

Para além da manipulação simbólica: o algoritmo computacional como estrutura em individuação

Algoritmos computacionais são frequentemente descritos como sequências de instruções para realização de alguma atividade. Na literatura da área, é possível encontrar definições que especificam cada vez mais a natureza da estrutura algorítmica, evidenciando variáveis como tempo, espaço, especificidade e eficiência. Dada a complexidade dos usos e relações elaboradas pelo algoritmo computacional no contemporâneo, pesquisas e reflexões que se situam fora da área de ciência da computação podem propor (ou reivindicar) novos caminhos de significação e compreensão para tal. Na arte computacional por exemplo, inquietações sobre autoria, autorreferência ou intencionalidade podem demandar outras perspectivas para definições e conceptualizações já formalizadas. Nesta perspectiva, propomos refletir a noção de algoritmo a partir do processo de individuação em Gilbert Simondon.

Um algoritmo computacional é um procedimento mecânico (mera manipulação simbólica) que, aplicado a certo *input*, elabora um *output* ou resultado num tempo finito e em número finito de passos (Abbagnano, 2012, p. 27). Para Vardi (2012), um algoritmo é uma máquina de estado abstrata (Gurevich, 2012) e um recursor (Moschovakis, 2002), mas não descrevem completamente por si só o que é um algoritmo. É possível compreender uma máquina de estado abstrata como uma versão reducionista de um sistema. Existem os estados com suas transições e as ações que respondem às condições requisitadas. É um conjunto finito de elementos. Cada estado pode ser lido como uma condição específica, e as transições são mudanças de determinada condição para outras, ao executar ações associadas a essa transição, quando solicitado. Cada passo que essa máquina realiza pode apenas causar uma mudança limitada e local nesse estado (Vardi, 2012). Hui (2019), por sua vez, compreende o algoritmo computacional como estrutura recursiva, onde uma função chama a si mesma em cada iteração até que um estado de parada seja atingido, que é um objetivo

executável ou uma de ser incomputável (Hui, 2019, p. 113). No entanto, como comenta Hill (2015) tanto para Gurevich quanto para Moschovakis, existe algo no algoritmo para além da manipulação simbólica.

Hill (2015) propõe uma definição com mais elementos: um algoritmo binário é uma estrutura de controle finita, abstrata, efetiva e composta, imperativamente dada, cumprindo um determinado propósito sob determinadas disposições. Contudo, reconhece uma esfera semântica na estrutura algorítmica, ao contrário de grande parte dos cientistas da computação. Para ele, “o que significa” é entendido como “o que faz”. Smith (ano) defende que símbolos contêm semântica: “isto é, ter uma interpretação real, ser interpretável, seja o que for – para que haja algo substantivo para que sua manipulação formal prossiga independentemente. Sem um caráter semântico para ser mantido crucialmente nos bastidores, a construção da manipulação de símbolos formais entraria em colapso no vazio”, o que para Hui (2016) sugere uma suposta “cognição” das máquinas ou intencionalidade.

MacCormick (2012) e Rappaport (2012) enfatizam a composição visual proposta pelo algoritmo computacional como qualidade fundamental para sua compreensão: o leitor não pode chegar a conclusão que um algoritmo é constituído por sequências de declarações bem recuadas iniciadas com verbos, como em um texto comum. A estrutura do algoritmo precisa ser reconhecida e explicitada (MacCormick, 2012).

Um algoritmo (para o executor E atingir a meta G) é:

Um procedimento (ou método) - ou seja, um conjunto finito (ou sequência) de declarações (ou

regras ou instruções) - de modo que cada declaração seja:

- Composto por um número finito de símbolos (ou marcas) de um alfabeto finito
- E inequívoco para E - ou seja,
- E sabe como fazer
- E pode fazer isso
- pode ser feito em um tempo finito
- e, depois de fazer isso, E sabe o que fazer a seguir—

- E o procedimento leva uma quantidade finita de tempo, ou seja, para,
- E termina com G realizado.

(Rappaport, 2012 *Apud* Hill, 2015)

A partir das características apresentadas pelas definições acima, iremos em direção à filosofia de Gilbert Simondon. Para o autor, a individuação é uma transdução em andamento e como sintetiza Hui (2016) está relacionada a um processo de resolução de tensões para chegar a metaestabilidade, resultando em mudanças de fase:

Por transdução, entende-se uma operação - física, biológica, mental, social - pela qual uma atividade se propaga de próximo em próximo no interior de um domínio. A transdução funda essa propagação sobre uma estruturação do domínio operada de lugar em lugar: cada região de estrutura constituída serve de princípio de constituição para a região seguinte, de modo que uma modificação se estende progressivamente, ao mesmo tempo que essa operação estruturante (SIMONDON, 2005, p. 29).

Um algoritmo computacional é, antes de tudo, uma estrutura transdutiva. A palavra transdução deriva do latim *trans*, que significa através de, e *ducere*, que significa conduzir (Hui, 2016). É em si mesma, um caminho de transformação, reestruturação e amplificação de domínios. Portanto, é também, uma estrutura relacional, tanto internamente quanto externamente. O resultado de cada função da sequência de instruções serve de princípio de constituição para a função seguinte, propagando uma modificação progressiva ao mesmo tempo que operação estruturante, nos termos de Simondon (2020). A ordem criada pelo algoritmo, onde cada função é executada após a outra, também pode ser observada como ordem transdutiva, uma vez que é operada de lugar em lugar, conservando e transmitindo a modificação, construindo sua dinâmica relacional interior.

