

Universidade Federal de Ouro Preto - UFOP
Instituto de Ciências Sociais Aplicadas - ICSA
Programa de Pós-Graduação em Economia Aplicada

Gustavo de Castro Silva Versiani Passos

Algoritmos Genéticos Aplicados a um Modelo Evolucionário de Dinâmica Industrial

Dissertação apresentada ao Programa de Pós-Graduação em Economia Aplicada para obtenção do título de mestre em Economia Aplicada.

Prof. Dr. Martin Harry Vargas Barrenechea

Ouro Preto

2018

P289a Passos, Gustavo de Castro Silva Versiani .
Algoritmos Genéticos Aplicados a um Modelo Evolucionário de Dinâmica Industrial [manuscrito] / Gustavo de Castro Silva Versiani Passos. - 2018.
68f.: il.: grafs; tabs.

Orientador: Prof. Dr. Martin Harry Vargas Barrenechea.

Dissertação (Mestrado) - Universidade Federal de Ouro Preto. Instituto de Ciências Sociais Aplicadas. Departamento de Ciências Econômicas e Gerenciais. Programa de Pós-Graduação em Economia.
Área de Concentração: Economia Aplicada.

1. Algoritmos Genéticos. 2. Modelos Evolucionários . 3. Aprendizagem. 4. Organização Industrial. I. Barrenechea, Martin Harry Vargas . II. Universidade Federal de Ouro Preto. III. Título.

CDU: 330.101.8

Gustavo de Castro Silva Versiani Passos

ALGORITMOS GENÉTICOS APLICADOS A UM MODELO EVOLUCIONÁRIO
DE DINÂMICA INDUSTRIAL.

Trabalho apresentado ao Programa de Pós-Graduação em Economia Aplicada do Instituto de Ciências Sociais e Aplicadas (ICSA) da Universidade Federal de Ouro Preto (UFOP) como requisito parcial para a obtenção do grau de Mestre em Economia Aplicada, aprovado em 26 de abril de 2018.

Banca Examinadora:



Prof.(a). Dr.(a). Martin Harry Vargas Barrenechea - Orientador
(UFOP)



Prof.(a). Dr.(a). Adélcio Carlos de Oliveira (UFSJ)



Prof.(a). Dr.(a). Antônio Francisco Neto (UFOP)

Agradecimentos

A conclusão dessa dissertação não teria sido possível sem todos aqueles que contribuíram para esse percurso. Em primeiro lugar, foi fundamental o apoio incondicional dos meus pais, dos meus irmãos e da Cíntia, que juntos simplificam quaisquer desafios e tornam todos os caminhos mais agradáveis.

Agradeço ao Prof. Dr. Martin H. Barrenechea por toda orientação e disponibilidade nessa busca por conhecimento, assim como a todos os professores e funcionários que fazem parte desse programa de mestrado e à FAPEMIG pelo apoio financeiro. Por último, agradeço a todos os meus colegas e amigos, em especial ao Douglas e à Anna Carolina, pelas contribuições e companhia nessa jornada.

*“Um investimento em conhecimento
sempre paga os melhores juros”
(Benjamin Franklin)*

Resumo

Essa dissertação consiste na aplicação de um algoritmo genético ao modelo evolucionário de dinâmica industrial de Nelson e Winter. Adapta-se o método de um trabalho anterior, [Yildizoglu \(2002\)](#), buscando verificar os efeitos da adição de firmas que utilizam o algoritmo na decisão de investimento em P&D inovativo, quando estas dividem o mercado com outros dois tipos de firmas, aquelas com estratégias fixas e as de estratégias aleatórias. A partir da descrição do modelo, utilizando de modelagem baseada em agentes, implementa-se os mercados, cuja simulação é realizada em NetLogo. Foram comparadas situações diferentes de investimento externo e diversas durações dos períodos de aprendizagem para os algoritmos genéticos. Os resultados obtidos foram contrastantes com trabalhos anteriores, pois, apesar de se confirmar que a presença de firmas, que fazem uso da aprendizagem, leva à dominância do mercado, o mesmo não acontece em relação à melhora da eficiência tecnológica e do bem-estar social.

Palavras-chave: Algoritmos Genéticos; Modelos Evolucionários Industriais; Aprendizagem.

Abstract

This dissertation aims to apply a genetic algorithm to an evolutionary industry model of Nelson and Winter. A methodology from a previous paper, Yildizoglu (2002), is adapted to verify the effects of the addition of firms that use a genetic algorithm to decide their innovative R&D investment. These firms share the market, separately, with two other different kinds, those that use a fixed rate of investment and those with random strategies. From the model's description, using an agent based model, both markets are implemented and simulated in NetLogo. Distinct situations of external credit and for various durations of learning period were compared. The results of the simulation brought contrasting findings when comparing with previous works. Although it was confirmed that learning implies in market dominance, the same was not true in respect to the improvement of technological efficiency and social welfare.

Keywords: *Genetic Algorithms; Evolutionary Industry Models; Learning.*

Sumário

1 - INTRODUÇÃO	8
2 - ALGORITMOS GENÉTICOS	16
2.1 - Métodos de Seleção	17
2.2 - Cruzamento	22
2.3 - Mutação	23
3 - DESCRIÇÃO DO MODELO	25
3.1 - Modelo Evolucionário(Protocolo ODD)	25
4 - TESTE DO ALGORITMO GENÉTICO	35
5 - RESULTADOS	39
6 - CONCLUSÃO	48
REFERÊNCIAS	51
APÊNDICE A	53
APÊNDICE B	58

1 - Introdução

A dinâmica industrial foi explicada pelo modelo evolucionário desenvolvido por [Nelson e Winter \(1982\)](#), no qual cada firma investe em P&D com o intuito de aumentar sua produtividade, este investimento é dividido em imitação e inovação. Existem dois tipos diferentes de firmas, as que apenas imitam e aquelas onde os processos de inovação e imitação ocorrem de forma simultânea. A inovação consiste na pesquisa por novos processos ou produtos, já a imitação, é a busca pela incorporação das melhores práticas já existentes no mercado, ou seja procura-se copiar a prática ou processo do concorrente. O nível de produtividade atingido por cada firma em determinado período é mantido para o próximo, dando início ao processo evolucionário da indústria.

Segundo o modelo de [Nelson e Winter \(1982\)](#), o fator relevante para que uma firma possa inovar, é seu tamanho, que se reflete na sua capacidade de investimento em P&D, pois a quantia investida consiste em um valor fixo acrescido de uma porcentagem do seu estoque de capital, portanto, firmas com mais capital possuem uma maior a capacidade de investimento das firmas, e conseqüentemente, maior é a probabilidade de sucesso da inovação. A relação entre a estrutura de mercado e a inovação é de duas vias, ou seja, a estrutura do mercado influencia na busca por inovação e a inovação exerce influência na estrutura de mercado. Assim, uma empresa pode crescer em relação aos seus competidores quando ela não é imitada, mas uma empresa, que possui como estratégia dominante a imitação, pode vir a dominar o mercado, pois os custos de pesquisa em inovação são maiores que aqueles atribuídos à imitação. Ademais, as simulações entre inovadores e imitadores acabam demonstrando que, devido às estruturas iniciais das simulações realizadas, em geral, a escolha por inovação não é lucrativa.

Na teoria econômica, quando os mercados se encontram em concorrência perfeita, o lucro das firmas, a longo prazo, é igual a zero, já em um oligopólio o lucro não é nulo, fazendo com que a entrada nesse mercado seja interessante para novas firmas. Para que esta ocorra, as firmas devem vencer as barreiras de entrada. O surgimento de novas firmas é estudado por [Winter \(1984\)](#) que considera as situações de entrada dessas novas firmas na indústria, analisando as condições de mercado que torna interessante o ingresso de uma empresa inovadora e o potencial dessas novas empresas. São definidas, também, as condições de saída do mercado, quando uma saída ocorre tem-se uma redução na oferta e conseqüentemente um aumento no preço dos produtos para o próximo período.

Sobre a conduta das firmas, [Witt \(1986\)](#) estuda o comportamento das empresas em diversos cenários de informação imperfeita, através da simulações utilizando três diferentes conjuntos de regras, representados por três modelos. O intuito do trabalho, é

verificar se as escolhas das firmas de maximizar seus lucros, ou para o caso de informação incompleta, agir como se os estivessem maximizando, dominam as estratégias de ajustes comportamentais e de aprendizagem. Para cada experimento utilizando os modelos, são estimadas as probabilidades de sobrevivência e falência das firmas. Os resultados obtidos não indicam uma ordem preferencial nas taxas de sobrevivência, sendo que essa ordem varia de acordo com o modelo utilizado. A saída voluntária errônea tem um papel mais relevante na taxa de sobrevivência que a falência. Portanto, Witt (1986) sustenta a ideia de que a maximização dos lucros não é condição necessária, nem suficiente, para aumentar a probabilidade de sobrevivência de uma firma em um mercado dinâmico.

A ideia de não maximização dos lucros é trabalhada por Schaffer (1989), que aplica os conceitos do fenômeno conhecido na literatura como “*spite effect*”, que se baseia no fato de que o indivíduo pode escolher um conjunto de estratégias que o leva a um ganho menor, desde que, o outro indivíduo, seja ainda mais prejudicado. No caso das firmas, a empresa opta por ter um ganho menor, não buscando a maximização dos lucros, mas seu concorrente, que buscou a maximização, acaba perdendo ainda mais, já que teve despesas resultante da busca pela melhora de seu desempenho. A partir de conceitos Hamiltonianos de biologia evolucionária, demonstra-se que, quando apenas uma firma altera seu comportamento, para um caso envolvendo mais de duas firmas, a maximização é vantajosa, mas, se mais de uma empresa altera seu comportamento e acabam utilizando estratégias “opostas”, a estratégia de maximização dos lucros, deixa de ser dominante.

Segundo Arifovic (1994), quando considera-se o aspecto cronológico, ou seja, a progressão do tempo pelo avanço dos períodos, deve-se considerar o conhecimento adquirido pelos agentes nas fases anteriores, este é o contexto da aprendizagem, a utilização da experiência prévia (conhecimento) para guiar as próximas escolhas. Essa aprendizagem, é classificada por Vriend (2000) em duas formas básicas, a individual e a populacional/social. A primeira consiste no conjunto de informações obtidas de forma independente, através de por exemplo, métodos de tentativa e erro ou conhecimento prévio. Já a aprendizagem social consiste na transmissão de comportamentos e características de um indivíduo para o outro, uma forma é através da imitação. Segundo o autor as duas abordagens são relevantes, logo, devem ser consideradas em conjunto, pois podem, inclusive, levar a resultados diferentes quando consideradas de forma isolada. Essa diferença entre os resultados é devido ao “*spite effect*”.

Uma forma de aprendizagem, desenvolvida por Holland (1975), é a utilização de algoritmos genéticos, uma linha dos algoritmos evolucionários. Os AG são fundamentados nos princípios da biologia evolutiva de seleção natural, no qual o meio seleciona os seres mais bem adaptados, que passam os genes que contém essas características para a sua prole, dando continuidade ao ciclo de evolução das espécies. Ao se utilizar Algoritmos Genéticos, procura-se, a partir de um conjunto inicial de estratégias, obter soluções aproximadas

para problemas de otimização, essas escolhas iniciais são combinadas e/ou modificadas de forma a se buscar novas e melhores estratégias, dando origem a uma nova geração, formando dessa forma um processo iterativo e, conseqüentemente, evolutivo.

Os AG podem ser aplicados, por exemplo, a um problema de otimização das firmas, no qual a informação é limitada ou inexistente, logo, não se sabe a forma mais eficiente de investimento, pode-se, então, utilizar AG para buscar estratégias que levam a melhores resultados. Os AG simularão a situação em que os agentes, ou seja, as firmas, possuem pouca ou nenhuma informação a respeito do mercado, as estratégias são formadas a partir da observação dos resultados relacionados a um conjunto de ações inicialmente aplicado. Portanto, é importante compreender dois aspectos, o primeiro é a dinâmica industrial, que consiste na forma como as firmas entram e saem do mercado, a quantia que investem e como esse investimento é feito, podendo ter como foco inovação ou imitação. O segundo aspecto que deve ser considerado, é a aprendizagem, que consiste na construção de uma base de conhecimento, que vai “guiar” as escolhas dessas firmas.

A aprendizagem consiste na acumulação de conhecimento pelos agentes no mercado, considera-se que os agentes estão sempre em busca do aprimoramento, pois eles possuem um conjunto limitado de informação que, a partir da experiência, vai evoluindo e permite, esses agentes, tomarem novas decisões e aproveitarem novas oportunidades. As formas de aprendizado são diversas, por exemplo, uma firma pode aprender ao imitar uma concorrente, ou, a partir do investimento em Pesquisa e Desenvolvimento, podem buscar novas tecnologias e a inovação dos seus processos. As firmas podem, também, criar memorizar quais estratégias deram certo no passado, identificando características do mercado, como as peculiaridades dos seus concorrentes e consumidores. O objetivo da aprendizagem é a criação de uma espécie de banco de dados, que possibilite uma vantagem competitiva, aumentando não só suas probabilidades de sobrevivência no mercado, como, ainda, suas perspectivas de crescimento, tanto na lucratividade, quanto na participação no mercado.

Utilizando os AG como forma de aprendizagem, tem-se de um conjunto inicial de estratégias, é realizado um processo iterativo que permite a busca por novas e melhores estratégias a partir de um experimento empírico, que resulta na manutenção e aprimoramento das características, levando a resultados mais satisfatórios.

O estudo de [Arifovic \(1994\)](#) compara diferentes métodos de aprendizagem, a adaptação das estratégias partir da utilização de informações adquiridas ao longo do tempo. O autor faz uso de dois AG, um básico e o outro aumentado, depois, aplica-os para o caso do modelo teia de aranha e compara com outras metodologias que utilizam aprendizagem com o intuito de se verificar divergências resultantes da aplicação dos AG. Foram realizadas análises tanto para o caso de uma única população, quanto para o

caso de múltiplas populações, com o intuito de se verificar as convergências dos preços esperados e das quantidades produzidas, com as previstas pela expectativa racional e pelo equilíbrio do modelo.

Os dois AG utilizados possuem três operadores genéticos: *seleção*, “*crossover*” (*cruzamento*), *mutação*. A *seleção* consiste em se fazer cópias de cromossomos que possuem melhor ajuste, pois esses possuem uma maior probabilidade de gerarem um cromossomo que contribua para a evolução genética. O *crossover* consiste em se “cruzar” dois dos cromossomos selecionados pela etapa anterior, trocando os valores binários da *string* de ambos cromossomos após determinada posição. Na etapa de *mutação* ocorre a troca aleatória dos valores binários da *string*, todo valor em uma posição tem uma probabilidade de ser trocado, independente das outras posições. No caso do algoritmo genético aumentado tem-se um quarto operador, a *eleição*, que consiste em se testar os cromossomos recém-gerados antes que eles se tornem membros da população. Isso faz com que apenas se selecione cromossomos com maior potencial de ajuste em relação aos seus cromossomos geradores, podendo então se selecionar, um, dois ou nenhum dos novos cromossomos para a população (ARIFOVIC, 1994).

Para os métodos que utilizaram aprendizagem, houve convergência para o caso de situação estável, onde a razão entre as inclinações das curvas de demanda e oferta são menores que um, e houve divergência para o caso instável, razão entre as inclinações maiores do que um, com exceção para o método que utiliza a média dos preços passados como previsão do preço atual, que, por sua vez, apresenta uma convergência suave. Já utilizando o AG básico, no qual não há a etapa de *eleição*, houve divergência do preço em relação à expectativa racional, esse resultado deve-se à etapa de *mutação* que acaba aumentando a variabilidade, impedindo assim a convergência, tanto para o caso de uma única população, quanto para populações múltiplas. Já para os AG aumentados houve convergência tanto para o caso de uma única, quanto de várias populações, assim como, para o caso estável e instável. Portanto, a etapa de *eleição* permite que se elimine os efeitos negativos da *mutação* (ARIFOVIC, 1994).

Por outro lado, o trabalho de Beckenbach (1999), afirma que muitos estudos consideram o AG como um otimizador da função alvo e, na maioria dos casos, são utilizados para encontrar o valor máximo de uma função unimodal invariante ao longo do tempo, o que implica na necessidade de se possuir um sistema de melhoria de performance exclusivo para este AG. Logo, isso vai contra a ideia do AG como um procedimento de busca adaptativo, que busca solucionar uma série de problemas de decisão nos quais se tem uma grande complexidade derivada das incertezas sobre as condições do ambiente. Assume-se, portanto, que essa situação pode ser representada por uma função multimodal que varia ao longo do tempo, sem que haja a interferência de observadores, que “guiam” o sistema para valores de máximo, pelo contrário, o AG representa a evolução das

habilidades de sobrevivência da população.

Como já mencionado os AG são uma tentativa de aplicar na economia conceitos biológicos relacionados à evolução das espécies, esta intenção ligada a uma visão semelhante sobre a complexidade dos sistemas, leva a uma afinidade natural entre AG e economia evolucionária. Os AG são utilizados como uma ferramenta de modelagem de uma população de agentes de decisão, que possuem conhecimento limitado e pressionam uns aos outros a inovar e imitar. Mas, os algoritmos genéticos são uma generalização que possui um viés biológico que precisa ser redefinido para ser aplicado a contextos econômicos. Existem diferenças fundamentais entre a evolução biológica e a evolução econômica, portanto cada operador do AG, seleção, recombinação e mutação devem ser entendidos, assim como suas inter-relações, no contexto econômico tanto em sua interpretação macro quanto na visão micro (BECKENBACH, 1999).

A aplicação dos AGs na economia pode ser potencializada com a utilização de parâmetros modificados para incorporar aspectos econômicos como, por exemplo, o tamanho da população e outras restrições financeiras, de custo e de viabilidade. A aplicação desses algoritmos deve ser limitada a situações de estruturas análogas e a micro interpretação dos algoritmos é mais plausível, sendo que, devido aos custos altos, sua utilização deve se limitar a atividades da cadeia principal de produção ou em situações extraordinárias de alto grau de incerteza (BECKENBACH, 1999).

A pesquisa de Shubik e Vriend (1999) utiliza programação dinâmica para relacionar abordagens de teoria dos jogos e simulações comportamentais a um mesmo problema de troca econômico. O experimento consiste na escolha de um modelo simples que permita encontrar apenas um equilíbrio e aplicar a um caso no qual as condições iniciais estejam longe deste, de forma a verificar se o valor teórico calculado através do princípio das expectativas racionais são atingidos para o caso das simulações comportamentais. Dessa forma, os autores buscam desconstruir ideias bem consolidadas de expectativas racionais. Concluem, no entanto, que caso as condições iniciais do sistema estejam longe do equilíbrio, mesmo sendo os agentes, bem informados, estes podem não atingir o equilíbrio.

A teoria econômica muitas vezes classifica o comportamento racional “*ex-ante*”, baseado no futuro, como bom e recomendado, enquanto o comportamento adaptativo e classificado como “*ex-post*”, ou seja, baseado no passado, é tratado como um mau comportamento. Mas, essas diferenças não são tão grandes assim, já que ao se buscar prever o futuro, devem ser feitas algumas suposições sobre características do passado permanecerem constantes. Logo, agentes que se baseiam em ações realizadas no passado podem aprender a se comportar como se fossem agentes baseados numa visão focada no futuro. Isso faz com que um Sistema de Classificação possa aprender a reconhecer bons padrões de ações, encontrando soluções implícitas para problemas de computação

dinâmica. Dessa forma, mesmo quando uma solução explícita não pode ser encontrada, um algoritmo adaptativo pode realizar os cálculos, de forma a encontrar uma solução, por exemplo, através de uma simulação (SHUBIK; VRIEND, 1999).

