXIDADE DA BUSCA DE EQUILÍBRIOS DE NASH É ESSENCIAL A UTILIZAÇÃO DE UM ALGORITMO CAPAZ DE ENCONTRAR ESTES EQUILÍBRIOS, QUANDO ELES EXISTIREM. NO ENTANTO, APLICAR MANUALMENTE OS ALGORITMOS PRESENTES NA LITERATURA TORNA-SE UMA TARREFA QUE REQUER UMA GRANDE QUANTIDADE DE TEMPO NOS CASOS ONDE TRÊS OU MAIS JOGADORES ESTÃO ENVOLVIDOS. O OBJETIVO DESTA PESQUISA FOI DESENVOLVER UM ALGORITMO PARA ENCONTRAR DE FORMA COMPUTACIONAL TODOS EQUILÍBRIOS PUROS DE NASH EM JOGOS COM “N” OS JOGADORES E “M” ESTRATÉGIAS (“N” E “M” NÚMEROS FINITOS), PARA SER UTILIZADO EM APLICAÇÃO SIMPLES DA TEORIA DOS JOGOS. A FIM DE TESTAR O ALGORITMO, UM SUPLEMENTO DO EXCEL® FOI PROGRAMADO EM VISUAL BASIC APPLICATION. ESTA FERRAMENTA PRÁTICA FACILITA ENCONTRAR EQUILÍBRIUM DE NASH E É DESTINADA A CONTRIBUIR PARA A DISSEMINAÇÃO DO CONHECIMENTO NA ÁREA DE TEORIA DOS JOGOS.*

Palavras-chaves: *TEORIA DOS JOGOS; EQUILÍBRIUM DE NASH; ALGORITMO*

Área: *6 - PESQUISA OPERACIONAL*

Sub-Área: *6.5 - TEORIA DA DECISÃO E TEORIA DOS JOGOS*

ALGORITHM AND PROGRAM TO FIND PURE NASH EQUILIBRIA IN GAMES WITH “N” PLAYERS AND “M” STRATEGIES: THE NASH EQUILIBRIUM FINDER – NEFINDER

Abstract: *GIVEN THE COMPLEXITY OF FINDING NASH EQUILIBRIA IT IS ESSENTIAL TO USE AN ALGORITHM CAPABLE OF FINDING THE EQUILIBRIA, WHEN THEY EXIST. HOWEVER, MANUALLY APPLYING THE ALGORITHMS PRESENT IN THE LITERATURE BECOMES A TASK THAT REQUIRES A LOT OF TIME IN THE CASES OF THREE OR MORE PLAYERS. THE OBJECTIVE OF THIS RESEARCH WAS TO DEVELOP AN ALGORITHM FOR FINDING ALL PURE NASH EQUILIBRIA IN GAMES WITH “N” PLAYERS AND “M” STRATEGIES (“N” AND “M” FINITE NUMBERS), TO SIMPLE APPLICATIONS IN GAME THEORY. IN ORDER TO TEST THE ALGORITHM AN EXCEL® ADD-IN WAS PROGRAMED IN VISUAL BASIC APPLICATION LANGUAGE. A NEW PRACTICAL TOOL THAT FACILITATES FINDING NASH EQUILIBRIUM IS AIMED TO CONTRIBUTE FOR THE DISSEMINATION OF KNOWLEDGE IN GAME THEORY AREA.*

Keyword: *GAME THEORY, NASH EQUILIBRIA, ALGORITHM*

1. Introdução

Enquanto que no passado os gestores tinham que tomar decisões com ausência de dados ou informações, nos dias de hoje a situação foi invertida. A grande disponibilidade de dados e informações faz com que os gestores tenham que desenvolver habilidade para obter dados e informações de qualidade para que o resultado da tomada de decisão seja satisfatório. Desta forma, existe a necessidade da busca por método/teorias capazes de formular os problemas multicritérios levando em consideração as diferentes opiniões dos envolvidos no processo de decisão de forma a se encontrar uma solução que atenda a todos os participantes de forma conjunta e estável.

A Teoria dos Jogos é uma técnica que permite atingir tal objetivo. Neste contexto, de acordo com Myerson (1996), o Equilíbrio de Nash se torna importante para o estudo do complexo comportamento humano na tomada de decisão. A aplicação do Equilíbrio de Nash permite a busca por uma alternativa de consenso e, não necessariamente, a Pareto-Ótima como nos métodos modelados para indivíduos. Read et al (2013) nomeiam os sistemas de otimização individual de “sistemas engenheiros”, que visam pela solução Pareto-Ótimo, mas que pode não ser facilmente aplicada, enquanto a Teoria dos Jogos busca chegar a uma solução estável e factível, mesmo não sendo uma solução do tipo Pareto-Ótima.

