

PEDRO DE ALMEIDA SACRAMENTO

**UM MODELO DE SISTEMAS MULTIAGENTES PARA AUXÍLIO NA FASE
CONCEITUAL DE PROJETOS DE EDIFÍCIOS**

Dissertação apresentada à Universidade Federal de Viçosa, como parte das exigências do Programa de Pós-Graduação em Ciência da Computação, para obtenção do título de *Magister Scientiae*.

VIÇOSA
MINAS GERAIS – BRASIL
2013

**Ficha catalográfica preparada pela Seção de Catalogação e
Classificação da Biblioteca Central da UFV**

T

S123m
2013

Sacramento, Pedro de Almeida, 1986-

Um modelo de sistemas multiagentes para auxílio na fase
conceitual de projetos de edifícios / Pedro de Almeida
Sacramento. – Viçosa, MG, 2013.
x, 58f. : il. ; 29cm.

Inclui apêndice.

Orientador: Alcione de Paiva Oliveira.

Dissertação (mestrado) - Universidade Federal de Viçosa.

Referências bibliográficas: f. 53-57.

1. Inteligência artificial. 2. Edifícios - Projetos de
construção - Simulação por computador. 3. Agentes
inteligentes (Software). 4. Engenharia de software.
5. Ontologia. I. Universidade Federal de Viçosa. Departamento
de Informática. Programa de Pós-Graduação em Ciência da
Computação. II. Título.

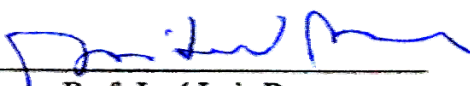
CDD 22. ed. 006.3

PEDRO DE ALMEIDA SACRAMENTO

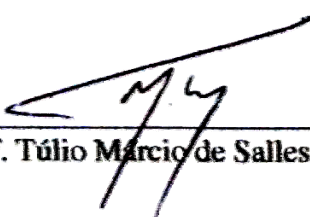
**UM MODELO DE SISTEMAS MULTIAGENTES PARA AUXÍLIO NA FASE
CONCEITUAL DE PROJETOS DE EDIFÍCIOS**

Dissertação apresentada à Universidade
Federal de Viçosa, como parte das
exigências do Programa de Pós-Graduação
em Ciência da Computação, para obtenção
do título de *Magister Scientiae*.

APROVADA: 06 de março de 2013



Prof. José Luis Braga
(Co-orientador)



Prof. Túlio Marcio de Salles Tibúrcio



Prof. Alcione de Paiva Oliveira
(Orientador)

Dedico

A Jesus,

*pois “O SENHOR é a minha luz e a minha salvação; a quem temerei? O SENHOR é a
força da minha vida; de quem me recearei? ” (Salmos 27:1).*

AGRADECIMENTOS

A Jesus, por ter me concedido vida até aqui e por poder contemplar mais esta bênção na minha família;

Ao Professor Alcione de Paiva Oliveira pela grande paciência, incentivo e bom humor durante toda a realização do trabalho;

Ao coorientador Professor José Luis Braga, pela direção, apoio e pelas incontáveis palavras de incentivo;

Ao Professor Túlio Tibúrcio, pelo auxílio e disposição em contribuir para minha formação acadêmica desde o meu tempo de graduando;

Aos professores Benício Ramalho, Carlos Goulart, Fábio Cerqueira, Kléos Lenz, Marcelo Picanço, Marcus Alvim e Ricardo “Cacau” pela direção, incentivo e apoio, especialmente quando ingressei no mestrado;

À Divisão de Projetos e Controle do Espaço Físico da UFV, que na pessoa da arquiteta Débora Gomide, graciosamente nos forneceu o modelo virtual de edifício necessário para a realização do estudo de caso presente neste trabalho;

Ao pessoal do grupo CoSy da Universidade de Bremen pelos materiais que concederam graciosamente e que auxiliaram muito no direcionamento da pesquisa;

Ao pessoal da Secretaria da Pós do DPI pela competência no que fazem, especialmente ao Altino pela bondade e paciência com relação às minhas dúvidas, principalmente no início e no final do semestre;

Aos demais professores do DPI, que sempre contribuíram para nosso aprendizado e desenvolvimento profissional;

À CAPES e à Universidade Aberta do Brasil, pela concessão da bolsa de pesquisa;

Aos professores Frederico Passos e a toda a equipe da CEAD por toda a amizade, companhia e auxílio;

Aos meus pais, Almir Alves do Sacramento e Enedina Maria de Almeida Sacramento, exemplos em tudo para mim;

Aos meus irmãos Paulo, Débora, Lucas, Tiago e Marcos Felipe, por todo o carinho;

Aos meus amigos Mateuzão, Diogo e Pedro Ivan, que acompanham essa história praticamente desde o tempo em que eu criava cabras;

Aos amigos e colegas de curso, pelo aprendizado e companhia durante esse tempo de mestrado, em especial àqueles que me acompanharam desde o primeiro dia de aula: Eltinho, Lucas Vegi, Wilker, Chupeta (*in memoriam*) e Patrícia Monnerat;

Ao pessoal da Segunda Igreja Batista em Viçosa, sempre disposto a dividir cargas em toda a minha caminhada até aqui;

A toda a equipe da FUNARBE, pelo carinho e companheirismo neste tempo de caminhada juntos;

À minha princesa, Vanessa, por todo o amor, apoio e disposição em caminhar ao meu lado;

Ao meu filho Israel, que desde antes de nascer já traz alegria ao nosso lar;

A todas as pessoas que me acompanharam até aqui.

BIOGRAFIA

PEDRO DE ALMEIDA SACRAMENTO, filho de Almir Alves do Sacramento e Enedina Maria de Almeida Sacramento, nasceu em 19 de abril de 1986, no Rio de Janeiro, RJ.

A paixão pela área de Informática começou cedo, sob influência e incentivo de seu pai, que também atua na área. Em 2003 ingressou no curso Técnico em Informática na Escola Técnica de Viçosa (ETEV), o que lhe permitiu estagiar como programador desde o referido ano até 2005 na Coordenadoria de Educação Aberta e à Distância da Universidade Federal de Viçosa (CEAD - UFV).

Em 2004 concluiu o curso Técnico em Informática e ingressou no curso de Arquitetura e Urbanismo da UFV, graduando-se em janeiro de 2010.

Atuou como programador e designer gráfico na Studium Comunicação Integrada Ltda (Viçosa, MG), em 2007. A partir de 2008 começou a pesquisar a respeito de novos recursos para ferramentas CAD e apresentou o projeto “Colabora – uma ferramenta online para CAD 2D colaborativo” na II Feira do APL de TI de Viçosa, evento paralelo ao 6º Congresso Brasileiro de Agroinformática da UFV.

Em agosto de 2010 iniciou o Mestrado do Programa de Pós-Graduação no Departamento de Informática da UFV, sendo bolsista CAPES-REUNI e bolsista da Universidade Aberta do Brasil (UAB), além de tutor de ensino à distância em trabalho conjunto com a CEAD – UFV no desenvolvimento de ferramentas de ensino com Novas Tecnologias de Informação e Comunicação (nTICs).

De 2004 a 2012 lecionou as disciplinas “Aplicativos Web”, “Programação para Internet” e “Introdução à Computação” na ETEV. Em 2012 iniciou o trabalho na Fundação Arthur Bernardes da UFV (FUNARBE) como Assistente em Desenvolvimento de Sistemas. Submeteu-se à defesa da dissertação do Mestrado em Ciência da Computação em março de 2013.

SUMÁRIO

RESUMO	ix
ABSTRACT	x
1. INTRODUÇÃO	1
1.1. O PROBLEMA E SUA IMPORTÂNCIA	1
1.2. HIPÓTESE.....	3
1.3. OBJETIVOS	3
1.3.1. OBJETIVO GERAL	3
1.3.2. OBJETIVOS ESPECÍFICOS.....	3
1.4. ORGANIZAÇÃO DA DISSERTAÇÃO	4
2. METODOLOGIA	5
2.1. MATERIAIS E MÉTODOS	6
3. FUNDAMENTAÇÃO TEÓRICA	8
3.1. PROJETO DE EDIFÍCIOS	8
3.1.1. BUILDING INFORMATION MODELING (BIM).....	9
3.2. SISTEMAS MULTIAGENTES	11
3.2.1. AGENTES.....	11
3.2.2. SISTEMAS MULTIAGENTES.....	14
3.2.3. SISTEMAS MULTIAGENTES NA CONSTRUÇÃO CIVIL.....	15
3.3. ONTOLOGIAS.....	16
3.3.1. ONTOLOGIA SUMO	19
3.4. RACIOCÍNIO ESPACIAL QUALITATIVO.....	19
3.5. VERIFICAÇÃO AUTOMÁTICA DE REGRAS	21
3.5.1. VERIFICAÇÃO AUTOMÁTICA DE REGRAS NA FASE CONCEITUAL ..	22
3.6. TRABALHOS CORRELATOS	24
3.6.1. dROFUS.....	24
3.6.2. SOLIBRI MODEL CHECKER.....	26
3.6.3. EDM MODEL SERVER.....	26
3.6.4. LicA.....	27

4. UMA ABORDAGEM MULTIAGENTES PARA VERIFICAÇÃO AUTOMÁTICA DE REGRAS EM MODELOS CONCEITUAIS DE EDIFÍCIOS	29
4.1. REQUISITOS PARA O SISTEMA DE VERIFICAÇÃO AUTOMÁTICA DE REGRAS.....	29
4.2. ESTRUTURA DO MODELO MULTIAGENTES.....	30
4.2.1. ONTOLOGIA UTILIZADA.....	31
4.2.2. ARQUITETURA DO SISTEMA MULTIAGENTES.....	35
4.2.3. COMUNICAÇÃO ENTRE OS AGENTES.....	36
4.2.4. AGENTE DESIGNER.....	37
4.2.5. AGENTE MANAGER	37
4.2.6. AGENTE VERIFICADOR DE REGRAS	40
4.2.7. RACIOCÍNIO INTERNO DOS AGENTES.....	43
5. ESTUDO DE CASO: VERIFICAÇÃO AUTOMÁTICA DE REGRAS DE ACESSIBILIDADE PARA CADEIRANTES NO NOVO EDIFÍCIO DO CCE DA UFV	44
5.1. OBJETIVO	44
5.2. METODOLOGIA.....	44
5.2.1. ESCOLHA DAS REGRAS	44
5.2.2. ESCOLHA DO EDIFÍCIO	47
5.2.3. APLICAÇÃO DO MODELO MULTIAGENTES.....	47
5.3. RESULTADOS E DISCUSSÃO	48
6. CONCLUSÕES E PERSPECTIVAS	51
REFERÊNCIAS BIBLIOGRÁFICAS.....	53
APÊNDICE A: SOFTWARES UTILIZADOS	58

RESUMO

SACRAMENTO, Pedro de Almeida, M. Sc., Universidade Federal de Viçosa, março de 2013. **Um Modelo de Sistemas Multiagentes para Auxílio na Fase Conceitual de Projetos de Edifícios.** Orientador: Alcione de Paiva Oliveira. Co-orientador: José Luis Braga.

O projeto de edificações é uma atividade multidisciplinar que envolve a interação de disciplinas como arquitetura, engenharia estrutural e engenharia elétrica, dentre outras. O surgimento de conflitos entre interesses dessas disciplinas é inevitável conforme os projetos crescem em volume de dados ou na complexidade da informação. Entretanto, esses conflitos precisam ser resolvidos para que a equipe obtenha uma boa solução de projeto. Sistemas de verificação automática de regras podem contribuir para a resolução desses conflitos, realizando detecção automática de erros referentes às diferentes disciplinas. No presente trabalho é apresentado um modelo de sistemas multiagentes (SMA) capaz de realizar essa detecção em modelos conceituais de edifícios representados com softwares CAD 3D tradicionais, contrastando com processos atuais que impõem grandes modificações nos processos de representação convencionalmente adotados pela indústria. O modelo foi desenvolvido baseando-se na ontologia *Suggested Upper Merged Ontology* (SUMO) e em um conjunto de perguntas básicas definidas na fase de fundamentação teórica. Após esta etapa, utilizou-se um projeto de edifício da Universidade Federal de Viçosa para compor o estudo de caso para aplicação do modelo desenvolvido. Optou-se por testar as regras de acessibilidade a cadeirantes adaptadas do documento *Accessibility Guidelines for Buildings and Facilities*. Para analisar por completo o edifício, o SMA desenvolvido gastou aproximadamente um segundo, sem a necessidade de intervenção humana. Este SMA gerou um resultado gráfico, identificando os locais internos do edifício que não permitem o deslocamento de cadeirantes. Em suma, conclui-se que a abordagem de SMA adotada nesta pesquisa pode ser utilizada, com os mesmos recursos aqui apresentados, como referencial para novas ferramentas de auxílio na fase conceitual de projetos de edifícios, visando a auxiliar a tomada de decisão dos projetistas desde as primeiras fases da elaboração de um projeto de edifícios.

ABSTRACT

SACRAMENTO, Pedro de Almeida, M. Sc., Universidade Federal de Viçosa, March, 2013. **A Model of Multiagent Systems to Aid the Conceptual Phase of Building Design.** Adviser: Alcione de Paiva Oliveira. Co-adviser: José Luis Braga.

Building design is a multidisciplinary activity that involves the interaction of disciplines such as architecture, structural engineering and electrical engineering, among others. The emergence of conflicts involving distinct disciplinary interests is inevitable as the project grows in data volume or complexity. However, these conflicts must be resolved in order for the team to get a good design solution. The usage of automated rule-checking systems for automatic verification of design rules can contribute to the resolution of such conflicts, performing automatic detection of errors related to different disciplines. In this paper we present a model of multi-agent systems (MAS) capable of performing detection in conceptual building models represented with traditional 3D CAD software, in contrast to current processes that require major changes in the processes of representation conventionally adopted by the industry. The model was developed based on the Suggested Upper Merged Ontology (SUMO) and seven basic questions defined in the theoretical stage. After this step, a building project of the Federal University of Viçosa was used to write a case study for applying the developed model. We chose to test the accessibility rules adapted from the Accessibility Guidelines for Buildings and Facilities. The MAS spent approximately one second to fully analyze the building, and performed the process without the need of human intervention. It generated graphical feedback identifying the internal locations of the building that does not allow the movement of wheelchairs. In sum, we conclude that the MAS adopted in this research can be used, with the same resources presented here, as a reference for new tools to aid in the conceptual phase of building design, aiming to assist designers decision-making since the early stages of building design process.

1. INTRODUÇÃO

A elaboração de projetos no setor da construção civil demanda a participação de diferentes profissionais como arquitetos, engenheiros eletricitas, engenheiros de estruturas, orçamentistas, etc. (REN et al., 2011). Nenhuma disciplina detém sozinha a plenitude do conhecimento demandado pelo processo de projeto (CAVIERES et al. 2011), por isso, estes profissionais precisam cooperar entre si para conduzir satisfatoriamente processos de tomada de decisão que envolvem interesses distintos como segurança, confiabilidade, desempenho, custos (REN et al., 2011) e restrições impostas pela legislação local (EASTMAN et al., 2009a).

Felizmente, o uso de ferramentas computacionais tem contribuído para aperfeiçoar os processos de projetos da construção civil, especialmente com o uso de modelos e simulações que têm promovido, dentre outros benefícios, a antecipação na identificação de problemas (LEE et al., 2012). Tal antecipação é um assunto de grande pertinência em processos industriais, especialmente porque quanto mais tarde um problema é detectado e corrigido, maior tende a ser o custo demandado em sua resolução (WANG et al. 2002). Deste modo, a indústria e a academia têm buscado a criação de ferramentas de auxílio à tomada de decisão capazes de acompanhar o projetista desde o início do ciclo de vida do edifício, ou seja, as primeiras fases de seu projeto (CAVIERES et al., 2011).

1.1. O PROBLEMA E SUA IMPORTÂNCIA

Antes de chegar à sua fase de execução, um projeto de edifício passa por etapas que basicamente podem ser divididas em: fase conceitual e fase de detalhamento (CAVIERES et al., 2011). Dentre estas, estima-se que mais de 80% do custo final de um edifício seja comprometido pelas decisões tomadas na fase conceitual (ATTIA et al., 2012). Essa estimativa foi observada também em outros setores, como a indústria aeroespacial e indústria de produtos eletrônicos, dentre outras (WILL, 1991). Como se trata de um momento de decisão fortemente capaz de influenciar tempo e custos envolvidos no ciclo de vida de edifícios, a fase conceitual é tida atualmente como uma

das direções mais promissoras para ferramentas de auxílio à tomada de decisão em projetos (CAVIERES et al., 2011).

Um recurso almejado para auxílio à tomada de decisão na fase conceitual de projetos são as ferramentas de verificação automática de regras visando à detecção automática de defeitos. Esse tipo de recurso pode ser obtido mediante a criação de processos automáticos de análise de modelos de edifícios sob a perspectiva de um conjunto de regras provenientes de uma ou mais disciplinas. A verificação automática de regras (que no inglês pode ser chamada de *automated rule-checking*) facilita para o profissional o acesso a um grande volume de informações baseadas em conhecimentos de outras disciplinas, além de poder agilizar e aumentar a confiabilidade de cada uma de suas escolhas (LEE et al., 2012). O uso desse tipo de recurso já tem sido pesquisado na área da construção civil, entretanto as pesquisas neste sentido se concentram, principalmente, no atendimento às fases detalhadas do projeto. Infelizmente, as abordagens que atendem bem à fase de detalhamento tendem a se mostrar inadequadas para utilização em projetos conceituais, o que faz da fase conceitual um desafio tanto para a academia quanto para a indústria (CAVIERES et al., 2011).

