



<http://dx.doi.org/10.23925/2237-9657.2020.v9i3p106-119>

O uso do comando sequência na Elaboração de Simuladores com o *software* GeoGebra

Sequence command use in Simulators Elaboration with GeoGebra software

LUIS ANDRÉS CASTILLO B.¹

RAFAEL ENRIQUE GUTIÉRREZ ARAUJO²

IVONNE C. SÁNCHEZ S.³

RESUMO

Visando reportar as vantagens e potencialidades de funcionalidades menos conhecidas do software GeoGebra para realizar construções geométricas, objetivou-se apresentar o uso do comando sequência em experiências de Elaboração de Simuladores com o GeoGebra (ESG) para a obtenção de desenhos dinâmicos. Para tal, reportam-se algumas experiências dessa atividade em que se tentava representar peças de mecanismos nas janelas 2D e 3D do software. As experiências mostram a eficiência do comando sequência para resolver, em menor tempo, tarefas de construção geométrica, cuja resolução, usando as ferramentas tradicionais do software, precisaria de maior tempo e esforço. Nesse sentido, considera-se que a inclusão do comando sequência no repertório de ferramentas dos participantes da ESG pode melhorar suas técnicas de construção e, portanto, os desenhos dinâmicos que tentam obter.

Palavras-chave: Tecnologias digitais; Geometria dinâmica; Geometria 3D.

ABSTRACT

In order to report the advantages and potential of other lesser known functionalities of GeoGebra software to realize geometric constructions, the aim this paper is to present the use of the sequence command in experiences of Simulators Elaboration with GeoGebra software (ESG) to obtain dynamic draws. For this, some experiences of this activity in which was tried to represent parts of mechanisms in the 2D and 3D windows of the software are reported. The experiences show the efficiency of sequence command to solve geometric construction tasks in a shorter time, which resolution would need more time and effort using traditional software construction tools. In this sense, it is considered that the inclusion of the sequence command in the tools repertoire of ESG participants can improve their construction techniques and, therefore, the dynamic draws they try to obtain.

Keywords: Digital technologies; Dynamic geometry; 3D geometry.

¹ Asociación Aprender en Red – luiscastleb@gmail.com – <https://orcid.org/0000-0002-5174-9148>

² Asociación Aprender en Red – rafael.gutierrez0593@gmail.com – <https://orcid.org/0000-0002-4003-8324>

³ Asociación Aprender en Red – ivonne.s.1812@gmail.com – <https://orcid.org/0000-0002-2485-1059>

Introdução

Ao longo das últimas décadas, as Tecnologias Digitais (TD) enquanto recursos para o ensino e a aprendizagem de matemática têm sido objeto de estudo por diversos pesquisadores do campo da Educação Matemática, tornando-se mais um elemento que compõe a atividade matemática da aula (ROJANO, 2014). A importância do uso das TD no ensino e aprendizagem de matemática tem sido reconhecida e adotada por diversos documentos curriculares, entre eles, os *Princípios e Normas para a Matemática Escolar* do Conselho Nacional de Professores de Matemática (NCTM). Nesse documento, considera-se a tecnologia como um dos princípios para a matemática escolar do século XXI ao estabelecer as possibilidades que ela oferece no desenvolvimento de práticas diversas, entre elas, as investigações matemáticas. Nessa linha, o NCTM (2000) profere que:

As tecnologias eletrônicas (calculadoras e computadores) são ferramentas essenciais para ensinar, aprender e fazer matemática [...]. Elas podem apoiar a investigação dos estudantes em todas as áreas da matemática, incluída a geometria [...]. A tecnologia enriquece o alcance e a qualidade das investigações ao proporcionar um meio para ver ideias matemáticas desde múltiplas perspectivas. A aprendizagem dos alunos é assistida pela retroalimentação que a tecnologia pode proporcionar [...]. A tecnologia também proporciona um enfoque à medida que os estudantes discutem entre eles e com seu professor os objetos na tela e os efeitos das diversas transformações dinâmicas que a tecnologia permite. (p. 24-25, tradução nossa).

Dadas as possibilidades que oferecem no desenvolvimento da atividade matemática da aula, as TD são consideradas como ferramentas que transformam as práticas matemáticas de professores e alunos, sempre que não sejam subutilizadas (BORBA; VILLARREAL, 2005; HOYLES, 2018). No caso da geometria dinâmica, o impacto da incorporação das TD no estudo da geometria pode-se observar, por exemplo, nas relações que se estabelecem entre um *desenho* e uma *construção* realizada na interface de um *Software* de Geometria Dinâmica (SGD). Com respeito a isso, Borba, Scucuglia e Gadanidis (2018) salientam que:

Distinções entre *desenho* e *construção* não faziam sentido quando construíamos objetos geométricos com lápis, papel e outras tecnologias, como régua e compasso, mas essa distinção começou a ser significativa com o uso de softwares de GD [Geometria Dinâmica] [...]. Em uma construção, a figura sempre preserva suas propriedades fundamentais quando um dos elementos “móveis” que a compõem é arrastado. Se arrastarmos uma figura e ela não mantiver suas propriedades fundamentais, a figura apenas é um desenho (p. 27-28, ênfase no original).

