

MARCELA TURIM KOSCHEVIC

**ARQUITETURA PARA SISTEMAS TUTORES
INTELIGENTES QUE UTILIZA A TEORIA QUALITATIVA
DE PROCESSOS PARA RECOMENDAÇÃO DE PRÓXIMOS
MELHORES EXERCÍCIOS**

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre. Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, Universidade Federal do Paraná.

Orientador: Prof. Dr. Andrey Ricardo Pimentel

CURITIBA

2015

MARCELA TURIM KOSCHEVIC

**ARQUITETURA PARA SISTEMAS TUTORES
INTELIGENTES QUE UTILIZA A TEORIA QUALITATIVA
DE PROCESSOS PARA RECOMENDAÇÃO DE PRÓXIMOS
MELHORES EXERCÍCIOS**

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre. Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, Universidade Federal do Paraná.

Orientador: Prof. Dr. Andrey Ricardo Pimentel

CURITIBA

2015

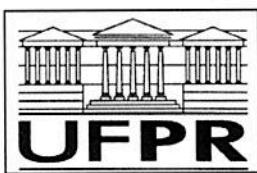
K86a Koschevic, Marcela Turim
Arquitetura para sistemas tutores inteligentes que utiliza a teoria qualitativa de processos para recomendação de próximos melhores exercícios/ Marcela Turim Koschevic. – Curitiba, 2015.
114 f. : il. color. ; 30 cm.

Dissertação - Universidade Federal do Paraná, Setor de Ciências Exatas,
Programa de Pós-graduação em Informática, 2015.

Orientador: Andrey Ricardo Pimentel .
Bibliografia: p. 109-114.

1. Sistemas tutoriais inteligentes. 2. Inteligência artificial - Aplicações
educacionais. 3. Aprendizagem - Métodos de ensino. I. Universidade Federal
do Paraná. II. Pimentel, Andrey Ricardo. III. Título.

CDD: 006.3330113



Ministério da Educação
Universidade Federal do Paraná
Programa de Pós-Graduação em Informática

PARECER

Nós, abaixo assinados, membros da Banca Examinadora da defesa de Dissertação de Mestrado em Informática, da aluna Marcela Turim Koschevic, avaliamos o trabalho intitulado, “Uma Arquitetura para Sistemas Tutores Inteligentes que Utiliza a Teoria Qualitativa de Processos para Recomendação de Próximos Melhores Exercícios”, cuja defesa foi realizada no dia 25 de agosto de 2015, às 09:30 horas, no Departamento de Informática do Setor de Ciências Exatas da Universidade Federal do Paraná. Após a avaliação, decidimos pela:

aprovação da candidata. reprovação da candidata.

Curitiba, 25 de agosto de 2015.

A blue ink signature of Prof. Dr. Andrey Ricardo Pimentel, which appears to read "Andrey Ricardo Pimentel".

Prof. Dr. Andrey Ricardo Pimentel
PPGInf - Orientador

A blue ink signature of Prof. Dr. João Alberto Fabro, which appears to read "João Alberto Fabro".

Prof. Dr. João Alberto Fabro
UTFPR – Membro Externo

A blue ink signature of Prof. Dr. Eduardo Todt, which appears to read "Eduardo Todt".

Prof. Dr. Eduardo Todt
PPGInf – Membro Interno



Às minhas queridas, mãe Juraci, e irmã Marivane, pelo incentivo, apoio e fé. Amo vocês.

AGRADECIMENTOS

À minha família pelo apoio incondicional, especialmente à minha irmã, Marivane, que dispôs muito de seu tempo para me ajudar neste período.

Ao meu orientador, professor Andrey Ricardo Pimentel, pela disponibilidade, incentivo e paciência ao longo desta jornada.

Aos professores do Departamento de Informática da UFPR.

Aos meus colegas de trabalho do Instituto Federal do Paraná, câmpus Foz do Iguaçu, pela compreensão e apoio.

Aos meus colegas do laboratório FAES/UFPR, os quais sempre estiveram dispostos a me auxiliar, em especial, àqueles que leram e fizeram sugestões para este trabalho.

Aos meus queridos alunos, pelas demonstrações de carinho e pelo estímulo para que eu seja uma pessoa e uma docente melhor.

À todas as pessoas que influenciaram meu trabalho, direta ou indiretamente.

Por fim, à Deus, que permitiu vivenciasse todas as dificuldades e sobretudo, superá-las.

*“Sem a curiosidade que me move, que me inquieta, que
me insere na busca, não aprendo nem ensino”.*

Paulo Freire

SUMÁRIO

RESUMO	xii
ABSTRACT	xii
1 INTRODUÇÃO	1
1.1 Contexto da pesquisa	6
1.2 Hipótese da pesquisa	7
1.3 Objetivos	7
1.4 Metodologia	8
1.5 Contribuição do estudo	8
1.6 Organização do documento	9
2 TRABALHOS RELACIONADOS	10
2.1 Sistemas tutores inteligentes	12
2.1.1 Arquitetura tradicional	12
2.1.2 Sistema de avaliação por múltipla escolha	14
2.1.3 Sistemas de recomendação	15
2.2 Teoria do raciocínio qualitativo	19
2.2.1 Abordagens da teoria qualitativa dos processos	21
2.2.2 Teoria qualitativa dos processos	24
2.2.3 Definições	28
2.2.4 Raciocínio qualitativo na educação	33
2.3 Considerações finais	34
3 ARQUITETURA DE UM STI BASEADO EM RACIOCÍNIO QUALITATIVO	35
3.1 Visão geral	35
3.2 Sistema tutor baseado em TQP	37

3.2.1	Simulação de estados	38
3.2.2	Modelos de domínio com a teoria qualitativa de processos	39
3.2.3	Modelo de domínio para conteúdos	39
3.2.3.1	Fragmentos de modelo	42
3.2.3.2	Análise dos cenários pelo PMA-TQP para conteúdos	43
3.2.4	Modelo de domínio para próximos melhores exercícios	50
3.2.4.1	Fragmentos de modelo para próximos melhores exercícios .	51
3.2.4.2	Análise dos cenários pelo PMA-TQP para exercícios	56
3.2.5	Considerações finais	69
4	ESTUDO DE CASO	71
4.1	Introdução	71
4.2	A disciplina de orientação a objetos	71
4.3	O processo de modelagem	73
4.4	Modelo de domínio com inserção de erros e acertos	77
4.5	Modelos conceituais de domínio	79
4.5.1	Modelo de domínio para classe	83
4.5.2	Modelo de domínio para objeto e classe	87
4.6	Considerações finais	94
5	CONSIDERAÇÕES FINAIS	96
5.1	Conclusão	96
5.2	Trabalhos futuros	97
APÊNDICE		99
A	FERRAMENTA GARP	99
A.1	Principais funcionalidades	99
A.2	Principais componentes	101
BIBLIOGRAFIA		114