As simulações são de grande utilidade quando aplicadas à economia evolucionária, Kwaśnicki (1999), analisou diferentes abordagens de simulação para desenvolvimento econômico, comparando tanto os modelos baseados na tradição Schumpeteriana, na qual a busca pelo lucro acima da média do mercado através de inovações regem o crescimento, como por exemplo, o modelo Nelson e Winter (1982), quanto as abordagens baseadas no agente. Os modelos baseados no agente, não possuem todos os parâmetros completamente explicitados, apenas existe um determinado conjunto de estratégias, que permite aos agentes realizarem “escolhas “ de forma a buscar sua sobrevivência. Quando aplicados economicamente, esses modelos atuam de forma semelhante, os agentes iniciam suas interações sem conhecimento prévio e aos poucos vão adquirindo conhecimento a partir da aprendizagem. Segundo Kwaśnicki (1999), os modelos Schumpeterianos possuem a vantagem de conseguirem relacionar o tempo de simulação com o tempo real, enquanto nos modelos baseados no agente, esse entendimento da relação dinâmica das alterações/tempo, é mais difícil. Logo, o autor sugere a interação entre essas duas vertentes de modelagem.

O estudo de Vriend (2000) chega à conclusão que para os AG de aprendizagem social, a produção das firmas converge para a produção Walrasiana, já no AG de aprendizagem individual, o “spite effect”, apesar de presente, não afeta de forma relevante o processo de aprendizagem. Tanto as firmas que utilizam um algoritmo, quanto as que utilizam o outro, buscam apenas aumentar sua recompensa absoluta, a diferença é que o aprendizado de cada uma delas está fundamentado em conjuntos de observações diversos. Isso implica na relevância da escolha da forma de aprendizado do AG, deve-se ter atenção e um maior cuidado ao se especificar e escolher entre a aprendizagem individual ou social, pois essa decisão pode ter um impacto relevante nos resultados gerados.

Buscando comparar os efeitos da aprendizagem via algoritmos genéticos com o investimento em P&D seguindo as regras fixas, Yildizoglu (2002) utilizou as duas estratégias em um modelo evolucionário simplificado de Nelson e Winter (1982), procurando verificar seus efeitos no bem-estar e na competitividade das firmas. Dessa forma, foi analisada a relevância da utilização dos algoritmos genéticos.

Em cada período as firmas que utilizam regras fixas de investimento, independentemente do comportamento do mercado, não modificam o método de investimento, investindo uma quantia mínima em P&D acrescida de uma proporção fixa do seu lucro. Já que esta forma de investimento é utilizada em muitos modelos evolucionários aplicados à indústria. Já para as firmas que utilizam o AG e que, portanto, têm na aprendizagem a justificativa para seu comportamento, lançam mão de estratégias de investimento em

P&D variáveis e dependentes das condições da indústria (YILDIZOGLU, 2002).

Nos AG, um agente possui um conjunto de estratégias, sendo cada estratégia definida como um cromossomo, que possui a mesma dimensão, ou seja, o mesmo número de genes que ao longo da vida do agente, vão sendo modificados (evoluem) como consequência da experiência e dos resultados obtidos pelo agente. Em seu trabalho, Yildizoglu (2002), utiliza os AG como forma de definir o percentual do lucro que será investido em P&D inovativo ao fim cada geração, período no qual o conjunto de estratégias foi utilizado. Os novos cromossomos dependerão da dinâmica industrial na qual a firma está inserida, portanto, o comportamento de cada um desses agentes depende das ações dos seus concorrentes. Isso faz com que as mudanças de estratégias não sejam um investimento que necessariamente tem retorno imediato, por isso, há a necessidade de serem testadas. Este teste pode ser realizado utilizando o conjunto de estratégias por vários períodos, para então verificar seu efeito na margem de lucro bruto e, assim, avaliar o ajuste dessa nova regra de investimento em relação às demais.

Verifica-se que presença de firmas que utilizam um método de aprendizagem levam, no longo prazo, a uma maior concentração de capital, a maiores lucros e a um menor preço do produto no mercado, o que implica em um maior bem-estar social. Mas, como os custos de se investir em aprendizagem são altos, se as firmas que utilizam metodologias de aprendizagem são poucas, essas podem ser levadas à saída do mercado por conta dos “erros” de investimento nos períodos iniciais. Além disso, mesmo quando as firmas que utilizam regras fixas investem proporções maiores do seu lucro, a atividade de P&D é dominada pelas firmas que utilizam o AG. Logo, a aprendizagem não deve ser ignorada nos modelos industriais e nos mercados, que contêm firmas que utilizam métodos de aprendizagem, têm-se uma maior eficiência tecnológica e social (YILDIZOGLU, 2002).

O trabalho de Yildizoglu (2002) encontrou convergência dos parâmetros para as firmas que utilizavam o algoritmo genético, com diminuição da dispersão ao longo do tempo, reduzindo assim a variabilidade dos resultados e a convergência em direção a determinado valor. Mas, França (2014) ao utilizar o método proposto por Yildizoglu (2002), encontra resultados contrastantes já que um dos seus AG apresentou uma redução da dispersão ao longo do tempo e, conseqüente, redução na variabilidade, levando a uma convergência, enquanto o outro dos AG utilizados, não apresentou convergência, sendo observado um aumento da variabilidade dos resultados, que acarretou em uma maior dispersão, não ocorrendo assim a convergência.

Tendo em vista que estudos semelhantes aplicando algoritmos genéticos a problemas econômicos levaram a resultados diferentes, propõe-se implementar um modelo industrial evolucionário como proposto por Nelson e Winter (1982), adicionando alguns conceitos e ajustes implementados por Yildizoglu (2002) e realizar a comparação dos resultados com

uma situação na qual estão presentes, além das firmas que utilizam AG, firmas que fazem uso de um processo aleatório de investimento, buscando dessa forma verificar e validar a aprendizagem das firmas via AG, além de se comparar o efeito do investimento externo por meio de crédito e da duração dos períodos de aprendizagem.

O objetivo principal deste trabalho é verificar o efeito da implementação de firmas que utilizam algoritmos genéticos em um modelo industrial evolucionário. Já entre os objetivos específicos, busca-se desenvolver um algoritmo genético, descrever o modelo utilizando “Protocolo ODD “ (*“Overview”*, *“Design concept”*, *“Details”*), simular em NetLogo o modelo, verificar a existência de aprendizagem pelas firmas que utilizam algoritmo genético, assim como a convergência das estratégias.

Este trabalho é relevante para o maior entendimento da teoria econômica, devido ao grande potencial dos Algoritmos Genéticos e de outros métodos de aprendizagem na previsão das tendências. Situações de grande volatilidade, como a variação cambial e os mercados financeiros, são de altíssima complexidade, possuem uma grande quantidade de parâmetros, que são impossíveis de serem, todos, explicitamente descritos em uma simulação. Mas, com a utilização dos AG pode ser possível, a partir de um conjunto inicial de regras “simples”, resolver de forma implícita esses problemas complexos. Portanto, é de grande interesse, entender e formular os AG de forma correta e eficiente, para que se tenha uma importante ferramenta na resolução de problemas, sejam eles, mais ou menos complexos.

Esta dissertação é dividida em seis capítulos, sendo o Capítulo 2 uma revisão bibliográfica dos Algoritmos Genéticos, o Capítulo 3 consiste na descrição do modelo, o Capítulo 4 apresenta um teste de convergência para o algoritmo genético e os resultados para o modelo são apresentados e discutidos no Capítulo 5. Apresenta-se as conclusões e considerações finais no Capítulo 6.

2 - Algoritmos Genéticos

Os algoritmos genéticos são uma linha dos algoritmos evolucionários que permite a utilização da aprendizagem, obtida através da experiência, nas gerações seguintes. Basicamente, novas estratégias são geradas a partir de estratégias anteriores, de forma a se encontrar soluções que apresentam um melhor ajuste (“*fitness*”), ou seja, define-se a função ajuste e busca-se otimizá-la através dos algoritmos. De acordo com [Mitchell \(1998\)](#), existem na literatura diversos casos em que os Algoritmos Genéticos são aplicados com sucesso, mas também casos em que o uso de AG leva a resultados fracos, isto se deve ao fato de que nem sempre utilizar um AG é uma opção competitiva ou a melhor opção. Desta forma, os pesquisadores normalmente aceitam que, em situações onde o espaço de buscas é extenso, não é suave e unimodal, não é bem conhecido, cuja função de ajuste possui muito ruído, ou, se procedimento não exige que um máximo global seja encontrado, então a chance da utilização do AG ser competitiva e, superar outros métodos de otimização, é maior. Mas, de toda forma, a chave para o sucesso dos AG está na codificação das possíveis soluções, dos operadores, das configurações dos parâmetros e da definição do critério de sucesso.

Segundo [Mitchell \(1998\)](#) existem três formas mais comuns de codificação para os algoritmos genéticos, a primeira e mais tradicional é a codificação binária, que consiste em se converter cada solução de seu conjunto de possíveis soluções em uma “*string*” com determinado comprimento contendo valores 0 ou 1. Além dessa forma existe a codificação utilizando vários valores ou muitos caracteres, nestes casos a conversão e transformação em variáveis binárias pode ser complexa e, portanto os cromossomos são representados por caracteres e números, diversos autores encontraram melhores resultados para esta forma de codificação em determinado problema. A terceira forma mais utilizada, é a codificação em árvore, onde cada estratégia é definida como um diagrama em forma de árvore, e cada “galho” pode ser trocado por outro “galho” de uma segunda “árvore”, este mecanismo possui um espaço de buscas aberto, pois as “árvores” podem possuir qualquer tamanho, este tipo de codificação é muito utilizado no desenvolvimento de programas.

Um algoritmo genético consiste basicamente de três fases, uma etapa de seleção dos indivíduos que se relacionarão, uma etapa de cruzamento (“*crossover*”), na qual serão gerados novos indivíduos a partir dos anteriores. E uma etapa de mutação, na qual características desses indivíduos são modificadas de forma aleatória, com o intuito de se obter uma maior variabilidade do espaço de busca, permitindo que valores com maior “*fitness*”, mas que não estejam tão próximos no espaço de busca, possam ser encontrados.

Para a situação em que temos uma codificação binária, pode-se ter, por exemplo,

um grupo de indivíduos no qual cada um possui seu respectivo conjunto de estratégias, são representadas por uma “string” binária, cada uma dessas “strings”, representam um cromossomo. A soma total dessas ‘strings’ resultam na quantidade de estratégias que serão adotadas a cada geração, pelo indivíduo. Cada um desses cromossomos possui genes, no caso, os elementos da “string” binária, o número total de genes de um cromossomo é o total de elementos da “string”. Para a codificação binária, cada gene é representado por um valor 0 ou 1, logo para os casos nos quais cada cromossomo possui n genes, são 2^n possibilidades de estratégias (MITCHELL, 1998).

Ao fim de cada geração, as estratégias são avaliadas e utilizando uma forma de seleção, são selecionados os cromossomos que irão se reproduzir, ou seja que trocam genes entre si gerando novos cromossomos. Estes novos cromossomos sofrem, ainda, uma etapa de mutação onde determinados genes são alterados, por exemplo, um gene 0 que sofre mutação, é modificado para 1 e vice-versa. Cada uma destas etapas possui diferentes abordagens na literatura.

2.1 - Métodos de Seleção

De acordo com Mitchell (1998), o próximo passo após a decisão da codificação a ser utilizada é definir quais indivíduos da população serão selecionados para gerar filhos e quantos filhos cada um gerará. Destaca-se alguns métodos comuns de seleção:

2.1.1 - “Roulette Wheel”

Segundo (MITCHELL, 1998), o método de roleta (“Roulette Wheel”) consiste na atribuição a cada um dos indivíduos de um pedaço da roleta, cujo tamanho, é proporcional ao seu “fitness”. A cada rodada essa roleta é “girada” e um indivíduo selecionado para compor a população de genitores, a roleta é girada N vezes onde N é o número de cromossomos na população. Uma forma de implementar este método é:

1º Passo - Obter o valor T que representa a soma de todos os “fitness” (valores esperados) para todos os indivíduos.

2º Passo - Sortear um número aleatório r , entre 0 e T .

3º Passo - Somar de forma ordenada(ordem que será mantida para os N indivíduos selecionados) cada um dos valores esperados até o primeiro valor maior ou igual a r .

4º Passo - Selecionar o indivíduo responsável por se atingir ou superar o valor aleatório r para pertencer à população de genitores.

5º Passo - Repetir os passos 3 e 4 até que se obtenha N indivíduos.

Um dos problemas da seleção por “Roulette Wheel” é o fato da probabilidade de

seleção de determinado cromossomo ser proporcional ao “*fitness*”. Pois, para as situações onde existem cromossomos com “*fitness*” muito maiores que o dos outros, existe a possibilidade de que haja uma vantagem excessiva para este cromossomo. Portanto, existe uma tendência à seleção dos cromossomos que inicialmente possuem um “*fitness*” muito elevado, aumentando, desta forma, a pressão de seleção. Uma forma de se reduzir esse efeito, é adicionar a todos os valores de “*fitness*” determinado valor, fazendo com que se reduza a razão entre esses valores de ajuste e, desta forma, se tenha uma redução na pressão de seleção (HANCOCK, 1994).

2.1.2 - “*Sigma Scaling*”

O método “*Sigma Scaling*” busca moderar a pressão de seleção ao longo do tempo, de forma que a mesma não seja muito intensa nas gerações iniciais (evitando a convergência precoce) nem muito fraca nas gerações finais (quando a população estabilizou e a diferença entre os ajustes é pequena). No método de “*Sigma Scaling*” o valor esperado para o indivíduo é função de seu “*fitness*”, da média populacional e do desvio padrão populacional. Dessa forma, procura-se ter uma pressão de seleção relativamente constante ao longo de toda a simulação (MITCHELL, 1998).

De acordo com Hancock (1994), o método de “*sigma scaling*”, ao utilizar as técnicas de linhas de base móveis, faz as strings, com valores abaixo do desvio padrão, possuírem “*fitness*” nulo, o que ajuda a impedir que a procura por melhores soluções pare, mas por outro lado, este método aumenta a possibilidade de convergência prematura para um indivíduo com o “*fitness*” grande. Pois, o método destaca sua vantagem em relação à média.

Um exemplo de sigma scalling é citado por Tanese(1989, apud Mitchell (1998)):

$$ExpVal(i, t) = \begin{cases} 1 + \frac{f(i) - f(t)}{2\sigma(t)} & se \sigma(t) \neq 0 \\ 1.0 & se \sigma(t) = 0, \end{cases} \quad (1)$$

Tem-se que, $ExpVal(i, t)$, é o valor esperado para o número de filhos por indivíduo, i , em um determinado tempo, t , $f(i)$ é o valor do “*fitness*” para o indivíduo, e $f(t)$ é o valor de “*fitness*” para a população no tempo t . Neste exemplo, um indivíduo com um “*fitness*” equivalente a um desvio padrão acima da média resulta em um valor esperado de 1.5 filho para a próxima geração. Já para o caso de valores negativos, tem-se que o valor esperado para o número de filhos é arbitrariamente resetado para 0.1, desta forma, garante-se que mesmo indivíduos com valores de “*fitness*” muito baixos possuem alguma probabilidade de serem escolhidos.

No início da simulação quando, normalmente, os desvios padrões dos “*fitness*” são

altos, os indivíduos com maior “*fitness*”, não estarão muitos desvios padrões acima da média, e logo não resultam uma parcela muito grande dos filhos. Já quando a simulação está em fases mais avançadas e a população, geralmente, apresenta uma maior convergência para determinado valor, os desvios padrões são tipicamente menores. Este é a razão que faz os indivíduos com maior “*fitness*” se destacarem, permitindo que a evolução continue a ocorrer (MITCHELL, 1998).

2.1.3 - Elitismo

Segundo Mitchell (1998), o elitismo consiste em manter um número determinado de cromossomos para a próxima geração, dessa forma evita-se que esses indivíduos sejam perdidos seja pelo operador de cruzamento, ou pela mutação. Assim, garante-se a permanência de determinadas características na população melhorando o desempenho do AG. Por exemplo, caso se deseje manter o indivíduo com melhor “*fitness*” para a próxima geração e, antes que ocorram as etapas de cruzamento e mutação, este indivíduo já é selecionado como um cromossomo da próxima geração. Na geração seguinte, será aferido, novos valores de “*fitness*” para todos os cromossomos, incluindo aquele da geração anterior. Nesta nova geração, será selecionado, mais uma vez o cromossomo, com maior “*fitness*” e que, portanto, já possui seu lugar garantido na próxima geração.

2.1.4 - Recombinação Elitista

De acordo com Thierens e Goldberg (1994), para a seleção por recombinação elitista consiste para cada casal de cromossomos, serem gerados dois descendentes, assim que são testados e os dois cromossomos, levando em conta as quatro estratégias, tem seus “*fitness*” comparados, aqueles dois com o maior “*fitness*” avançam para a próxima geração. Para realizar a seleção por recombinação elitista deve-se realizar o seguinte procedimento:

1º Passo - Inicialização da população

2º Passo - De forma aleatória selecionar os pares de pais.

3º Passo - Gerar filhos a partir dos pares formados.

4º Passo - Avaliar o “*fitness*” dos filhos.

5º Passo - Manter apenas os dois cromossomos de cada família (dois cromossomos pais e dois cromossomos filhos) com maior “*fitness*” como membros da próxima geração.

6º Passo - Avançar para a próxima geração, começando do 2º Passo e repetir até que o número de gerações seja considerado suficiente.

2.1.5 - Seleção Boltzmann

Enquanto a seleção por Sigma Scaling busca manter a pressão de seleção constante, o método de seleção Boltzmann consiste em atribuir pressões diferentes ao longo da simulação de acordo com a necessidade. Por exemplo, é possível no início da simulação atribuir pressões de seleção menores de forma a manter a seleção lenta permitindo que se mantenha grande a variabilidade da população, ao passo que, a medida que a simulação ocorre, a pressão vai aumentando gradativamente, de forma a tornar a variabilidade menor e levando à convergência para as situações de melhores “*fitness*”. Dessa forma, aumenta-se a probabilidade de que seja encontrada a região de busca correta, próxima ao ponto de máximo global, uma típica implementação do método é (MITCHELL, 1998):

O método de Boltzmann Selection foi utilizado por Mahfoud e Goldberg (1995) para o caso de recozimento simulado, já que, segundo os autores, o caso estudado segue uma distribuição de Boltzmann e Maza e Tidor (1993) em seu trabalho comparou o método de seleção de Boltzmann com a seleção proporcional e concluiu que o método de Boltzmann para seleção apresentou convergência para o valor correto em uma quantidade de gerações menor que o método de seleção proporcional ao fitness, portanto para este experimento, houve uma maior eficiência para o método de Boltzmann.

