

# Testes de Sistemas Multiagente: Abordagens, Ferramentas e Desafios

**Matusalen Costa Alves<sup>1,2</sup>, Iallen Gábio de Sousa Santos<sup>1,2</sup>, Mayllon Veras da Silva<sup>1,2</sup>,  
Ricardo Moura Sekeff Budaruiche<sup>1,2</sup>, Wanderson de Vasconcelos Rodrigues da Silva<sup>1,2</sup>**

<sup>1</sup>Instituto Federal de Educação, Ciência e Tecnologia do Piauí (IFPI)  
Av. Rio dos Matos, S/N, Bairro Germano, Piripiri – PI

<sup>2</sup>Laboratório de Pesquisas e Estudos em Computação (LAPEC)  
Av. Rio dos Matos, S/N, Bairro Germano, Piripiri – PI

matusalencalves@gmail.com, iallen@ifpi.edu.br, veras@ifpi.edu.br

ricardo.sekeff@ifpi.edu.br, wanderson.vasconcelos@ifpi.edu.br

**Abstract.** This paper presents a systematic mapping of the literature on testing in Multi-Agent Systems, focusing on two main dimensions: (i) software engineering and test automation approaches, and (ii) model-based testing strategies and frameworks. Thirteen selected papers were qualitatively analyzed, revealing recent advances such as the use of formal methods (e.g., Colored Petri Nets), platforms like JADE, and AI-driven testing solutions with Large Language Models. Despite the progress, gaps remain regarding non-functional testing, methodological standardization, and empirical validation. The study provides an overview of current trends and highlights directions for future research.

**Resumo.** Este artigo apresenta um mapeamento sistemático da literatura sobre testes em Sistemas Multiagente, com foco em duas vertentes principais: (i) engenharia de software e automação de testes, e (ii) abordagens baseadas em modelos e frameworks específicos. Foram analisados qualitativamente 13 artigos selecionados, revelando avanços como o uso de métodos formais (ex.: Redes de Petri Coloridas), plataformas como JADE e soluções baseadas em inteligência artificial, incluindo Modelos de Linguagem de Larga Escala. Apesar dos avanços, ainda existem lacunas quanto à padronização metodológica, testes não funcionais e validações empíricas. O estudo fornece uma visão crítica das tendências atuais e sugere caminhos para pesquisas futuras.

## 1. Introdução

Os Sistemas Multiagente são compostos por agentes autônomos que interagem entre si em ambientes dinâmicos e distribuídos. Esses agentes são caracterizados por propriedades como proatividade, adaptabilidade e comportamento não determinístico, o que aumenta significativamente a complexidade do desenvolvimento e do teste desses sistemas. As estratégias tradicionais de verificação e validação, projetadas para sistemas centralizados e previsíveis, mostram-se insuficientes para lidar com tais particularidades.

Nesse contexto, diferentes formas de teste têm sido propostas para lidar com a complexidade dos Sistemas Multiagente. A literatura recente apresenta desde abordagens baseadas em agentes inteligentes que automatizam a geração e execução de testes,

até técnicas formais apoiadas por frameworks especializados. Arquiteturas como aquelas baseadas em crenças, desejos e intenções, deliberativas e reativas, são frequentemente empregadas, muitas vezes integradas a plataformas como o *Java Agent Development Framework*, com o objetivo de tornar o processo de verificação mais robusto e adaptado às dinâmicas desses sistemas.

Apesar destes avanços recentes, ainda há lacunas na literatura. Testes contínuos e de regressão são pouco abordados, mesmo sendo essenciais em sistemas que evoluem ao longo do tempo. Além disso, questões não funcionais como desempenho, escalabilidade e segurança recebem atenção limitada, comprometendo a confiabilidade dos sistemas em aplicações reais.

Diante disso, o presente trabalho realiza um mapeamento sistemático da literatura com o objetivo de identificar, analisar e categorizar as abordagens existentes voltadas ao teste de Sistemas Multiagente. A investigação concentra-se nas arquiteturas, técnicas e ferramentas utilizadas, bem como nas lacunas e oportunidades que se apresentam no estado atual da pesquisa.

