

CARLOS AUGUSTO BOSSOLANI

**Uma Análise Semiótica
da Evolução do Software**

Mestrado em Comunicação e Semiótica

Pontifícia Universidade Católica
São Paulo – 2007

CARLOS AUGUSTO BOSSOLANI

Uma Análise Semiótica da Evolução do Software

Mestrado em Comunicação e Semiótica

Dissertação apresentada à Banca Examinadora
da Pontifícia Universidade Católica de São
Paulo, como exigência parcial para obtenção
do título de MESTRE em Comunicação e
Semiótica, sob a orientação da Professora
Doutora Christine Greiner.

Pontifícia Universidade Católica
São Paulo – 2007

Banca Examinadora

Dedico este trabalho ao meu pai, que sempre me
incentivou e me apoiou incondicionalmente.

SUMÁRIO

Resumo	2
Abstract.....	3
Introdução.....	4
1. Apresentação do Problema	7
1.1. Um Pequeno Histórico Sobre o Nascimento da Idéia	7
1.2. Um Exemplo Hipotético do Efeito Baldwin em Ação	10
1.3. Nossa Aprendizado Também é Guiado.....	13
1.4. A Evolução do Software é Guiada	15
2. Um Esboço da Filosofia Evolucionista de Charles Sanders Peirce.....	18
2.1. As Categorias no Mundo dos Fenômenos	20
2.2. O Universo em Evolução.....	24
2.2.1. Existência e Lei são Reais	24
2.2.2. O Acaso é Real	26
2.2.3. O Continuum e a Primazia da Lei da Mente	28
2.3. Teleologia e Causa Final	32
2.4. Semiótica e as Formas de Representação	38
2.4.1. A Formação do Símbolo.....	41
2.5. Evolução, Linguagem e Mensagem.....	43
2.6. Evolução Teleológica	45
3. O Aspecto Teleológico da Evolução do Software	51
3.1. Entendendo a Máquina de Turing de Forma Mais Intuitiva.....	51
3.2. Uma Definição Mais Formal da Máquina de Turing	57
3.3. Software, Linguagens e Mensagens	63
3.4. Linguagem de Máquina	64
3.5. Os Vários Níveis do Software	67
3.6. A Evolução do Software é Guiada	69
4. Conclusão e Desenvolvimentos Futuros.....	76
5. Bibliografia.....	79

RESUMO

A motivação principal da presente dissertação foi a crescente popularização, entre os cientistas, de uma teoria de que a evolução biológica pode ser guiada pela evolução cultural. Conhecida como efeito Baldwin, ela afirma que, quando as coisas a serem aprendidas se tornam relativamente fixas, cria-se uma pressão evolutiva para que elas sejam incorporadas geneticamente, via seleção natural.

O trabalho partiu da constatação de que existem diversas ocorrências de um efeito similar ao efeito Baldwin, nas quais um nível evolutivo mais flexível guia a evolução de outro nível, menos flexível. O problema abordado pela pesquisa é a inexistência de uma explicação única para essas diversas ocorrências.

Dessas ocorrências, a evolução do software foi adotada como objeto de estudos do trabalho. Seu aspecto teleológico pode ser mais claramente apreciado ao se observar que a evolução das linguagens de programação é guiada pela evolução dos programas, e que todo programa é escrito utilizando uma linguagem de programação. Como os programas mudam mais rapidamente do que as linguagens, os mais bem sucedidos tendem a ser incorporados ou determinar que características devem estar presentes na próxima geração de linguagens, guiando assim sua evolução.

Diante da relação entre software e linguagem, o principal objetivo do trabalho é demonstrar que o aspecto teleológico da evolução do software decorre, principalmente, do papel que a semiose exerce nela. A pesquisa bibliográfica é utilizada como metodologia de trabalho, e é adotada como base teórica a semiótica e outros elementos da filosofia evolucionista de Peirce. O trabalho apresenta, primeiramente, os fundamentos teóricos, explicando como a evolução corresponde à ação do signo e porque esse processo é teleológico. Em seguida, aborda a evolução do software apresentando situações nas quais isso pode ser observado.

Ao apontar uma situação concreta na qual a evolução é guiada e ao demonstrar que isso se deve à ação do signo, o trabalho contribui para a área de comunicação e semiótica de duas maneiras: evidencia a relação entre computação e semiose e argumenta em favor da importância da ação do signo na evolução, chamando a atenção para o seu aspecto teleológico.

Palavras chave: semiótica, semiose, software, evolução, teleologia.

ABSTRACT

The main motivation of this work was the growing popularization, among scientists, of a theory which asserts that the biological evolution may be guided by the cultural evolution. Known as the Baldwin Effect, it asserts that when things to be learned become relatively fixed, this creates an evolutionary pressure for those things to be incorporated genetically, by natural selection.

The work started with the verification that there are several occurrences of an effect similar to the Baldwin Effect, where one, more flexible, evolutionary level guides the evolution of another level, less flexible. The problem approached by the research was the inexistence of a single explanation for these several occurrences.

Of these occurrences, the evolution of software was taken as the work's object of studies. Its teleological aspect can be more clearly appreciated by observing that the evolution of the programming languages is guided by the evolution of the programs, and that every program is written using a programming language. Since programs change more quickly as the languages, those more successful tend to be incorporated or to determine which characteristics should be present in the next generation of languages, guiding, in this manner, their evolution.

Given the relationship between software and language, the main objective of the work is to demonstrate that the teleological aspect of the software evolution originate, mainly, from the role played by semiosis in it. Bibliographical research is used as the methodology of work, and the theoretical background used is Semiotics and other elements of Peirce's evolutionary philosophy. The work presents, at first, the theoretical foundations, explaining how evolution corresponds to the action of sign and why this process is teleological. Next, it approaches the software evolution presenting situations where this can be observed. By pointing a concrete situation where evolution is guided, and by demonstrating what this is caused by the action of sign, the work has contributed to the area of communication and semiotics in two manners: it makes evident the relationship between computation and semiosis and argues in favor of the importance of the sign action in evolution, calling attention to its teleological aspect.

Key words: semiotics, semiosis, software, evolution, teleology.

INTRODUÇÃO

A semiótica é a ciência que estuda a ação do signo, ou seja, que estuda os processos de significação, representação e comunicação, de uma maneira geral. Embora as idéias relacionadas à semiótica existam desde Platão, foi a concepção moderna do filósofo americano Charles Sanders Peirce que deu a ela um caráter totalmente novo e mais abrangente. Para Peirce, a semiótica é também uma teoria evolutiva porque a ação do signo é um processo de crescimento contínuo, que é subjacente não somente à comunicação, mas também ao aprendizado e à evolução. Dessa maneira, a ação do signo não se limita à mente humana, ocorrendo em todos os níveis da realidade.

A proposta fundamental deste trabalho é que a ação do signo não é um processo cego, e a evolução, para Peirce, tem sempre um propósito, ou seja, é sempre um processo guiado. Sabemos que, hoje em dia, a grande maioria dos biólogos evolucionistas rejeita a idéia de que a evolução biológica possa ser guiada, embora isso nem sempre tenha sido assim.

Quando Charles Darwin criou sua teoria da evolução por seleção natural ele não conhecia os mecanismos por trás da hereditariedade. A idéia de Jean-Baptiste Lamarck de que as características adquiridas ao longo da vida de um animal podiam ser herdadas pelos descendentes, embora possa parecer absurda hoje em dia, não era totalmente descartada nem pelo próprio Darwin. O interessante é que, segundo Lamarck, a evolução poderia ser guiada porque ocorria pelo esforço, e o esforço sempre tem um propósito. No caso em que uma girafa, por exemplo, se esforça para alcançar as folhas no topo de uma árvore, e devido a esse esforço seu pescoço cresce, é fácil perceber como esse crescimento ocorreu guiado por um propósito. Se a característica adquirida pelo esforço é transmitida para os descendentes, é possível dizer que, ao longo do tempo, a evolução daquela característica foi, da mesma forma, guiada por um propósito.

Foram somente algumas décadas depois da publicação do trabalho de Darwin, que as idéias decorrentes do trabalho de Mendel, sobre a genética ser a base da hereditariedade, foram finalmente unidas à teoria da evolução. Hoje se sabe que um animal não pode transmitir para seus descendentes as características adquiridas por ele durante sua vida. Isso porque elas são expressões dos seus genes, e estes não podem ser modificados pelo esforço que ele exerce durante a vida. As mutações genéticas são eventos aleatórios muito raros. A evolução, portanto, é um processo de tentativa e erro, no qual os animais resultantes dessas raras mutações, que por acaso são mais adaptados ao seu ambiente, sobrevivem e transmitem seus genes para as gerações seguintes. Não há como prever que mutações ocorrerão e que

efeitos elas terão. Esse é um dos motivos pelo qual a maioria dos biólogos evolucionistas contemporâneos rejeita a idéia de que a evolução biológica possa ser guiada.

A despeito dessa rejeição, hoje em dia tem se popularizado entre os cientistas uma teoria não Lamarckista que afirma que a evolução biológica pode ser guiada. Ela foi proposta em 1896 por James Mark Baldwin, mas na época foi ignorada pela maioria dos seus contemporâneos. Tendo ficado conhecida como efeito Baldwin, ela sustenta que a evolução cultural ou o aprendizado pode guiar a evolução biológica no sentido que, quando as coisas a serem aprendidas se tornam relativamente fixas, isso cria uma pressão evolutiva para que essas coisas sejam incorporadas geneticamente, via seleção natural.

Este trabalho partiu da constatação de que algo similar ao efeito Baldwin ocorre em outras situações além da evolução biológica. Dessas situações, a que chamou mais a atenção foi a evolução do software, devido a existir uma relação íntima entre o software e a semiótica, que pode ser constatada no fato de que todo programa de computador é escrito utilizando uma linguagem.

A evolução do software é, portanto, concomitante com a evolução das linguagens com as quais se criam os programas. Existe uma hierarquia que vai desde linguagem de máquina, que é composta pelas instruções básicas do computador, até as linguagens de programação de mais alto nível, que se assemelham às linguagens humanas. Acontece que a evolução das linguagens de programação é guiada, e isso ocorre da seguinte maneira. Uma linguagem de programação permite que infinitos programas sejam escritos. Quando determinados programas escritos em uma linguagem se tornam relativamente fixos, isso cria uma pressão para que os elementos desses programas sejam fixados na próxima geração de linguagens de programação. Assim, fazendo uma analogia com o efeito Baldwin, poderíamos dizer que a evolução das linguagens de programação equivale à evolução biológica, e a dos programas à evolução cultural.

Dessa maneira, embora os cientistas contemporâneos argumentem que a evolução por seleção natural seja cega, o efeito Baldwin apresenta uma hipótese forte de que a evolução pode ser guiada, e algo similar pode ser observado em outras situações, especialmente na evolução do software. Se diferentes formas de evolução apresentam características similares, deve existir um substrato comum a todas elas, e uma explicação única para esse fenômeno. Defenderemos, ao longo do trabalho, que o substrato comum a todas as formas de evolução é o signo, e que a semiótica, e outros elementos da filosofia evolucionista de Peirce, podem fornecer a explicação para esse fenômeno.

Estudaremos especificamente o caso em que a evolução do software é guiada, como a semiótica e a filosofia de Peirce podem nos ajudar a compreender esse fato, e como isso pode lançar luz em outras situações similares. Vale ressaltar a importância de estudar outros elementos do pensamento de Peirce além da semiótica. Isso se deve ao fato de que ele desenvolveu, ao longo de sua vida, uma filosofia que ele chamou de arquitetônica, ou seja, uma filosofia que foi organizada como uma grande obra de arquitetura, na qual alguns elementos formam a base para outros. Dessa maneira, não poderíamos nos restringir à semiótica prescindindo de outros elementos fundamentais de seu pensamento, pois todos se relacionam e se reforçam mutuamente. Assim, traçaremos um esboço da sua filosofia como um todo, destacando a semiótica e sua participação no aspecto guiado da evolução.

Para atingir nosso objetivo este trabalho está organizado da seguinte maneira. O primeiro capítulo apresenta um pequeno histórico de como a idéia principal deste trabalho surgiu. É discutido com maior detalhe o efeito Baldwin, e são fornecidos alguns exemplos de situações nas quais é possível identificar que o software evoluiu de maneira guiada. O segundo capítulo trás um esboço dos principais elementos do pensamento de Peirce que serão importantes para entender por que a evolução é guiada. O terceiro e último capítulo inicia com uma explicação mais detalhada sobre o que são realmente o computador e o software, abordando, em seguida, sua relação com a semiótica. Finalmente é discutida a questão da evolução do software ser guiada, sob a luz do pensamento peirceano.

1. APRESENTAÇÃO DO PROBLEMA

1.1. Um Pequeno Histórico Sobre o Nascimento da Idéia

A melhor forma de compreender a idéia principal deste trabalho é conhecendo, primeiramente, a ocasião na qual ela surgiu. Isso se deu em um período no qual eu trabalhava no projeto de um software de controle em tempo real para um aparelho de TV. Na ocasião eu estava especificamente ocupado com alguns módulos do software que permitiriam que o usuário alterasse os atributos da imagem como brilho, contraste, cor e nitidez. É bastante óbvio que esses atributos devem ser flexíveis, para que cada usuário da TV os configure de acordo com sua preferência, e, portanto, não podem ser determinados no momento da elaboração do software.

O que me fez refletir, na verdade, e que acabou sendo o germen da idéia central desse trabalho, foi outro requisito que tinha sido determinado para esse software, que pedia que além dos controles separados para cada atributo da imagem, três configurações de imagem pré-programadas deveriam estar presentes no software: “padrão”, “sinal fraco” e “vídeo game”. Ao escolher uma dessas configurações, através do controle remoto, ajustes pré-programados de brilho, contraste, cor e nitidez seriam aplicados de uma só vez. Essa funcionalidade tinha como objetivo facilitar a vida do usuário da TV, que ainda poderia mudar livremente cada atributo da imagem, mas, se quisesse ganhar tempo, poderia simplesmente escolher uma das configurações pré-programadas, e os atributos fixados nela seriam aplicados imediatamente.

A idéia que me ocorreu na ocasião é que não teria sido possível criar essas configurações pré-programadas de imagem nos primeiros aparelhos de TV que foram construídos. O problema é que não há como saber, de antemão, quais são os valores ideais a serem fixados nessas configurações. A configuração pré-programada “vídeo game”, por exemplo, não poderia ser prevista pelos engenheiros que projetaram a primeira TV, antes de existir o vídeo game. Para determinar os valores dos atributos, é necessário, além da flexibilidade de poder mudá-los facilmente, ter tempo para que padrões se estabeleçam.

Dessa forma, embora os valores dos atributos que seriam fixados no software que eu desenvolvia tivessem sido fornecidos para mim pelos engenheiros que projetaram a TV, sem que eles me dissessem o critério utilizado, é possível ter uma idéia de como se chegou a esses valores. Se o objetivo é facilitar a vida dos usuários, os valores das configurações devem agradar a maioria deles. Podendo mudar separadamente, a qualquer momento, cada atributo

da imagem, os usuários puderam determinar de forma coletiva, ao longo do tempo, quais seriam os valores ideais desses atributos. Os valores mais utilizados, ou mais populares, são os melhores candidatos a ser fixados nas opções pré-programadas. É evidente que isso depende, dentre outras coisas, das características do sinal de TV transmitido (se é por ar ou por cabo, se é fraco ou forte), do que tipo de programas as pessoas assistem mais frequentemente (jogos de futebol, filmes etc.), se elas jogam vídeo game ou assistem a filmes através do vídeo cassete ou aparelho de DVD, etc. Como todas essas eventualidades não podem ser previstas, e nem é possível determinar sequer que surgirão padrões estáveis. Os atributos devem ser flexíveis primeiramente, e somente em uma segunda etapa, caso padrões se estabelecerem, podem ser fixados para facilitar a vida do usuário.

Podemos perceber, nesse caso, que a utilização do software pelos usuários influencia a programação deste pelo programador. A diferença é que os usuários “mudam” o software da TV muito mais rapidamente, e o programador, por sua vez, não tem essa flexibilidade. As pessoas não costumam comprar aparelhos de TV com muita freqüência e uma vez que os aparelhos de TV são fabricados com uma versão do software, este não poderia mais ser alterado. Portanto, uma nova versão desse tipo de software pode demorar anos para ser feita, enquanto que os usuários da TV podem mudar a qualquer momento os atributos da imagem da TV. Como veremos mais adiante, um processo similar a este é característica da evolução de qualquer tipo de software, de uma forma ou de outra. Nesse processo existem pelo menos dois níveis de flexibilidade e as mudanças no nível mais flexível guiam as mudanças no nível mais rígido.

Uma vez que comecei a pensar nesse fato, várias situações evidenciaram a sua pertinência. Uma delas me ocorreu, certa vez, ao ir à academia onde havia uma esteira eletrônica dessas modernas. Esta esteira, além de poder ser “manualmente” utilizada, ou seja, de permitir que o usuário altere sua velocidade, inclinação e tempo de funcionamento, também tinha quatro treinos pré-programados, que eram automaticamente iniciados ao simples toque de um botão. Não pude deixar de pensar que a forma que os usuários utilizavam a esteira poderia servir de feedback, mesmo que indiretamente, para o programador que fixaria os treinos pré-programados no seu software de controle.

Outra ocorrência é a do software do forno de microondas. Os primeiros fornos somente podiam ser acionados para funcionar com uma determinada potência por um determinado tempo. Hoje é raro encontrar um forno de microondas que não tenha um botão “pipoca”. Alguns deles já têm receitas requintadas pré-programadas em seu software. Imagino que antes que os usuários pudessem experimentar preparar todo tipo de alimento nos

fornos de microondas, não poderia se ter certeza que a pipoca seria um alimento tão popularmente preparado a ponto de merecer um botão exclusivo.

Não estou querendo dizer, com estes exemplos, que o usuário final sempre é quem determina que elementos devam ser pré-programados. Quero chamar a atenção somente para o fato que a flexibilidade aumenta a velocidade de descoberta dos elementos que posteriormente podem ser pré-programados, ou fixados.

Esse fenômeno, portanto, aparenta ser bastante comum no mundo do software. Foi grande meu entusiasmo ao descobrir que ele ocorre em lugares e situações que eu nunca havia imaginado. Existe uma hipótese evolucionista não Lamarckista que afirma que o aprendizado ou a plasticidade fenotípica dos seres vivos pode guiar e acelerar a evolução biológica. Foi James Mark Baldwin (1896) que, sentindo-se oprimido pela idéia de que a teoria de Darwin eliminava a participação na mente na evolução dos organismos, “resolveu demonstrar que os animais, por força de suas próprias atividades inteligentes no mundo, poderiam acelerar ou guiar ainda mais as evoluções que ocorriam em sua espécie” (DENNETT, 1998, p. 81). Essa teoria ficou conhecida como efeito Baldwin.

A primeira vez que eu li sobre o efeito Baldwin foi em um livro do filosofo Daniel Dennett (*Consciousness Explained*, 1992). Embora o tema fosse outro – o fenômeno da consciência –, no sétimo capítulo Dennett propõe uma hipótese de como a consciência pode ter evoluído, e apresenta o efeito Baldwin como um fator importante nessa evolução. Ele começa por argumentar que a Natureza, ou mais especificamente, o processo de evolução por seleção natural, cria soluções relativamente fixas para problemas que não mudam muito. Desde muito antes do surgimento da vida na terra, por exemplo, a gravidade já exercia sua força, fato que nunca mudou. Desta forma, a natureza pôde nos equipar com detectores de “qual é o lado de cima” baseados na gravidade.

Entretanto, existem coisas que mudam muito rapidamente. Nesses casos, nenhum projeto pré-estabelecido pode lidar com todas as eventualidades. Os melhores organismos serão aqueles que puderem se adaptar ou se re-projetar (re-programar) em tempo hábil. Dennett chama todos os processos de re-projeto, que ocorrem após o nascimento, de fixação de projeto pós-natal.

Às vezes esse re-projeto é chamado de aprendizado e às vezes é chamado apenas de desenvolvimento [...] vamos chamar qualquer processo desse tipo de [...] fixação de projeto pós-natal. [...] Quando você nasce, ainda existe espaço para variação, que eventualmente é fixada por um processo ou outro em um elemento relativamente permanente de projeto para o resto da sua vida. (DENNETT, 1992, p. 183).

De acordo com ele, esse processo de fixação de projeto pós-natal só pode ser algo análogo ao processo de fixação de projeto pré-natal, ou seja, um processo de seleção natural ocorrendo dentro do indivíduo, um processo evolutivo que ocorre no cérebro. Segundo ele o primeiro novo meio da evolução é o processo de fixação de projeto pós-natal. O que realmente chamou minha atenção foi o fato de que essa “evolução” que ocorre no cérebro pode acelerar a evolução genética:

[...] essa capacidade, ela mesma produto da evolução genética por seleção natural, não apenas proporciona aos organismos que a possuem uma vantagem sobre seus primos “hard-wired”¹ que não podem se re-projetar, mas também reflete no próprio processo de evolução genética e o acelera. Esse é um fenômeno conhecido sob vários nomes, mas melhor conhecido como o efeito Baldwin. (DENNETT, 1992, p. 184).

Uma explicação sucinta do efeito Baldwin é dada abaixo:

[...] se o aprendizado ajuda na sobrevivência então os organismos mais aptos a aprender vão ter mais descendentes, aumentando assim a freqüência dos genes responsáveis pelo aprendizado. E se o ambiente permanecer relativamente fixo, então as melhores coisas a serem aprendidas permanecerão constantes, isso pode levar, via seleção, a uma codificação genética dos traços que tinham que ser aprendidos. (MITCHELL, 1998, p. 88)

Eu não pude deixar de relacionar o efeito Baldwin com o fenômeno que ocorre na evolução do software como foi exposto acima. Da mesma forma como o aprendizado acelera a evolução biológica, a programação (ou a utilização) do software pelo usuário serve de feedback e acelera a programação do mesmo pelo programador. Mas, para que possamos entender melhor essa relação devemos, antes de tudo, compreender melhor o efeito Baldwin. Para isso vamos analisar um exemplo hipotético.

1.2. Um Exemplo Hipotético do Efeito Baldwin em Ação

Vamos imaginar que uma determinada espécie de animal vive em um ambiente onde existem, além de outras fontes de alimento, duas plantas muito parecidas, cujo fruto serve de alimento para os animais dessa espécie. Apesar de semelhantes, as plantas guardam duas diferenças fundamentais. A primeira diferença é que uma das plantas tem folhas verdes e a outra tem folhas vermelhas. A outra diferença, que é muito mais importante, é que os frutos das plantas de folhas vermelhas são ligeiramente tóxicos, e podem até causar a morte se ingeridos com muita freqüência. Entretanto, os frutos da planta de folhas verdes, por sua vez, além de não serem tóxicos são uma boa fonte de nutrientes para esses animais.

¹ O termo “hard-wired” geralmente se refere a um programa ou dispositivo que foi criado de forma que não pode ser mudado facilmente.