Logo, o método de Seleção Boltzmann consiste em aplicar menores diferenças de “*fitness*” para os períodos iniciais e ir aumentando aos poucos essas diferenças, o objetivo é que isso aconteça de forma gradual ao longo da simulação, evitando uma pressão de seleção muito elevada nos períodos iniciais.

2.1.6 - Seleção por Rank

A seleção por Rank consiste em ranquear as estratégias de acordo com seu fitness, independentemente de seus valores absolutos, isso evita que a convergência ocorra de forma muito rápida, pois mesmo para diferenças consideráveis nos valores de “*fitness*”, a nova prole não será advinda apenas de uma minoria de pais, pois se garante que mesmo os valores pequenos possuirão probabilidade de serem escolhidos (MITCHELL, 1998).

Para o caso da seleção por “*rank*”, Blickle e Thiele (1996) cita a seleção por “*rank*” linear e exponencial, para o primeiro caso, as estratégias ordenadas de acordo com o fitness, tem sua probabilidade de serem escolhidas como pais, seguindo uma proporção linear, enquanto que para o segundo caso, essas probabilidades segue uma ordem exponencial. O autor salienta que a seleção por rank “exponencial”, é aquela que possui a maior variância, quando comparada com a “rank linear”, “tournament selection” e “truncation selection”, levando-se em conta a suposição de que uma maior variância é desejável, logo a exponencial rank seria a melhor opção de seleção.

2.1.7 - Seleção por Torneio

A seleção por torneio consiste em selecionar a partir da população uma série de indivíduos dos quais se afere o “*fitness*”, em seguida são atribuídas probabilidade entre 0 a 1 para cada um desses indivíduos e gera-se um valor aleatório para que sejam decididos quais desses indivíduos serão os genitores da próxima geração (MITCHELL, 1998).

Em seu estudo Miller e Goldberg (1995) afirma que entre as vantagens da seleção por torneio, destaca-se a pressão de seleção ajustável, por exemplo um torneio escolhendo um menor número de candidatos leva a uma diminuição na pressão de seleção, já um torneio com uma maior quantidade de candidatos leva a uma pressão de seleção maior.

2.1.8 - Seleção por Estado Estacionário

Segundo Mitchell (1998), na seleção de estado estacionário são mantidos a maior parte dos indivíduos de cada geração, sendo que apenas uma minoria, normalmente os com menor ajuste, são substituídos por proles resultantes daqueles indivíduos que possuem maior “*fitness*”. Este método, portanto resulta em uma pressão de seleção pequena, já que apenas uma minoria dos cromossomos será substituída, sendo que a população de pais consiste na maioria dos cromossomos originais.

Em seu estudo, Rogers e Prügel-Bennett (1999), comparam a utilização da seleção por estado estacionário, na qual a maior parte dos cromossomos são mantidas para a próxima geração, com a uma seleção onde todos os cromossomos são modificados por novos cromossomos advindos dos cruzamentos entre os cromossomos originais. Os autores concluem que o método que mantém maior parte dos cromossomos originais possui uma taxa de redução da variância populacional igual ao dobro daquela referente ao outro método e ao mesmo tempo o algoritmo genético que utiliza a seleção por estado estacionário evolui a uma taxa duas vezes maior que quando comparada a uma geração do algoritmo que gera novos cromossomos a cada nova geração. Este fato, indica um favorecimento ao uso do método de estado estacionário.

2.1.9 - Seleção por Truncamento

Na seleção por truncamento, de acordo com Blickle e Thiele (1996), seleciona-se apenas uma fração, T , dos melhores cromossomos que serão selecionados para compor o conjunto de pais da próxima geração. Em seguida, esses pais serão selecionados com probabilidades iguais. Dessa forma a pressão de seleção, como no caso da seleção por torneio, também é ajustável. Já que este método de seleção depende do tamanho da fração, há uma redução na pressão de seleção quando se tem maiores valores de T , e há um aumento, da pressão de seleção, para menores frações. A vantagem deste método, é que ele reduz a possibilidade de uma convergência precoce, pois, os indivíduos selecionados como

possíveis pais para a próxima geração, possuem uma probabilidade igual de gerarem novos cromossomos e, dessa forma, passam suas características para os próximos períodos.

2.2 - Cruzamento

Na biologia, teorias relacionadas ao cruzamento de espécies, como demonstrado pelas “Leis de Mendel”, justificam a transmissão de determinadas características e o surgimento de outras a partir da reprodução entre dois indivíduos. Na reprodução sexuada, sempre há a troca de material genético, ou seja, o novo indivíduo é composto por alguns genes originados do lado materno e outros do lado paterno, formando um conjunto de características completamente novo.

Algumas características deste novo conjunto podem ser determinantes, por exemplo, para a não sobrevivência do indivíduo no caso da presença de genes letais, ou para a sobrevivência de outros, como no estudo das “Mariposas de Manchester”, clássico exemplo da seleção natural.

Na cidade de Manchester, Reino Unido, existiam, de uma mesma espécie, predominantemente mariposas de asas brancas, enquanto que as de asas pretas eram minoria, mas com a revolução industrial, veio o aumento da fuligem e poluição, o que fez com que as mariposas de asas brancas virassem um alvo mais fácil para predadores. Logo, a população daquelas de asas pretas se expandiu e tornou-se predominante (KETTLEWELL, 1955).

Este exemplo demonstra como determinadas características, passadas de geração para geração, podem se tornar relevantes, caso já não o sejam, devido a fatores que alteram o ambiente e isto, acaba ocorrendo também nos algoritmos genéticos, que segundo Holland (1992), são programas que, como a seleção natural, evoluem, muitas vezes, de forma não prevista nem pelos seus programadores.

O cruzamento nos AG, é um operador genético que possibilita a evolução através da mistura de características de dois indivíduos (cromossomos). De acordo com Mitchell (1998), o cruzamento consiste em utilizar partes do material genético de um cromossomo acrescida de genes de outro. No caso de um cruzamento de único ponto, cujo exemplo pode ser observado na Tabela 1 apenas uma parte de cada indivíduo é selecionada, no exemplo de indivíduos de 7 genes, podem ser selecionados 3 genes de um, 4 do outro e vice-versa para se formar os dois novos indivíduos.

Tabela 1 – Exemplo de crossover em um único ponto.

Cromossomos Pais	Ponto de Crossover	Nova Geração
1 0 1 0 0 1 1	1 0 1 0 0 1 1	1 0 1 0 1 0 0
0 1 1 0 1 0 0	0 1 1 0 1 0 0	0 1 1 0 0 1 1

Um dos problemas do método de cruzamento em um único ponto é que a sequência dos genes escolhida não se altera, o que gera uma limitação já que não é possível por exemplo pegar os primeiros dois genes do cromossomo 1, três do cromossomo 2 e os últimos dois, mais uma vez, do cromossomo 1. Sempre no caso de um cruzamento entre 1 e 2 o último gene seria do cromossomo 2, dando origem ao chamado viés “*endpoint*” (viés do ponto final em tradução livre). Para eliminar esse viés, pode-se utilizar o cruzamento utilizando dois pontos, Tabela 2, no qual utiliza-se parte de um cromossomo 1, parte de um cromossomo 2 e finaliza-se o cruzamento ao se adicionar uma nova parte do cromossomo 1 (MITCHELL, 1998).

Tabela 2 – Exemplo de crossover em dois pontos.

Cromossomos Pais	Pontos de Crossover	Nova Geração
1 0 1 0 0 1 1	1 0 1 0 0 1 1	1 0 1 0 1 0 0
0 1 1 0 1 0 0	0 1 1 0 1 0 0	0 1 1 0 0 1 1

2.3 - Mutação

As mutações em termos biológicos, são alterações no genótipo de um indivíduo, ou seja, este passa a ter genes que não tem sua origem nos seus ascendentes. Estas novas características são obtidas a partir da falha na reprodução desses genes, por exemplo em uma falha de replicação da fita de DNA, e, os indivíduos passam a possuir determinadas características genéticas que podem ser mantidas para as futuras gerações, modificando ao longo prazo determinado aspecto da espécie.

Os parâmetros que regem as mutações, ainda não são completamente explicados, mas é aceito que algumas situações aumentam a incidência de mutação, por exemplo o contato com Raios-X. Nos algoritmos genéticos, essa incidência de mutação, é considerada como a probabilidade de um gene se modificar, ou a de um cromossomo ter um gene alterado. A mutação possui um papel fundamental pois possibilita, por exemplo, que outras características ou estratégias, que não estavam sendo utilizadas, pois o material genético das gerações anteriores não continham esses aspectos, possam ser adquiridas pelo indivíduo e, talvez, se mostrarem úteis na evolução e busca por melhores soluções.

O operador genético da mutação é, portanto, fundamental para o aumento da variabilidade dos indivíduos e evita, por exemplo, a convergência destes para uma solução não otimizada. Mas, se por um lado, a mutação é importante para que se aumente o universo dos cromossomos testados, aumentando assim o espaço de busca, deve-se também, levar em conta, que maiores probabilidades de mutação implicam em uma maior variabilidade da amostra e que conseqüentemente dificultaria a convergência para a melhor solução (MITCHELL, 1998).

Na prática, este operador consiste na modificação dos genes de um indivíduo, por exemplo em uma “*string*” com codificação binária, a etapa de mutação consistiria em uma probabilidade desses genes serem alterados, ou seja um gene 0, ao sofrer mutação, se torna um gene 1 e vice-versa. Na literatura encontra-se casos de aplicação nos algoritmos genéticos, nos quais a mutação é aplicada como uma probabilidade de cada gene ser modificado de forma independente, mas também casos em que cada indivíduo é testado de forma a verificar se este sofrerá mutação, por exemplo, no caso de sucesso do evento estocástico, um gene aleatório, deste indivíduo, será modificado resultando, em um indivíduo diferente do anterior, caso o evento não o cromossomo permanece o mesmo para a próxima geração. Exemplos de mutação dos cromossomos podem ser observados na Tabela 3.

Tabela 3 – Exemplo de Mutação em cromossomos.

Cromossomos	Genes em Mutação	Novos Cromossomos
1 0 1 0 0 1 1	1 0 1 0 0 1 1	1 0 0 0 0 1 1
0 1 1 0 1 0 0	0 1 1 0 1 0 0	0 0 1 0 1 1 0

3 - Descrição do Modelo

Este trabalho consiste na simulação de um modelo evolucionário baseado no modelo de [Nelson e Winter \(1982\)](#), acrescido da ideia proposta por [Yildizoglu \(2002\)](#) de se acrescentar firmas que possuem um certo nível de inteligência. Estas firmas utilizam determinadas regras para escolherem seu investimento no futuro. Compara-se, portanto, o desempenho das firmas que utilizam o AG com aquele de firmas que lançam mão das estratégias fixas de investimento em P&D, como proposto no trabalho original. Será realizada também, uma comparação entre esta indústria, com outra na qual coexistem firmas que utilizam o método de aprendizagem, com agentes que possuem inteligência zero, ou seja, firmas que definem seus investimentos em P&D de forma aleatória ao fim de cada período.

O procedimento metodológico consiste primeiro na definição do algoritmo genético a ser utilizado, o qual é testado para problemas de otimização com solução simples, como forma de verificar sua eficácia. Em seguida, este algoritmo é aplicado ao modelo evolucionário de [Nelson e Winter \(1982\)](#), acrescido de alguns parâmetros e aspectos propostos e aplicados por [Yildizoglu \(2002\)](#).

Buscando facilitar a compreensão e reprodução posterior do modelo, utiliza-se o protocolo ODD (“*Overview*”, “*Design concept*”, “*Details*”), descrito por [Railsback e Grimm \(2012\)](#), para detalhar os modelos utilizados. O algoritmo é testado e, em seguida, é implementado em conjunto com modelo evolucionário, verifica-se a existência ou não de convergência e as consequências da aplicação do AG para a indústria como um todo, seus efeitos sociais e individuais. As simulações são realizadas utilizando o software NetLogo ([WILENSKY, 1999](#)).

3.1 - Modelo Evolucionário(Protocolo ODD):

Propósito: Verificar os efeitos na estrutura de mercado da introdução de um algoritmo genético em um modelo de organização industrial a partir da implementação do modelo evolucionário proposto por [Nelson e Winter \(1982\)](#) acrescido de firmas que utilizam Algoritmos Genéticos para definirem seu investimento em P&D inovativo, como proposto por [Yildizoglu \(2002\)](#). São analisadas duas situações diferentes, na primeira as firmas que utilizam AG são simuladas em conjunto com aquelas que fazem uso de estratégias fixas de investimento em P&D. Em um segundo momento, simula-se a situação em que firmas, as quais fazem uso dos AG, dividem o mercado com firmas sem inteligência, ou seja, que decidem seu investimento inovativo de forma aleatória. Procura-se comparar o

desempenho destas diferentes estratégias nos resultados da firma, verificando a relevância da utilização de Algoritmos Genéticos em um modelo evolucionário.

Entidades, variáveis de estado e escala de tempo: Temos três tipos de agentes(entidades), as firmas que utilizam o algoritmo genético(F.Gen), as firmas, F.NW, que utilizam o modelo de investimento descrito em Nelson e Winter (1982) e aquelas firmas sem inteligência, que ajustam seu investimento em inovação de forma aleatória a cada período(F.RND). No modelo proposto por Nelson e Winter (1982), a escala de tempo consiste em um período que representa um trimestre e a simulação dura por 100 períodos. É necessário ajustar esta escala de tempo para que se tenha intervalos suficientes para que os Algoritmos sejam testados, Yildizoglu (2002) fixa a duração das simulações em 6000 intervalos, portanto, propõe-se dividir cada trimestre do modelo original em 60 períodos, para manter um número de intervalos similar.

As firmas que utilizam algoritmos genéticos possuem como características: estratégias fixas de investimento em P&D imitativo, seguindo o valor de r_{in} proposto no trabalho de Nelson e Winter (1982), e estratégias variáveis de investimento em P&D inovativo, ou seja, seu r_{in} é alterado de acordo com o Algoritmo Genético utilizado.

Para as firmas que utilizam o modelo Nelson e Winter (1982), as características são: Estratégias fixas para ambos os investimentos em P&D, sendo que os percentuais do capital investidos em imitação e inovação são constantes, seguindo os valores de r_{in} e r_{im} , propostos no trabalho original, estas constantes variam de acordo com o número total de firmas.

Já as firmas que utilizam o investimento em inovação aleatório, são caracterizadas por: Estratégia fixas de investimento em imitação e estratégias variáveis em inovação, ou seja, é estabelecido um fator multiplicador ao valor de r_{in} proposto no modelo original, e este novo r_{in} se torna o valor máximo do percentual de investimento que estas firmas podem investir, a partir daí são gerados números aleatórios a cada período que definem qual percentual será investido em P&D.

Todos os tipos de firmas possuem como variáveis: O estoque de capital(K) e a produtividade(A). No caso das firmas que utilizam algoritmos genéticos e daquelas que usam a aleatoriedade, o percentual do capital investido em P&D inovativo, r_{in} , também é variável.

No modelo Nelson e Winter (1982), as simulações duram 100 períodos, onde cada um desses intervalos de tempo representa um trimestre. Portanto, é necessário ajustar estes intervalos para que haja tempo suficiente para que, os cromossomos(estratégias) dos Algoritmos, sejam testados e o processo iterativo/evolutivo ocorra. Este, ocorre através da repetição por diversas gerações das etapas de cruzamento e mutação, possibilitando, assim, o desenvolvimento do processo evolucionário. Em seu trabalho, Yildizoglu (2002)

fixa a duração das simulações em 6000 intervalos, portanto, seguindo esta orientação, é necessário dividir cada trimestre do modelo original em 60 períodos, para obter uma duração similar. São realizadas 20 simulações para cada situação.

Visão Geral do Processo e Planejamento: Após a inicialização dos procedimentos, os agentes aplicam um percentual de seu capital em P&D, utilizando suas estratégias pré-definidas, fixas, aleatórias ou pelo AG. As firmas, que utilizam estratégias fixas de investimento, aplicam, ao final de cada intervalo de tempo, uma porcentagem fixa de seu capital em pesquisa buscando a inovação/imitação das melhores práticas do mercado. Possuem o intuito de melhorar sua produtividade e aumentar seus lucros nos próximos períodos.

Já as firmas, que utilizam algoritmos genéticos, aplicam uma porcentagem fixa de seu capital como investimento em imitação, esta, é igual a taxa das firmas F.NW. Mas, seu investimento em inovação é definido a partir de um conjunto inicial de estratégias(cromossomos), que é atualizado de acordo com um fator de “*fitness*”. O parâmetro de ajuste é definido, neste modelo, como a média da taxa de lucro dos períodos nos quais a estratégia foi utilizada. Os cromossomos são atualizados ao final de cada geração(após todo o conjunto de estratégias ser testado), logo, a decisão de quanto se investe em P&D se altera ao longo do tempo.

As firmas que utilizam estratégias aleatórias, mantêm fixo o percentual de investimento imitativo e toma decisões de investimento em inovação, a cada período, de forma aleatória. Ou seja, são sorteados os percentuais de capital, que serão investidos, em P&D inovativo, a cada rodada, sendo que máximo é igual a cinco vezes o valor que as firmas de estratégias fixas investem.

Ao fim de cada intervalo, as firmas que utilizam os AG podem alterar suas estratégias, já as firmas NW vão utilizar a mesma estratégia em todos os períodos. Diferese do trabalho desenvolvido por Yildizoglu (2002), a não previsão, como no modelo original, da saída de firmas do mercado. Outra diferença é que são realizadas simulações utilizando diversas durações para o período de aprendizagem, intervalo total no qual as mesmas estratégias são utilizadas e, a taxa de lucro média para cada decisão, aferida. Dessa forma, procura-se verificar o efeito desta duração no desempenho das firmas e na estrutura do mercado.

Conceitos de Projeto (“*Design concepts*”):

Princípios básicos: É gerado um número pré-definido de firmas, 32, as quais utilizam um percentual do seu capital para se investir em P&D inovativo e imitativo, r_{in} e r_{im} , respectivamente. Em cada período, as firmas tentam inovar(gerar uma produtividade melhor que a anterior) ou imitar(copiar a melhor produtividade do mercado). Ao fim de cada intervalo de tempo, seleciona-se a maior produtividade entre, aquela obtida através

do investimento em pesquisa e a produtividade do período anterior. A produtividade do período seguinte será definida como o maior valor entre as três.

Um percentual de firmas utiliza as estratégias de investimento em P&D inovativo fixas(F.NW) e o restante das firmas, as F.Gen, decidem a porcentagem do capital que será investida em P&D inovativo a partir de um algoritmo genético. O valor máximo desse percentual é definido como um múltiplo, fixado em 5, do percentual fixo investido pelas F.NW. Esta é forma de limitar o percentual do capital que pode ser investido, evitando que as firmas comprometam percentuais exagerados do seu capital, causando distúrbios na estrutura do mercado. O algoritmo genético utilizado, busca selecionar estratégias, percentuais do capital investidos em inovação, r_{inf} , que resultam em maiores taxas de lucro para os próximos períodos.

O lucro de um período é utilizado para determinar o investimento no período seguinte. No caso de lucros positivos, a firma terá a disposição além do próprio lucro, um crédito bancário, **bank**, pré-definido em relação ao lucro(0, 1 ou 2.5). O investimento no estoque de capital do próximo período, $K_{i(t+1)}$ é definido de forma a ser sempre não negativo. Portanto, mesmo que o lucro seja negativo, o estoque de capital da firma será diminuído em percentual correspondente apenas, à taxa de depreciação, σ , pois, o investimento em capital, neste período, será nulo.