LISTA DE FIGURAS

2.1 Arquitetura tradicional de um STI	13
2.2 Comportamento da simulação qualitativa e das equações diferenciais	26
3.1 Arquitetura geral do sistema tutor	35
3.2 Arquitetura do sistema tutor baseado em TQP	37
3.3 Modelo genérico conceitual para relacionamento entre conteúdos	40
3.4 Modelo Geral de STI com RQ	41
3.5 Modelo TQP para quantidades gerais	41
3.6 Fragmentos de modelo gerais para as relações P+(B, A), P+(D, C), I+(D, A) e I+(B, C)	42
3.7 Cenário positivista para uma quantidade	44
3.8 Cenário negativista para uma quantidade	45
3.9 Cenário misto para uma quantidade	46
3.10 Cenário positivista para duas quantidade	47
3.11 Cenário negativista para duas quantidades	48
3.12 Cenário misto para duas quantidades	49
3.13 Árvore de entidades para questões	51
3.14 Árvore de fragmentos de modelos para questões	52
3.15 Fragmentos de modelo para o sistema de recomendação de questões	53
3.16 Associação entre as questões do Grupo X	54
3.17 Associação entre as questões do Grupo X e do Grupo Y	54
3.18 Cenário positivista para transição de estados em questão única	57
3.19 Cenário negativista para transição de estados em questão única	58
3.20 Cenário intermediário para transição de estados em questão única	59
3.21 Cenário positivista para transição de estados com três questões	61
3.22 Grafo positivista para transição de estados com três questões	61
3.23 Gráfico positivista para transição de estados com três questões	62

3.24 Grafo negativista para transição de estados com três questões	63
3.25 Gráfico negativista para transição de estados com três questões	64
3.26 Cenário de recomendação para transição de estados com três questões . . .	65
3.27 Grafo de recomendação para transição de estados com três questões . . .	65
3.28 Gráfico de recomendação para transição de estados com três questões . . .	66
3.29 Cenário positivista para transição de estados com cinco questões	67
3.30 Grafo positivista para transição de estados com cinco questões	67
3.31 Gráfico positivista para transição de estados com cinco questões	68
3.32 Gráfico negativista para transição de estados com cinco questões	69
3.33 Cenário de indicação para transição de estados com cinco questões	70
3.34 Grafo de indicação para transição de estados com cinco questões	70
3.35 Gráfico de indicação para transição de estados com cinco questões	70
4.1 Fragmento de modelo contendo as entidades do domínio para orientação a objetos	74
4.2 Árvore de fragmentos de modelo para as entidades classe e objeto	75
4.3 Fragmentos de modelo para as entidades classe e objeto	76
4.4 Modelo de domínio para conteúdos e questões - Recomendação para acertos	80
4.5 Modelo de domínio para conteúdos e questões - Recomendação para erros .	81
4.6 Exemplo de questão para o domínio Objeto	82
4.7 Exemplo de questão para o domínio Classe	82
4.8 Fragmento de modelo para classe com acerto e erro	83
4.9 Simulação positivista para classe com acerto e erro	84
4.10 Simulação negativista para classe com acerto e erro	85
4.11 Simulação mediana para classe com acerto e erro	86
4.12 Fragmentos de modelo para a orientação a objetos que envolvem as enti- dades classe e objeto	87
4.13 Cenário positivista que envolvem as entidades classe e objeto	88
4.14 Grafo positivista que envolvem as entidades classe e objeto	89
4.15 Gráfico positivista que envolvem as entidades classe e objeto	89

4.16 Cenário negativista que envolvem as entidades classe e objeto	90
4.17 Grafo negativista que envolvem as entidades classe e objeto	90
4.18 Gráfico negativista que envolvem as entidades classe e objeto	91
4.19 Cenário médio que envolvem as entidades classe e objeto	92
4.20 Grafo médio que envolvem as entidades classe e objeto	92
4.21 Gráfico médio que envolvem as entidades classe e objeto	93
4.22 Cenário com acertos médios e erros altos que envolvem as entidades classe e objeto	93
4.23 Grafo com acertos médios e erros altos que envolvem as entidades classe e objeto	94
4.24 Gráfico com acertos médios e erros altos que envolvem as entidades classe e objeto	95
A.1 Motor de raciocínio da ferramenta GARP	100
A.2 Entidades genéricas no GARP	101
A.3 Entidades criadas a partir do paradigma de orientação a objetos	102
A.4 Exemplo de configuração criada no GARP	103
A.5 Exemplo de espaços quantitativos criados no GARP	104
A.6 Exemplo de relações causais criadas no GARP	106
A.7 Exemplo de cenário criado no GARP	107
A.8 Exemplo de grafo de estados criado no GARP	108
A.9 Exemplo de gráfico com descrições de estados criado no GARP	108

RESUMO

A utilização de sistemas tutores inteligentes para a recomendação de conteúdos e exercícios é um assunto explorado por alguns pesquisadores. Muitos deles usam técnicas da inteligência artificial para fazer essas recomendações. A proposta desse trabalho é utilizar uma abordagem diferenciada das tradicionais para fazer esse processo. A abordagem usada neste trabalho é a teoria qualitativa dos processos, uma ontologia definida dentro a inteligência artificial que usa métodos de modelagem semelhantes ao processo de cognição humana. Nesse ponto a modelagem de um sistema torna-se interessante pois faz o uso de relações causais para definir o comportamento do sistema.

Nesta pesquisa encontra-se um estudo sobre as principais abordagens para sistemas tutores inteligentes e também um estudo sobre a teoria do raciocínio qualitativo. Foi desenvolvida uma arquitetura que permite explorar o uso da teoria do raciocínio qualitativo como forma de modelar um domínio de conhecimento. Relações causais foram aplicadas nesse domínio de conhecimento para que seja possível estimar qual o melhor conteúdo ou a melhor questão que pode ser recomendada para estudantes que utilizam o sistema.

Como estudo de caso, foram simulados cenários onde a disciplina de orientação a objetos foi usada como exemplo. Estes cenários exploraram o comportamento do sistema de recomendação para determinadas ações do estudante, sendo elas positivistas, negativistas ou intermediárias. A conclusão sobre os resultados foi analisada sob a ótica da teoria qualitativa dos processos e descrita na arquitetura e no estudo de caso. Os principais resultados obtidos indicam que é possível fazer recomendações de conteúdos e exercícios por meio do processo de modelagem apresentado nesta arquitetura.

Palavras-chave: Sistemas tutores inteligentes, teoria do raciocínio qualitativo, teoria qualitativa dos processos, próximo melhor exercício.