Existe no cérebro dos animais dessa espécie uma rede neural específica, responsável pela escolha da planta cujos frutos serão comidos, baseada na cor das folhas. Essa rede neural tem dez conexões que determinam seu funcionamento, sendo que cada conexão é determinada por um gene. Cada um desses genes vai determinar se a conexão estará ligada ou desligada e, portanto vai poder ter dois alelos, “desligado” ou “ligado”, que serão representados, respectivamente, por D e L. Vamos supor que o animal que evita frutos tóxicos, ou seja, que evita as plantas de folhas vermelhas será aquele que tenha o seguinte genótipo: LLLLLDDDDD. Todos os outros animais nascem sem “saber” evitar a planta tóxica. Exemplos de genótipos desses outros animais seriam LLLDDLDLDD, LLLLLLLL, LDLDLDLDLD.

Fica evidente que os animais que evitam os frutos tóxicos têm maiores chances de sobrevivência. Mas fica claro também que possuir essa capacidade é algo extremamente improvável. Podemos criar um gráfico que é conhecido como “paisagem adaptativa” para melhor visualizar essa situação. Nesse gráfico, a ordenada representa a chance de sobrevivência ou aptidão e a abscissa representa os diferentes genótipos. Ao observar o gráfico (Figura 1) percebemos que os indivíduos com os genes para evitar a planta tóxica são como “agulhas no palheiro”, ou melhor, como “postes em uma clareira”.

Se compararmos o gráfico a uma montanha, é muito difícil para a natureza “escalar” a “montanha adaptativa” porque o animal tem ou não a capacidade de evitar plantas tóxicas. A “agulha no palheiro” ou “poste na clareira” representa a única possibilidade de se ter maiores chances de sobrevivência. Para essa configuração ser alcançada seria necessário que todos os dez genes fossem “descobertos” de uma só vez pela Natureza. Mesmo que ela fosse descoberta ela não seria facilmente passada para os seus descendentes. Se um indivíduo que tenha atingido essa configuração acasala-se com um indivíduo que não seja quase idêntico a ele, essa configuração será provavelmente perdida.(HINTON; NOWLAN, 1987).

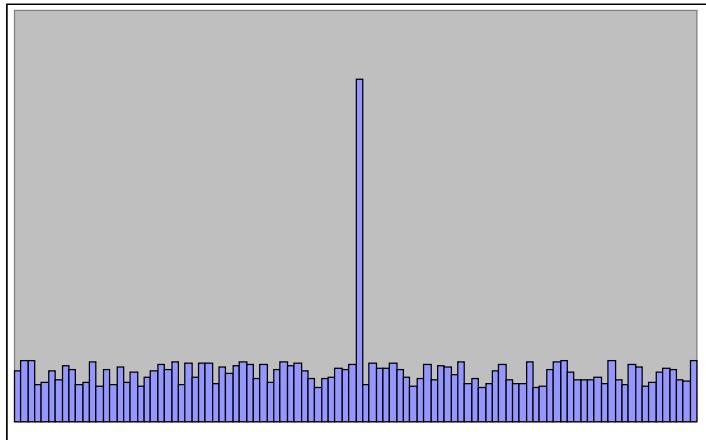


Figura 1

Consideremos uma outra possibilidade. Imagine que os genes que determinam o comportamento de escolha da planta, ao invés de terem apenas dois alelos, agora têm também um terceiro, que determinará que a respectiva conexão não esteja pré-determinada no momento do nascimento, mas sim possa ser ligada ou desligada durante a vida do animal, através de um processo de aprendizado. Dessa forma, o animal pode ingerir frutos tóxicos e aprender com o erro, fixando assim essas conexões flexíveis. Esse terceiro alelo será representado por “?”. Assim, podemos ter, por exemplo, um animal com o seguinte genótipo: LLLL?DDDD. Isso significa que a conexão no cérebro desse animal, determinada pelo quinto gene, não foi geneticamente ligada nem desligada, mas é flexível para ser ligada ou desligada de acordo com o processo de aprendizado.

Essa flexibilidade ou plasticidade fenotípica permite que existam agora diferentes níveis de chances de sobrevivência e não apenas dois como anteriormente. O indivíduo pode nascer mais “longe” ou mais “perto” da configuração que faz com que ele evite plantas tóxicas. Por exemplo, o indivíduo ???????D, tem que aprender nove conexões, enquanto que o indivíduo LLLLL???? tem que aprender apenas cinco. A paisagem adaptativa, sob essas condições, fica muito mais suave. (Figura 2).

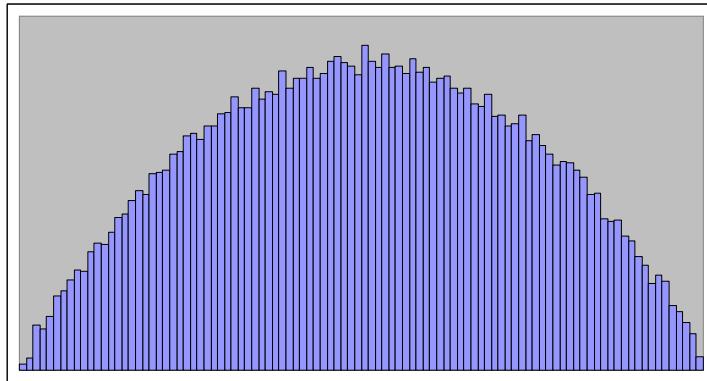


Figura 2

Desse modo, aqueles que estão mais “perto” da configuração que confere a capacidade de evitar plantas tóxicas tendem a ser favorecidos pela seleção natural e a próxima geração tende a ter mais indivíduos que estejam mais perto do alvo. Após várias gerações a tendência é que os animais já nasçam com todas as conexões fixadas geneticamente. Podemos dizer que quando algo que precisa ser aprendido não muda ao longo do tempo, como o caso da planta tóxica, cria-se uma força seletiva para “transferir” para os genes uma informação que faz parte do aprendizado. Quando isso ocorre, dizemos que a evolução é guiada.

Graças ao efeito Baldwin, podemos dizer que as espécies testam previamente a eficácia de um projeto particular diferente por exploração fenotípica (individual) do espaço de possibilidades próximas. Se uma configuração vencedora em particular é descoberta, isso vai criar uma nova pressão seletiva: organismos que estão mais próximos dessa descoberta na “paisagem adaptativa” vão ter uma vantagem clara sobre aqueles que estão mais distantes. Isso significa que **espécies com plasticidade tendem a evoluir mais rapidamente (e com uma “visão mais clara”) do que aqueles sem ela**. Então, evolução no segundo meio, plasticidade fenotípica, pode melhorar a evolução no primeiro meio, variação genética. (DENNETT, 1992, pp. 186-187, grifo nosso).

De posse de uma maior compreensão sobre o funcionamento do efeito Baldwin, podemos avaliar melhor como ele se assemelha ao processo através do qual o software evolui ao longo do tempo. Mas, antes disso, gostaríamos de fornecer um argumento a mais de que o processo em questão é mais abrangente do que ele é considerado.

1.3. Nossa Aprendizagem Também é Guiada

O efeito Baldwin descreve como dois níveis evolutivos diferentes interagem, sendo que a evolução do nível mais flexível guia a do nível mais rígido. Podemos afirmar que o aprendizado em si, é um tipo de processo evolutivo, considerando, é claro, as devidas diferenças. O que queremos salientar agora é que aquilo que chamamos de aprendizado, pelo

menos em nossa espécie, também possui no mínimo dois níveis de flexibilidade. Nós somos possuidores de pelo menos dois tipos de memória: memória de curto prazo e memória de longo prazo. O fato que queremos evidenciar é que esses dois níveis também interagem de forma similar à descrita pelo efeito Baldwin.

Os cientistas acreditam que as nossas memórias são organizadas no cérebro como mudanças nas conexões sinápticas entre os neurônios, sendo que essas mudanças podem ocorrer de três formas diferentes. Na primeira forma, novos neurônios podem ser recrutados das áreas generativas do cérebro para participar da memória. Na segunda forma, novas conexões sinápticas entre neurônios existentes são formadas, o que modifica a topologia da rede. Na terceira forma, há uma mudança na força das conexões sinápticas existentes, e nenhuma nova ligação é envolvida. Esta força entre as conexões pode ser entendida como a propensão de um neurônio ser disparado ou não em resposta a um estímulo enviado por outro neurônio. (AUNGER, 2002, pp. 189-190).

Embora os dois primeiros métodos sejam fisiológicos, ou seja, necessitem do crescimento de novas células ou partes delas, já o terceiro mecanismo – a habilidade do neurônio de alterar a força das suas conexões sinápticas – não requer, a princípio, síntese de proteína. Na verdade, “o conceito de memória de curto prazo é definido pela ausência de síntese de proteína, que é a consequência normal da expressão do gene” (AUNGER, 2002, p. 189).

A memória de curto prazo, obviamente, tem uma plasticidade muito maior que a de longo prazo. As alterações das forças das conexões sinápticas acontecem em um período de tempo da ordem do milisegundo. E baseados em tudo que se sabe até agora, é altamente improvável que a expressão do gene se dê nessa grandeza de tempo.

Embora não haja síntese de proteína nas alterações das memórias de curto prazo, a memória de curto prazo e a de longo prazo estão relacionadas. As mudanças que ocorrem nas forças das conexões sinápticas, que são responsáveis pela memória de curto prazo, disparam um processo de síntese de proteína localizado em cada sinapse, que consolidam essa memória em memória de longo prazo. Ainda que não esteja dentro do escopo desse trabalho fornecer detalhes de como este processo ocorre, sabemos que esse processo de consolidação da memória de longo prazo é “guiado” pela memória de curto prazo.

Mesmo que eles sejam tipos de resposta bastante distintos – uma elétrica, a outra estrutural – memória de curto prazo e memória de longo prazo estão ligadas. Qualquer resposta neural de curto prazo provavelmente dispara alguma cascata de consolidação de memória de longo prazo, o que sugere que as respostas de curto prazo e de longo prazo estão integradas. As mudanças de curto prazo marcam o traço da memória, o qual o mecanismo de longo prazo então fixa no

lugar. A memória de curto prazo pode não depender de genes durante o instante da marcação, mas ela depende dos genes para produzir seus mecanismos. E então a transcrição do gene é necessária para consolidar seus ganhos. **Portanto memória de longo prazo é dependente das mudanças de curto prazo na plasticidade neural.**(AUNGER, 2002, p. 192, grifo nosso).

Vemos que algo similar ao efeito Baldwin acontece também no próprio processo de aprendizado, tendo as mudanças que ocorrem em curto prazo sendo fixadas em uma etapa posterior. Diante disso temos fortes motivos para acreditar que o fenômeno em questão não se restringe à evolução biológica, e deve ser mais abrangente. Após essa exposição, e munidos de uma compreensão maior sobre o efeito Baldwin, podemos finalmente discutir melhor sua relação com a evolução do software.

1.4. A Evolução do Software é Guiada

Da mesma forma que a evolução por seleção natural agindo no nível genético não poderia prever todas as eventualidades, dizemos que o programador do software do aparelho de TV, por exemplo, não poderia prever quais seriam os valores ideais de brilho, contraste, cor e nitidez que deveriam ser fixados em uma configuração pré-programada. Nem o programador do software da esteira eletrônica teria como saber de antemão quais seriam os treinos mais populares que poderiam ser pré-programados. O programador do software do forno de microondas também não tinha como saber que a pipoca seria um alimento tão popularmente preparado nele.

Todos esses projetos se beneficiaram em se tornar flexíveis para que o usuário determinasse que informações seriam fixadas em uma segunda etapa, acelerando o processo. Estamos fazendo uma analogia da relação do aprendizado e a evolução biológica com os padrões de utilização dos aparelhos e o software que os controlam. Podemos dizer dessa forma que a evolução do software da TV, do forno de microondas e da esteira são guiadas pela utilização do usuário.

Mas no que diz respeito ao software, podemos ir muito além. Quando falamos de software estamos falando de vários níveis que estão interligados e que mudam em velocidades diferentes. Podemos até dizer que o software é um aspecto mais flexível do computador e que a evolução do computador é guiada pela evolução do software. Ao dizer isso, queremos dizer que novos processadores não são criados aleatoriamente, mas sim que sua criação é fortemente influenciada (guiada) pelo software que “rodava” nas versões anteriores dos processadores. É fácil entender como isso ocorre. Sendo o software mais flexível, ele muda mais rapidamente, assim é de se esperar que a sua evolução seja mais rápida. Na medida em

que determinadas soluções em software ganham estabilidade e param de mudar, se cria um estado de coisas, no qual são privilegiados novos processadores que incorporem, implementem ou facilitem a utilização dessas soluções já estabelecidas.

Esse mesmo mecanismo pode ser observado na relação entre os diferentes níveis do software. Podemos destacar a evolução das linguagens de programação guiada pela evolução dos programas com elas escritos. As linguagens de programação tendem a incorporar ou a sintetizar, em seus elementos, programas inteiros ou trechos deles que se tornaram estáveis ao longo do tempo. Para ilustrar, vamos imaginar uma linguagem de programação qualquer. Dos infinitos programas possíveis de serem escritos com ela, imaginemos que os que se tornem mais comuns sejam aqueles que são escritos para acessar grandes bancos de dados. Cria-se, nesse momento, uma pressão muito forte para que novas versões dessa linguagem incorporem, em si, comandos que simplifiquem a criação de programas para acessar bancos de dados. Nesse sentido, podemos dizer que a evolução dos programas guia a evolução das linguagens.

Os próprios programas, por sua vez, tendem a incorporar, em novas versões, comandos simples que sintetizam padrões de utilização desses programas, estabelecidos pelos seus usuários. Exemplificando, se das diversas formas diferentes de se utilizar um programa, três se mostram, ao longo do tempo, as mais utilizadas, cria-se uma forte tendência para que as novas versões do mesmo programa apresentem atalhos ou formas mais simples e mais rápidas de realizar essas três tarefas mais comuns. Foi o que tentamos demonstrar nos exemplos dos softwares da TV, da esteira eletrônica e do forno de microondas.

Vemos que o nível mais flexível sempre guia a evolução do nível mais rígido. As diferentes formas de se utilizar um programa mudam mais rapidamente que os programas, guiando assim sua evolução. Os diferentes programas escritos, por sua vez, mudam mais rapidamente que as linguagens de programação, guiando sua evolução, sendo que estas, por sua vez, mudam mais rapidamente que os sistemas operacionais que por sua vez mudam mais rapidamente que o processador, e assim por diante.

Na evolução do software, portanto, podemos facilmente identificar um fenômeno similar ao descrito pelo efeito Baldwin. A razão pela qual isso acontece é, justamente, o problema abordado por este trabalho. Em outras palavras, o principal objetivo deste trabalho é descobrir por que a evolução do software é guiada, de forma similar à descrita pelo efeito Baldwin.

O fato de características similares ao efeito Baldwin se mostrarem presentes em várias situações diferentes, como as que foram mostradas até agora, leva a crer que existem leis mais

gerais que descrevem características comuns a toda forma de evolução. Essas leis devem explicar por que a evolução é guiada nesses casos que foram salientados, inclusive na evolução do software. Um dos objetivos deste trabalho é demonstrar que essas leis podem ser encontradas na semiótica e em outras idéias evolutivas do filósofo americano Charles Sanders Peirce. Para ele, a evolução age nos níveis mais básicos da realidade, e toda regularidade do universo é resultado dela. O substrato comum de todas as formas de evolução é o signo, e decorre de suas idéias que a evolução é intrinsecamente guiada. Assim sendo, conhecer melhor essas idéias é a nossa primeira tarefa.

2. UM ESBOÇO DA FILOSOFIA EVOLUCIONISTA DE CHARLES SANDERS PEIRCE

Peirce aprendeu, através da leitura de Immanuel Kant, que a filosofia deve ser como uma obra arquitetônica, ou seja, como uma construção onde certas partes servem de base para outras. Foi com isso em mente que ele desenvolveu a filosofia na qual nós pretendemos encontrar subsídios para entender porque a evolução é guiada. Dessa maneira, se torna muito difícil estudar apenas um de seus elementos isoladamente, e, portanto, apresentaremos um esboço das principais idéias, salientando sua inter-relação.

Antes de iniciar essa tarefa, é importante fornecer argumentos de que a filosofia de Peirce contém aquilo que procuramos. A existência da interação entre os diferentes níveis da evolução, e o fato de esse fenômeno ser constatado em diversas situações, levam a concluir duas coisas. A primeira é que deve existir um substrato comum a todas as formas de evolução. A segunda é que não devem existir barreiras entre os diferentes níveis evolutivos. Devemos, portanto, buscar nas idéias de Peirce pelo menos esses dois elementos.

O substrato comum da evolução é o signo, e a semiótica – ciência que estuda a ação do signo – irá fornecer as informações necessárias para compreendermos o que há em comum a todas as formas de evolução. Para que isso fique mais claro, devemos apreciar, mesmo que de forma superficial no momento, a relação da semiótica com as categorias fundamentais do pensamento de Peirce.

Peirce concordava com Kant que deve existir um número pequeno de categorias fundamentais do pensamento, e, independentemente de qual seja esse número, sua definição deve ser o primeiro passo na construção de uma filosofia arquitetônica. Ele também concordava com Kant que se existem quaisquer conceitos fundamentais que possam dar unidade a todo conhecimento, eles serão derivados da lógica. Entretanto, ele mantinha que as categorias deviam ser mais fundamentais que a própria lógica. Esse aparente paradoxo foi resolvido quando ele separou a *lógica da matemática* da *lógica propriamente dita*. (PARKER, 1963, p. 28)

Na lógica da matemática, Peirce viu uma “ciência a priori que descobre todas as formas possíveis de *relação*, sem se preocupar se essas formas se manifestam no fenômeno” (PARKER, 1963, p. 4). Nela ele descobriu três formas possíveis de relação – monádica, diádica e triádica –, que fornecem os conceitos formais para as três categorias fundamentais, que são a base de todo o seu pensamento: primeiridade, segundidade e terceiridade.

Com a descoberta dos relativos monádicos, diádicos e triádicos na lógica da matemática, nós temos os conceitos formais das três categorias cenopitagóricas. A questão da sua aplicabilidade material fica para ser abordada na fenomenologia; sua necessidade e suficiência como categorias lógicas deve esperar análise na parte matemática da lógica propriamente dita. (PARKER, 1963, p. 43).

Peirce, portanto, reduziu as doze categorias de Kant em apenas três, com as quais ele construiu toda sua filosofia arquitetônica. Isso é importante, no momento, devido ao fato de que a idéia de evolução já está presente logo nas categorias fundamentais. Como veremos com mais detalhes ao longo do trabalho, a terceira categoria é a do crescimento, da evolução.

Ao aplicar suas categorias à lógica, Peirce obteve uma teoria mais abrangente do que a lógica tradicional de Kant, que, segundo ele, deixava de fora formas fundamentais do raciocínio. De acordo com Peirce existem três formas elementares de raciocínio, que derivam, por sua vez, de princípios mais gerais da representação (PARKER, 1963, p. 7). Assim, a lógica para ele, deveria ser a filosofia da representação. Sendo a semiótica a ciência responsável por estudar todas as formas de representação, ele concluiu que semiótica era outro nome para lógica. (PARKER, 1963, p. 134)

Levando em consideração que a terceira categoria é a categoria do crescimento, a representação é encarada por Peirce como processo e não uma relação estática. Esse processo corresponde à ação inteligente do signo. Ele é contínuo, ou seja, estende-se infinitamente no tempo, e cada uma de suas partes pode ser infinitamente dividida. Peirce chamou esse processo de semiose. Queremos salientar, e isso ficará mais claro ao longo do trabalho, que a semiose corresponde ao próprio processo pelo qual se dá a evolução. Acontece que para Peirce a semiótica tinha um status ontológico, e disso decorre que a semiose não ocorre somente nas mentes das pessoas, mas ocorre na realidade. Assim, toda forma de evolução, que possa haver, é um tipo de semiose. O signo ou representação é, portanto, o substrato comum da evolução.

A idéia de que não existem barreiras entre os diferentes níveis evolutivos também está presente na filosofia de Peirce. Partindo do princípio de não aceitar coisas inexplicáveis como possíveis explicações, ele defende que a todas as coisas devem ser consideradas contínuas sempre que possível. Decorre desse princípio que não existe separação entre a mente e a matéria e sim um *continuum*. Segundo esse mesmo princípio, se deve esperar que não existam reais barreiras entre os diferentes níveis evolutivos. É exatamente isso que se afirma ao se alegar que a evolução é semiose, e que esta, por sua vez, é um processo contínuo.

Veremos ao longo do trabalho que a idéia de semiose e de continuidade são a base para compreender o que é uma ação guiada. Para Peirce, dizer que uma ação é guiada

equivale a dizer que ela tende a um fim ou tem uma *causa final*. Nesses casos também se pode chamá-la de *teleológica*. Para ele, todo evento tem um aspecto teleológico, e a evolução é intrinsecamente guiada.

Diante do que foi dito, já é possível traçar um plano de estudos deste segundo capítulo. Em primeiro lugar abordaremos sucintamente a fenomenologia de Peirce. Nela encontraremos uma explicação mais acessível e intuitiva das categorias, que servirá de base para a discussão que se segue. Em seguida, estudaremos alguns aspectos importantes da sua metafísica. Tomaremos conhecimento de que o universo para Peirce é o resultado de um processo evolutivo ainda em andamento. Veremos que a evolução se dá a partir do acaso aliado a uma tendência à aquisição de hábitos. A questão da continuidade será, então, abordada, e discutiremos algumas de suas implicações para a filosofia de Peirce, e sua relação com a evolução.

O próximo passo será entender o que é uma ação guiada. Veremos que Peirce resgata a idéia de causa final de Aristóteles, e que todo evento para ele tem um elemento de acaso, de necessidade e de finalidade. Prosseguiremos estudando a semiótica e o processo de semiose, analisaremos a relação entre evolução, linguagem e mensagem. Finalmente relacionaremos todos esses assuntos, demonstrando que uma ação guiada depende da semiose. Veremos que evolução para Peirce é intrinsecamente teleológica, e ao final teremos subsídios suficientes para entender como a evolução do software pode ser guiada.

2.1. As Categorias no Mundo dos Fenômenos

Algum tempo após ter derivado logicamente as categorias, Peirce passou a buscar suas ocorrências no mundo dos fenômenos. A fenomenologia é a primeira das ciências da filosofia, sendo também chamada de faneroscopia ou doutrina das categorias. O termo faneroscopia se refere ao estudo do faneron que, segundo Peirce, é:

[...] o total coletivo de tudo aquilo que está de qualquer maneira ou em qualquer sentido presente na mente, sem nenhuma consideração se isso corresponde a qualquer coisa real ou não. (CP, 1.284).

O objetivo dessa ciência é fazer um inventário de todas as nossas experiências, classificando-as em categorias. Ela não tem como objetivo estudar a realidade, mas somente aquilo que aparece em nossos sentidos, independente de qualquer realidade subjacente. Assim sendo, ela não requer nenhuma arte observacional especial, tendo como matéria prima a

própria experiência cotidiana de qualquer indivíduo. Uma lembrança, um sentimento, algo que vimos ou escutamos, tudo isso serve de matéria prima para a fenomenologia, que não faz distinção entre os mundos interior e exterior.

Peirce acabou chegando à conclusão que todos os modos de ser da experiência podem ser reduzidos a apenas três categorias universais, que ele havia identificado anteriormente, que são irredutíveis e suficientes, e as quais, como sabemos, ele chamou de: primeiridade, segundidade e terceiridade. Assim, sua fenomenologia mostra que as relações formais estudadas na lógica da matemática têm correlatos materiais na experiência. Tudo que há no faneron, portanto, deve pertencer a pelo menos uma dessas categorias universais.