Os períodos, durante os quais uma estratégia é testada, são definidos por [Yildizoglu \(2002\)](#) como iguais a 5. No entanto, realiza-se, também, simulações para outros períodos de aprendizagem com intuito de se verificar o efeito do aumento e da diminuição desses períodos na estrutura de mercado

Para o universo onde comparamos firmas F.Gen com firmas F.NW, um percentual de firmas utiliza as estratégias de investimento fixas(F.NW) e o restante das firmas decidem a porcentagem do lucro que será investida em P&D inovativo a partir de um algoritmo genético(F.Gen). Para o caso do universo onde são comparadas as F.Gen com as F.RND, o percentual das firmas com estratégia de investimento fixas F.NW é substituído pelas firmas que ajustam seu investimento em inovação de forma aleatória, F.RND. O fator multiplicador do percentual, que limita o valor máximo do r_{in} , é mantido igual àquele utilizado pelas F.Gen.As proporções de firmas que utilizam algoritmos genéticos em cada um dos mercados foram:

- 0% de firmas F.Gen
- 25% de firmas F.Gen
- 50% de firmas F.Gen
- 75% de firmas F.Gen
- 100% de firmas F.Gen

Em seu trabalho, Yildizoglu (2002) aplica conceitos propostos por Winter (1984), no qual é proposto a entrada e saída de firmas do mercado, definindo que quando o lucro de uma firma tende recorrentemente a zero, seu estoque de capital irá diminuir devido à depreciação e, caso atinja um valor inferior ao estoque de capital mínimo pré-definido, tem-se a saída da mesma do mercado, pois a firma perde sua capacidade de investimento. Prefere-se, no entanto, ignorar este parâmetro, pois considera-se que ainda que sejam mantidas no mercado, uma firma com capital muito pequeno, possui uma parcela ínfima da produção e não altera as estruturas deste mercado. Difere-se ainda do trabalho de Yildizoglu (2002), a utilização dos percentuais de investimento em inovação e imitação similares ao do trabalho original, sem alterações, o que resulta em percentuais diferentes para P&D inovativo e imitativo. Além disso o autor estipula um percentual mínimo para ser investido em P&D, parâmetro que não é incluído neste trabalho.

Adaptação: As firmas que utilizam algoritmo genético, armazenam as informações dos resultados individuais daquela geração. Portanto, possuem certo nível de inteligência e suas decisões são pautadas por aprendizagem. O ajuste da estratégia de investimento é medido a partir da taxa de lucro médio da firma. As estratégias de investimento que resultam em maiores taxas de lucro são “mais vencedoras” e terão uma maior probabilidade de se tornarem pontos de partida para as novas estratégias que serão definidas pelo AG, através dos mecanismos de cruzamento e mutação.

Objetivos: Através do uso de algoritmos genéticos, otimizar o lucro a partir do melhor ajuste entre dispêndios em inovação, para aumento da produtividade, e a expansão do capital.

Aprendizagem: As firmas F.Gen fazem uso do conhecimento adquirido a partir dos resultados observados após a utilização de determinadas estratégias. Os cromossomos com melhores resultados possuem maiores chances de serem escolhidos nos próximos períodos (seleção por “roulette-wheel”) na etapa cruzamento, e, portanto, possuem uma maior probabilidade de terem seu material genético passado para a geração seguinte. A melhor estratégia é mantida imutável para a próxima geração, evitando assim, que devido aos operadores genéticos do algoritmo, se percam estratégias vencedoras (elitismo).

Previsões: As firmas que utilizam o algoritmo genético partem do pressuposto que as estratégias vencedoras do passado, ou seja, as que obtiveram melhores resultados, são candidatas mais promissoras a obterem mais sucesso no próximo período.

Detecção: Os agentes possuem acesso à evolução dos dados do seu lucro, podendo, dessa forma, inferir se as suas estratégias obtiveram sucesso. A “memória” de cada um dos agentes é definida como igual ao número de estratégias utilizadas em cada geração, Cr, são armazenados os dados da taxa de lucro média, obtida ao longo dos períodos nos quais cada uma das estratégias foi utilizada. Este conjunto de períodos é o intervalo/período de

aprendizagem..

Interação: Os agentes não interagem diretamente, mas quando têm sucesso na imitação, copiam a melhor produtividade entre os agentes do mercado, ou seja, obtêm a maior taxa de produtividade daquele período.

Estocasticidade: O investimento em P&D, no modelo, é dividido em inovação e imitação. O resultado, do investimento, é um evento aleatório, assim como, a eficiência da inovação. A inovação consiste em um processo, no qual o sucesso inovativo, do investimento em P&D, é determinado, para cada firma, por:

$$P[d_{in} = 1] = a_{in} \times r_{in} \times K_j \quad (2)$$

Na qual, para as firmas *F.NW*, $r_{in} = r_{in_{NW}}$, para as *F.Gen*, $r_{in} = r_{in_{NW}} \times rd_{ij} \times f$, onde f é o fator de multiplicação de $r_{in_{NW}}$ que limita o valor máximo de r_{in} e rd_{ij} , é o percentual do universo de busca referente ao cromossomo utilizado pelo AG. No caso das firmas *F.RND*, $r_{in} = r_{in_{NW}} \times RN \times f$, ou seja, a probabilidade de sucesso da inovação é proporcional ao fator multiplicador e a RN um número aleatório contido no intervalo $[0,1]$. O parâmetro de calibração, a_{in} , é definido como 0.125, de acordo com os valores de investimento em pesquisa do modelo proposto por Nelson e Winter (1982).

Caso ocorra o sucesso neste primeiro evento, um novo valor de produtividade é obtido através da geração de um número aleatório na distribuição log-normal, com média dependente do tempo:

$$\log(\tilde{A}jt) \longrightarrow N\left(A0 + (1 + \alpha)^{t/f_t}, \sigma\right) \quad (3)$$

Na qual, $A0$ é a produtividade inicial, constante e igual a 0.16, α é definido como 0.01 e sigma é 0.05, deve-se destacar que a relação exponencial ao tempo, dividido pelo fato de ajuste temporal, $f_t = 60$, é uma particularidade deste modelo, já que o trabalho de Nelson e Winter (1982) considera uma relação linear ao tempo.

Para a imitação, tem-se um novo evento estocástico, que no caso de sucesso, resulta na cópia da melhor prática do mercado, dessa forma, a probabilidade de sucesso deste evento, é definida por:

$$P[d_{imt} = 1] = a_{im} \times RD_{jt} \quad (4)$$

Ou seja, a probabilidade é proporcional à quantia investida em P&D multiplicada por um parâmetro de calibração, que faz com que, independentemente do valor investido, essa probabilidade seja inferior a 1. Escolhe-se este parâmetro como 1.25, de acordo com o

modelo de Nelson e Winter (1982). O resultado do processo de imitação é definido como:

$$\hat{A}_{jt} = A_{jt} + d_{imt} \times (A_t^* - A_{jt}) \quad (5)$$

No caso de fracasso do evento probabilístico, $d_{imt} = 0$, temos que a produtividade após o investimento em imitação é igual à produtividade do período anterior. Para o caso de sucesso, $d_{imt} = 1$, tem-se que a nova produtividade da firma, após o investimento em imitação, é igual à maior produtividade entre as firmas do mercado, A_t^* .

Além desses eventos estocásticos, no caso das firmas F.Gen, temos estocasticidade, tanto na ocorrência da etapa de cruzamento, $P[X] = 0.7$, quanto na etapa de mutação dos cromossomos, $P[M] = 0.03$ para cada gene.

Observação: Os indicadores de bem-estar social são o preço de mercado, taxa de lucro média e a concentração de capital. Indicadores de eficiência técnica: Produtividade média e máxima, estoque de capital médio e máximo para cada tipo de firma.

Inicialização: Dois tipos de agentes em diferentes proporções, firmas F.Gen e F.NW ou F.Gen e F.RND dependendo do caso escolhido, produzem um produto homogêneo e inicialmente possuem os mesmos valores de produtividade, capital e participação no mercado, esses agentes têm estratégias de investimento fixas ou variáveis e no início do primeiro período tomarão decisões de investimento diferentes.

Dados de Input: O ambiente é considerado invariável ao longo do tempo, logo não temos dados de *input*.

Submodelos: Cada firma produz o mesmo bem homogêneo seguindo a equação de produção:

$$Q_j = A_j \times K_j \quad (6)$$

Dessa forma, a taxa de lucro líquido de cada firma, π_j , é dada por:

$$\pi_j = pA_j - c - r_{im} - r_{in} \quad (7)$$

Em que c é o custo de capital, constante pré-definida, e p é o preço do produto no mercado, definido pelo seguinte equilíbrio:

$$Q = \sum_j Q_j \quad (8)$$

$$p = p(Q) = \frac{D}{Q^{1/\eta}} \quad (9)$$

Onde Q é a oferta total, D é a demanda, constante e pré-definida como 67, e η , é a elasticidade da demanda, considerada de acordo com o modelo original igual a 1. Após definido o preço pelo mercado, é possível calcular, a taxa de lucro, π_j que, após multiplicação pelo estoque de capital da firma, K_j , resulta no lucro líquido de cada firma, Π_j :

$$\Pi_j = \pi_j \times K_j \quad (10)$$

A produtividade efetiva das firmas, para o período seguinte, será dada pela seleção do maior valor de produtividade resultante dos eventos estocásticos:

$$A_{j,t+1} = \max(A_{jt}, \tilde{A}_{jt}, \hat{A}_{jt}) \quad (11)$$

Ou seja, compara-se a produtividade do período anterior, a produtividade após a inovação e a produtividade após o processo de imitação, selecionando a maior delas como a produtividade efetiva para o próximo período.

Firmas com estratégias fixas utilizam um percentual fixo do seu capital como investimento em inovação e imitação, de acordo com [Nelson e Winter \(1982\)](#) para o caso de um mercado 32 firmas estes valores são 0.00097, e 0.0194, respectivamente. Esses valores foram escolhidos segundo os autores, pois correspondem a uma média de sucesso de duas inovações por ano e que um sucesso em imitação fosse tão provável para a indústria como um todo quanto a um sucesso em inovação. Portanto o valor investido em P&D inovativo para as firmas, será igual à maior taxa de investimento em inovação multiplicada pelo estoque de capital das firmas

$$RD_{INjt} = r_{inj} \times K_j \quad (12)$$

O estoque de capital das firmas após cada período, para as firmas que utilizam o AG como método de aprendizagem, é calculado por:

$$K_{j(t+1)} = I \left(\frac{P_t \times A_{j(t+1)}}{c}, \frac{Q_{it}}{Q_t}, \pi_{jt}, \frac{\delta}{ft} \right) \times K_{jt} + \left(1 - \frac{\delta}{ft} \right) \times K_{jt} \quad (13)$$

Onde I é a equação do investimento, uma função da razão entre preço e custo de produção, $\frac{P_t \times A_{j(t+1)}}{c}$, do market share, $\frac{Q_{it}}{Q_t}$ do lucro da firma no período anterior, π_{jt} e da taxa de depreciação ajustada, $\frac{\delta}{ft}$, que corresponde à taxa de depreciação no trimestre, definida em 0.03 por [Nelson e Winter \(1982\)](#), dividida pelo fator de ajuste do tempo para os 6000 períodos definida como igual a 60. O investimento é definido de forma a ser não

negativo:

$$I(\rho, s, \pi, \delta_t) = \max\left[0, \min\left[\left(1 + \delta_t\right) - \frac{(2-s)}{\rho * (2-2s)}, f(\pi)\right]\right] \quad (14)$$

Sendo que $f(\pi)$ é definido de acordo com o investimento externo disponível por:

$$f(\pi) = \begin{cases} \delta_t + \pi & \text{se } f(\pi) \leq 0 \text{ ou } BANK = 0 \\ \delta_t + 2\pi & \text{se } f(\pi) > 0 \text{ e } BANK = 1 \\ \delta_t + 3.5\pi & \text{se } f(\pi) > 0 \text{ e } BANK = 2.5 \end{cases} \quad (15)$$

O Algoritmo Genético: São gerados um número pré-definido de agentes, que possuem um número fixo de estratégias, aleatoriamente geradas e que conta com um número constante de genes. Estes genes são os elementos de uma “string” binária. Os agentes testam seus cromossomos e guardam a informação sobre o desempenho de cada um deles e, ao final de cada geração, o melhor cromossomo é selecionado para compor a geração seguinte. O restante dos novos cromossomos serão gerados através dos operadores de cruzamento e mutação.

Como apresentado por Yildizoglu (2002) cada cromossomo possui G genes, portanto, para o caso de G igual a 7 o conjunto de possibilidades do intervalo é:

$$\Delta = \sum_{i=0}^{G-1} 1 \cdot 2^i = 127 \quad (16)$$

Implementa-se esta primeira geração essas estratégias de forma aleatória e cada um destes cromossomos representa um percentual de investimento em P&D e é utilizado de forma a se obter uma maior taxa de lucro, levando em conta o equilíbrio entre em P&D buscando aumento produtividade e o investimento no estoque de capital, os dois fatores cruciais para determinação da taxa de lucro. Repete-se o processo para cada um dos cromossomos pelo período de aprendizagem, que consiste no número de períodos em que cada uma das estratégias é utilizado.

O algoritmo genético aplicado possui três etapas, uma de elitismo, na qual a melhor estratégia é mantida para o próximo período sem sofrer modificação pelas duas etapas posteriores, a etapa de cruzamento e a etapa de mutação.

A etapa de cruzamento consiste na seleção de dois “genitores”, cromossomos “pais”, que através do método “roulette-wheel”, no qual a probabilidade de um cromossomo ser escolhido é proporcional ao seu “fitness”, que é definido como o resultado da função. Os dois irão se reproduzir com uma probabilidade $P[X=1]$ igual a 70%. E, caso haja sucesso nesse evento aleatório, ocorrerá a reprodução de um único ponto desses cromossomos, ou seja a partir de uma posição da “string” haverá a troca de material genético, gerando dois

novos cromossomos, os quais irão compor a nova lista de estratégias. Quando a reprodução não ocorrer, os próprios genitores serão adicionados ao conjunto de estratégias. Para o caso de 8 cromossomos, esse processo é repetido por quatro vezes, gerando 8 novas estratégias, das quais uma é eliminada, de forma que, ao se adicionar a estratégia vencedora do período anterior (elitismo), mantém-se a mesma dimensão do conjunto de cromossomos, ou seja, 8 estratégias.

A etapa de mutação consiste em um novo processo estocástico para cada gene das estratégias, são excluídos aqueles do cromossomo vencedor da geração anterior. Sofrerá mutação os genes que obtiverem sucesso no evento probabilístico, $P[M=1]$, no qual a probabilidade de ocorrer mutação do gene, é igual a 3%. Ou seja, no caso de sucesso no evento, o gene correspondente será trocado, portanto, se este era 0, se torna 1, ou vice-versa. A mutação de cada gene é independente e a estratégia vencedora não sofre mutação, pois na teoria ela já possui as características mais desejadas.

Ao fim do período de aprendizagem de cada estratégia, o maior valor médio da taxa de lucro entre os cromossomos, é considerado a estratégia vencedora e, portanto, estes são selecionados e mantidos para a próxima geração (elitismo). As outras estratégias são determinadas por eventos probabilísticos, como explicados anteriormente, na Seção 2.1, representados por duas etapas, uma de cruzamento e outra de mutação, com probabilidades de sucesso relativas de 0.7 e 0.03, respectivamente, probabilidades que foram aplicadas no trabalho de [Yildizoglu \(2002\)](#). Após a ocorrência dos operadores genéticos da mutação e cruzamento, são selecionados os novos cromossomos para a próxima geração, que é novamente testada, sendo repetido esse processo de forma iterativa até o fim da simulação.

4 - Teste do Algoritmo Genético

Com o intuito de se testar o algoritmo genético e verificar sua convergência ao valor máximo de uma função. Busca-se encontrar um valor aproximado para o máximo global através da convergência do algoritmo quando aplicado de duas equações, uma que conta com um único máximo, e a outra que possui, ainda, um máximo local. A convergência será realizada utilizando um algoritmo genético que permite testar partes de um intervalo de busca pré-determinado.

O conjunto inicial de estratégias é definido aleatoriamente, utiliza-se a função `random` do NetLogo para criar `Cr` “strings” binárias de comprimento `G`. Elas são criadas a partir do código a seguir:

```
set mylist n-values Cr [n-values G [random 2]]; cria a lista de estratégias
```

A parte do espaço de busca que será representada pelo cromossomo C_i pode ser calculada por:

$$rd_{jt} = Li \times (C_i)_{10} \times \frac{1}{\Delta} \times Ib \quad (17)$$

Onde intervalo de busca das possíveis estratégias:

$$Ib = Ls - Li \quad (18)$$

As duas equações que serão maximizadas são:

-**Equação I:** $f(x) = 15 - x^2$, cujo máximo global é (0 , 15)

-**Equação II:** $f(x) = -5x^2 - 9x^3 + 16x^2 + 15x + 12$, cujo máximo global é aproximadamente (- 1.97 , 38.04) e possui um ponto de máximo local.

O modelo do algoritmo genético foi implementado, Apêndice A, utilizando 10 agentes, cada um com 8 cromossomos compondo seu conjunto de estratégias, todos contendo 7 genes, o que resulta em 127 divisões do universo pesquisado. Como inicialmente não se sabe em quais pontos se localizam os máximos das equações, deve-se estimar os limites inferiores e superiores do espaço busca. Podemos observar o resultado da convergência de x obtido para o caso onde $Li = -20$ e $Ls = 40$, a Figura 1 ilustra os resultados para a Equação I.

