

# Geração Procedural de Conteúdo para Criação de Fases de Jogos Educativos usando Gramática

Wendell Oliveira de Araújo<sup>1</sup>, Eduardo Henrique da Silva Aranha<sup>1</sup>

<sup>1</sup>Departamento de Informática e Matemática Aplicada  
Universidade Federal do Rio Grande do Norte (UFRN) – Natal, RN – Brasil

Wendell.cmd@gmail.com, eduardoaranha@dimap.ufrn.br

**Abstract.** *Procedural content generation (PCG) is the automatic creation of game contents, such as textures, sounds and levels. Thus, PCG contributes to the creation of new game levels without the need for human intervention. Thus, the research seeks to make use of PCG techniques for the level creation of educational games that require the player to achieve certain pedagogical goals throughout the game. For this, we propose a three-stage generation approach. Besides, this approach was applied using a grammar-based PCG technique. Through this approach, we show how to generate playable phases by joining entertainment games for educational purposes.*

**Resumo.** *A geração procedural de conteúdo (PCG) trata da criação automática de conteúdos como texturas, sons, fases de jogos através de algoritmos. Assim, a PCG contribui para a criação de novas fases sem que haja a necessidade de intervenção humana. Com isso, a pesquisa busca fazer uso das técnicas de PCG para a criação de fases de jogos educativos que requeiram do jogador atingir certos objetivos pedagógicos ao longo do jogo. Para isso, propomos uma abordagem de geração em três etapas. Além disso, esta abordagem foi aplicada usando uma técnica de PCG baseada em gramática. Através dessa abordagem, apresentamos como gerar fases jogáveis fazendo a união de jogos de entretenimento com objetivos pedagógicos.*

## 1. Introdução

No cenário atual, pode ser observado que alunos consideram algumas matérias desmotivantes e que por isso não se empenham para aprendê-las, passando com dificuldades nas mesmas [Mattar 2010]. Isto se dá em muito dos casos ao fato do avanço da tecnologia com atividades mais interativas e divertidas para os alunos, levando assim o aluno ao desinteresse pelos métodos de ensinamentos tradicionais e metodologias pouco atrativas perante as novas tecnologias. Assim, por conta desse desinteresse, vários estudantes enfrentam reprovações ou avançam sem ter aprendido os conteúdos adequadamente [Madeira et al. 2015].

Nesse cenário, os jogos passam a ter um papel importante na educação. De acordo com [Savi E Ulbricht 2008], os jogos digitais educacionais podem trazer diversos benefícios como engajamento dos estudantes, facilitação do aprendizado, desenvolvimento de habilidades cognitivas, aprendizado por descoberta, experiência de novas identidades, socialização, coordenação motora e um comportamento expert no conteúdo que é proposto pelo jogo, visto que se um aluno se torna expert no jogo, ele aumenta suas chances em se tornar um expert (ou pelo menos um aluno interessado) no assunto abordado. Além disso, algumas pesquisas procuram realizar a adaptação em jogos

para promover cada vez mais este engajamento dos alunos, porém sendo ainda pouco explorada, com poucas pesquisas relacionadas ao tema [Araújo et al. 2016].

Muito se estima sobre o custo para o desenvolvimento de um jogo, pois a complexidade do jogo tem uma relação direta com o custo de produção do mesmo. A relação com o custo também se dá ao fato dos jogos envolverem diversos profissionais, como programadores, artistas, músicos, game designers, entre diversos outros. Contudo, algumas empresas conseguem reduzir um pouco os custos atribuindo mais de uma função para funcionários específicos e por meio de ferramentas de terceiros que auxiliam no desenvolvimento de gráficos, animações, física e demais componentes de um jogo [Gregory 2009]. Entretanto, há uma carência de ferramentas que auxiliem o desenvolvimento de partes que exigem mais criatividade, como a criação de níveis, texturas, músicas, missões e comportamentos de personagens não jogáveis. Dessa forma, esse tipo de conteúdo muitas vezes deve ser produzido para cada projeto que é desenvolvido, deixando assim uma produção manual bastante custosa [Amato e Moscato 2017].