Aplicações da Teoria dos Jogos tem sido relatadas para a resolução de casos de alocação de recursos naturais ou a preservação dos mesmos em problemas multicritérios em grupos, todos envolvendo aspectos econômicos, sociais e ambientais. Como exemplo, cita-se o estudo realizado por Leoneti et al (2010) onde os autores comparam um modelo de auxílio a tomada de decisão para estabelecer um sistema de tratamento de água e esgoto no qual o enfoque é a diminuição dos custos (otimização) com os resultados encontrados pelo Equilíbrio de Nash que leva em consideração tanto o fator econômico quanto ambiental. Um estudo similar foi realizado por Lee (2012) no qual é comparado um modelo convencional para resolução de um problema multi-objetivo com os resultados encontrados pelo equilíbrio de Nash em um estudo de caso para estabelecer um reservatório de águas em Tseng-Wen com mais de um agente envolvido.

Existe na literatura algoritmos para a procura do Equilíbrio de Nash. Entretanto, aplicar estes algoritmos manualmente na busca pelos equilíbrios de Nash se torna uma tarefa que exige bastante tempo nos casos com três ou mais jogadores, pois nestas situações são necessários diversos cálculos. Desta forma, surge a necessidade do uso de *softwares* para a aplicação ágil e perfeita dos algoritmos desenvolvidos. De acordo com a *Game Theory*

Society (2015), um programa para encontrar os equilíbrios de Nash é útil para aplicações experimentais por aqueles que querem encontrar os resultados teóricos de seus modelos e comparar com os resultados experimentais.

Neste contexto, o objetivo desta pesquisa foi desenvolver um programa capaz de encontrar todos os equilíbrios puros de Nash em jogos com “n” jogadores e “m” estratégias (“n” e “m” números finitos). Para o desenvolvimento do programa foi elaborado um algoritmo próprio utilizando como base os algoritmos descritos em artigos encontrados na literatura. Justifica-se a aplicação, pois, enquanto que os algoritmos tradicionais tem como principal objetivo diminuir a complexidade da procura do equilíbrio, sem necessariamente garantir que todos os equilíbrios sejam encontrados, o algoritmo e seu derivado *software* aqui propostos tem como objetivo encontrar todos os equilíbrios, abrindo mão da diminuição da complexidade, mas tornando a aplicação mais amigável à áreas de ciências diferentes da economia e matemática, que são as áreas tradicionais de aplicação da Teoria dos Jogos.

2. Referencial Teórico

2.1 Algoritmos para encontrar os Equilíbrios de Nash

Alguns trabalhos com o intuito de desenvolver um algoritmo para o cálculo do Equilíbrio de Nash estão presentes na literatura. O famoso artigo de Lemke e Howson (1964) é um dos primeiros artigos a tratar deste problema. A tese do trabalho apresentado é de que é possível provar algebricamente que um ponto de equilíbrio encontra-se em um caminho juntando uma sequência de pontos adjacentes extremos de um poliedro convexo. Tal caminho é calculado utilizando métodos computacionais de programação linear.

Lemke e Howson apresentam uma prova algébrica, válida para qualquer campo ordenado, da existência de pontos de equilíbrio para jogos com dois jogadores. É mostrado que o ponto de equilíbrio pode ser encontrado buscando uma sequência de pontos adjacentes de um certo poliedro convexo. Os autores afirmam que o caminho para encontrar o equilíbrio de Nash pode ser encontrado utilizando métodos de resolução de problemas lineares. Concluindo a demonstração, chegasse a uma prescrição para se encontrar um ponto de equilíbrio de Nash existente (caso exista). O esquema formado pela prova algébrica é capaz de encontrar pelo menos um equilíbrio de Nash, mas não necessariamente todos os equilíbrios.

Outro trabalho, proposto por Dickhaut e Kaplan (1991), também trás um algoritmo com a finalidade de encontrar os equilíbrios de Nash. Os autores introduzem o trabalho

definindo três métodos para se encontrar uma solução para jogos. Esses métodos são o *Minimax*, *Dominant* e o Equilíbrio de Nash. Na abordagem do *Minimax* os jogadores estão interessados em minimizar a máxima insatisfação que pode ser oferecido pelo oponente. Já na abordagem *Dominant* a alternativa escolhida é a dominante, ou seja, a melhor alternativa possível dada as possíveis ações do outro jogador. Por fim, no Equilíbrio de Nash a jogada de cada jogador é a melhor resposta para a jogada do outro jogador. Segundo os autores, o trabalho publicado é útil para iniciantes no processo de encontrar os equilíbrios de Nash, sendo também útil para os mais experientes que podem não estar certos se encontraram todas as alternativas possíveis.