ABSTRACT

The use of intelligent tutoring systems for recommending content and exercises is a subject explored by some researchers. Many of them use techniques of artificial intelligence to make these recommendations. The purpose of this work is to use a different approach from traditional to do this process. The approach used in this study is the qualitative processes theory, a defined ontology in artificial intelligence, but using modeling methods similar to human cognition process.

At this point modeling a system becomes interesting because it makes use of causal relationship to define system behavior. This research is a study of the main approaches to intelligent tutoring systems and also a study on the qualitative reasoning theory. Was developed an architecture that allows you to explore the use of the qualitative reasoning theory as a way to model a domain of knowledge. Causal relationships have been applied in this area of knowledge so that it is possible estimate what the best content and the best issue that can be recommended for students who use the system.

As a case study, we simulated scenarios in which the discipline of object orientation was used as an example. These scenarios explored the recommendation system behavior for certain student actions, they being positivist, negativistic or intermediate. The conclusion on the results was analyzed from the perspective of qualitative processes theory and described in the architecture and in the case study. The main results indicate that it is possible to make content recommendations and exercises through the modeling process presented in this architecture.

Keywords: intelligent tutoring systems, qualitative reasoning theory, qualitative processes theory, better next exercise.

CAPÍTULO 1

INTRODUÇÃO

A inserção de computadores no processo de ensino-aprendizagem de conteúdos curriculares de diversos níveis e modalidades da educação é chamada de Informática na Educação. Uma abordagem adequada para essa denominação enfatiza que o professor da disciplina curricular possui conhecimento sobre os potenciais educacionais do computador. Enfatiza também que ele é capaz de alternar adequadamente atividades tradicionais de ensino-aprendizagem com as atividades realizadas por meio do computador [58].

As atividades da Informática na Educação podem ser realizadas de duas maneiras: para transmitir a informação para o estudante e reforçar o processo instrucionista, ou para criar condições nas quais o estudante consiga construir conhecimento. No ensino tradicional, o conteúdo a ser ministrado é determinado por um currículo e não pelo aprendiz. Neste modelo, a ênfase é centrada no conteúdo que deve ser memorizado e não nas habilidades que permitem o uso efetivo desse conteúdo [58].

No entanto, discute-se que o conhecimento deve ser construído e contextualizado, e essa abordagem é conhecida como construtivismo. Esta abordagem diz que o conhecimento pode ser construído com base na realização concreta de ações que produzem produtos palpáveis (artigos, objetos) e que seja de interesse pessoal de quem produz. Deve ser contextualizado tendo em vista a vinculação do produto à realidade da pessoa ou do local em que o produto vai ser construído e utilizado. Dessa forma, pode-se afirmar que o currículo pode ser construído pelo professor, juntamente com os estudantes e servir como um norteador para as tarefas a serem realizadas em sala de aula [58].

Seguindo esse raciocínio, o professor deixa de ser o entregador da informação e passa a ser facilitador, supervisor e consultor do estudante no processo de resolução de seus problemas. O professor proporciona ao estudante a chance de transformar a enorme carga de informações que ele adquire em conhecimento aplicável. Assim sendo, o professor

incentiva o processo de melhoria contínua do estudante. Mas para que tudo isso seja possível, o professor deve conhecer adequadamente seus estudantes para buscar estratégias de ensino adequadas a cada um deles.

Neste modelo de ensino-aprendizagem, o estudante deve estar constantemente interessado em aprimorar suas ideias e habilidades. Ele deve ser ativo, capaz de assumir responsabilidades, tomar decisões e buscar soluções [58].

A informática pode ser usada para implementação de atividades pedagógicas que proporcionem a formação de estudantes com habilidades fundamentais para serem bons profissionais. No entanto, a utilização da Informática na Educação não é indicação de que os estudantes estão compreendendo como resolver suas suas atividades. A qualidade da interação aprendiz-objeto é particularmente pertinente no caso do uso da Informática e de diferentes softwares educacionais. Do mesmo modo que não é o objeto que leva à compreensão, não é o computador que permite ao aluno entender ou não um determinado conceito. A compreensão é fruto de como o computador é utilizado e de como o aluno está sendo desafiado na atividade de uso desse recurso [58].

A criação de sistemas computacionais com fins educacionais tem acompanhado a própria história e evolução dos computadores. O uso do computador na educação surgiu no final da década de 50 e representava as possibilidades tecnológicas da época. É importante observar que os paradigmas de aprendizado embutidos nesses sistemas refletem e situam o contexto educacional vigente¹ [58]. A chamada instrução programada² foi a base para os primeiros sistemas e representava uma automatização do processo de ensino/aprendizado condizente com as possibilidades tecnológicas da época em que foram desenvolvidos [56].

Os sistemas computacionais continuaram a evoluir, incorporando avanços tecnológicos, principalmente da área de Inteligência Artificial (IA), que possibilitaram certas sofis-

¹Historicamente, os primeiros sistemas computacionais para uso no ensino surgiram ainda na década de 60 e faziam parte da categoria de sistemas Computer Assisted Instruction (CAI), inspirados no método da instrução programada [58].

²A instrução programada é um método de ensino surgido na década de 50 e consiste na organização do material a ser ensinado em segmentos logicamente encadeados, chamados módulos. Os módulos são, então, apresentados ao aprendiz, de forma gradual e sequencial. Dessa maneira, o estudante pode seguir seu próprio ritmo, retornando a módulos anteriores, quando sente necessidade, ou avançando para conteúdo de módulos futuros [58].

ticações nos sistemas computacionais derivados, atualmente chamados Sistemas Tutores Inteligentes (STI's). Os sistemas atuais imitam a ação de um tutor, gerando problemas de acordo com o nível de conhecimento obtido pelo estudante, comparando as respostas dos estudantes com as de especialistas no domínio, diagnosticando fraquezas, associando explicações específicas para certos tipos de erros, decidindo quando e como intervir [58].

Dos sistemas baseados no paradigma instrucionista, em que pouca ou nenhuma iniciativa e controle são reservados ao estudante, um novo paradigma educacional começou a nortear o desenvolvimento de sistemas computacionais para uso em educação, fundamentado nas ideias construcionistas de Papert [56].

Atualmente, uma classe de sistemas computacionais, baseados na ideia de ferramentas para uma interação rica em ambientes interessantes, é proposta para promover o aprendizado construcionista. O objetivo é encorajar o estudante a tomar a iniciativa, e o aprendizado é entendido não como mera aquisição de conhecimento, mas como uma evolução em direção à *expertise*, na qual componentes como planejamento, descrição, execução e reflexão são parte do ciclo interativo do aprender [58].