Em meio a essa complexidade e alto custo de desenvolvimento, uma alternativa é o uso de Geração Procedural de Conteúdo (PCG). A PCG se refere à criação de conteúdo realizada por meio de algoritmos [Togelius et al. 2011]. Os usos das técnicas de PCG apresentam diversos benefícios como: agilidade na produção; redução dos custos; alto valor de rejogabilidade; alta variação no jogo proporcionando novas experiências aos usuários; adaptação automática da dificuldade do jogo de acordo com o jogador [Korn et al. 2017]; redução do custo de memória [Stephenson e Renz 2016]; e auxílio aos designers servindo como fonte de inspiração [Moghadam e Rafsanjani 2017]. Por isso, alguns projetos comerciais e de pesquisa na área de jogos digitais vem optando por utilizar alguma técnica de PCG, ou chegam até mesmo a desenvolver sua própria PCG para atender necessidades específicas do projeto. Até porque alguns estudos indicam que uma PCG bem produzida pode conseguir criar um conteúdo com uma qualidade tão boa quando um conteúdo feito manualmente [Korn et al. 2017].

Entretanto, no contexto de jogos educativos a PCG é uma tarefa ainda mais complicada, pois envolve geração de aspectos de entretenimento e de educação. Visando suportar essa atividade, o objetivo principal deste trabalho é propor uma abordagem para integrar componentes pedagógicos no processo de PCG para geração de níveis de um jogo. Em outras palavras, viabilizar a criação de novos níveis de um jogo envolvendo conteúdo educacional sem que haja a necessidade de uma equipe de desenvolvedores e de professores. A abordagem proposta é genérica, podendo ser aplicada a jogos e conteúdos pedagógicos diversos.

## 2. Background

De acordo com o [Togelius et al. 2011], a definição de geração procedural de conteúdo (PCG) é a criação algorítmica de conteúdo do jogo com entrada de usuários limitada ou indireta. Em outras palavras, a PCG refere-se a um software de computador que pode criar conteúdo de jogo por conta própria ou em conjunto com seres humanos. O termo conteúdo pode se referir a várias partes de um jogo: níveis, mapas, regras de jogo, texturas, histórias, músicas, armas, veículos, personagens, etc.

Entre as técnicas de PCG mais comuns temos a *search-based PCG*, que é o uso de técnicas de computação evolutiva, algoritmos de busca/otimização estocástica e

métodos similares para se buscar um conteúdo desejado. Um exemplo disto é visto no trabalho realizado por [Stephenson e Renz 2016]. Para geração de fases de um jogo, podemos destacar os métodos *Cellular automata*, *space partitioning*, *perlin-noise*, *grammar-based*, *Answer Set Programming*, entre outros [Shaker et al. 2016], [Smith et al. 2016], [Korn et al. 2017]. [Rodrigues et al. 2017] utilizou PCG usando uma abordagem aleatória para geração de fases em um jogo educativo de matemática, no qual apresentou resultados positivos ao aplicá-lo com estudantes, melhorando sua pontuação no teste escrito e promoveu o entusiasmo para praticarem mais. Para o desenvolvimento deste trabalho, foi escolhido o método baseado em gramática, tendo como vantagem a possibilidade de se garantir pelas regras da gramática a construção de conteúdo (fases) jogáveis.

### 2.1. PCG baseada em gramática (*grammar-based PCG*)

Uma gramática (formal) é um conjunto de regras de geração para reescrever *Strings*, ou seja, transformar uma *String* em outra [Shaker et al. 2016]. Da mesma forma, [Prusinkiewicz e Lindenmayer 1990] também afirma que as gramáticas envolvem uma técnica de reescrita, na qual objetos complexos (fases de um jogo, por exemplo) são gerados substituindo sucessivamente partes de um objeto inicial simples (estrutura geral da fase) usando um conjunto de regras ou produções de reescrita.

Na geração de conteúdo para jogos, a gramática foi aplicada de diversas formas diferentes. Em [Font et al. 2016] é apresentado um gerador de fases através de algoritmos evolutivos baseados em gramática. Outra aplicação de *grammar-based PCG* pode ser vista em [Smith e Treanor et al. 2009], onde uma primeira gramática é usada para gerar ritmos e uma outra gera a geometria do cenário com base nos ritmos gerados.

Métodos baseados em gramáticas também foram usados na geração de níveis na competição *Mario AI Competition*, na qual os autores de [Shaker e Nicolau et al. 2012] evoluíram os níveis jogáveis usando gramática evolutiva.