Os dados de entrada e saída, por sua vez, constroem em primeiro momento sua dinâmica relacional exterior. Mas não só isso. O algoritmo nos dias de hoje não está isolado, mas parte das redes computacionais, em associação com diversos outros algoritmos e com os humanos. O resultado de sua dinâmica relacional interior, ou seja, seus dados de saída servem de princípio constituintes para outros algoritmos, dando continuidade ao processo transdutivo. Como Hui (2016) explicita, as redes são atualizadas de acordo com certos parâmetros e algoritmos. Essas atualizações influenciam diretamente no comportamento dos usuários, e fazem parte do que Simondon chamou transdução psíquica, que acontece com os seres vivos. A relação dos humanos com os objetos técnicos acontece no nível das funções de transdução (Simondon, 2020, p.156) Por outro lado, também faz parte de sua dinâmica relacional exterior e dessa vez, no tempo histórico, as mudanças que se sucederam entre as lógicas algorítmicas, colocando o algoritmo como fenômeno cultural.

Enquanto estrutura transdutiva, o algoritmo é também auto-referente. A cada reestruturação em um novo domínio a transdução conserva informação do domínio anterior, ou seja, o resultado de cada operação faz referência a anterior, de modo que, toda a estrutura fale de si mesma. Como comentado anteriormente, Hui (2016) observa os algoritmos a partir da recursividade, e estende para além da prova matemática para aplicações mais amplas, como pensamento algorítmico. Para o autor, a recursão é uma gênese da capacidade de autoposicionamento e auto-realização do algoritmo. Desse modo, é estrutura dotada de intencionalidade. Em primeiro momento, pelo próprio objetivo para qual o algoritmo foi escrito, todo algoritmo existe para um fim efetivo, mas para além disso, intencionalidade de descoberta de dimensões, segundo Simondon, dimensões que o sistema faz comunicar.

Tomar o algoritmo computacional como estrutura transdutiva primeiramente, é portanto, tomá-lo como ser em individuação. Observá-lo como objeto intencional e relacional com o mundo nos permite olhar para suas características gerais de modo mais crítico, mas também inventivo. A noção de algoritmo como uma sequência de instruções

para realizar uma atividade não reflete a complexidade em que o objeto está situado nas redes computacionais, pois, não exprime de maneira clara sua característica relacional. Ainda que o algoritmo seja uma estrutura de controle e imperativamente dada, como comenta Hill (2012), à luz da transdução ambas as características denotam apenas alguns dos modos de relação possíveis, permitindo que outras relações e reestruturações sejam imaginadas e implementadas.

Nesse sentido, sob a ótica da individuação, propomos a noção de algoritmo computacional como estrutura transdutiva, que se organiza através de instruções lógicas finitas, mas que não se extingue no *output*¹.

Palavras-chave

Algoritmo Computacional; Individuação; Transdução, Gilbert Simondon.

Referências:

ABBAGNANO, N.; Trad. Ivone Castilho Benedetti. **Dicionário de filosofia**. 6. ed. São Paulo: Martins Fontes, 2012.

CANTWELL SMITH, B. **On the Origin of Objects**. Cambridge, Mass.: MIT Press, 1996.

GUREVICH, Y.; CALUDE, C. S. A Dialogue with Yuri Gurevich about Mathematics.

Computer Science and Life. Bulletin of Euro. Assoc. for Theor. Computer Science, p. 1-13, Jun. 2012

HILL, R. K. What an Algorithm Is. **Philosophy & Technology**, v. 29, n. 1, p. 35–59, 11 jan. 2015.

HUI, Y. **Recursivity and contingency**. London, United Kingdom ; Lanham, Maryland:

¹ Este artigo é um recorte da pesquisa de doutorado em andamento **** desenvolvida no Programa de Pós-Graduação em Artes da Escola de Belas Artes da Universidade Federal de Minas Gerais.

Rowman & Littlefield International, Ltd, 2019.

MACCORMICK, J. **Nine algorithms that changed the future : the ingenious ideas that drive today's computers.** Princeton, N.J.: Princeton University Press, 2012.

MOSCHOVAKIS, Y. N. **On founding the theory of algorithms.** Preprint. Fev, 2002.
Disponível em: <http://www.math.ucla.edu/ynm/papers/foundalg.pdf>

SIMONDON, G. Trad. Vera Ribeiro. **Do modo de existência dos objetos técnicos.** 1. ed.
Rio de Janeiro: Contraponto, 2020.

SIMONDON, G. Trad. Luís Eduardo Ponciano Aragon e Guilherme Ivo. **A individuação à luz das noções de forma e de informação.** 1. ed. São Paulo: Editora 34, 2020.

VARDI, M. Y. What is an algorithm? **Communications of the ACM**, v. 55, n. 3, p. 5–5, mar. 2012.