Passemos a considerar cada uma das categorias separadamente para melhor compreendê-las. Comecemos pela primeira categoria. As experiências de primeiridade são caracterizadas pela novidade, liberdade e simplicidade. O que é primeiro é o que é sem considerar nenhuma relação com outra coisa. Ele é simples e não tem partes. É o estado de consciência no qual se experiencia uma mera qualidade de sentimento, como a qualidade de ser vermelho, independente de qualquer ocorrência dessa qualidade.

Aquela mera *qualidade*, ou *talidade*, não é em si mesma uma ocorrência, como é ver um objeto vermelho; ela é um mero *pode-ser*. Seu único ser consiste no fato de que poderia existir tal peculiar e positiva talidade em um faneron. (CP, 1.304)

Podemos dizer que um primeiro é um estado.

Um sentimento, então, não é um evento, uma ocorrência, um acontecimento, uma vez que um acontecimento não pode ser o que é, a menos que existisse um tempo em que ele não aconteceu; e, assim, ele não é em si mesmo tudo a que ele é, mas é relativo a um estado prévio. Um sentimento é um *estado*, que assim é em sua totalidade, em todo momento de tempo e na medida em que ele dure (CP, 1.307).

A palavra “primeiro” sugere que não há outro que o determina. Assim, o que é primeiro é livre, ou seja, é uma mera possibilidade. Ele pode ser instanciado em alguma ocorrência, mas, enquanto primeiro, ele deve ser considerado aparte de qualquer ocorrência.

Nas experiências de primeiridade, não há consciência de passagem do tempo, elas são absolutamente presentes, sem considerar o passado nem o futuro. Aquilo que está imediatamente presente na consciência, portanto, corresponde à primeiridade. Se quiséssemos analisar o conteúdo imediato da consciência perderíamos a “presentidade” requerida pela primeira categoria, pois a “análise envolve comparação com uma experiência pretérita” (IBRI, 1992, p.10)

As experiências classificadas sob a segunda categoria são caracterizadas por sentimentos de resistência, dualidade, reação, que estão presentes em situações nas quais forçamos algo que resiste a nós, como quando tentamos abrir uma porta trancada, por exemplo.

Estamos continuamente nos chocando contra o fato duro. Esperávamos uma coisa, ou passivamente a tínhamos por certa, e tínhamos sua imagem em nossas mentes, mas a experiência força essa idéia ao chão e nos compõe a pensar muito diferentemente. (CP, 1.324)

Para ilustrar melhor podemos pensar na situação onde desejamos que faça um tempo ensolarado, para que possamos fazer um passeio e, ao olhar pela janela, constatamos que está chovendo. Esse é o fato duro ao qual Peirce se refere. Chove independentemente de nosso desejo de que não chova. O fato é segundo em relação ao nosso ego e por isso essa experiência é classificada sob a segunda categoria. Nesse tipo de experiência, temos a sensação de que as coisas não são o que queremos que elas sejam, sentimos uma oposição, que faz com que o ego reaja contra aquilo que não é ego, o não-ego. Não somos capazes de fazer parar de chover apenas com a nossa volição. Por isso, pode-se dizer que a experiência é o pivô do pensamento, pois ela impõe seu modo de ser a ele.

Uma experiência de segundadade é sempre individual, ela acontece apenas uma vez, não podendo ser repetida. Ela ocorre aqui e agora. “Se ela é repetida, a repetição é outra ocorrência, não importando quão parecida com a primeira ela possa ser” (CP, 7.538). Ao querermos sol e constatarmos a chuva, sentimos a dureza do fato, mas ao percebermos que a chuva continua, temos a consciência de uma experiência que já não faz mais parte de segunda categoria. Nesse caso já estaríamos formando um conceito ao longo do tempo. Como dissemos, as experiências da segundadade são individuais, acontecem aqui e agora.

Sob a segunda categoria também estão todas as experiências passadas. A mesma sensação de resistência que sentimos quando constatamos algo que reage à nossa vontade temos ao considerar uma experiência passada. Por mais que quiséssemos não poderíamos mudar o passado. Ele é segundo em relação ao nosso ego, resistindo à nossa vontade de que ele mude.

As experiências que pertencem à terceiridade são aquelas nas quais há mediação entre dois elementos, ou seja, um terceiro aproxima um primeiro a um segundo. Nesse tipo de experiência há consciência de síntese, de aprendizado. Assim, sempre quando percebemos que dois fenômenos fazem parte de um mesmo conceito, temos a consciência desse tipo de experiência. Nas palavras de Ibri: “[p]arece haver na mente uma tendência à generalização

que busca subsumir ao conceito um número maior de fenômenos, tornando-o, por isso, mais geral.” (IBRI, 1992, p. 14). Dessa forma, estão relacionadas à terceiridade as idéias de generalidade, infinitude, continuidade, difusão, crescimento e inteligência.

De acordo com Peirce, mediar ou estabelecer uma relação entre dois elementos, através de um terceiro, é equivalente à representação.

Agora Terceiridade não é nada além do que o caráter de um objeto que incorpora a Qualidade de Estar Entre (Betweenness) ou mediação na sua forma mais simples e rudimentar; e eu a uso como o nome daquele elemento do fenômeno que é predominante onde quer que Mediação seja predominante, e que atinge sua plenitude na **Representação**. (CP, 5.104, grifo nosso)

Terceiridade, como eu uso o termo, é apenas um sinônimo para **Representação** [...] (CP, 5.105, grifo nosso)

A terceira categoria é a idéia daquilo que é tal qual é por ser um Terceiro, ou Meio entre um Segundo e seu Primeiro. Isso é o mesmo que dizer que ele é **Representação** como um elemento do Fenômeno. (CP, 5.66, grifo nosso)

Fica evidente que as experiências de terceiridade envolvem fluxo de tempo. Mas, de acordo com Peirce, toda consciência de fluxo do tempo é uma experiência de terceiridade. Assim sendo, “Todo fluxo de tempo envolve aprendizagem; e toda aprendizagem envolve fluxo de tempo.” (CP, 7.536).

Para ilustrar, podemos voltar ao exemplo da chuva. Ao olharmos pela janela e constatarmos a chuva, temos uma experiência de segundidate, pois o fato é segundo em relação ao nosso ego. Mas, ao perceber a permanência desse fato, como quando verificamos que a chuva **continua**, temos uma experiência de terceiridade. Isso porque, para nós, a chuva atual é a “mesma” que a chuva anterior. Em nossa mente é criada uma representação mediadora que aproxima a chuva de agora com a de antes.

Já é possível estabelecer a relação entre as categorias na fenomenologia. No mundo dos fenômenos, o fluxo do tempo se traduz no aprendizado, ou seja, na aquisição de representações mediadoras que subsumem ao conceito vários fenômenos individuais. As experiências desse tipo são classificadas sob a terceira categoria. Já as experiências de cada fenômeno individual são de segundidate, pois são ocorrências que acontecem apenas uma vez, aqui e agora. Cada uma delas instancia uma qualidade que, considerada por si só, independente da ocorrência, é uma experiência de primeiridade.

Como dissemos, a fenomenologia estuda somente o que aparece nos fenômenos, e não o mundo como ele é. Nossa próximo passo será conhecer a teoria da realidade de Peirce, que revelará uma realidade na qual toda a regularidade é resultado da evolução.

2.2. O Universo em Evolução

A metafísica é a ciência que estuda a realidade subjacente aos fenômenos, ou seja, que estuda o mundo como ele é, para que ele apareça da forma como a fenomenologia o apreende. Veremos que na filosofia de Peirce existe uma simetria entre o mundo como fenômeno e o mundo como realidade. Essa simetria é devida ao fato de que as categorias fundamentais não são apenas categorias da experiência, mas são também categorias da realidade.

2.2.1. Existência e Lei são Reais

Peirce era um realista e não acreditava que a realidade é uma criação da mente. Nesse sentido ele discordava fundamentalmente de Kant. Para Peirce, “[o] real é aquilo que não é o que eventualmente dele pensamos, mas que permanece não afetado pelo que possamos dele pensar” (CP, 8.12). Assim sendo, o real se distingue da ficção ou do sonho por não ser uma invenção nossa, ou seja, por não existir somente porque nós o imaginamos. Dessa maneira, o elemento de alteridade, característico da segunda categoria, é predominante na idéia de realidade, embora não corresponda à totalidade dela.

Na idéia de realidade, a Segundidate é predominante; pois o real é aquilo que insiste em forçar seu modo de ser à recognição como alguma outra coisa que não a criação da mente. (CP, 1.325).

Vimos na fenomenologia que as experiências subsumidas à segunda categoria eram caracterizadas por uma reação contra a consciência, um não-ego reagia contra um ego. Algo resistia a nossa vontade. O passo dado por Peirce para estabelecer um equivalente no mundo real foi considerar que as coisas além de reagirem contra nosso ego, reagem entre si. Existência, para Peirce, é reação, e as coisas existem porque reagem entre si:

Quando dizemos que uma coisa *existe*, o que queremos dizer é que ela reage sobre outras coisas (CP, 7.534)

O que quer que exista, *ex-sists*, isto é, realmente age sobre outros existentes, assim obtém uma auto-identidade e é definitivamente individual. (CP, 8.191).

Existência é aquele modo de ser que reside em oposição a outro. Dizer que uma mesa existe é dizer que ela é dura, pesada, opaca, ressonante, ou seja, produz efeitos imediatos sobre os sentidos e, também, que produz efeitos puramente físicos, atrai a terra (isto é, é pesada), dinamicamente reage contra outras coisas (isto é, tem inércia), resiste à pressão (isto é, é elástica), tem uma definida capacidade para o calor, etc. Dizer que existe uma mesa fantasma a partir de sua incapacidade de afetar quaisquer sentidos ou de produzir quaisquer efeitos físicos que sejam, é falar em uma mesa imaginária. Uma coisa sem oposições, ipso facto não existe. (CP, 1.457).

Portanto, existência corresponde à realidade da segunda categoria. Para Peirce o que existe é real, pois é o que é independente de nosso pensamento. Existe, pois reage contra nosso ego e contra as outras coisas existentes.

Entretanto, existência é reação, e uma reação é algo individual que ocorre aqui e agora. Quando algo dura no tempo já perde o status de individual. “[...] a permanência de uma reação fá-la descharacterizar-se como tal por se tornar uma regularidade *no tempo*. ” (IBRI, 1992, p.30). Para Peirce, à regularidade das ocorrências no tempo equivale a realidade da terceira categoria, como afirma Ibri:

[...] decorre daí que a terceira categoria, na sua fundação ontológica e não mais fenomênica apenas, entretece-se numa regra objetiva que subsume os individuais que lhe são correlatos, ou seja, numa regularidade real que se mantém alter para consciência. (IBRI, 1992, p. 31).

De acordo com Peirce, existência sem regularidade é puro caos. As coisas do mundo real para serem objetos de nossa representação mediadora devem ter o mínimo de permanência ou regularidade no tempo. Não podemos ter um conceito, ou uma representação geral, de algo absolutamente efêmero e que não se repete. O simples fato de sermos capazes de formarmos representações gerais da realidade se deve à presença da regularidade nessa realidade. Ou seja, “[...] a generalidade exterior parece ser o fundamento da generalidade do pensamento como representação mediadora” (IBRI, 1992, p. 30).

Os gerais do mundo exterior correspondem a tudo aquilo que não depende de nosso pensamento e tem regularidade: o espaço, o tempo, as leis da natureza. Ao abandonarmos uma pedra no ar, por exemplo, sabemos que ela irá de encontro ao chão. Existe uma regularidade. A lei da gravidade é um geral, pois subsume todas as ocorrências individuais de pedras que caem ao chão ao serem abandonadas no ar. Ela é real, pois é alter, ou seja, não é uma criação do nosso pensamento.

Assim sendo, o conceito de real de Peirce contempla tanto as ocorrências individuais quanto as leis gerais. Como vimos, as ocorrências individuais são caracterizadas pelas reações que ocorrem aqui e agora, e são classificadas sob a segunda categoria. As leis são os gerais que regem essas ocorrências e são classificadas sob a terceira categoria. Entretanto, Peirce não acreditava que todas as ocorrências são estritamente determinadas pela lei. O elemento de primeiridade presente na fenomenologia tem seu equivalente do mundo exterior na realidade do acaso.

2.2.2. O Acaso é Real

A discussão a respeito de o universo ser ou não estritamente regido pela lei é uma questão tão antiga quanto a própria filosofia. O determinismo, de uma forma simples, é a tese de que existe, em qualquer instante, exatamente um futuro fisicamente possível, ou seja, que tudo que ocorre no universo ocorre devido a uma lei, e todo acontecimento é necessário e não poderia ser diferente.

A maioria dos cientistas pós-renascentistas acredita nessa tese. Para eles o universo é como um relógio e a tarefa do cientista é tão somente descobrir os mecanismos de funcionamento desse relógio, ou seja, as leis que regem o funcionamento do universo. Einstein, por exemplo, nunca admitiu a realidade do acaso, e costumava dizer que “Deus não joga dados com o universo”. Seu pensamento pode ser identificado na famosa passagem de Pierre Simon Laplace:

Um intelecto que a qualquer dado momento conhecesse todas as forças que animam a Natureza e as posições mútuas dos seres que a compreendem, se esse intelecto fosse vasto o suficiente para submeter seus dados à análise, poderia condensar em uma única formula o movimento do maior dos corpos do universo e aquele do menor dos átomos: para tal intelecto nada poderia ser incerto; e o futuro assim como o passado seriam presentes diante de seus olhos (LAPLACE 1814, apud DENNET, 2004, p. 28).

Peirce rejeitava a idéia de que o universo é estritamente regido pela lei. Para ele o acaso é real, e ele apresentou fortes argumentos para defender essa tese. Em nossa opinião, o mais forte deles é o seguinte:

Quando tenho perguntado a pensadores qual a razão que tiveram para crer que todo fato no universo é precisamente determinado pela lei, a primeira resposta tem usualmente sido que a proposição é uma "pressuposição" ou postulado do raciocínio científico. Bem, se isto é o melhor que pode ser dito a respeito do assunto, a crença está condenada. Suponhamo-la "postulada": tal não a torna verdadeira nem tampouco fornece o menor motivo racional para conferir-lhe qualquer credencial. É como se alguém viesse a pedir emprestado dinheiro e, quando solicitada uma garantia, ele replicasse que "postulou" o empréstimo. (CP, 6.39).

A pergunta de Peirce é muito simples mas esmagadora: que motivo nós temos para acreditar em um universo estritamente regido pela lei? Para ele, admitir isso como sendo uma pressuposição não pode ser aceitável, pois estabelece uma barreira no caminho do conhecimento. Não podemos simplesmente aceitar que existem leis da física que são eternas e imutáveis, e que regem o funcionamento do universo nos seus mínimos detalhes, sem nos questionarmos sobre a origem dessas leis. Isso seria ignorar o problema e não resolvê-lo. Peirce acreditava em um universo cognoscível, ou seja, acessível ao nosso conhecimento.

Sendo a lei uma regularidade no tempo, ela não é aquilo que há de mais simples, e, portanto, demanda uma explicação. Mas, por outro lado, aquilo que é absolutamente simples, desprovido de qualquer lei ou de qualquer regularidade, não precisa de explicação. Peirce, portanto, acreditava em um elemento de ausência de lei, de liberdade, que antecede a lei e até mesmo a existência. Ele acreditava na realidade do acaso.

Para entendermos o acaso como realidade, podemos pensar em uma situação onde jogamos um dado de seis faces. Antes de jogarmos o dado, não temos como saber qual das faces irá cair para cima. A chance de cada face cair para cima é igual, ou seja, elas são equiprováveis. Podemos dizer, então, que o acaso é “o modo de ser de uma distribuição fortuita, a exemplo daquela obtida em qualquer experimento equiprovável, como o é um jogo de dados” (IBRI, 1992, p. 39). Dizemos que nessa situação o lance de dados é “ocasionado” pelo acaso pois não há influência de nenhum evento passado ou futuro.

[...] há independência entre cada um dos resultados, fazendo com que um evento particular não decorra do anterior, nem forneça condições para a definição do próximo. (IBRI, 1992, p. 39)

Se todas as ocorrências são igualmente possíveis, nenhuma é mais provável do que a outra, ou seja, há liberdade por não haver necessidade. O acaso, portanto, está associado à idéia de possibilidade, liberdade e espontaneidade. Ele corresponde, objetivamente, à primeira categoria.

Podemos agora entender melhor a relação que existe entre as categorias na metafísica Peirceana. Vemos que aquilo que existe, enquanto segundo, tem um elemento de liberdade que vem do acaso, enquanto primeiro, e não é totalmente regido pela terceiridade da lei, como argumentariam os deterministas.

Como um princípio objetivo, ele [o acaso] subsume a diversidade e variedade da natureza, fazendo com que a segundidade do fato não seja estritamente regida pela terceiridade da lei; a existência possui, assim, um elemento de espontaneidade, conferido pela primeiridade do acaso. (IBRI, 1992, p. 40).

A citação acima suscita uma questão. Embora Peirce acreditasse na realidade do acaso, ele também acreditava na realidade das leis. Mas se as leis não são eternas e imutáveis, qual é a sua origem? Para Peirce, as leis são o resultado de um processo evolutivo.

Que espécie de explicação pode, então, haver? Respondo que podemos esperar por uma explicação evolucionária. Podemos supor que as leis da natureza são resultados de um processo evolucionário . (CP, 7.12)

As leis só podem ter se originado de um estado no qual não havia leis, apenas existência cega, determinada pelo acaso. Peirce afirma que existe uma tendência universal de todas as coisas a adquirir hábitos. Esse é o próprio mecanismo da evolução: acaso, existência cega e uma tendência de aquisição de hábitos.

Mas, se as leis da natureza são resultados da evolução, esta evolução deve proceder de acordo com algum princípio; e este princípio será, em si mesmo, da natureza de uma lei. Porém, ele deve ser uma lei que pode evoluir ou se desenvolver por si mesma [...] Evidentemente ela deve ser uma tendência à generalização – uma tendência generalizadora [...] Contudo, a tendência generalizadora é a grande lei da mente, a lei de associação, a lei de aquisição de hábitos [...] Assim, sou levado à hipótese de que as leis do universo têm sido formadas sob uma tendência universal de todas as coisas à generalização e à aquisição de hábitos. (CP, 7.515).

Essa tendência à aquisição de hábitos também é uma lei, entretanto, ela própria só pode ter surgido por acaso. Ela tem a propriedade de reforçar a si própria: “[...] subjacente a todas as outras leis está a única tendência que pode crescer pela sua própria virtude, a tendência de todas as coisas a adquirir hábitos”. (CP, 6.101).

Como vimos, o evolucionismo de Peirce deriva do juízo de que não podemos estabelecer uma barreira no caminho do conhecimento. Admitir uma lei que exista sem ter sido o resultado da evolução seria estabelecer essa barreira. A idéia de evolução, portanto, está relacionada à idéia de que existe uma continuidade entre nós e o objeto que desejamos conhecer, ou seja, eles são co-naturais. A idéia de continuidade, como veremos, é uma das idéias de maior importância na filosofia de Peirce. Como foi dito no início deste capítulo, buscamos na sua filosofia a idéia de que não existem barreiras entre os diferentes níveis evolutivos. Assim sendo, no próximo tópico abordaremos, na filosofia de Peirce, o conceito de continuidade e a sua relação com a evolução.

2.2.3. O Continuum e a Primazia da Lei da Mente

Para entender a importância da idéia de continuidade no pensamento de Peirce, devemos entender, antes de tudo, o que é um continuum para ele. A definição dada por ele foi formada a partir das definições de Kant e Aristóteles. Kant estabelece a relação entre o continuum e a infinitude dizendo que o continuum é aquilo que é infinitamente divisível. Entretanto, para Peirce, a infinita divisibilidade é uma condição necessária mas não suficiente: “[q]uebrando grãos de areia mais e mais, irá apenas torná-la mais fragmentada. Não irá fundir os grãos numa continuidade não rompida.” (CP, 6.168.) Peirce adiciona, então,

à idéia de continuum de Kant, a definição de Aristóteles – “um *continuum* é alguma coisa cujas partes têm um limite comum.” (CP, 6.122) Para Peirce, portanto, o “*continuum* é alguma coisa infinitamente divisível cujas partes têm um limite comum.” (IBRI, 1992, p. 66.)

O exemplo mais comum de continuum é uma linha, que é um continuum devido a **poder** ser dividida em **infinitas** partes, que têm um limite comum. Entretanto, vale ressaltar duas questões. Em primeiro lugar, dissemos que ela **pode** ser dividida, mas ao identificarmos um ponto em uma linha estamos justamente quebrando a continuidade, estabelecendo uma singularidade tópica, e destituindo o caráter de continuidade.

[...] uma linha, por exemplo, não contém nenhum ponto até que a continuidade seja quebrada por marcar os pontos. Assim, parece necessário afirmar que um *continuum*, onde ele é contínuo e não fragmentado, não contém partes definidas; que suas partes são criadas no ato de defini-las e a sua precisa definição quebra a continuidade. (CP, 6.168)

Outro ponto para salientarmos é que dissemos que uma linha tem **infinitas** partes, e isso é importante. Segundo Peirce, nenhuma pluralidade de individuais pode esgotar um continuum, que é o modo de ser do todo e não de suas partes. O continuum está, portanto, associado à idéia de generalidade, pois aquilo que é geral não tem partes distintas. Assim sendo, nenhum conjunto finito de pontos equivale a uma linha.

É possível perceber a relação do continuum com a primeira categoria. Quando definimos o acaso, falamos que ele é o modo de ser de uma distribuição equiprovável, como aquela dos possíveis resultados de um jogo de dados. O que vale ressaltar a respeito desse modo de ser, é que ele é geral, não existindo maneiras de identificar um caso individual.

*O possível é necessariamente geral; e nenhuma quantidade de especificação geral pode reduzir uma classe geral de possibilidades a um caso individual. É apenas a atualidade, a força da existência, que irrompe a fluidez do geral e produz uma unidade discreta. Desde Kant, a idéia de que o tempo e o espaço introduzem continuidade na natureza tem sido bastante difundida. Mas isto é um *anacoluthon*. Tempo e espaço são contínuos porque incorporam condições de possibilidade, e o possível é geral, e continuidade e generalidade são dois nomes para a mesma ausência de distinção de individuais.* (CP, 4.172)

No viés da generalidade, podemos associar a idéia de continuidade também à terceiridade. Como vimos na fenomenologia e na metafísica, a terceiridade corresponde à formação de conceitos ou hábitos gerais que sintetizam ocorrências singulares. Peirce afirma que: “[a] continuidade representa a Terceiridade quase à perfeição” (CP, 1.337).

Para entender isso melhor, tomemos um conceito geral, como a idéia de árvore, por exemplo. Esse conceito, sendo geral, está associado à terceiridade. Notemos que ele não diz respeito a uma única árvore. Ele não pode ser reduzido a nenhuma ocorrência individual pois

não fornece descriminação suficiente para determinar uma árvore específica. Se eu digo “árvore” eu não estou dizendo se ela é grande ou pequena, se tem frutos, ou se tem folhas ásperas. O conceito também não pode ser esgotado por nenhuma pluralidade de individuais. Mesmo que se conheçam todas as árvores do mundo, ainda é possível, através de uma mutação genética, nascer uma árvore absolutamente nova, que ainda sim será chamada de “árvore”. Dessa forma, infinitas árvores estão contidas no conceito, e elas estão relacionadas umas às outras justamente por fazerem parte do mesmo conceito. Repare que o conceito pode ser infinitamente dividido também. Podemos tomar um sub-conjunto de todas as árvores grandes, e dentro desse conjunto, tomar o sub-conjunto de todas as árvores grandes mas com folhas pequenas e, assim, infinitamente.