## 3. Abordagem Proposta

Para sistematizar a aplicação PCG a jogos digitais educacionais, este trabalho propõe uma abordagem baseada em gramática. Com isto, a geração das fases é realizada com base nas regras impostas pela mesma, criando restrições específicas para os cenários e evitando assim cenários impossíveis de serem concluídos.

Essa abordagem foi desenvolvida considerando os seguintes requisitos:

1. Facilitar o processo de PCG aplicado a jogos digitais educacionais;
2. Suportar diferentes estilos de jogos digitais;
3. Suportar mecânicas e componentes relativos a diferentes objetivos pedagógicos;
4. Permitir combinar diferentes jogos digitais (entretenimento) com diferentes objetivos pedagógicos, desde que compatíveis (plataforma, top-down, etc.);
5. Permitir o desenvolvimento incremental de jogos digitais e de mecânicas e componentes pedagógicos.

Estes requisitos, quando satisfeitos, podem suportar o desenvolvimento de plataformas de jogos digitais educativos que sejam desenvolvidas de forma incremental, e de forma que as fases dos jogos possam trabalhar diferentes objetivos pedagógicos, de acordo com a necessidade dos estudantes.

Assim, pensando em uma geração automatizada de fases com conteúdo educativo, a abordagem proposta divide-se em duas fontes de componentes (modelo de jogo e modelo pedagógico) e três etapas de PCG. A Figura 2 apresenta uma visão geral da abordagem proposta, cujos elementos são detalhados a seguir.

### 3.1. Fontes de Componentes do Jogo

Esta abordagem propõe o uso de cenários e mecânicas de jogos clássicos e consagrados no mercado para garantir o interesse dos estudantes pelos jogos educativos. O **Modelo de Jogo** (Fig. 1) representa na abordagem proposta a fonte de componentes de um jogo de entretenimento. Esses componentes são organizados em dois grupos: (i) os que formam a estrutura básica de um cenário do jogo (chão, paredes, herói, etc.); (ii) demais elementos que completam o cenário e regulam o grau de dificuldade da fase do jogo, configurando os desafios enfrentados (inimigos, obstáculos, etc.) e recompensas disponíveis (moedas de ouro, etc.).

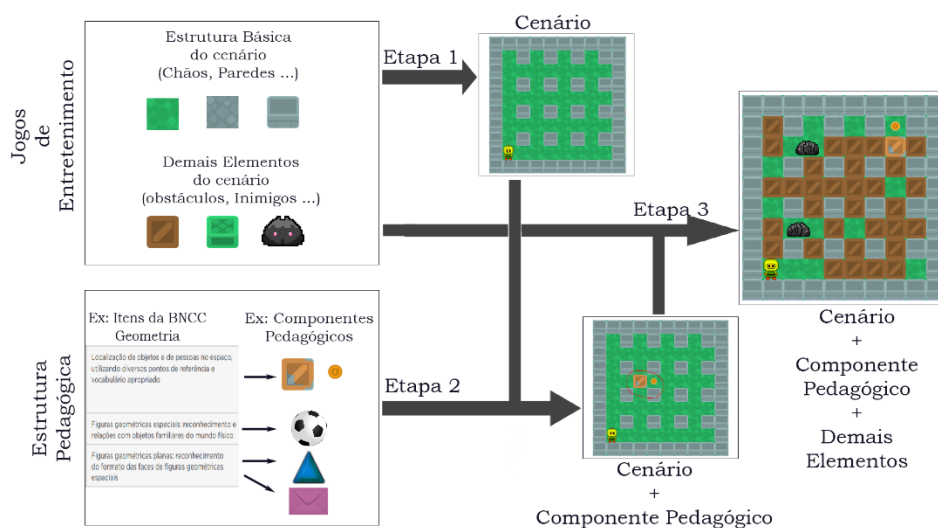


Figura 1. Visão Geral da Abordagem Proposta.

Para se inserir aspectos pedagógicos em um jogo, é utilizado um **Modelo Pedagógico**. Este modelo é organizado em habilidades que o jogo precisa trabalhar e componentes de jogo que podem ser utilizados para trabalhar essas habilidades. As habilidades podem ser as definidas pela Base Nacional Comum Curricular (BNCC) do MEC ou por outro documento, como a dos referenciais para formação de computação na educação básica da Sociedade Brasileira de Computação (SBC). As habilidades selecionadas para serem trabalhadas no jogo precisam ser contempladas com pelo menos um componente que implementa alguma mecânica que trabalha aquela habilidade específica.