A evolução de um estudante rumo à *expertise* é objeto de estudo da psicologia e deu origem a teorias de como o aprendizado surge e qual é o caminho motivador para que ele aconteça. A atenção do estudante é a energia que ele controla e é uma ferramenta importante para melhorar a qualidade da experiência de aprendizagem [15]. A implementação de técnicas motivacionais exige modelar o sistema, incluindo a representação do domínio e do modelo do estudante, em alguns aspectos. Em particular, o sistema tutor tem de detectar o estado motivacional do aluno e reagir com o objetivo de motivar os alunos que estejam menos confiantes ou descontentes. O sistema também precisa manter a disposição dos alunos já motivados [18].

A teoria do fluxo define um estado mental no qual as pessoas parecem fluir, quando mostram um esforço produtivo e motivado, associado a várias emoções relacionadas a comportamentos positivos e funcionais. Sugere-se que ao experimentar o estado de fluxo em uma atividade por várias vezes, a pessoa passa a executá-la sem esperar benefícios externos. Assim, a atividade se torna intrinsecamente motivada [30].

Como um estado psicológico ótimo, o fluxo representa aqueles momentos nos quais tudo está a favor do desempenho, a consciência é completamente tomada pela experiência, e é frequentemente associado a altos níveis de desempenho e a uma experiência muito positiva [16].

O fluxo ocorre quando há completo envolvimento com a tarefa a ser realizada. No fluxo, a pessoa se sente forte, não se preocupa consigo mesmo ou com o fracasso. O fluir pode ser definido como uma experiência que se destaca por ser, de alguma maneira, melhor do que a média, onde o indivíduo é totalmente absorvido pelo que faz, e onde a experiência é muito gratificante em si mesma [30].

Se a teoria do fluxo for aplicada em questões de ensino-aprendizagem, podemos traçar uma relação entre o estado psíquico ideal do estudante e como o sistema tutor organizará seus conteúdos e questões para que o estudante permaneça na zona de fluxo. Conforme apresentado na teoria de Csikszentmihalyi, o estado ideal para o estudante usar sua concentração e atenção de forma plena é quando ele possui habilidades altas e desafios altos também. Ou seja, o nível dos desafios deve ser proporcional ao nível da habilidade do estudante.

Quando um estudante inicia seus estudos, pode ser que seu nível de habilidade seja muito baixo, portanto os desafios não devem ser muito altos para não provocar ansiedade no estudante, nem baixos demais, para não provocar apatia. Quando o estudante melhora suas habilidades, o ideal é que os desafios cresçam ligeiramente também.

Em um sistema tutor inteligente, algumas abordagens podem ser tomadas para que o próprio sistema estimule o estudante a permanecer na zona de fluxo. A abordagem objeto de estudo deste trabalho é o uso da recomendação de próximos melhores exercícios e conteúdos baseados no desempenho do estudante. Esta abordagem foi escolhida pois mensurar o próximo melhor exercício ou conteúdo é uma prática que tem a ver com a inserção de dificuldades ou facilidades para um estudante. Se a próxima questão ou conteúdo for muito difícil, o estudante poderá perder o estímulo pois não consegue suprir o desafio com suas habilidades, o que caracteriza o quadro de ansiedade previsto na teoria de fluxo. Por outro lado, se a questão ou conteúdo proposto for muito fácil para o

estudante, ele pode ficar entediado. Ambas as situações podem provocar a desistência do estudante.

Para elaborar um modelo sistêmico de recomendação de próximos melhores exercícios e conteúdos, foi realizado um estudo sobre uma abordagem levantada por Anderson, através da qual admite-se o uso de raciocínios causais para modelar sistemas tutores inteligentes, como uma alternativa ao uso da IA [5].

Os raciocínios causais servem bem para a construção de modelos de recomendação porque são teorias baseadas na relação de causa-efeito. Essa mesma relação é estabelecida na teoria do fluxo, na qual a concentração e a atenção exercem efeitos diretos sobre a aprendizagem.

Neste trabalho foi realizada a construção de uma arquitetura para sistemas tutores que recomendam atividades de acordo com a análise do estado de um estudante. A arquitetura possui um modelo especialista baseado na teoria do raciocínio qualitativo, que é uma forma de modelar um sistema sem precisar conhecer profundamente a IA, permitindo que pessoas fora área de Informática possam construir o modelo especialista. O raciocínio qualitativo é uma teoria de raciocínio causal³, que condiciona as relações entre os elementos que possuem valores e indicam qual deverá ser o comportamento das recomendações, tendo como atributo de mudança, os resultados obtidos pelo estudante. Esta teoria, modela e atribui pesos às relações entre elementos do mundo físico, e pode ser usada para representar as unidades da aprendizagem.

Na arquitetura, é possível estimar o desempenho do estudante, e com base nisso, calcular o rendimento qualitativo que ele está obtendo nas atividades, por meio de monitoramento dos passos desse estudante. A arquitetura realiza simulações para determinar quais são as melhores atividades a serem exibidas. Quando o estudante obtém um rendimento baixo em alguma atividade, a arquitetura permite a indicação uma intervenção, reforçando o conteúdo e apresentando atividades com nível de dificuldade menor. Da mesma forma, quando o estudante está obtendo um alto rendimento, atividades de alto nível de dificuldade são apresentadas a ele como uma forma de desafio. O professor é

³Trata relações de causa e efeito.

responsável por alimentar o sistema, no qual deve definir os níveis de dificuldade das atividades e a ligação entre os conteúdos. O sistema, por meio da medida de desempenho do estudante, se encarrega de apresentar o currículo mais adequado para ele.

Em geral, a aprendizagem é algo particular a cada indivíduo. Compreende-se que cada um possui sua própria velocidade de aprendizagem. As abordagens tradicionais instrucionistas podem ser muito rígidas, pois não contemplam abordagens dinâmicas, que geralmente despendem mais tempo, conhecimento e habilidades do docente. A elaboração de um sistema tutor que envolva certa inteligência que garanta uma dinamicidade é visto como um atrativo e uma solução ao excesso de rigidez tradicional presente nos modelos educacionais antigos. Além do mais, o ensino individualizado com foco no desenvolvimento e na melhora das habilidades pode ser capaz de manter a motivação do estudante.

O presente trabalho apresenta um estudo sobre sistemas tutores inteligentes, uma arquitetura baseada na teoria qualitativa dos processos e um estudo de caso que faz o uso da arquitetura por meio de simulações que qualificam o modelo proposto.

1.1 Contexto da pesquisa

Esta pesquisa está inserida no contexto de Sistemas Tutores Inteligentes e busca abordar o assunto de uma forma teórica e prática.

Diversos assuntos podem ser ensinados por meio de sistemas tutores inteligentes e para cada tipo de assunto podem ser usadas abordagens diferentes. Este trabalho apresenta uma arquitetura de sistema tutor que utiliza a abordagem das teorias de relações causais, mais especificamente, a teoria de raciocínio qualitativo (RQ), proposta por Kenneth A. Forbus [25], como meio de modelar as interações que existem entre as entidades dos sistemas físicos. Para o estudo de caso, foi eleita uma disciplina de informática, a orientação a objetos, como meio de demonstrar uma situação de aprendizagem.