Segundo os trabalhos sobre algoritmos, Govindan e Wilson (2003) propuseram um novo algoritmo para encontrar os Equilíbrios de Nash em jogos finitos. Os autores propuseram a combinação de dois métodos de caminho para resolução de sistemas de equações, sendo o Método de Homotopia de Eaves¹ e Método Global de Newton² para a formulação do algoritmo. Como resultado da combinação dos métodos os autores apresentam um pseudocódigo capaz de encontrar todos os equilíbrios de Nash em jogos com 2 ou mais jogadores. O algoritmo desenvolvido foi testado para jogos com dois ou três jogadores e até 16 estratégias, quatro jogadores e até dez estratégias, cinco jogadores e até oito estratégias e seis jogadores e até 4 estratégias. Após os testes os autores puderam notar que o método é rápido para jogos com dois jogadores, mas o tempo cresce exponencialmente com o aumento deste número. Apesar da elevação do tempo nos cálculos os autores afirmam que o método se tornou mais rápido devido as propriedades do Método Global de Newton que procura rapidamente o primeiro equilíbrio nas proximidades da vizinhança da trajetória. Além disso, observam que para a implantação do método é necessário que os jogos estejam na forma normal.

Govindan e Wilson (2004), apresentam outro trabalho que segue as características do primeiro, de 2003. Nesta nova proposta é desenvolvido mais um método que também advém da mescla de dois outros métodos, sendo eles algoritmo de Lemke-Howson e, novamente, o Método Global de Newton. Este novo método foi desenvolvido para encontrar todos os equilíbrios de Nash em jogos com “n” jogadores. O algoritmo utiliza uma sequência de jogos multimatriz (varias matrizes representando os jogadores e seus pagamentos) através das interações entre os jogadores para convergir aos pontos de equilíbrio. O objetivo do novo

¹ Algoritmo desenvolvido por Eaves (1972) para encontrar pontos fixos

² Método desenvolvido por Smale (1976) denomina Método Global de Newton uma equação diferencial ordinária associada a um sistema de “n” funções reais e “n” variáveis reais.

método é de oferecer um desempenho melhor para o Método Global de Newton, que foi utilizado no trabalho anterior. Este objetivo é alcançado ao atingir três características, sendo a primeira a busca pelo equilíbrio através de aproximações iterativas que já se iniciam próximas do alvo, a segunda é a aproximação do alvo por um jogo polimatriz de tal forma que seja possível aplicar o método de Lemke-Howson, e por fim, a última característica é a redução da dimensão do problema. Novamente, como resultado é apresentado um pseudocódigo para se calcular os equilíbrios de Nash em jogos finitos e também são realizados testes protótipos para verificar a eficiência do mesmo.

Echenique (2007) traz um algoritmo que é capaz de encontrar todos os equilíbrios puros de Nash em jogos com “n” jogadores. O autor classifica seu algoritmo como sendo o primeiro algoritmo não trivial desenvolvido para encontrar todos os equilíbrios puros. O autor classifica como métodos triviais os que se baseiam em fixar um jogador e, para cada conjunto de estratégias do jogador oponente, encontrar a melhor resposta. Então, verifica-se se o jogador oponente quer trocar a alternativa fixada. O método trivial testa todos os conjuntos de estratégias para verificar quais são os equilíbrios. Echenique (2007) apresenta comparações do método com outros algoritmos ditos triviais para demonstrar a rapidez do método proposto. O autor descreve o algoritmo em linguagem superior (informal) para que seja possível compreender o método e também disponibiliza um *link* para o *download* do algoritmo capaz de encontrar todos os equilíbrios puros de Nash. Entretanto, apesar do artigo tratar de um algoritmo capaz de encontrar todos os equilíbrios puros de Nash em jogos com “n” jogadores e “m” estratégias, o código disponibilizado no *link* é destinado a jogos com apenas dois jogadores e o algoritmo é de difícil interpretação.

Porter, Nudelman e Shoham (2008) apresentaram dois métodos para encontrar este equilíbrio em jogos na forma estratégica ou normal, sendo um para jogos com 2 jogadores (figura 2) e um para jogos de “n” jogadores. Os autores procuraram identificar heurísticas que fossem relativamente simples, mas que contivessem um poder computacional grande suficiente para tornar a aplicação prática. Segundo os autores, este problema tem sido descrito como o mais fundamental na interface das ciências da computação e da Teoria dos Jogos. Apesar de várias décadas de investigação sobre este problema a sua complexidade computacional é alta e os novos algoritmos são relativamente escassos e ineficazes (PORTER; NUDELMAN; SHOHAM, 2008).

Porter, Nudelman e Shoham (2008) divergiram dos tradicionais algoritmos aplicados em Teoria dos Jogos em basicamente dois pontos: construíram um novo algoritmo baseado em heurísticas simples; e testaram o algoritmo proposto contra os algoritmos existentes.