Essa organização permite, como mostrado nas seções a seguir, a combinação de diferentes modelos de jogos com diferentes modelos pedagógicos, desde que as mecânicas sejam compatíveis (jogos plataforma, *top-down*, etc.).

### 3.2. Etapa 1 – Geração da Estrutura do jogo

Uma vez especificado o modelo de jogo, a primeira etapa de PCG é responsável por definir o mapeamento inicial da fase, definindo toda a área trafegável pelo personagem.

O objetivo principal desta etapa é montar a base inicial do cenário e passar a informação para as próximas etapas dos locais disponíveis para adição de componentes pedagógicos e demais elementos que formarão o cenário do jogo.

Portanto, essa etapa irá utilizar os componentes do modelo de jogo utilizados para montar a estrutura básica do cenário, como chãos e paredes, o personagem principal e outros elementos obrigatórios no cenário e com alguma função específica e importante para o jogo. Uma gramática precisa ser especificada para gerar essa estrutura básica do jogo.

### **3.3. Etapa 2 – Geração dos Componentes Pedagógicos**

Com as áreas trafegáveis definidas, o próximo passo é o de inserir componentes pedagógicos, uma vez que a fase a ser gerada tem obrigatoriamente um caráter educacional. Para isso, verifica-se as habilidades selecionadas no modelo pedagógico e os chamados componentes pedagógicos que conseguem trabalhar essas habilidades no jogo.

Caso a geração desta etapa não encontre um local adequado para inserir um determinado componente pedagógico, a PCG desta etapa pode realizar modificações na estrutura básica do cenário, como ampliar os espaços disponíveis, etc. Também é possível balancear o nível de dificuldade educacional com base nas informações disponíveis sobre o jogador. Em termos de gramática, é necessário definir as regras de inserção de cada tipo de componente pedagógico, considerando suas especificidades.

### **3.4. Etapa 3 – Geração dos Demais Elementos**

Com a alocação realizada dos componentes obrigatórios para o jogo educacional, outros componentes podem ser inseridos para se tentar garantir uma melhor atratividade para o jogo. São inseridos itens de decoração do cenário (vasos, quadros, estantes), de acordo com o modelo de jogo selecionado, além de inimigos, obstáculos, outros desafios e recompensas, características presentes em qualquer jogo de sucesso.

A PCG utilizada nesta etapa pode ainda levar em consideração as habilidades de jogo do estudante, regulando o nível de dificuldade das mecânicas de entretenimento. Esta etapa é a última para garantir que os elementos adicionados não interfiram na solução do objetivo pedagógico e de entretenimento proposto para que se vença a fase. Com relação à gramática, é necessário definir as regras de inserção de cada tipo de componente, podendo variar de regras simples a regras mais elaboradas, envolvendo a inserção de mais de um componente em conjunto.

## **4. Avaliação da Abordagem Proposta**

Para avaliar a abordagem proposta por este trabalho, a mesma é aplicada a dois jogos digitais consagrados e fazendo uso da técnica de PCG baseada em gramática. Um conjunto de componentes pedagógicos são utilizados para trabalhar habilidades matemáticas, e será avaliada a geração de fases considerando o cruzamento entre os jogos e componentes pedagógicos implementados. A avaliação irá assim verificar se os requisitos apresentados anteriormente estão sendo satisfeitos pela proposta.

#### 4.1. Modelos de Jogos

Para realização do estudo proposto, foram desenvolvidos os componentes para se montar dois modelos de jogos digitais consagrados, o *Sokoban* e o *Bomberman*. O *Bomberman* é um jogo de estratégia no qual o personagem pode colocar bombas, explodindo caixas e inimigos que tentam impedi-lo de chegar ao ponto de saída (Figura 2-1). Já o *Sokoban* é um jogo de movimentação de caixas, sendo seu principal objetivo colocá-las em cima dos pontos marcados (Figura 2-2).