Nessa temática, Laborde (1997) refere-se à *construção* geométrica realizada na interface de um SGD como *desenho dinâmico* e, ao problema da sua produção, como *tarefa de construção*, que é explorada em contextos de ensino e aprendizagem com propósitos diversos, da mobilização de saberes geométricos à criação de cenários para o desenvolvimento de investigações matemáticas (ALQAHTANI; POWELL, 2016; OLIVERO; ROBUTTI, 2007; SANTOS-TRIGO; REYES-RODRÍGUEZ, 2016; TROCKI; HOLLEBRANDS, 2018). No primeiro caso, as diferentes ferramentas de construção do SGD utilizado possuem um papel principal na resolução de tarefas de construção, na medida em que elas são portadoras de conteúdos conceituais dos objetos geométricos que se tentam representar, susceptíveis de mobilização durante a atividade de resolução da tarefa.

Porém, o uso dessas ferramentas “tradicionais” nem sempre constitui a melhor opção para alunos e professores que se engajam na resolução de tarefas de construção geométrica particulares, como aquelas que são resolvidas para elaborar *simuladores computacionais* de fenômenos da cotidianidade. Nesse cenário, SGD como o GeoGebra possuem diversas funcionalidades dinâmicas que permitem ao usuário realizar construções geométricas, consideradas complexas, com maior eficiência e precisão do que as ferramentas de construção e medida tradicionais. Uma dessas funcionalidades é o *comando sequência* do GeoGebra, o qual tem sido pouco explorado como muitas outras funcionalidades desse *software*. Com o intuito de reportar as vantagens e potencialidades de funcionalidades menos conhecidas do GeoGebra para realizar construções geométricas, neste artigo objetivou-se apresentar o uso do comando sequência em experiências de elaboração de simuladores computacionais com o *software* GeoGebra.

1. O contexto do estudo

A partir do ano 2013 foi desenvolvido em diferentes instituições escolares públicas do oeste da Venezuela o projeto educativo *Club GeoGebra*, que consistia na criação de grupos formados por alunos (de níveis escolares equivalentes aos Anos Finais do Ensino Fundamental e Ensino Médio do Brasil) e professores que ensinam matemática, com o interesse de elaborar simuladores com o *software* GeoGebra. No geral, a *Elaboração de Simuladores com o software GeoGebra* (ESG) é uma atividade cujo objetivo é obter um modelo digital, na interface do *software*, que represente as formas e movimentos de um fenômeno determinado da realidade. Na Figura 1 ilustra-se um exemplo do simulador de um motor V6, elaborado na interface do GeoGebra.

Para fazer operativa à ESG, os alunos se envolvem na realização de projetos de simulação que se desenvolvem em quatro fases: (i) seleção do fenômeno, (ii)

elaboração do simulador, (iii) sistematização da experiência e (iv) socialização da experiência. À medida que se engajam no desenvolvimento dos seus projetos, os alunos aprendem a usar as diferentes ferramentas de construção e medida do *software* para elaborar seus simuladores.

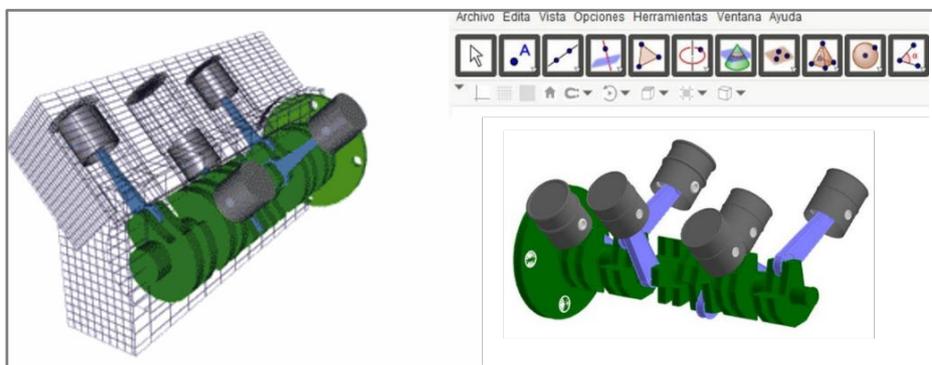


FIGURA 1: Simulador de um motor V6 elaborado com o *software* GeoGebra

FONTE: Adaptada de <https://goo.gl/2QP62>. Acesso em: 10 ago. 2019

Um aspecto que chamou a atenção, durante o desenvolvimento das experiências de trabalho com alunos e professores no contexto da ESG, foi a tendência deles de utilizarem, com maior ênfase, as ferramentas de construção e medida do *software* para resolver tarefas de construção. Um fato que lhes trazia, em ocasiões, complicações para finalizar a atividade com sucesso. Com o propósito de mostrar a possibilidade de ir mais além do uso dessas ferramentas “tradicionais”, descrevem-se as experiências de dois alunos dos Anos Finais do Ensino Fundamental e Ensino Médio, que usaram o comando sequência do GeoGebra para representar peças de mecanismos da cotidianidade na janela 3D do *software*.

