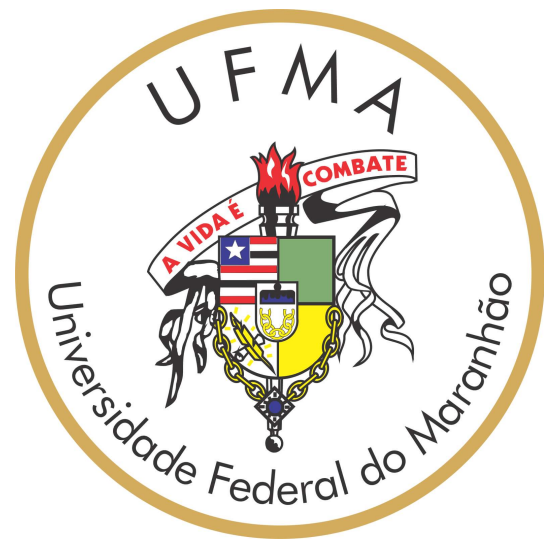


# Procedural terrain generator for platform games using Markov chain



Gleidson Mendes Costa<sup>1</sup>, Tiago Bonini Borchardt<sup>2</sup>, Apresentador: Juliano Leonardo Soares<sup>3</sup>

Vision and Image Processing Laboratory - VIPLab, Universidade Federal do Maranhão - UFMA<sup>1</sup>

Vision and Image Processing Laboratory - VIPLab, Universidade Federal do Maranhão - UFMA<sup>2</sup>

Universidade Federal de Santa Maria - UFSM<sup>3</sup>



## Introdução

O designer de games é uma tarefa muito complexa e que depende muitos recursos e tempo dentro das empresas de desenvolvimentos de games, por isso dentre destes segmentos há o uso de algoritmos para agilizar e otimizar a criação destes designer, mas principalmente cuidando para que a qualidade e experiencia do usuário se mantenha a melhor possível.

E visto disto o terreno é a parte mais extensiva no seu desenvolvimento, alguns jogos precisam apresentar terrenos diferentes em cada nível para que o jogo não tornar-se repetitivo para os jogadores. A geração de Terreno processual (PTG) é uma técnica que pode otimizar muito deste trabalho usando algoritmos algoritmos pseudo-aleatórios para criar novos terrenos aleatoriamente por sistemas automatizados.

Atualmente, existem muitas técnicas eficazes que pode ser aplicado à geração processual de conteúdo de jogos. No entanto, os principais algoritmos são baseados na função de ruído. O algoritmo de diamante quadrado tem natureza fractal baseada em subdivisão recursiva. Funções de ruído são comparadas à geração de textura do terreno: Ruído de valor, Perlin, Simplex e Whorley Ruído. Os autores concluem que cada técnica apresenta uma ponto fraco ou fortr, por exemplo, o ruído simplex é muito difícil de implementar em um contexto de jogo, o Value Noise é muito simples, mas precisa de uma função de interpolação secundária baseada em movimento browniano fracionário para gerar um bom resultado e assim por diante.

Os jogos de plataforma 2D estão em ascensão devido aos recentes grandes sucessos em sistemas operacionais móveis, por exemplo, Angry Birds e Super Mario Run, cada um com mais de 100 milhões de downloads. Um jogo de plataforma é um gênero de videogame de jogo de ação onde o jogador controla um personagem ou avatar para pular entre plataformas suspensas e evite obstáculos.



Figura 1: Left: Game mobile Flappy Bird. Right: Game mobile Super Mario Run.

O objetivo deste trabalho é apresentar uma metodologia para geração procedural de terrenos para jogos em plataforma 2D. A abordagem desenvolvida foi baseada no uso de Markov modelos, especificamente as cadeias ocultas de Markov, para produzir terreno novo probabilisticamente. O designer do jogo pode definir o comportamento do terreno do jogo relatando uma matriz de transição, onde cada estado é um elemento terreno. Por fim, um método de filtragem foi aplicado para verificar a consistência e viabilidade do terreno gerado. Portanto, essa ferramenta pode ajudar o designer do jogo durante o jogo desenvolvimento e melhorar a qualidade geral do jogo.