Figura 2. Exemplo de cenário dos jogos *Bomberman* e *Sokoban*

#### 4.2. Modelo Pedagógico

Foram selecionadas algumas habilidades matemáticas a partir da BNCC [Ministério da Educação 2017], para alunos do 1º e 2º anos do ensino fundamental:

- **EF01MA11:** Descrever a localização de pessoas e de objetos no espaço em relação à sua própria posição, utilizando termos como à direita, à esquerda, em frente, atrás.
- **EF01MA12:** Descrever a localização de pessoas e de objetos no espaço segundo um dado ponto de referência, compreendendo que, para a utilização de termos que se referem à posição, como direita, esquerda, em cima, em baixo, é necessário explicitar-se o referencial.
- **EF01MA13:** Relacionar figuras geométricas espaciais (cones, cilindros, esferas e blocos retangulares) a objetos familiares do mundo físico.
- **EF01MA14:** Identificar e nomear figuras planas (círculo, quadrado, retângulo e triângulo) em desenhos apresentados em diferentes disposições ou em contornos de faces de sólidos geométricos.

Foram implementadas duas mecânicas, uma de localização de objetos (a partir de pontos de referência) para ser utilizada com as habilidades *EF01MA11* e *EF01MA12*, e uma outra de coleta de objetos (exigindo reconhecimento de formas geométricas) para as habilidades *EF01MA13* e *EF01MA14*.

#### 4.3. Configurando a Geração das Fases

Foram escritas as gramáticas para as três etapas definidas para a abordagem proposta por este trabalho, bem como mecanismos para implementar a PCG a partir delas. As gramáticas para os dois jogos considerando apenas a 1º etapa são mostradas a seguir. Alguns dos elementos da gramática são explicados a seguir:

- **<ParedesX>:** Preenche uma linha inteira com paredes.
- **<ParedesAltX>:** Preenche uma linha inteira alternando os elementos entre parede e chão em cada posição da linha.

- **<BlocosX>**: Preenche uma linha com posições possuindo caixas ou não, para serem removidas com as bombas colocadas pelo bomberman, sempre com o início e fim da linha sendo paredes.
- **<VazioX>**: Preenche uma linha somente com o chão e as paredes nas extremidades.
- **<Personagem>**: Cria uma linha inteira com o personagem no início da linha e completa o restante com o chão do cenário.
- **ProximoX e ProximoY**: ambos servem para que ocorra uma modificação nas posições X e Y que os objetos são colocados, guardando assim uma posição atual em que se encontra.
- **<DLevel>**: Esta ação pega todos os espaços vazios que sobraram no cenário e decide se continua vazio ou se coloca um bloco no local.

<pre>// Chamada da Geração da fase do Bomberman: &lt;Jogo&gt; ::= //Estrutura   &lt;ParedesX&gt;&lt;Personagem&gt;   (&lt;ParedesAltX&gt; ProximoY &lt;VazioX&gt; ProximoY)<sup>(Linhas-4)</sup>   &lt;ParedesAltX&gt; ProximoY   [&lt;VazioX&gt; ProximoY]<sup>(Linhas%2 != 0)</sup>   &lt;ParedesX&gt;   //Escolha do Objetivo Pedagógico   &lt;Localizacao&gt;   //Elementos de cenário   &lt;DLevel&gt;   &lt;Inimigos&gt;<sup>2</sup>  // Chamada da Geração da fase do Sokoban: &lt;Jogo&gt; ::= //Estrutura   &lt;ParedesX&gt;(&lt;VazioX&gt; ProximoY)<sup>(Linhas-2)</sup> &lt;ParedesX&gt; &lt;Player&gt;   //Objetivo Pedagógico   Coleta();   //Adicionais ao cenário   &lt;Dificuldade&gt;&lt;Enemies&gt;<sup>2</sup></pre>	<pre>// Para Geração do Bomberman: &lt;ParedesX&gt; ::= &lt;Colisivel&gt;<sup>Nº colunas</sup> ProximoY &lt;ParedesAltX&gt; ::= &lt;Colisivel&gt; (&lt;Chao&gt; &lt;Colisivel&gt;)<sup>(Nº colunas-1)</sup> [&lt;Colisivel&gt;]<sup>(Nº colunas % 2 = 0)</sup> &lt;BlocosX&gt; ::= &lt;Colisivel&gt; &lt;Espacos&gt;<sup>(Nº colunas-1)</sup> &lt;Colisivel&gt; &lt;VazioX&gt; ::= &lt;Colisivel&gt; &lt;Chao&gt;<sup>(Nº colunas-1)</sup> &lt;Espacos&gt; ::= &lt;Chao&gt; [75%]   &lt;Bloco&gt;[25%] &lt;Personagem&gt; ::= &lt;Colisivel&gt; &lt;Player&gt; &lt;Chao&gt;<sup>(Nº colunas-2)</sup> &lt;Colisivel&gt; &lt;Colisivel&gt; ::= &lt;Parede&gt; &lt;Bloco&gt; ::= bloco ProximoX &lt;Chao&gt; ::= chao ProximoX &lt;Player&gt; ::= player &lt;Parede&gt; ::= parede ProximoX &lt;Inimigos&gt; ::= inimigos &lt;Colunas&gt; ::= [5..20] &lt;Linhas&gt; ::= [5..10] &lt;DLevel&gt; ::= &lt;Espacos&gt;<sup>(Nº espaços vazios)</sup>  // Para geração do Sokoban adicionar essas gramáticas ao Bomberman &lt;Dificuldade&gt; ::= &lt;Caixa&gt;<sup>Total_Caixas</sup> &lt;Botao&gt;<sup>Total_Caixas</sup> &lt;Caixa&gt; ::= caixa chao ProximoX &lt;Botao&gt; ::= botao ProximoX &lt;Total_Caixas&gt; ::= [2..5]</pre>
--	---