O restante deste trabalho está organizado da seguinte forma: a Seção 2 apresenta a fundamentação teórica sobre Sistemas Multiagente e testes de software; a Seção 3 detalha a metodologia adotada neste mapeamento sistemático; a Seção 4 descreve os resultados qualitativos obtidos; a Seção 5 discute criticamente os achados à luz das questões de pesquisa; por fim, a Seção 6 apresenta as considerações finais e sugestões para trabalhos futuros.

## 2. Fundamentação Teórica

Esta Seção apresenta os conceitos essenciais para a compreensão dos desafios envolvidos no teste de Sistemas Multiagente, destacando suas particularidades arquiteturais, os tipos de teste aplicáveis e as ferramentas mais recorrentes na literatura.

### 2.1. Sistemas Multiagente

Sistemas Multiagente (SMA) são compostos por agentes autônomos que interagem em ambientes distribuídos para atingir metas individuais ou coletivas [Elkholy et al. 2020]. Tais agentes possuem características como autonomia, pró-atividade e adaptabilidade, resultando em comportamentos dinâmicos e não determinísticos [Machado et al. 2025]. Arquiteturas como agentes reativos, deliberativos e BDI (*Belief-Desire-Intention*) influenciam diretamente a forma de tomada de decisão e cooperação entre os agentes [Gonçalves et al. 2022].

### 2.2. Testes de Software

O teste de software em Sistemas Multiagente é desafiador devido à descentralização, à autonomia dos agentes e à complexidade das interações sociais. Técnicas tradicionais tendem a ser insuficientes nesse contexto [Kalache et al. 2023]. Diversas abordagens têm sido exploradas, como testes formais, simulações baseadas em modelos, frameworks específicos e o uso de agentes para automatizar o próprio processo de teste, prática conhecida como *Agent-Based Software Testing*. Essa última permite testes nos níveis unitário, social, de integração e aceitação, embora ainda haja limitações quanto à cobertura de requisitos não funcionais, como desempenho e segurança [Schwabl et al. 2024].

### 2.3. Ferramentas e Abordagens Formais

Ferramentas como JADE (*Java Agent Development Framework*) são amplamente empregadas por oferecerem suporte à comunicação e coordenação entre agentes, facilitando o desenvolvimento e teste em ambientes distribuídos [Kalache et al. 2023]. Complementarmente, técnicas formais como Redes de Petri Coloridas (CPN) têm sido aplicadas para modelar interações sociais e organizacionais em SMA, permitindo a geração automatizada de casos de teste com maior precisão e cobertura [Machado et al. 2025, Gonçalves et al. 2022].

## 3. Metodologia de Pesquisa

Segundo Gil (2008), a pesquisa exploratória visa proporcionar maior familiaridade com temas ainda pouco investigados, enquanto a abordagem qualitativa busca interpretar os fenômenos com base na análise de dados não numéricos. Alinhado a essa perspectiva, este estudo configura-se como uma pesquisa exploratória e qualitativa, conduzida por meio de um mapeamento sistemático da literatura.

O objetivo é identificar e categorizar as abordagens existentes para testes em Sistemas Multiagente (SMA), destacando tendências, lacunas e oportunidades de pesquisa. Tal escolha metodológica justifica-se pela necessidade de consolidar o conhecimento disperso na área, fornecendo uma visão estruturada e crítica do estado da arte.

Para assegurar rigor e reproduzibilidade, foram seguidas as etapas descritas por Gil (2008), que incluem: formulação de questões de pesquisa, seleção de bases acadêmicas relevantes, definição de critérios de inclusão e exclusão, e categorização temática dos estudos selecionados.

### 3.1. Questões de Pesquisa

A definição das questões norteadoras foi essencial para delimitar o escopo da investigação e orientar todo o processo de busca, seleção e análise dos estudos. As seguintes perguntas guiaram o desenvolvimento da pesquisa:

- **QP1:** Quais arquiteturas e abordagens de teste têm sido utilizadas em soluções para SMA?
- **QP2:** Que tipos e níveis de teste são abordados nos estudos recentes sobre SMA?
- **QP3:** Quais são as principais lacunas e oportunidades de pesquisa identificadas na literatura?