## Imagens / Amostras

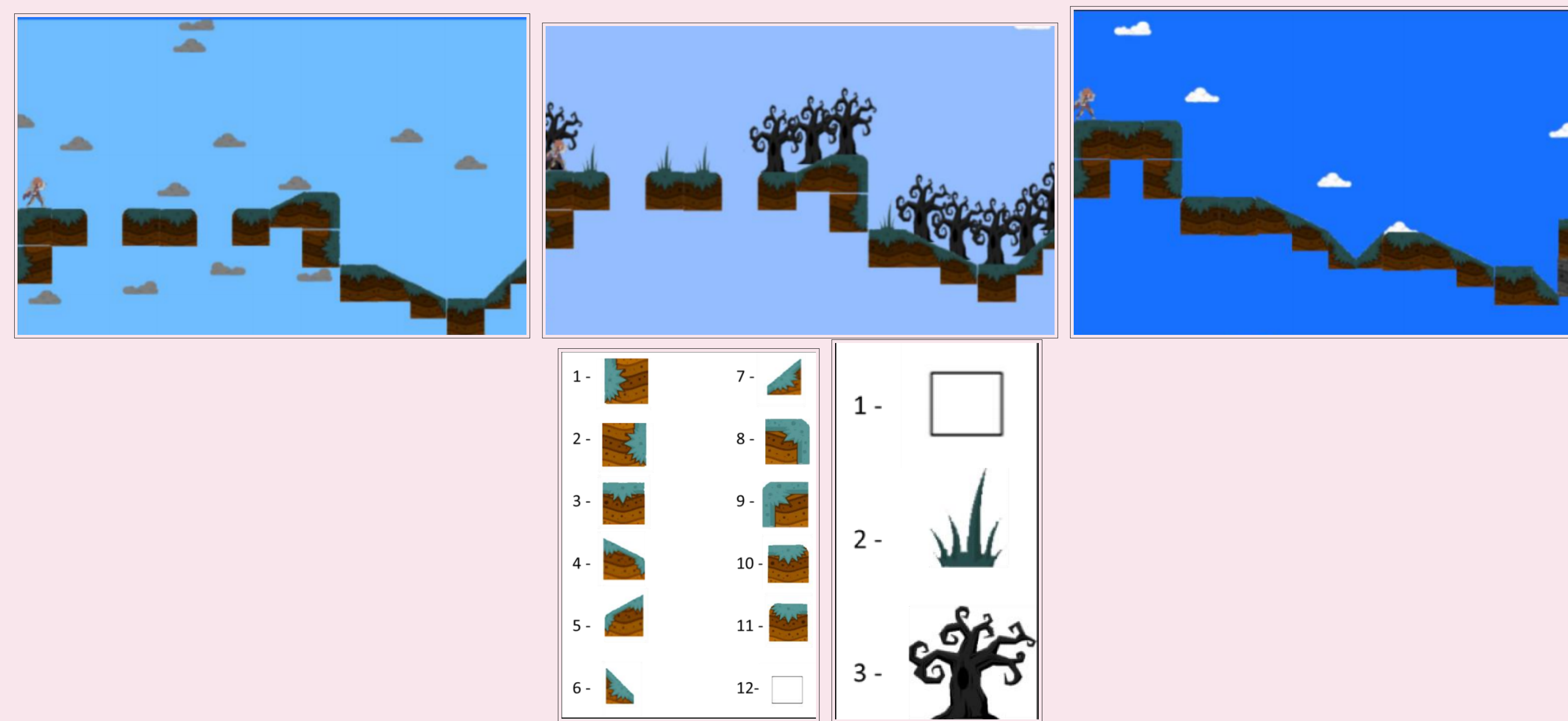


Figura 2: Topo: Imagens geradas com Markov chain, Abaixo: esquerda: Amostras de terreno, direita: Elementos decorativos

## Metodologia

A metodologia é baseada nas seguintes etapas:

1. Defina os estados do alfabeto  $S = s_0, s_1, \dots, s_n$ ;
2. Defina o vetor de probabilidades de inicialização;
3. Defina as probabilidades da matriz de transição;
4. Classifica o estado inicial conforme a 2;
5. Classifica o próximo estado conforme a 3;
6. Valida o estado gerado, se não for valido, repete a etapa 5;
7. Repita as etapas 5 e 6 até chegar ao final do nível.

As primeiras 3 (três) etapas dependem da aplicação, onde o designer do jogo define os componentes que compõem o terreno. O designer do jogo pode alterar os elementos do alfabeto na primeira etapa. Os elementos inseridos pelo designer são usados para gerar o terreno. A probabilidade de cada amostra aparecer é definido no segundo passo. O designer do jogo é livre para escolher o valores de probabilidade do vetor e da matriz A. A etapa 4 define o estado inicial. As etapas 5 e 7 compilam o terreno do jogo, seguindo a probabilidade do jogador Processo Markoviano observável. A etapa 6 é necessária para evitar sequências que escapam o comportamento esperado do terreno, por exemplo, distâncias que o personagem não pode pular depois de criar as plataformas para o nível atual no jogo, outro processo de Markov é executado para preencher o cenário com elementos decorativos. Esse processo é baseado em HMM. Uma matriz de emissão é definida com base no alfabeto dos estados. Cada componente gerado pelas etapas anteriores receber um elemento decorativo predeterminado pelo jogo desenhador e classificado em conformidade com HMM.