Especificamente, eles superaram os algoritmos anteriores de Lemke-Howson em jogos de 2 jogadores, e de Govindan-Wilson em jogos de “n” jogadores. O primeiro passo de ambos os algoritmos é verificar se existe no jogo um equilíbrio de estratégia pura. Esta etapa é feita de maneira rápida, mesmo em jogos grandes. Após isto, os algoritmos passam a explorar os perfis dos jogadores e são capazes de encontrar todos os equilíbrios de Nash.

No primeiro conjunto de experimentos, Porter, Nudelman e Shoham (2008) comparam o desempenho do primeiro algoritmo com o de Lemke-Howson, para 2 jogadores. Foram definidos 300 jogos, elaborados a partir de dois jogadores com 24 distribuições diferentes. Desta forma, os autores concluíram que os algoritmos propostos tendem a achar os equilíbrios mais rapidamente do que os outros algoritmos. Em média, o equilíbrio encontrado por Lemke-Howson em jogos de 2 jogadores têm mais de 10 ações por jogador (de um total de 300), enquanto que o equilíbrio encontrado por Govindan-Wilson em jogos de 6 jogadores têm, em média, 2,3 ações por jogador (PORTER; NUDELMAN; SHOHAM, 2008).

2.2 Softwares para encontrar Equilíbrio de Nash

Apesar da existência de diversos algoritmos, aplicá-los manualmente na busca pelos equilíbrios de Nash se torna uma tarefa que exige bastante tempo nos casos com três ou mais jogadores, pois nestas situações são necessários diversos cálculos. Desta forma, surge a necessidade do uso de *softwares* para a aplicação ágil e perfeita dos algoritmos desenvolvidos. De acordo com a *Game Theory Society* (2015), um programa para encontrar os equilíbrios de Nash é útil para aplicações experimentais por aqueles que querem encontrar os resultados teóricos de seus modelos e comparar com os resultados experimentais. Neste sentido, Dickhaut e Kaplan (1991) publicaram um trabalho em que é apresentado um programa que encontra todos os equilíbrios de Nash em jogos com dois jogadores e, segundo os autores, o trabalho é útil para iniciantes no processo de encontrar os equilíbrios de Nash. Dentre outros exemplos na literatura, destacam-se os *softwares*: *Gambit* (McKelvey et al., 2007), *Game Theory Explorer* (GTE) (Egesdal, M, et al, 2014) e o *GamePlan* (Langlois, J. P., 2000).

Desenvolvido durante 25 anos, o *Gambit* é um *software open-source* cuja finalidade é encontrar os Equilíbrios Nash apresentando uma biblioteca de algoritmos para tal proposta. A linguagem de programação utilizada na última versão foi o Python. O programa pode ser utilizado no sistema Windows, Linux ou Mac. Com ele é possível construir e analisar jogos finitos na forma extensiva e estratégica para jogos não cooperativos. Apesar da intensa evolução, o *software* suporta apenas jogos finitos, ou seja, os algoritmos desenvolvidos não se

aplicam para jogos em que a combinação de estratégias possíveis para todos os jogadores é ilimitada. Além disso, o *Gambit* só se aplica em jogos não cooperativos sendo jogos em que as regras são explícitas e conhecidas por todos além de as ações dos jogadores serem independentes. Por fim, para jogos complexos o programa se torna impraticável devido à longa demora para encontrar os resultados.

O *Game Theory Explorer* (GTE) foi desenvolvido com o objetivo futuro de ser integrado aos módulos do *Gambit*, sendo um *software open-source* em que, através de uma interface gráfica para *web browser*, permite a construção iterativa de jogos na forma extensiva e estratégica e encontrar os equilíbrios para esses jogos. Apesar de ser *online*, é possível criar e gravar os jogos no computador. Os jogos podem ser gravados em formato de imagens para posteriormente serem utilizados em apresentações, caso necessário. O foco do projeto foi desenvolver um *software* com uma interface amigável para o uso de pessoas não especialistas de tal forma que a visualização dos jogos se torne intuitiva. As formas gráficas, tais como os jogos em formato de árvore, podem ser customizadas, por exemplo, na vertical e/ou na horizontal. Com o uso do GTE é possível encontrar todos os equilíbrios de Nash para dois jogadores. Vale ressaltar que o tempo de processamento do programa cresce de forma exponencial com o aumento do número de estratégias possíveis. Desta forma, o número de estratégias se restringe em torno de 15 a 20 por jogador. A partir daí o tempo de processamento se torna longo demais exigindo o uso de outras ferramentas para encontrar os equilíbrios. O programa foi escrito em linguagem *ActionScript* e *JavaScript* e pode ser acessado via *web browser* e exibido em “*Flash Player*”.