Por ser um paradigma já consolidado, a orientação a objetos pode ser modelada em um nível de conceitos, tornando possível verificar a dependência que existe entre esses conceitos. Com base nisso, a disciplina foi modelada seguindo os preceitos da teoria de raciocínio qualitativo, com valores indicativos do quanto a aprendizagem de um conceito

influencia na aprendizagem de um outro conceito.

1.2 Hipótese da pesquisa

Esta pesquisa considera que a teoria qualitativa dos processos, uma das ontologias definidas pela TQP, pode ser utilizada como ferramenta para a construção de modelos de demínio em um STI. É também considerada parte da hipótese que a eleição da próxima melhor atividade a ser apresentada ao estudante, pode ser indicada pela simulação de modelos definidos com o raciocínio qualitativo.

1.3 Objetivos

Com o intuito de verificar e validar a hipótese apresentada foram definidos os objetivos para este trabalho de mestrado, sendo um deles o objetivo geral e três específicos que seguem descritos no texto em sequência.

Objetivo geral

O objetivo geral desta pesquisa é definir uma arquitetura para eleição de próximos melhores exercícios, aplicada em um contexto de ensino-aprendizagem de um paradigma de programação (orientação a objetos), com apoio da teoria de raciocínio qualitativo.

Objetivos específicos

Além do objetivo geral, foram definidos os seguintes objetivos específicos:

1. Apresentar um estudo sobre as principais abordagens para sistemas tutores inteligentes e sobre a teoria do raciocínio qualitativo.
2. Construir uma arquitetura baseada na teoria do raciocínio qualitativo para a recomendação de próximos melhores exercícios e conteúdos.
3. Aplicar a arquitetura proposta em um estudo de caso com simulações sob perspectivas diferentes.

4. Analisar e comparar os resultados da simulações realizadas considerando distintas perspectivas de aprendizado (positivista, negativista e intermediária).

1.4 Metodologia

Este trabalho está baseado no estudo e na definição de uma arquitetura de STI e de um modelo de eleição para próximas melhores questões no contexto da orientação a objetos. A fundamentação deste trabalho está descrita nas etapas a seguir:

A primeira etapa consistiu em realizar uma revisão literária sobre os assuntos abordados. Nesta etapa foram realizadas comparações entre casos citados na literatura.

A segunda etapa consistiu na definição da proposta para essa dissertação. Nessa fase foi levantado o que deveria ser realizado no trabalho, construído e discutido.

A terceira etapa consistiu na definição de uma arquitetura para um sistema tutor inteligente.

Na quarta etapa foi construído um mapa conceitual da disciplina de orientação a objetos. Este mapa conceitual foi transformado em um modelo baseado em raciocínio qualitativo, mais especificamente, um modelo que segue a teoria qualitativa dos processos. A modelagem seguindo a teoria qualitativa dos processos foi amplamente revisada e testada.

A quinta etapa consistiu na discussão sobre os resultados obtidos e na criação de propostas futuras.

A metodologia de pesquisa usada neste trabalho foi estruturada a partir do estudo sobre a escolha de melhores próximos exercícios, como isso se aplica á um sistema tutor inteligente, a viabilidade desse procedimento e a uma conclusão efetiva sobre a proposta.

1.5 Contribuição do estudo

Os resultados desta pesquisa contribuem para o desenvolvimento de objetos de aprendizagem, sendo este um setor que recebe pesquisa e fomento de setores federais.

A proposta deste estudo colabora com a criação de uma a arquitetura de eleição de

melhores próximos exercícios e conteúdos para escolher as atividades mais adequadas a serem apresentadas aos estudantes, de acordo com nível de habilidade que eles possuem.

O estado de fluxo, que representa um estado psicológico ótimo, pode ocorrer em qualquer nível de habilidade, desde que haja um equilíbrio ideal entre o desafio percebido e o nível de habilidade. Quando a informação que chega a consciência é coerente com os objetivos, a energia psíquica flui sem esforço. O fluxo representa momentos nos quais há favoritismo ao desempenho e a experiências positivas.

Para manter uma pessoa em estado de fluxo, é necessário que os desafios propostos sejam condizentes com suas habilidades, ou seja, não pode-se exigir pouco e nem muito, pois ambas as situações podem ocasionar a desistência do estudante, por ansiedade, por tédio ou por apatia.

Em resumo, neste trabalho, a abordagem utilizada para que manter o estudante focado é a recomendação automática de conteúdos e exercícios, baseada no desempenho do estudante. Dessa forma, o estudante terá a sua disposição, desafios mais adequados às suas habilidades, mantendo assim sua motivação.

1.6 Organização do documento

O presente trabalho está organizado da seguinte forma:

O Capítulo 2 apresenta o embasamento conceitual, e nele são abordados os aspectos de sistemas tutores inteligente, as principais arquiteturas existentes, medidas cognitivas, e alguns tópicos referentes à teoria de raciocínio qualitativo, tanto de forma geral quanto de forma aplicada na educação.

O Capítulo 3 propõe uma arquitetura para sistemas tutores inteligentes que integra a teoria do raciocínio qualitativo, e o Capítulo 4 apresenta um estudo de caso em que essa arquitetura é aplicada no ensino na disciplina de orientação a objetos.

O Capítulo 5 finaliza com as principais conclusões desta pesquisa e elenca possíveis trabalhos futuros.

CAPÍTULO 2

TRABALHOS RELACIONADOS

O ensino deve ser planejado passo a passo de modo a obter os resultados desejados. No contexto de ensino-aprendizagem, e dentro de abordagens da aprendizagem por reforço positivo, várias teorias foram desenvolvidas para tentar entender e melhorar o processo de construção do conhecimento [23].

Estudos realizados deram embasamento à construção das chamadas máquinas de aprendizagem e dos sistemas tutores inteligentes, onde o próprio aluno evolui gradualmente para atingir níveis desejáveis de conhecimento em um determinado assunto. Um Sistema Tutor Inteligente (STI) é baseado na hipótese de que o processo de pensamento de um estudante pode ser modelado, rastreado e corrigido [55], tornando possível não apenas ensinar, mas descobrir os caminhos utilizados pelo aprendiz para chegar ao conhecimento desejado [56].

As máquinas de aprendizagem propostas, segundo seu autor, podem ser colocadas em sala de aula para auxiliar o professor no ensino dos conteúdos, servindo como reforço ao aprendizado do estudante. Inicialmente, elas foram programadas com perguntas de múltipla escolha sobre um determinado assunto. O estudante então tinha que posicionar o botão na casa correspondente a resposta correta. Em caso de erro, o aluno não passaria para a próxima pergunta. Em algumas situações era acoplada uma luz que acendia toda vez que o estudante apresentava a resposta correta. Como cada aluno tinha um aparelho, o ritmo da sequência de perguntas era controlado por cada estudante [56].