Levando em consideração que a idéia de continuidade está presente logo nas categorias fundamentais, é de se esperar que ela seja uma idéia central no pensamento de Peirce. Ele defendia que se deve considerar as coisas contínuas sempre que possível, chamando de Sinequismo a tendência em considerar a continuidade como uma idéia de fundamental importância para a filosofia (CP, 6.169).

Se o continuum representa a terceira categoria quase à perfeição, e a terceira categoria está associada à evolução, então existe uma íntima relação entre evolução e continuidade. Como vimos, a idéia de uma realidade em evolução decorre, principalmente, do fato de Peirce não aceitar que existam coisas sem explicação. O sinequista, portanto, não aceita que a existência de coisas inexplicáveis seja uma explicação possível.

[...] sinequismo equivale ao princípio de que coisas inexplicáveis não serão consideradas como possíveis explicações; que o que quer que seja suposto como final é suposto ser inexplicável (CP, 6.173)

Para que algo tenha explicação, deve haver um continuum entre nós e o objeto que queremos entender. De forma similar, essa explicação demanda que o objeto seja resultado da evolução, caso contrário ele será inexplicável. Assim sendo, a evolução é também um postulado lógico:

Evolução é um postulado da lógica, em si mesma; pois o que é uma explicação senão a adoção de uma suposição mais simples para descrever um estado de coisas complexo. (W 4, p. 547, apud REYNOLDS, 1966, p. 18)

Uma decorrência importante da idéia de continuidade e evolução para filosofia de Peirce se observa na abordagem que ele dá à antiga questão da relação entre mente e matéria.

Ele sustenta que entre mente e matéria não há separação, mas apenas um continuum, que as mantém co-naturais. Consideremos a seguinte passagem:

A velha noção dualística de mente e matéria, tão proeminente no cartesianismo, como dois tipos de substância radicalmente diferentes, dificilmente hoje irá encontrar defensores. Rejeitando-a, somos levados a alguma forma de hiliozísmo, e o caso contrário denominado monismo. Surge, então, a questão se, de um lado, a lei física e, de outro, a lei psíquica devem ser consideradas:

- a) como independentes, constituindo uma doutrina freqüentemente chamada *monismo*, mas que eu denominaria *neutralismo*; ou
- b) a lei psíquica como derivada e especial, e apenas a lei física como primordial, o que é *materialismo*; ou
- c) a lei física como derivada e especial, e somente a lei psíquica como primordial, o que é *idealismo*. (CP, 6.24).

Sabemos que o dualismo não encontra defensores por, dentre outras coisas, não fornecer explicação para como a mente consegue interagir com a matéria. Já o neutralismo, para Peirce, “é suficientemente condenado pela máxima lógica conhecida como a navalha de Ockham, isto é, que não devem ser supostos mais elementos independentes que o necessário.” (CP, 6.24)

O materialismo, como queremos salientar, também não poderia ser aceito por Peirce, em decorrência do evolucionismo e do sinequismo. Isso porque a matéria tem o modo de ser de uma lei, e as leis, segundo Peirce, são produtos da evolução. A única lei que não é produto da evolução é a tendência à aquisição de hábitos, que surgiu do acaso e tem primazia em relação às outras leis. Não admitindo um elemento de incognocibilidade no universo, ele não poderia aceitar a matéria, ou qualquer elemento último, como base de todas as coisas. Isso fica evidente na passagem abaixo.

O sinequista, por exemplo, nunca estaria satisfeito com a hipótese de que a matéria é composta de átomos, todos esféricos e exatamente iguais. Se essa é a única hipótese com a qual os matemáticos estão, por enquanto, em condições de lidar, deve ser suposto que ela deve ter traços de semelhança com a verdade. Mas nem a eternidade dos átomos nem sua precisa semelhança é, na visão do sinequista, um elemento de hipótese que seja admissível mesmo hipoteticamente. **Pois isso seria tentar explicar o fenômeno por meio de uma absoluta inexplicabilidade.** (CP, 6.173)

Dessa maneira, para Peirce: “A única teoria inteligível do universo é a do idealismo objetivo, de que matéria é mente esgotada, hábitos inveterados tornando-se leis físicas.” (CP, 6.24-25).

Em obediência ao princípio, ou máxima de continuidade, segundo o qual devemos imaginar as coisas contínuas na medida em que o possamos, realce-se que devemos supor uma continuidade entre os caracteres da mente e da matéria, tal que a matéria nada seria senão mente que teve seus hábitos cristalizados, fazendo-a agir com um alto e peculiar grau de regularidade mecânica ou rotina. (CP, 6.277)

Dissemos anteriormente que o processo evolutivo é o aumento da terceiridade. Depois de estabelecida a relação entre evolução e continuidade, é possível compreender que o processo evolutivo resulta no aumento da continuidade. “Uma continuidade perfeita seria uma cristalização final da terceira categoria.” (IBRI, 1992, p. 63)

Como vimos, Peirce afirma que a lei da mente tem primazia em relação à lei física. Com seu idealismo objetivo ele afirma que matéria é mente envelhecida e, portanto, o universo é uma grande mente. Nós estamos acostumados a atribuir ações guiadas às pessoas, pois acreditamos que ações desse tipo demandam uma mente, porque elas têm um propósito e são auto-corretivas. Diante desses novos conceitos, já é possível rever essa noção. Podemos perceber porque para Peirce as ações com propósito ou finalidade também ocorrem na natureza.

[...] Particularmente, o sinequista não admitirá que os fenômenos físicos e psíquicos sejam totalmente distintos – tanto como pertencendo a diferentes categorias de substâncias, quanto como sendo lados totalmente separados de um escudo – mas insistirá que todos os fenômenos são de um único caráter, embora alguns sejam mais mentais e espontâneos, outros mais materiais e regulares. Ainda, **todos da mesma forma apresentam aquela mistura de liberdade e restrição, que não apenas os permite ser, mas, os faz ser teleológicos, ou ter uma finalidade.** (CP, 7.570, grifo nosso)

No próximo tópico abordaremos a idéia de uma ação guiada ou teleológica e o conceito de causa final para Peirce.

2.3. Teleologia e Causa Final

Para Peirce, o aspecto teleológico dos fenômenos deriva do fato deles serem parcialmente ocasionados por causas finais. Segundo ele, o conceito de causa final é um dos mais importantes da filosofia e, infelizmente, um dos mais negligenciados. Ele chegou a afirmar que “[...] o não reconhecimento da causa final [...] foi e ainda é produtor de mais erros e absurdos filosóficos que qualquer outra fonte de erros e absurdos”. (MS 478, apud HULSWIT, 2002, p.75)

De acordo com Peirce, os nossos propósitos conscientes são bons exemplos de causas finais, pois temos familiaridade com eles: “um propósito é meramente aquela forma de causa final que é mais familiar à nossa experiência” (CP, 1.211). Portanto, para melhor entendermos a causa final, devemos pensar em situações nas quais agimos guiados por um objetivo. Quando uma pessoa tem como objetivo, por exemplo, construir uma casa, se pode dizer que as ações tomadas para alcançar esse objetivo são guiadas, como quando ela compra material

de construção, ou quando prepara os alicerces da casa. Entretanto, a causa final, como veremos, não se limita aos nossos propósitos conscientes. Abaixo está sua definição, dada por Peirce:

[...] nós devemos entender como causa final aquele modo de ocasionar os fatos de acordo com o qual uma descrição geral do resultado pode ser dada, de forma independente de qualquer compulsão de alcançá-lo dessa ou daquela forma em particular; como se os meios pudessem ser adaptados ao fim. O resultado geral pode ser alcançado em uma ocasião de uma maneira, e em outra ocasião de outra maneira. A causação final não determina de que maneira particular ele será alcançado, mas apenas que o resultado deve ter um caráter geral. (CP, 1.211)

É importante salientar que os objetivos ou propósitos são sempre gerais e nunca singulares. A grande aversão dos cientistas e filósofos modernos em dar explicações teleológicas aos processos naturais, tem base, de acordo com Peirce, no erro de se acreditar que uma causa final é um evento futuro concreto que influencia o presente (HULSWIT, 2002, p. 76). O objetivo de construir uma casa, por exemplo, não contém a descrição da casa nos mínimos detalhes. Nem contém detalhes minuciosos sobre a maneira como a casa será construída. Os propósitos podem ser mais ou menos específicos mas nunca singulares. Dessa forma, um propósito não pode ser um evento futuro. Uma causa final é sempre uma idéia geral:

A primeira coisa que nós percebemos ao considerar nosso próprio comportamento guiado por um objetivo é que, ao contrário do que usualmente se acredita, nossos objetivos não são nem coisas nem eventos. Segundo Peirce, objetivos não são nada além de ‘desejos operativos’, cujos objetos nunca são concretos, mas sempre gerais. Algo desejado é sempre algo de algum *tipo*. Nós queremos certo *tipo* de torta de maçã, não um espécime individual específico. Certamente, existem todos os tipos de níveis de generalidade, e um objetivo pode ser mais específico (menos geral) do que outro: nós podemos querer uma torta de maçã feita de um *tipo* especial de maçãs e de massa. Mas mesmo assim, o objeto permanece geral. Dessa forma, nós podemos ver que causas finais são *gerais*, e não concretas. (HULSWIT, 2002, p.77)

Mas, o propósito de construir uma casa não basta para que a casa seja construída. Ações precisam ser tomadas. Uma causa final escolhe os tipos de meios que serão necessários para se atingir um fim. Ela alcança seus objetivos através da seleção das causas eficientes. Vejamos como Peirce define causação eficiente:

Causação eficiente [...] é uma compulsão determinada por uma condição particular das coisas, e é uma compulsão agindo para fazer uma situação mudar de uma maneira perfeitamente determinada; e o que o caráter geral do resultado pode ser não concerne de forma alguma à causa eficiente (CP, 1.212)

A idéia moderna, mais comumente aceita, de causa e efeito é a que mais se assemelha ao conceito de causa eficiente. De acordo com essa idéia moderna, um efeito necessariamente sucede uma causa. Quando dizemos, por exemplo, que uma janela quebrou ao ser atingida por uma bola, estamos fazendo referência a essa idéia. No caso, o impacto da bola é a causa e o vidro ter quebrado é o efeito. Argumenta-se que essa é a única das quatro causas aristotélicas que sobreviveram. Entretanto, Peirce chama a atenção para o fato de que essa idéia moderna é diferente da concepção original de Aristóteles:

Considera-se geralmente que a palavra causa foi simplesmente reduzida para uma das quatro causas Aristotélicas a qual foi nomeada devido à circunstância de que ela sozinha produz um efeito. Mas esta noção de que nosso conceito de causa é aquele da causa eficiente Aristotélica não sobrevive ao exame. A causa eficiente foi em primeiro lugar geralmente uma coisa e não um evento, e algo que não precisa fazer nada; sua mera existência podia ser suficiente. Nem o efeito necessariamente sempre sucedia. É verdade que quando ele sucedia ele era dito ter sido compelido. Mas ele não era necessário em nosso sentido moderno. Quer dizer, ele não era invariável. (RLT, p. 198, apud HULSWIT, 2002, p. 183)

A crença atual em um universo mecânico, estritamente regido pela lei, é a crença de que não apenas existem somente causas eficientes no sentido Aristotélico, mas também na idéia moderna de que todo efeito é necessariamente antecedido por uma causa. Como vimos, Peirce rejeita a idéia de um universo mecânico, e assevera a importância do acaso real. Ele acredita, diferentemente de muitos filósofos contemporâneos, que a existência da causa final não exclui a causa eficiente, muito pelo contrário. A causa eficiente é um componente da causação, mas não o único. A relação entre a causa final e a causa eficiente pode ser apreciada nas seguintes passagens:

A corte não pode ser imaginada sem um xerife. Causalidade final não pode ser imaginada sem causalidade eficiente [...]. O xerife ainda teria seu pulso, mesmo que não existisse corte: mas causa eficiente, separada de uma causa final na forma de uma lei, nem ao menos possuiria eficiência (CP, 1.213)

[...] Causação final sem causação eficiente é desamparada [...]. Causação eficiente sem causação final, entretanto, é pior que desamparada, de longe; ela é mero caos; e mesmo o caos não chega a ser caos, sem a causa final; ele é um nada vazio. (CP, 1.220)

Percebemos que a causa final e a causa eficiente não são tipos de causas diferentes, mas ambas são componentes presentes em todo evento:

[...] causação final e causação eficiente não são dois tipos diferentes de causação, sendo que cada uma agiria em uma situação diferente. Em cada ato de causação, existe um componente de causa eficiente e um componente de causa final. (HULSWIT, 2002, p. 82)

Peirce redefiniu, portanto, o próprio conceito de causa e efeito, sendo importante ressaltar que, de acordo com essa nova definição, o acaso também é outro componente sempre presente em todo evento, e ele é fundamental para seu aspecto teleológico:

Desde que cada evento envolve um elemento de acaso objetivo, e desde que a ação do acaso tem “resultados teleológicos inevitáveis” (que são determinados por certas leis estatísticas), cada evento deve ser parte de uma cadeia causal que se desenvolve em uma direção definida. Assim, acaso objetivo deve envolver teleologia.

Mas, de outra forma, não pode haver teleologia sem acaso. Sem acaso, todos os processos seriam puramente mecânicos (determinísticos). A variedade no mundo assim como a possibilidade de existir diferentes rotas levando ao mesmo estado final, podem, na visão de Peirce, somente ser explicadas pelo acaso. (HULSWIT, 2002, p. 85)

Conseqüentemente, todo evento tem um componente de causa final, um componente de causa eficiente e um componente de acaso. Para compreender melhor essa relação, vamos imaginar que uma pessoa pretende atravessar uma rua, saindo de um ponto A e chegando a um ponto B. A causa final – estar no ponto B – pode ser realizada através de uma cadeia de eventos. Cada evento intermediário é causado em parte pelo evento anterior, e esse é o componente de causa eficiente, que limita as opções. Mas, cada evento intermediário não é absolutamente determinado pelo evento anterior. Existe um elemento de liberdade que é caracterizado pelo acaso. Assim, existindo liberdade, a causa final, que é o terceiro componente, seleciona uma das diferentes opções disponíveis para cada evento, que irá melhor contribuir para que ela seja realizada.

Dessa forma, se imaginarmos que a distância entre A e B é de 100 metros, estando no meio do caminho entre A e B, não seria possível para um ser humano dar um passo de 50 metros e voltar ao início. As possibilidades são limitadas, entretanto não são absolutamente limitadas. É possível dar um passo em qualquer direção, inclusive para trás, mas essa ação não contribuiria para alcançar o objetivo, que é chegar em B, e não seria selecionada pela causa final. A melhor opção, obviamente seria dar um passo à frente, na direção de B.

A participação do acaso pode ser especialmente percebida ao se imaginar que no meio do caminho entre A e B exista uma pedra muito alta, fazendo com que seja necessário contorná-la. Isso pode ser feito tanto pela esquerda quanto pela direita. Se ambos os eventos – contornar a pedra pela esquerda, ou pela direita – contribuem da mesma forma para alcançar o objetivo, não há como dizer qual decisão será tomada: isso cabe ao acaso. Graças ao acaso, é possível alcançar o mesmo fim seguindo vários caminhos diferentes. Alguém que quer construir uma casa, por exemplo, pode fazê-lo em mais ou menos tempo, comprando um tipo

ou outro de material, pintando-a de uma cor ou de outra. O resultado final pode ser alcançado por várias vias diferentes. A casa, como causa final, é sempre uma idéia geral.

Por uma tendência para um fim, eu quero dizer que certo resultado será obtido, ou aproximado, e de certa maneira que, dentro dos limites, se a sua obtenção por uma linha de causação mecânica for impedida, ele será obtido, ou aproximado, por uma linha de causação mecânica independente. (NEM 4, p. 66, apud HULSWIT, 2002, p. 78)

A seguinte passagem de T. L. Short, ilustra de maneira bastante clara o que Peirce quis dizer na passagem acima:

[...] se você colocar um obstáculo atravessando o trajeto de uma linha de formigas que andam apressadamente pra frente e pra trás entre seu formigueiro e alguma fonte de comida, você irá descobrir que, após alguma exploração aleatória ao longo das bordas do obstáculo, as formigas tomarão um novo caminho (se existir um) entre o lar e a comida. O que acontece é que elas não são simplesmente levadas para a comida pelos caminhos que elas tomam, mas elas tomam aqueles caminhos porque dessa forma elas serão levadas para a comida. Para explicar porque os meios são variados e continuam a dar resultados do mesmo tipo geral, nós devemos supor que aquele tipo é a causa final do processo todo. (SHORT, 1981b, p. 370)

Até o momento estamos considerando que a causa final permanece inalterada durante o processo de sua realização. Foi dito que a participação do acaso se limita aos diferentes meios que podem ser selecionados para se alcançar um fim. Entretanto, Peirce acreditava que a causa final tem um aspecto criativo, e não é a mera busca de um fim pré-determinado.

[...] essa teleologia é mais que uma mera busca de um fim pré-determinado; ela é uma teleologia desenvolvente. [...]. Uma idéia geral, viva e consciente agora, já é determinante de atos no futuro até um ponto no qual ela não é agora consciente. Essa referência ao futuro é um elemento essencial da personalidade. Estivessem os fins de uma pessoa já explícitos, não haveria espaço para desenvolvimento, para crescimento, para vida: e consequentemente não haveria personalidade. A mera realização de propósitos pré-determinados é mecânica. (EP 1, p. 221, apud HULSWIT, 2002, p. 87)

O objetivo de construir uma casa, por exemplo, pode criar vários outros sub-objetivos, como contratar construtores, comprar material de construção etc, sendo que cada um desses pode, por sua vez, criar novos sub-objetivos. Nesses casos, esses novos objetivos, que são mais específicos, estão subordinados ou a serviço dos objetivos mais gerais. Ainda, o objetivo de construir uma casa pode se transformar em um outro objetivo diferente, como por exemplo o de fazer uma faculdade de engenharia civil. Menno Hulswit explica isso de forma bastante esclarecedora:

[...] no processo de serem realizadas, as próprias causas finais podem mudar. Além disso, causas finais diferem em grau de generalidade, e as mais gerais podem, enquanto sendo realizadas,

gerar causas finais menos gerais, ou subordinadas. Nas palavras de T.L. Short isso significa: “o acaso então leva a novos fins, mas apenas quando o novo é uma forma de realizar um fim mais geral e que já esteja operante”. As novas causas finais subordinadas desenvolvidas devem se encaixar dentro do esquema geral das causas finais mais gerais. Por exemplo, se a causa final mais geral é a idéia de escrever um artigo sobre algum aspecto da cosmologia de Peirce, então a mudança de idéia de escrever sobre o conceito de causação eficiente de Peirce para a idéia de escrever sobre seu conceito de teleologia se encaixa dentro do esquema da causa final mais geral. Além disso, os novos propósitos subordinados, que surgem por acaso, são selecionados pela causa final mais geral. Isso tem grandes consequências, como T.L. Short novamente expressa bem: “Causação final dessa forma resulta, não na uniformidade morta de um plano simples, mas heterogeneidade imprevisível de empreendimentos, personalidades, e espécies que preenchem o mundo”. (HULSWIT, 2002, pp. 87-88)

É importante salientar que, embora Peirce tenha usado o termo ‘teleologia desenvolvente’ para falar da personalidade humana, é possível aplicar a idéia para a teleologia em geral:

[...] crescimento se refere ao processo no qual os próprios propósitos podem evoluir. Mas, ‘teleologia desenvolvente’ não está restrita à personalidade humana. Ela é aplicável à idéia de teleologia em geral: aprendendo com o aspecto desenvolvente de nossos próprios propósitos humanos, nós podemos indutivamente inferir que todas as causas finais na natureza estão, pelo menos em princípio, sujeitas à evolução. (HULSWIT, 2002, p. 87)

Mesmo que já tenhamos dito, vale reforçar que, embora Peirce tenha utilizado os propósitos conscientes das pessoas como o ponto de partida para sua análise da causa final, para ele a causa final é operativa na natureza, sendo que todo evento no universo tem um elemento de finalidade. A partir disso, podemos começar a compreender por que, segundo ele, a evolução é guiada. Entretanto, algo ainda não está totalmente claro. A pergunta que não quer calar é: como a causa final é possível? Explicaremos.

No exemplo de alguém que atravessa a rua, dissemos que a pessoa sai do ponto A e chega ao ponto B, e, portanto, estar no ponto B é a causa final. Afirmamos que a causa final seleciona as causas eficientes que melhor contribuem para a consecução do objetivo. Assim, a cada momento, a pessoa dá um passo na direção de B e não na direção oposta. A questão que queremos levantar é: como isso é possível?

O que queremos dizer é que, se a causa final seleciona os eventos, como um evento pode ser selecionado antes de existir, ou seja, como algo no mundo pode selecionar alguma coisa que não existe em detrimento de outras que nem virão a existir? Como saber se determinado evento irá contribuir ou não para o objetivo antes que toda a cadeia de eventos culmine na consecução do objetivo? Somente após toda sequência de eventos ter acontecido é possível avaliar, em retrospecto, se um determinado evento, ou uma determinada escolha,

contribuiu para atingir aquele objetivo. Caso contrário estaria sendo realizada uma premonição.

A resposta para essa questão reside na possibilidade de simular ações que é fornecida pela representação simbólica. A semiose, portanto, tem um papel fundamental na causa final, e o nosso próximo passo será estudar a semiótica, para compreender melhor esse papel.

2.4. Semiótica e as Formas de Representação

A lógica, segundo Peirce, é a ciência da terceiridade, que é, por sua vez, a categoria do crescimento, da evolução, da aquisição de hábitos. Como já dissemos, terceiridade não é senão outro nome para representação. Dessa maneira, Peirce acabou chegando à conclusão que a lógica deveria ser a filosofia da representação.

Agora é possível que a *lógica* deva ser a ciência da Terceiridade em geral. Mas como eu a tenho estudado, ela é simplesmente a ciência daquilo que deve ser uma representação verdadeira [...] Ela é, resumidamente, a filosofia da representação (CP, 1.539)

Sendo a semiótica a ciência que estuda todas as formas de representação, ela equivale à lógica, na filosofia Peirceana. Assim, levando em consideração que a terceira categoria é a categoria do crescimento, a semiótica ou lógica, para Peirce, tem caráter de processo, que a faz ser diferente da lógica tradicional, fazendo-a ser mais parecida com a lógica de Hegel. Dessa forma, a representação é uma ação e não uma relação estática. Essa ação é genuinamente triádica, e nela estão sempre presentes o Objeto, o Signo ou Representamen, e o Interpretante. Abaixo estão duas das muitas definições dadas por Peirce para essa ação triádica.