No cerne do desafio apresentado pela fase conceitual de projetos está a representação da informação (CAVIERES et al., 2011). Cavieres et al. (2011), Wang et al. (2002) e De Vries e Wagter (1990) argumentam que a informação de um projeto conceitual é consideravelmente vaga e imprecisa; enquanto a informação na fase de detalhamento, como sugere o nome, deve ser detalhada e precisa. Diante disso, segundo os mesmos autores, o projeto conceitual exige o uso de representações altamente flexíveis e genéricas, uma demanda que tem desqualificado o uso das ferramentas tradicionalmente utilizadas no detalhamento de projetos e, por consequência, os atuais recursos para verificação automática de regras nesta fase.

A análise das relações espaciais estabelecidas entre as geometrias de um edifício qualquer pode trazer à tona informações implícitas que são muito relevantes para sua compreensão e manipulação visando tomada de decisão, mesmo que o edifício tenha sido representado de maneira vaga, imprecisa ou pouco detalhada. Exemplos disso são os sistemas de navegação adotados por robôs autônomos capazes de realizar navegação *indoor* em edifícios a partir de análises qualitativas de espaços apenas parcialmente conhecidos (COHN e RENZ, 2008). As principais pesquisas neste sentido abordam a questão sob o nome de raciocínio qualitativo espacial, uma área apta a contribuir para a

manipulação da informação de projetos conceituais visando à verificação automática de regras.

1.2. HIPÓTESE

Amparado por pesquisas na área de raciocínio qualitativo como Cohn e Renz (2008), Renz e Nebel (2007) e Hogenboom et al. (2010) a hipótese do presente trabalho é que a utilização de raciocínio espacial qualitativo possibilita a produção automática de informações a respeito de um edifício e que essas informações são suficientes para amparar um processo relevante de verificação automática de regras.

1.3. OBJETIVOS

1.3.1. OBJETIVO GERAL

O objetivo geral do trabalho é elaborar, implementar e testar um modelo de SMA capaz de auxiliar a tomada de decisão durante a fase conceitual da elaboração de projetos de edifícios provendo recursos para verificação automática de regras. Os resultados gerados constituem importante referencial para a condução de futuras pesquisas relacionadas à verificação automática de regras em edifícios.

1.3.2. OBJETIVOS ESPECÍFICOS

Os objetivos específicos deste trabalho podem ser assim dispostos:

- Analisar o atual estado da arte no uso de sistemas computacionais para verificação automática de regras em projetos de edifícios;
- Elaborar um modelo de SMA para verificação automática de regras em projetos de edifícios em fase conceitual;
- Elaborar um estudo de caso com a aplicação do modelo para verificação automática de regras baseadas nas especificações do documento *Accessibility*

Guidelines for Buildings and Facilities (ADAAG) do US Access Board (1998) utilizando um projeto de edifício da UFV.

1.4. ORGANIZAÇÃO DA DISSERTAÇÃO

Para melhor exposição das atividades realizadas na presente pesquisa, a dissertação foi estruturada em seis capítulos:

- **Capítulo 1 – Introdução:** apresenta o problema que motivou a presente pesquisa e sua importância, a hipótese levantada e os objetivos deste trabalho;
- **Capítulo 2 – Metodologia:** contém as atividades realizadas para o desenvolvimento do trabalho, bem como a organização dada ao mesmo e os materiais e métodos utilizados para execução dos capítulos seguintes;
- **Capítulo 3 – Fundamentação Teórica:** expõe assuntos, termos e conceitos importantes para a compreensão do problema apresentado na dissertação;
- **Capítulo 4 – Uma abordagem multiagentes para verificação automática de regras em modelos conceituais de edifícios:** apresenta o modelo de sistema multiagentes proposto e os resultados e conclusões encontrados na elaboração desta etapa da pesquisa;
- **Capítulo 5 – Estudo de caso: Verificação automática de regras de acessibilidade para cadeirantes no novo edifício do CCE da UFV:** apresenta os resultados e conclusões da aplicação do modelo multiagentes desenvolvido num estudo de caso, detalhando também os materiais e métodos utilizados para que se chegasse a esta fase de desenvolvimento da pesquisa;
- **Capítulo 6 – Conclusões Gerais e Perspectivas:** sintetiza as conclusões obtidas no presente trabalho, apresenta os possíveis desdobramentos e perspectivas de continuidade desta pesquisa em ocasiões futuras.

2. METODOLOGIA

A realização do trabalho foi composta de atividades que podem ser descritas conforme a seguir (Figura 1):

- **Atividade 1** – Revisão de literatura sobre o estado da arte na verificação automática de regras em projetos de edifícios e sobre as diferenças entre as necessidades da fase conceitual e da fase detalhada de projetos.
- **Atividade 2** – Revisão de literatura sobre requisitos para sistemas de verificação automática de regras em projetos conceituais.
- **Atividade 3** – Definição de um modelo de SMA de acordo com os requisitos levantados.
- **Atividade 4** – Implementação de um SMA conforme o modelo proposto.
- **Atividade 5** – Escolha de um projeto de edifício para realização de estudo de caso.
- **Atividade 6** - Realização do estudo de caso e análise dos resultados.
- **Atividade 7** – Levantamento das conclusões e perspectivas futuras para continuidade da pesquisa.



Figura 1: Síntese das atividades cumpridas para o desenvolvimento do presente trabalho.

2.1. MATERIAIS E MÉTODOS

O primeiro passo para a execução desta pesquisa foi realizar uma investigação do referencial teórico necessário para amparar o seu desenvolvimento. A partir desta, partiu-se para a atividade de elaboração do modelo multiagentes proposto para contribuir na solução de problemas da fase conceitual de projetos de edifícios. O desenvolvimento deste modelo foi guiado por um conjunto de perguntas apresentadas por Macal e North (2010) como questões básicas para o desenvolvimento de modelos de sistemas multiagentes:

- a) Que problemas especificamente devem ser solucionados pelo modelo? Que perguntas específicas o modelo deve responder? Qual o valor agregado que a

abordagem sistemas multiagentes traz para o problema e que outras abordagens não poderiam trazer?

- b) O que os agentes devem representar no modelo? Quem são os tomadores de decisão no sistema? Que entidades possuem comportamentos? Qual informação dos agentes é apenas descritiva (atributos estáticos)? Que atributos dos agentes seriam calculados de maneira endógena pelo modelo e atualizada nos agentes (atributos dinâmicos)?
- c) O que é o ambiente dos agentes? Como os agentes interagem no ambiente? A mobilidade dos agentes é uma consideração importante?
- d) Quais são os comportamentos de interesse dos agentes? Que decisões os agentes realizam? Em que se baseiam os comportamentos efetuados? Que ações estão sendo tomadas pelos agentes?
- e) Como os agentes interagem entre si? E com o ambiente? Quão expansivas ou focadas são as ações dos agentes?
- f) De onde os dados devem vir, especialmente nos comportamentos dos agentes, para este modelo?
- g) Como validar o modelo, especificamente o comportamento dos agentes?

Uma vez que todos os requisitos acima foram observados na elaboração do modelo multiagentes, este foi aplicado num estudo de caso com um edifício em construção da Universidade Federal de Viçosa (UFV) e testado com base nas regras de acessibilidade a cadeirantes adaptadas do ADAAG. Após o estudo de caso, foram levantadas as conclusões gerais do desenvolvimento de toda a pesquisa e as perspectivas de continuidade da mesma em ocasiões posteriores.

3. FUNDAMENTAÇÃO TEÓRICA

3.1. PROJETO DE EDIFÍCIOS

O desenvolvimento de um projeto de edifício pode ser dividido em três fases: fase conceitual, detalhamento e execução (WANG et al., 2002). Como se observa na Figura 2, o poder de impacto das decisões tomadas nesse processo decresce à medida que o projeto se torna maduro, motivo pelo qual a primeira dessas três fases é destacada como uma oportunidade. Essas três fases envolvem um grande desafio, pois, a verificação de cada decisão de projeto demanda a manipulação de um grande volume de informações provenientes de diferentes áreas do conhecimento. Entretanto, os limites impostos pela cognição humana conduzem o projetista a considerar em suas decisões um conjunto muito pequeno desse universo, ficando sujeito a tomar decisões sem deter a compreensão plena de suas implicações (GANE et al., 2009).

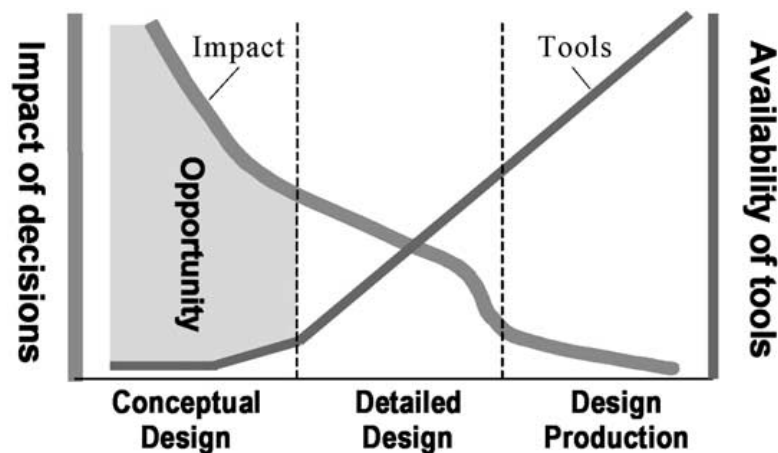


Figura 2: As etapas do processo de projeto e a oportunidade da fase conceitual. Fonte: Wang et al. (2002).

Obviamente, a indústria possui diferentes artifícios para minorar o risco de existência de defeitos no edifício construído, dentre os quais está o levantamento de requisitos. Na fase conceitual do projeto de edifícios, os projetistas não somente identificam um conjunto de requisitos como estabelecem prioridades dentre os requisitos levantados, criam soluções preliminares e avaliam estas soluções. O uso de sistemas de verificação automática de regras nesse momento pode contribuir para a tomada de decisão dos projetistas, provendo recursos como estimativas de custo,

validação espacial, verificação de circulação e segurança (LEE et al., 2012). O resultado final dessa fase é a definição dos requisitos, preferências e critérios de avaliação para o projeto final (GANE et al., 2009). Já na fase de detalhamento, os projetistas elaboram com maior resolução as diferentes partes do projeto como os projetos estrutural, elétrico e hidráulico. O resultado dessa fase é um conjunto de desenhos e especificações técnicas aptas a servirem de amparo para a efetiva construção do edifício projetado (GRAY e HUGHES, 2012). A fase de produção é a etapa em que o edifício é executado, cujo produto final é a própria edificação construída.

3.1.1. BUILDING INFORMATION MODELING (BIM)

Building Information Modeling (BIM) é um termo presente em grande parte das pesquisas na área de softwares para auxílio à tomada de decisão em projetos de edifícios, como apresentado nos trabalhos de Eastman et al. (2009a), Lee et al. (2012) e Martins e Monteiro (2013). BIM se refere a um processo de criação e utilização de modelos virtuais de edifícios que simulam o planejamento, projeto, construção e operação de uma obra. Os modelos BIM podem carregar informações referentes a todo o ciclo de vida do edifício (Figura 3). Tais informações auxiliam arquitetos, engenheiros e contratantes a melhor visualizarem um projeto e identificarem seus potenciais problemas. Esta abordagem contrasta com abordagens CAD tradicionais, onde as partes do edifício são representadas como um conjunto de geometrias (AZHAR et al., 2008).

desafiador, pois o ciclo envolve muitos participantes que interagem com a obra utilizando tecnologias muito distintas (MARTINS e MONTEIRO, 2013).

3.2. SISTEMAS MULTIAGENTES

Um SMA é caracterizado por um conjunto de agentes autônomos que cooperam, coordenam e negociam entre si (WOOLDRIDGE, 2002). Tais sistemas são comumente utilizados para atacar problemas mal-estruturados em ambientes heterogêneos onde a informação muda rapidamente, o que ocorre no domínio de projetos de edifícios. Dentre os benefícios que o uso de sistemas multiagentes pode oferecer estão a efetiva decomposição de problemas de grande escala, a melhoria da colaboração e o trabalho paralelo de equipes de projeto (REN et al., 2011).

Segundo Macal e North (2010), um SMA tipicamente possui três elementos:

- Um conjunto de agentes, seus atributos e comportamentos;
- Um conjunto de relacionamentos e métodos de interação, ou seja, uma topologia que estabelece que os agentes podem se comunicar entre si e qual a maneira desta comunicação ocorrer;
- O ambiente dos agentes, de modo que o agente não interaja apenas com outros agentes, mas também com o ambiente em que se encontra.

3.2.1. AGENTES

Segundo Macal e North (2010), enquanto alguns autores defendem que qualquer componente independente (*software*, modelo, indivíduo) é um agente, outros insistem que apenas um componente adaptativo, capaz de aprender e mudar seu comportamento de acordo com suas experiências, pode ser considerado um agente. Apesar dessas questões divergentes, os autores enunciam um conjunto de características consideradas essenciais para um agente:

- **Um agente é um indivíduo modular, contido em si mesmo e identificável de maneira única.** Agentes possuem atributos que lhes permitem ser distinguidos e reconhecidos por outros agentes. Estes atributos funcionam como fronteiras que

delimitam aquilo que faz parte do agente, o que não faz parte e o que consiste em um atributo compartilhado com outros agentes.

- **Um agente é autônomo e dirigido por si mesmo.** O agente pode funcionar de maneira independente no ambiente em um conjunto de situações de interesse dentro do modelo. O agente possui comportamentos que relacionam a informação percebida com as decisões e ações a serem tomadas. As informações que o agente manipula vêm de suas interações com o ambiente e com outros agentes. O comportamento de um agente pode ser especificado por qualquer coisa, desde regras simples a modelos abstratos, como redes neurais e algoritmos genéticos.
- **Um agente possui um estado que varia ao longo do tempo.** Assim como um sistema possui um estado que consiste em um conjunto de suas variáveis de estado, um agente também possui um estado que representa as variáveis essenciais associadas com sua situação atual. O estado de um agente consiste em um subconjunto de seus atributos. O estado de um modelo baseado em agentes é o *status* coletivo de todos os agentes além do estado do ambiente. Além disso, os comportamentos do agente são condicionados em seu estado e, portanto, quanto mais rico for o conjunto de estados possíveis para um agente, mais rico será o conjunto de comportamentos que o agente poderá apresentar.
- **Um agente é social, interagindo de maneira dinâmica com outros agentes que influenciam seu comportamento.** Agentes possuem protocolos para interação com outros agentes, como comunicação, deslocamento e capacidade de responder ao ambiente e aos outros agentes. Além disso, os agentes são capazes de reconhecer e distinguir características de outros agentes.

Macal e North (2010) apresentam também características adicionais que podem caracterizar os agentes em um sistema:

- **Um agente pode ser adaptativo.** O agente pode se adaptar a situações ao longo do tempo, tendo a capacidade de aprender com experiências anteriores, possuindo algum tipo de memória. Além da adaptação no nível individual, também pode ocorrer a adaptação de uma população de agentes, mediante a existência de mecanismos para aumento da população de indivíduos mais adequados e redução dos menos adequados em um ambiente.

- **Um agente pode ser orientado por objetivos.** O agente pode possuir objetivos a atingir com seus comportamentos. Isso permite que o agente avalie o resultado de seu comportamento e possa ajustar seu comportamento ao longo do tempo, como descrito no tópico anterior.
- **Um agente pode fazer parte de um conjunto heterogêneo.** As características e comportamentos dos agentes podem variar quanto: à extensão, sofisticação, quantidade de informação utilizada para tomar uma decisão, aos modelos internos dos agentes a respeito do mundo exterior, à visão do agente a respeito das possíveis reações de outros agentes em relação a suas ações, à extensão da memória do agente com relação aos eventos passados e à granularidade da informação manipulada, dentre outros. Além disso, os agentes podem ser dotados de diferentes quantidades de informação acumulada como resultado de suas interações, o que promove diferenciação entre os agentes.

Cada agente possui, segundo Russell e Norvig (2010), capacidade de perceber o ambiente através de sensores (*sensors* ou *perceptors*) e de agir neste ambiente com uso de atuadores (*actuators* ou *effectors*), como mostrado na Figura 4. Os autores comparam essas características com o funcionamento do corpo humano: um agente humano possui órgãos para perceber (ouvidos, olhos e outros), e órgãos para atuar (pernas, mãos, cordas vocais, etc). Russell e Norvig (2010) exemplificam dizendo que as câmeras e motores de um agente robô são, respectivamente, sensores e atuadores, e que de modo semelhante um software agente recebe arquivos, pacotes da rede e pressionamento de teclas com seus sensores, atuando com a exibição de algo na tela, o envio de pacotes na rede, etc.