Atualmente, as pesquisas sobre sistemas tutores inteligentes estão voltadas para entender como as pessoas organizam o conhecimento que adquirem a partir de suas experiências e produzem, com base nisto, um comportamento inteligente [4].

Muitos sistemas tutores inteligentes podem ser vistos apenas como uma ferramenta, por meio da qual, professores fornecem orientação e *feedback*, enquanto os alunos tra-

lham em exercícios práticos. Ao dar apoio tecnológico aos docentes, os STIs permitem que os professores concentrem seus esforços em outras questões, como o planejamento de currículos e a entrega de novos conceitos aos estudantes [34].

Anderson afirma em seus trabalhos que uma categoria de sistemas tutores está focada em entender a capacidade humana de simular e raciocinar sobre processos dinâmicos. Segundo ele, este é um componente importante para a resolução de problemas e envolve o conhecimento sobre o raciocínio que existe em uma estrutura causal [5].

Os modelos baseados na teoria do raciocínio qualitativo (RQ) são relativamente imaturos em comparação com os esquemas de formalismos e bases de regras da inteligência artificial (IA). No entanto, diversos trabalhos recentes estão desenvolvendo modelos baseados em raciocínio qualitativo e apresentando resultados satisfatórios [26, 37, 39].

Existem ferramentas utilizadas nos processos de ensino-aprendizagem de programação, que vão desde softwares simples usados em universidades nas quais foram desenvolvidos até projetos comerciais que são usados em uma série de instituições de todo o mundo. O uso de materiais didáticos com suporte de tecnologias é uma prática cada vez mais crescente nas áreas de ensino e não raras são as vezes em que recursos tecnológicos dão suporte para a dinâmica da proposição de um conceito, revisão ou aprofundamento [40].

Alguns sistemas tutores inteligentes foram desenvolvidos para a área de ensino de orientação a objetos. Um exemplo é o sistema proposto por Galhardo e Zaina, que faz o uso de simulações gráficas em um ambiente web para ensinar os conceitos da orientação a objetos para os estudantes. Este software apresenta questões aos estudantes como forma de verificar a aprendizagem. As questões apresentadas são cadastradas por um professor, assim como a sequência em que devem ser exibidas. Neste software, não há nenhum mecanismo inteligente que faça a recomendação de questões [28].

Na literatura foram encontrados também alguns softwares, que não são STI's, usados para o ensino de orientação a objetos. Um exemplo é o software proposto por Forest, que apresenta exercícios para os estudantes mas não usa nenhum tipo de inteligência para essas recomendações [27].

Este capítulo introduz os principais conceitos envolvidos no desenvolvimento de uma

arquitetura para sistemas tutores inteligentes integrada à teoria do raciocínio qualitativo, abordando os seguintes tópicos:

- Os sistemas tutores inteligentes e suas principais características;
- A teoria do raciocínio qualitativo e as suas abordagens;

2.1 Sistemas tutores inteligentes

Um sistema tutor inteligente é um programa de computador projetado para incorporar técnicas de inteligência artificial (IA), a fim de fornecer tutores que sabem o “que”, a “quem” e “como” ensinar. A IA proporciona tentativas para produzir um comportamento de computador que, quando executado por um ser humano, poderia ser descrito como inteligente. Um STI pode igualmente ser considerado como uma tentativa de produzir um comportamento de computador que, quando executado por um ser humano, poderia ser descrito como ensino incentivado [47].

Em sua essência os STI's são programas que modificam suas bases de conhecimento, percebem as intervenções dos alunos e são dotados da capacidade de aprender e adaptar suas estratégias de ensino a partir das interações do aprendiz [59] .

A concepção e desenvolvimento de STI's acabou por juntar áreas como a ciência da computação, a psicologia cognitiva e a pesquisa educacional. Esta área de intersecção é denominada ciência cognitiva.

2.1.1 Arquitetura tradicional

Anteriormente aos STI's, os sistemas tradicionais IAC (Instrução Assistida por Computador) apresentavam os componentes combinados em uma única estrutura, ou seja, não existia uma divisão ou separação por níveis e por funções desempenhadas, causando diversos problemas no sistema quando era necessário fazer alterações em algum nível. A partir da necessidade de separar o sistema em componentes originou-se a divisão [33, 47]:

1. O conhecimento do domínio;

2. O conhecimento da pessoa que está sendo ensinada;
3. O conhecimento das estratégias de ensino;
4. O conhecimento de como aplicar o conhecimento das estratégias de ensino e prover a individualização.

A partir dessas subdivisões foram denominados quatro módulos principais, que, interligados, compõem a arquitetura clássica de um STI, e que são ilustrados na Figura 2.1 [47]:

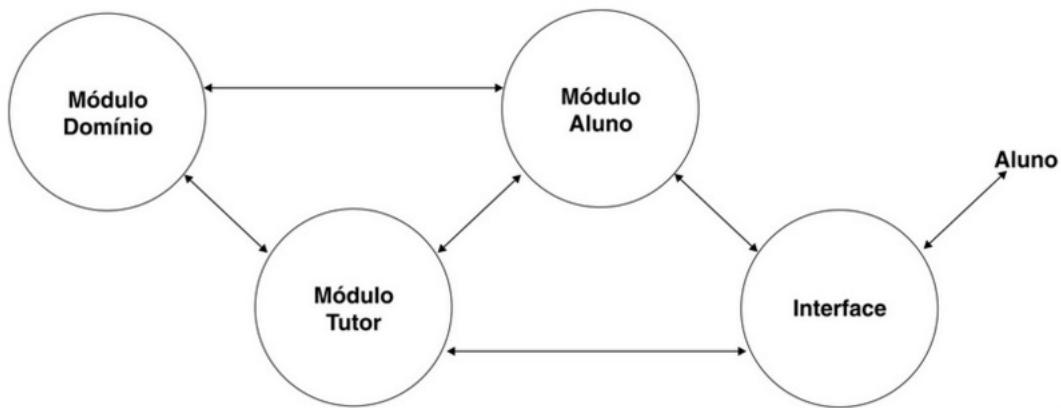


Figura 2.1: Arquitetura tradicional de um STI

- O Módulo Aluno representa o conhecimento e as habilidades cognitivas do aluno em um dado momento. Também contém uma representação do estado do conhecimento do aluno no momento em que usa um STI.
- O Módulo Tutor é onde são definidas e aplicadas as estratégias pedagógicas para o ensino, contém os objetivos a serem alcançados e as táticas de ensino. Este módulo é responsável por selecionar os problemas, monitorar o desempenho do aprendiz, fornecer assistência à aprendizagem, além de integrar o conhecimento sobre métodos de ensino, técnicas de ensino e domínio a ser explorado.
- O Módulo Domínio tem como objetivo principal armazenar todo o conhecimento dependente e independente do âmbito da STI, ou seja, deve conter todo o conhecimento do conteúdo a ser abordado.