O *GamePlan* foi desenvolvido para criar e resolver uma ampla gama de modelos de jogos nas formas normal e extensiva, para jogos com informação perfeita ou imperfeita, repetitivos ou não e com estruturas gráficas (Langlois, J. F., 2000). Segundo Langlois (2000) no desenvolvimento do *GamePlan* buscou-se atingir quatro objetivos. O primeiro é de ser um *software* “amigável”, ou seja, deve ser fácil a criação, edição e resolução dos jogos. Desta forma, o programa apresenta várias cores para diferenciar os jogadores, os pagamentos e outras informações com a finalidade de facilitar a distinção dos dados. O segundo objetivo é ser flexível. Assim buscou-se um programa capaz de suportar praticamente todo tipo de estrutura de jogos, tendo como limite apenas a capacidade de memória e velocidade dos computadores. O terceiro objetivo foi de ser um *software* exaustivo. Para tanto, há três opções de resolução para encontrar o(s) equilíbrio(s) de Nash, sendo elas “*Pure*”, “*Explorer*” e “*Sample*” e também há três conceitos de solução aplicados, Nash, Bayesian Perfeito e Sequencial. Por fim, o quarto objetivo do *software* foi de ser rápido.

3. Metodologia de pesquisa

O presente estudo é uma pesquisa aplicada, com a proposição de um algoritmo matemático para encontrar todos os equilíbrios puros em jogos com “n” jogadores e “m” estratégias (“n” e “m” números finitos) utilizando como base algoritmos já descritos em artigos encontrados na literatura. Visando sua utilização em Teoria dos Jogos aplicada, o método é capaz de encontrar todos os equilíbrios puros para 2 ou mais jogadores em jogos não cooperativos com 2 ou mais estratégias (alternativas).

Primeiramente, foi realizada uma pesquisa exploratória por meio de uma busca de publicações de artigos na base de dados do *SciencDirect* para encontrar algoritmos que encontrem os equilíbrios de Nash em jogos. Em seguida, foi realizada uma pesquisa para encontrar *softwares* disponíveis para encontrar o Equilíbrio de Nash. A busca foi realizada em sites especializados, tais como os disponíveis no site da *Game Theory Society*. Os softwares encontrados foram o *Gambit*, voltado para usuários especialistas, o *Game Theory Explorer*, cujo diferencial é sua interface gráfica e o *GamePlan*, capaz de encontrar todos os equilíbrios de Nash tanto em jogos com informação perfeita ou imperfeita, repetitivos ou não.

Por fim, para a aplicação do algoritmo desenvolvido, um *software* foi proposto como suplemento do Microsoft Excel. Para isto, inicialmente foi realizado o estudo da linguagem *Visual Basic for Applications* – VBA para que fosse possível a programação do algoritmo. O estudo foi feito de forma remota, utilizando aulas oferecidas pelo *Wise Owl Tutorials* no ambiente de internet. Foram assistidas 18 aulas de cerca de 30 minutos cada uma e, em cada uma delas os conceitos de programação em VBA foram testados. O *software* Gambit foi utilizado para calibrar e validar o software.

4. Resultados e discussões

4.1 Proposição de um algoritmo para jogos de “n” jogadores com “m” estratégias

Dentre os algoritmos disponíveis na literatura, o algoritmo proposto por Porter, Nudelman e Shoham (2008) se mostrou o mais amplo e fácil de ser implementado, pois o algoritmo é baseado em uma heurística simples e encontra os equilíbrios de forma mais rápida com relação aos demais algoritmos que foram comparados. Além disso, o algoritmo é voltado para jogos com “n” jogadores e “m” estratégias encontrando tanto os equilíbrios puros quanto os mistos. Todavia, a interpretação e posterior programação do algoritmo proposto por Porter, Nudelman e Shoham (2008) foi inviabilizada, dada a dificuldade de interpretação do mesmo e

também pela forma genérica com que foi apresentado, considerando a leitura por profissionais de ciências aplicadas. Em um segundo movimento, recorrendo então aos demais algoritmos encontrados na literatura, outros problemas foram encontrados. No caso dos outros trabalhos estudados, ora os autores não apresentaram os algoritmos em linguagem computacional, ora os algoritmos apresentados não satisfaziam os requisitos básicos, ou seja, não eram aplicáveis a jogos com “n” jogadores e “m” estratégias. Desta forma, um novo algoritmo foi proposto.

O novo algoritmo baseou-se na definição do equilíbrio de Nash. Segundo Nash (1951), o equilíbrio é encontrado quando em um jogo existam nas linhas as escolhas l_1, l_2, \dots, l_L , e na coluna as escolhas c_1, c_2, \dots, c_C , e para cada escolha “l” feita por linha, haja uma melhor resposta para coluna $b_c(l)$, e para escolha “c” feita por coluna, haja uma melhor resposta para a linha $b_l(c)$. De forma geral, o equilíbrio de Nash é um par de estratégias (l^*, c^*) tal que: $c^* = b_c(l^*)$; e $l^* = b_l(c^*)$. De outra forma, para que uma combinação de estratégias seja considerada um equilíbrio de Nash é necessário que, para cada estratégia s_i^* , tenhamos $f_i(s_i^*, s_{-i}^*) \geq f_i(s_i, s_{-i})$, para todo s_i e todo “i”, onde f_i é a função de recompensa para cada jogador “i”. O fluxograma da Figura 1 apresenta, resumidamente, o algoritmo proposto.