Verifica-se que mesmo para um universo de busca relativamente grande e a utilização da média dos conjuntos das estratégias como uma aproximação do valor da estratégia ótima, encontrou-se um resultado para o valor máximo aproximadamente (0.44, 14,16) que está, considerando os desvios padrões, englobando o ponto do máximo real (0,15). Portanto, a

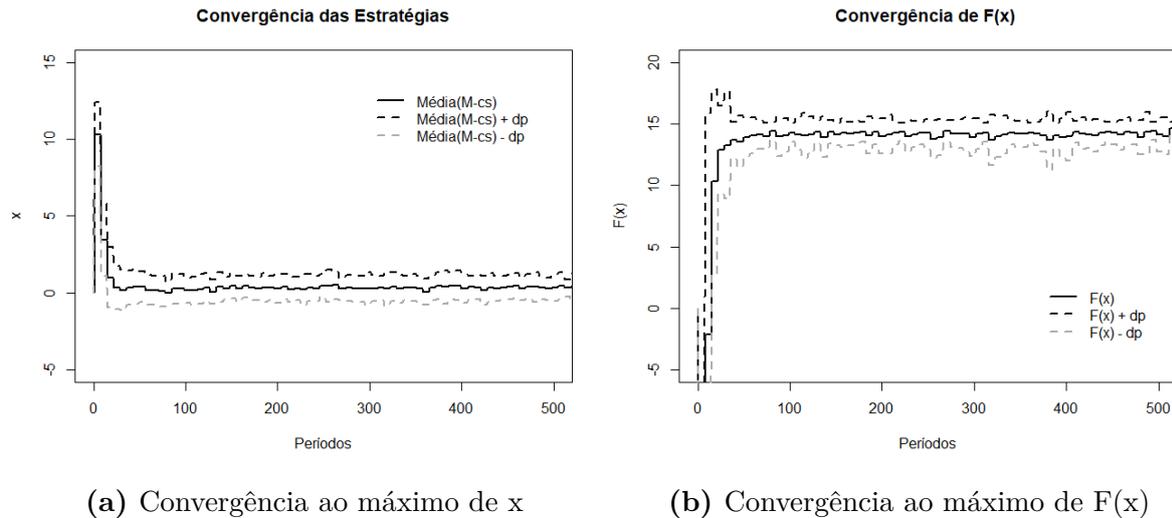
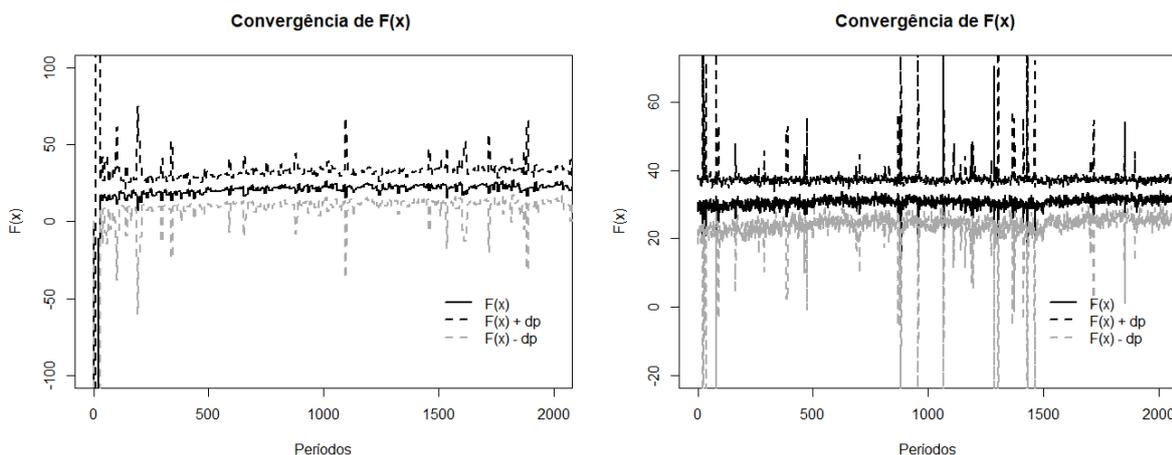


Figura 1 – Convergência das estratégias para a Equação I.

aproximação é satisfatória e mesmo desconsiderando o aumento da variabilidade, resultante da etapa de mutação, a convergência do algoritmo foi razoável.

Já para a Equação II, foi encontrado, para o mesmo intervalo de busca, o valor de convergência para a estratégia aproximadamente igual a (-1.24) , que, apesar de não ser um valor tão distante do valor máximo para $x(-1.89)$, resulta em um valor de $F(x)$, como demonstrado na Figura 2a muito distante do máximo(22.94 sendo que deveria estar próximo de 38.04), mesmo reduzindo o espaço de busca para $L_i = -6$ e $L_s = 10$, a convergência da função às proximidades do valor máximo real não ocorre, Figura 2b. Logo para este caso a aproximação da média do conjunto das estratégias não é satisfatória.



(a) Convergência de $F(x)$, $L_i = -20$ e $L_s = 40$. **(b)** Convergência de $F(x)$, $L_i = -6$ e $L_s = 10$.

Figura 2 – Convergência para o máximo da função para a Equação II.

A opção mais satisfatória neste caso é utilizar a convergência do valor que é mantido a cada geração pela etapa de elitismo(o cromossomo que resulta no maior valor para a função é mantido), pois este não possui a variabilidade resultante da etapa de mutação.

A variabilidade é importante, pois permite que outras faixas do espaço sejam pesquisadas, deste modo, por exemplo, em um problema com mais de um máximo, a mutação permite que outros pontos de estabilidade sejam encontrados, evitando a convergência prematura, por exemplo, para um ponto de máximo local.

Verifica-se que, ao se selecionar apenas a média das melhores estratégias para cada geração, tanto para a Equação I, Figuras 3a e 3b, quanto para a Equação II, Figuras 3c e 3d, houve não só a convergência para as proximidades do valor máximo, como também foram encontrados valores com um grau de precisão considerável.

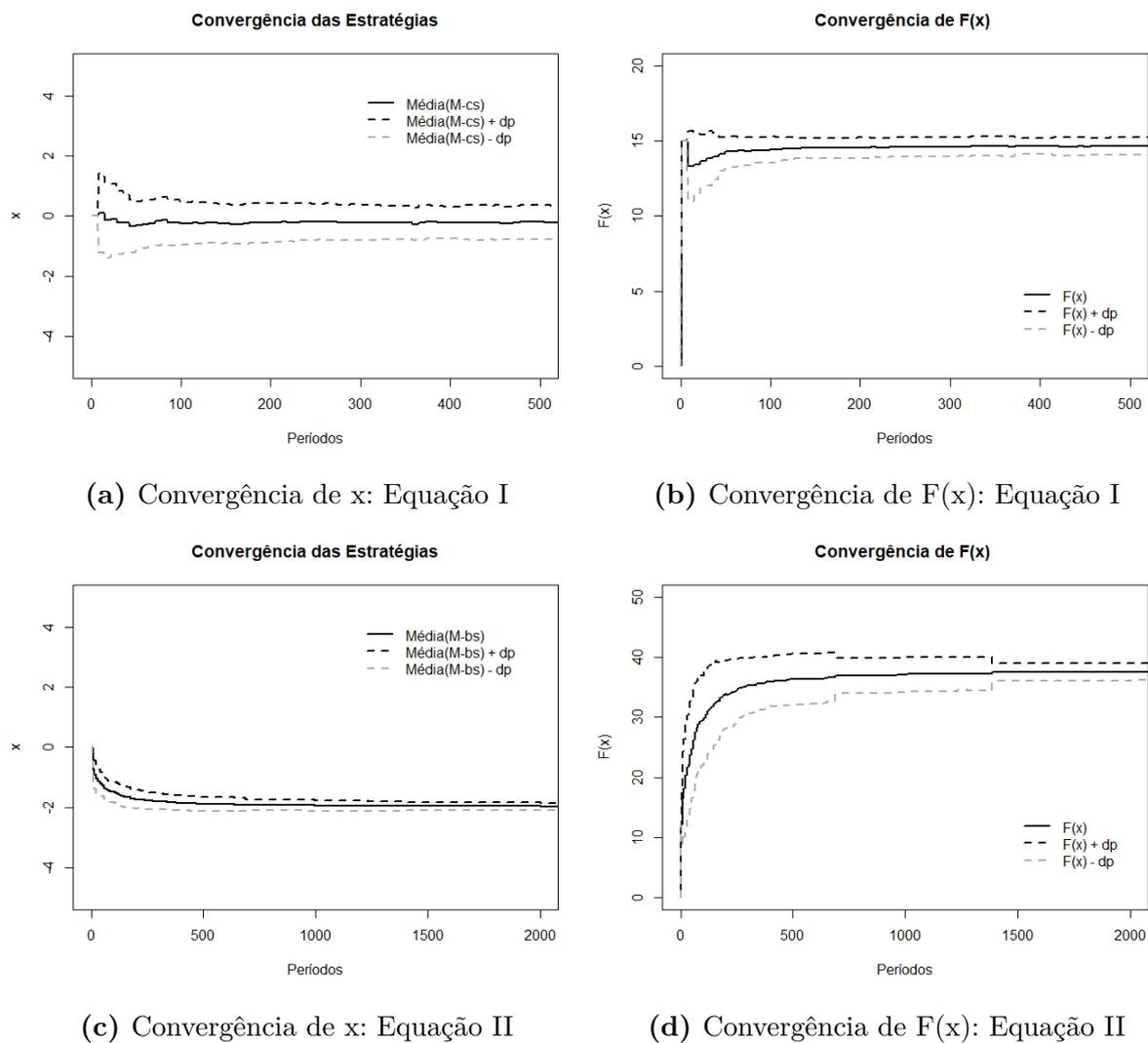


Figura 3 – Convergência para o máximo de cada uma das equações utilizando apenas a média das melhores estratégias de cada geração.

Portanto, observa-se que cada etapa do algoritmo genético escolhido tem uma função a ser desempenhada, o elitismo garante que características positivas não se percam, a mutação e o cruzamento permitem que outras partes do espaço de busca sejam investigadas e, para precisões ainda maiores pode-se diminuir o espaço de busca ou aumentar o número de genes. Foi verificado que o algoritmo genético realmente retornou valores aproximados

para os máximos de ambas as funções o que indica que o algoritmo apresentam convergência, com exatidão, para determinados valores de “*fitness*”. Este é um forte indício que o algoritmo está apto para ser testado no modelo industrial evolucionário, originalmente proposto por [Nelson e Winter \(1982\)](#).

5 - Resultados

Através código do modelo evolucionário, Apêndice B, foram realizadas simulações utilizando dois mercados diferentes, um contendo apenas firmas F.Gen e F.NW, em diferentes proporções e outro contendo firmas F.Gen e F.RND.

Analisa-se a produtividade média ao fim da simulação dos dois mercados para as diferentes configurações considerando, primeiramente, o efeito do investimento externo por meio de crédito no desempenho do mercado, Figura 4, nesta situação o número de períodos de aprendizagem foi mantido constante em 5:

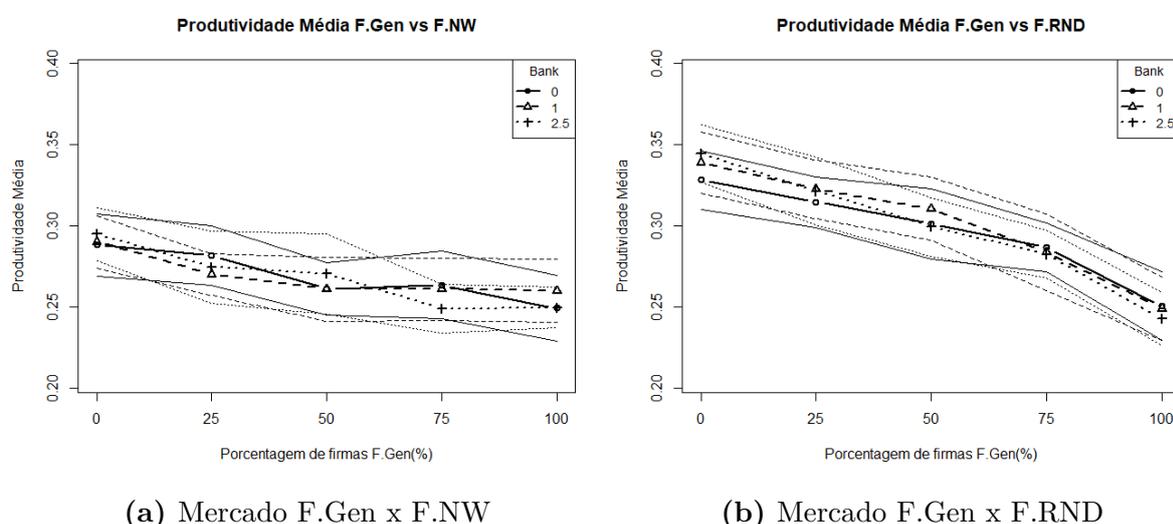


Figura 4 – Produtividade média para diferentes proporções de F.Gen em diferentes mercados com investimento externo variável acrescidas de linhas de desvio padrão

Verifica-se que os níveis de produtividade média dos dois mercados são similares quando o percentuais das firmas que utilizam o algoritmo genético é de 100%, já para percentuais menores de firmas genéticas, no mercado em que as firmas que utilizam as estratégias fixas, Figura 4a, tem-se níveis de produtividade médias menores que no mercado das firmas que utilizam a aleatoriedade na decisão, Figura 4b. Esta situação se deve ao fato que mesmo havendo a aleatoriedade o percentual do capital que será investido em inovação pelas firmas F.RND será, na média, maior que aquele das firmas F.NW, já que elas podem investir até 5 vezes mais em inovação que uma firma F.NW investe, o que justifica os níveis de produtividade mais altos para maiores percentuais de firmas F.RND. Em ambos os casos há uma diminuição na produtividade média à medida que se aumenta a proporção de firmas F.Gen e pode-se afirmar que a disponibilidade de crédito externo não leva a alterações nos níveis de produtividade média, fato que corrobora com os resultados encontrados no trabalho de Nelson e Winter (1982).

Quando analisa-se o valor de produtividade média em relação ao percentual de firmas F.Gen, levando-se em conta o número de períodos de aprendizagem e mantendo fixo o crédito bancário em 0, temos que, como pode ser observado na Figura 5, as produtividades médias se comportam de forma diferente dependendo do número de períodos de aprendizagem e do mercado envolvido.

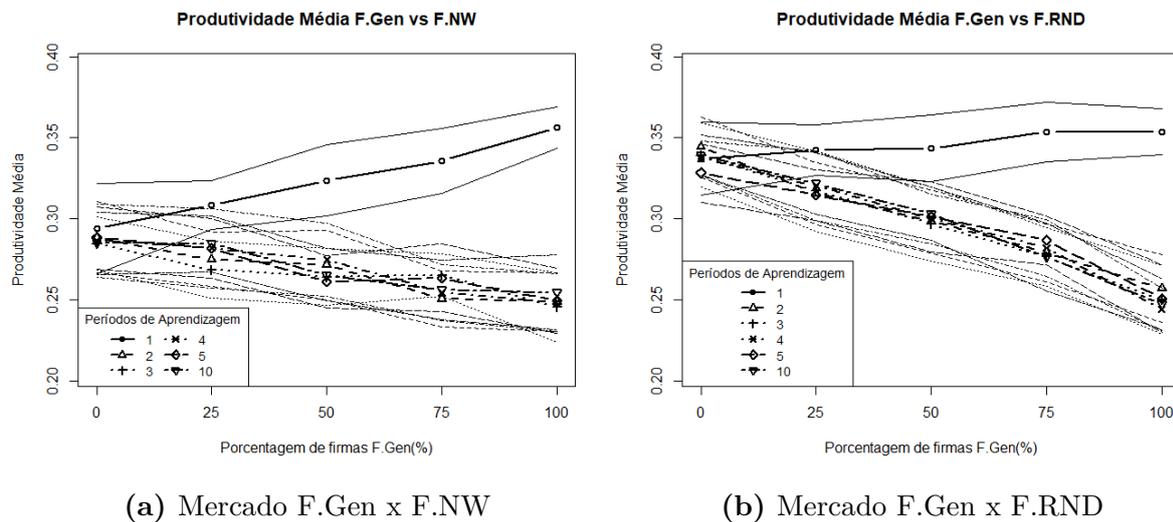


Figura 5 – Produtividade média e respectivas linhas de desvio padrão ($\pm dp$) para diferentes proporções de F.Gen em diferentes mercados com períodos de aprendizagem variáveis

Em um mercado com firmas F.NW, F.Gen e período de aprendizagem igual a 1, Figura 5a, faz com que a produtividade média aumente à medida que se aumenta o percentual de F.Gen, mas a medida que se aumenta a duração do período de aprendizagem mesmo em apenas uma unidade, igual a 2, a produtividade média passa a diminuir à medida que se aumenta a proporção de firmas F.Gen.

Já no mercado com firmas F.RND e F.Gen, Figura 5b, quando o período de aprendizagem é igual a unidade, tem-se o nível de produtividade média praticamente constante para todos os percentuais de firmas F.Gen. Para os níveis de períodos de aprendizagem maiores que 1, o modelo comportamento similar ao do outro mercado, com queda na produtividade à medida que se aumenta a proporção de firmas F.Gen.

A não alteração na produtividade média com o aumento da proporção de F.Gen nos mercados divididos com F.RND, é um indício de que, para o período de aprendizagem mínimo, as firmas F.Gen se comportam como firmas que utilizam características de escolha de estratégias aleatórias. Esta afirmação é fortalecida pela situação dos mercados com firmas F.NW, já que a produtividade média aumenta à medida que se aumenta a proporção das firmas que utilizam os algoritmos genéticos apenas para o mesmo período de aprendizagem. Para as situações em que as durações são maiores, em ambos os casos o comportamento da produtividade média não se modifica à medida em que é aumentada a duração, pelo contrário, em cada um dos casos verifica-se sobreposição das médias e dos

desvios padrões.

Considerando o preço dos produtos em ambos os mercados, não há variação significativa quando consideradas alterações no número de períodos, Figura 6, ou, no crédito disponível. Observa-se que a média dos preços finais e seus respectivos desvios padrões não modificam seu comportamento para os três períodos diferentes.

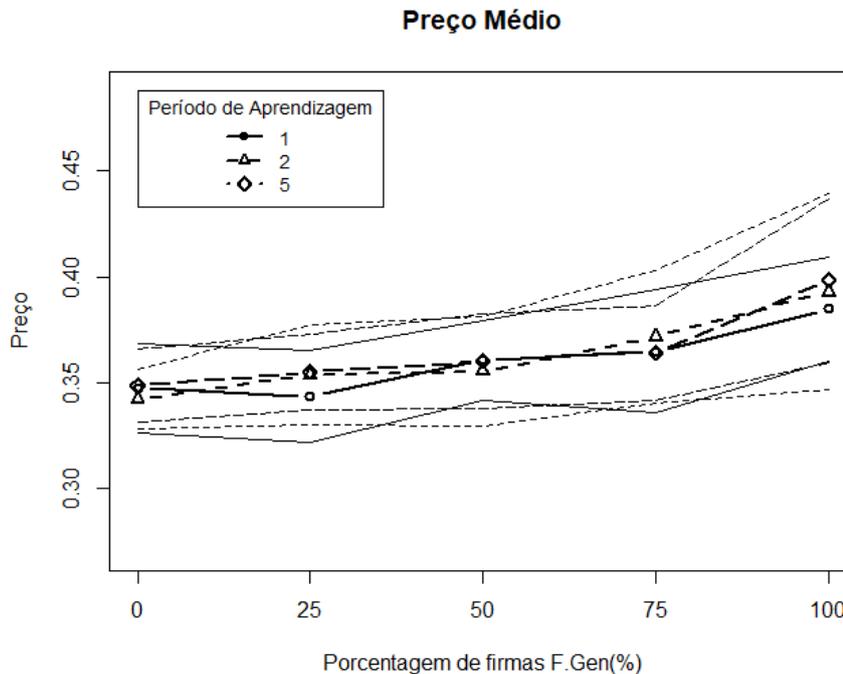


Figura 6 – Preço médio e respectivas linhas de desvio padrão ($\pm dp$) para diferentes porcentagens de F.Gen em diferentes mercados com períodos de aprendizagem variáveis em um mercado F.Gen vs F.RND.

Ao se comparar os preços médios finais, ao fim dos 6000 períodos, para os dois mercados, considerando a situação em que a duração dos períodos de aprendizagem é igual a 5 e o crédito bancário é nulo, Figura 7, verifica-se que os preços médios finais são mais elevados para o mercado das firmas de estratégias fixas(F.NW). Esta situação se verifica para todas as proporções de firmas que utilizam o algoritmo genético, com exceção do caso de 100% das firmas(idêntico para ambos os mercados).