Um **Signo, ou Representamen**, é um **Primeiro** que está em uma tal relação genuína com um **Segundo**, chamado seu **Objeto**, de modo que seja capaz de determinar um **Terceiro**, chamado seu **interpretante**, para assumir a mesma relação triádica com seu Objeto na qual ele próprio está com o mesmo Objeto. (CP, 2.274, grifo nosso).

Um **Signo** é qualquer coisa que está relacionado com uma Segunda coisa, seu **Objeto**, em respeito a uma Qualidade, de tal maneira a trazer uma Terceira coisa, seu **interpretante**, para uma relação com o mesmo Objeto, e isso de tal maneira a trazer uma Quarta coisa para uma relação como o mesmo Objeto da mesma forma, ad infinitum (CP, 2.92, grifo nosso).

A segunda definição evidencia que a representação é um processo, e que esse processo se estende infinitamente no tempo: um signo representa um objeto para uma mente, gerando

nessa mente um interpretante, que por sua vez também é um signo, e vai gerar outro interpretante, e assim infinitamente. A esse processo Peirce deu o nome de semiose.

Para compreender melhor o signo, vamos estudá-lo dentro da mais importante das inúmeras classificações desenvolvidas por Peirce. Nela o signo é classificado de acordo com a forma como ele se relaciona com seu objeto. Segundo ela, o signo pode ser um ícone, um índice ou um símbolo.

Um ícone é um signo que representa seu objeto por apresentar semelhanças com ele. Uma foto de uma pessoa, por exemplo, representa iconicamente aquela pessoa. Já um índice é um signo que representa seu objeto por estar fisicamente ligado a ele ou por co-ocorrer no espaço ou no tempo. O cheiro de fumaça, por exemplo, desperta na mente de uma pessoa a idéia de fogo. Podemos dizer que a fumaça é um índice do fogo. E, finalmente, um símbolo é um signo que representa seu objeto por uma lei ou convenção. A palavra “árvore”, por exemplo, é um símbolo. Ela não tem nenhuma relação física, direta, com o objeto, e o representa por uma convenção. No caso do símbolo, tanto o objeto quanto o interpretante são idéias gerais. A palavra “árvore” não representa nenhuma árvore específica mas a idéia geral de árvore. É importante salientar que cada ocorrência da palavra “árvore” é uma réplica ou corporificação, e que “a palavra, em si mesma, não tem existência embora tenha um ser real que consiste no fato que os existentes se *deverão* conformar a ela” (PEIRCE, 2003, p. 71).

A classificação dos signos entre ícones, índices e símbolos, pode ser vista de forma hierárquica, como demonstrou Terrence Deacon (1998). De acordo com ele, essa hierarquia se baseia no fato de que os índices são formados de ícones e os símbolos são formados de índices. Além disso, essa hierarquia também pode ser observada no fato de que o mesmo signo pode ter diferentes níveis de interpretação. Ele cita o exemplo de um arqueólogo que encontra marcas em uma tábua de barro. Inicialmente esse arqueólogo interpretará essas marcas como sendo ícones umas das outras, ou seja, apenas como sendo semelhantes entre si, por não ter a capacidade para gerar novos interpretantes. Se alguma relação for detectada entre essas marcas e outros artefatos, como por exemplo, diferentes marcas indicam o número de artefatos transportados, então a interpretação das mesmas marcas passa a ser indicial. Nas palavras de Deacon:

Não é apenas o caso que nós somos capazes de interpretar o mesmo signo de maneiras diferentes, mas mais importante, essas diferentes interpretações podem ser arranjadas em um tipo de ordem crescente que reflete uma competência anterior em identificar relações associativas de nível mais alto. Em outras palavras, a própria referência é hierárquica em sua estrutura. (DEACON, 1998, p. 73).

O ícone, como já dissemos, representa seu objeto por semelhança. Dentro da hierarquia apresentada por Deacon, o ícone é a forma mais simples de representação, não dependendo de nenhuma outra. A representação icônica pára no nível do reconhecimento. Reconhecer algo significa que uma sensação atual é icônica de outra que temos na lembrança, como quando reconhecemos que um cheiro que estamos sentindo agora é de fumaça. Reconhecer algo como sendo um ícone pode ser visto como falhar em interpretá-lo como índice de outra coisa.

Um índice, por sua vez, é um signo que representa seu objeto por contigüidade ou co-ocorrência espaço-temporal. Diferentemente do ícone, que não depende de nenhuma outra forma de representação, a representação indicial depende da representação icônica. Essa dependência pode ser apreciada no exemplo da fumaça como índice de fogo:

O cheiro de fumaça traz à mente experiências passadas similares (representando-as iconicamente). Cada uma dessas experiências vem à mente devido às suas similaridades entre si e com o evento presente. Mas o que é acrescido é que, muitas dessas experiências passadas também compartilham outras similaridades. Em muitas dessas ocasiões eu também percebi algo queimando que era a fonte da fumaça, e dessa forma essas experiências foram ícones umas das outras.

Existe uma característica importante acrescida a estes reconhecimentos icônicos. A *correlação repetida* entre o cheiro da fumaça e a presença de chamas em cada caso adiciona um terceiro nível de iconicidade de ordem maior. (DEACON, 1998, p. 78).

Assim, para que uma coisa indique outra, é necessário ter, primeiramente, a capacidade de reconhecer as coisas, ou seja, a capacidade de representá-las iconicamente. A partir de várias lembranças similares, é possível “separar” duas características distintas, que co-existem no espaço ou no tempo, dentro dessas lembranças. Após um processo desses, o cheiro de fumaça, por exemplo, não traz à mente apenas lembranças similares, mas traz a idéia de fogo.

Entenderemos agora como se forma a representação mais complexa e a mais importante para a nossa discussão: a representação simbólica. Deacon fornece uma explicação particularmente esclarecedora de como a representação simbólica se forma a partir de representações indiciais, e, portanto, iremos reapresentá-la aqui, de forma resumida. Essa explicação é importante pois, como veremos adiante, ela ajudará a compreender o papel da semiótica na evolução, e a relação das linguagens e mensagens com esta. A explicação é feita através da descrição de uma série de experimentos realizados com chimpanzés por Sue Savage-Rumbaugh e Duane Rumbaugh (SAVAGE-RUMBAUGH 1978; SAVAGE-RUMBAUGH 1980), que evidenciam o processo de aquisição de representações simbólicas por esses chimpanzés.

2.4.1. A Formação do Símbolo

Nos experimentos, dois chimpanzés – Sherman e Austin – foram ensinados a utilizar um teclado de computador especial feito de lexigramas – formas abstratas simples – com grandes teclas iluminadas. Os chimpanzés foram treinados a pressionar seqüências de lexigramas que representassem uma relação simples de verbo + substantivo, sendo que os substantivos representavam alimentos e os verbos as ações “dê-me líquido” e “dê-me sólido”. Eles tinham que pressionar a seqüência certa, de acordo com o alimento disponível para ter acesso a esse alimento. Inicialmente eles foram treinados com quatro alimentos mais os dois verbos, portanto, seis teclas. Para facilitar a compreensão, vamos imaginar que o teclado utilizado pelos chimpanzés, ao invés de lexigramas tivesse letras e fosse como a figura abaixo:

S	M	B
L	R	A

As duas letras da esquerda estariam associadas aos verbos ou ações:

- S = Dê-me sólido
- L = Dê-me líquido

As quatro letras da direita estariam associadas aos alimentos. Deacon não especifica quais eram os alimentos na experiência e nós vamos atribuir alimentos arbitrários às letras:

- M = Maça
- B = Banana
- R = Refrigerante
- A = Água

Portanto, as mensagens corretas seriam:

- LR = Dê-me líquido + Refrigerante
- LA = Dê-me líquido + Água
- SM = Dê-me sólido + Maça
- SB = Dê-me sólido + Banana

Inicialmente, cada par teve de ser ensinado separadamente e, após essa tarefa, os chimpanzés foram expostos a todas as opções de uma vez. Foi, então, pedido a eles que escolhessem a seqüência mais apropriada baseados na disponibilidade da comida. Os chimpanzés tiveram problemas com essa tarefa, cujo conhecimento aparentemente não estava implícito no treinamento.

Isso era evidente no fato de que alguns chimpanzés tendiam a repetir estereotipicamente apenas a mais recente combinação simples aprendida, enquanto outros encadeavam juntas todas as opções, independentemente do significado pretendido e do que eles sabiam a respeito da situação. Eles tinham aprendido as associações individuais mas falharam em aprender o sistema de relações do qual essas correlações faziam parte.(DEACON, 1998, p. 85).

Essa inabilidade dos chimpanzés é devida ao fato deles terem criado representações indiciais da comida e não simbólicas. Para eles, a seqüência “LR”, por exemplo, era simplesmente um signo que **indicava** refrigerante. Eles não aprenderam que “L” significa líquido e “S” significa sólido. Os botões “L” e “R” não eram, ainda, conceitos abstratos.

Então os treinadores passaram a treiná-los explicitamente nas combinações que não tinham significado, punindo ou não recompensando os erros. No final do processo os chimpanzés foram capazes de reconhecer as seqüências certas todas as vezes. Para comprovar que os chimpanzés tinham realmente aprendido a regra sólido/líquido, novos alimentos foram adicionados e Sherman e Austin foram capazes de produzir a seqüência correta na primeira vez ou após poucos erros, ao invés de ter que passar por centenas de tentativas, como anteriormente.

Eles descobriram que a relação que um lexígrafo tem com um objeto é uma *função* da relação que ele tem com outros lexígramas, não apenas uma função do aparecimento correlacionado de ambos lexígrafo e objeto. Essa é a essência do relacionamento simbólico.(DEACON, 1998, p. 86).

Agora, os chimpanzés não tinham simplesmente aprendido a relação entre uma seqüência de letras e um alimento, como “LR” significando “dê-me refrigerante”, e “LA” significando “dê-me água”. Eles também conseguiram aprender que entre essas duas mensagens, a letra L era um elemento comum, e que significava que o alimento era líquido.

Eles aprenderam uma regra abstrata que surgiu a partir do momento em que foi estabelecida uma relação entre as representações indiciais que eles já possuíam. A experiência demonstrou que as representações indiciais são necessárias para construção de representação

simbólica. A representação simbólica emerge através de um sistema de representações e os símbolos não podem ser aprendidos um de cada vez como o que acontece com os índices.

2.5. Evolução, Linguagem e Mensagem

A experiência realizada com os chimpanzés é interessante pois além de evidenciar a relação entre ícones, índices e símbolos, ilustra o processo de aquisição simbólica. Para entender a relação desse processo com a evolução, devemos entender melhor a natureza do símbolo.

Todo símbolo é um legissigno. Além de classificar os signos de acordo com sua relação com o objeto, ou seja, como ícones, índices e símbolos, Peirce também classificou o signo de acordo com sua própria natureza. Segundo essa classificação, um signo pode ser um qualissigno, um sinsigno ou um legissigno. Um qualissigno é uma qualidade que é um signo. Um sinsigno é uma ocorrência individual, um singular que é signo. Um legissigno é uma lei que é signo. Essa lei, é uma lei de gerar interpretantes.

[...] o que o faz agir como tal é sua tendencialidade, isto é, tende a gerar o mesmo interpretante ou interpretantes semanticamente correlatos. Isto porque a natureza da lei é exatamente aquela de funcionar como mediação pela qual as ocorrências particulares se conformarão à generalidade imposta pela lei. (SANTAELLA, 2000, p. 105)

Todo legissigno significa e é atualizado através de ocorrências, de casos particulares que são suas réplicas. A réplica de um legissigno é um sinsigno especial.

[...] os legi-signos precisam de réplicas para se atualizarem.[...] A réplica de um símbolo é um tipo especial de índice que age para aplicar a regra geral ou hábito de ação ou expectativa associada com o símbolo a algo particular. (SANTAELLA, 2000, p. 135).

Essas instâncias de manifestação são denominadas “réplicas”. Trata-se de sin-signos de tipo especial, isto é, são sin-signos porque ocorrem aqui e agora, mas são réplicas porque corporificam e atualizam um legi-signo. (SANTAELLA, 2000, p.102).

Como todo símbolo é um legissigno, ele também significa e é atualizado através de suas réplicas: “[e]le [o símbolo] é portanto ele próprio um tipo geral ou uma lei, isto é, um Legissigno. Como tal ele age através de uma Réplica.” (CP, 2.249).

O processo de formação de símbolos, portanto, é um processo de formação de uma lei de interpretação a partir de réplicas, ou seja, a partir de ocorrências individuais. O passo necessário para compreender melhor o papel da semiótica na evolução é lembrar que a semiótica, para Peirce, não apenas descreve as leis do raciocínio, mas descreve um processo que ocorre no mundo. Esse fato se torna evidente se considerarmos as decorrências do seu

idealismo objetivo, que afirma que não existe separação entre mente e matéria, e que a lei da mente tem primazia em relação à lei da matéria. Ainda, levando em consideração a classificação das ciências na filosofia de Peirce, vemos que a metafísica é fundada na semiótica, e, portanto, devemos esperar que a ação do signo ocorra na realidade.

Sendo a semiótica a ciência da terceiridade, a semiose corresponde ao processo como a evolução se dá na realidade. Podemos concluir que a evolução equivale ao crescimento de representações gerais, ou seja, ao crescimento dos símbolos. Os outros tipos de semiose – ação do ícone e do índice – são etapas na formação de representações simbólicas.

Uma lei é em si nada além de uma formula geral ou símbolo. (CP, 5.107). [e] Terceiridade, como eu uso o termo, é apenas um sinônimo para Representação. [...] Agora é apropriado dizer que **um princípio geral que é operativo no mundo real é da natureza essencial da Representação e do Símbolo** devido ao seu modus operandi ser o mesmo daquele através do qual as palavras produzem efeitos físicos (CP, 5.105, grifo nosso).

De forma simplificada podemos dizer que a evolução resulta no aumento de representações gerais – os símbolos – a partir de réplicas. Mas, da mesma maneira que um símbolo é atualizado pelas réplicas, um sistema de símbolos é atualizado por um sistema de réplicas. Iremos chamar um sistema de símbolos de linguagem, e um sistema de réplicas de mensagem.

Levando isso em consideração, podemos traçar um esboço do funcionamento da evolução, em termos de linguagens e mensagens. Tudo que é geral ou regular no universo é da natureza do símbolo, e resulta da terceiridade. Os símbolos geralmente são organizados em sistemas – as linguagens. Estas definem regras que limitam os tipos de mensagens que podem ser produzidas, mas, mesmo assim, sempre é possível produzir um número infinito delas. As mensagens, são ocorrências individuais, acontecem aqui e agora, são sinsignos. Limitadas pelas linguagens (terceiridade) mais ainda usufruindo de um elemento de liberdade (primeiridade) novas mensagens são produzidas (segundidade). Após esse passo, através da tendência à aquisição de hábitos, as mensagens atualizam as linguagens. Isso equivale a dizer que as réplicas atualizam os símbolos. Esse é um processo sintético, de inferência indutiva, através do qual o conteúdo das mensagens de maior “incidência” é incorporado na linguagem. Estas se modificam, se atualizam, se organizam e adquirem a capacidade de gerar novos tipos de mensagens. E o processo se repete, infinitamente.

A evolução biológica pode ser compreendida sob esse ponto de vista. O pool de genes (o conjunto de todos os genes) pode ser visto como uma linguagem. Cada gene, ou um determinado conjunto deles, pode ser visto como uma mensagem (escrita em DNA). O pool

de genes (linguagem) limita os próximos genes (mensagens) que podem ser gerados. Não é possível nascer um homem voador em apenas uma geração, por exemplo. De tempos em tempos novas mensagens são geradas: os organismos se reproduzem, possivelmente combinando seu DNA ou sofrendo mutações. A seleção natural é o processo sintético que modifica o pool de genes a partir das mensagens de maior ocorrência (genes mais bem sucedidos).

Após ter compreendido melhor a semiótica, e sua relação com a evolução, temos subsídios para compreender a relação da semiose com a causa final e, consequentemente, o aspecto teleológico da evolução.

2.6. Evolução Teleológica

Como vimos, os índices são formados de ícones, e, através deles, estão relativamente “conectados” às coisas do mundo. Entretanto, uma representação simbólica não está diretamente “conectada” a nenhuma ocorrência individual. Ela é sempre abstrata, mantendo um distanciamento, uma alienação em relação à realidade se comparada com ícones e índices. A idéia de líquido, por exemplo, não está diretamente ligada a nenhuma ocorrência específica de um líquido, e permanece sempre geral. Alguns símbolos, ainda, podem ser mais abstratos do que outros, ou seja, essa distância pode aumentar na medida em que “[...] símbolos não representam apenas coisas no mundo, eles também representam uns aos outros“. (DEACON, 1998, p. 99).

Percebemos essa relação entre os símbolos quando procuramos o significado de uma palavra no dicionário. O significado será dado através de uma combinação de outras palavras, cujo significado, por sua vez, também está associado a uma combinação de outras palavras. De acordo com Deacon, dessa inter-relação entre as palavras decorre o fato de não esquecermos o significado de uma palavra, mesmo quando não a usamos muito frequentemente. Isso porque ela está relacionada com muitas outras e todas se reforçam mutuamente, fato que não acontece com a representação indicial, pois cada representação é independente da outra. Um fato importante que decorre dessa característica da representação simbólica – sua alienação em relação à realidade – é que isso é justamente o que permite que a ação seja guiada. Explicaremos.

Retomando a discussão sobre causa final, lembremos que havíamos utilizado como ilustração o caso de uma pessoa que sai do ponto A com objetivo de chegar ao ponto B, que é a causa final. A pergunta que havia ficado sem resposta é a seguinte: como é possível a causa

final selecionar os eventos que irão contribuir para alcançá-la? Em outras palavras, como é possível saber que dar um passo na direção de B é um evento que irá contribuir para o objetivo de chegar em B, antes que toda a cadeia de eventos culmine na consecução desse objetivo?

É possível afirmar que descartando a premonição e sem contar com a experiência do passado, o que restam são o acaso e a compulsão cega. Teriam que ser efetuadas cegamente uma série de tentativas até que uma delas realize o propósito, ou seja classificada dentro daquela idéia geral que é a causa final, para se saber, em retrospectiva, se determinado evento contribuiu para sua realização ou não. No caso de ir de A até B, a única forma de realizar essa tarefa seria dando passos em direções aleatórias para que eventualmente se chegue ao ponto B. Somente após lá chegar, essa experiência adquirida poderia ser utilizada para balizar tarefas semelhantes no futuro. Estamos afirmando, portanto, que **somente contando com a experiência é possível ter uma ação guiada**.

Entretanto, existe outra maneira de dizer que agimos baseados na experiência. Sabemos que nem todas as ações guiadas das pessoas são balizadas no aprendizado obtido **por ter agido no mundo** no passado. Dizemos que as pessoas **pensam antes de agir**, e ao dizer isso também pretendemos dizer que suas ações não são cegas. As pessoas formulam hipóteses de possíveis ações em suas mentes e testam essas hipóteses, e depois agem baseadas nos resultados desses testes. Agir baseado no resultado dos testes feitos nas hipóteses, também é agir baseado no passado. Entretanto, testar hipóteses não significa agir no mundo, mas sim agir em um modelo hipotético, simplificado, do mundo. Formular e testar hipóteses equivale a realizar simulações. Assim, podemos dizer que uma ação baseada em uma simulação é uma ação guiada.

Portanto, sabemos que é possível uma pessoa sair do ponto A e chegar ao ponto B, de forma guiada, sem realizar premonição e sem contar com a experiência de ter realizado a mesma tarefa no passado. A pessoa faz isso simulando a ação em sua mente e agindo baseada no resultado da simulação. Mas, **esse modelo hipotético do mundo, criado na mente, só pode ser construído com representações simbólicas**, porque, como dissemos, elas são apartadas da realidade, e permitem representar coisas abstratas, que ainda não existem.

[...] Símbolos fornecem os meios para pensar sobre pensamentos de maneiras as quais nós não poderíamos de outra forma pensar sobre eles. **Eles nos permitem, por exemplo, criar Abstrações, sem as quais nós não teríamos o grande motor da descoberta.** (CP, 4.531)

Concluímos que capacidade abstrativa do símbolo é o que possibilita a causa final, e isso não poderia ser diferente, pois, como dissemos, uma causa final é uma idéia geral. Para ter outro exemplo de ação guiada como sendo ação baseada em simulação, podemos pensar na situação onde alguém cria um modelo de uma casa em sua mente, ou até em um computador, antes de construí-la de verdade. Repare que o modelo muda muito mais rapidamente que a casa real, e derrubar uma parece no modelo é muito mais fácil e rápido. Assim, a construção da casa pode ser guiada pelos resultados obtidos nas simulações feitas no modelo.

Estamos afirmando que uma ação é guiada quando é baseada numa simulação. Mas, e quando essa ação guiada é o próprio processo evolutivo? Isso é fundamental para nossa discussão. Somos levados a concluir que a evolução é guiada quando ela, também, é baseada na simulação. Mas, é possível perceber que a simulação não pode ser um processo muito diferente da própria evolução. A simulação é um processo de aprendizado, de formação de representações gerais a partir de ocorrências individuais. É um processo de síntese característico da terceiridade. Dessa maneira, a única diferença entre evolução guiada e simulação é que a simulação é um processo evolutivo que está “a serviço” de outro.

Chegamos à questão mais importante do trabalho. A evolução é guiada porque existem diferentes níveis evolutivos e a evolução no nível mais flexível guia a evolução do nível mais rígido. O processo evolutivo deixa de ser cego quando ele é baseado em outro.

Poderíamos estabelecer que a simulação em si – a realização de testes no modelo hipotético – é um processo cego, e a ação baseada na simulação é guiada. Entretanto, a simulação não pode ser *totalmente* cega. Ela deve ser baseada em outro processo evolutivo também. Lembremos que a simulação e a evolução são instâncias de semiose, e esse processo é um processo contínuo. Se o leitor se recorda da definição de um continuum, lembrará que ele pode ser sempre infinitamente dividido. Isso quer dizer que o processo de semiose pode ser dividido em partes, e cada parte, por sua vez, também pode ser dividida em partes, isso infinitamente. Dessa forma, concluímos que a evolução sempre tem um aspecto teleológico, mesmo que seja imperceptível. Essa noção está presente nas passagens abaixo.

[...] evolução não é nada mais nem menos do que do que a busca para alcançar um fim definido (CP, 1.204). [e] [...] a evolução física trabalha guiada por fins da mesma maneira que a ação mental trabalha guiada por fins, e portanto [...] a causação final é sozinha primária (CP, 6.101).

Agora, analisemos, cuidadosamente, a relação que Deacon estabelece entre os processos representacionais com a causa final, asseverando a importância da representação simbólica.

[...] Através da internalização da tentativa e erro físicos, e ainda internalizando modelos abstratos de processos físicos que podem ser extrapolados até os seus extremos possíveis e impossíveis, nós somos capazes daquilo que a evolução genética não é: premeditação. **Processos representacionais são o substrato para “causalidade final,” aquele uso no dia-a-dia de fins imaginados para guiar a seleção de meios presentes.** Processos simbólicos nos libertaram dos limites do presente e do possível imediatos. (Deacon, 1998, p. 458-459, grifo nosso)

Essa passagem é importante para nós tanto positivamente quanto negativamente. Nós concordamos que a capacidade de internalizar os processos de tentativa e erro é o que permite uma premeditação. Isso é o equivalente a dizer que a simulação possibilitada pela representação simbólica é o que permite a ação guiada. Entretanto, ao contrário de Deacon, nós argumentamos que a evolução genética também é capaz dessa premeditação, justamente pelo mesmo motivo. Ao dizer que a evolução biológica pode ter sido guiada pela evolução cultural, através do efeito Baldwin, se quer dizer justamente isso. Poderíamos dizer que, nesse caso, a evolução cultural seria a simulação “a serviço” da evolução biológica, uma “sub-semiose” do processo de semiose biológica.