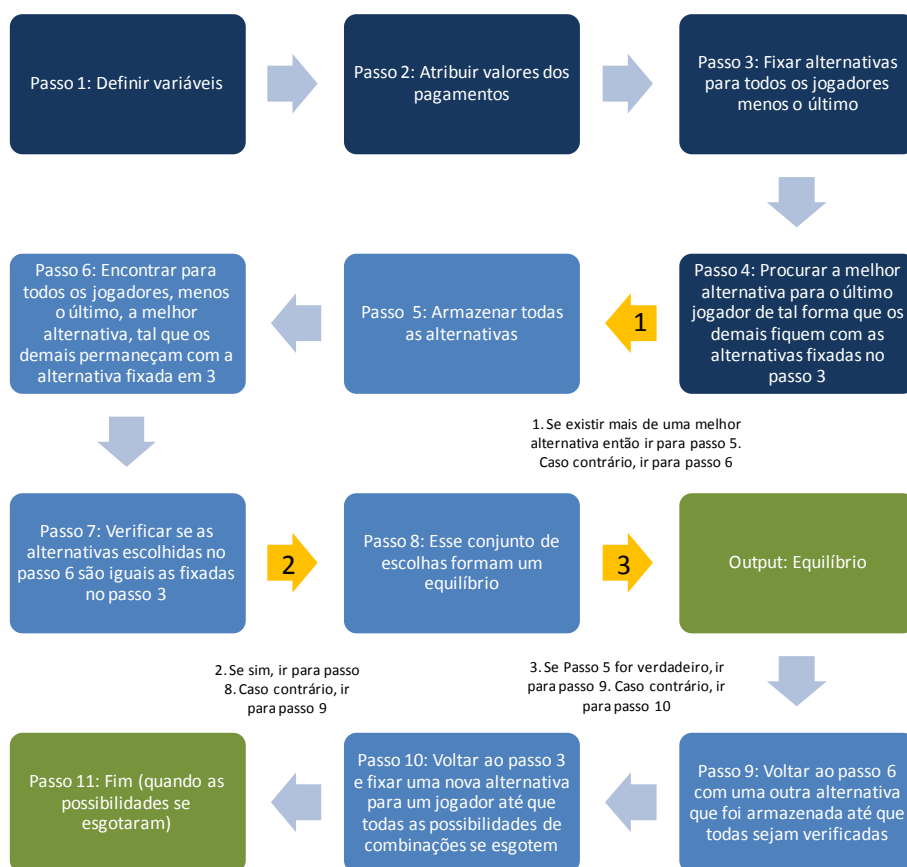


FIGURA 1 - Fluxograma do algoritmo proposto

Como exemplo, um jogo com três jogadores e duas alternativas é aqui apresentado. As combinações possíveis para se fixar as alternativas estão apresentadas na Figura 2. Nesta figura, estão apresentadas todas as etapas de fixação de alternativas (passo3) que seriam necessárias para um jogo com três jogadores e duas alternativas. A primeira vez que o algoritmo chega ao passo 3 será fixada a alternativa 1 para os jogadores 1 e 2. Na próxima iteração, deve-se fixar alguma outra combinação descrita na figura, por exemplo, fixar a alternativa 2 para o jogador 1 e a alternativa 1 para o jogador 2. A cada iteração uma nova combinação deve ser escolhida para se fixar as alternativas aos jogadores até que todas as combinações possíveis tenham sido testadas.

	Jogador 1	Jogador 2	Jogador 3
Combinações possíveis de alternativas para se fixar	1	1	livre
	2	1	livre
	1	2	livre
	2	2	livre

FIGURA 2 - Jogo de 3 jogadores e 2 alternativas

4.2 NEFinder: Nash Equilibrium Finder

Após o desenvolvimento do algoritmo, um *software* foi escrito em linguagem VBA e o algoritmo foi programado como um suplemento para Excel®. A linguagem VBA foi escolhida para que fosse possível executar o programa dentro do Excel® facilitando sua utilização pelos usuários devido à grande disseminação e emprego do pacote *Microsoft Office*. O *software* recebeu o nome de *Nash Equilibrium Finder*, com o codinome *NEFinder*, que tem como objetivo ser uma ferramenta para a procura de equilíbrios de Nash puros em ambiente VBA, uma vez que o *Gambit* apresenta versão somente para o ambiente Phyton.