2. Elaboração de Simuladores com o GeoGebra

No geral, pode entender-se um simulador computacional como um modelo digital de um fenômeno real ou hipotético, utilizado para compreender e prever o comportamento desse fenômeno (CLARK; NELSON; SENGUPTA; D’ ANGELO, 2009). Esses autores salientam o fato desses simuladores serem amplamente utilizados em diversos campos científicos, como engenharia, medicina, economia, matemática e ciências naturais. No que diz respeito ao campo educativo, esses simuladores são utilizados como artefatos que servem de ponte para o estudo e análise de fenômenos próprios da matemática e das ciências naturais, procurando que os alunos desenvolvam experiências científicas significativas. Ao interagirem com o simulador, os alunos têm a possibilidade de manipular e/ou modificar as

variáveis que definem o fenômeno modelado, podendo analisar e compreender o seu comportamento à medida que o tempo transcorre.

Considerando as ideias acima, a ESG é concebida como uma atividade que consiste em obter um simulador computacional na interface do GeoGebra, por meio do uso das ferramentas (de construção e medida) e funcionalidades dinâmicas do *software*. O processo de obtenção desse simulador se organiza em torno a um conjunto de *tarefas de simulação* que são resolvidas em sequência, na medida em que cada uma dessas tarefas se relaciona com a representação de uma das partes que compõem o fenômeno a modelar (RUBIO; PRIETO; ORTIZ, 2016). Nesse sentido, a resolução de uma tarefa de simulação pressupõe: (i) elaborar um esboço daquela parte do fenômeno que se deseja representar, (ii) identificar as formas e movimentos da parte esboçada matematicamente, e (iii) construir os desenhos dinâmicos que permitam modelar essas formas e movimentos no *software*.

Dependendo do tipo de objetos matemáticos identificados na etapa (ii) acima, os desenhos dinâmicos são elaborados na janela 2D ou 3D do GeoGebra. Logo, ora na janela 2D ora na 3D, a complexidade de cada objeto matemático torna necessário considerar as diferentes ferramentas e funcionalidades do *software* que permitam obter, com consistência e eficácia, os desenhos dinâmicos que modelam o fenômeno de estudo. Dentre as funcionalidades dinâmicas que o *software* GeoGebra dispõe para otimizar a elaboração de um simulador, ressaltam os *comandos* que permitem tanto construir objetos nas diferentes aparências do programa quanto modificar propriedades de outros construídos previamente (CASTILLO; PRIETO, 2018).

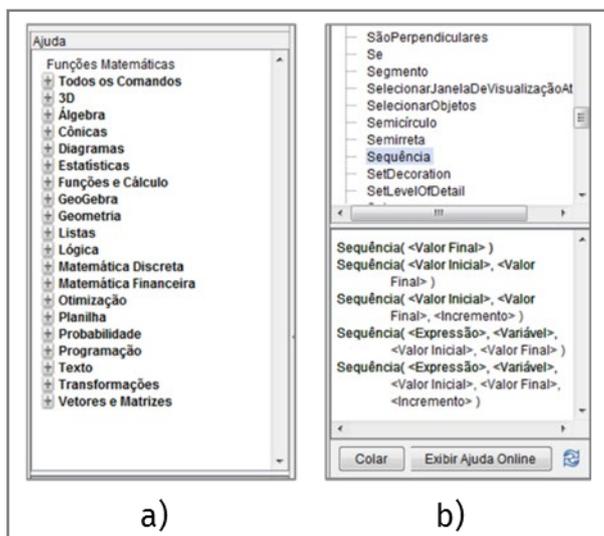


FIGURA 2: Painel de ajuda de comandos do *software* GeoGebra

FONTE: Elaborada pelos autores

Cada comando possui uma sintaxe predefinida pelo *software* que informa sobre cada parâmetro que é necessário para executá-lo. Logo, para executar um comando no GeoGebra é suficiente ingressar os parâmetros associados à sua sintaxe, por meio do *Campo de Entrada*. Vale a pena ressaltar o fato de o *software* dispor de um painel de ajuda, em que se ilustram todos os comandos disponíveis, organizados por categorias (Figura 2a). Quando um dos comandos é selecionado, a sintaxe correspondente é ilustrada na zona inferior do painel (Figura 2b).

Um dos comandos que dispõe o *software* GeoGebra é a *sequência*, que permite gerar listas de objetos que se relacionam por meio de uma expressão ou lei simbólica dada pelo comando. Nas seções a seguir, serão descritas as experiências de dois alunos que, em situação de ESG, utilizaram o comando sequência para resolver uma tarefa de construção geométrica.