## Resultados

Vetor das probabilidades iniciais

1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	0	0	0	0	1	0	0	0

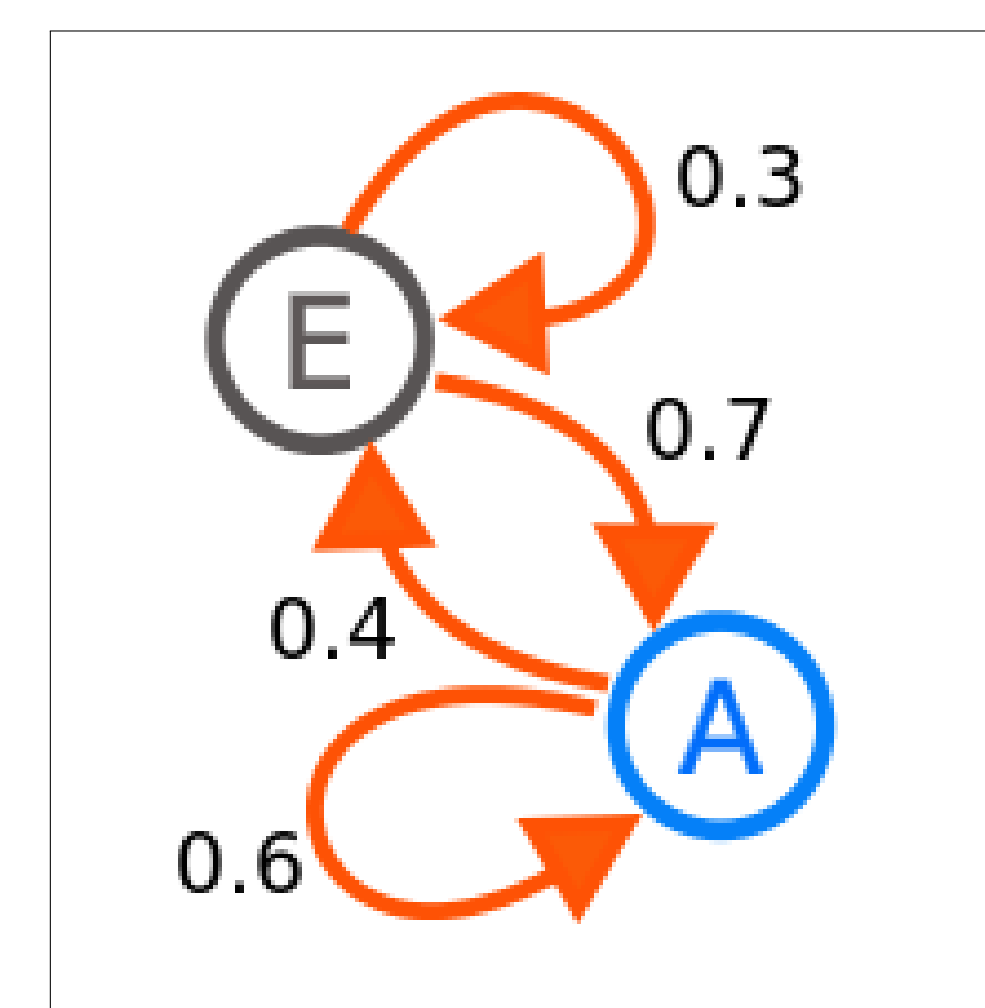
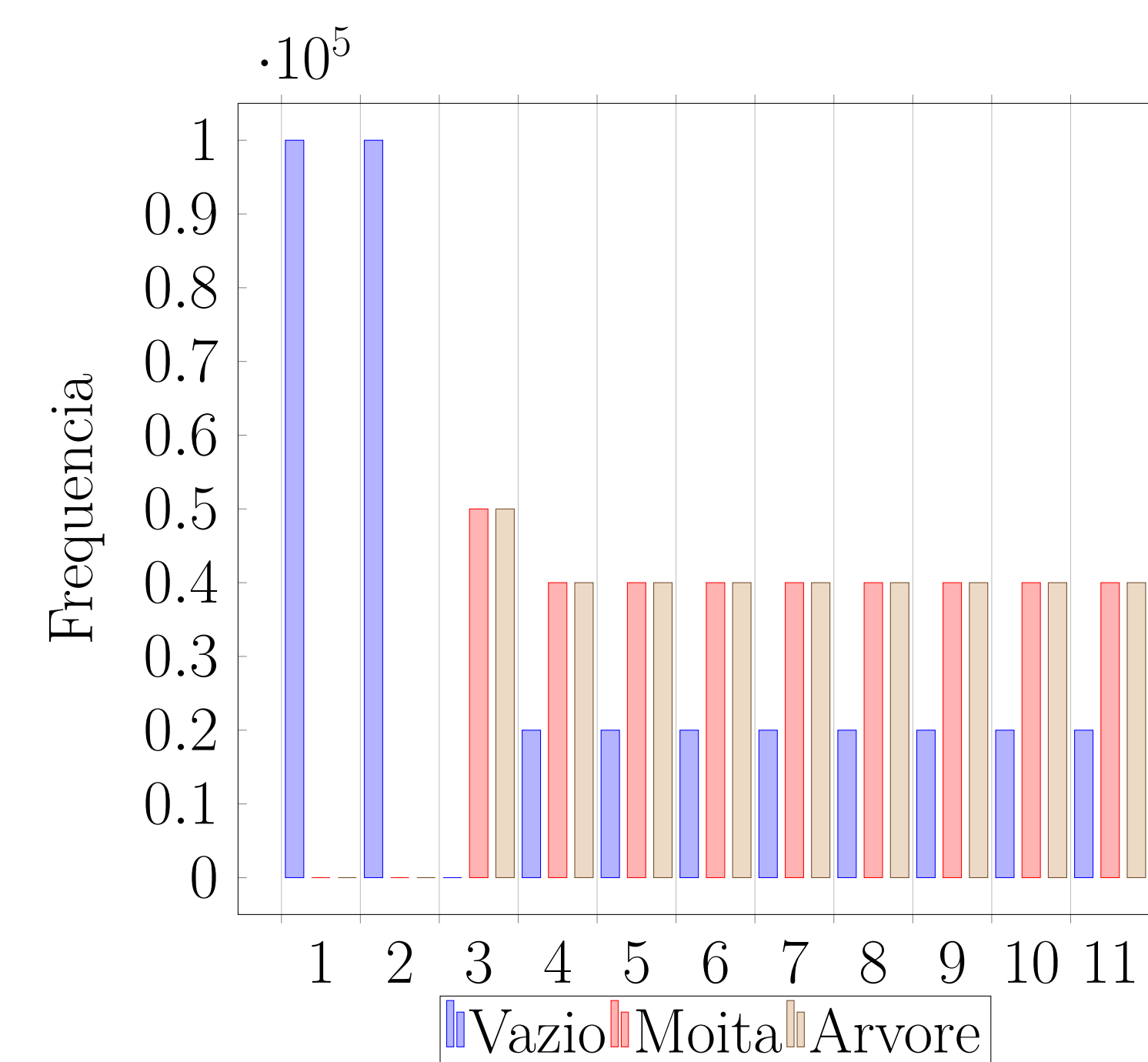