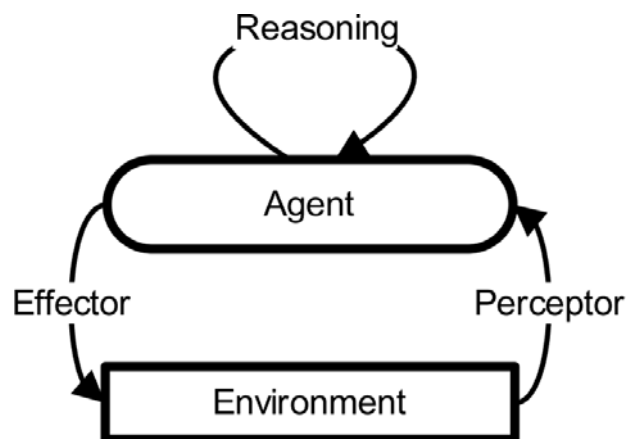


Figura 4: O agente interage com o ambiente com sensores e atuadores. Fonte: Beetz (2009).

A Figura 4 também destaca a presença de raciocínio (*reasoning*) em um agente que, por conseguinte, está relacionada à sua capacidade de tomada de decisão. Segundo Macal e North (2010), a capacidade de decidir de maneira autônoma, ou seja, de responder às situações encontradas sem a necessidade de direção externa, constitui a principal característica de um agente. Os agentes geralmente são ativos, tomando iniciativas para atingir seus objetivos ao invés de simplesmente responder aos estímulos provenientes dos outros agentes ou percebidos no ambiente.

Tipicamente, em um modelo baseado em agentes, tudo o que está associado ao agente se torna um atributo ou método que opera no agente. Esses atributos podem ser estáticos ou dinâmicos (como o nome do agente e sua memória, respectivamente) e os métodos incluem comportamentos que relacionam a situação atual do agente com suas possíveis ações, como, por exemplo, numa rede neural (MACAL e NORTH, 2010).

3.2.2. SISTEMAS MULTIAGENTES

Os SMAs podem ser usados para a modelagem de processos dinâmicos e dependentes do tempo ou até mesmo, de uma maneira mais genérica, de processos de otimização e busca. A modelagem de um modelo como esses envolve também a definição dos relacionamentos e interações que o agente estabelece com outros agentes, o que é tão importante quanto a modelagem do comportamento individual do mesmo. Ou seja, a definição de quem é o agente, a quem ele se conecta e os mecanismos da dinâmica de suas interações são questões primárias a se considerar para modelagem de um modelo baseado em agentes (MACAL e NORTH, 2010).

Segundo Fischer et al. (1998), sistemas que manipulam conhecimento apresentam o risco de entrar em colapso mediante a realização de pequenas mudanças não previstas em sua concepção. Em um contexto de manipulação de conhecimento multidisciplinar esse tipo de limitação pode ser muito prejudicial para o sistema, uma vez que a evolução do conhecimento das diferentes disciplinas tenderá a demandar a realização desse tipo de mudança. Entretanto, Segundo Macal e North (2010), um SMA pode ser adaptativo, possuindo a capacidade de atacar a fragilidade descrita por Fischer et al. (1998). O benefício disso está na possibilidade da realização de mudanças

profundas na maneira como cada disciplina analisa um projeto sem que isso comprometa negativamente o funcionamento do sistema como um todo.

Macal e North (2010) explicam que a observação de sistemas baseados em agentes revela aspectos como capacidade de adaptação de populações inteiras de agentes, de modo que o sistema possa se ajustar a novos contextos. Esses aspectos envolvem o surgimento de estruturas e comportamentos não modelados explicitamente no ato da concepção do modelo, que emergem através das interações dos agentes em decorrência da diversidade de seus atributos. A ênfase nessa heterogeneidade dos agentes faz parte dos aspectos que distinguem um sistema baseado em agentes de sistemas modelados com outras abordagens, como dinâmicas de sistemas (MACAL e NORTH, 2010).

3.2.3. SISTEMAS MULTIAGENTES NA CONSTRUÇÃO CIVIL

O trabalho de Ren et al. (2011) ressalta que o uso de SMAs permite a modelagem de sistemas onde cada disciplina pode ser representada por um agente, contribuindo para a modelagem de conhecimentos multidisciplinares de maneira intuitiva, modular e expansível. Geralmente os agentes em ambientes de projeto são utilizados para possibilitar a integração entre ferramentas tradicionais, facilitar o compartilhamento de informações, dados e conhecimento, além de contribuir para que conflitos sejam resolvidos (REN et al., 2011; FAUS et al., 2012).

A importância da construção de um sistema onde cada disciplina pode ter seu próprio agente é ressaltada pelo fato de que as disciplinas envolvidas no processo de projeto podem utilizar modelos muito distintos como: simulações de conforto térmico utilizando dinâmica de fluidos computacional (TREECK et al., 2006), análise de relações de custos e impactos ambientais com uso de algoritmos genéticos (WANG et al., 2005), criação de rotas de fuga para multidões em pânico utilizando algoritmos para *route-planning* em tempo real (ZHANG et al., 2012). Com uso de SMAs, diferentes abordagens como estas podem ser integradas em um mesmo sistema que provê *feedback* para o projetista amparando sua tomada de decisão em situações de conflito implementando essas diferentes perspectivas (REN et al., 2011).

Moulin et al. (2011) apresentam o uso de sistemas multiagentes aliado à utilização de periféricos não convencionais no auxílio à atividade de projeto conceitual

com mediação de uma superfície multi-toque gerenciada por um SMA. A decisão de utilizar SMA nesse contexto foi motivada pela necessidade de integração de dados provenientes de diferentes fontes, como documentos, outros projetos e bases de conhecimento. Apesar de possuir um foco distinto, o trabalho de Moulin et al. (2011) é relevante para a presente pesquisa por demonstrar que a utilização de SMAs também pode ser abordada sob a perspectiva da utilização de novos dispositivos.

3.3. ONTOLOGIAS

Uma ontologia é, na ciência da computação, uma estrutura de termos e suas relações em um determinado domínio. Ontologias permitem a captura de conceitos, podendo ser utilizadas para diferentes finalidades como representação, reuso, compartilhamento, aquisição e integração de conhecimento (MOREIRA et al. 2004).

A grande contribuição do uso de uma ontologia está na possibilidade de compartilhamento de significado em diferentes partes de um mesmo sistema. Isto se torna possível pelo poder de descrição de conceitos que permite que estes sejam compreendidos consistentemente quando se fizer necessário (LACLAVÍK, 2005). Na Figura 5 é exibida parte da ontologia *Suggested Upper Merged Ontology* (SUMO) que amparou o desenvolvimento da ontologia adotada para este trabalho.

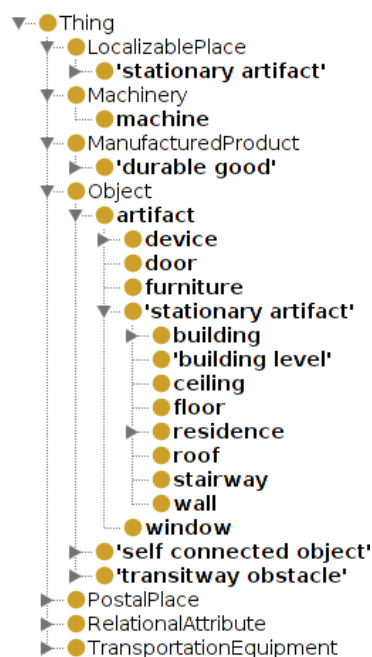


Figura 5: Vista parcial da ontologia SUMO utilizada neste trabalho.

Ontologias podem ser utilizadas de diferentes maneiras em um software. Por exemplo: uma ontologia pode ser utilizada apenas durante a fase de desenvolvimento do software ou em tempo de execução; pode ter seus conceitos invisíveis para os usuários ou possuir uma interface específica para a interação dos mesmos. Niles e Pease (2001) argumentam de maneira contundente a respeito da importância do uso de ontologias de maneira visível para o usuário, enfatizando que os mecanismos de pesquisa na internet são severamente prejudicados quando implementam buscas com uso de palavras-chave ao invés de utilizar conceitos formalizados em uma ontologia.

Especialmente no desenvolvimento de sistemas para *web*, o uso de ontologias tem crescido muito nos últimos anos, caracterizando a chamada *web-semântica*, um avanço nos mecanismos de estruturação de dados da rede como descrito há mais de 10 anos por Berners-Lee et al. (2001). Paulheim e Probst (2010) ressaltam que, no desenvolvimento de softwares como um todo, até o final da década passada, a utilização de ontologias ocorria principalmente em camadas da aplicação invisíveis ao usuário. Entretanto, estes autores também citam que recentemente pesquisas observaram que mais de 90% das aplicações de *web-semântica* possuem uma interface para que o usuário interaja diretamente com uma ontologia. A Figura 6 apresenta um conjunto com algumas das principais ontologias utilizadas nessa nova estrutura da *web*.

3.3.1. ONTOLOGIA SUMO

O trabalho de Niles e Pease (2001) apresenta uma ontologia formal de nível topo chamada SUMO. Esta é composta pela própria SUMO e um conjunto de ontologias de domínios específicos como: comunicação, economia, finanças, geografia, sistemas físicos, pessoas, etc. Trata-se da maior ontologia formal gratuita existente atualmente, possuindo mais de 25.000 termos e mais de 80.000 axiomas. Os axiomas da SUMO são descritos na linguagem *Standard Upper Ontology Knowledge Interchange Format* (SUO-KIF), uma linguagem de lógica de primeira ordem. Todos os termos da *WordNet* foram mapeados manualmente para os conceitos correspondentes na SUMO (MASCARDI et al., 2007).

Por se tratar de uma grande ontologia formal gratuita, a SUMO foi escolhida para a presente pesquisa, e seus conceitos foram utilizados para a atribuição de significado às geometrias que compõem edifícios. A utilização de uma ontologia formal de nível topo contribui para que o sistema seja capaz de se integrar a outros sistemas (NILES e PEASE, 2001).

3.4. RACIOCÍNIO ESPACIAL QUALITATIVO

O Raciocínio Qualitativo (RQ) é uma abordagem para manipulação de conhecimento de senso comum que, ao contrário da abordagem *fuzzy*, não se baseia em aproximações com valores reais, mas cria apenas as distinções necessárias para uma representação adequada de acordo com uma aplicação. Em lugar de números, o raciocínio é efetuado com a manipulação de vocabulários limitados que representam relações qualitativas (ex.: longe, próximo, grande, pequeno) ou categorias qualitativas para os números (ex.: {+, -, 0}) (RENZ e NEBEL, 2007; HOGENBOOM et al., 2010). Segundo Renz e Nebel (2007) a abordagem qualitativa é mais próxima da maneira como os seres humanos representam e raciocinam sobre conhecimentos de senso comum, permitindo a manipulação de conhecimento incompleto. Tais aspectos são muito relevantes no contexto da tomada de decisão multidisciplinar (REN et al., 2011).

Dentre os conhecimentos manipulados com RQ, destacam-se as áreas de Raciocínio Temporal (RT) e Raciocínio Espacial (RE), estando o último diretamente relacionado com esta pesquisa. São exemplos de aplicação do RE: criação de

mecanismos intuitivos para interação de usuários com grandes bases de dados *Geographic Information Systems* (GIS), amparo à navegação de robôs, raciocínio de senso comum a respeito de sistemas físicos, dentre outros. Como nas primeiras fases de um projeto a descrição do conhecimento é significativamente qualitativa, a abordagem RE pode ser aplicada para manipulação da informação em projetos conceituais (COHN e RENZ, 2008).

No campo de RE existem métodos como *Region Connection Calculus 8* (RCC-8) de Randell, Cui e Cohn e o *9-intersection model* de Egenhofer (RENTZ e NEBEL, 2007). Segundo Renz e Nebel (2007), estes métodos permitem a qualificação de relações topológicas entre quaisquer dois elementos no espaço e para isso definem um conjunto com oito relações básicas, que na Figura 7 recebem a nomenclatura estabelecida pelo RCC-8.

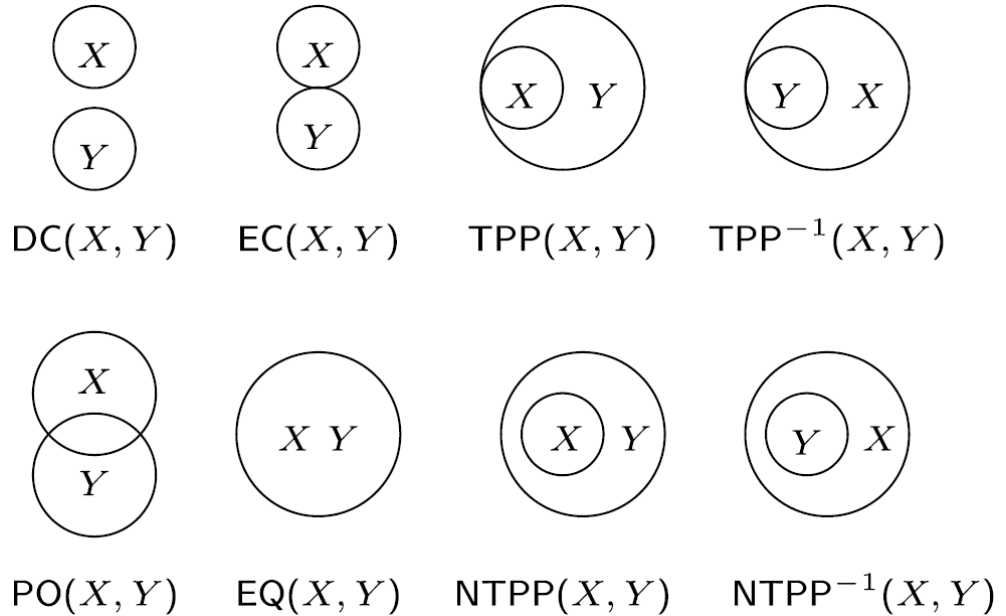


Figura 7: Oito relações espaciais descritas conforme nomenclatura do RCC-8. Fonte: Nebel e Freksa (2011).

No trabalho de Clementini et al. (1994) as oito relações de X em relação a Y presentes na Figura 7 são descritas como: *disjoint*, *meet*, *coveredBy*, *covers*, *overlap*, *equal*, *inside* e *contains* (tomando-as da esquerda para a direita e de cima para baixo). Esses nomes ou nomes similares são utilizados em trabalhos recentes na área de GIS e web-semântica como Koubarakis et al. (2012).

As oito relações espaciais da Figura 7 podem ser facilmente inferidas a partir da análise das regiões no espaço representadas por geometrias que compõem um modelo de edifício 3D. A inferência dessas relações possui ampla aplicação especialmente no contexto de ferramentas GIS (COHN e RENZ, 2008). Essas relações também se aplicam ao processo de verificação automática de regras proposto no presente trabalho, o que é apresentado com mais detalhes no capítulo que trata da descrição do modelo proposto.

Os sistemas que utilizam raciocínio qualitativo comumente utilizam também uma ontologia que formaliza os conceitos atribuídos às diferentes regiões no espaço. A qualificação semântica de regiões no espaço, com uso de uma ontologia aliada à análise das relações espaciais que elas estabelecem entre si, produz informações relevantes para o auxílio à tomada de decisão (COHN e RENZ, 2008). No presente trabalho, por exemplo, aborda-se a utilização da inferência de relações qualitativas entre regiões no espaço visando à extração de informações relevantes para a verificação automática de regras em um modelo de edifício.

3.5. VERIFICAÇÃO AUTOMÁTICA DE REGRAS

A verificação automática de regras segundo Eastman et al., (2009a) consiste na utilização de softwares que não modificam o modelo de uma edificação, entretanto, avaliam uma proposta de projeto com base nos aspectos dos elementos que a compõem, seus atributos e suas relações. Os critérios para avaliação do edifício se expressam por meio de regras, restrições ou condições. Uma vez executadas, essas verificações produzirão pareceres como “passou”, “falhou”, “cuidado” ou “desconhecido” a respeito das partes do edifício. Grande parte dos trabalhos no campo de verificação automática de regras aborda questões como verificação de restrições legais e critérios de acessibilidade (EASTMAN et al., 2009).

Os primeiros trabalhos nesse sentido datam da década de 80 e a partir da década de 90 sofreram um impacto significativo com o advento do padrão IFC, um padrão industrial que visa à neutralidade, sendo atualmente implementado na maioria dos softwares BIM (EASTMAN, 2009a).

3.5.1. VERIFICAÇÃO AUTOMÁTICA DE REGRAS NA FASE CONCEITUAL

Geralmente a razão de existir de um edifício é determinada pelos espaços que são delimitados por suas paredes, portas, tetos, pisos etc. (LEE et al., 2012). Hetherington et al. (2011) argumentam, entretanto, que esses espaços nem sempre são representados de maneira explícita em um modelo, sendo apenas modelados de maneira indireta através dos elementos que os enclausuram (Figura 8). Na Figura 9 ressalta-se que a verificação automática de regras em edifícios conceituais exige a representação explícita desses espaços, que devem ser representados de maneira coesa, semântica e geometricamente como proposto por padrões de representação como o *City Geography Markup Language* (CityGML), um padrão de representação de modelos virtuais de cidades (OGC, 2012).