3. A experiência de Luis Ángel

A primeira experiência descrita neste artigo reporta as atividades desenvolvidas por um aluno do 6º ano do Ensino Fundamental e um professor de matemática que utilizavam o *Club GeoGebra* de uma escola pública, localizada em uma cidade do Estado Zulia (Venezuela), durante o período escolar 2016–2017. Especificamente, reporta-se o trabalho desenvolvido por Luis Ángel, um aluno que, com a orientação do seu professor, decidiu realizar um projeto de simulação que tinha por objetivo representar um motor de dois tempos (Figura 3a) na janela 3D do *software* GeoGebra.

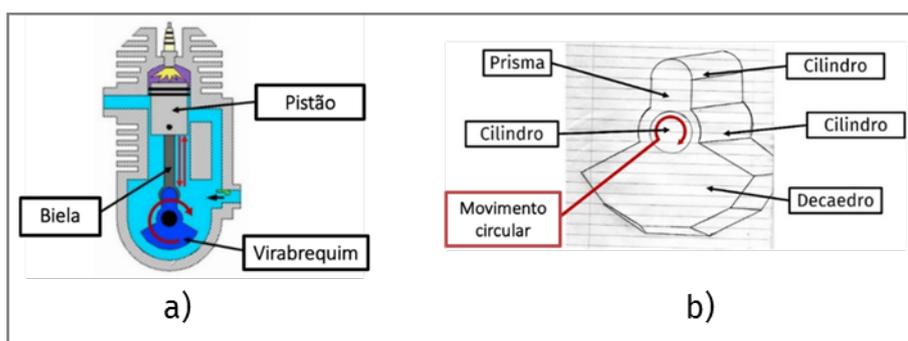


FIGURA 3: Fenômeno da simulação do projeto de Luis Ángel e esboço do virabrequim

FONTE: Adaptada de <http://bit.ly/2kehEV0>. Acesso em: 10 ago. 2019

O aluno decidiu começar a elaboração do seu simulador pela representação do virabrequim do motor. Logo, a primeira tarefa de simulação que ele resolveu foi representar o virabrequim do motor de dois tempos na janela 3D do *software* GeoGebra. Para iniciar a resolução dessa tarefa, Luis Ángel elaborou um esboço do virabrequim para reconhecer aquelas formas e movimentos que lhe permitissem

representar essa peça do motor no *software*. Após tal realização, o aluno identificou no esboço um prisma, três cilindros, um decaedro (irregular) e um movimento circular, como os objetos geométricos que melhor poderiam modelar as formas e movimentos do virabrequim do motor de dois tempos (Figura 3b).

Representar no GeoGebra cada objeto geométrico identificado implicou para Luis Ángel resolver uma tarefa de construção. Em suma, o aluno resolveu um total de seis tarefas de construção, das quais interessa reportar a terceira, segundo a ordem de resolução: construir um decaedro a partir de um ponto interior. Esse ponto interior, nomeado C' , refere-se àquele que foi representado na primeira tarefa de construção e que reproduz o movimento circular do virabrequim.

Para resolver essa tarefa, o aluno devia determinar os 16 vértices do decaedro, oito da face da frente e oito da face traseira. Como resultado de resolver a primeira tarefa de construção, o aluno obteve um cilindro cujos centros das suas faces são os pontos A (centro do sistema de coordenadas) e B (ponto inserido no eixo x). Considerando esses objetos construídos, Luis Ángel determinou os primeiros vértices da face da frente do decaedro, nomeados D e E , interceptando a reta $\overrightarrow{AC'}$ com a borda da face do cilindro, cujo centro é o ponto A (Figura 4a). O terceiro vértice, nomeado F' , foi determinado após: (i) interceptar uma circunferência (centrada em D , de raio um valor estimado e com direção o eixo y) com a reta $\overrightarrow{AC'}$, e (ii) rotacionar o ponto F , obtido da intercepção anterior, em torno do eixo y com um valor angular de 15° em sentido anti-horário (Figura 4b).

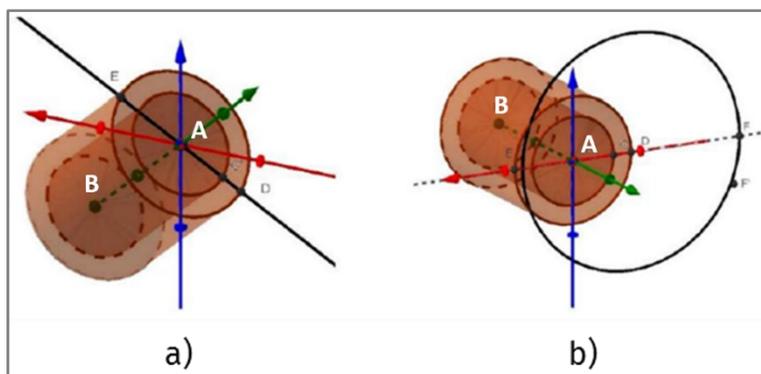


FIGURA 4: Localização dos três primeiros vértices do decaedro

FONTE: Elaborada pelos autores

Para determinar os cinco vértices restantes da face da frente do decaedro, Luis Ángel percebeu que podia aplicar a mesma transformação geométrica (rotação), mudando o valor da medida angular para cada caso. Porém, o próprio aluno reconheceu que esse procedimento não era o mais eficaz. Assim, optou por utilizar uma sequência para reduzir o número de passos do seu processo de construção. Desse modo, o aluno utilizou uma sequência cuja sintaxe é a seguinte:

Sequência[< Expressão >, < Variável >, < Valor inicial >, < Valor final >, < Incremento >]

Na sintaxe acima, a *expressão* define o tipo de elemento que será gerado na sequência. No caso dele, Luis Ángel utilizou a rotação como a expressão da sequência, a qual foi aplicada ao ponto F' em torno do eixo y com um valor angular i (*variável* da sequência). O *valor inicial* e *valor final* dizem respeito do primeiro e último valor que toma a variável i na sequência, respectivamente. No caso do aluno, esses valores foram de 30° e 170° , respectivamente. Por fim, o *incremento* refere-se ao incremento que terá a variável no desenvolvimento da sequência, o qual foi de 30° . De acordo com os valores citados, a sequência foi executada usando a expressão *Sequência* [*Girar*[$F', i, \text{eixo}Y$], $i, 30^\circ, 170^\circ, 30^\circ$], cuja aplicação no *Campo de Entrada* do *software* gerou os cinco vértices restantes da face da frente do decaedro (Figura 5).

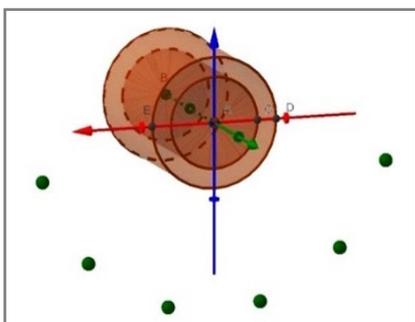


FIGURA 5: Obtenção dos cinco vértices da face da frente do decaedro usando uma sequência
FONTE: Elaborada pelos autores

Obtidos os vértices anteriores, o aluno usou o comando *polígono* e, como parâmetro, usou outro comando chamado *elemento*, ou seja, Luis Ángel usou e aplicou no *Campo de Entrada* do *software* um comando dentro do outro, gerando assim as faces da frente do decaedro. Posteriormente, o aluno realizou uma translação à face da frente para obter a face traseira do decaedro (Figura 6a).

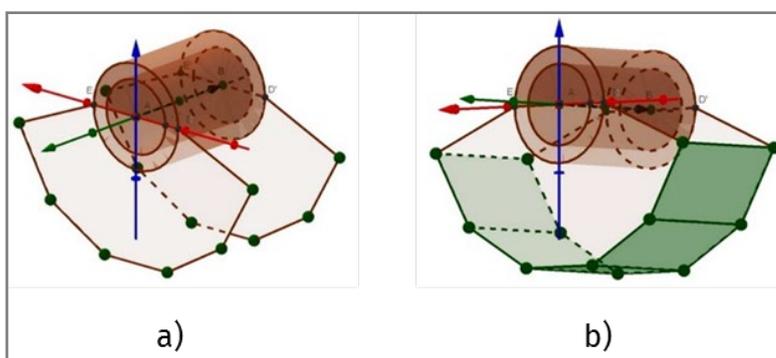


FIGURA 6: Construção das faces inferiores do decaedro usando o comando sequência
FONTE: Elaborada pelos autores

Visando evitar o uso reiterado dos comandos *polígono* e *elemento lista*, da mesma forma em que se evitou o uso das ferramentas de construção do *software* no início da tarefa, o aluno utilizou novamente o comando sequência. Isso aconteceu no momento em que ele procurava determinar as faces inferiores do decaedro. Para tal, Luis Ángel executou a seguinte sequência:

$$\text{Sequência}[\text{Polígono}[\text{Elemento}[\text{lista1}, 1 + n], \text{Elemento}[\text{lista1}, 2 + n], \text{Elemento}[\text{lista1}', 2 + n], \text{Elemento}[\text{lista1}', 1 + n]], n, 0, 5].$$

Após a aplicação dessa sequência, o *software* cria uma lista de cinco polígonos (fases inferiores do decaedro) cujos vértices são as listas de pontos obtidas anteriormente, por meio das primeiras sequências aplicadas (Figura 6b).

4. A experiência de Emanuel

A segunda experiência descrita reporta o trabalho realizado por Emanuel, um aluno do último ano do Ensino Médio que, em conjunto com seu professor orientador do *Clube GeoGebra*, decidiu desenvolver um projeto de simulação que teve por objetivo representar um motor V6 (Figura 7a) na janela 3D do *software* GeoGebra. Similar a Luis Ángel, Emanuel decidiu começar a elaboração do seu simulador pela representação do virabrequim do motor. Assim, a primeira tarefa de simulação que o aluno resolveu foi representar o virabrequim do motor V6 na janela 3D do GeoGebra. Após desenhar o esboço de uma primeira parte do virabrequim, Emanuel identificou dois tridecaedros (irregulares), um cilindro e um movimento circular (Figura 7b), objetos geométricos que foram considerados pelo aluno como aqueles que poderiam modelar as formas e movimentos do virabrequim do motor V6.