Como acabamos de ver, para Peirce a evolução é sempre guiada, em maior ou menor grau. Das três teorias evolutivas descritas por ele, a mais completa de todas, e que descreve melhor a realidade, segundo ele, é aquela que apresenta um elemento de teleologia. As três formas de evolução são enumeradas na seguinte passagem.

Três modos de evolução foram assim trazidos diante de nós: evolução por variação fortuita, evolução por necessidade mecânica, e evolução por amor criativo. Nós podemos designá-las evolução ticástica, ou ticasmo, evolução anancástica, ou anancasmo, e evolução agapástica, ou agapismo. (CP, 6.302).

A evolução ticástica se dá através da variação fortuita, ou seja, de afastamentos acidentais do hábito. Oposta à evolução ticástica está a anancástica, que sustenta que a evolução ocorre apenas por necessidade mecânica.

O desenvolvimento de pensamento ticástico, portanto, consistirá em pequenos afastamentos das idéias habituais em direções diferentes indistintamente, sem nenhum propósito e sem nenhuma restrição quer por circunstâncias externas ou pela força da lógica, sendo que esses novos afastamentos serão seguidos por resultados imprevistos que tendem a fixar alguns deles como hábitos mais do que outros. (CP, 6.307)

Diametralmente opostas à evolução pelo acaso estão aquelas teorias que atribuem todo o progresso a um princípio interno necessário, ou outra forma de necessidade (CP, 6.298)

Nós sabemos agora, que, na filosofia de Peirce, o acaso ou a necessidade não poderiam ser considerados, cada um isoladamente, como sendo responsáveis pela evolução. Peirce identifica nas idéias de Lamarck uma teoria mais completa. Ele argumenta que a característica presente na teoria de Lamarck, que a distingue das outras é que nela a evolução ocorre pelo esforço. Ao usar o termo esforço, ele não se refere, obviamente, à compulsão ou necessidade lógica. O esforço é caracterizado por ser um esforço para atingir um objetivo, ou seja, guiado por uma causa final. O elemento distintivo da teoria de Lamarck, portanto, é o elemento de teleologia.

Evolução por acaso e evolução por necessidade mecânica são concepções lutando uma com a outra. Um terceiro método, que suplanta sua disputa, reside envolto na teoria de Lamarck (CP, 6.299)

Assim, a teoria Lamarckista apenas explica o desenvolvimento de características para as quais os indivíduos se esforçam para alcançar (CP, 6.16). [e] Os Lamarckistas também supuseram que, embora algumas das modificações de forma assim transmitidas foram originalmente devidas a causas mecânicas, mesmo assim **os fatores chefes de sua primeira produção foram os intensos empenhos de esforços** e o crescimento excessivo acarretados pelo exercício, juntamente com as ações opostas. Agora, **esforço, desde que ele é dirigido para um fim, é essencialmente psíquico**, mesmo que ele seja às vezes inconsciente; e **o crescimento devido ao exercício, como eu argumentei em meu último papel, segue uma lei inteiramente contrária àquela da mecânica**. (CP, 6.299, grifo nosso).

Dessa forma, Peirce acredita que a teoria de Lamarck contém o elemento teleológico que falta nas evoluções ticástica e anancástica. Segundo ele corresponde à evolução agapástica – a evolução pelo amor criativo – uma teoria mais completa da evolução que tem elementos de acaso, necessidade e também de teleologia. Peirce relaciona, portanto, o Lamarckismo com o agapismo, e afirma que o ticasmo e o anancasmo são formas degeneradas de agapismo.

O desenvolvimento do pensamento agapástico deve, se ele existir, ser distinguido pela sua característica de **ter um propósito**, sendo que esse propósito é o desenvolvimento de uma idéia (CP, 6.315).

[...] A evolução Lamarckista coincide com a descrição geral da ação do amor (CP, 6.300) [e] Todos os três modos de evolução são compostos dos mesmos elementos gerais. Agapismo os exibe mais claramente. [...] Assim sendo, ticasmo e anancasmo são formas degeneradas de agapismo (CP, 6.303).

É evidente que atualmente não se considera a teoria de Lamarck como uma teoria científica válida. Mas isso não é, de forma alguma, um argumento contra a evolução agapástica. Embora Peirce não tenha conhecido os mecanismos da hereditariedade, isso não é relevante, pois a evolução para Peirce, como vimos, não se restringe à evolução biológica,

tendo suas raízes já nas categorias fundamentais. Para ele, o importante a respeito da teoria de Lamarck é que ela permitia que a mente, de certa forma, guiasse a evolução, o que é mais condizente com suas idéias. Por esse motivo, o efeito Baldwin reforça e é reforçado pelas idéias evolucionistas de Peirce, pois apresenta o caráter teleológico que deve estar presente na evolução. A grande diferença entre a evolução guiada como descrita pelo efeito Baldwin em relação a teoria de Lamarck é uma diferença de escala. De acordo com o efeito Baldwin, a mente pode influenciar a evolução biológica, mas as mudanças podem levar gerações, e não são restritas a um único indivíduo. O biossemiótico Jesper Hoffmeyer, na citação abaixo, chama a atenção para o fato de que no nível da espécie as características adquiridas são transmitidas.

É, entretanto, importante notar que mesmo que as características adquiridas não sejam normalmente herdadas no nível do indivíduo, esse tipo de herança é exatamente o que ocorre no nível da população ou da espécie. No caso da população, o ponto principal é que o pool de genes de cada geração é o produto do que aconteceu na geração passada. **O “crime” de Lamarck foi, portanto, que ele não diferenciou entre o indivíduo e a população.** (HOFFMEYER, 1996, pp. 48-49, grifo nosso).

Chegamos ao final deste capítulo com a expectativa de que, após tudo o que foi dito, tenha ficado claro que as causas finais são realmente operativas na natureza, que a evolução é intrinsecamente guiada e isso se deve, dentre outras coisas, ao papel que a semiose, como um processo contínuo, exerce na evolução. O próximo capítulo utilizará conhecimento adquirido aqui para abordar o aspecto teleológico da evolução do software.

3. O ASPECTO TELEOLÓGICO DA EVOLUÇÃO DO SOFTWARE

Neste capítulo compreenderemos como a semiótica e as idéias evolucionistas de Peirce podem ajudar a entender por que a evolução do software é guiada. O capítulo está organizado da seguinte maneira. A primeira tarefa será compreender melhor o que realmente é o software. Essa tarefa está diretamente relacionada à de entender o que é um computador, e ambos os conceitos são abrangidos pela idéia de máquina de Turing. Dessa forma, estudaremos primeiramente a máquina de Turing, definindo o que é o computador e o que é o software. Em seguida abordaremos a relação entre a máquina de Turing, as linguagens, e as mensagens. Veremos como é possível existir vários níveis de software, e como o software evolui através da interação entre as linguagens e as mensagens. Verificaremos como a evolução do software é guiada através da interação entre os diferentes níveis evolutivos. Por último, apresentaremos as conclusões que foram tiradas deste trabalho, oferecendo algumas sugestões sobre possíveis desenvolvimentos futuros.

3.1. Entendendo a Máquina de Turing de Forma Mais Intuitiva

A tese de Church-Turing é uma hipótese que diz que todo cálculo possível pode ser feito por um algoritmo rodando em um computador que tenha tempo e capacidade de armazenamento suficientes. Na sua versão da tese, o matemático britânico Alan Turing, considerado o pai da ciência da computação, concebeu uma máquina abstrata chamada de máquina de Turing e argumentou que toda máquina computacional é, na prática, uma máquina de Turing.

O processo que levou Turing às suas conclusões partiu da reflexão acerca de como as pessoas fazem cálculos ou realizam computações, e da possibilidade disso ser feito por máquinas. Em seu famoso artigo “Computing Machinery and Intelligence” (TURING, 1950), ele descreve quais os requisitos mínimos para que uma máquina possa fazer computação, de forma análoga a como uma pessoa, que não tivesse o conhecimento acerca do cálculo a ser realizado, faria. Segundo ele, para essa pessoa (um “computador humano”) realizar essa tarefa, ela deveria seguir regras fixas não tendo autoridade para desviar em nenhum detalhe. Essas regras seriam fornecidas em um livro, que seria alterado sempre que se desejasse realizar outra computação. Essa pessoa contaria também com uma quantidade ilimitada de papel no qual ela poderia fazer os cálculos.

É importante fornecer um exemplo de uma computação simples, resolvida de forma similar a como esse “computador humano” de Turing resolveria. Dessa forma, além de ilustrar de maneira aproximada a linha de raciocínio que Turing pode ter seguido, estaremos fornecendo um modelo mais intuitivo para podemos pensar a máquina de Turing nas discussões que se seguem.

Para facilitar ao máximo a compreensão, pensaremos em um problema artificialmente simples: somar dois números de um dígito cada. Para seguir o exemplo de Turing, a pessoa que irá resolver esse problema terá que seguir regras fixas, não tendo autonomia para alterá-las. As regras ou instruções estarão escritas em um livro na ordem em que elas devem ser executadas, começando pela instrução zero. A pessoa terá uma folha de papel na qual ela realizará os cálculos, que, em nosso exemplo, terá linhas numeradas em ordem crescente, começando na linha zero.

É necessário que as instruções que a pessoa executará sejam as mais simples possíveis. Portanto, no nosso problema ilustrativo, teremos apenas as instruções:

- Escrever uma marca M em uma linha L da folha de papel.
- Pular para a instrução I, ou seja, executar a instrução I e continuar na seqüência.
- Pular para uma instrução I, no caso de uma linha L conter uma marca M.
- Escrever uma marca M em uma linha L, no caso de uma linha L₂ conter a marca M₂, sendo que M pode ser igual a M₂ e L pode ser igual a L₂.
- Parar

Note que a primeira instrução prescreve a capacidade de produzir uma marca que pode ser posteriormente identificada, a segunda, a capacidade de desviar o fluxo de execução para outra instrução e a terceira e a quarta, a capacidade de identificar uma marca. A última determinará quando o cálculo acabou.

Para resolver esse problema especificamente, precisaremos apenas de quatro linhas da folha de papel. As linhas 0 e 1 conterão os números de um dígito cada que serão somados, e as linhas 2 e 3 o resultado da soma, que pode ocupar dois dígitos. Nesse caso, a linha 3 guardará a dezena do resultado e a linha 2, a unidade. Portanto, a folha de papel após a operação 9 + 9 ter sido executada deverá ter a seguinte configuração:

```

linha 0: 9
linha 1: 9
linha 2: 8
linha 3: 1

```

A seguir são enumeradas as instruções que estariam contidas no livro entregue à pessoa incumbida de realizar a operação em questão. Como queremos argumentar que podemos realizar o cálculo somente usando instruções semelhantes às dadas acima, é fundamental fornecer todas as instruções integralmente.

```

000: Se a linha 0 contiver '1', pule para instrução 28.
001: Se a linha 0 contiver '2', pule para instrução 49.
002: Se a linha 0 contiver '3', pule para instrução 70.
003: Se a linha 0 contiver '4', pule para instrução 91.
004: Se a linha 0 contiver '5', pule para instrução 112.
005: Se a linha 0 contiver '6', pule para instrução 133.
006: Se a linha 0 contiver '7', pule para instrução 154.
007: Se a linha 0 contiver '8', pule para instrução 175.
008: Se a linha 0 contiver '9', pule para instrução 196.
009: Se a linha 1 contiver '0', escreva '0' na linha 2.
010: Se a linha 1 contiver '0', escreva '0' na linha 3.
011: Se a linha 1 contiver '1', escreva '1' na linha 2.
012: Se a linha 1 contiver '1', escreva '0' na linha 3.
013: Se a linha 1 contiver '2', escreva '2' na linha 2.
014: Se a linha 1 contiver '2', escreva '0' na linha 3.
015: Se a linha 1 contiver '3', escreva '3' na linha 2.
016: Se a linha 1 contiver '3', escreva '0' na linha 3.
017: Se a linha 1 contiver '4', escreva '4' na linha 2.
018: Se a linha 1 contiver '4', escreva '0' na linha 3.
019: Se a linha 1 contiver '5', escreva '5' na linha 2.
020: Se a linha 1 contiver '5', escreva '0' na linha 3.
021: Se a linha 1 contiver '6', escreva '6' na linha 2.
022: Se a linha 1 contiver '6', escreva '0' na linha 3.
023: Se a linha 1 contiver '7', escreva '7' na linha 2.
024: Se a linha 1 contiver '7', escreva '0' na linha 3.
025: Se a linha 1 contiver '8', escreva '8' na linha 2.
026: Se a linha 1 contiver '8', escreva '0' na linha 3.
027: Pare.
028: Se a linha 1 contiver '0', escreva '1' na linha 2.
029: Se a linha 1 contiver '0', escreva '0' na linha 3.
030: Se a linha 1 contiver '1', escreva '2' na linha 2.
031: Se a linha 1 contiver '1', escreva '0' na linha 3.
032: Se a linha 1 contiver '2', escreva '3' na linha 2.
033: Se a linha 1 contiver '2', escreva '0' na linha 3.
034: Se a linha 1 contiver '3', escreva '4' na linha 2.
035: Se a linha 1 contiver '3', escreva '0' na linha 3.
036: Se a linha 1 contiver '4', escreva '5' na linha 2.
037: Se a linha 1 contiver '4', escreva '0' na linha 3.
038: Se a linha 1 contiver '5', escreva '6' na linha 2.
039: Se a linha 1 contiver '5', escreva '0' na linha 3.
040: Se a linha 1 contiver '6', escreva '7' na linha 2.
041: Se a linha 1 contiver '6', escreva '0' na linha 3.
042: Se a linha 1 contiver '7', escreva '8' na linha 2.
043: Se a linha 1 contiver '7', escreva '0' na linha 3.
044: Se a linha 1 contiver '8', escreva '9' na linha 2.
045: Se a linha 1 contiver '8', escreva '0' na linha 3.
046: Se a linha 1 contiver '9', escreva '0' na linha 2.
047: Se a linha 1 contiver '9', escreva '1' na linha 3.
048: Pare.
049: Se a linha 1 contiver '0', escreva '2' na linha 2.
050: Se a linha 1 contiver '0', escreva '0' na linha 3.
051: Se a linha 1 contiver '1', escreva '3' na linha 2.
052: Se a linha 1 contiver '1', escreva '0' na linha 3.
053: Se a linha 1 contiver '2', escreva '4' na linha 2.
054: Se a linha 1 contiver '2', escreva '0' na linha 3.
055: Se a linha 1 contiver '3', escreva '5' na linha 2.
056: Se a linha 1 contiver '3', escreva '0' na linha 3.

```



```

211: Se a linha 1 contiver '7', escreva '1' na linha 3.
212: Se a linha 1 contiver '8', escreva '7' na linha 2.
213: Se a linha 1 contiver '8', escreva '1' na linha 3.
214: Se a linha 1 contiver '9', escreva '8' na linha 2.
215: Se a linha 1 contiver '9', escreva '1' na linha 3.
216: Pare.

```

Dadas as instruções, vamos supor que se deseja somar os números 2 e 3. Para tanto, a folha de papel deve estar configurada da seguinte forma:

```

linha 0 : 2
linha 1 : 3
linha 2 : 0
linha 3 : 0

```

A pessoa deve então começar a obedecer cegamente a cada instrução, começando da 000:

```

000: Se a linha 0 contiver '1', pule para instrução 28.
001: Se a linha 0 contiver '2', pule para instrução 49.

```

Repare que ao encontrar a instrução 001, a pessoa passa a ler as instruções da linha 049.

```

049: Se a linha 1 contiver '0', escreva '2' na linha 2.
050: Se a linha 1 contiver '0', escreva '0' na linha 3.
051: Se a linha 1 contiver '1', escreva '3' na linha 2.
052: Se a linha 1 contiver '1', escreva '0' na linha 3.
053: Se a linha 1 contiver '2', escreva '4' na linha 2.
054: Se a linha 1 contiver '2', escreva '0' na linha 3.
055: Se a linha 1 contiver '3', escreva '5' na linha 2.

```

A instrução 055 manda, pela primeira vez, que algo seja escrito no papel, que fica assim:

```

linha 0 : 2
linha 1 : 3
linha 2 : 5
linha 3 : 0

```

Continuando:

```

056: Se a linha 1 contiver '3', escreva '0' na linha 3.
057: Se a linha 1 contiver '4', escreva '6' na linha 2.
...
069: Pare

```

Ao encontrar a instrução 069, se considera o cálculo terminado e o resultado pode ser encontrado nas linhas 2 e 3 do papel:

```
linha 0 : 2
linha 1 : 3
linha 2 : 5
linha 3 : 0
```

Encorajamos o leitor a tentar outros números e refletir acerca da simplicidade das instruções.

3.2. Uma Definição Mais Formal da Máquina de Turing

A primeira impressão que se tem ao ver a quantidade de instruções executadas para realizar um cálculo tão simples é que deve existir uma forma mais inteligente de se fazer isso. Esse é um ponto importante, pois a conclusão que Turing chegou foi justamente a que por mais complexos que os cálculos possam ser eles podem sempre ser reduzidos a uma seqüência de instruções simples, que não demandam inteligência. Portanto, a inteligência pode ser decomposta em partes cada vez menos inteligentes até chegar em um mecanismo “estúpido”.

Partindo desse raciocínio, Turing descobriu que as instruções poderiam ser até mais simples que as do exemplo dado acima, e concluiu que, dada a simplicidade das instruções, elas poderiam ser executadas por uma máquina. A máquina de Turing, é uma máquina de estados abstrata que pode estar em um único estado de um número finito deles. Nela o papel foi substituído por uma fita, que tem início mas não tem fim, e na qual as marcas são escritas e lidas. A máquina possui uma cabeça que se move sobre a fita que vai escrever e ler as marcas. O conjunto de marcas deve ser finito e é chamado de alfabeto. O livro de instruções foi substituído por uma tabela de transição de estados, que determina qual ação a máquina deve executar a cada instante. Cada ação depende do estado atual da máquina e da marca que está sendo lida pela cabeça de leitura no momento. As ações permitem à máquina: mudar de estado, mover cabeça para esquerda ou direita e escrever uma marca na posição atual. A mudança de estados e a escrita de uma marca são opcionais.

Formalmente, a máquina de Turing pode ser descrita por uma 6-tupla, $(Q, \Sigma, T, \delta, q_{\text{inicial}}, q_{\text{aceita}})$, onde:

1. Q é o conjunto de estados
2. Σ é o alfabeto de entrada não contendo a marca especial \square
3. T é o alfabeto da fita, onde $\square \in T$ e $\Sigma \subseteq T$

4. $\delta: Q \times T \rightarrow Q \times T \times \{L, R\}$ é a função ou tabela de transição
5. $q_{inicial} \in Q$ é o estado inicial
6. $q_{aceita} \in Q$ é o estado de aceitação

O estado de aceitação indica um estado no qual ao chegar, a máquina deve parar sua execução. Alguns autores costumam descrever um sétimo elemento que é o estado de rejeição, mas para manter a definição mais simples não precisamos considerá-lo. Para melhor compreender essa definição vamos ilustrá-la através de um exemplo simples. Vamos construir uma máquina de Turing que simplesmente lê a fita da esquerda para a direita e ao encontrar o símbolo ‘a’ substitui pelo símbolo ‘b’, até que ela encontre o símbolo ‘□’ e pare. A descrição formal de tal máquina é:

$$Q = \{q_0, q_1\}$$

$$\Sigma = \{a, b\}$$

$$T = \{a, b, \square\}$$

$$q_{inicial} = q_0$$

$$q_{aceita} = q_1$$

Falta ainda determinar o coração da definição de uma máquina de Turing que é a função de transição δ , que nos diz como a máquina vai de um estado para o próximo. Ela pode ser descrita por uma tabela:

Estado	Símbolo		
	a	b	□
q_0	(q_0, b, R)	-	(q_1, \square, R)
q_1	-	-	-

Tabela 1

O significado dessa tabela é o seguinte. Na intersecção entre a marca ‘a’ e o estado q_0 , encontramos a tripla (q_0, b, R) que significa que quando a máquina estiver no estado q_0 e a cabeça ler ‘a’ a ela escreverá ‘b’ na mesma posição, continuará no estado q_0 e moverá a cabeça para a direita ($R=$ Direita, $L=$ Esquerda). Na intersecção entre a marca ‘□’ e o estado q_0 temos a tripla (q_1, \square, R) que significa que quando a máquina estiver no estado q_0 e encontrar

‘□’ ela escreverá ‘□’, mudará o estado para q_1 e moverá a cabeça para a direita. Note que para as outras combinações não existe transições.

A tabela de transição está representada graficamente na Figura 3 abaixo

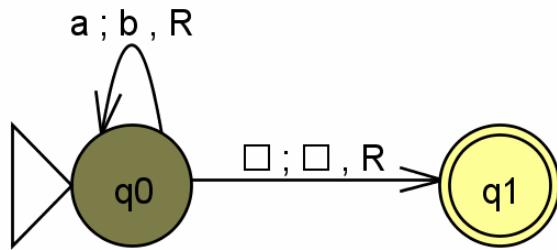


Figura 3

No grafo da Figura 3, os vértices representam os estados (q_0 e q_1) e as arestas representam as transições para um novo estado. A inscrição sob a aresta contém, na ordem, a marca que deve ser lida pela cabeça para que a transição ocorra, a marca que será escrita em seu lugar, e a direção que a cabeça deve se mover após a transição. O estado inicial é o que tem um triângulo ao lado, e o estado de aceitação tem uma borda dupla. A cor mais escura do estado q_0 indica que ele é o estado atual.

A Figura 4 contém uma possível configuração inicial da fita.

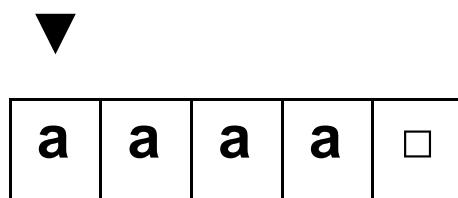


Figura 4

Na Figura 4, notamos que a cabeça da máquina é representada por um triângulo preto, que inicialmente é posicionado sobre a primeira célula da fita. De acordo com o nosso exemplo, o estado inicial da máquina é q_0 , e portanto sua primeira ação será ler o ‘a’ contido

sob a cabeça, substituir por ‘b’, mover a cabeça para a direita, e permanecer no mesmo estado. O resultado dessa ação pode ser visualizado na Figura 5.