O preço médio mais elevado nos mercados nos quais as F.NW estão presentes, indicam que nestes casos o bem-estar social do mercado é menor, pois os preços médios são mais altos, o que diminui o poder de compra dos consumidores. Por outro lado, nos dois mercados, o aumento do percentual de firmas F.Gen parece reduzir o bem-estar social, pois indica o aumento do preço médio, este achado vai contra uma das afirmações apresentadas no trabalho de Yildizoglu (2002), que indicava um aumento do bem-estar social devido à presença de firmas que utilizavam os algoritmos genéticos, já que, naquele caso, o preço do produto diminuía à medida que se aumentava proporção das firmas que

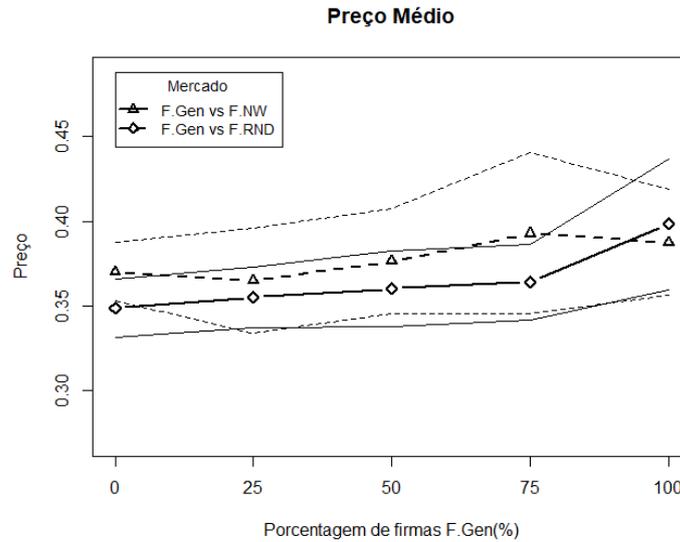


Figura 7 – Preço médio para diferentes percentuais de F.Gen em diferentes mercados utilizavam o algoritmo.

Na Figura 8, é possível analisar a evolução dos preços médios ao longo da simulação, para ambos os mercados com um percentual de Firmas F.Gen de 50%, período de aprendizagem igual a 5 e o crédito nulo. Observa-se que ao longo de toda a simulação o preço médio da situação onde estão presentes firmas F.NW permanece maior do que aquela onde não se tem, consequência provável dos valores de produtividade média mais altos devido a um investimento maior em P&D inovativo por parte das firmas F.RND.

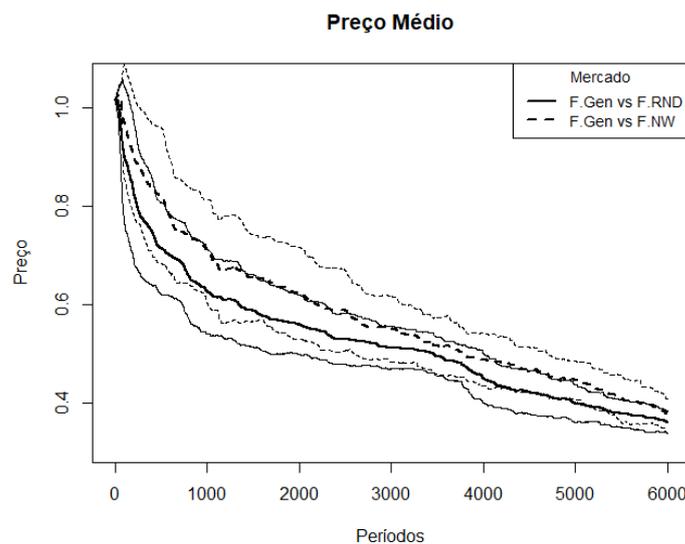


Figura 8 – Preço médio ao longo da simulação e respectivas linhas de desvio padrão($\pm dp$) para os dois mercados.

Como era de se esperar, ao se analisar o investimento em P&D inovativo acumulado médio para os mercados na mesma situação descrita no parágrafo anterior, Figura 9, observa-se que o investimento no caso dos mercados F.RND e F.Gen é muito maior que o

investimento do mercado que contém firmas F.NW, neste sentido é de se esperar que o primeiro tenha maior sucesso em inovação e atinja níveis de produtividade maiores, o que leva a menores preços médios no mercado e consequente maior bem-estar social.

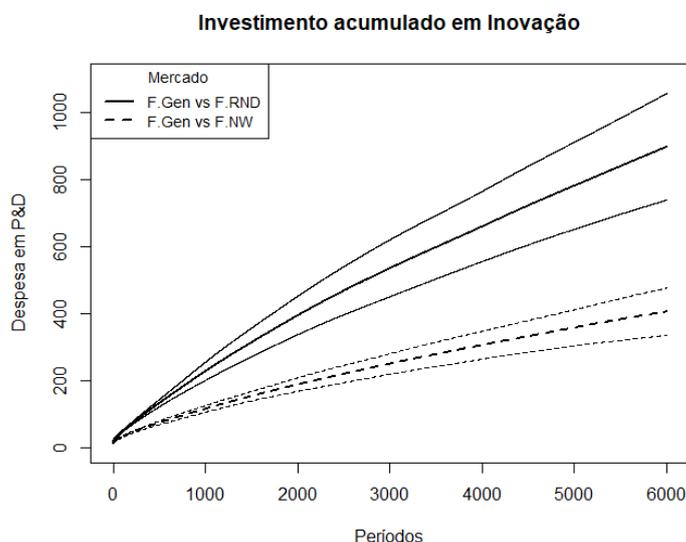
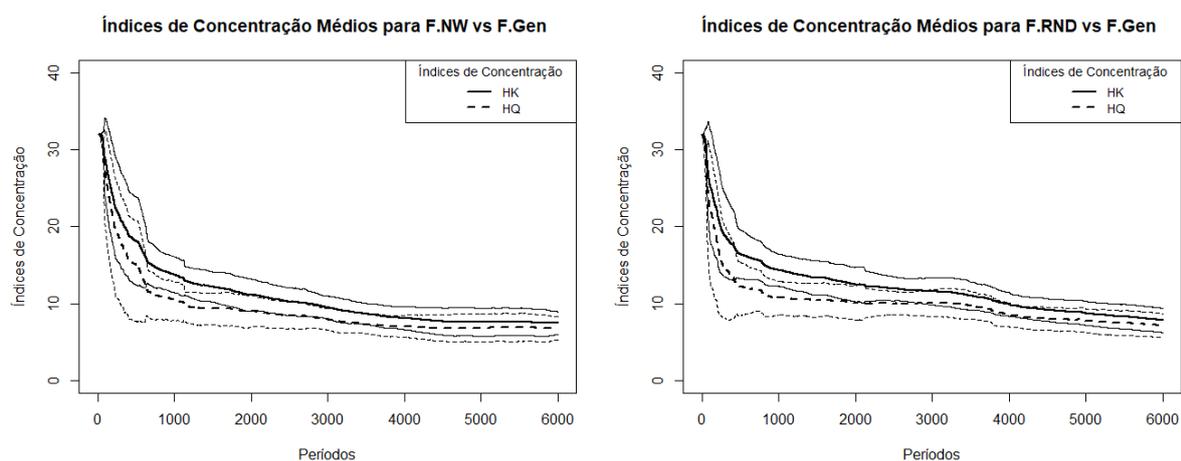


Figura 9 – Investimento acumulado em P&D inovativo ao longo da simulação e respectivas linhas de desvio padrão($\pm dp$) para os dois mercados.

Quando se analisa a média do inverso dos índices de Herfindahl para concentração de capital, HK, e de produção HQ, Figura 10, verifica-se comportamento similar para o caso de ambos os mercados considerando 50% de firmas genéticas, período de aprendizado igual a 5 e crédito bancário nulo.



(a) Mercado F.Gen x F.NW

(b) Mercado F.Gen x F.RND

Figura 10 – Evolução da concentração e suas respectivas linhas de desvio padrão($\pm dp$) para diferentes mercados.

Compara-se em seguida, os índices de concentração ao fim da simulação para cada um dos percentuais de firmas genéticas considerando os dois mercados, Figura 11. É possível verificar que tanto os índices de produção quanto os de capital aumentam à

medida que se aumenta a proporção das firmas que utilizam o algoritmo genético, o que significa que o mercado fica menos concentrado à medida que se acrescentam firmas que utilizam AG. Verifica-se que ambos os índices de concentração, quando a proporção é 0% de firmas F.Gen e 25% de firmas F.Gen, para o mercado contendo F.NW, Figura 11b e Figura 11c são menores, para as mesmas proporções, que no mercado contendo firmas F.RND, Figura 11b e Figura 11c. Logo, conclui-se que o mercado contendo F.NW é, em geral, mais concentrado para menores proporções de firmas que utilizam AG. Já para situações de maiores proporções, essa diferença não se verifica, sendo as distribuições dos índices similares.

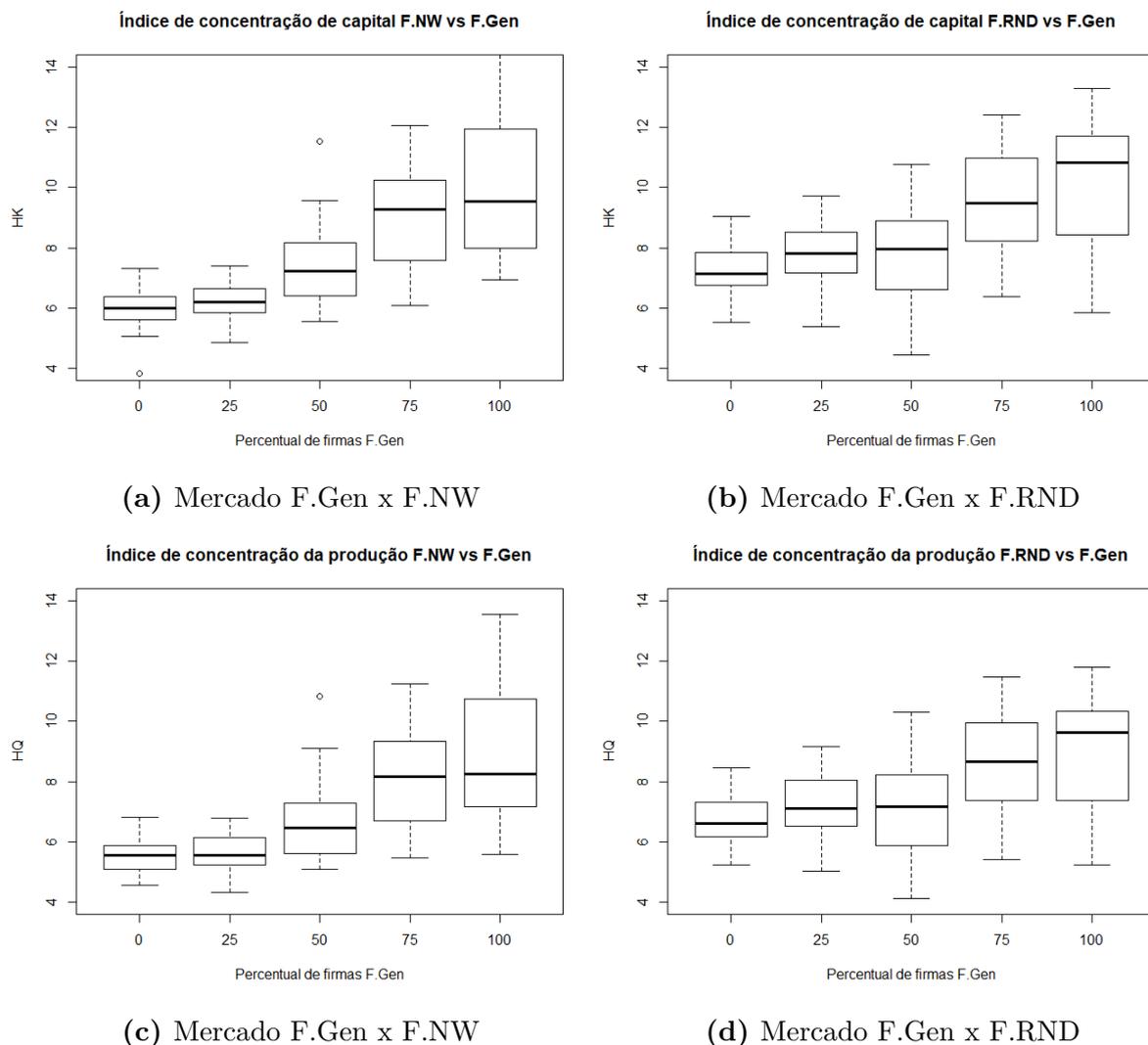
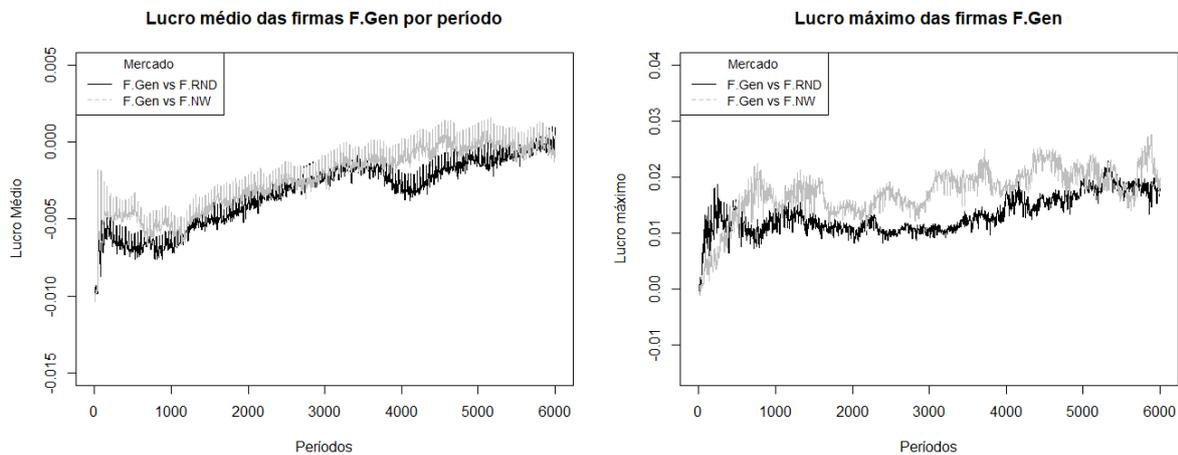


Figura 11 – Evolução da concentração do mercado e suas respectivas linhas de desvio padrão($\pm dp$) pelo percentual de firmas F.Gen

Analisando os resultados de lucro líquido, 12, é possível verificar que no mercado no qual estão presentes firmas genéticas e firmas com estratégias fixas, temos níveis de lucro, tanto médio, Figura 12a, quanto máximo, Figura 12b, superiores quando se comparado ao universo de firmas com estratégias aleatórias. Este fato é explicado pelos níveis de

produtividade mais altas das F.RND, que provocam, provavelmente, uma redução no lucro das F.Gen. Deve-se destacar que a mudança de nível brusca, para o lucro no mercado F.Gen x F.RND, próximo ao período 4000 na Figura 12a, indica um sucesso em inovação que modifica a estrutura do mercado.



(a) Lucro médio para as firmas F.Gen

(b) Lucro máximo para as firmas F.Gen

Figura 12 – Evolução dos lucros médio e máximo e respectivas linhas de desvio padrão ($\pm dp$) para as F.Gen nos dois mercados.

É fundamental realizar o estudo da convergência dos valores selecionados pelo algoritmo genético, verifica-se que, para todas as proporções de F.Gen, períodos de aprendizagem e disponibilidade de crédito bancário, o comportamento das firmas F.Gen é semelhante. A Figura 13, ilustra a situação de convergência da média das estratégias, utilizadas pelas F.Gen, para valores próximos de 0, na situação de 50% de firmas que utilizam o AG, período de aprendizagem fixado em 5 e crédito bancário nulo.

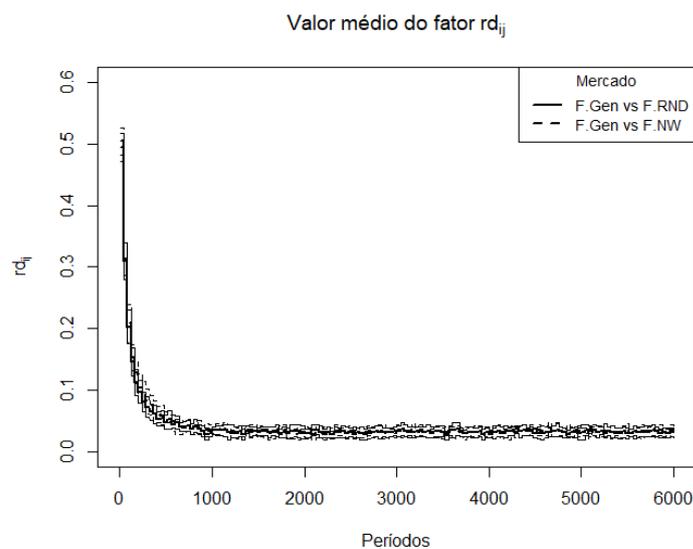


Figura 13 – Convergência das estratégias F.Gen ao longo do tempo.

Verifica-se que os Algoritmos Genéticos identificam como melhor estratégia, aquela nula, a qual só não é mantida fixa porque existe a etapa de mutação em atuação. Esta, faz com que novas estratégias sejam criadas de forma a se pesquisar o espaço de busca na procura por melhores candidatos. A convergência é de fato curiosa, já que as firmas que utilizam o AG tendem, então, à mesma situação das firmas imitativas no modelo Nelson e Winter (1982).

Naquele modelo as firmas imitativas não investem em P&D inovativo, se limitando a tentar imitar e investir em capital físico. As firmas que utilizam os Algoritmos Genéticos encontram exatamente esta mesma situação como convergência. É necessário verificar se esta situação configura uma vantagem competitiva transformada em dominância do mercado. Para essa confirmação, analisa-se a parcela do mercado, para cada um dos universos de simulação, com diferentes percentuais de firmas genéticas, Figura 14 .

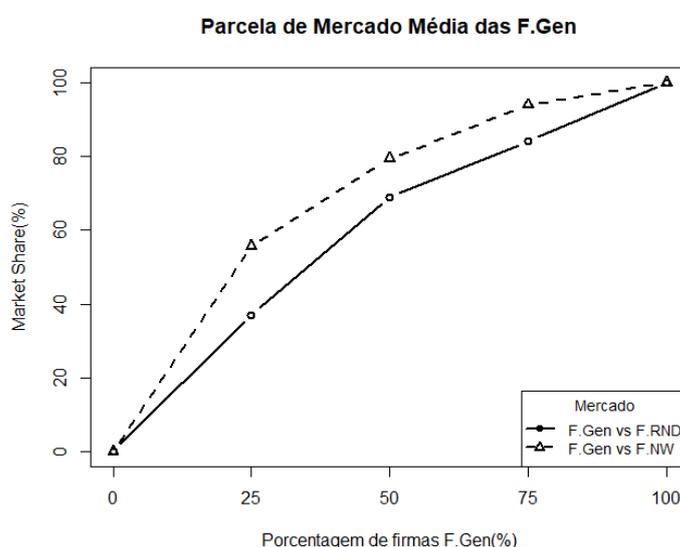


Figura 14 – Parcela de mercado média das F.Gen para diferentes proporções.