### 3.2. Bases de Dados Consultadas

Para garantir abrangência e relevância na recuperação dos trabalhos, foram selecionadas bases de dados amplamente reconhecidas nas áreas de ciência da computação e engenharia de software, como IEEE Xplore, ACM Digital Library, Scopus, ScienceDirect e Portal de Periódicos da CAPES.

### 3.3. Critérios e Estratégias de Busca

A busca pelos artigos foi realizada por meio de combinações de palavras-chave relacionadas ao tema central, utilizando operadores booleanos (*AND*, *OR*) para ampliar a cobertura. Para isto, os termos empregados incluem: *multi-agent systems*, *software testing*, *agent-based testing*, *formal verification* e *framework*. Como critérios de inclusão, foram considerados apenas artigos:

1. publicados entre 2021 e 2025;
2. redigidos em língua inglesa;
3. revisados por pares e disponíveis na íntegra;
4. que abordassem explicitamente estratégias de teste aplicadas a SMA.
5. não duplicados, incompletos ou fora do escopo foram descartados.

### **3.4. Processo de Seleção e Refinamento**

A seleção dos estudos foi realizada em duas etapas. Na primeira, 59 artigos foram recuperados com base nas estratégias de busca aplicadas. Títulos, resumos e palavras-chave foram analisados segundo critérios previamente definidos. Foram incluídos apenas artigos publicados entre 2021 e 2025, em inglês, disponíveis na íntegra, revisados por pares e que tratassesem de testes aplicados a Sistemas Multiagente.

Excluíram-se estudos duplicados, com foco puramente teórico, sem aplicação em Sistemas Multiagente, resumos sem texto completo, revisões sem análise original e trabalhos centrados em outros tipos de sistemas. Ao final dessa triagem inicial, 25 artigos foram mantidos para leitura aprofundada.

Na segunda etapa, os textos completos foram analisados quanto à sua relevância para as questões de pesquisa. Ao final, 13 estudos compuseram o corpus final, por apresentarem contribuições significativas, alinhamento metodológico e aderência ao escopo do mapeamento.

### **3.5. Extração e Classificação dos Dados**

Após a seleção dos 13 artigos, realizou-se a extração sistemática dos dados. Cada publicação teve seus metadados registrados em formato BibTeX, além da coleta de informações qualitativas sobre técnicas de teste, níveis abordados (como teste unitário, de integração e funcional), arquiteturas de agentes, ferramentas empregadas e principais contribuições para a área.

Com base nesses dados, os estudos foram organizados em duas categorias: (i) Engenharia de Software e Automação de Testes em Sistemas Multiagente; e (ii) Testes Baseados em Modelo e Frameworks. Essa classificação orientou a análise qualitativa e permitiu identificar abordagens recorrentes, tendências e lacunas da literatura.

## **4. Resultados Qualitativos**

Conforme discutido na Seção 3, os artigos foram buscados, selecionados e agrupados em duas categorias relacionadas ao objetivo de estudo deste trabalho.

### **4.1. Engenharia de Software e Automação de Testes em SMA**

Nesta categoria, foram agrupados estudos que propõem estratégias de automação, melhoria de desempenho e robustez dos testes aplicados a SMA. Essas abordagens integram técnicas modernas, como aprendizado de máquina, modelagem baseada em agentes e simulações complexas, voltadas para o ciclo de vida do desenvolvimento de software.

O trabalho de [Yang et al. 2024] explora o uso de modelos de linguagem de larga escala (LLMs) para automatizar testes white-box em compiladores, evidenciando como técnicas modernas de inteligência artificial podem ser aplicadas à análise e verificação

de código. Na mesma linha, [Khoei et al. 2025] apresenta um SMA orientado por LLMs capaz de tomar decisões automatizadas em pipelines de liberação de software automotivo, contribuindo para a gestão de riscos em sistemas sensíveis.