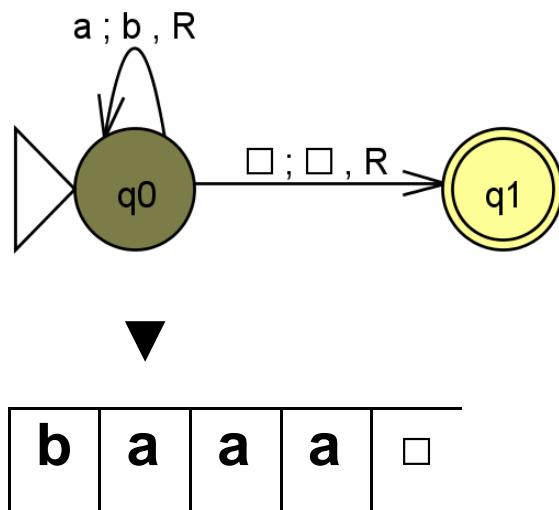


Figura 5

A máquina repetirá a mesma operação até que ela encontre o símbolo ‘ \square ’ e mude para o estado q_1 , que é o estado de aceitação (Figura 6). Ao encontrar o estado de aceitação, a máquina pára.

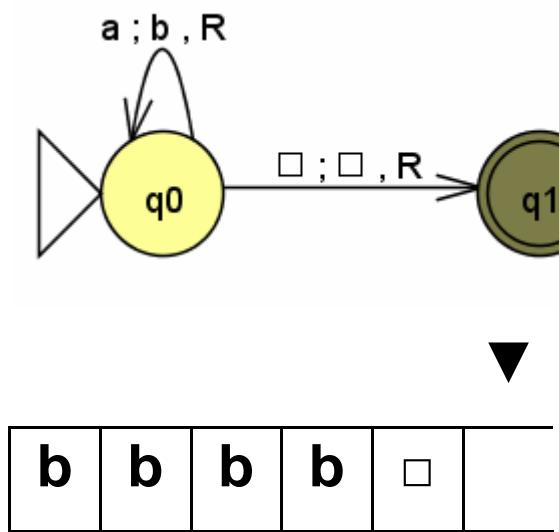


Figura 6

Temos agora uma idéia mais precisa sobre o que é uma máquina de Turing e como Turing chegou a ela. Poderíamos, se assim quiséssemos, reescrever o problema da soma de dois números, traduzindo as instruções de “alto nível”, e implementando-o como uma máquina de Turing. Mas, como a representação dessa máquina ficaria muito grande, iremos apresentar, como ilustração, somente uma parte dela. A máquina da Figura 8 soma o dígito ‘1’ com qualquer outro dígito e escreve o resultado na seqüência. A fita deve conter, portanto, o dígito ‘1’, um outro dígito qualquer e a marca especial ‘□’, como na Figura 7.

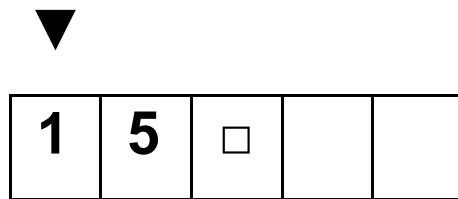


Figura 7

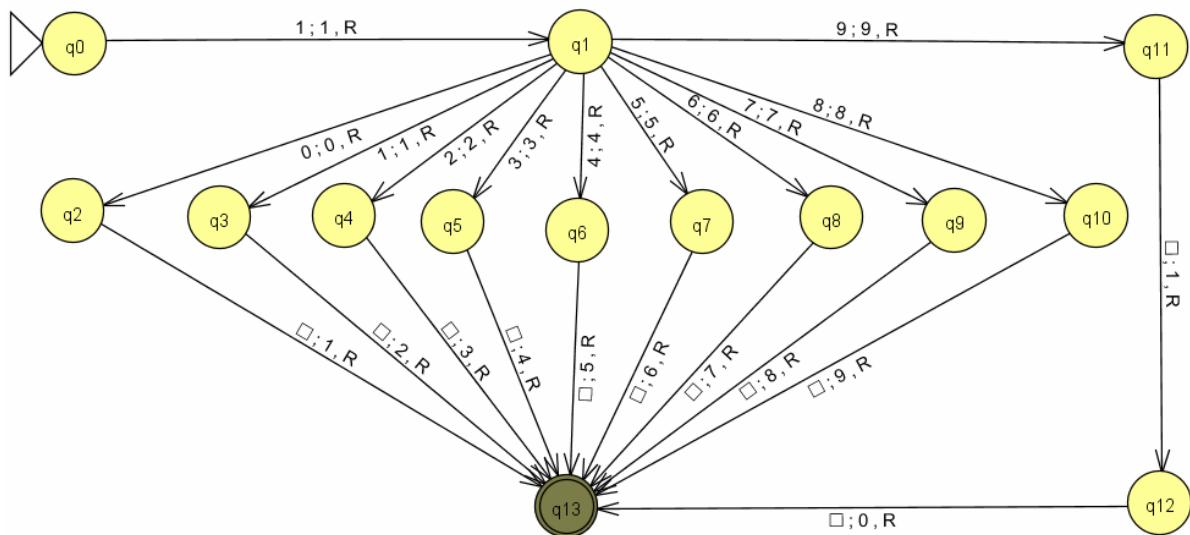


Figura 8

Após a execução a fita apresentará o resultado da soma (Figura 9)

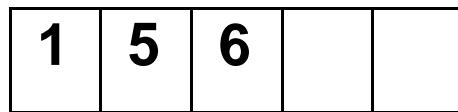


Figura 9

Para que a máquina fosse completa, ela teria que somar qualquer dígito com qualquer outro. Como dissemos, não apresentaremos tal máquina aqui por questão de espaço, mas o leitor é encorajado a completá-la.

Como podemos observar, a grande diferença que existe entre os dois modelos é que, na máquina de Turing, podemos mudar para qualquer estado dependendo do estado atual e da marca sob a cabeça, mas não podemos fazer com que a cabeça “pule” para uma posição qualquer da fita, da mesma forma como os comandos de “alto nível” podem escrever ou ler de qualquer linha do papel. Mas, essa leitura e gravação aleatórias (em qualquer posição da fita) poderiam ser implementadas na máquina de Turing, bastando, para tanto, marcar determinadas células da fita que indicariam onde determinada informação começaria e terminaria. Assim “caminhando” para direita ou esquerda, de célula em célula, a máquina encontraria a posição desejada.

Levando em consideração a equivalência entre os dois modelos, passaremos a utilizar o modelo de “alto nível” e da máquina de Turing de forma intercambiável, dependendo da situação, para que a compreensão se torne mais fácil.

Já podemos, neste momento, compreender o que é um computador. Podemos dizer, de forma simplificada, que uma máquina de Turing é equivalente a um computador com um programa fixo. Mas, nossos computadores modernos não rodam apenas um programa. Um elemento está faltando. Turing imaginou que se fosse possível codificar a tabela de transição de qualquer máquina de Turing em uma seqüência de marcas, seria possível gravar essa seqüência na própria fita da máquina. Essa nova máquina, que fosse capaz de ler a tabela de transição de sua fita, seria mais flexível que as outras, e poderia simular qualquer uma delas. Essa máquina ficou conhecida como máquina universal de Turing.

Pode ser mostrado que uma única máquina especial desse tipo pode ser construída para fazer o trabalho de todas. Ela poderia na verdade ser construída para servir de modelo para qualquer máquina. Essa máquina especial deve ser chamada **máquina universal**. (TURING, 1986b, grifo nosso)

A importância dessa máquina universal é clara. Nós não precisamos ter uma infinidade de máquinas diferentes realizando trabalhos diferentes. Uma única é o suficiente. O problema de engenharia de produzir várias máquinas para vários trabalhos é substituído pelo trabalho de escritório de “programação” de uma **máquina universal** para fazer esses trabalhos (TURING, 1986a, p. 7, grifo nosso)

Os nossos computadores modernos são máquinas universais de Turing, e seus programas são as máquinas de Turing que eles simulam. Software, portanto, é a descrição

codificada de uma máquina de Turing que será simulada por outra máquina universal de Turing – o computador.

Providos de uma maior compreensão sobre a natureza do software, já podemos abordar a questão do aspecto teleológico da sua evolução. O primeiro passo nesse sentido é compreender a relação do software com as linguagens e as mensagens.

3.3. Software, Linguagens e Mensagens

No capítulo anterior, dissemos que a evolução equivale ao crescimento de representações gerais que se dá a através de um processo de síntese das ocorrências individuais. Demos o nome de **linguagem** ao sistema de representações gerais, e chamamos as ocorrências individuais de **mensagens**. Chegamos à conclusão que é possível dizer que a evolução, de forma geral, se dá através da produção de mensagens e posterior síntese dessas mensagens nas linguagens. Para estabelecer a relação entre a semiose e a evolução do software, e assim compreender o aspecto teleológico da evolução deste, é necessário compreender a relação entre máquina de Turing, linguagens e mensagens.

Uma determinada seqüência de marcas na fita de uma máquina de Turing é uma mensagem. Lembremos que a máquina de Turing tem um estado chamado “estado de aceitação”, que indica que a computação chegou ao fim, da maneira esperada. Acontece que, mesmo que uma fita contenha uma mensagem cujos elementos são todos pertencentes ao alfabeto da máquina, pode ser o caso que o estado de aceitação nunca seja alcançado. Para ilustrar isso, vamos relembrar a máquina de Turing da Figura 3 acima.

Essa máquina lê uma seqüência de ‘a’s que ela substitui por ‘b’s até encontrar ‘□’. Dessa forma, todas as mensagens abaixo permitem que a máquina alcance o estado de aceitação.

□
a□
aa□
aaa□
aaaa□

Agora, considere a seguinte mensagem: **aab□**. Ao interpretá-la, a máquina nunca alcançará o estado de aceitação. Portanto, existem determinadas mensagens que uma máquina

de Turing aceita ou reconhece, e outras não. Chamamos de linguagem o conjunto de todas as mensagens aceitas pela máquina de Turing. Logo, toda máquina de Turing reconhece uma linguagem.

3.4. Linguagem de Máquina

Como já dissemos, um computador é uma máquina universal de Turing, e, nesse tipo de máquina, a tabela de transição de outra máquina – a máquina simulada – é escrita na fita. No caso do computador, a memória faz o papel da fita, e, dessa forma, as marcas que correspondem às instruções que definem a máquina simulada estão presentes nessa memória, juntas com outras marcas, que serão utilizadas pela máquina simulada. Dizemos que essas instruções são os programas e as outras marcas são os dados utilizados pelos programas. Programar um computador, portanto, significa “ensiná-lo” a simular outra máquina de Turing. Toda máquina de Turing reconhece uma linguagem, no caso do computador, esta é chamada de linguagem de máquina.

Um programa para “rodar” em um computador deve ser escrito na linguagem de máquina daquele computador. Para entender melhor no que consiste uma linguagem de máquina, é necessário saber um pouco mais sobre a arquitetura interna do computador e como ele executa um programa. Um computador contém uma unidade central de processamento ou CPU (Central Processing Unit) além de dispositivos de entrada e saída (teclado, mouse, monitor, impressora, etc). Todos esses elementos estão ligados através fios que chamamos de barramento. A CPU é responsável por ler, através do barramento, as instruções da memória e executar uma ação que corresponde àquela instrução. A linguagem de máquina, portanto, corresponde ao conjunto das instruções que a CPU do computador reconhece.

Na maioria dos computadores digitais modernos, tanto as instruções quanto os dados utilizados por elas são representados na memória como números binários, ou seja, seqüências de zeros e uns. Isso foi feito para facilitar sua construção, e, pelo mesmo motivo, ele sempre lê as informações em pedaços de tamanho definidos, que geralmente são de 8, 16, 32 ou 64 dígitos binários ou bits. Cada pedaço de informação tem um endereço, ou seja, tem um número que possibilita a CPU ler diretamente aquela informação da memória. As instruções que compõe um programa estão dispostas na memória em seqüência.

A CPU possui um número pequeno de registradores, que constituem uma memória interna, própria, de acesso mais rápido, que ela utiliza para ajudar a realizar suas operações. Geralmente, cada registrador tem em bits o tamanho igual ao tamanho dos pedaços da

memória, ou seja, se um computador lê a memória em pedaços de 32 bits, os registradores geralmente têm 32 bits também. A CPU tem um registrador especial que se chama ponteiro de instrução, que contém o endereço (um número) da próxima instrução que a CPU deve ler da memória.

Quando o computador é ligado, o ponteiro de instrução é configurado para “apontar” para a primeira instrução da memória. O computador funciona carregando da memória a instrução “apontada”, executando essa instrução, incrementando o ponteiro de instrução e repetindo o processo. Pode acontecer de uma instrução mudar diretamente o ponteiro de instrução, dessa maneira a CPU executa um “salto”, e passa executar instruções a partir de outro endereço.

Os bits que compõem uma instrução, além de determinarem qual operação, também podem conter parâmetros adicionais. Por exemplo, uma instrução para somar dois números pode especificar que sejam somados dois números contidos em dois registradores da CPU, ou um número contido em um registrador e outro na memória, ou um número que já está contido nos próprios bits da instrução e outro no registrador. A Tabela 2 ilustra uma possível configuração da memória de um computador hipotético, que é acessada em pedaços de 16 bits cada.

Endereço do “pedaço” da memória	Conteúdo da memória naquele endereço (em bits)
00000000	1010110111101100
00000001	1001101010101101
00000002	1110110010011010
00000003	1001101011101100

Tabela 2

Na memória representada na tabela acima, qualquer endereço pode conter tanto uma instrução como qualquer outra informação. As marcas são arbitrárias e a interpretação dependerá da CPU. O número 1010110111101100, por exemplo, poderia ser uma instrução que ordenasse ao computador somar dois números. Abaixo (Tabela 3) fornecemos um exemplo de uma possível tabela de instruções do nosso computador hipotético, para o leitor ter uma idéia das instruções que são comuns em linguagem de máquina.

Instrução	O que faz?
1011001100110000	Armazena no registrador A o número representado pelos últimos 8 bits da instrução
1010110111101100	Soma o conteúdo do registrador A com o conteúdo do registrador B, e armazena o resultado em A.
1001101010101101	Salta para um endereço especificado nos últimos 12 bits da instrução.
1110110010011010	Subtrai do registrador A o número representado pelos últimos 8 bits da instrução.
1001101011101100	Compara o número contido no registrador A com o número contido no registrador B.

Tabela 3

Um programa em linguagem de máquina poderia ser similar a seqüência de zeros e uns abaixo:

```

10011010101101
1011001100110000
1110110010011010
1001101011101100
1010110111101100
1001100101001100
0100100010010010
0100100101110010

```

É fácil perceber que escrever um programa em linguagem de máquina é uma tarefa árdua e suscetível a erros. A complexidade dos programas que podem ser escritos nessa linguagem é restrita diante da limitação do ser humano em lidar com essa complexidade. Mas, existem outras maneiras de programar um computador, que não seja na sua própria linguagem, como veremos a seguir.

3.5. Os Vários Níveis do Software

Se uma máquina universal de Turing pode simular qualquer outra máquina de Turing, ela também pode simular outra máquina universal de Turing. Lembremos que uma máquina universal de Turing armazena, na própria fita, a tabela de transição de outra máquina de Turing qualquer. Acontece que essa outra máquina de Turing também pode ler, da mesma fita, a tabela de transição de uma terceira máquina de Turing.

Na prática, isso significa que utilizando a linguagem de máquina, é possível escrever um programa que lê outro programa, escrito em outra linguagem, e o execute. O programa que faz isso, ou seja, que interpreta ou simula outro, é chamado de interpretador. Mas também existe a possibilidade de ler um programa escrito em uma linguagem diferente da de máquina, e traduzi-lo para linguagem de máquina de uma só vez, para depois executá-lo. O tipo de programa que faz isso é chamado de tradutor ou compilador. A diferença entre as tarefas desses dois tipos de programas é parecida com a diferença entre traduzir em tempo real o que alguém está falando de uma língua para outra, em relação a traduzir um texto pronto de uma vez só.

Assim, embora o computador só “entenda” a linguagem de máquina, ele pode “entender” programas escritos em outra linguagem, contanto que se tenha um programa, escrito em linguagem de máquina, que interprete ou traduza esses outros programas. O mais importante é que esse processo pode ser repetido infinitamente, em vários níveis. Dessa forma, usando linguagem de máquina é possível escrever um programa P_0 que reconhece programas escritos em outra linguagem, digamos, L_0 . Usando L_0 é possível escrever um programa P_1 , que reconhece outros programas escritos em uma linguagem L_1 . É possível, então, utilizar L_1 para escrever um programa P_2 , que reconhece outros programas escritos em linguagem L_2 , e assim por diante.

No capítulo anterior, argumentamos que a evolução é guiada porque existem vários níveis evolutivos, e que o nível mais flexível pode ser visto como uma simulação que guia a evolução do nível mais rígido. Também afirmamos que a evolução se dá através da interação entre linguagens e mensagens. Dissemos que mensagens são geradas, e, posteriormente, através do processo de síntese que corresponde à evolução, são incorporadas nas linguagens. Já estamos em condição de verificar porque é possível afirmar que isso ocorre no software.

O computador é uma máquina universal de Turing, que pode simular outra máquina universal de Turing, que, por sua vez, pode simular outra, e assim por diante. Existem, dessa forma, várias camadas, cada uma equivalendo a uma máquina de Turing específica. Acontece

que cada máquina de Turing reconhece uma linguagem, e cada linguagem muda através de um processo que fixa (sintetiza) nela mesma, elementos dos programas escritos com ela. Além dos programas mudarem mais rapidamente que as linguagens, cada camada, ou cada linguagem, muda mais rapidamente que a anterior. Assim, a evolução do nível mais flexível guia a evolução do nível mais rígido. Como iremos ilustrar com mais detalhes nos próximos tópicos, é assim que o software evolui.

Nesse momento é importante tocar em um assunto que não foi discutido ainda. Estamos falando que o software evolui, mas é preciso salientar que ele não faz isso sozinho. Como vimos, uma máquina de Turing produz signos a partir de outros signos seguindo uma tabela de transição, mas ela não é capaz de criar essa tabela. Essa tarefa é realizada pelos programadores. O computador ainda não tem criatividade e não tem a capacidade de aprender sozinho. Portanto, podemos dizer que o computador não realiza semiose, mas sim quasi-semiose, como evidencia Winfried Nöth.

A evidência da natureza quasi-semiótica do processamento de dados vem da natureza diádica dos signos envolvidos. A visão de que o processamento de signos em computadores é baseado em relações diádicas está implícita em uma teoria amplamente sustentada que afirma que computadores podem processar apenas *sinais*, isto é, estímulos mecânicos seguidos por reações automáticas (NÖTH, 2003, p. 85)

Entretanto, ainda segundo Nöth, ocorre verdadeira semiose na interface entre o homem e o computador.

[...] processamento de signos na interface entre humanos e computadores é semiose genuína. Signos são produzidos por humanos, mediados por máquinas, e interpretados por humanos (NÖTH, 2003, p. 86)

É o homem quem fornece o elemento de criatividade e a capacidade de aprender necessários para criar novos programas e aprender com a experiência. O software evolui na mente humana e, assim, poderíamos dizer que a evolução do software é um elemento da evolução cultural, e não poderia ser diferente da evolução de qualquer outra idéia ou artefato criado pelo homem. Seria válido questionar se faz sentido falar sobre a evolução software especificamente.

Argumentamos que é importante abordar a evolução do software de forma específica sim, pois o software é diferente de qualquer outra idéia ou artefato humano. O que distingue o software é sua natureza lingüística. Toda máquina de Turing reconhece uma linguagem, e todo programa é uma mensagem. Falta muito menos para que o computador adquira a

capacidade de realizar semiose genuína, e o software adquira a capacidade de evoluir sozinho, do que para que qualquer outro artefato humano adquira a mesma capacidade. A necessidade de um elemento humano não invalida dizer que o software evolui, contanto que estejamos conscientes disso. Além disso, as linguagens de programação e os programas são criados conscientemente pelos homens. Decorre disso que se torna muito mais fácil entender como o aspecto teleológico da evolução se dá através da interação entre mensagem e linguagem, estudando a evolução do software, do que estudando qualquer outra forma de evolução.

Após ter esclarecido esse ponto, o próximo passo é analisar mais detalhadamente o aspecto teleológico da evolução do software.

3.6. A Evolução do Software é Guiada

Como vimos, programar em linguagem de máquina é uma tarefa difícil. Geralmente, quando se cria um novo computador, se cria um programa chamado **assembler**, que lê outros programas escritos em uma linguagem chamada **assembly** e os traduz para a linguagem de máquina. O assembly é uma linguagem que é feita para ser muito parecida com a linguagem de máquina, sendo que cada instrução em linguagem assembly equivale a uma instrução em linguagem de máquina. A diferença é que as instruções em assembly são mnemônicos que ajudam o programador a lembrar o significado da operação. Abaixo (Tabela 4) estão comandos que são comuns em linguagens assembly, que, no caso, estão mapeados a uma linguagem de máquina hipotética.

Linguagem de máquina	Linguagem Assembly	O que faz?
1011001100110000	ADD	Soma dois números
1001011001100111	JUMP	Salta para uma determinada instrução
1110110010110011	SUB	Subtrai dois números
1001101000010010	CMP	Compara dois números
1001111100100111	MOV	Move um número da memória para o registrador ou vice-versa
1011001111101100	JL	Salta para uma instrução se após uma comparação um

		número é menor que o outro
1110011110001100	INC	Incrementa o número que está no registrador na memória

Tabela 4

Repare que escrever um programa em linguagem assembly é muito mais simples que escrever o mesmo programa em linguagem de máquina. A listagem abaixo contém o trecho de um programa escrito em assembly para os computadores com processadores Intel da família x86. O leitor não precisa entender o que o programa faz. Basta constatar que é muito mais fácil lembrar o significado da instrução **add** do que o seu equivalente em linguagem de máquina (algo similar a 1011001100110000).

```

mov  bx, 0
mov  ax, 0
loop1:
add  bx, 7
inc  ax
cmp  ax,10
jl   loop1

```

Note que com o surgimento da linguagem assembly se tornou desnecessário utilizar a linguagem de máquina, e passou a ser possível escrever programas mais complexos. Embora a linguagem assembly não tenha nenhuma instrução a mais do que a linguagem de máquina, já é possível verificar a existência de níveis diferentes no software. Também é possível perceber a relação entre as linguagens e as mensagens. O assembler, sendo um programa, é uma mensagem reconhecida pelo computador, mas ele é também uma linguagem, que reconhece outras mensagens, que não são reconhecidas diretamente pelo computador.

Mas a evolução do software não parou na linguagem assembly. Após ter utilizado assembly durante algum tempo, a complexidade dos programas que poderiam ser construídos também chegou a um limite. Aliado a isso, os programadores acabaram descobrindo que muitas das coisas que eles faziam, utilizando assembly, eram repetitivas.

Parece que existiam [...] alguns padrões fundamentais que surgiram naturalmente quando os seres humanos tentaram formular algoritmos. [...] Em outras palavras, algoritmos pareciam ter certos componentes de nível mais alto, em termos dos quais eles poderiam ser muito mais facilmente e esteticamente especificados do que na bastante restrita linguagem de máquina, ou linguagem assembly. (HOFSTADTER, 1980, p. 292)

Foram criados programas, similares ao assembler, mas desta vez, capazes de traduzir programas escritos em linguagens de alto nível para linguagem de máquina. A diferença é que

uma instrução de uma linguagem de alto nível equivale a várias instruções de linguagem de máquina.