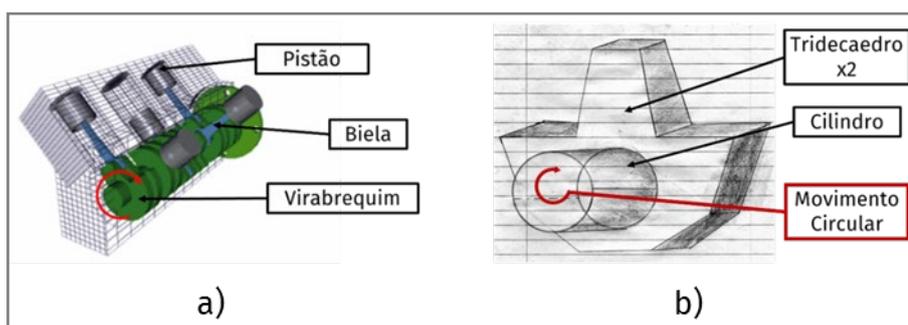


FIGURA 7: Fenômeno da simulação do projeto de Emanuel e esboço do virabrequim
FONTE: Adaptada de <http://bit.ly/2lO9t1O>. Acesso em: 10 ago. 2019

Por conta do número de objetos geométricos identificados no seu esboço, o aluno teve que resolver três tarefas de construção, das quais interessa reportar a maneira em que foi resolvida a segunda delas, formulada nos seguintes termos:

construir um tridecaedro a partir de um ponto interior. Esse ponto interior, nomeado B , refere-se àquele que foi representado no processo de resolução da primeira tarefa de construção e que representa um dos centros do cilindro construído previamente.

Para resolver a tarefa, Emanuel teve que construir: (i) a face da frente, (ii) a face traseira e (iii) as faces laterais do tridecaedro. Fazer isso significou determinar os 22 vértices da figura, 11 da face da frente e 11 da traseira. Como resultado de ter resolvido a primeira tarefa de construção, Emanuel obteve na interface do *software* um cilindro cujos centros das bases são os pontos A (centro do sistema de coordenadas) e B (ponto inserido no eixo x). Considerando esses objetos geométricos construídos, Emanuel determinou o primeiro vértice da face da frente do tridecaedro, nomeado C , interceptando a reta a' com uma circunferência cujo centro é o ponto B (Figura 8a). Para localizar seis dos 10 vértices do tridecaedro, o aluno percebeu que podia rotacionar o ponto C de forma iterada, aumentando o ângulo de rotação à razão de 30° até chegar no valor de 180° .

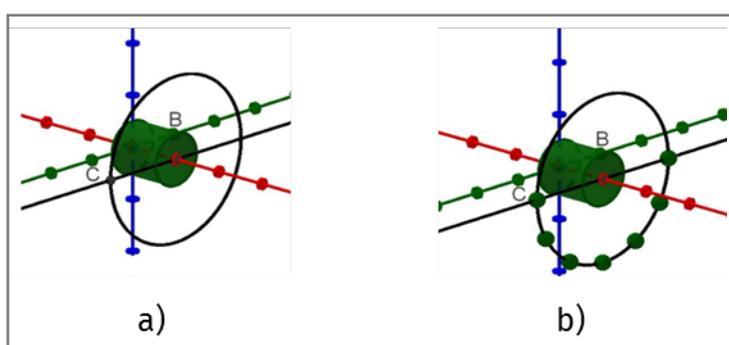


FIGURA 8: Localização dos primeiros sete vértices do tridecaedro
FONTE: Elaborada pelos autores

Para evitar o uso repetitivo da ferramenta *Girar em torno de Uma Reta*, do *software*, Emanuel decidiu executar o comando sequência, cuja sintaxe é:

Sequência[< Expressão >, < Variável >, < Valor inicial >, < Valor final >, < Incremento >]

Emanuel usou o comando *Girar*, vinculado à transformação rotação, como a *expressão* da sequência, aplicada a C em torno do eixo y com um valor angular i (*variável* da sequência). O *valor inicial* e *valor final* dizem respeito do primeiro e último valor que toma a variável i na sequência, respectivamente. No caso do aluno, esses valores foram de 30° e 180° , respectivamente. Finalmente, o *incremento* da sequência foi de 30° . De acordo com os valores mencionados, a sequência executou-se nos termos seguintes: *Sequência*[*Girar*[$C, i, eixoY$], $i, 30^\circ, 180^\circ, 30^\circ$]. A sintaxe anterior do comando *sequência* foi aplicada no *Campo de Entrada* do *software*, gerando os seis vértices da face da

frente do tridecaedro (Figura 8b). Os quatro vértices restantes da figura, nomeados como D , E , F e G (Figura 9), foram obtidos mediante construções geométricas usando as ferramentas de construção e medida do *software*.