Em contextos distribuídos e críticos, [O’Neill and Soh 2022] discute mecanismos de transferência de inteligência entre agentes para elevar a tolerância a falhas em sistemas IoT heterogêneos, reforçando a confiabilidade operacional dos testes em ambientes dinâmicos. Complementando esse enfoque em resiliência, [Palmieri et al. 2023] propõe uma arquitetura baseada em co-simulação de gêmeos digitais aplicados a SMA ciberfísicos, demonstrando como a decomposição modular e a simulação distribuída possibilitam testes realistas de desempenho e segurança, como no caso de pelotões de veículos em ambientes adversos.

A perspectiva de automação de processos de desenvolvimento também está presente em [Medvedev and Aksyonov 2021], que desenvolve um modelo de simulação multiagente para avaliar a qualidade de pipelines CI/CD, permitindo o teste de configurações e cenários alternativos com base em lógica de filas e comportamento de agentes.

Por fim, [Kumazawa et al. 2021] propõe estratégias otimizadas de exploração de estados para verificação formal via model checking, utilizando Otimização por Colônia de Formigas. Essa técnica, inspirada no comportamento de agentes sociais, visa reduzir o custo computacional dos testes, mantendo uma cobertura eficiente do espaço de estados.

Esses estudos demonstram uma tendência crescente em integrar SMA com técnicas modernas de automação, simulação e inteligência artificial, resultando em soluções que ampliam a confiabilidade, a escalabilidade e a eficácia dos testes de software em sistemas complexos e distribuídos.

#### 4.2. Testes Baseados em Modelo e Frameworks para SMA

Nesta categoria, estão reunidas propostas que utilizam representações formais, abordagens baseadas em modelos e plataformas especializadas para estruturar e automatizar o processo de teste em SMA. Tais técnicas são especialmente relevantes para validar comportamentos emergentes, interações sociais e conformidade com requisitos organizacionais.

Dentre as abordagens formais, destacam-se os estudos de [Gonçalves et al. 2022] e [Machado et al. 2025], que utilizam CPN para representar e testar SMA baseados no modelo organizacional *Moise+*. Essa representação gráfica permite simular e verificar propriedades complexas de interação entre agentes, além de viabilizar a geração sistemática de casos de teste a partir da estrutura formalizada do sistema.

O framework JADE é explorado por [Kalache et al. 2023], que propõe uma arquitetura de testes em dois níveis: testes unitários no nível individual dos agentes e testes de interação no nível coletivo. A proposta oferece ferramentas específicas para controle, execução e monitoramento de testes, com foco na modularidade e reutilização dos componentes testados.

No campo dos testes não tradicionais, o estudo de [Zhang et al. 2024] aplica a técnica de *Metamorphic Testing* na validação de algoritmos de busca de caminhos em ambientes multiagente, como o *Multi-Agent Path Finding*. Essa técnica se destaca por sua capacidade de detectar falhas em sistemas onde oráculos tradicionais não são facilmente

definidos, ao verificar relações metamórficas entre entradas e saídas esperadas.

Complementarmente, a proposta de [Kissoum and Redjimi 2022] apresenta uma abordagem multinível de teste para SMA, estruturada a partir do paradigma de redes de referência. A técnica abrange desde a verificação de componentes isolados até a avaliação de aspectos globais do sistema, contribuindo para uma cobertura abrangente do comportamento multiagente em diferentes camadas de abstração.

As contribuições desta categoria ressaltam o papel das abordagens formais e baseadas em modelo como ferramentas fundamentais para garantir a confiabilidade, a rastreabilidade e a reproduzibilidade dos testes em sistemas distribuídos e de alta complexidade.

## 5. Discussão dos Resultados

A análise qualitativa dos 13 artigos selecionados permitiu responder de forma estruturada às três questões norteadoras definidas neste estudo. A seguir, discutem-se os principais achados à luz dessas perguntas.

### QP1: Quais arquiteturas e abordagens de teste têm sido utilizadas em soluções para SMA?

As abordagens de teste em SMA identificadas na literatura recente estão distribuídas entre soluções centradas na engenharia de software e outras baseadas em formalismos e frameworks específicos. No primeiro grupo, destacam-se as propostas que integram inteligência artificial ao processo de teste, como o uso de LLMs para geração automática de casos de teste em contextos complexos e distribuídos, conforme exemplificado nos estudos de [Yang et al. 2024] e [Khoei et al. 2025]. Abordagens orientadas a agentes com inteligência transferível também surgem como forma de aumentar a robustez dos testes em ambientes heterogêneos, como em [O'Neill and Soh 2022].