Figura 3. Gramática para geração da fase do *Bomberman* e do *Sokoban*.

Como o *Sokoban* tem um estilo de fase similar ao *Bomberman* a mesma gramática pode ser utilizada como base para a geração de fases do mesmo, adicionando-se novos elementos ao jogo do *Sokoban* como a inclusão de inimigos para atrapalhar o jogador.

Fazendo uso das gramáticas apresentadas, foi possível gerar a estrutura básica de cada um dos jogos propostos. Para se adicionar o aspecto pedagógico (etapa 2), são utilizados os componentes de localização e coleta de objetos. Assim, temos uma segunda gramática para esta etapa:

<pre>//Objetivos Pedagógicos &lt;ObjPedag&gt; ::= &lt;Localizacao&gt;[50%]   &lt;Coleta&gt;[50%] &lt;Localizacao&gt; ::= &lt;OPLocal&gt; &lt;Coleta&gt; ::= &lt;Itens&gt;<sup>Total_Itens</sup>  //Pedagógico - Localização &lt;OPLocal&gt; ::= &lt;Cima&gt;[25%]   &lt;Baixo&gt;[25%]     &lt;Direita&gt;[25%]   &lt;Esquerda&gt;[25%] &lt;Cima&gt; ::= &lt;ObjRef(y,x)&gt; ObjPerdido(y+1,x)&gt; &lt;Baixo&gt; ::= &lt;ObjRef(y,x)&gt; &lt;ObjPerdido(y-1,x)&gt; &lt;Esquerda&gt; ::= &lt;ObjRef(y,x)&gt; &lt;ObjPerdido(y,x-1)&gt; &lt;Direita&gt; ::= &lt;ObjRef(y,x)&gt; &lt;ObjPerdido(y,x+1)&gt; &lt;ObjRef&gt; ::= objref &lt;ObjPerdido&gt; ::= objperdido</pre>	<pre>//Pedagógico - Coleta &lt;Itens&gt; ::= &lt;Quadrangulares&gt;[25%]     &lt;Retangulares&gt;[25%]     &lt;Triangulares&gt; [25%]     &lt;Circulares&gt;[25%] &lt;Quadrangulares&gt; ::= quadrado &lt;Retangulares&gt; ::= retangulo &lt;Triangulares&gt; ::= triangulo &lt;Circulares&gt; ::= circulo &lt;Total_itens&gt; ::= [2...5]</pre>
---	--

Figura 4. Gramática para geração dos desafios pedagógicos.



Conforme observado na Figura 4, pode ser observado que algumas trocas ou ações são realizadas baseadas numa porcentagem para ser escolhida. Dessa forma, para as fases geradas, deixamos todos com a mesma probabilidade de escolha, buscando assim trazer uma maior diversificação de componentes em casa fase. Porém, esses valores sejam alterados para se trabalhar mais (ou apenas) uma determinada habilidade.

Uma terceira gramática foi escrita para a etapa 3, que insere os demais componentes de jogo, visando torná-lo mais atrativo e desafiador para o jogador.

#### 4.4. Gerando as Fases

A abordagem foi executada de acordo com o implementado e explicado nas seções anteriores. Diversos cenários foram gerados e jogados, como os mostrados na Figura 5. Vale ressaltar que o número de linhas e colunas são definidos aleatoriamente.