Apesar de ter obtido todos os vértices da fase da frente do tridecaedro, Emanuel ainda não podia realizar a construção devido a que seis dos onze vértices da figura não eram reconhecidos como pontos pelo *software*, o que impossibilitava o uso da ferramenta *Polígono*. Essa situação se devia ao fato de que esses seis pontos foram obtidos através de uma sequência, cujo produto é uma lista de pontos, uma vez que é executada (no caso do aluno, a *lista1*).

Visando superar esse inconveniente, Emanuel decidiu agrupar os 11 vértices do tridecaedro em uma única lista, que fosse a união da primeira, obtida com uma segunda que agrupasse os outros cinco vértices da figura. Assim, o aluno criou a *lista2*, inserindo a expressão $lista2 = \{C, D, F, G, E\}$ no *Campo de Entrada* do GeoGebra e, posteriormente, combinou as duas listas usando o comando *concatenar*, expressado nos seguintes termos: *Concatenar* ($\{lista1, lista2\}$). Dessa forma, foi obtida a *lista3* que continha todos os vértices do tridecaedro (Figura 9a). Finalmente, utilizou-se o comando *polígono* do GeoGebra (aquele que permite construir polígonos a partir de listas de pontos) para desenhar a face da frente da figura. Nesse sentido, o aluno executou o comando *polígono* cuja sintaxe é *Polígono* [\langle Lista de pontos \rangle], substituindo o parâmetro pela *lista3*. A expressão inserida no *Campo de Entrada* foi *Polígono* [*lista3*], obtendo assim a face da frente do tridecaedro (Figura 9b).

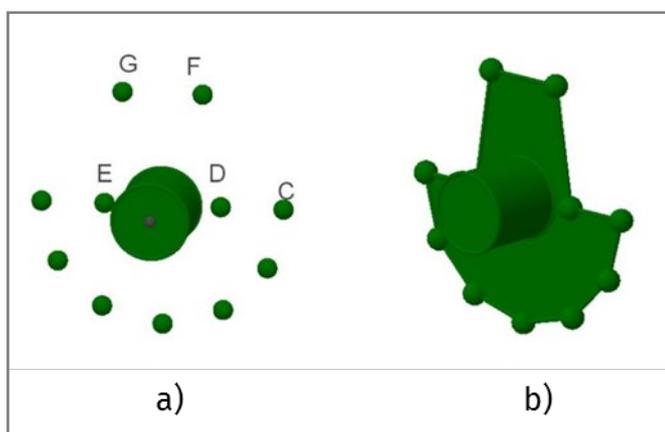


FIGURA 9: Vértices e face da frente do tridecaedro
FONTE: Elaborada pelos autores

A face traseira da figura foi construída após aplicar uma translação à *lista3* (o que gerou a *lista3'*) e executar a expressão *Polígono* [*lista3'*] no *Campo de Entrada* do *software* (Figura 10a). Uma vez determinados os 22 vértices do

tridecaedro, era possível construir as faces laterais dessa figura. Por conta dos 22 vértices estarem agrupados nas listas *lista3* e *lista3'*, a construção das faces laterais do tridecaedro foi realizada utilizando o comando *Polígono* [*< Point >*, ..., *< Point >*], que requer como parâmetros pontos previamente construídos. No entanto, o comando *Elemento* [*< Lista >*, *< Número (posição) >*] resultou idôneo para selecionar esses pontos provenientes de uma lista, na medida em que tal comando seleciona um objeto da lista indicada no primeiro parâmetro e, como segundo, a posição ocupada por esse objeto na lista. Considerando os comandos acima, Emanuel definiu a seguinte expressão para a primeira face lateral:

Polígono[*Elemento*[*lista3*,1], [*Elemento*[*lista3'*, 1], *Elemento*[*lista3'*, 2], [*Elemento*[*lista3*,2]]

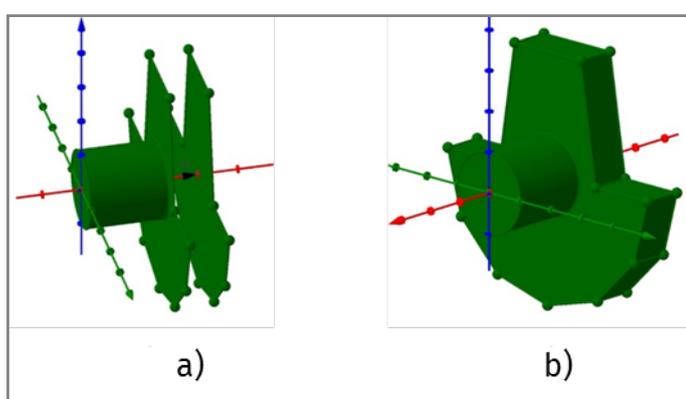


FIGURA 10: Tridecaedro construído na interface 3D do *software* GeoGebra
FONTE: Elaborada pelos autores

Novamente, para evitar a execução de ações repetitivas (como a construção do restante das faces laterais do tridecaedro), Emanuel usou o comando *sequência*, considerando os comandos e sintaxes anteriores. Em concreto, o aluno executou a seguinte sequência, mediante a qual foram obtidas as faces laterais restantes do tridecaedro, finalizando a sua construção (Figura 10b).