No segundo grupo, observou-se o uso de arquiteturas formais e frameworks especializados, como a modelagem por CPN integrada ao modelo organizacional Moise+ [Gonçalves et al. 2022, Machado et al. 2025], além de estruturas consolidadas como a plataforma JADE [Kalaché et al. 2023]. Esses frameworks fornecem ambientes controlados e estruturados para testes, possibilitando a reprodução de cenários complexos e a verificação formal das interações entre agentes.

### QP2: Que tipos e níveis de teste são abordados nos estudos recentes sobre SMA?

Os estudos analisados abrangem uma variedade de tipos e níveis de teste. No nível mais elementar, identificaram-se testes unitários voltados à lógica interna dos agentes, como proposto em [Kalaché et al. 2023]. Já em níveis mais altos, foram destacados testes de integração entre agentes, testes funcionais, estruturais e testes baseados em comportamentos sociais (ex.: papéis e normas organizacionais).

Além disso, estratégias como o *Metamorphic Testing* [Zhang et al. 2024] e abordagens multinível baseadas em modelos de referência [Kissoum and Redjimi 2022] demonstram um esforço para ampliar a profundidade e a cobertura dos testes. Também foi observada a aplicação de simulações multiagente para análise de pipelines de CI/CD [Medvedev and Aksyonov 2021], e o uso de digital twins para validação em tempo real de cenários físicos, como sistemas de veículos autônomos [Palmieri et al. 2023].

No entanto, testes de desempenho, segurança, escalabilidade ou regressão ainda aparecem com baixa recorrência, o que limita a avaliação dos sistemas sob condições realistas de operação.

### **QP3: Quais são as principais lacunas e oportunidades de pesquisa identificadas na literatura?**

Apesar da diversidade de abordagens, a literatura recente apresenta algumas lacunas notáveis. Em primeiro lugar, há baixa padronização dos métodos de avaliação: muitos artigos propõem frameworks ou técnicas, mas poucos realizam validações empíricas robustas, como estudos comparativos, testes com usuários finais ou experimentos em ambientes produtivos. Essa carência compromete a generalização dos resultados e a adoção das soluções propostas.

Outro ponto crítico é a cobertura limitada de testes não funcionais. A maioria dos trabalhos se concentra em aspectos funcionais ou de comportamento, negligenciando elementos fundamentais como desempenho, confiabilidade, resiliência e testabilidade em sistemas com agentes adaptativos ou com aprendizado.

Por fim, observa-se uma oportunidade relevante de pesquisa na integração entre abordagens formais e automatizadas. Trabalhos como os de [Machado et al. 2025] e [Yang et al. 2024] sinalizam caminhos promissores nesse sentido, mas ainda são incipientes. Há espaço para desenvolver ferramentas híbridas que combinem o rigor dos métodos formais com a flexibilidade das soluções baseadas em IA, com foco em escalabilidade, reusabilidade e adaptação a diferentes domínios de aplicação.

## **6. Considerações Finais**

Este artigo apresentou um mapeamento sistemático da literatura sobre testes em SMA, com foco em duas vertentes centrais: (i) engenharia de software e automação de testes, e (ii) abordagens baseadas em modelos e frameworks específicos para SMA. A análise qualitativa dos 13 artigos selecionados permitiu identificar avanços relevantes, como a aplicação de técnicas formais, a exemplo das CPN, o uso de plataformas consolidadas como JADE e a incorporação de soluções recentes baseadas em inteligência artificial, com destaque para modelos de LLMs.

Apesar desses avanços, o estudo evidenciou lacunas que ainda dificultam a consolidação de práticas eficazes de teste em SMA. Entre elas, destacam-se a ausência de padronização metodológica, a limitada cobertura de testes não funcionais e a escassez de validações empíricas em ambientes reais. Aspectos como a testabilidade de sistemas adaptativos, a avaliação de regressões e a integração contínua permanecem pouco explorados na literatura recente.