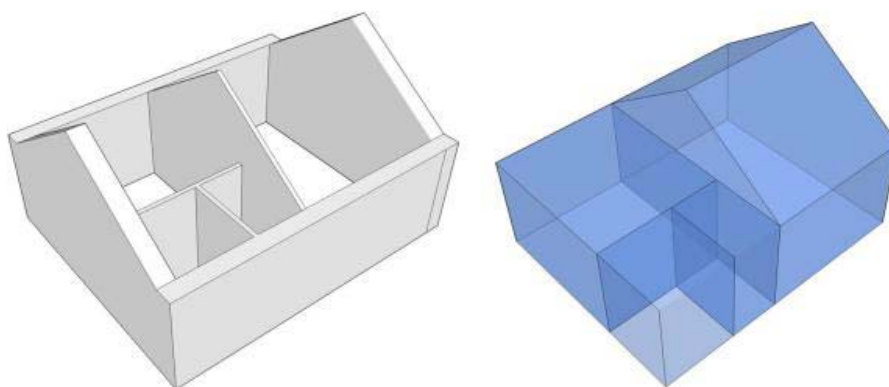


Figura 8: Os espaços no interior de um edifício podem ser representados explicitamente (à direita) ou apenas de maneira implícita pelos elementos que o enclausuram (à esquerda). A representação explícita é mais adequada para verificação de regras em projetos conceituais. Fonte: Hetherington et al. (2011).

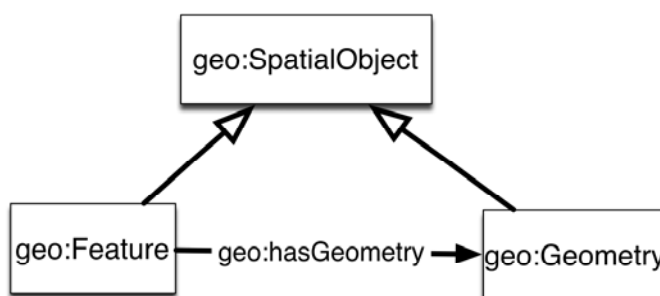


Figura 9: A verificação de regras em edifícios exige a modelagem coesa das estruturas semântica e geométrica do edifício. Fonte: Bermudez (2012).

Durante a definição de requisitos em um projeto de edifício em fase conceitual, os espaços recebem nomes aos quais são relacionadas as propriedades desejadas para o edifício. Nesse contexto, fazem parte dos principais atributos a respeito do espaço tanto o seu nome quanto sua área. Dependendo do uso do modelo de edifício, um espaço também pode demandar a definição de outras questões como os pontos de acesso, visando a análise de questões referentes à acessibilidade e circulação (LEE et al., 2012).

Eastman et al. (2009b) argumentam que a questão da definição de nomes para os espaços é um desafio para a preparação de modelos de edifícios no contexto da verificação automática de regras. Diferentes perspectivas a respeito do projeto utilizam nomes distintos para qualificar as mesmas partes de um edifício. No trabalho de Eastman et al. (2009b) o problema foi parcialmente contornado com a utilização de uma convenção na nomenclatura dos ambientes e de uma ferramenta desenvolvida *ad-hoc* para atender a necessidade de mapeamento de conceitos apresentados por diferentes programas.

Entretanto, Eastman et al. (2009b) ressaltam a necessidade do estabelecimento de um vocabulário geral apto à ampla utilização, o que, segundo eles, levaria anos para ser construído. Felizmente, a definição desse tipo de vocabulário no contexto de projetos conceituais pode ser amparada por uma ontologia, o que neste trabalho, por exemplo, foi realizado com auxílio da ontologia SUMO.

Em resumo, a verificação automática de regras em edifícios conceituais pode ser realizada em edifícios representados com uma estrutura geométrica, onde suas geometrias são representadas e com uma estrutura semântica, onde as geometrias recebem significado de acordo com os conceitos de uma ontologia. Adicionalmente, a partir da estrutura geométrica, é possível inferir novas informações pertinentes ao processo de verificação automática de regras, como mostrado no grafo de acessibilidade da Figura 10, inferido mediante a análise de relações de adjacência e conectividade a exemplo do trabalho de Dong e Brück (2011) que aplica essa análise no contexto de navegação *indoor*. O resultado desse tipo de inferência pode ser um conjunto de relações qualitativas entre as regiões no espaço como “A porta A se conecta ao ambiente B”, o que enriquece o poder de raciocínio de um sistema de verificação automática de regras.

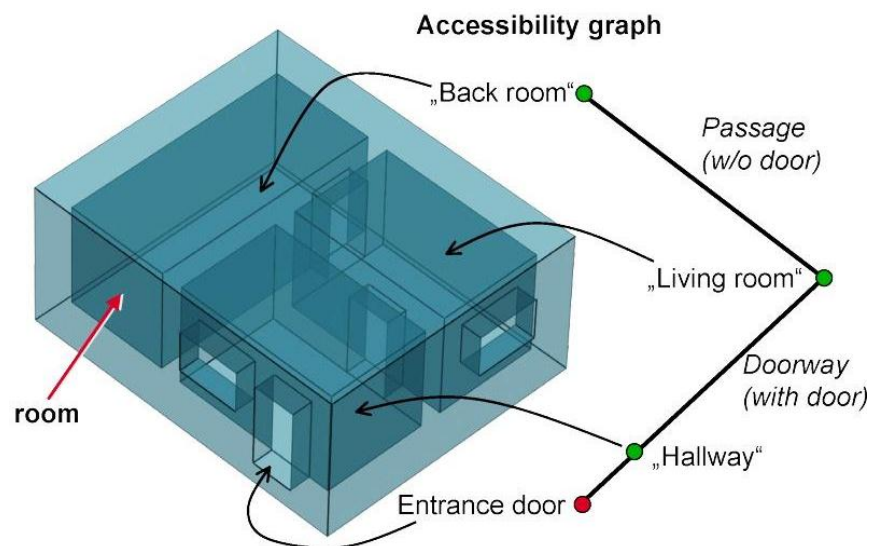


Figura 10: Relações espaciais inferidas em um modelo de edifício. Fonte: OGC (2012).

3.6. TRABALHOS CORRELATOS

Os trabalhos descritos a seguir correspondem a trabalhos recentes e relevantes no campo da verificação automática de regras em modelos de edifícios, compondo uma contribuição significativa para a presente pesquisa.

3.6.1. dROFUS

dRofus é um sistema cliente-servidor que possui um banco de dados para gerenciamento de requisitos funcionais e construtivos de projetos de edifícios, desde as fases iniciais de projeto até sua conclusão. O sistema é voltado especificamente para gerenciamento das áreas dos ambientes, móveis, utensílios e equipamentos. dRofus é capaz de avaliar um projeto de acordo com os requisitos especificados pelo usuário, enfatizando as discrepâncias observadas entre os valores das áreas inseridas pelo usuário como desejáveis e os valores observados nos modelos de edifício (EASTMAN et al., 2009a). O dRofus provê *feedback* gráfico intuitivo a respeito de suas verificações (Figura 11).

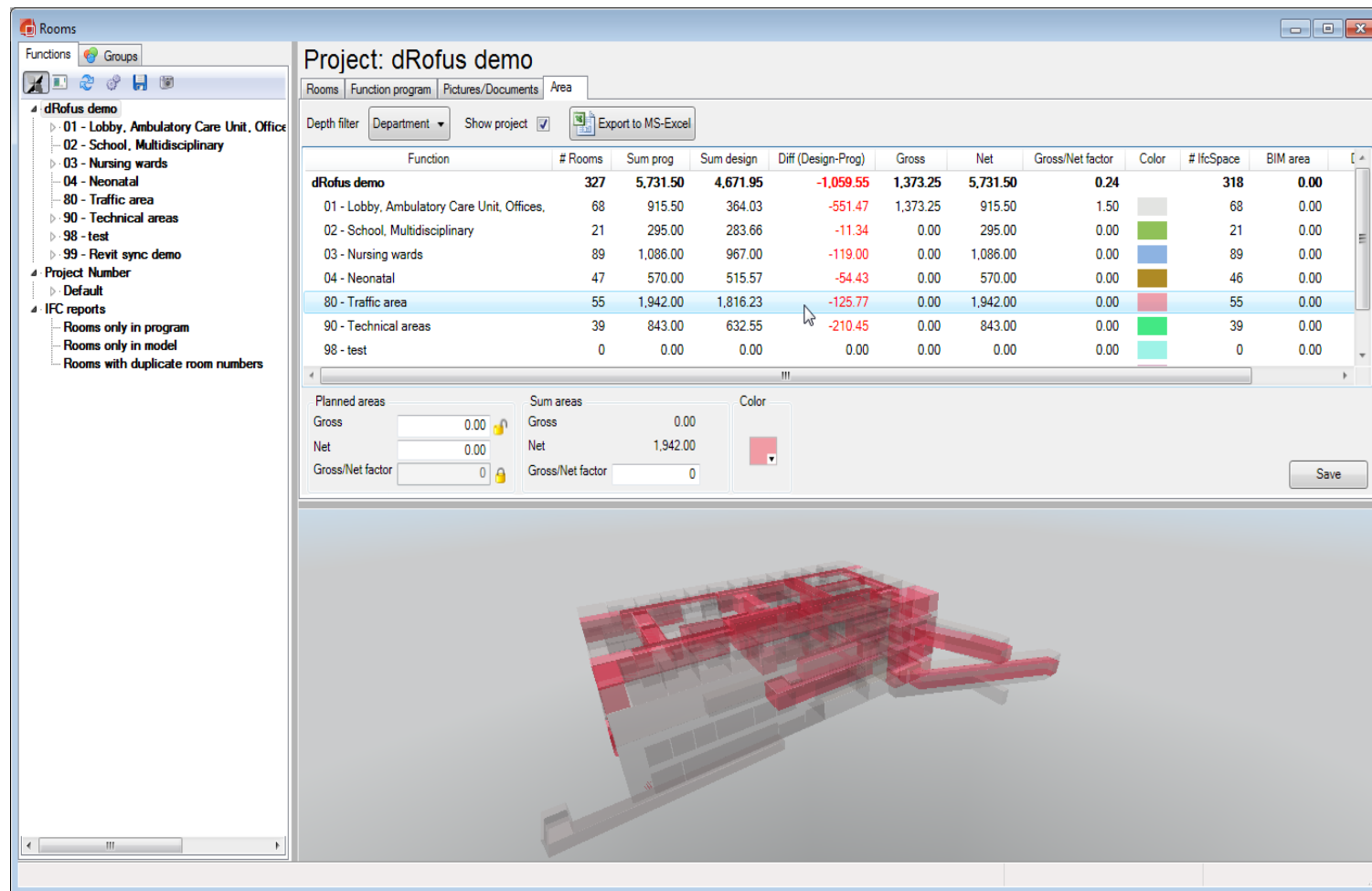


Figura 11: Interface gráfica do dRofus. Fonte: Nosyko (2012).

3.6.2. SOLIBRI MODEL CHECKER

Solibri Model Checker (SMC) é uma plataforma Java que lê arquivos IFC e executa operações como detecção de sobreposições entre geometrias, convenções de nomes e atributos, verificação de um conjunto predefinido de regras englobando acessibilidade, comparação entre os espaços desejados e os espaços encontrados no modelo de edifício, verificação do comprimento das rotas de fuga em caso de incêndio, etc (Figura 12). As regras possuem parâmetros que podem ser modificados (EASTMAN et al., 2009) e o sistema atualmente suporta também a adição de novas regras pelo usuário (KULUSJÄRVI e WIDNEY, 2010).

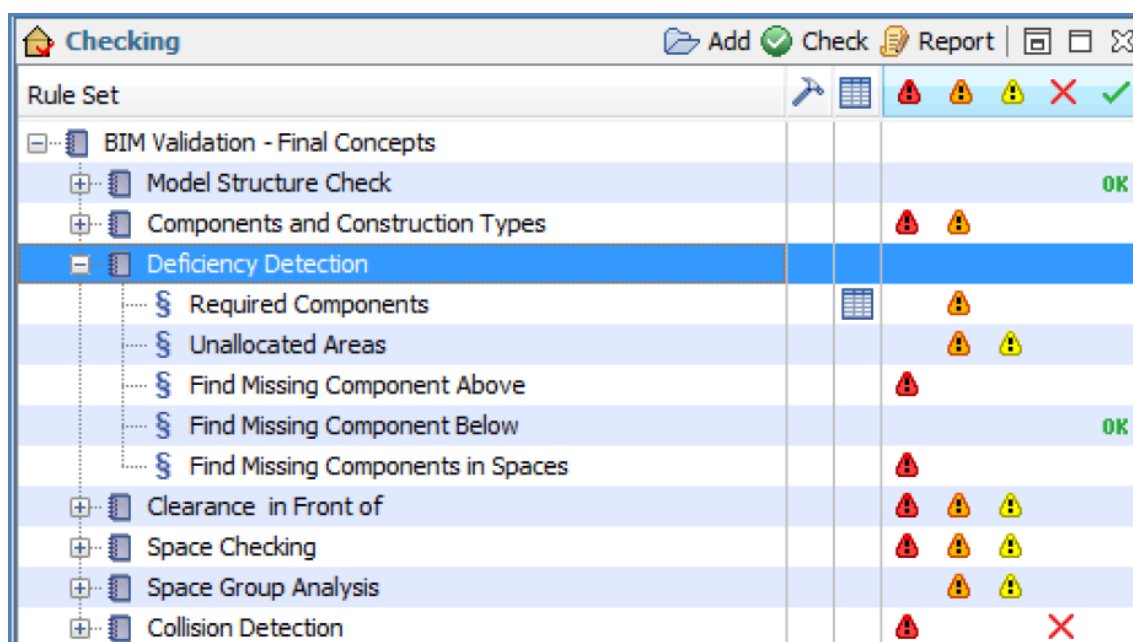


Figura 12: Exemplo de regras de detecção de defeitos. Fonte: Kulusjärvi e Widney (2010).

3.6.3. EDM MODEL SERVER

Jotne EDM Model Server (EDM) é uma ferramenta para manipulação de modelos de edifícios que utiliza uma linguagem padrão para realização de consultas. O EDM possui um banco de regras que pode ser modificado pelo usuário. As regras são escritas em linguagem EXPRESS, a mesma em que o esquema IFC foi escrito. EDM também provê relatórios textuais (EASTMAN et al., 2009a; BELL et al., 2009).

O EDM foi utilizado como base de uma importante iniciativa chamada “CORENET e-PlanCheck”, com a qual o governo de Cingapura iniciou a verificação

automática de regras em projetos no país, uma iniciativa que inspirou a maior parte das iniciativas posteriores neste sentido desde as primeiras tentativas do governo em 1995 (EASTMAN et al., 2009a).

3.6.4. LicA

O trabalho de Martins e Monteiro (2013) apresenta uma pesquisa conduzida na Universidade do Porto em Portugal que resultou em um sistema chamado “LicA” para verificação automática de regras em projetos de redes hidráulicas domésticas (Figura 13). Por deliberação dos pesquisadores, a pesquisa foi conduzida com um escopo muito restrito, entretanto, o trabalho também almeja aplicabilidade em outros contextos, como a verificação de regras de outros países.

LicA consiste em um banco de dados e em uma interface para usuário que possui como uma das principais características o uso de um código de cores para representar resultados de verificação automática de regras no projeto. Esta foi a principal influência do LicA para a presente pesquisa, auxiliando a representação gráfica dos resultados gerados na verificação automática de regras (Figura 14).

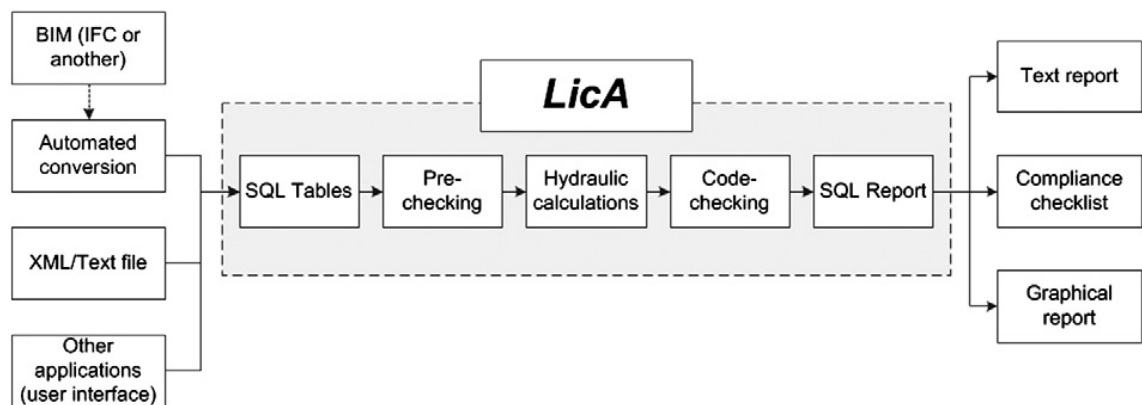


Figura 13: Arquitetura conceitual do LicA. Fonte: Martins e Monteiro (2013).