- O Módulo Interface é responsável pela interação do aprendiz com o sistema. A interface é a principal porta de entrada para a comunicação com o aprendiz.

De uma forma geral, os STI's se caracterizam por representar separadamente o conteúdo a ser explorado (módulo domínio) e as estratégias a serem exploradas (módulo tutor), caracterizando o aprendiz (módulo do estudante) para oferecer um ensino personalizado por meio de uma interface (módulo de interface) planejada e organizada [40]. Este é o modelo clássico criado para os STI's. Algumas derivações desse modelo foram propostas por McTaggart [44], Kaplan [35], Clancey [14] e Wenger [1].

2.1.2 Sistema de avaliação por múltipla escolha

Questões de múltipla escolha são a forma mais simples de avaliação, e o procedimento de avaliação é frequentemente incorporado nas próprias questões. Elas podem ser usadas para avaliar diversos níveis cognitivos [13]. Enquanto questões de múltipla escolha mal construídas não avaliam nada além da lógica para serem respondidas, questões bem elaboradas são uma boa ferramenta para avaliar o conhecimento e compreensão [41] ou mesmo níveis cognitivos superiores [36].

A forma mais comum é uma questão de múltipla escolha com quatro ou mais alternativas, das quais pelo menos uma é correta. No entanto, o número de escolhas corretas, parcialmente corretas e incorretas podem variar, dependendo da configuração dos professores envolvidos na criação das questões [17].

Normalmente, um estudante é recompensado com pontos para a escolha correta e parcialmente correta. Geralmente uma resposta incorreta implica em zero ou um número negativo de pontos. O procedimento de avaliação de questões de múltipla escolha é muito simples. Esse mecanismo compara a resposta do estudante com o valor correto e de acordo com uma fórmula de classificação, atribui os pontos obtidos à pontuação total do estudante [17].

A simplicidade de questões de múltipla escolha tornou-as muito populares em sistemas

gestores de aprendizagem, como o Moodle¹ e o WebCT². No entanto, no ensino de programação, questões de múltipla escolha são mais úteis para aprendizagem teórica básica, e não são tão boas para ganhar habilidades práticas na resolução de tarefas de programação [17].

O uso de questões de múltipla escolha permite gerar diferentes permutações de conjuntos de perguntas, e os testes podem ser individualizados evitando fraudes e memorização por parte dos estudantes. Além disso, questionários de múltipla escolha podem ter sua complexidade aumentada em virtude do desempenho do estudante.

Apesar do grande uso de questões de múltiplas escolhas em sistemas de avaliação, existem poucos exemplos de sistemas que aliam algum tipo de inteligência para sugerir quais questões são mais adequadas ao contexto do estudante [22].

2.1.3 Sistemas de recomendação

Em uma sessão de ensino (aula), um professor pode usar sua experiência para recomendar exercícios para os estudantes ou pode buscar amparo em medidas cognitivas. Tais medidas possuem a função de quantificar cognitivamente o enunciado de um exercício ou de um exemplo. Elas são responsáveis por estimar o esforço exigido do aprendiz em termos de conhecimentos adquiridos e habilidades desenvolvidas durante a progressão do estudo [51].

Na informática, alguns sistemas tutores inteligentes, mensuram a chamada carga cognitiva de enunciados, que é a capacidade de exercitar um estudante durante uma atividade de programação de computadores. Para estas atividades, geralmente são adotados como parâmetros determinantes da complexidade de um exercício, o uso de linhas de código, a complexidade estrutural e os detalhes de implementação. Com base no valor obtido da carga cognitiva, é possível recomendar o próximo enunciado ou exemplo a ser apresentado.

Um exemplo de STI que usa a abordagem de carga cognitiva é a ferramenta de autoria Sequence. Ela foi concebida inicialmente para a área médica, e posteriormente foi adaptada ao ensino de programação. A ferramenta avalia um enunciado quanto à sua

¹<http://moodle.org/>

²<http://www.webct.com/>

complexidade de carga cognitiva, mas também considera outras variáveis, como, a precisão sintática e semântica, análise do problema, reutilização de soluções e simulação mental [50].

Outras ferramentas de recomendação de próximos melhores exercícios são descritas a seguir.

Sistema Tutor Inteligente JV²M

Sistema destinado ao ensino do processo de compilação em linguagens orientadas a objetos por meio de um ambiente virtual. Trata-se de uma abordagem que instiga o estudante a aprender fazendo, na qual, ele se concentra em tarefas que refletem objetivos pedagógicos, com explicações em momentos específicos [43].

O JV²M³ apresenta problemas de complexidade crescente ao aprendiz, dando-lhe explicações contextualizadas. O próximo exercício é apresentado quando o sistema detecta que o aprendiz tem conhecimento suficiente para avançar ao próximo nível [54].

O sistema faz a recomendação dos próximos melhores exercícios por meio de quatro elementos:

- **Base de casos (cenários):** Responsável por armazenar os exercícios que podem ser apresentados aos estudantes;
- **Modelo do estudante:** Mantém as informações sobre o domínio dos estudantes em relação ao conhecimento ensinado;
- **Módulo pedagógico:** Indica qual o próximo exercício a ser apresentado ao estudante, de acordo com o estado do conhecimento advindo dos módulos anteriores;
- **Ambiente interativo de aprendizagem:** Representa a interface do sistema, um espaço no qual os estudantes interagem para resolver os exercícios.

Neste sistema, o exercício é selecionado pelo módulo pedagógico, de acordo com o modelo do estudante. O ambiente apresenta um mundo virtual para o aprendiz interagir

³O sistema se chama JV²M para dar a ideia de duas vezes virtual, já que ele é uma representação virtual da máquina virtual do Java (JVM) [31]

com objetos para resolver o exercício. Por fim, o modelo do aprendiz é atualizado de acordo com a performance do estudante ao resolver o exercício [54].

Sistema Tutor Inteligente SQL-Tutor

O SQL-Tutor é um STI criado para tutoriar o ensino da linguagem SQL (linguagem de pesquisa para banco de dados relacionais) para estudantes universitários. Ele é composto pelo módulo de interface, módulo pedagógico e pelo modelo do aprendiz. O sistema possui as definições de várias bases de dados, um conjunto de problemas e suas soluções ideais. Um nível estático de complexidade é atribuído a cada problema, por um especialista no domínio, baseado nos conceitos necessários para a resolução. O SQL-Tutor não dispõe de um resolvedor de problemas para avaliar a resposta dada pelo aprendiz. No entanto, ele compara a resposta com a solução ideal, usando o conhecimento do domínio representado por restrições [46].