Ainda assim, o panorama é promissor. Observa-se uma tendência crescente de convergência entre abordagens formais tradicionais e tecnologias emergentes, o que sinaliza o potencial para soluções mais robustas, adaptativas e escaláveis. Como direções futuras, recomenda-se a ampliação da base de estudos analisados, a condução de comparações sistemáticas entre diferentes estratégias de teste e a realização de experimentos em contextos reais, capazes de demonstrar a viabilidade e a eficácia prática das abordagens propostas.

Espera-se que os achados deste trabalho sirvam como subsídio para pesquisadores

e profissionais da área, oferecendo uma visão crítica e fundamentada sobre o estado atual dos testes em SMA, bem como orientações claras para futuras investigações e aprimoramentos na área.

## Referências

- Elkholy, W., El-Menshawy, M., Bentahar, J., Elqortobi, M., Laarej, A., and Dssouli, R. (2020). Model checking intelligent avionics systems for test cases generation using multi-agent systems. *Expert Systems with Applications*, 156:113458.
- Gonçalves, E. M. N., Machado, R. A., Rodrigues, B. C., and Adamatti, D. (2022). Cpn4m: Testing multi-agent systems under organizational model moise+ using colored petri nets. *Applied Sciences*, 12(12):5857.
- Kalache, A., Badri, M., Mokhati, F., and Babahenini, M. C. (2023). A testing framework for jade agent-based software. *Multiagent and Grid Systems*, 19(1):61–98.
- Khoei, A. G., Yu, Y., Feldt, R., Freimanis, A., Rhodin, P. A., and Parthasarathy, D. (2025). Gonogo: An efficient llm-based multi-agent system for streamlining automotive software release decision-making. In *Testing Software and Systems*, pages 30–45. Springer Nature Switzerland.
- Kissoum, Y. and Redjimi, M. (2022). Multi-level testing approach for multi-agent systems. *International Journal of Organizational and Collective Intelligence*, 12(1):1–23.
- Kumazawa, T., Takimoto, M., and Kambayashi, Y. (2021). Exploration strategies for model checking with ant colony optimization. In Nguyen, N. T., Iliadis, L., Maglogiannis, I., and Trawiński, B., editors, *Computational Collective Intelligence*, pages 264–276. Springer International Publishing.
- Machado, R. A., da Silva Zelindro Cardoso, A., Farias, G. P., Gonçalves, E. M. N., and Adamatti, D. F. (2025). A formal testing method for multi-agent systems using colored petri nets. *Autonomous Agents and Multi-Agent Systems*, 39(1):10.
- Medvedev, D. and Aksyonov, K. (2021). The development of a simulation model for assessing the ci/cd pipeline quality in the development of information systems based on a multi-agent approach. *MATEC Web of Conferences*, 346:03095.
- O'Neill, V. and Soh, B. (2022). Improving fault tolerance and reliability of heterogeneous multi-agent iot systems using intelligence transfer. *Electronics*, 11(17):2724.
- Palmieri, M., Quadri, C., Fagiolini, A., and Bernardeschi, C. (2023). Co-simulated digital twin on the network edge: A vehicle platoon. *Computer Communications*, 212:35–47.
- Schwabl, P., Haim, M., and Unkel, J. (2024). Aligning agent-based testing (abt) with the experimental research paradigm: A literature review and best practices. *Journal of Computational Social Science*, 7(2):1625–1644.
- Yang, C., Deng, Y., Lu, R., Yao, J., Liu, J., Jabbarvand, R., and Zhang, L. (2024). Whitefox: White-box compiler fuzzing empowered by large language models. *Proceedings of the ACM on Programming Languages*, 8(OOPSLA2):296.
- Zhang, X.-Y., Liu, Y., Arcaini, P., Jiang, M., and Zheng, Z. (2024). Met-mapf: A metamorphic testing approach for multi-agent path finding algorithms. *ACM Transactions on Software Engineering and Methodology*, 33(8):198.