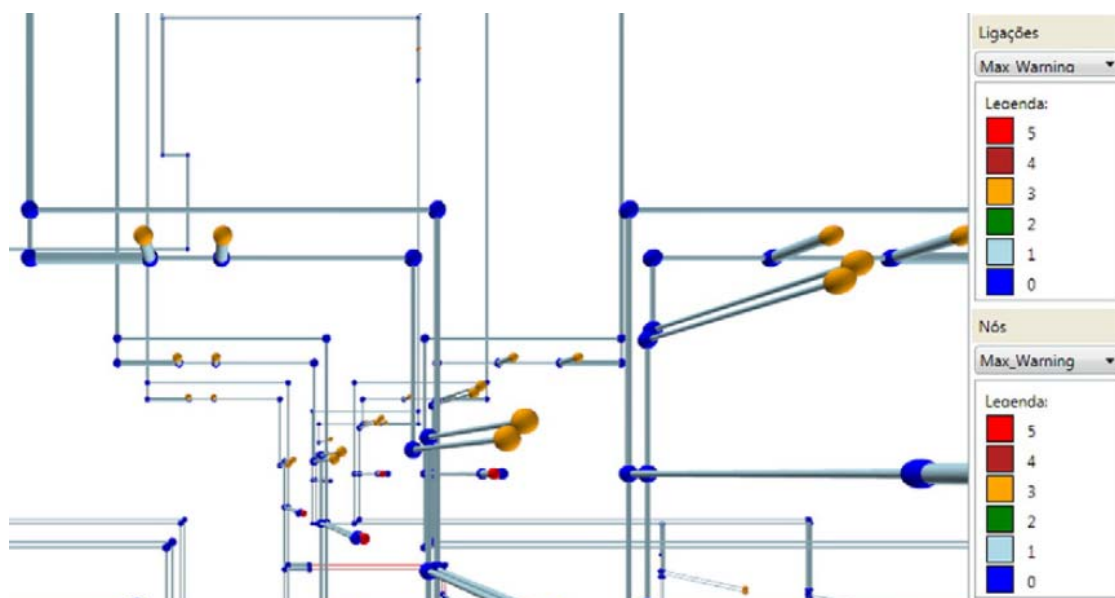


Figura 14: Interface Gráfica de verificação de regras do LicA utilizando um código de cores que indicam adequação ou não às regras verificadas. Fonte: Martins e Monteiro (2013).

Em relação aos trabalhos apresentados anteriormente, o sistema proposto pela presente pesquisa possui os seguintes principais diferenciais: a utilização de uma abordagem SMA e a presença de um processo de verificação de regras de maneira incremental, baseado nas relações de “estar dentro de” ou “estar adjacente a”, estabelecidas entre as geometrias do edifício. Além disso, ferramentas tradicionais de CAD 3D são aptas a produzir modelos de edifício sobre os quais é possível realizar um processo de verificação automática de regras com a abordagem proposta, o que se apresenta como uma grande contribuição para a indústria diante das atuais dificuldades impostas pelos recursos disponíveis até o presente momento.

4. UMA ABORDAGEM MULTIAGENTES PARA VERIFICAÇÃO AUTOMÁTICA DE REGRAS EM MODELOS CONCEITUAIS DE EDIFÍCIOS

O modelo proposto foi descrito conforme os itens apresentados na Figura 15, detalhados posteriormente nos tópicos que se seguem:

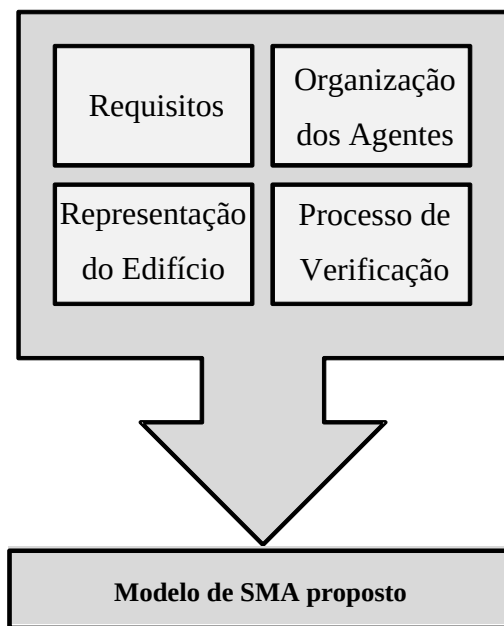


Figura 15: Composição do modelo de SMA proposto.

4.1. REQUISITOS PARA O SISTEMA DE VERIFICAÇÃO AUTOMÁTICA DE REGRAS

Com base na revisão de literatura realizada, identificou-se um conjunto de requisitos não funcionais para o sistema de verificação automática de regras no contexto da manipulação de projetos conceituais. Os requisitos listados a seguir foram selecionados por terem se destacado na literatura como parte dos principais desafios referentes a: modelagem conceitual de edifícios; verificação automática de regras e manipulação de conhecimento de projeto em sistemas especialistas.

- Interoperabilidade com ferramentas já utilizadas pela indústria (MARTINS e MONTEIRO, 2013);

- Capacidade de manipulação de regras com modelos distintos provenientes de diferentes disciplinas (REN et al., 2011);
- Capacidade de adaptação às mudanças em regras e modelos referentes às diferentes disciplinas ao longo do tempo (FISCHER et al., 1998);
- Baixa ou nenhuma necessidade de entrada manual de parâmetros para a verificação automática de regras (MARTINS e MONTEIRO, 2013);
- Facilidade de utilização para pessoas de diferentes disciplinas (REN et al., 2011);
- Exibição de resultados de maneira intuitiva (MARTINS e MONTEIRO, 2013).

4.2. ESTRUTURA DO MODELO MULTIAGENTES

O modelo de SMA proposto foi elaborado para possuir o seguinte funcionamento na verificação automática de regras: o projetista envia um projeto para o sistema e o sistema retorna um relatório com a análise do projeto, como indicado pelo processo da Figura 16. Dado um conjunto de regras conhecidas e um modelo virtual de edifício, o SMA deve ser capaz de indicar quais geometrias violam as regras e de relacionar as geometrias às regras violadas. O valor agregado pela utilização de SMAs neste contexto está em uma efetiva decomposição do conhecimento manipulado, pois as restrições de cada disciplina são modeladas em um agente específico para aquela disciplina.

A vantagem da utilização de SMAs sobre outras abordagens está na criação de sistemas capazes de se adequar às mudanças de maneira autônoma ao longo do tempo. Esta é uma das principais demandas para sistemas de auxílio à tomada de decisão em projetos e um dos maiores desafios para este tipo de modelo no que se refere à capacidade de manipulação do conhecimento multidisciplinar.

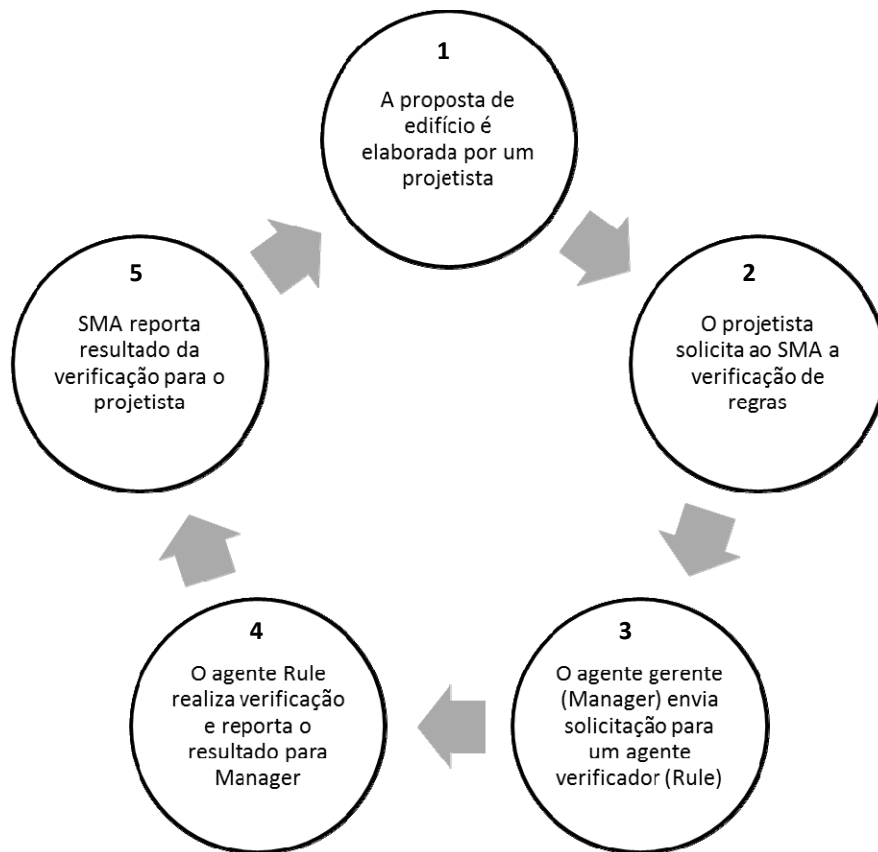


Figura 16: Processo de verificação de regras no SMA proposto.

4.2.1. ONTOLOGIA UTILIZADA

A ontologia SUMO é uma ontologia formal que possui mapeamento para todos os termos da WordNet, uma base de dados de conhecimento linguístico. A partir de qualquer termo desta, é possível identificar facilmente qual o conceito mais próximo descrito pela ontologia. Neste trabalho uma versão em formato OWL (*Web Ontology Language*) da ontologia SUMO foi utilizada para definição dos conceitos utilizados para qualificar as geometrias de um projeto conceitual de edifício (ex.: *Building*, *Door*, *Stairway*, *Room*). Como convenção para o presente trabalho foi definido que o conjunto composto por uma geometria e um conceito definido pela ontologia recebe o nome de “bloco”.

O conjunto de conceitos descritos pela ontologia SUMO é suficientemente amplo para descrever os blocos de um modelo conceitual de edifício. Entretanto, a análise qualitativa das relações espaciais de interesse no presente trabalho demanda o acréscimo de atributos que por convenção foram chamados de “estar dentro de” (*is inside*) e “tocar/estar adjacente a” (*touches*). Em relação ao trabalho de Clementini et al.

(1994), é possível dizer que as relações *coveredBy* e *inside* foram unificadas na relação “*touches*” assim como as relações *covers* e *contains* foram unificadas na relação *is inside*, como proposto pelo trabalho de Grigni et al. (1995). As relações adotadas ficaram, portanto, como descrito a seguir:

- *Touches*: Indica que um indivíduo toca outro indivíduo no espaço (tangenciando ou sobrepondo parcialmente). Ex.: Se uma porta está conectada a uma parede é possível dizer que “porta *touches* parede”;
- *Is Inside*: Indica que todos os pontos que descrevem a geometria de um bloco estão contidos no volume descrito por outro bloco. Ex.: Se um móvel A está dentro de uma sala B, é possível dizer que “móvel A *is inside* sala B”.

Com a adição dos atributos *touches* e *is inside* à ontologia foi possível distinguir de maneira qualitativa as situações como as observadas em cada par de blocos da Figura 17:

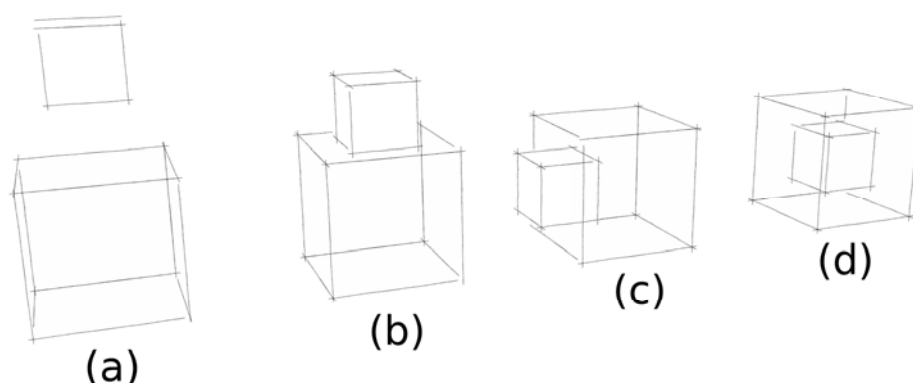


Figura 17: Exemplos de relações espaciais entre blocos: (a) Os dois blocos não se tocam e nenhum está dentro do outro; (b) Os blocos se tocam (*touches*), pois se tangenciam superficialmente; (c) Os blocos também se tocam (*touches*), pois o bloco menor está parcialmente contido no maior; (d) Os blocos não se tocam, mas o menor está completamente dentro do maior (*is inside*).

A utilização da ontologia SUMO no presente trabalho prescindiu da utilização dos axiomas em *Knowledge Interchange Format* (KIF). Como as disciplinas podem ser implementadas de maneira muito distinta, optou-se por deixar que a verificação de axiomas seja realizada pelos próprios no contexto de cada disciplina, que interagem com a ontologia realizando consultas em linguagem *SPARQL Protocol and RDF Query Language* (SPARQL) que tem sido amplamente utilizada no contexto de *web* semântica (KOUBARAKIS et al., 2012). Entretanto, nada impede a utilização desses axiomas em implementações do modelo ora apresentado.

Ferramenta para recorte de ontologias

Como a ontologia SUMO é uma ontologia grande, uma ferramenta para recorte de grandes ontologias foi desenvolvida durante a realização do presente trabalho (Figura 18). A ferramenta é apta a facilitar a manipulação de grandes ontologias, permitindo ao usuário extrair de uma grande ontologia apenas os conceitos de seu interesse, de modo a produzir ontologias menores. Esse recurso foi necessário no trabalho, para o qual foi criada a partir da SUMO uma ontologia contendo apenas conceitos do domínio de projetos de edifícios.

A ferramenta é capaz de manipular grandes ontologias sem a exigência de grande consumo de memória, pois dispensa a necessidade de carregamento completo da ontologia na mesma. Ao invés disso, a ferramenta manipula o conteúdo da ontologia a partir de um banco de dados persistente, solicitando apenas os fragmentos de interesse conforme a demanda do usuário. Os conceitos considerados pertinentes são adicionados a uma lista de conceitos para importação, que posteriormente gerarão um arquivo OWL contendo uma nova ontologia possuindo apenas o conjunto de conceitos de interesse do usuário.

A busca de um conceito de interesse na ontologia original pode ser feita através de palavras-chave que estejam presentes na definição do conceito buscado. Por exemplo, ao digitar o termo “*Door*” na ferramenta de busca o usuário poderá visualizar a definição do conceito de porta. Além disso, quando o usuário inserir o conceito *Door* em sua lista de importação, ele perceberá que o conceito está relacionado aos conceitos *Artifact*, *TransitwayObstacle* e *Device*. Esses três conceitos são, portanto, listados pela ferramenta como dependências do conceito *Door*. O usuário possui um recurso para completar automaticamente dependências como estas, diretamente relacionadas aos termos escolhidos pelo mesmo. Com esses recursos, o usuário é capaz de criar uma lista de conceitos a partir da qual será construída uma nova ontologia.

Sumo / Wordnet3.owl

Filtro

Dependências

Código	Name	File	Dependency
488628	Ceiling	sumo.owl	#stationaryartifact,#axiom-1753627867mid-level-ontology.kif,#axi
499689	Door	sumo.owl	#artifact,#transitwayobstacle,#device,#axiom-1460225469mid-le
505069	Floor	sumo.owl	#stationaryartifact,#axiom-392188366mid-level-ontology.kif,#axio
507141	Furniture	sumo.owl	#artifact,#durablegood,#engineeringcomponent,#machine,#music
544902	Ramp	sumo.owl	#selfconnectedobject,#axiom-650584809mid-level-ontology.kif,#a
545692	ResidentialBuilding	sumo.owl	#building,#residence,#axiom114205418merge.kif
547915	Roof	sumo.owl	#stationaryartifact,#axiom-2047633521mid-level-ontology.kif,wn#
552367	Stairway	sumo.owl	#stationaryartifact,#axiom442609845mid-level-ontology.kif,#axiom
559172	Wall	sumo.owl	#stationaryartifact,#axiom-1018975999mid-level-ontology.kif,#axi
561079	Window	sumo.owl	#artifact,#axiom1834287489mid-level-ontology.kif,#axiom-171343

Showing all 10 rows

Termos

ceiling
door
floor
furniture
residentialbuilding
ramp
roof
stairway
wall
window

Recarregar
Completar Deps.
Download OWL

Quebrados:

Dependências:

stationaryartifact
axiom-1753627867mid-level-ontology.kif
axiom355521288mid-level-ontology.kif
artifact
transitwayobstacle
device
axiom-1460225469mid-level-ontology.kif
axiom-1934810595mid-level-ontology.kif
wn30-102742663
wn30-102812505
wn30-102964934
wn30-103176970
wn30-103221720
wn30-103224032
wn30-103307274
wn30-103399240
wn30-103427296
wn30-104086446
wn30-104329834
axiom-392188366mid-level-ontology.kif

Figura 18: Ferramenta para recorte da ontologia.

4.2.2. ARQUITETURA DO SISTEMA MULTIAGENTES

No modelo proposto, um agente gerente armazena as informações de um modelo de edifício em fase conceitual e envia solicitações de verificação de regras para um conjunto de agentes verificadores de regras. O modelo de edifício é inserido no SMA com a participação do agente *Designer*, que deve possuir interface direta com o usuário, podendo ser esta interface, por exemplo, um sistema *web* que recebe modelos de edifício. O agente gerente, também chamado *Manager*, é responsável por armazenar todos os dados de projeto necessários para o processo de verificação de regras, além de armazenar os resultados das verificações realizadas.

O ambiente dos agentes é o próprio conjunto de blocos que descrevem o edifício e os agentes navegam em meio a esses blocos sinalizando e informando ao agente gerente aqueles que violam suas regras conhecidas. Apenas o agente gerente conhece o projeto por completo, portanto, os demais agentes ampliam sua percepção a respeito do ambiente conforme navegam pelo edifício. Uma vez que os agentes compartilham um mesmo ambiente no modelo de edifício, é possível que o agente responsável por uma disciplina específica seja influenciado pelas modificações no ambiente provenientes da verificação de regras realizada por outra disciplina qualquer.

Os agentes propostos no modelo se dividem, portanto, em três categorias: *Manager*, *Designer* e *Rule*. Apenas um agente assume o papel de *Manager*, entretanto vários agentes podem assumir o papel de verificadores de regras, como mostrado na Figura 19.