É possível perceber que, para todas as situações as firmas F.Gen dominam o mercado, pois seu “*market share*” sempre supera a sua proporção em relação às outras firmas. É nítido que quando as firmas genéticas dividem mercado com as firmas de estratégias fixas, elas levam mais vantagem que quando dividem com as que possuem estratégias aleatórias, Este fato se deve aos níveis de produtividade mais altos dessas firmas F.RND, mas mesmo nesse caso, as firmas que utilizam o algoritmo genético superam suas proporções como demonstrado, Tabela 4, a seguir:

Os resultados confirmam que as firmas que utilizam o algoritmo genético na maior parte das vezes domina o mercado, demonstrando que a técnica utilizada por elas é uma estratégia vencedora. Dessa forma, compartilha-se dos achados de Yildizoglu (2002), onde

Tabela 4 – Parcela do mercado das firmas F.Gen

Porcentagem de Firmas que utilizam AG(%)					
Mercado	0%	25%	50%	75%	100%
F.NW vs F.Gen	0.00	55.75	79.42	94.00	100
(Desvio Padrão)	(0.00)	(18.95)	(17.65)	(6.11)	(0.00)
F.RND vs F.Gen	0.00	37.07	68.90	84.15	100
(Desvio Padrão)	(0.00)	(20.43)	(12.72)	(10.58)	(0.00)

se afirma que as firmas, que utilizam o AG, aprendem e sua aprendizagem é individual. Os resultados demonstram que houve uma convergência, apesar de diferente daquela assinalada [Yildizoglu \(2002\)](#), na qual as firmas genéticas tendem ao longo da simulação para o caso de firmas puramente imitativas do modelo de [Nelson e Winter \(1982\)](#). Esta convergência, aliadas aos investimentos em inovação que ocorrem nos períodos iniciais, se mostra uma estratégia que domina o mercado.

6 - Conclusão

O modelo evolucionário de [Nelson e Winter \(1982\)](#) é bastante utilizado como base para trabalhos e possibilita inúmeras formas de adaptações. Em uma dessas adaptações, [Yildizoglu \(2002\)](#) implementou uma simplificação deste modelo, no qual aplicou um Algoritmo Genético como forma de reger as decisões das empresas. Entre as conclusões o autor indica a comprovação da aprendizagem pelas firmas que utilizam o AG, encontrando uma convergência entre as estratégias individuais aliada a uma dominância do mercado por essas firmas.

Contudo, os procedimentos envolvidos e os ajustes implementados por [Yildizoglu \(2002\)](#) não foram possíveis de serem executados o que não permitiu a replicação com exatidão o modelo e identificar resultados similares, impossibilitando a validação do experimento. Neste sentido, foi proposto a criação de um novo modelo que procurou ser ao máximo fidedigno ao modelo original de [Nelson e Winter \(1982\)](#), aplicando algumas modificações propostas pelo modelo simplificado e abrindo mão de outras.

O número de firmas utilizado nas simulações foi idêntico ao mercado competitivo do modelo original, assim como as respectivas taxas de investimento em P&D inovativo e imitativo. O número de períodos seguiu aquele utilizado no modelo simplificado, dessa forma decidiu-se por realizar um ajuste temporal na taxa de depreciação de cada período e nos eventos estocásticos que determinam os sucessos em inovação e imitação. Foi decidido não implementar a saída de firmas no mercado, assim como limitar o investimento possível em inovação pelas firmas que não possuem estratégias ficas a no máximo cinco vezes o percentual do capital aplicado pelas firmas do modelo original. O percentual é inclusive um outro parâmetro que foi modificado em relação ao trabalho de [Yildizoglu \(2002\)](#), o autor utiliza como referência o lucro no período anterior, enquanto no trabalho inicial, utiliza-se, como presente estudo, o percentual do capital.

Primeiramente, foi definido o algoritmo genético de forma que suas estratégias fossem escolhidas seguindo determinadas regras:

- As estratégias para cada firma são geradas inicialmente de forma aleatória;
- Nas próximas gerações o elitismo aplicado por [Yildizoglu \(2002\)](#) garante que a melhor estratégia é mantida na próxima geração;
- O método “*Roulette Wheel*” explicado por [Mitchell \(1998\)](#) é utilizado para definir as outras estratégias através dos operadores genéticos de cruzamento e mutação.

O algoritmo foi testado, como proposto por [França \(2014\)](#), a partir da sua aplicação para solucionar dois problemas de maximização de solução simples. Verificou-se que devido

a variabilidade resultante intrínseca ao operador da mutação não permitiu considerar a média de todo o conjunto de estratégias, dos respectivos períodos, como solução aproximada para um dos problemas de maximização, devendo utilizar apenas a melhor das estratégias desses períodos como uma solução. Para o outro problema, ambas as opções resultaram em aproximações satisfatórias, mas a precisão de quando se utiliza apenas a média das melhores estratégias é maior. Uma vez verificada a calibragem do algoritmo genético, este estava pronto para ser aplicado ao modelo.

O modelo foi descrito através do “Protocolo ODD” proposto por [Railsback e Grimm \(2012\)](#). Em seguida, o modelo foi implementado e foram realizadas simulações em NetLogo utilizando dois modelos distintos, o primeiro deles com firmas que utilizam estratégias fixas de investimento em P&D inovativo junto com firmas que utilizam um algoritmo genético, e o segundo, que simulava firmas com estratégias aleatórias de decisão sobre o investimento em inovação juntas com as mesmas firmas que utilizam o algoritmo. Foram testadas diferentes proporções de firmas genéticas, três disponibilidades de crédito, e seis durações para os períodos de aprendizagem, buscando verificar os efeitos de cada uma dessas variáveis na estrutura de mercado.

Em relação à disponibilidade de crédito, os resultados encontrados corroboram com os achados de [Nelson e Winter \(1982\)](#) e se verifica que não há influência direta do aumento do financiamento em aumentos da produtividade nem em mudanças significativas da estrutura de mercado.

Quando são consideradas as durações do período de aprendizagem, foi verificado, que para durações iguais à unidade não houve seletividade suficiente das estratégias pelas firmas que utilizam o algoritmo genético, portanto não houve convergência dessas estratégias, na realidade as firmas acabaram tendo comportamento e consequente desempenho muito similar àquelas que utilizavam estratégias aleatórias. A partir de períodos de aprendizagem iguais a dois intervalos, foram encontrados resultados semelhantes entre si, demonstrando que para este algoritmo genético e nestes casos específicos estudados, períodos “curtos”, foram suficientes para oferecer a seletividade necessária para o processo evolutivo das estratégias.

Em ambos os modelos a presença de firmas genéticas diminuíram os níveis de produtividade média no mercado, levando conseqüentemente ao aumento do preço médio nos mercados. Portanto, a presença de firmas que utilizam o algoritmo genético provoca uma diminuição do bem-estar social, já que aumentos no preço reduzem o poder de compra dos consumidores. Este resultado é contrastante com aquele apresentado por [Yildizoglu \(2002\)](#), que verificou uma diminuição do preço e consequente aumento do bem-estar social.

Ao decorrer de todas as simulações foi verificada a queda do preço médio que, ao fim, se encontraram na mesma faixa daqueles encontrados por [Nelson e Winter \(1982\)](#),

o que indica a adequação ao modelo. Quanto ao investimento em inovação encontrado, verificou-se, como era esperado, que os mercados com firmas aleatórias investiram mais que aqueles que continham divisões de mercado semelhantes, mas englobavam firmas com estratégias fixas. Isto é explicado pelo fato das firmas com estratégias aleatórias possuírem um limite da taxa de investimento cinco vezes maior que aquele das firmas com estratégias fixas, portanto, na média, a taxa de investimento será superior o que explica inclusive os maiores níveis de produtividade para as firmas com decisões aleatórias.

Os indicadores de concentração mostraram que, para ambos os mercados, maiores percentuais de firmas que pautam seu investimento inovativo utilizando o algoritmo genético levam a um mercado menos concentrado. Portanto, capital e produção são mais igualmente distribuídos, podendo-se inclusive conjecturar uma maior competitividade do mercado, hipótese, a qual, não é confirmada por menores níveis de preço.

Ao se analisar a possível convergência das estratégias das firmas que utilizam o algoritmo genético, foi encontrado resultados parcialmente similares aos apresentados por [Yildizoglu \(2002\)](#), pois houve uma convergência. Mas, diferentemente do trabalho anterior, o investimento em inovação convergiu para níveis próximos de zero, sendo a estratégia nula aquela que apresentou melhores resultados ao fim da simulação. Este fato é curioso, porque o comportamento das firmas genéticas com o avanço da simulação foi se tornando, portanto, um comportamento idêntico ao das firmas imitativas, aquelas que não investem em inovação, no modelo de [Nelson e Winter \(1982\)](#).

Analisando os efeitos dessa estratégia de redução dos investimentos em P&D inovativo, verifica-se que é uma estratégia se mostra vencedora, já que as firmas genéticas para qualquer uma das suas proporções, superam o percentual equivalente à sua proporção na parcela do mercado, ou seja, além da parte que lhe caberia caso todas as firmas tivessem o mesmo tamanho e produtividade, elas conseguem conquistar uma parcela extra do mercado. Este efeito é mais fortemente observado no mercado com estratégias fixas e para menores percentuais de firmas com decisões por algoritmo genético, mas também é evidente no mercado de firmas aleatórias e para maiores proporções.

Assim como afirmado trabalho de [Yildizoglu \(2002\)](#), conclui-se que as firmas utilizadoras do Algoritmo Genético realmente aprendem, já que mesmo atuando de forma independente, seu nível de inteligência (tomada de decisão segundo determinados parâmetros) as permite ajustar suas estratégias de forma a possuírem uma dominância do mercado, logo, é nítida a existência e atuação do processo de aprendizagem individual.

Referências

- ARIFOVIC, J. Genetic algorithm learning and the cobweb model. *Journal of Economic dynamics and Control*, Elsevier, v. 18, n. 1, p. 3–28, 1994. [9](#), [10](#), [11](#)
- BECKENBACH, F. Learning by genetic algorithms in economics? In: *Computational techniques for modelling learning in economics*. New York, NY: Springer, 1999. p. 73–100. [11](#), [12](#)
- BLICKLE, T.; THIELE, L. A comparison of selection schemes used in evolutionary algorithms. *Evolutionary Computation*, MIT Press, v. 4, n. 4, p. 361–394, 1996. [20](#), [21](#)
- FRANÇA, P. S. *Modelagem Evolucionária de Estratégias Competitivas em P & D para NetLogo: Uma Abordagem de Modelos Baseados em Agentes*. TCC - Ciências Econômicas, Mariana, 2014. [14](#), [48](#)
- HANCOCK, P. J. An empirical comparison of selection methods in evolutionary algorithms. In: SPRINGER. *AISB Workshop on Evolutionary Computing*. [S.l.], 1994. p. 80–94. [18](#)
- HOLLAND, J. H. *Adaptation in natural and artificial systems. an introductory analysis with application to biology, control, and artificial intelligence*. University of Michigan Press, Ann Arbor, MI, 1975. [9](#)
- HOLLAND, J. H. Genetic algorithms. *Scientific american*, JSTOR, v. 267, n. 1, p. 66–73, 1992. [22](#)
- KETTLEWELL, H. B. D. Recognition of appropriate backgrounds by the pale and black phases of lepidoptera. *Nature*, Nature Publishing Group, v. 175, n. 4465, p. 943, 1955. [22](#)
- KWAŚNICKI, W. Evolutionary economics and simulation. In: *Computational Techniques for Modelling Learning in Economics*. New York, NY: Springer, 1999. p. 3–44. [13](#)
- MAHFOUD, S. W.; GOLDBERG, D. E. Parallel recombinative simulated annealing: a genetic algorithm. *Parallel computing*, Elsevier, v. 21, n. 1, p. 1–28, 1995. [20](#)
- MAZA, M. d. l.; TIDOR, B. An analysis of selection procedures with particular attention paid to proportional and boltzmann selection. In: MORGAN KAUFMANN PUBLISHERS INC. *Proceedings of the 5th International Conference on Genetic Algorithms*. [S.l.], 1993. p. 124–131. [20](#)
- MILLER, B. L.; GOLDBERG, D. E. Genetic algorithms, tournament selection, and the effects of noise. *Complex systems*, [Champaign, IL, USA: Complex Systems Publications, Inc., c1987-, v. 9, n. 3, p. 193–212, 1995. [21](#)
- MITCHELL, M. *An introduction to genetic algorithms*. [S.l.]: MIT press, 1998. [16](#), [17](#), [18](#), [19](#), [20](#), [21](#), [22](#), [23](#), [48](#)
- NELSON, R. R.; WINTER, S. G. *An evolutionary theory of economic change*. Cambridge, MA: Harvard Business School Press, 1982. [8](#), [13](#), [14](#), [25](#), [26](#), [30](#), [31](#), [32](#), [38](#), [39](#), [46](#), [47](#), [48](#), [49](#), [50](#)

- RAILSBACK, S. F.; GRIMM, V. *Agent-based and individual-based modeling: a practical introduction*. [S.l.]: Princeton University Press, 2012. 25, 49
- ROGERS, A.; PRÜGEL-BENNETT, A. Modelling the dynamics of a steady-state genetic algorithm. *Foundations of genetic algorithms*, Morgan Kaufmann, v. 5, p. 57–68, 1999. 21
- SCHAFFER, M. E. Are profit-maximisers the best survivors?: A darwinian model of economic natural selection. *Journal of Economic Behavior & Organization*, Elsevier, v. 12, n. 1, p. 29–45, 1989. 9
- SHUBIK, M.; VRIEND, N. J. A behavioral approach to a strategic market game. In: *Computational techniques for modelling learning in economics*. New York, NY: Springer, 1999. p. 261–282. 12, 13
- THIERENS, D.; GOLDBERG, D. Convergence models of genetic algorithm selection schemes. *Parallel problem solving from nature—PPSN III*, Springer, p. 119–129, 1994. 19
- VRIEND, N. J. An illustration of the essential difference between individual and social learning, and its consequences for computational analyses. *Journal of economic dynamics and control*, Elsevier, v. 24, n. 1, p. 1–19, 2000. 9, 13
- WILENSKY, U. Netlogo.(ccl.northwestern.edu/netlogo). *Center for Connected Learning and Computer Based Modeling, Northwestern University*, 1999. 25
- WINTER, S. G. Schumpeterian competition in alternative technological regimes. *Journal of Economic Behavior & Organization*, Elsevier, v. 5, n. 3-4, p. 287–320, 1984. 8, 29
- WITT, U. Firms' market behavior under imperfect information and economic natural selection. *Journal of Economic Behavior & Organization*, Elsevier, v. 7, n. 3, p. 265–290, 1986. 8, 9
- YILDIZOGLU, M. Competing R & D strategies in an evolutionary industry model. *Computational Economics*, Springer, v. 19, n. 1, p. 51–65, 2002. 5, 6, 13, 14, 25, 26, 27, 28, 29, 33, 34, 41, 46, 47, 48, 49, 50

APÊNDICE A

Código em NetLogo[®] referente ao Protocolo ODD para o Algoritmo genético proposto na seção 2.1:

```
globals [
DELTA ; set of possible strategies
mylistP; list of possibilities and coefficients of each gene
mean-cs; mean value for the current used strategies
test; value for f(mean-cs),
mean-bs; mean value for the best strategy of the agents
test-bs
]
extensions [ rnd ]; Roulette Wheel Selection extensions
turtles-own[
cs; current strategy
u; vector of fitness values
v; vector of the result of crossover
mylist; list of strategies
rdij; set of values which eachone is related with an strategy
pg ; previous generation
ng ; new generation after crossover
ib; search space
mean-rdij; mean value of the set of rdij values
bs; best strategy rdij from each genetarion
]

to setup
clear-all
crt nt; number of turtles
let mylist1 n-values G [i -> i];
let mylist2 sort-by > mylist1
set mylistP (map ^ n-values G [2] mylist2); create list of genes coefficients
set DELTA sum mylistP; number of possible strategies
ask turtles[
set u []
set rdij []
set ib ls - li
set mylist n-values Cr [n-values G [random 2]]; create list of Cr chromossomes
```

```

foreach mylist[
[ strategy ] ->
let k sum (map * strategy mylistP) / DELTA
set rdij lput (li + k * ib) rdij ];create the list of rdij values related to
mylist
reset-ticks
end

to go
ask turtles[set mean-rdij mean rdij];
if ticks > nbPeriods [stop] ; stop rule, if ticks are grater than the
declared periods simulation stops
ask turtles [ifelse ticks = 0 [set cs item 0 rdij; set current strategy as
the first item of the rdij list
set u lput fitness cs u; tests fitness of the current strategy
][
each-strategy-test; procedure that tests each strategy for a predefined
number of timesteps(in this case 1) and returns fitness of cs
if ticks mod ( Cr - 1 ) = 0[; after each strategy is tested the chromossomes
are changed
final-chromossomes; procedure that defines elitism, crossover, mutation to
create the next generation of strategy
set cs item 0 rdij; set current strategy as the first item of the new rdij
list
set u lput fitness cs u];tests fitness of the current strategy
]
]
set mean-cs mean[mean rdij] of turtles; sets mean-cs as the mean value for
all the turtles mean(rdij)
set mean-bs mean[bs] of turtles; sets mean-bs as the mean value for all the
turtles best strategy's rdij
set test fitness mean-cs; report the y=f(mean-cs)
set test-bs fitness mean-bs; report the y=f(best-cs)
tick
end

to each-strategy-test ; Test each turtle strategy
if ticks mod 1 = 0 [
set rdij lput (item 0 rdij) rdij
set rdij but-first rdij
set cs item 0 rdij

```

```

;show averageprofitrate
set u lput fitness cs u; return vector of fitnesses
]
end

to final-chromossomes;; Turtle procedure - defines new set of strategy
let u-max max (u); selects each turtle greatest fitness
let pmax position u-max u; returns the position of the greatest fitness value
set u remove-item pmax u; removes the greatest fitness value
set u fput u-max u; returns the gratest fitness to be the first item of the
  list of chromossomes
set mylist fput (item pmax mylist) mylist; gives the best fitness chromossome
  also the first position
set mylist remove-item (pmax + 1) mylist; removes the chromossome that was
  moved from the list
; write (ticks / 40) write "Geração: " write u
; write "Estratégias :" write mylist
set pg mylist; set pg as mylist(previous generation of chromossomes)
next-generation; procedure that defines the new generation
set u []; cleans the vector of values of fitness
end

to next-generation;; GenFirms procedure
set v [];cleans vector of result of crossover
let children []; set local variable of the offspring generated by the
  selected parents
let ulist (map + u n-values Cr [sqrt ( min(u) ^ 2 )]); sums the module of the
  minimum value of fitness to each value of fitness to possibilate each of
  the chromosses but the one with minor fitness to be chosen as parents
repeat Cr / 2[; Cr/2 crossovers happens
;let parent1 item (random 8) mylist; random selection of parent 1
;let parent2 item (random 8) mylist; random selection of parent 2
let hlist rnd:weighted-n-of-list 2 ulist [ [w] -> w ]; the extension of the
  roulette wheel, or fitness proportionate selection, using the ulist as a
  list of fitness to select 2 parents
let yposition random 2 ; random selection of the position of each parent in
  the crossover procedure
let parent1 item ( position (item yposition hlist) ulist ) mylist; the first
  parent is selected
let parent2 item (position (item ( 1 - yposition ) hlist) ulist ) mylist; the
  second parent is selected

