

# Sistema de Simulação Urbana para Gerenciamento de Coleta de Resíduos Sólidos em Teresina

João Carlos Vieira Galvão Almeida  
Acadêmico de Engenharia de Software  
iCEV - Instituto de Ensino Superior  
Teresina, Piauí  
joao\_carlos.almeida@somosicev.com.com

Nicolas Rodrigues Ferreira de  
Carvalho  
Acadêmico de Engenharia de  
Software iCEV - Instituto de  
Ensino Superior  
Teresina, Piauí  
nicolas.carvalho@somosicev.com

**Abstract**— Este artigo apresenta o desenvolvimento de um sistema de simulação computacional voltado à gestão da coleta urbana de resíduos sólidos em Teresina, Piauí, Brasil, utilizando a linguagem Java. O sistema adota a modelagem baseada em agentes para representar o fluxo de resíduos por zonas geradoras, caminhões coletores, estações de transferência e aterros sanitários. A proposta visa otimizar a logística, reduzir tempos de espera e mitigar impactos ambientais. A versão mais recente do sistema incorpora uma arquitetura orientada a objetos robusta, algoritmos sofisticados de processamento de filas e parâmetros configuráveis, permitindo maior precisão nas simulações e análises operacionais.

**Index** — Gestão de Resíduos Sólidos, Simulação Urbana, Logística de Coleta, Java, Otimização..

## I. INTRODUÇÃO

O aumento da urbanização nos centros urbanos brasileiros tem elevado consideravelmente o volume de resíduos sólidos gerados diariamente, o que exige dos municípios sistemas de coleta mais eficazes, combinando eficiência operacional com sustentabilidade ambiental. Em cidades como Teresina, o planejamento da coleta deve considerar diversos fatores logísticos, como rotas, tempos de espera, capacidades dos veículos e características dos pontos de descarte.

Simulações computacionais, sobretudo aquelas baseadas em agentes, têm se mostrado uma ferramenta indispensável para apoiar gestores públicos na avaliação de cenários, identificação de gargalos e antecipação de demandas. Como destacado por Macal e North, essa abordagem permite modelar comportamentos individualizados de caminhões, zonas e pontos de descarte, promovendo uma visão dinâmica e detalhada do sistema como um todo [1].

A versão mais recente do sistema desenvolvido amplia sua capacidade analítica, incorporando novas entidades, algoritmos e parâmetros configuráveis, conforme será detalhado neste trabalho.

## II. FUNDAMENTAÇÃO TEORICA

A modelagem baseada em agentes é reconhecida pela sua capacidade de representar sistemas compostos por múltiplas entidades autônomas que interagem em tempo real em ambientes simulados. Essa técnica é amplamente empregada em estudos urbanos para compreender a dinâmica de sistemas complexos e propor estratégias que visem à melhoria de sua eficiência [1].

Estudos têm demonstrado que a aplicação dessa abordagem na gestão de resíduos sólidos resulta na redução do tempo de coleta, no melhor aproveitamento de recursos e na diminuição das emissões de poluentes [2]. O uso de linguagens orientadas a objetos, como Java, facilita a criação de arquiteturas modulares, permitindo a representação fiel das diversas camadas operacionais do sistema.

Além disso, o sistema utiliza conceitos fundamentais como estruturas de dados, teoria das filas, algoritmos de alocação e análise de desempenho, aspectos essenciais para simular a movimentação dos agentes e o acúmulo de resíduos nas zonas urbanas e estações de transferência [3].

## III. METODOLOGIA

O sistema foi implementado em Java, utilizando um design orientado a objetos que organiza a aplicação em pacotes e classes específicas.

### A. Arquitetura e Modelagem

As principais entidades modeladas no sistema são:

- **Zona:** Representa áreas geradoras de resíduos, com métodos para gerar e coletar lixo.
- **Caminhão Pequeno:** Responsável pela coleta nas zonas e transporte até as estações de transferência.
- **Caminhão Grande:** Realiza o transporte dos resíduos das estações até o aterro, com maior capacidade de carga.
- **Estação de Transferência:** Estrutura que gerencia filas de caminhões pequenos e grandes, controlando a movimentação de resíduos.
- **Simulador:** Classe central que orquestra o avanço do tempo e as interações entre as entidades.
- **Configuração:** Classe que permite parametrizar as simulações, como número de veículos, capacidades, tempos de viagem e limites operacionais.
- **Interface Gráfica:** Implementada com Java Swing, visualiza as zonas (círculos verdes), caminhões pequenos (retângulos azuis), caminhões grandes (retângulos laranjas) e estações (quadrados vermelhos).

Esse modelo segue as melhores práticas de desenvolvimento orientado a objetos, conforme preconizado pela documentação oficial da linguagem Java [4].

## B. Fluxo da Simulação

A simulação ocorre em ciclos temporais de 60 minutos, com as seguintes etapas principais:

1. As zonas geram resíduos conforme uma distribuição Gaussiana.
2. Caminhões pequenos são atribuídos às zonas com maior acúmulo de resíduos.
3. Os resíduos coletados são descarregados nas estações de transferência.
4. Quando a fila ou a quantidade acumulada atinge determinados limites, caminhões grandes são acionados para transportar os resíduos até o aterro.
5. Eventos são registrados e a interface gráfica é atualizada.