No caso do *Bomberman*, o jogo trata de localizar a moeda que aparece nas imagens (estariam invisíveis ao jogador) a partir de dicas indicadas ao iniciar a partida (ex: escondida à direita da caixa mais clara). Já nas fases mostradas para o *Sokoban*, o jogador precisa reconhecer as formas e coletá-las conforme ele seja informado. Em outros cenários, fases foram geradas para os dois jogos invertendo-se o componente pedagógico e consequentemente as habilidades trabalhadas.

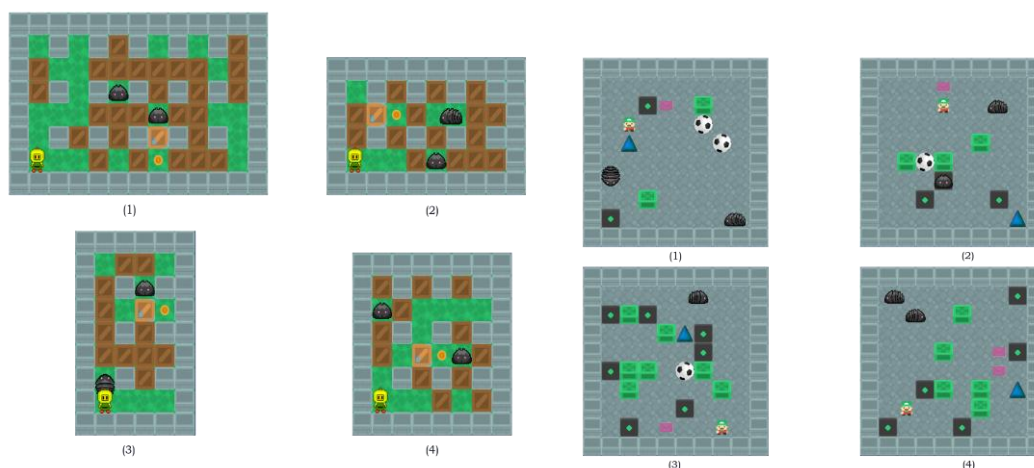


Figura 5. Fases geradas pela abordagem proposta.

#### 4.5. Resultados

Conforme visto, foi possível criar fases jogáveis usando a abordagem proposta. Entretanto, algumas limitações na implementação precisariam ser melhor trabalhadas para poderem ser usadas na prática. Por exemplo, não foi levada em consideração a posição que inimigos deveriam nascer, aparecendo em alguns casos um inimigo tão próximo ao jogador que não teria como o mesmo conseguir sair vivo.

O estudo mostra que mesmo para jogos com mecânicas e objetivos diferentes, pode haver um reuso de componentes pedagógicos para acelerar a produção de jogos com fins educacionais diferentes, mas com mesmo modelo de jogo como base. Em termo de esforço de criação das gramáticas, houve ainda uma grande similaridade entre os jogos *Bomberman* e *Sokoban*, diminuindo ainda mais o esforço de uso da abordagem.



Portanto, foi possível gerar fases jogáveis com o uso de mecânicas de jogos que são voltadas para o entretenimento aliados a componentes com objetivos pedagógicos. Esta abordagem pode assim viabilizar a produção em maior escala de jogos educacionais, uma vez que permite a aplicação de componentes pedagógicos a diferentes modelos de jogos, desde que possuam mecânicas compatíveis. Estes resultados indicam o atendimento da abordagem proposta em relação aos requisitos definidos para a mesma.

Uma limitação do estudo apresentado se refere à qualidade das fases geradas e ao reduzido número de jogos e habilidades que podem ser aplicados. Desta forma, mais estudos precisam ser realizados para se verificar possíveis limitações da abordagem quanto a variabilidade de jogos existentes e de habilidades que possam ser trabalhadas.

## 5. Considerações finais

Muito se tem aplicado em termos de geração de fases voltadas para jogos de entretenimento, mas pouco tem se visto quando considerado jogos educativos. Neste contexto, este trabalho apresenta uma abordagem para a geração de fases para jogos educativos fazendo uso da PCG com base em gramática. Essa abordagem conta com modelos de jogo e pedagógico, e em uma PCG em três etapas.