Em suma, é possível dizer que:

- O agente *Designer* é responsável por fazer a interface entre os *Designers* e o SMA;
- O agente *Manager* coordena o trabalho dos outros agentes gerenciando o compartilhamento de informações dos agentes entre si e com os projetistas; e
- Os agentes *Rule* (*Rule1*, *Rule2*, ..., *RuleN*) são responsáveis por verificar regras ou conjuntos de regras específicas em modelo de edifício. Como não conhecem todo o projeto, esses agentes recebem sob demanda os blocos do edifício necessários para realização de suas verificações.

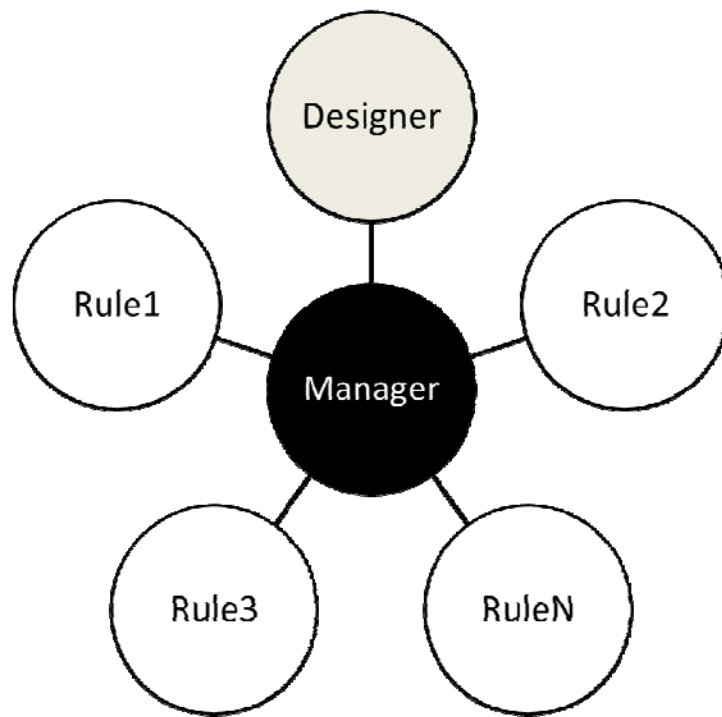


Figura 19: Conjunto de agentes do modelo proposto.

4.2.3. COMUNICAÇÃO ENTRE OS AGENTES

Os agentes no modelo proposto são todos capazes de se comunicar diretamente com o agente gerente, mas os agentes responsáveis por diferentes disciplinas não são capazes de se comunicar diretamente entre si. Essa decisão foi tomada sob a premissa de que se uma informação é pertinente a mais de uma disciplina, ela também pode ser pertinente ao agente gerente. Portanto, toda comunicação entre agentes de diferentes disciplinas é sempre mediada pelo agente gerente.

As mensagens trocadas entre os agentes são compostas por duas informações básicas: o cabeçalho da mensagem e um conjunto de blocos. O cabeçalho indica o remetente, destinatário, o modo como a solicitação está sendo realizada (performativo) e uma ação, que serve para informar ao destinatário o que fazer com o conteúdo da mensagem. No SMA utilizado para realização do estudo de caso da presente pesquisa esse conteúdo foi codificado em JSON (*JavaScript Object Notation*), um padrão adequado dentro das especificações FIPA, segundo Moulin et al. (2011).

4.2.4. AGENTE DESIGNER

O agente *Designer* possui comunicação direta com o projetista e com o agente gerente. Como o projeto é manipulado pelo gerente como um conjunto de blocos, o agente designer pode receber modelos virtuais de edifício em formatos amplamente utilizados pela indústria, como *COLLADA* (BARNERS e FLINCH, 2008) e *Wavefront* (MURRAY e VAN RYPER, 2005) e convertê-los em um conjunto de blocos. Para isso, o agente *Designer* pode utilizar a própria configuração das geometrias no espaço para distinguir os diferentes blocos. A atribuição de conceitos a esses blocos pode ser feita com base no material que cada bloco recebe, fazendo-se necessário apenas que o usuário estabeleça ou adote para isso uma convenção. No estudo de caso deste trabalho, por exemplo, uma paleta de cores foi definida *a priori* de modo que as cores dos blocos são utilizadas para relacionar conceitos às geometrias.

4.2.5. AGENTE MANAGER

O agente *Manager* possui uma base de dados persistente onde pode armazenar um grande volume de fatos, que podem ser fornecidos sob demanda aos demais agentes. O *Manager* possui em sua mente uma estrutura interna do modelo de edifício que organiza as relações entre todos os blocos conforme suas relações *touches* e *is inside*.

A relação *is inside* é utilizada pelo agente *Manager* para organizar todos os blocos do edifício em uma hierarquia. A relação é transitiva, pois se B está dentro de A, tudo o que estiver dentro de B também estará dentro de A. A análise dessa relação sobre todos os blocos de um edifício conduz ao surgimento de um grafo em árvore capaz de conter todos os blocos do edifício.

Para garantir que o grafo de relações *is inside* irá se configurar como uma árvore, convencionou-se que cada bloco assume como pai apenas o bloco de menor volume que o contém. Como não é possível que um bloco de maior volume esteja completamente no interior de um bloco de menor volume, essa escolha conduzirá à criação de uma árvore que tenderá a descrever apenas os relacionamentos mais relevantes a serem considerados sob o aspecto de “estar dentro de”. Por exemplo: se um bloco cadeira C estiver dentro de uma sala S que pertence a um edifício E, é mais relevante dizer que C *is inside* S do que dizer que C *is inside* E, pois a relação entre C e

E pode ser inferida a partir da relação entre C e S, mas a relação entre C e S não pode ser inferida a partir da relação entre C e E. Além disso, todo bloco que não estiver dentro de nenhum outro estará dentro do nodo raiz da árvore. Logo, se um bloco A não está dentro de nenhum outro bloco, “A is inside raiz”. A Figura 20 ilustra um exemplo de estruturação de edifício construído a partir das relações *is inside*.

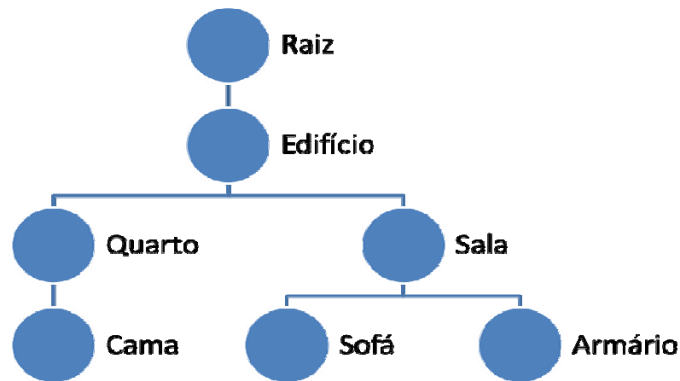


Figura 20: Exemplo de árvore de relações *is inside*. Neste grafo os blocos são filhos do bloco de menor volume capaz de conter por completo toda sua geometria.

Já a relação *touches* é uma relação simétrica utilizada pelo agente gerente para identificação de adjacência e conectividade entre os elementos do espaço. O uso de inferências para determinação de relações de adjacência permite a um agente compreender a topologia de um edifício, como se observa na Figura 21. Assim como em Dong e Brück (2011), o presente trabalho aborda a utilização da relação de adjacência para navegação no espaço, sendo esta uma relação fundamental para o processo de verificação de regras proposto neste trabalho.

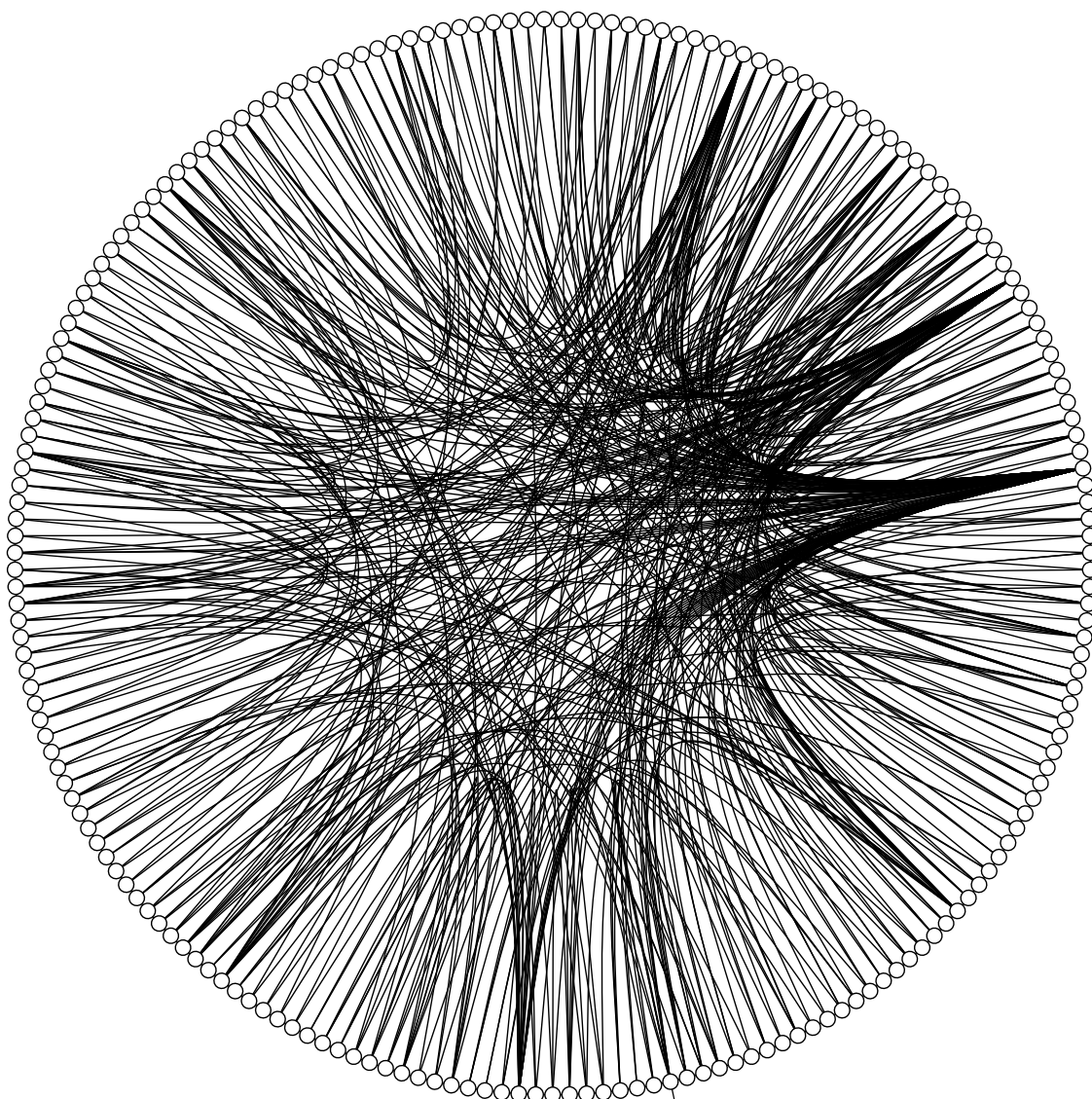


Figura 21: A figura acima apresenta as relações de adjacência entre os blocos do edifício. Os pequenos círculos brancos que contornam a figura representam as 199 geometrias que compuseram o modelo de edifício utilizado no estudo de caso do presente trabalho. As linhas conectam os blocos que possuem relação *touches* entre si.

A partir da estrutura do modelo de edifício construída pelas relações *touches* e *is inside*, o agente *Manager* é capaz de responder a requisições como “forneça os blocos adjacentes ao bloco X” ou “forneça os blocos no interior do bloco Y”, o que é suficiente para a navegação dos agentes verificadores de regras no edifício.

4.2.6. AGENTE VERIFICADOR DE REGRAS

Enquanto o *Manager* gerencia uma grande base de fatos, os agentes verificadores de regras, chamados *Rule*, tendem a manipular informações menores, restritas ao domínio dos problemas específicos que eles resolvem. Um agente *Rule* avalia as partes que compõem um projeto à luz de uma regra ou de um conjunto de regras de seu domínio.

Na abordagem proposta, o agente realiza o processo de verificação automática de regras explorando o conteúdo do edifício pouco a pouco, conforme o ciclo mostrado na Figura 22. O mecanismo adotado é chamado neste trabalho de “mecanismo iterativo de verificação de regras”, pois o agente solicita novas informações a respeito do edifício iterativamente conforme avança no processo de verificação. Isso evita que um agente receba informações desnecessárias durante o processo de verificação, além de contribuir para a realização de verificação de regras em grandes modelos de edifício.

O processo de verificação de regras proposto é também um processo de navegação pelos blocos de um edifício. O agente tentará visitar todos os blocos do edifício em um processo de busca por profundidade. Os deslocamentos válidos dentro do edifício durante essa navegação dependem da conectividade entre os blocos e de um conjunto de regras internas do agente chamadas Regras de Navegação.

As Regras de Navegação podem ser implementadas por cada agente de uma maneira distinta e visam permitir ao agente simular o deslocamento de um tipo específico de indivíduo (ex.: idoso, mulher grávida, cadeirante, criança). Um agente verificador de regras pode simular, por exemplo, o deslocamento de um cadeirante e para isso obedecerá restrições específicas. Por exemplo: o agente poderá não caminhar de um bloco para outro quando a diferença de nível entre os blocos gerar um degrau difícil de transpor com uma cadeira de rodas. Além das questões geométricas, as regras de navegação também consideram os conceitos relacionados aos blocos. Por exemplo: a ontologia utilizada pelo SMA pode definir vários tipos de escada, atribuindo conceitos específicos para cada uma delas. O agente pode, portanto, possuir como regra de navegação a incapacidade de subir qualquer bloco que carregue o conceito “escada”.

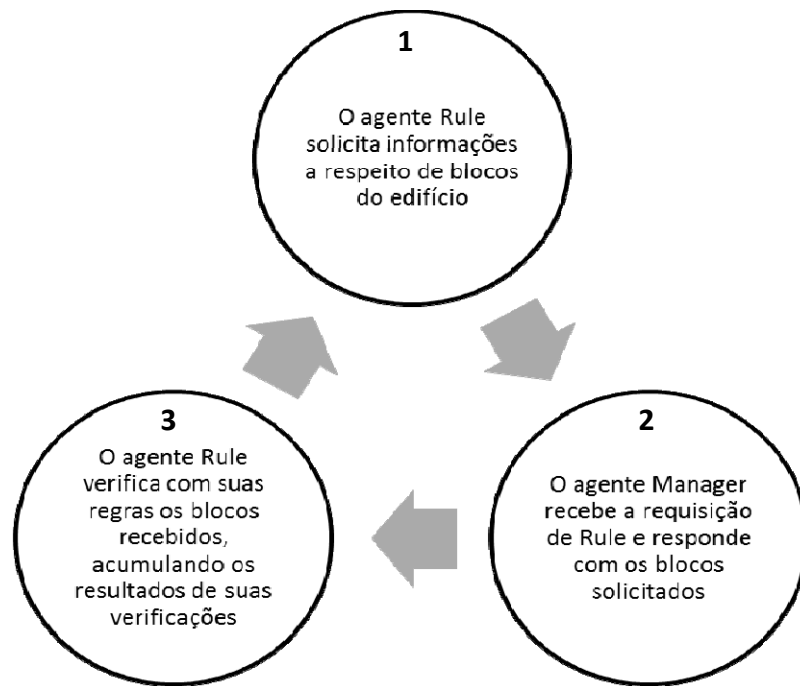


Figura 22: Ciclo de verificação iterativa de regras. O ciclo termina quando Rule acredita que já visitou todos os blocos necessários para realizar sua verificação automática de regras.

O processo de verificação de regras de um agente *Rule* se inicia com a escolha de um ponto inicial de sua trajetória. A partir desse momento, o agente navega pelo edifício visitando todos os blocos que conseguir alcançar. Enquanto um modelo do edifício é construído de maneira incremental em sua mente, o agente acumula críticas a respeito dos blocos visitados. Na primeira iteração do processo, o agente conhece um único bloco e sua vizinhança. Após iniciado o processo de verificação, o agente passa a caminhar e conhecer novos blocos. O processo de verificação ocorre, portanto, da seguinte maneira:

- a) *Início:* O agente está no bloco inicial;
- b) *Observação do entorno:* O agente solicita ao *manager* informações sobre todos os blocos adjacentes ao bloco atual. Quando receber a resposta, passa para o próximo passo;
- c) *Verificação de regra:* O agente já conhece o entorno do bloco atual e, portanto, já está apto a realizar a verificação de regras no mesmo. O agente verifica um conjunto de regras no bloco atual e, se o bloco violar alguma de suas regras internas, o bloco é colocado em uma lista contendo todos os blocos conhecidos que violam a mesma regra;

- d) *Deslocamento*: O agente tenta caminhar para o próximo bloco, que deverá ser adjacente ao bloco atual e ser um bloco não visitado. O deslocamento de um bloco para outro não pode infringir as regras de deslocamento do agente. Todo bloco que infringir as regras de deslocamento é adicionado a uma lista chamada “Candidatos Não Alcançáveis”. Se algum dos blocos adjacentes for considerado alcançável, o agente se desloca para ele e repete o item (b). Se nenhum bloco adjacente for alcançável, o agente realiza a operação de *backtracking*, descrita no próximo passo;
- e) *Backtracking*: Se o agente verificar que se encontra no primeiro bloco visitado no edifício, ele passa para o próximo item. Entretanto, se o agente observar que o bloco não é o mesmo de seu ponto de partida, ele retorna ao último bloco conhecido que apresentou mais de um vizinho visitável. Feito isso, o agente repete o passo (b);
- f) *Visita aos blocos não alcançáveis*: O agente visita o primeiro candidato não alcançável e navega pelos blocos a partir dos blocos não alcançáveis, à semelhança do mecanismo de navegação descrito anteriormente. A diferença na navegação do agente nesse momento está no fato de que ele irá caminhar de um bloco para qualquer outro bloco adjacente, mesmo que isso infrinja suas regras de navegação. Todo bloco visitado nesse contexto recebe uma marcação indicando se tratar de um bloco não alcançável. Quando não houver mais blocos a serem visitados, o agente tentará visitar o próximo candidato não alcançável. O processo se repetirá até que se esgote a lista de candidatos não alcançáveis. Quando todos os blocos percebidos pelo agente tiverem sido visitados, ele passará para o próximo passo;
- g) *Emissão de Relatório*: O agente emite um relatório listando os blocos que infringiram cada uma das regras conhecidas e lista também os blocos marcados como “não alcançáveis”.