A lógica de promoção de caminhões pequenos para grandes é baseada no tempo de espera e no volume acumulado, seguindo critérios inspirados em estudos de simulação de sistemas logísticos complexos [5].

## C. Estruturas de Dados

O sistema utiliza diversas estruturas de dados para gerenciar as operações:

- **Filas (FIFO):** Gerenciam o fluxo de caminhões em estações de transferência.

- **Listas:** Armazenam o histórico de eventos para fins de análise e relatórios.
- **Pilhas:** Embora não utilizadas diretamente, estão previstas para futuras extensões, como rastreamento de estados temporários.
- **Arrays:** Estruturas fixas representam zonas e monitoram a quantidade de resíduos coletados.

A adoção dessas estruturas segue as orientações clássicas da teoria de estruturas de dados para sistemas orientados a eventos [6].

## IV. RESULTADOS E DISCUSSÃO

Durante a execução da simulação, observou-se que o sistema lida eficientemente com a variabilidade na geração de resíduos e a distribuição dos caminhões. A atualização do código introduziu parâmetros configuráveis que aprimoraram a análise operacional, tais como:

Parâmetro	Valor Atual
Número de caminhões pequenos	30
Capacidade dos caminhões pequenos	2, 4, 8, 10 toneladas
Capacidade do caminhão grande	20 toneladas
Limite de espera para caminhões pequenos	60 minutos
Limite de fila da estação	10 caminhões
Tempo máximo de rota	180 minutos
Tempos de viagem	Pico: 120-180 min / Fora de pico: 30-90 min

A complexidade algorítmica do sistema foi cuidadosamente avaliada para garantir sua eficiência. O ciclo de simulação apresenta complexidade  $O(N + M + S)$ , onde  $N$  corresponde ao número de caminhões pequenos,  $M$  ao número de caminhões enfileirados e  $S$  ao número de estações de transferência.

A análise demonstrou que o tempo médio de espera nas estações aumenta consideravelmente quando a remoção de resíduos é insuficiente, reforçando a necessidade de otimização dinâmica na alocação de caminhões grandes, conforme também evidenciado em outros estudos sobre logística de resíduos sólidos urbanos [3].

### A. Estimativa de Capacidade Operacional

A partir dos dados da simulação, obtiveram-se os seguintes resultados:

Item	Valor
Quantidade mínima de caminhões grandes	5 caminhões
Total de resíduos processados	119,45 toneladas
Tempo médio de viagem	97,82 minutos
Ativações de caminhões grandes	6 caminhões acionados
Eficiência por viagem	2,5 toneladas por viagem

Esses resultados indicam que cinco caminhões grandes são suficientes para garantir a eficiência da coleta, evitando acúmulo excessivo nas estações.

Além disso, a versão atual permite a geração de relatórios automáticos com estatísticas detalhadas, consolidando informações sobre resíduos coletados, viagens realizadas e eficiência operacional, conforme práticas recomendadas na modelagem de sistemas logísticos [7].

## V. CONCLUSÃO

O sistema de simulação desenvolvido representa de forma eficaz a dinâmica da coleta urbana de resíduos sólidos em Teresina. A atualização do sistema, com a introdução de novos parâmetros, algoritmos e estruturas, ampliou sua capacidade analítica e sua aplicabilidade prática.

Como proposta de evolução, planeja-se incluir:

- Algoritmos de otimização de rotas.
- Simulações multidiárias com análise de acúmulo progressivo.
- Ajustes de parâmetros em tempo real.
- Inclusão de restrições ambientais e econômicas.
- Implementação de roteamento dinâmico baseado em técnicas de aprendizado de máquina.

Esses aprimoramentos visam transformar o sistema em uma ferramenta ainda mais robusta e aderente às necessidades da gestão pública e da sustentabilidade urbana, conforme sugerem as tendências recentes em planejamento urbano sustentável [5].

## REFERENCIAS

[1] C. M. Macal and M. J. North, "Tutorial on agent-based modeling and simulation," *Journal of Simulation*, vol. 4, no. 3, pp. 151–162, 2010.

[2] D. Moya, C. Aldás, G. López, and P. Kaparaju, "Municipal solid waste management systems and greenhouse gas emissions: A life cycle perspective," *Journal of Cleaner Production*, vol. 144, pp. 415–429, 2017.

[3] A. P. da Silva, R. S. V. Nascimento, and R. M. Lima, "Simulação da logística de coleta de resíduos sólidos urbanos: um estudo de caso com abordagem baseada em agentes," *Revista Gestão e Sustentabilidade Ambiental*, vol. 8, no. 2, pp. 130–148, 2019.

[4] Oracle, "Java Platform Standard Edition Documentation," [Online]. Available: <https://docs.oracle.com/javase/>

[5] A. I. Ferraz and M. Santos, "Planejamento urbano sustentável e logística reversa," *Cadernos Metrópole*, vol. 17, no. 34, pp. 325–348, 2015.

[6] A. M. Law, *Simulation Modeling and Analysis*, 5th ed., New York, NY: McGraw-Hill, 2015.

[7] J. W. Forrester, *Urban Dynamics*. Cambridge, MA: MIT Press, 1969.

[8] Documentação Técnica do Sistema de Simulação de Coleta de Resíduos, 2025.