Sequência [*Polígono* [*Elemento* (*lista3*,*p*), *Elemento* (*lista3'*,*p*), *Elemento* (*lista3'*,
 + 1), *Elemento* (*lista3*,*p* + 1)],*p*, 1,11,1]

Reflexões finais

As experiências de ESG descritas neste artigo reportam algumas das vantagens que supõe o uso do comando *sequência* para resolver tarefas de construção geométrica com o *software* GeoGebra. Assim, o uso desse tipo de comando permite ao usuário tanto diminuir o número de objetos construídos no programa (logo, no número de operações executadas) quanto modificar e/ou

vincular objetos geométricos para evitar a repetição de técnicas de construção de novos objetos (ou, no máximo, ter que reconstruir todo o desenho dinâmico).

Além disso, este artigo mostrou que o acréscimo do número de objetos construídos pelo comando sequência depende também de sintaxes mais complexas que integrem outros comandos do *software*. Portanto, promover o uso dos comandos, em geral, e das sequências, em particular, supõe um aspecto importante, na medida em que essas funcionalidades permitem articular e/ou vincular os objetos matemáticos representados na interface do *software*, garantindo a obtenção de um desenho dinâmico com maior fidelidade quanto à realidade modelada.

Para finalizar, considera-se pertinente que professores de matemática, com interesse em incorporar TD em sala de aula, possam dispor de oportunidades de formação profissional em que se mobilizem saberes necessários para utilizar comandos (como a sequência) em atividades educativas, como as descritas neste artigo ou similares, utilizando SGD para desenvolver aprendizagem em geometria.

Referências

- ALQAHTANI, M. M.; POWELL, A. B. Instrumental Appropriation of a Collaborative, Dynamic-Geometry Environment and Geometrical Understanding. *International Journal of Education in Mathematics, Science and Technology*, v. 4, n. 2, p. 72-83, 2016.
- BORBA, M. C.; SCUCUGLIA, R.; GADANIDIS, G. *Fases das tecnologias digitais em Educação Matemática*. 2ª ed. Belo Horizonte: Autêntica, 2018.
- BORBA, M. C.; VILLARREAL, M.E. *Humans-With-Media and the Reorganization of Mathematical Thinking: information and communication technologies, modeling, experimentation and visualization*. New York: Springer, 2005. v 39.
- CASTILLO, L. A.; PRIETO, J. L. El uso de comandos y guiones en la elaboración de simuladores con GeoGebra. *UNIÓN*, n. 52, p. 250-262, 2018.
- CLARK, D.; NELSON, B.; SENGUPTA, P.; D' ANGELO, C. *Rethinking science learning through digital games and simulations: Genres, examples, and evidence*. Learning science: Computer games, simulations, and education workshop sponsored by the National Academy of Sciences. *Anais [...]*, Washington, DC: 2009.
- HOYLES, C. Transforming the mathematical practices of learners and teachers through digital technology. *Research in Mathematics Education*, v. 20, n. 3, p. 209-228, 2018.

LABORDE, C. Cabri-geómetra o una nueva relación con la geometría. In: PUIG, L. (Org.). *Investigar y enseñar. Variedades de la educación matemática*. Madrid: Una empresa docente, 1997. p. 33-48.

National Council of Teachers of Mathematics. *Principles and Standards for School Mathematics*. Reston, VA: Author, 2000.

OLIVERO, F.; ROBUTTI, O. Measuring in dynamic geometry environments as a tool for conjecturing and proving. *International Journal of Computers for Mathematical Learning*, v. 12, n. 2, p. 135-156, 2007.

ROJANO, T. El futuro de las tecnologías digitales en la educación matemática: prospectiva a 30 años de investigación intensiva en el campo. *Educación Matemática*, v. 26, n. Especial, p. 11-30, 2014.

RUBIO, L.; PRIETO, J. L.; ORTIZ, J. La matemática en la simulación con GeoGebra. Una experiencia con el movimiento en caída libre. *International Journal of Educational Research and Innovation (IJERI)*, v. 2, p. 90-111, 2016.

SANTOS-TRIGO, M.; REYES-RODRÍGUEZ, A. The use of digital technology in finding multiple paths to solve and extend an equilateral triangle task. *International Journal of Mathematical Education in Science and Technology*, v. 47, n. 1, p. 58-81, 2016.

TROCKI, A.; HOLLEBRANDS, K. The Development of a Framework for Assessing Dynamic Geometry Task Quality. *Digital Experiences in Mathematics Education*, p. 1-29, 2018.