```

```

ifelse (random-float 100.0 < 70.0);probability of success of crossover
  defined as 70
[
set children crossover parent1 parent2; set the crossover procedure for
  parent 1 and parent 2
set v sentence children v ; set the crossover result as the offspring
]
[set v lput parent1 v
set v lput parent2 v]]; crossover fails, the parents remains as the
  chromossmes
mutate; mutation occurs in the new Cr chromossomes created after the
  crossover loop
set ng mylist;set to verify if the crossover, mutation and elitism have
  happened, one has only to compare pg with ng
set rdij [];sets the values of rdij for the new generation
(
foreach mylist[
[ strategy ] ->
let k sum (map * strategy mylistP) / DELTA
set rdij lput (li + k * ib) rdij]
)
set bs item 0 rdij; selects generation best strategy
end

to-report crossover [strategy1 strategy2];; crossover procedure
let split-point 1 + random (length strategy1 - 1); selection of crossover
  split point
report list (sentence (sublist strategy1 0 split-point);generates list of two
  chromossomes that were binded at the split point
(sublist strategy2 split-point length strategy2))
(sentence (sublist strategy2 0 split-point)
(sublist strategy1 split-point length strategy1))
end

to mutate
;; GenFirms procedure
let h[]
let p[]
set p item 0 mylist; Elitism, the old chromossome with best fitness is
  selected to not suffer mutation
(

```

```
foreach v[; each gene of every chromosome is subjected to a mutation draw
[ strategy ] ->
let k map [i -> ifelse-value (random-float 100.0 < 3.0) [1 - i] [i]]
    strategy;mutation draw probability, success turns 0 in 1 or 1 in 0,
    failure the gene is the same
set h lput k h ]
)
set mylist h; set the new list of strategy as a result of the crossover and
mutation draws
set mylist replace-item 0 mylist p; returns the best strategy as the first
member of the list(Elitism)
end

to-report fitness [#x]; fitness
let f.x 0;local variable for f(x)
if equation = 1 [; first equation
set f.x ( 15 - ( #x ^ 2 ) )]
if equation = 2 [; second equation
set f.x (12 - 5 * #x ^ 4 - 9 * #x ^ 3 + 16 * #x ^ 2 + 15 * #x )
]
report f.x;returns f(x) depending on the equation that was selected
end
```

APÊNDICE B

Código em NetLogo® referente ao Protocolo ODD para os modelos propostos na seção 2.2:

```
globals [
;N Number of firms
price      ; Market price
totOutput  ; Total production of firms
; Indicators on economic and technological performance
meanA      ; Average productivity of firms
maxA       ; Maximal productivity of firms
minA       ; Minimal productivity of firms
meanProf-innov ; Average profit of innovators
meanProf-imit ; Average profit of pure imitators
profMargin ; Margin of profit
prodIN/prodIM ; Ratio between F.NW and Immitators productivities
capshare-in ; Capital share of innovators
capshare-gen ; Capital share of innovators
HK         ; Concentration index over capital
HQ         ; Concentration index over output
ain        ; Innovative R&D calibration parameter
aIM        ; Imitative R&D calibration parameter
timestep
cexpRD     ; cumulative expenses in RD
cexpRD.Gen ; cumulative expenses in expenses for genetics firms
DELTA     ; set of possible strategies
mylistP   ; Coefficient of genes
mean-A-NW ; mean [A] of F.NW
max-A-NW  ; max [A] of F.NW
mean-K-NW ; mean [K] of F.NW
max-K-NW  ; max [K] of F.NW
mean-LProfit-NW ; mean [Pi] of F.NW
max-LProfit-NW ; max [Pi] of F.NW
mean-rinf-NW ; mean [rinf] of F.NW
mean-Q-NW   ; mean [output] of F.NW
max-Q-NW   ; max [output] of F.NW
mean-A-Gen  ; mean [A] of F.Gen
max-A-Gen  ; max [A] of F.Gen
```

```

mean-K-Gen; mean [K] of F.Gen
max-K-Gen; max [K] of F.Gen
mean-LProfit-Gen; mean [Pi] of F.Gen
max-LProfit-Gen; max [Pi] of F.Gen
mean-rdij-Gen; mean [mean rdij] of F.Gen
mean-rinf-Gen; mean [rinf] of F.Gen
mean-Q-Gen; mean [output] of F.Gen
max-Q-Gen; max [output] of F.Gen
mean-A-RND; mean [A] of F.RND
max-A-RND;; max [A] of F.RND
mean-K-RND; mean [K] of F.RND
max-K-RND; max [K] of F.RND
mean-LProfit-RND; mean [Pi] of F.RND
max-LProfit-RND; max [Pi] of F.RND
mean-rinf-RND; mean [rinf] of F.RND
mean-Q-RND; mean [output] of F.RND
max-Q-RND; max [output] of F.RND
]
extensions [ rnd ]; Roulette Wheel Selection extensions
breed [F.Gen Gen]; uses AG investment
breed [F.NW NW]; uses NW investment
breed [F.RND RndS]; uses random investment
F.NW-own [
output ; output of the firm
A ; productivity of the firm
K ; capital stock of the firm
Pi ; gross profit of the firm
RD ; Total R&D investment of the firm
aInF ; Innovative R&D calibration parameter
aImF ; Imitative R&D calibration parameter
RDIn ; Innovative R&D investment of the firm
RDIM ; Imitative R&D investment of the firm
prodInnovate ; Productivity discovered thanks to the innovation
prodImitate ; Productivity discovered thanks to the imitation
pi_{j}; profit rate for N & Winter
rimf; current capital rate of investment in imitative R&D
rinf; current capital rate of investment in innovative R&D
invp; possible investment
RD0; initial investment
]
F.RND-own [

```

```

mylist; lst of strategies
pg ; previous generation
ng ; new generation after crossover
output ; output of the firm
A ; productivity of the firm
K ; capital stock of the firm
Pi ; net profit of the firm
RD ; Total R&D investment of the firm
aInF ; Innovative R&D calibration parameter
aImF ; Imitative R&D calibration parameter
RDIn ; Innovative R&D investment of the imitator will be 0
RDIM ; Imitative R&D investment of the firm
prodInnovate ; Productivity discovered thanks to the innovation
prodImitate ; Productivity discovered thanks to the imitation
pi_{j}; profit rate for N & Winter
rimf; current capital rate of investment in imitative R&D
rinf; current capital rate of investment in innovative R&D
invp; possible investment
RD0; initial investment
]
F.Gen-own [
y; vector of profit rates for every learning period in which a strategy is
    used(has length equal to learningperiods)
u; fitness vector (vector of profit rates averages for each generation
    strategy, length is equal to G)
v; auxiliary vector
mylist; lst of strategies
pg ; previous generation
ng ; new generation after crossover
output ; output of the firm
A ; productivity of the firm
K ; capital stock of the firm
Pi ; net profit of the firm
RD ; Total R&D investment of the firm
aInF ; Innovative R&D calibration parameter
aImF ; Imitative R&D calibration parameter
RDIn ; Innovative R&D investment of the imitator will be 0
RDIM ; Imitative R&D investment of the firm
prodInnovate ; Productivity discovered thanks to the innovation
prodImitate ; Productivity discovered thanks to the imitation
pi_{j}; profit rate for N & Winter

```

```

rimf; current capital rate of investment in imitative R&D
rinf; current capital rate of investment in innovative R&D
invp; possible investment
RD0; initial investment
rdij ; Innovative investment related to strategy
]
to setup
clear-all ; Clean the memory
if N = 2[
set rim 0.00143
set rin 0.0287
set K0 139.58]
if N = 4[
set rim 0.00112
set rin 0.0223
set K0 89.70]
if N = 8[
set rim 0.00102
set rin 0.0205
set K0 48.85]
if N = 16[
set rim 0.00099
set rin 0.0197
set K0 25.34]
if N = 32[
set rim 0.00097
set rin .0194
set K0 12.89]
; Innovation and imitation parameters set using the initial probabilities and
the initial values
;set N nbGen + nbInnov ; total number of firms
set nbGen N * %NGen / 100 ; Number of Genetic firms
set nbInnov N * (100 - %NGen) / 100 ; Total number of firms
if universe = 'NWvsAG'[
create-F.NW nbInnov ; Create the innovators
]
create-F.Gen nbGen ; Create the genetics
if universe = 'RNDvsA' [
create-F.RND nbInnov ; create random strategies
]
; Initial behaviour of firms for computing the ain and aim parameters

```

```

let output0 A0 * K0
let price0 invDemand ( N * output0 )
set price price0
;Setting these parameters at the firm level for each type of firm
if universe = 'NWvsAG' [
ask F.NW [
set A A0
set K K0 ;/ timefactor
set aInF aIn
set aImF aIm
set rimf rim
set rinf rin
]
]
if universe = 'RNDvsA' [
ask F.RND[
set A A0
set K K0 ;/ timefactor
set aInF aIn
set aImF aIm
set rimf rim
set rinf rin * random-float 1 * factor
]
]
ask F.Gen [
set A A0
set K K0
set aInF aIn
set aImF aIm
set rimf rim
let mylist1 n-values G [i -> i]
let mylist2 sort-by > mylist1
set mylistP (map ^ n-values G [2] mylist2); create list of genes
set DELTA sum mylistP ; Define number of possible strategies minus the 0
  strategie
set u []
set y []
set rdij []
set mylist n-values Cr [n-values G [random 2]]; create list of chromossomes
  for each Genfirm]
foreach mylist[

```

```

[ strategy ] ->
let x sum (map * strategy mylistP) / DELTA
set rdij lput x rdij]
set rinf rin * item 0 rdij
]
ask F.Gen [ set rinf rin * item 0 rdij * factor]
ask turtles [ set RD0 (rinf) * K0]
set cexprD sum [RD0] of turtles
set cexprD.Gen sum [RD0] of F.Gen
; Innovation and imitation parameters deducted from probabilities
let probInnov 0.125 * rin * K0
let probImitate 1.25 * rim * K0
set aIn probInnov / (rin * K0 );
set aIm probImitate / (rim * K0);
reset-ticks ; Initialize the ticks counter
set timestep 0
end

to go
if ticks > nbPeriods [stop]
; Computing the output of each firm
ask turtles [
set output A * K / timefactor
set pi_{j} price * A - c - rimf - rinf
set Pi pi_{j} * K / timefactor
]
;Computing the total output and the intraperiod price
set totOutput sum [output] of turtles
set price invDemand (totOutput * timefactor )
;Productivity indicators of the current period
set maxA max [A] of turtles
set meanA mean [A] of turtles
set minA min [A] of turtles
; capital share of innovators
set capshare-in sum [K] of F.NW / sum [K] of turtles
; capital share of genetics
set capshare-gen sum [K] of F.Gen / sum [K] of turtles
;Concentration indicators of the current period
let totCapital sum [K] of turtles
set HK (totCapital) ^ 2 / sum [K ^ 2] of turtles
set HQ (totOutput) ^ 2 / sum [output ^ 2] of turtles

```

```

;
ask F.Gen [ifelse ticks = 0 [set rinf item 0 rdij * rin * factor
set y lput pi_{j} y ][
ifelse ticks mod (Cr * learningperiod) = 0
[
each-strategy-test
final-chromossomes
][
each-strategy-test]
]
]
; Moving towards the next period (new productivity and capital stock)
ask turtles [

; Compute gross profits
set Pi (price * A - (c + rimf + rinf)) * K / timefactor

; Compute R&D investments for pure imitators and others
ifelse (breed = F.Gen)
[
set rinf item 0 rdij * rin * factor
set rimf rim
]
[if breed = F.NW[
set rinf rin
set rimf rim]
if breed = F.RND[
set rinf rin * (random-float 1) * factor
set rimf rim
]
]
;Cumulative expenditure on innovative R&D .
; Compute productivities discovered thanks to innovation and imitation
set prodInnovate innovation
set prodImitate imitation
; Set the new productivity of the firm
set A max (list A prodInnovate prodImitate)

; Compute the new capital stock of the firm
set K investK
;investment in innovative RD

```

```

set RD0 rinf * K / timefactor
]
indicators
tick
set timestep timestep + 1
end

; test each strategy for 5 timesteps
to each-strategy-test ; Test GenFirms strategies
if ticks mod learningperiod = 0 [
set rdij lput (item 0 rdij) rdij
set rdij but-first rdij
set rinf rin * item 0 rdij * factor
;show averageprofitrate
set u lput averagepi_{j} u
set y []
]
set rinf item 0 rdij * rin * factor
set y lput pi_{j} y
end

to-report averagepi_{j}
let average sum y / learningperiod
report average
end

to final-chromossomes;; GenFirms procedure
let u-max max (u)
let pmax position u-max u
set u remove-item pmax u
set u fput u-max u
set mylist fput (item pmax mylist) mylist
set mylist remove-item (pmax + 1) mylist
; write (ticks / 40) write "Geração:" write u
; write "Estratégias :" write mylist
set pg mylist
next-generation
set u []
end

to next-generation;; GenFirms procedure

```

```

set v []
let children []
let ulist (map + u n-values Cr [sqrt ( min(u) ^ 2 )])
repeat Cr / 2 [
;let parent1 item (random 8) mylist; random selection of parent 1
;let parent2 item (random 8) mylist; random selection of parent 2
let hlist rnd:weighted-n-of-list 2 ulist [ [w] -> w ]
let yposition random 2
let parent1 item ( position (item yposition hlist) ulist ) mylist
let parent2 item (position (item ( 1 - yposition ) hlist) ulist ) mylist
ifelse (random-float 100.0 < ProbX)
[set children crossover parent1 parent2
set v sentence children v]
[set v lput parent1 v
set v lput parent2 v]]
mutate
set ng mylist
set rdij []
(
foreach mylist[
[ strategy ] ->
let x sum (map * strategy mylistP) / DELTA
set rdij lput x rdij]
)
end

```

```

to-report crossover [strategy1 strategy2];; GenFirms procedure
let split-point 1 + random (length strategy1 - 1)
report list (sentence (sublist strategy1 0 split-point)
(sublist strategy2 split-point length strategy2))
(sentence (sublist strategy2 0 split-point)
(sublist strategy1 split-point length strategy1))
end

```

```

to mutate
;; GenFirms procedure
let h[]
let p[]
set p item 0 mylist
(
foreach v[

```

```

[ strategy ] ->
let k1 map [i -> ifelse-value (random-float 100.0 < ProbM) [1 - i] [i]]
    strategy
set h lput k1 h ]
)
set mylist h
set mylist replace-item 0 mylist p
end
; Imitation procedure
to-report imitation
; Draw a number between 0 and 1
let draw random-float 1.
; Compare it with the rescaled value of the R&D
ifelse draw <= rimf * aim * K / timefactor [report maxA][report 0.]
end

; Innovation procedure

to-report innovation
let draw random-float 1.
let mi (A0 * ((1 + alpha) ^ (ticks / timefactor)))
let sd STD
let zeta ln ( 1 + ( sd ^ 2 / mi ^ 2 ))
let M (ln mi) - ( zeta / 2 )
let S sqrt zeta
;comparado com o valor de investimento em P&D
ifelse draw <= ain * rinf * K / timefactor [report exp (random-normal M S)
][report 0.]
end
;Investment procedure
to-report investK
; Compute firms investment
let rho price * A / c
let s output / totOutput
let invd (1 + \delta / timefactor) - ( ((2 - s) ) / (rho * (2 - ( 2 * s)) )
); investimento desejavel
ifelse pi_{j} <= 0 [ set invp ( \delta / timefactor + pi_{j} ) ][ set invp (
\delta / timefactor + ((1 + bank) * pi_{j} ) )] ; possible
investment
;retorno do novo estoque de capital
report ((max (list (min (list invd invp)) 0)) * K + (1 - \delta / timefactor)

```

```

    * K )
; Compute the investment of the firm - must be positive
end

; The constant-elasticity inverse demand function
to-report invDemand [totQuant]

report Dem / (totQuant ^ Eta)
end

to parallel.universe

end

to indicators
set cexpRD cexpRD + sum [RD0] of turtles
set cexpRD.Gen cexpRD.Gen + sum [RD0] of F.Gen

;profit indicators
ifelse nbInnov != 0 [if universe = 'NWvsAG' [set meanProf-innov mean [Pi]
  of F.NW]
if universe = 'RNDvsA' [set meanProf-innov mean [Pi] of F.RND]] []
ifelse nbGen != 0 [set meanProf-imit mean [Pi] of F.Gen] []
; profit margin (percentage)
; set profMargin ([profit] of turtles * [capital] of turtles)
;ratio prod innovators/imitators
if universe = 'NWvsAG' [ifelse nbInnov != 0 [ifelse nbGen != 0 [set
  prodIN/prodIM mean [A] of F.NW / mean [A] of F.Gen] []] []]
if universe = 'RNDvsA' [ifelse nbInnov != 0 [ifelse nbGen != 0 [set
  prodIN/prodIM mean [A] of F.RND / mean [A] of F.Gen] []] []]

if universe = 'RNDvsA' [
if nbInnov != 0 [
set mean-A-RND mean [A] of F.RND
set max-A-RND max [A] of F.RND
set mean-K-RND mean [K] of F.RND
set max-K-RND max [K] of F.RND
set mean-LProfit-RND mean [Pi] of F.RND
set max-LProfit-RND max [Pi] of F.RND
set mean-rinf-RND mean [rinf] of F.RND
set mean-Q-RND mean [output] of F.RND

```

```
set max-Q-RND max [output] of F.RND
]
if nbGen != 0 [
set mean-A-Gen mean [A] of F.Gen
set max-A-Gen max [A] of F.Gen
set mean-K-Gen mean [K] of F.Gen
set max-K-Gen max [K] of F.Gen
set mean-LProfit-Gen mean [Pi] of F.Gen
set max-LProfit-Gen max [Pi] of F.Gen
set mean-rinf-Gen mean [rinf] of F.Gen
set mean-rdij-Gen mean [mean rdij] of F.Gen
set mean-Q-Gen mean [output] of F.Gen
set max-Q-Gen max [output] of F.Gen
]
]
if universe = ‘‘NWvsAG’’ [
if nbInnov != 0 [
set mean-A-NW mean [A] of F.NW
set max-A-NW max [A] of F.NW
set mean-K-NW mean [K] of F.NW
set max-K-NW max [K] of F.NW
set mean-LProfit-NW mean [Pi] of F.NW
set max-LProfit-NW max [Pi] of F.NW
set mean-rinf-NW mean [rinf] of F.NW
set mean-Q-NW mean [output] of F.NW
set max-Q-NW max [output] of F.NW
]
if nbGen != 0 [
set mean-A-Gen mean [A] of F.Gen
set max-A-Gen max [A] of F.Gen
set mean-K-Gen mean [K] of F.Gen
set max-K-Gen max [K] of F.Gen
set mean-LProfit-Gen mean [Pi] of F.Gen
set max-LProfit-Gen max [Pi] of F.Gen
set mean-rinf-Gen mean [rinf] of F.Gen
set mean-rdij-Gen mean [mean rdij] of F.Gen
set mean-Q-Gen mean [output] of F.Gen
set max-Q-Gen max [output] of F.Gen
]
]
end
```