Finalmente, o parecer emitido pelo agente ao final do processo de verificação consiste em conjuntos de blocos com a indicação das regras violadas em cada um deles e, adicionalmente, com a indicação dos blocos que não podem ser alcançados segundo as regras de navegação do agente.

4.2.7. RACIOCÍNIO INTERNO DOS AGENTES

Os agentes raciocinam a respeito do edifício como um conjunto de blocos que estabelecem entre si relações *touches* e *is inside*. Os agentes compartilham a utilização da mesma ontologia para acesso à definição dos conceitos manipulados. Como o modelo proposto foi concebido para representar conhecimentos de diferentes disciplinas, o raciocínio interno dos agentes pode assumir implementações muito distintas. No estudo de caso apresentado por este trabalho, o raciocínio dos agentes manipula as dimensões lineares e os conceitos presentes nos blocos do edifício para efetuar a verificação automática de regras. Entretanto, os agentes podem utilizar qualquer modelo cuja utilização se mostre viável para manipulação do conhecimento de uma disciplina, como modelos de otimização na análise de edifícios, a exemplo de Ren et al. (2011).

5. ESTUDO DE CASO: VERIFICAÇÃO AUTOMÁTICA DE REGRAS DE ACESSIBILIDADE PARA CADEIRANTES NO NOVO EDIFÍCIO DO CCE DA UFV

5.1. OBJETIVO

O objetivo do estudo de caso foi validar o modelo SMA proposto no capítulo anterior, a partir da verificação de regras num modelo conceitual de um edifício real.

5.2. METODOLOGIA

O estudo de caso foi realizado utilizando-se o SMA desenvolvido conforme descrito no capítulo anterior, e um projeto de edifício anexo ao atual edifício do Centro de Ciências Exatas (CCE) no Campus da UFV. As regras verificadas sobre o modelo de edifício foram baseadas nas regras de acessibilidade da *Accessibility Guidelines for Buildings and Facilities* (ADAAG). A acessibilidade a edifícios é um desafio significativo para a tomada de decisão multidisciplinar durante o processo de projeto (EASTMAN et al., 2009a), motivo pelo qual foi escolhida neste trabalho.

5.2.1. ESCOLHA DAS REGRAS

Para fins de demonstração, um conjunto de regras da ADAAG foi simplificado de modo a produzir duas regras que permitiram a realização de testes no trabalho proposto:

Regra I – Dimensões mínimas

- Um ambiente deve possuir dimensões mínimas maiores do que 1.53m, visando permitir mudanças de direção da cadeira de rodas, conforme item 4.2.3 da ADAAG (US ACCESS BOARD, 1998).

- Portas não podem ter menos de 80cm de largura, conforme item 4.13 da ADAAG (US ACCESS BOARD, 1998).

Regra II – Circulação vertical

- Pelo menos uma dentre todas as possíveis rotas de acesso a um lugar no edifício deve prescindir do uso de escadas. Caso contrário, esta parte será inacessível para cadeirantes conforme o item A4.8.1 da ADAAG (US ACCESS BOARD, 1998).

A Regra I foi escolhida por demonstrar duas verificações sob as quais cada elemento no espaço é avaliado individualmente (Figura 23 A). Já a Regra II foi adotada por ser uma regra capaz de demonstrar uma situação onde os elementos no espaço são aprovados ou reprovados não apenas de acordo com seus atributos individuais, mas pelas relações que estabelecem com os outros elementos no espaço (Figura 23 B).

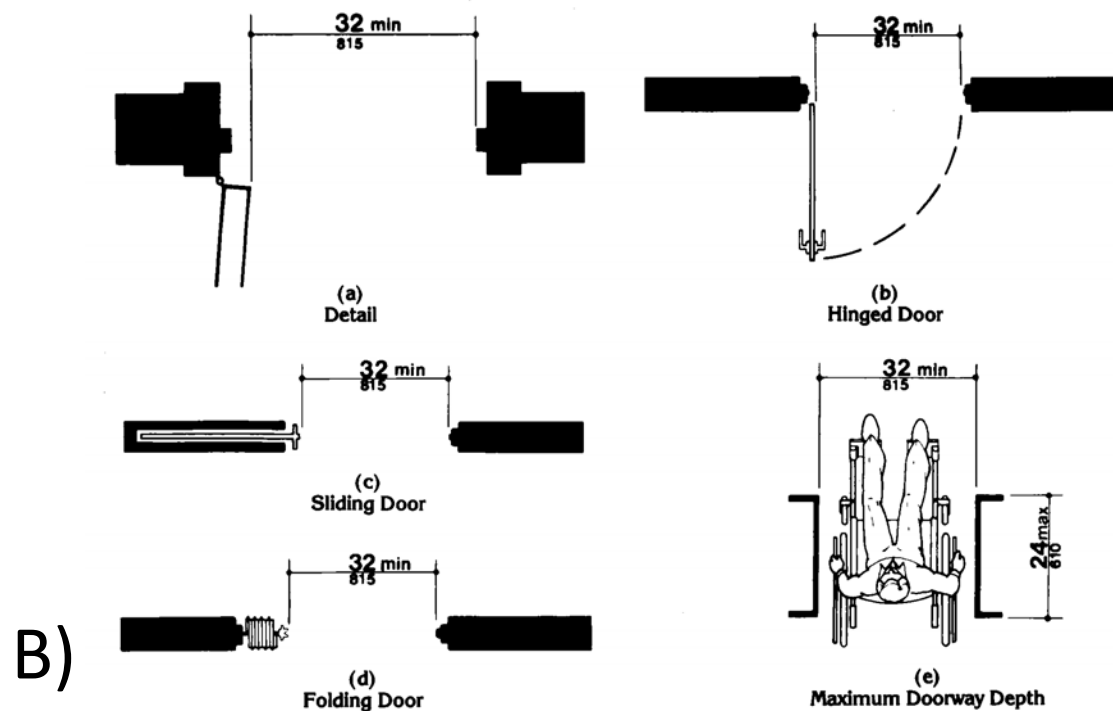
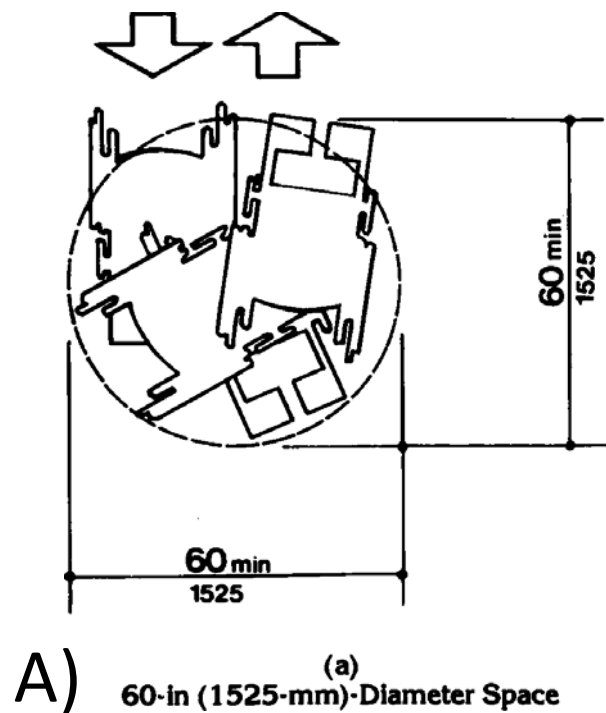


Figura 23: Ilustração das regras da ADAAG utilizadas neste trabalho. A) Espaço livre para mudança de direção com cadeira de rodas; B) Dimensões mínimas para portas. Fonte: US Access Board (1998).

5.2.2. ESCOLHA DO EDIFÍCIO

O edifício escolhido para este estudo de caso foi o projeto do novo prédio do Centro de Ciências Exatas (CCE) da UFV. Este foi escolhido por se tratar de um edifício de múltiplos pavimentos, o que implica em um edifício relevante para a verificação de regras de acessibilidade. O fato de se tratar de uma das obras mais recentes do Campus da UFV facilitou o acesso ao projeto, cuja versão data de 28 de agosto de 2012, podendo ter sofrido mudanças desde então. Além disso, trata-se de um edifício anexo ao edifício onde se encontra o Departamento de Informática da UFV, onde foi conduzido este trabalho.

O modelo de edifício conceitual foi completamente representado com 199 blocos distintos, para o qual foram disponibilizados quatro conceitos distintos: *Room*, *Door*, *Stairway* e *Building*. Os elevadores do edifício foram suprimidos propositalmente para que fosse verificada no agente a capacidade de detecção de regras de circulação vertical; portanto, os problemas de acessibilidade identificados se aplicam ao modelo conceitual de edifício proposto para testes, e não ao edifício real.

5.2.3. APLICAÇÃO DO MODELO MULTIAGENTES

O edifício foi modelado em um arquivo 3D no formato *Wavefront*, no qual um padrão de cores para os materiais aplicados às geometrias indicou o conceito da ontologia presente em cada bloco do edifício. Para este estudo de caso, as geometrias foram todas representadas com prismas de base retangular e as cores adotadas para representação dos conceitos *Door*, *Stairway*, *Room* e *Building* foram, respectivamente: cinza, azul, amarelo e branco.

O sistema multiagentes leu os dados presentes no arquivo *Wavefront*, identificou automaticamente as relações de adjacência (*touches*) e relações dentro e fora (*is inside*) entre os blocos do edifício, e foi capaz de preparar de maneira autônoma as informações necessárias para o processo de verificação de regras. O agente percorreu o modelo de edifício visitando todos os seus blocos. O grafo da Figura 24 mostra os blocos visitados pelo agente durante o processo de verificação de regras.

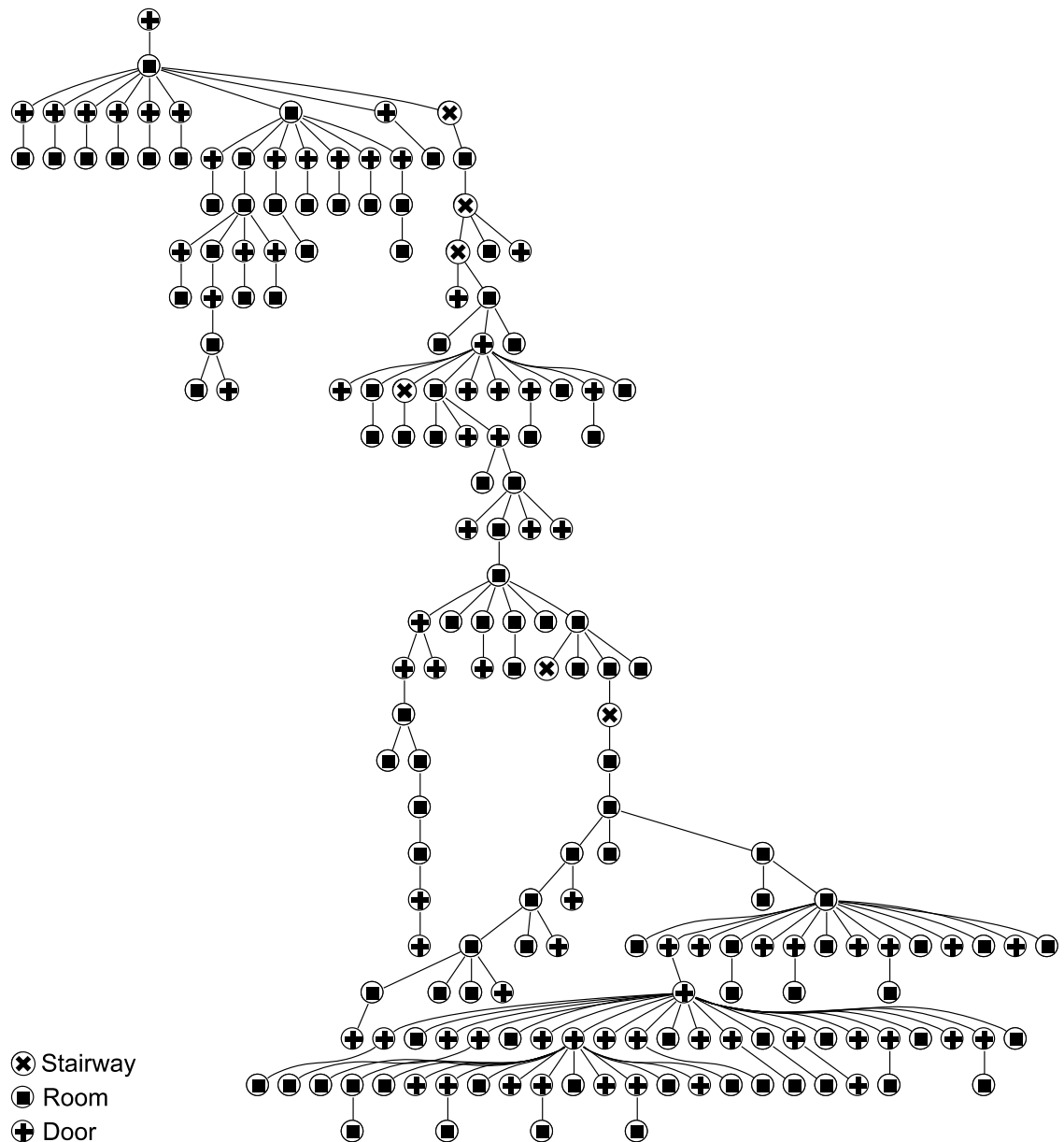


Figura 24: O grafo acima representa os blocos visitados pelo agente verificador de regras. A leitura do grafo em sequência de busca por profundidade mostra a ordem em que os blocos foram visitados durante o processo.

5.3. RESULTADOS E DISCUSSÃO

O SMA desenvolvido neste trabalho foi capaz de identificar 100% das relações de adjacência e das relações dentro/fora de maneira automática no modelo de edifício. Assim como em Martins e Monteiro (2013), a análise estatística dos resultados obtidos em contraste com métodos tradicionais não foi considerada algo possível, pois a verificação automática de regras modifica o processo de projeto, tornando difícil a

comparação entre os resultados obtidos com esta abordagem e os resultados obtidos em processos tradicionais.

O sistema realizou a verificação automática de regras em 199 blocos em aproximadamente 1,13 segundo em um computador Core I5 2.40Ghz com 4GB de memória RAM. O *Bootstrap* do sistema multiagentes demorou, em média 4,86 segundos (Tabela 1).

Tabela 1: Duração média de execução dos processos de *Bootstrap* e verificação de regras.

PROCESSO	DURAÇÃO MÉDIA (segundos)
<i>Bootstrap</i>	4,86
Verificação das regras I e II em 199 geometrias	1,13

Ao final do processo de verificação, o sistema emitiu o parecer indicando os blocos que infringiram as regras conhecidas. O número de blocos que violou cada uma das regras está quantificado na Tabela 2.

Tabela 2: Resultados do processo de verificação automática de regras

Regra verificada	Blocos que infringiram a regra
Regra I – Dimensões mínimas	30
Regra II – Circulação vertical	139

Na Figura 25 são exibidos: A) o modelo de edifício; B) o resultado da verificação da Regra I e C) o resultado da verificação da regra II. Os blocos marcados em vermelho foram aqueles que infringiram as regras analisadas. O *feedback* visual do sistema utilizado no estudo de caso pode ser complementado com informações textuais, o que não foi julgado necessário neste exemplo.