O *NEFinder* apresenta duas telas importantes. A primeira está relacionada com a criação do jogo. Nesta tela é possível definir o número de jogadores, alternativas e o nome da nova planilha onde o *template* do jogo será criado. O programa limita o número de jogadores e alternativas a no mínimo dois, pois se trata de uma decisão em grupo e não existirá jogo com apenas uma única alternativa. A Figura 3 apresenta a tela de criação do jogo *NEFinder*.

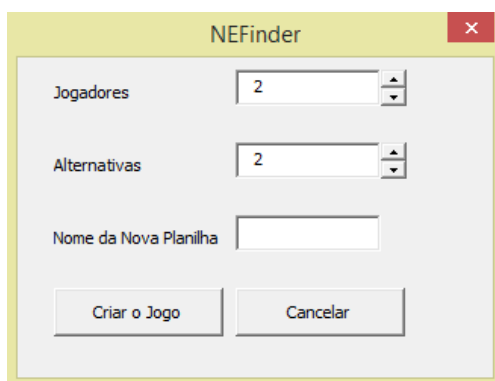


FIGURA 3 - NEFinder - Tela de definição dos parâmetros dos jogos

Após definir o número de jogadores e alternativas o jogo é criado na forma estratégica ao clicar em “Criar o Jogo”. Cada jogador será representado por uma coluna e todas as diferentes possibilidades de estratégias estarão dispostas em linhas. A Figura 4 apresenta a configuração do jogo para o caso de 2 jogadores e 2 estratégias, sendo que ao lado do quadro cinza estão as estratégias possíveis com cores diferentes, sendo que cada cor representa a estratégia de cada jogador. Por exemplo, a linha 4 da figura representa o par de alternativa em que ambos os jogadores optam pela alternativa 1, já na linha 5 o jogador 1 (representado em vermelho) escolhe a alternativa 2 enquanto que o jogador 2 (representado pela cor verde) opta pela alternativa 1. As cores de cada jogador são definidas automaticamente pelo programa para facilitar a visualização pelo usuário. Os campos do quadro cinza representam os valores dos pagamentos de cada jogador. Ressalta-se que os valores devem ser numéricos e positivos.

	A	B	C	D	E
1	Jogo para 2 jogadores 2 alternativas.				
2					
3			jogador 1	jogador 2	
4	1	1			
5	2	1			
6	1	2			
7	2	2			
8					
9					

FIGURA 4 - Template do jogo para dois jogadores e duas estratégias

Para o preenchimento dos pagamentos, a leitura das estratégias é feita nas linhas. Por exemplo, se a linha contiver o número dois em vermelho e o número um em verde significa que o jogador representado pela cor vermelha escolhe a alternativa dois e o jogador representado pela cor verde escolhe a alternativa um. Acima do quadro cinza estão os jogadores. Desta forma, o quadro cinza representa uma matriz bidimensional, a qual será preenchida com os valores dos pagamentos. Supondo que o pagamento do jogador 1, para o

caso em que ambos os jogadores optem pela estratégia 1, seja de “dois”, deve ser preenchido com o valor “dois” onde há o encontro da coluna “jogador 1” com a linha das estratégias correspondente a escolha da alternativa 1 para ambos os jogadores. Na figura 5, o valor do pagamento do exemplo, seria inserido na célula C4.

	A	B	C	D	E
1	Jogo para 2 jogadores 2 alternativas.				
2					
3			jogador 1	jogador 2	
4	1	1	2	1	
5	2	1	0	0	
6	1	2	0	0	
7	2	2	1	2	
8					
9					

FIGURA 5 - Modelo de preenchimento dos pagamentos dos jogadores

Como visto na Figura 5, o pagamento do jogador 1 é de “dois”, caso ambos os jogadores optem pela alternativa um. Assim, este valor deve ser preenchido no ponto de encontro da coluna “Jogador 1” com a linha das estratégias em que tanto o jogador um quanto o jogador dois escolhem a alternativa um. Desta forma, o valor dois deve ser preenchido na célula C4. Para o mesmo par de alternativas escolhidas, o pagamento do jogador dois é de “um”. Assim, é mantida a mesma linha, mas a coluna em questão é trocada para a coluna “Jogador 2”. Portanto, o valor do pagamento do jogador dois no caso de ambos optarem pela alternativa um deve ser inserido na célula D4. O mesmo raciocínio é aplicado até que todos os pagamentos sejam aplicados nos campos em cinza, conforme demonstrado na figura acima.

A segunda tela está relacionada ao cálculo do equilíbrio. Dentro do suplemento existe outro botão que serve para tal finalidade, bastando selecionar a área em cinza onde foram inseridos os pagamentos e clicar em “Calcular” para que os equilíbrios e os pagamentos de cada jogador sejam encontrados (Figura 6). Cada equilíbrio será apresentado na mesma planilha em que o jogo foi criado.