Foi realizada uma avaliação da abordagem através de sua aplicação a dois jogos consagrados e quatro habilidades matemáticas definidas na BNCC. Os resultados alcançados nessas aplicações se mostraram promissores e apresentam possibilidade de reutilizar gramáticas para outros tipos de jogos, assim como combinar esses jogos para apresentarem mecânicas e objetivos diferentes. Uma contribuição importante deste trabalho é possibilitar a construção de diversos jogos com esforço reduzido, implementando-se componentes individuais e montando-se as fases com o uso de PCG. Esse processo pode ainda envolver aspectos de personalização e de adaptabilidade, caso se tenha informações sobre o estudante, aumentando ainda mais os ganhos pelo uso da abordagem.

## Referências

- Amato, F., Moscato, F. (2017) “Formal Procedural Content Generation in games driven by social analyses”, In: 31st International Conference on Advanced Information Networking and Applications Workshops - AINA, p. 674-679.
- Araújo, W. O., Silva, T. R., Aranha, E. H. S. (2016) “Aplicação de Jogos Adaptativos na Educação: Uma Revisão Sistemática da Literatura”, In: Simpósio Brasileiro de Informática na Educação – SBIE, p. 587-596.
- Gregory, J. (2009) Game Engine Architecture. pages 25–28. A K Peters.
- Korn, O., Blatz, M., Rees, A., Schaal, J., Schwind, V., Gorlich, D. (2017) “Procedural Content Generation for Game Props? A Study on the Effects on User Experience”, In: Computers in Entertainment (CIE) – Theoretical and Practical Computer Applications in Entertainment, Nr. 15(2).
- Liapis, A., Yannakakis, G. N., Togelius, J. (2013) “Sentient Sketchbook: Computer-Assisted Game Level Authoring”, In: Proceedings of the 8th Conference on the Foundations of Digital Games (FDG).

- Madeira, C., Câmara, L., Beserra, I., Tavares, R. (2015) “Mathmare: um jogo de plataforma envolvendo desafios matemáticos do ensino médio”, In: XIV Simpósio Brasileiro de Jogos e Entretenimento Digital, Teresina.
- Mattar, J. (2010) Games em educação: como os nativos digitais aprendem. São Paulo: editora Pearson Prentice Hall.
- Moghadam, A. B., Rafsanjani, M. K. (2017) “A Genetic Approach in Procedural Content Generation for Platformer Games Level Creation”, In: 2nd Conference on Swarm Intelligence and Evolutionary Computation (CSIEC), Iran.
- Persson, M. N. Minecraft. (2009). <https://minecraft.net/pt-br/>, Janeiro 2018.
- Prusinkiewicz, P., Lindenmayer, A. (1990) “The Algorithmic Beauty of Plants”, In: Springer.
- Rodrigues, L. A. L., Bonidia, R. P., Brancher, J. D. (2017) “A Math Educacional Computer Game Using Procedural Content Generation”, In: XXVIII Simpósio Brasileiro de Informática na Educação – SBIE, p.756-765.
- Savi, R., Ulbricht, V. R. (2008) “Jogos Digitais Educacionais: Benefícios e Desafios”, In: Revista Novas Tecnologias na Educação - RENOTE, v.6, n.2.
- Shaker, N., Nicolau, M. et al. (2012) “Evolving Levels for Super Mario Bros Using Grammatical Evolution”, In: Computational Intelligence and Games (CIG), IEEE Conference on. IEEE, pp. 304-311.
- Shaker, N., Togelius, J., Nelson, M. J. (2016) “Procedural Content Generation in Games: A Textbook and an Overview of Current Research”, In: Springer. ISBN 978-3-319-42714-0.
- Smith, G., Treanor, M. et al. (2009) “Rhythm-Based Level Generation for 2D Platformers”, In: Proceedings of the 4th International Conference on Foundations of Digital Games. ACM, pp. 175-182.
- Smith, T., Padget, J., Vidler, A. (2016) “Design Space Descriptions for Logical Generation of Content”, In: Frontiers in Artificial Intelligence and Applications, pg. 209-214.
- Stephenson, M., Renz, J. (2016) “Procedural generation of complex stable structures for angry birds levels”, In: IEEE Computational Intelligence and Games Conference (CIG).
- Togelius, J., Kastbjerg, E., Sschedl, D., Yannakakis, G. N. (2011) “What is Procedural Content Generation?: Mario on the Borderline”, In: Proceedings of the 2nd Workshop on Procedural Content Generation in Games.