Figura 3: Uma cadeia de Markov simples de dois estados.



## Conclusão

Neste artigo, foi aplicado um modelo estocástico para a criação de terreno para jogos de plataforma 2D. O método PTG que baseia-se na probabilidade de um novo estado a ser criado a partir de o estado atual, onde cada estado representa um diferente segmento de terreno. A terra tende a se conectar probabilisticamente. Contudo, apenas uma aplicação dessa probabilidade o método pode inviabilizar o resultado em um jogo, pois pode criar novos estados que prejudicam a consistência do terreno. Para Para evitar terrenos inconsistentes, um filtro de regras foi aplicado. O modelo Markov oculto foi aplicado como um segundo passo para preencher o terreno criado com entidades decorativas do jogo como árvores, arbustos, pedras, etc. A metodologia mostra resultados aceitáveis, como pode ser visto em Figura 2 no topo. O método aplicado para filtrar os novos estados produzidos para manter a consistência são úteis e o uso isolado de cadeias de Markov foi insuficiente para obter resultado produtivo. Assim, devem ser feitos avanços para melhorar o uso de cadeias de Markov ocultas, porque apesar de obter resultados satisfatórios, erros ocorrem, por exemplo, a instância de elementos em lugares que não deveriam existir, como no caso da instância de um elemento decorativo em cima de um segmento de terreno vazio.

## Referências

- [1] T. J. Rose and A. G. Bakaoukas. "algorithms and approaches for procedural terrain generation - a brief review of current techniques." *8th International Conference on Games and Virtual Worlds for Serious Applications, Barcelona, 2*, 2016.
- [2] G. S. P. Miller. "The definition and rendering of terrain maps." *ACM SIGGRAPH Comput. Graph.*, 20:39–48, 1986.
- [3] T. Archer. "Procedurally generating terrain." *44th Midwest Instruction and Computing Symposium*, 2011.
- [4] S. Rabin. *Introduction to Game Development*, volume 1. Cengage Learning, 2012.
- [5] S. Snodgrass and S. Ontañón. "A hierarchical mdmc approach to 2d video game map generation." *Proceedings of The Eleventh AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 2015.
- [6] K. O. Stanley J. Togelius, G. N. Yannakakis and C. Browne. "Search-based procedural content generation: A taxonomy and survey." *IEEE Transactions on Computational Intelligence and AI in Games.*, 3:172–186, 2011.