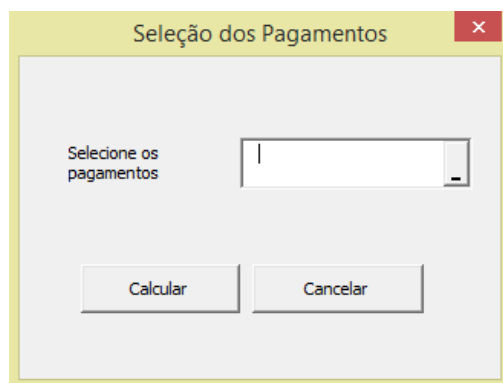


FIGURA 6 - NEFinder - Tela de seleção dos pagamentos

Apesar da existência de um *software* mais amplo como o *Gambit*, capaz de encontrar todos os equilíbrios de Nash para jogos com “n” jogadores e “m” estratégias de forma menos complexa, o *NEFinder* apresentou algumas vantagens quanto ao seu manuseio. O fato de ser implantando dentro do ambiente Excel® torna o preenchimento dos valores de pagamentos mais ágil, além do uso do Excel® ser mais difundido e conhecido pelos usuários. Outra vantagem do *NEFinder* é relacionado, principalmente, a simulação de diversos jogos, pois pode ser automatizado através de macros. O *NEFinder* está em processo de registro pela Agência USP de Inovação, sob o protocolo 14.1.01106.81.9.

5. Considerações finais

Após a revisão bibliográfica foi possível identificar os principais artigos referentes aos algoritmos voltados para encontrar os equilíbrios de Nash. Entretanto, alguns problemas foram encontrados quanto à aplicação dos algoritmos, a saber: (i) algoritmo descrito em linguagem inferior, de difícil interpretação; (ii) algoritmo não capaz de encontrar todos os equilíbrios (puros e mistos); e (iii) algoritmo não desenvolvido para “n” jogadores e “m” estratégias (n e “m” números reais e finitos). Desta forma, o desenvolvimento de um novo algoritmo foi proposto para contornar as limitações encontradas nos disponíveis na literatura. O algoritmo foi concluído e, para sua utilização, foi programado o *software NEFinder* que é capaz de encontrar todos os equilíbrios puros em jogos não cooperativos com “n” jogadores e “m” estratégias em ambiente Excel®. Com intuito de expandir a aplicação do programa, será necessário o desenvolvimento de um método capaz de encontrar todos os equilíbrios mistos, o que poderá ser explorado em pesquisas futuras.

Agradecimentos: à FAPESP pela bolsa de Iniciação Científica

Referências

- DICKHAUT, J.; KAPLAN, T. A program for finding Nash equilibria. *The Mathematica Journal*, 1991.
- EAVES, B. "Homotopies for the computation of fixed points". *Math. Program.* 3 (1972) 25–237.
- ECHENIQUE, F.. "Finding all equilibria in games of strategic complements." *Journal of Economic Theory* 135.1 (2007): 514-532.
- EGESDAL, M.; JORDANA, A. G.; PRAUSE, M.; SAVANI, R.; STENGEL, B. V. *Game Theory Explorer*. 2014. Disponível em <<http://www.gametheoryexplorer.org/>>. Acesso: 15/10/2014.
- GAME THEORY SOCIETY 2015. Disponível em < <http://www.gametheorysociety.org/>>. Acesso em: 11/10/2014

- GOVINDAN, S.; WILSON, R. Computing Nash equilibria by iterated polymatrix approximation. *Journal of Economic Dynamics and Control*, Elsevier, 2004.
- GOVINDAN, S.; WILSON, R. A global Newton method to compute Nash equilibria. *J. Econ. Theory*, 2003, p. 65–86.
- LEE, C.,S.. "Multi-objective game-theory models for conflict analysis in reservoir watershed management." *Chemosphere* 87.6 (2012): 608-613.
- LEMKE, C., HOWSON, J. Equilibrium points of bimatrix games. *J. Soc. Ind. Appl. Math.* Vol. 12, 1964, p. 413–423.
- LEONETI, A., B.; OLIVEIRA, S., V., W., B; OLIVEIRA, M., M., B. "O equilíbrio de Nash como uma solução para o conflito entre eficiência e custo na escolha de sistemas de tratamento de esgoto sanitário com o auxílio de um modelo de tomada de decisão." *Eng Sanit Ambient* 15.1 (2010): 53-64.
- MYERSON , R., B. "John Nash's Contribution to Economics". *Games and Economic Behavior*, Volume 14, Issue 2, 1996, Pages 287-295.
- PORTER, R.; NUDELMAN, E.; SHOHAM, Y. Simple search methods for finding a Nash equilibrium. *Games and Economic Behavior* 63 (2008) 642–662.
- READ, L., MADANI, K.; INANLOO, B.. "Optimality versus stability in water resource allocation." *Journal of environmental management* 133 (2014): 343-354.