Tipicamente, um componente de alto nível de um algoritmo consiste não de uma ou duas instruções de linguagem de máquina, mas de uma coleção inteira delas, não necessariamente todas contíguas na memória. Tal componente pode ser representado em uma linguagem de alto nível por um único termo – um pedaço. (HOFSTADTER, 1980, p. 292)

Exemplificando, para somar 3 números que estão na memória e colocar o resultado em outra posição da memória em assembly, precisaríamos de 4 instruções.

```
mov  ax, a
add  ax, b
add  ax, c
mov  r, ax
```

Programa 1

Para obtermos o mesmo resultado utilizando uma linguagem de nível mais alto, como a linguagem C, por exemplo, podemos utilizar uma única instrução que é muito mais intuitiva e fácil de entender:

```
r = a + b + c;
```

Programa 2

É fácil perceber que a evolução do assembler para linguagens de nível mais alto foi guiada. Vários programas foram construídos em assembly para depois se perceber que alguns deles se repetiam muito e podiam ser sintetizados em elementos da linguagem. Dessa maneira surgiram linguagens como a linguagem C. Estamos vendo como as linguagens evoluem guiadas pelos programas.

Mas, com o surgimento de linguagens de alto nível, criou-se uma camada intermediária entre a linguagem e os programas. As linguagens de alto nível trouxeram mecanismos que permitem que um bloco de comandos da própria linguagem seja criado e receba um nome. Esse bloco poderá, depois disso, ser chamado em qualquer outra parte do programa. Esse mecanismo permite que um programa elimine suas próprias repetições, através de um processo de identificação de padrões dentro dele mesmo.

Era claro que a mais poderosa adição a qualquer linguagem de programação seria a habilidade de *definir* novas entidades de alto nível em termos das previamente conhecidas, e chamá-las pelo nome. Isso traria a capacidade de criar blocos diretamente para a linguagem. Ao invés de existir um repertório determinado de instruções com os quais todos os programas tinham que ser explicitamente montados, o programador poderia construir seus próprios módulos, cada

um com seu próprio nome, cada um utilizável em qualquer lugar dentro do programa, justamente como se ele fosse uma característica embutida da linguagem. (HOFSTADTER, 1980, p. 292)

Esses blocos de instrução são chamados sub-rotinas. Vale lembrar que a possibilidade de isolar uma seqüência de instruções em um bloco e executá-lo repetidas vezes sempre esteve presente na linguagem de máquina e assembly. É possível fazer isso através das instruções que desviam o fluxo de execução. Acontece que nessas linguagens essa tarefa não é uma das mais simples, e na medida em que a complexidade do programa aumenta, ela fica cada vez mais difícil. A vantagem das linguagens de alto nível, nesse aspecto, é que elas tornaram esse processo muito mais simples.

O programa abaixo, escrito em Pascal, exemplifica a definição e chamada de uma sub-rotina. Nele, nós definimos uma sub-rotina com o nome de “**maior**” que recebe 3 números como parâmetros e escreve na tela qual deles é o maior. Note que o programa principal usa (referencia) a sub-rotina três vezes.

```
program exemplo;

procedure maior(a, b, c: integer);
var m : integer;
begin
  m:=a;
  if b > m then m := b;
  if c > m then m := c;
  writeln(m);
end;

begin
  maior(1, 2, 3);
  maior(5, 12, 31);
  maior(949, 232, 22);
end.
```

Programa 3

Se não tivéssemos a capacidade de definir sub-rotinas, teríamos que repetir no programa principal todas as instruções que pertencem à sub-rotina, para cada vez que quiséssemos calcular o maior entre três números. Assim, ao criar um programa em uma linguagem de alto nível como Pascal, que fornece o recurso de definir sub-rotinas, é comum identificarmos instruções que se repetem, as quais podem ser isoladas para serem referenciadas pelo resto do programa.

As sub-rotinas criaram um nível intermediário entre a linguagem e os programas porque elas mudam mais rapidamente que as linguagens, mas mudam menos rapidamente que os programas. No processo de criação dos programas, os blocos de instruções que repetem muito, acabam sendo sintetizados, virando sub-rotinas. As sub-rotinas são como se fossem elementos da linguagem, com a diferença que elas são mais flexíveis. Construir programas a partir de sub-rotinas prontas não difere muito de construir programas a partir de instruções de

uma linguagem. Dessa maneira, podemos dizer que a evolução das sub-rotinas é guiada pela evolução dos programas. Um programador poderia começar a escrever um programa pelas sub-rotinas se ele soubesse, de antemão, quais seriam os blocos de instruções que mais iriam se repetir. Mas não há como ter esse conhecimento.

Com o tempo, o que acaba acontecendo é que as sub-rotinas acabam virando elementos da linguagem. Assim, a evolução das linguagens é guiada pela evolução das sub-rotinas, cuja evolução, por sua vez, é guiada pela evolução dos programas.

Mas os próprios programas também evoluem e sua evolução também é guiada. A maioria dos programas é flexível, e isso significa que diferentes usuários podem utilizar o mesmo programa de forma diferente. Para ilustrar isso, daremos exemplos em uma linguagem de programação hipotética, parecida com o português. Observe o programa abaixo.

```
Leia a  
Leia b  
Escreva a+b
```

Programa X

O programa X tem duas variáveis – **a** e **b** –. As variáveis são apenas nomes de regiões (endereços) na memória do computador onde informações são armazenadas. O programa recebe dois números e os armazena nessas variáveis. Depois ele calcula e mostra na tela a soma desses dois números. Sabemos que um programa desses não é muito útil, mas nós iremos utilizá-lo apenas para ilustração. Vamos imaginar que, ao longo do tempo, o programa X passa ser usado quase sempre com a variável **a** recebendo o valor 51. Isso cria uma forte pressão para que alguém construa o programa Y, abaixo.

```
Leia b  
Escreva b+51
```

Programa Y

O programa Y, em relação ao programa X, é menor, mais simples, utiliza menos memória e é mais fácil de ser usado. Vemos que a flexibilidade proporcionada pela variável **a** do programa X permitiu a “descoberta” do valor 51 que foi fixado no programa Y. Podemos dizer que sua presença criou um novo nível evolutivo, que pode ser visto como uma simulação que guiou a evolução do programa X para o programa Y. Imagine que tivéssemos desde o início apenas uma variável. Para descobrir o programa Y, teríamos que escrever uma

série de programas diferentes, até descobrir que 51 seria um valor estável.

```
Leia b  
Escreva b+1
```

Programa X₁

```
Leia b  
Escreva b+2
```

Programa X₂

...

```
Leia b  
Escreva b+51
```

Programa X₅₁

Repare que dessa maneira, além da evolução ser mais lenta, ela não seria guiada. Podemos dizer, portanto, que a evolução dos programas é guiada quando eles podem ser utilizados de diferentes maneiras e sua utilização guia a criação e a modificação em novas versões.

Chegamos ao final dessa discussão compreendendo como a evolução do software é guiada através da interação dos seus diferentes níveis evolutivos. Como acabamos de demonstrar, os programas em si evoluem guiados pela utilização do usuário, em um processo onde novas versões dos programas são criadas levando em consideração as “descobertas” feitas pelos usuários destes. Se os programas são construídos utilizando linguagens que permitem definir sub-rotinas, vimos que os blocos repetitivos que são “descobertos” nesses programas podem ser sintetizados nas sub-rotinas e, assim, a evolução das sub-rotinas é guiada pela evolução dos programas. As linguagens de programação, ao longo do tempo, também evoluem, e sua evolução se dá através da síntese de sub-rotinas ou de trechos de programas que foram “descobertos”. Ainda, existem diferentes níveis de linguagens de programação e a evolução das mais flexíveis guia a evolução das mais rígidas. Finalmente, podemos até considerar que o computador também evolui guiado pela evolução do software, no sentido em que novos processadores não são criados aleatoriamente, mas sua criação é guiada pelo software que “rodava” nas versões anteriores dos processadores.

É claro que a evolução das linguagens de programação não parou nas linguagens como C e Pascal. No momento da elaboração deste trabalho, a maioria dos programas é escrita em linguagens orientadas a objeto, como Java e C++, e novos tipos de linguagens estão surgindo.

Entretanto não iremos discuti-las neste trabalho, por uma questão de espaço. Além disso, independentemente da tecnologia, qualquer linguagem não deixa de ser uma máquina universal de Turing, e deixamos claro que o aspecto teleológico da evolução do software deriva da relação entre a máquina de Turing, as linguagens e mensagens, e do fato da máquina universal de Turing permitir a criação de vários níveis de software, sendo que um determinado nível simula o outro.

4. CONCLUSÃO E DESENVOLVIMENTOS FUTUROS

Este trabalho evidenciou como a evolução do software pode ocorrer através da interação entre linguagens e mensagens. Ele também tornou manifesto que a evolução do software é guiada, e que isso se dá através da interação entre vários níveis evolutivos, de flexibilidades diferentes, em um processo onde a evolução no nível mais flexível guia a evolução no nível mais rígido. Mas, como vimos, as idéias de Peirce, utilizadas para compreender o aspecto teleológico da evolução do software, se aplicam a toda forma de evolução. Assim, este trabalho buscou contribuir para que se considere com mais seriedade essas idéias. Isso é importante pois essas idéias são negligenciadas pela maioria dos cientistas contemporâneos, e poderiam fornecer muitas explicações importantes, principalmente para a teoria da evolução.

É sabido que, a despeito dos avanços da ciência, a única explicação para a evolução que tem ampla aceitação entre os cientistas, hoje em dia, é o neodarwinismo, que une as idéias originais de Darwin às novas descobertas como a genética, o DNA, dentre outras. Segundo os neodarwinistas, a evolução biológica ocorre através da sobrevivência diferenciada de entidades replicadoras autônomas – os genes. Os seres vivos são máquinas de sobrevivência construídas pelos genes, para auxiliá-los na luta pela sobrevivência (DAWKINS, 1990). O problema é que, atualmente, o neodarwinismo diz respeito apenas à evolução biológica. Embora existam grandes esforços em utilizar seus princípios para criar uma teoria mais abrangente, que também explique a evolução cultural, (DAWKINS, 1990; AUNGER, 2002), indícios apontam para o fato de que essa síntese não pode deixar de lado o signo (DEACON, 1999).

Vimos que para Peirce, não só a vida, mas toda a regularidade no universo é resultado da evolução, e o signo é o substrato comum a todos os níveis evolutivos. A semiótica e as idéias de Peirce podem, assim, ser a chave para a unificação da evolução biológica e cultural. Atualmente, isso tem sido defendido pelos proponentes da biossemiótica. Esta é uma ciência que estuda a ação do signo no reino biológico, tentando compreender o fenômeno da vida sob a luz da semiótica, estudando não somente o papel do signo nos processos de comunicação entre animais, mas também na própria evolução em si, quer seja biológica ou cultural. Hoffmeyer, um dos biossemióticos mais conhecidos, chama a atenção para o fato de que “[u]ma unificação da biologia, uma verdadeira ‘síntese moderna’, deve basear seu entendimento da evolução em uma teoria semiótica da vida” (HOFFMEYER, 1997).

Mas, assim como demonstramos neste trabalho, ao considerar mais seriamente o papel do signo na evolução, é necessário rever a idéia de que a evolução é cega. Dessa forma, um dos próximos passos para a biossemiótica, que é de fundamental importância, é fornecer uma explicação semiótica para o efeito Baldwin. Durante a pesquisa, encontramos vários trabalhos a respeito do efeito Baldwin, e também, separadamente, sobre o aspecto teleológico da evolução decorrente do pensamento de Peirce. Entretanto, não encontramos nenhum material que relacionasse o efeito Baldwin com as idéias de Peirce. Embora este trabalho tenha contribuído com essa questão, é muito importante que esta tarefa seja levada adiante pelos biossemióticos.

Tal tarefa, entretanto, pode se desdobrar ainda mais, indo além de unificar a evolução biológica e a cultural, e de explicar seu aspecto teleológico. Como vimos, o motivo pelo qual devemos considerar que a evolução é sempre teleológica é o mesmo pelo qual devemos considerar que existem infinitos níveis evolutivos. Segundo Peirce, a evolução é sempre guiada, em maior ou menor grau, o que significa que sempre existe um outro nível subjacente a qualquer nível evolutivo. Disso decorre que a evolução não pode ter começado com a vida. Deve existir um processo evolutivo subjacente ou anterior à evolução biológica.

Não é por acaso que tem se popularizado entre os cientistas, a idéia de que a seleção natural não pode ser a único mecanismo por trás da evolução. Cientistas como o biólogo Stuart Kauffman e o químico Ilya Prigogine defendem a idéia de que devem existir leis mais básicas do aumento da complexidade e de auto-organização no universo. Prigogine, que foi pioneiro no estudo de sistemas auto-organizativos, recebeu o prêmio Nobel por seu trabalho com estruturas dissipativas em sistemas termodinâmicos afastados do equilíbrio, demonstrando que esses sistemas tendem a se auto-organizar. Kauffman sustenta que sistemas dinâmicos distantes do equilíbrio tendem a apresentar auto-organização (KAUFFMAN, 1995). De acordo com ele, a auto-organização é um pré-requisito para evolução por seleção natural.

Assim, as idéias presentes no pensamento de Peirce podem fornecer o arcabouço para compreender tanto as leis mais gerais de auto-organização as quais se refere Kauffman e Prigogine, como também a evolução biológica e a cultural, todos sob a mesma luz, levando em consideração que todos os níveis fazem parte de um continuum, e que o nível mais flexível guia a evolução do nível mais rígido.

Mas este trabalho não é importante somente sob o ponto de vista da evolução. Ao estudarmos a evolução do software, estudamos também a relação entre a semiótica e a computação. Como vimos, a semiótica não é somente uma teoria dos processos

comunicativos e evolutivos, mas também uma teoria cognitiva. Cognição também é ação do signo e nós só pensamos através deles. Turing, acreditando que era possível decompor a inteligência em partes cada vez menos inteligentes, acabou concebendo uma máquina abstrata capaz de fazer algo muito parecido com aquilo que Peirce chamou de semiose. Existe, portanto, uma relação íntima entre a semiótica e a computação, que deve ser melhor explorada. Neste trabalho não pudemos levar essa relação adiante, mas tal tarefa é de fundamental importância.

Dissemos que o computador realiza quasi-semiose porque, embora ele seja capaz de gerar signos a partir de outros, seguindo regras pré-estabelecidas, ele não tem criatividade e nem capacidade de aprender com a experiência. Mas isso está diretamente relacionado com o principal problema da inteligência artificial: construir uma máquina capaz de pensar da forma como nós pensamos. Ao compreendermos melhor o aprendizado e a criatividade, seremos capazes de criar máquinas que realizam semiose genuína, e que são, nesse aspecto, mais parecidas conosco.

Uma característica da nossa forma de pensar é que nós temos sempre propósitos. Dizemos que não somos programados como um computador porque nossos objetivos são nossos, e a forma como eles serão realizados não está pré-determinada, como em um computador. Assim, nossa inteligência está intimamente relacionada com a causa final. De acordo com Peirce, inteligência é caracterizada pela antecipação de um estado futuro, assim, ação é inteligente se ela é teleológica. Entender os mecanismos da causa final e sua relação com a semiótica pode ajudar a entender os mecanismos da ação inteligente, do tipo da qual nós realizamos, e também contribuir para o campo da inteligência artificial.

Diante das perspectivas que a semiótica e as ideias de Peirce abrem nas diversas áreas do conhecimento científico, somos forçados a concluir que é de fundamental importância que seu estudo se intensifique cada vez mais. A semiótica sozinha tem um poder unificador muito grande, sendo uma teoria que pode ajudar a compreender a comunicação, a evolução, a cognição e a computação, todos sob a mesma luz. Frente ao gigantesco esforço que resta ainda ser realizado, esperamos que este trabalho tenha dado uma pequena mas importante contribuição.

5. BIBLIOGRAFIA

- AUNGER, R. **The Electric Meme: A New Theory of How We Think.** New York: Free Press, 2002.
- BALDWIN, J. M. A New Factor in Evolution. **American Naturalist**, v. 30, p. 441-451, p. 536-553, 1896.
- BERGSON, H. **A Evolução Criadora.** São Paulo: Martin Fontes, 2005.
- CALVIN, W. H. The Brain as a Darwin Machine. **Nature**, v. 330, p. 33-34, 5 nov. 1987.
- DAWKINS, R. **The Extended Phenotype.** New York: Oxford University Press, 1999.
- DAWKINS, R. **The Selfish Gene.** 2nd ed. New York: Oxford University Press, 1990.
- DEACON, T. W. Memes as Signs. **The Semiotic Review of Books**, v. 10(3), p. 1-3, 1999.
- DEACON, T. W. **The Symbolic Species: The Co-evolution of Language and Brain.** New York: Norton, 1998.
- DENNETT, D. C. **A Perigosa Idéia de Darwin.** Tradução Talita M. Rodrigues. Rio de Janeiro: Rocco, 1998.
- DENNETT, D. C. **Consciousness Explained.** [S.l.]: Back Bay Books, 1992.
- DENNETT, D. C. **Kinds of Minds.** New York: Basic Books, 1997.
- DENNETT, D. C. **Freedom Evolves.** New York: Penguin Books, 2004.
- HINTON, G. E.; NOWLAN, S. J. How Learning Can Guide Evolution. **Complex Systems**, v. 1, p. 495-502, 1987.
- HOFFMEYER, J. Biosemiotics: Towards a New Synthesis in Biology. **European Journal for Semiotic Studies**, v. 9, n. 2, p. 355-376, 1997.
- HOFFMEYER, J. Code Duality Revisited. **SEED**, v. 2 (1), p. 97-116, 2002. Disponível em: <<http://www.library.utoronto.ca/see/SEED/Vol2-1/Hoffmeyer/Hoffmeyer.htm>>
- HOFFMEYER, J. **Signs of Meaning in the Universe.** Tradução Barbara J. Haveland. Indianapolis: Indiana University Press, 1996.
- HOFFMEYER, J.; EMMECHE, C. Code-Duality and the Semiotics of Nature. In: **On Semiotic Modeling.** New Yourk: Mouton de Gruyter, p. 117-166, 1991.
- HOFSTADTER, D. R. **Gödel, Escher, Bach: an Ethernal Golden Braid.** London: Penguin Books, 1980.
- HULSWIT, M. **From Cause to Causation: A Peircean Perspective.** Dordrecht: Kluwer Academic Publishers, 2002.
- IBRI, I. A. **Kósmos Noétós: A Arquitetura Metafísica de Charles S. Peirce.** São Paulo: Perspectiva, 1992.

- KAUFFMAN, S. A. **At Home in the Universe: The Search for the Laws of Self-Organization and Complexity**. New York: Oxford University Press, 1995.
- LAPLACE, P. S. **A Philosophical Essay on Probabilities**. Tradução F. W. Truscot e F. L. Emory. New York: Dover, 1814.
- LEWONTIN, R. C. The Dream of the Human Genome. **New York Review**, p. 31-40, 28 mai. 1992.
- MITCHELL, M. **An Introduction to Genetic Algorithms**. Massachusetts: MIT Press, 1998.
- NÖTH, W. Semiotic Machines. **SEED**, v. 3, p. 81-99, 2003. Disponível em: <<http://www.library.utoronto.ca/see/SEED/Vol3-3/Winfried.htm>>
- PAPE, H. Final Causality in Peirce's Semiotics and His Classification of the Sciences. **Transactions of the Charles S. Peirce Society**, v. 29, n. 4, p. 581-608, 1993.
- PARKER, K. A. **The Continuity of Peirce's Thought**. Nashville: Vanderbuilt University Press, 1963.
- PEIRCE, C. S. **Collected Papers of Charles Sanders Peirce**. C. Hartshorne e P Weiss. eds. (v. 1-6) e A. W. Burks. ed. (v. 7-8). Cambrige: Harvard University Press. (aqui referidos como CP; os números das citações referem-se respectivamente aos volumes e parágrafos), 1931-58.
- PIERCE, C. S. **Manuscritos Inéditos**. (referidos como MS; o número em seguida à sigla refere-se à numeração do manuscrito, segundo a paginação estabelecida pelo Institute for Studies in Pragmaticism, Lubbock, Texas), s.d.
- PIERCE, C. S. **Semiótica**. 3. ed. São Paulo: Perspectiva, 2003.
- PEIRCE, C. S. **Writings of Charles Sanders Peirce: A Chronological Edition**. Max Fish et al. eds. Bloomington: Indiana University Press, v. 1-5 (aqui referidos por W seguido do volume), 1982.
- PEIRCE, C. S. **Reasoning and the Logic of Things**. Kenneth Laine Ketner. ed. Cambridge: Harvard University Press (aqui referido por RLT), 1992.
- PEIRCE, C. S. **The New Elements of Mathematics**. Caroline Eisele. The Hague: Mouton Publishers, v. 1-4 (aqui referidos por NEM seguido do volume), 1976.
- PEIRCE, C. S. **The Essential Peirce: Selected Philosophical Writings**. Nathan Houser e Christian J. W. Kloesel. eds. (v. 1) e Peirce Edition Project. eds. (v.2). Bloomington: Indiana University Press, (aqui referidos por EP seguido do volume), 1992 e 1998.
- PINKER, S. **The language Instinct**. London: Penguin Books, 1995.
- PINKER, S.; BLOOM, P. Natural language and natural selection. **Behavioral and Brain Sciences**, Cambridge, v. 13, n.4, p. 707-784, 1990.

- REYNOLDS, A. **Peirce's Scientific Metaphysics**. Nashville: Vanderbuilt University Press, 1966.
- SANTAELLA, L. **A Teoria Geral dos Signos**. São Paulo: Pioneira, 2000.
- SANTAELLA, L. What is a Symbol. **SEED**, v. 3, p. 54-60, 2003. Disponível em :
[<http://www.library.utoronto.ca/see/SEED/Vol3-3/Santaella.htm>](http://www.library.utoronto.ca/see/SEED/Vol3-3/Santaella.htm)
- SHORT, T. L. Semeiosis and Intentionality. **Transactions of the Charles S. Peirce Society**, v.17, n. 3, p. 197-223, 1981a.
- SHORT, T. L. Peirce's Concept of Final Causation. **Transactions of the Charles S. Peirce Society**, v.17, n. 4, p. 369-382, 1981b.
- SIPSER, M. **Introduction to the Theory of Computation**. Boston: PWS Publishing Company, 1997.
- SAVAGE-RUMBAUGH, E. S., RUMBAUGH D. M., BOYSEN S. Symbolization, Language and Chimpanzees: A Theoretical Reevaluation Based on Initial Language Acquisition Processes in Four Young Pan Troglodytes. **Brain and Language**, v. 6, p. 265, 1978.
- SAVAGE-RUMBAUGH, E. S., RUMBAUGH, D. M., SMITH , S. T., LAWSON, J. Reference: The linguistic essential. **Science**, v. 210, p. 922 – 925, 1980.
- TURING, A. M. Proposal for Development in the Mathematics Division of an Automatic Computing Engine (ACE). In: Carpenter, B.E., Doran, R.W. (eds). **A.M. Turing's ACE Report of 1946 and Other Papers**. Cambridge: MIT Press, 1986a.
- TURING, A.M. Lecture to the London Mathematical Society on 20 February 1947. In: Carpenter, B.E., Doran, R.W. (eds). **A.M. Turing's ACE Report of 1946 and Other Papers**. Cambridge: MIT Press, 1986b.
- TURING, A. M. Computing Machinery and Intelligence. **Mind**, n. 49, p. 433-460, 1950