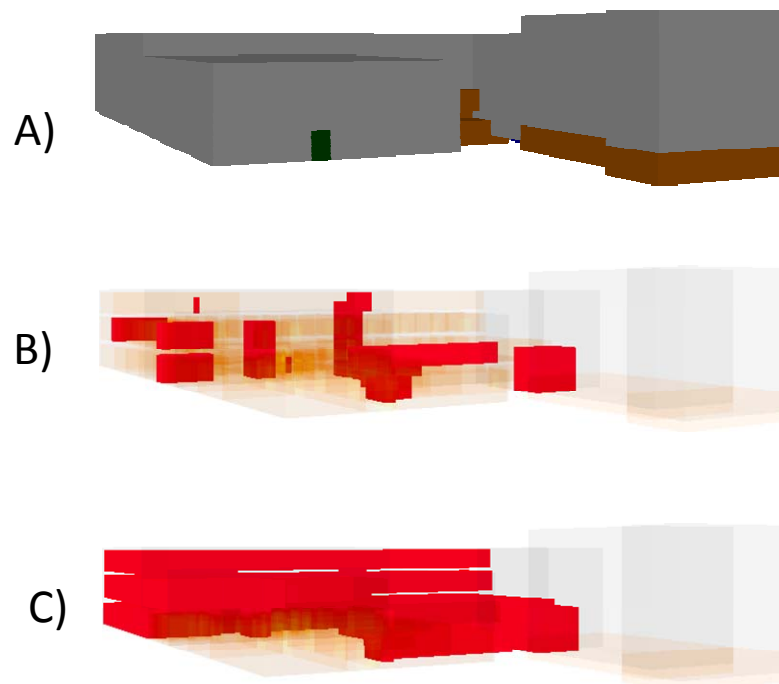


Figura 25: Resultado do processo automático de verificação: A) Modelo do edifício; B) Identificação automática de elementos que violam a Regra I (dimensões mínimas); C) Identificação automática de elementos que violam a Regra II (circulação vertical).

6. CONCLUSÕES E PERSPECTIVAS

Este trabalho demonstrou a viabilidade da verificação de regras em modelos de edifícios representados com uso de ferramentas de CAD 3D tradicional. O modelo proposto utilizou uma estrutura de dados onde qualquer elemento de um edifício pode ser expresso com blocos que relacionam cada geometria do edifício a um conceito definido por uma ontologia.

O modelo se mostrou capaz de trabalhar de maneira interoperável, por manipular a informação no nível de detalhamento encontrado em arquivos meramente geométricos como *COLLADA* ou *Wavefront*, amplamente utilizados pela indústria no contexto de CAD 3D e suficientemente expressivos para atrelar conceitos às geometrias a partir de seus materiais, sob a abordagem proposta. Como a atribuição de materiais a geometrias é um processo intrínseco à atividade de modelagem de edifícios, é possível dizer que o processo de verificação de regras proposto não exige entrada adicional de parâmetros manuais visando a essa verificação. Apesar da grande contribuição do BIM para a automação de processos em projetos de edifícios, é importante considerar que a manipulação de representações mais simples pode contribuir para que a indústria realize uma transição suave para os processos futuros.

Quanto à abordagem com uso de sistemas multiagentes, a decisão de modelar um agente para cada disciplina confere ao modelo condições para receber a implementação de regras de diferentes disciplinas de um modo adaptável ao longo do tempo, possibilitando a remoção e adição de novos agentes sem que isso comprometa o funcionamento do sistema. A facilidade de utilização por diferentes disciplinas pode ser obtida com a utilização de um mecanismo de destaque das geometrias que infringem regras, a exemplo de Nosyko (2012).

Com o estudo de caso realizado, observou-se que um sistema multiagentes conforme a abordagem proposta é capaz de realizar a verificação automática de regras em modelos de edifício meramente geométricos. O exemplo demonstrou também que a verificação de regras pode ser realizada em tempo satisfatório com o uso de um computador convencional, quando da realização do processo sobre um modelo conceitual de edifício de médio porte.

Uma sugestão para futuras pesquisas é a criação de métricas que auxiliem a avaliação dos mecanismos de verificação automática de regras do ponto de vista da interface com o usuário, amparando a realização de comparações entre diferentes

ferramentas propostas pelas futuras pesquisas nesta área. Outra sugestão seria a realização de estudos que abordem a integração de regras de diferentes países em um mesmo sistema multiagentes.

O modelo proposto pode ser implementado como uma ferramenta *web* com a qual os projetistas possam enviar seus projetos e receber relatórios a respeito dos mesmos. Além disso, a ferramenta pode ser utilizada em tempo real pelos projetistas durante o processo de representação de modelos de edifícios. Com a implementação de modelos estatísticos, o sistema pode trazer ao conhecimento do projetista as decisões tomadas por outros pares ao resolverem problemas semelhantes aos detectados em seu projeto. Além disso, a verificação automática de regras também pode ser abordada de maneira mais ampla com raciocínio qualitativo espaço-temporal, o que já tem se apresentado como um recurso para o auxílio à evacuação de prédios em situações de catástrofe.

REFERÊNCIAS BIBLIOGRÁFICAS

ATTIA, S.; GRATIA, E.; DE HERDE, A.; HENSEN, J. L. Simulation-based decision support tool for early stages of zero-energy building design. **Energy and buildings**. v.49, p. 2-15, jun. 2012.

AZHAR, S.; HEIN, M.; SKETO, B. Building information modeling (BIM): Benefits, Risks and Challenges. In: Proceedings of the 44th ASC Annual Conference. 2008, Auburn, **Anais eletrônicos...** 2008. p. 2-5. Disponível em: <<http://ascpro.ascweb.org/chair/paper/CPGT182002008.pdf>>. Acesso em: 7 jan. 2013.

BARNERS, M.; FINCH, E. L. **Collada-digital asset schema release 1.4. 1**. California, 2008. 378 p. Disponível em: <http://www.khronos.org/files/collada_spec_1_5.pdf>. Acesso em: 7 jan. 2013.

BEETZ, J. **Facilitating distributed collaboration in the AEC / FM sector using Semantic Web Technologies**. 2009. 153f. Tese (Doutorado em Arquitetura) – Eindhoven University of Technology, Eindhoven, Holanda, 2009.

BELL, H.; BJØRKHAUG, L.; HJELSETH, E. **Standardized Computable Rules**. Noruega: Standards Norway, 2009. Disponível em: <http://www.standard.no/Global/PDF/Bygg,%20anlegg%20og%20eiendom/Rule_Checking_Report%20web%20stor%20fil.pdf>. Acesso em: 28 jan. 2010.

BERMUDEZ, L. **Is the OGC playing with Linked Data?**. Disponível em: <<http://www.opengeospatial.org/blog/1673>>. Acesso em: 11 nov. 2012.

BERNERS-LEE, T.; HENDLER, J.; LASSILA, O. The Semantic Web. **Scientific American**. New York, 17 mai. 2001. **Scientific American**. Disponível em: <<http://www.scientificamerican.com/article.cfm?id=the-semantic-web>>. Acesso em: 13 jan. 2013.

BIZER, C.; JENTZSCH, A.; CYGANIAK, R. **State of the LOD Cloud**. 2011. Disponível em: <<http://www4.wiwi.fu-berlin.de/locloud/state>>. Acesso em: 28 jan. 2013.

CAVIERES, A.; GENTRY, R.; AL-HADDAD, T. (2011). Knowledge-based parametric tools for concrete masonry walls: Conceptual design and preliminary structural analysis. **Automation in Construction**, v. 20, n.6, p. 716-728. fev. 2011.

CLEMENTINI, E.; SHARMA, J.; EGENHOFER, M.J. Modelling topological spatial relations: Strategies for query processing. **Computers & Graphics**. v. 18, n. 6, p. 815-822. 1994. Disponível em: < <http://www.spatial.maine.edu/~max/9intooptimization.pdf>>. Acessado em: 20 jan. 2013.

COHN, A. G.; RENZ, J. Qualitative Spatial Representation and Reasoning. **Foundations of Artificial Intelligence**. v.3, p. 551-596. 2008.

DE VRIES, M.; WAGTER, H. (1990). A CAAD model for use in early design phases. In: MCCULLOUGH, M.; MITCHELL, W. J.; PURCELL, P. **The Electronic Design Studio: Architectural Education in the Computer Era**. MIT Press, 1990. v. 1, cap. 14, 215-228.

DONG, T.; BRÜCK, T. Qualitative Spatial Knowledge Acquisition Based on the Connection Relation. In: COGNITIVE 2011, The Third International Conference on Advanced Cognitive Technologies and Applications. Rome. **Anais...** 2011. p.70-75.

EASTMAN, C.; LEE, J.-M.; JEONG, Y.-S.; LEE, J.-K. Automatic rule-based checking of building designs. **Automation in Construction**. v. 18. n. 8, p. 1011-1033. 2009a.

EASTMAN, C.; LEE, J.-K.; SHEWARD, H.; SANGUINETTI, P.; JEONG, Y.-S.; LEE, J.; ABDELMOHSEN, S. Automated Assessment of Early Concept Designs. **Architectural Design**. v. 79, n. 2, p. 52-57. 2009b.

FAUS, J.; GRIMALDO, F.; BARBER, F. Multiagent system for detecting and solving design-time conflicts in civil infrastructure. In: **Trends in Practical Applications of Agents and Multiagent Systems**. Salamanca:Springer Berlin Heidelberg, 2012. v. 1, p. 57-64.

FISCHER, G.; NAKAKOJI, K.; OSTWALD, J.; STAHL, G.; SUMNER, T. Embedding Critics in Design Environments. In: MAYBURY, M. T. WAHLSTER, W. **Readings in Intelligent User Interfaces**. San Francisco:Morgan Kaufmann, 1998. p. 537-561.

FU, C.; AOUAD, G.; LEE, A.; MASHALL-PONTING, A.; WU, S. IFC model viewer to support nD model application. **Automation in Construction**, v. 15, n. 2, p.178-185. 2006.

GANE, V.; HAYMAKER, J. Benchmarking current conceptual high-rise design processes. **Journal of Architectural Engineering**. v. 16, n. 3, p. 100-111. 2009.

GRAY, C.; HUGHES, W. **Building design management**. Woburn:Routledge, 2012.

GRIGNI, M.; PAPADIAS, D.; PAPADIMITRIOU, C. Topological Inference. In International Joint Conference on Artificial Intelligence. 1995. **Anais...** Lawrence Erlbaum Associates LTD. 1995. v. 14, p. 901-907.

HETHERINGTON, R.; LANEY, R.; PEAKE, S.; OLDHAM, D. Integrated building design, information and simulation modelling: the need for a new hierarchy. In: Building Simulation. 2011, Sydney, **Anais ...** 2011. p. 14-16. Disponível em: <http://oro.open.ac.uk/29470/1/P_1708_V2.pdf>. Acesso em: 2 jan. 2013.

HOGENBOOM, F.; FRASINCAR, F.; KAYMAK, U. A Review of Approaches for Representing RCC8 in OWL. In: Proceedings of the 2010 ACM Symposium on Applied Computing. 2010, Sierre:ACM, **Anais...** 2010. p. 1444-1445. ACM, 2010.

KOUBARAKIS, M.; KARPATHIOTAKIS, M.; KYZIRAKOS, K. Data Models and Query Languages for Linked Geospatial Data. In: EITER, T.; KRENNWALLNER, T. **Reasoning Web - Semantic Technologies for Advanced Query Answering**. Vienna: Springer Berlin Heidelberg, 2012.v. 1, p. 290-328. 2012.

KULUSJÄRVI, H.; WIDNEY, J. **Solibri Model Checker: for better quality & Higher Accuracy - Introducing Deficiency Detection**. 2010. 5 p. Disponível em: <<http://www.solibri.com/documents/english-material/introducing-deficiency-detection/download.html>>. Acesso em: 01 nov. 2010.

LACLAVÍK, M. **Ontology and Agent based Approach for Knowledge Management**. 2005. 121 f. Tese (Doutorado em Informática) – Slovak Academy of Sciences, 2005.

LEE, J., LEE, J.; JEONG, Y.; SHEWARD, H.; SANGUINETTI, P.; ABDELMOHSEN, S.; EASTMAN, C. M. Development of space database for automated building design review systems. **Automation in Construction**, v. 24, p. 203-212. 2012.

MACAL, C. M.; NORTH, M. J. Tutorial on agent-based modelling and simulation. **Journal of Simulation**, v. 4, n. 3, p. 151-162. 2010.

MARTINS, J. P.; MONTEIRO, A. LicA: A BIM based automated code-checking application for water distribution systems. **Automation in Construction**. v. 29, n. 23, p. 12-23. 2013.

MASCARDI, V.; CORDÌ, V. A.; ROSSO, P. Comparison of Upper Ontologies. In: Proc. Conf. on Agenti e industria: Applicazioni tecnologiche degli agenti software, WOA07, 2007. **Anais ...** Genova: Seneca Edizioni Torino, 2007. p. 55-64.

MOULIN, C.; JONES, A.; BARTHÈS, J. P.; LENNE, D. Preliminary Design on Multi-touch surfaces Managed by Multi-agent Systems. **International Journal of Energy, Information and Communications**, v. 2, n. 4, p. 195-210. 2011.

MOREIRA, A.; ALVARENGA, L.; OLIVEIRA, A. P. Thesaurus and Ontologies: a study over the definitions found in the computer and information science literature, by means of analytical-synthetic method. **Knowledge Organization**, Alemanha, v. 31, n. 4, p. 231-244. 2004.

MURRAY, J.; VAN RYPER, W. **Wavefront OBJ File Format Summary**. Disponível em: <<http://www.fileformat.info/format/wavefrontobj/egff.htm>>. Acesso em: 2013-02-06.

NEBEL, B.; FREKSA, C. AI Approaches to Cognitive Systems – The Example of Spatial Cognition. **Informatik-Spektrum**. v. 34, n. 5, p. 462–468. 2011.

NILES, I.; PEASE, A. Towards a standard upper ontology. In: Proceedings of the international conference of formal ontology in information systems. Ogunquit:ACM. **Anais...** 2001. p. 2-9.

NOSYKO. **dRofus User Manual**. Oslo, 2012. 89 p. Disponível em: <<http://www.drofus.no>>. Acesso em: 16 jan. 2013.

OGC. **OpenGIS City Geography Markup Language (CityGML) Encoding Standard, Version 2.0. 0, International OGC Standard**. 2012, 344 p. Disponível em: < www.opengeospatial.org/standards/citygml>. Acesso em: 27 set. 2012.

PAULHEIM, H.; PROBST, F. Ontology-enhanced user interfaces: A survey. **International Journal on Semantic Web and Information Systems (IJSWIS)**, v. 6, n. 2, p. 36-59. 2010.

REN, Z.; YANG, F.; BOUCHLAGHEM, N. M.; ANUMBA, C. J. (2011). Multi-disciplinary collaborative building design—A comparative study between multi-agent systems and multi-disciplinary optimisation approaches. **Automation in Construction**, v. 20, n. 5, p. 537–549. 2011.

RENZ, J.; NEBEL, B. Qualitative spatial reasoning using constraint calculi. In: AIELLO, M. PRATT-HARTMANN, I. BENTHEM, J. V. **Handbook of spatial logics**. Springer Berlin Heidelberg. 2007. cap. 1. p. 161-215.

RUSSELL, S.; NORVIG, P. **Artificial Intelligence: A Modern Approach**. 3. ed. third edition. Nova Jersey: Prentice Hall, 2010. 1132 p.

TREECK, C. V.; WENISCH, P.; PFAFFINGER, M.; EGGER, M.; WENISCH, O.; RANK, E. Towards Interactive Indoor Thermal Comfort Simulation. In European Conference on Computational Fluid Dynamics, 2006, Delft. **Anais...** Delft. 2006. p. 1-15.

US ACCESS BOARD. **Appendix A to Part 1191**. Americans with disabilities act: accessibility guidelines for buildings and facilities (ADAAG). Washington, Dc, 1998.

WANG, L.; SHEN, W.; XIE, H.; NEELAMKAVIL, J.; PARDASANI, A. Collaborative conceptual design—state of the art and future trends. **Computer-Aided Design**. v. 34 n. 13, p. 981–996. 2002.

WANG, W.; RIVARD, H.; ZMEUREANU, R. An object-oriented framework for simulation-based green building design optimization with genetic algorithms. **Advanced Engineering Informatics**. v. 19, n. 1, p. 5–23. 2005.

WILL, P. Simulation and modeling in early concept design: An industrial perspective. **Research in Engineering Design**. v.3, n. 1. p. 1-14, 1991.

WOOLDRIDGE, M. **An Introduction to Multiagent Systems**. 2. ed. Nova York: Wiley, 2002.

ZHANG, L.; WANG, Y.; SHI, H.; ZHANG, L. Modeling and analyzing 3D complex building interiors for effective evacuation simulations. **Fire Safety Journal**, n. 53, p. 1-12, 2012.

Apêndice A

SOFTWARES UTILIZADOS

A lista abaixo apresenta os principais softwares utilizados para realização do presente trabalho:

- **Apache Jena 2.6.4** e **Apache Fuseki 0.2.5** (<http://www.jena.apache.org>) ferramentas para desenvolvimento de aplicações de *web-semântica* com suporte a manipulação de dados RDF e OWL via consultas SPARQL;
- **Ferramenta de Recorte de Ontologias** ferramenta de autoria própria para extração porções específicas de uma ontologia em formato OWL.
- **Stanford Protegè 4.1.0** (<http://protege.stanford.edu>) ferramenta de manipulação de ontologias em formato OWL;
- **Google Gson 2.2.2** (<http://code.google.com/p/google-gson>) biblioteca para manipulação de dados JSON em Java, necessária para a comunicação entre agentes;
- **Oracle Java 3D 1.5.2** (<http://www.java3d.org>) biblioteca importação e visualização de geometrias de modelos de edifício em Java;
- **Telecom Italia JADE - Java Agent Development Framework 4.2** (<http://jade.tilab.com>) infraestrutura de sistemas multiagentes em Java;
- **Trimble Sketchup 8.0.4811** (<http://www.sketchup.com>) e **Blender 3D 2.6.4** (<http://www.blender3d.org>) ferramenta para criação e manipulação de modelos tridimensionais de edifício.