Neste sistema, o estudante pode selecionar os problemas que deseja resolver de quatro maneiras: escolher uma série de problemas de uma base de dados, decidir a partir de uma lista de alternativas proposta pelo sistema, solicitar que o STI escolha um problema com base em seu perfil ou informar uma cláusula SQL que deseja exercitar [45].

O propósito de apresentar essas alternativas é encorajar o aprendiz a refletir sobre o próprio conhecimento, identificando suas dificuldades e habilidades. Dessa forma, ele assume a responsabilidade de resolver um exercício de acordo com sua capacidade [46].

A visualização de um problema proposto pelo sistema, com base no perfil do estudante, tem efeito positivo sobre o processo de ensino, especialmente para os aprendizes que possuem dificuldades, uma vez que é possível recomendar exercícios que condizem com suas habilidades. Este efeito pode ser justificado pela teoria de fluxo [15].

Para sugerir uma cláusula a ser aprendida no SQL-Tutor, o modelo pedagógico observa o nível do aprendiz, que varia de 1 a 9 e é proporcional ao número de restrições existentes na base. Caso esse nível do estudante seja menor que 3, uma das três cláusulas iniciais da SQL (SELECT, FROM e WHERE) é selecionada, visto que envolvem os problemas mais fáceis. Para os aprendizes que possuem nível maior que 3, o sistema considera

apresentar seis cláusulas na escolha do problema (SELECT, FROM, WHERE, GROUP BY, HAVING e ORDER BY). A cláusula candidata passa a ser aquela em que o aprendiz mostrou maior dificuldade. Esse procedimento é realizado considerando todas as restrições relevantes para uma cláusula e então é calculada a média das probabilidades do aprendiz conhecer essas restrições [45].

Após escolha a cláusula a ser exercitada, o SQL-Tutor procura por problemas correspondentes. Dentre todos os problemas relevantes à cláusula em questão, o sistema seleciona aqueles que são apropriados ao nível atual do aprendiz. Estes são problemas cujos níveis são iguais ou superiores à habilidade atual do aprendiz.

Sistema Tutor Inteligente PHP ITS

O sistema PHP ITS (*PHP Intelligent Tutoring System*) é uma ferramenta utilizada para ensinar desenvolvimento web para programadores iniciantes. Ele procura orientar cada aprendiz para que estude os conteúdos mais adequados ao seu nível de conhecimento. Isto é possível pois os atividades são sugeridas por meio de uma lista de exercícios, classificadas por ordem de indicação. O exercício mais indicado é a escolha padrão proposta pelo sistema, mas o aprendiz pode selecionar qualquer outro exercício da lista, se desejar [60].

A escolha automática do próximo exercício é benéfica para individualizar e aperfeiçoar a interação em um STI. No entanto, o PHP ITS nem sempre é exato ao estimar o conhecimento do aprendiz e mesmo que o sistema faça corretamente essa estimativa, há espaço para que alguns aprendizes não aceitem as sugestões, seja porque desejam exercitar mais sobre um conteúdo, ou mesmo para passar adiante [60].

A seleção do próximo melhor exercício no PHP ITS é baseada em um método próprio que identifica o aprendizado. Esse método estabelece um limite de probabilidade acima do qual um conteúdo é considerado como aprendido. O sistema assume o valor 0,85 como o limite inferior de probabilidade para considerar que o conteúdo foi aprendido [60].

Neste sistema, é considerado o exercício mais adequado ao indivíduo, aquele que abrange o menor número de conteúdos ainda não aprendidos. Caso exista mais de um exercício com a mesma quantidade de conteúdos, a escolha pode ser feita aleato-

riamente [60].

O motivo para selecionar os exercícios com menor número de conteúdos desconhecidos garante que haverá pouca quantidade de material novo a ser apresentado. Dessa forma, o estudante constrói seu conhecimento sem visualizar múltiplos conteúdos novos em um único exercício, evitando uma sobrecarga cognitiva.

2.2 Teoria do raciocínio qualitativo

A teoria do Raciocínio Qualitativo (RQ), do inglês *Qualitative Reasoning Theory*, é um meio de expressar o conhecimento conceitual, tais como a estrutura física de um sistema, a causalidade, o início e o fim de processos, os pressupostos e condições para que os fatos sejam verdadeiros e o comportamento qualitativamente distinto [11].

O RQ é visto como uma área da inteligência artificial que se preocupa com a descrição das propriedades contínuas do mundo. Essas propriedades são o tempo, o espaço e as quantidades, que usam um sistema discreto de símbolos para dar suporte ao raciocínio automatizado com pouca informação [6].

O RQ é uma teoria que auxilia a construção de modelos que capturam o conhecimento de especialistas na estrutura, comportamento e funcionalidade dos sistemas. As abordagens em RQ proporcionam vocabulário para descrever objetos, situações, relações, causalidade, premissas e mecanismos de mudanças. Usando esse vocabulário, é possível capturar conhecimento conceitual sobre sistemas e seu comportamento e usar tal conhecimento para produzir, automaticamente, conclusões relevantes [2].

Para Golinska, o RQ surgiu como um subcampo da inteligência artificial para lidar com representação e raciocínio sobre aspectos contínuos de entidades em sistemas simbólicos, mas de forma semelhante ao raciocínio humano. A principal questão na abordagem RQ é desenvolver uma ferramenta adequada para situações de modelagem em que a informação não é suficientemente precisa ou não pode ser descrito por valores numéricos [29].

O uso do RQ permite a geração automática de explicações em um contexto de ambientes de aprendizado interativo. Esses ambientes são aplicativos que permitem ao modelador construir modelos capazes de gerar explicações e predições sobre o comportamento de um

sistema, a partir da descrição de sua estrutura [8].

Os métodos baseados em RQ permitem a representação do conhecimento integrado das características e comportamentos relevantes dos sistemas, do mesmo modo que a modelagem tradicional, sem requerer descrições analíticas aprofundadas ou informações numéricas completas sobre as funções [7]. Esses métodos proporcionam a integração do conhecimento, apresentado em modelos autoexplicativos [32].

O raciocínio qualitativo é, portanto, um campo de pesquisa destinado a solucionar diversos problemas e tem produzido uma variedade de teorias, ferramentas e aplicativos.

Algumas características do RQ são típicas para muitas abordagens existentes. Algumas das mais importantes são:

- **Causalidade:** A análise e a explicação do comportamento de um sistema em termos de causa-efeito são centrais para o raciocínio humano e para a comunicação dos resultados. Essas representações não estão presentes nos modelos matemáticos. A formalização desses conceitos e a aplicação no raciocínio automatizado é a base para gerar explicações nos modelos baseados em RQ [11].
- **Modelagem composicional:** O RQ utiliza uma biblioteca de fragmentos de modelo que proporciona as bases para automatização e reuso, em vários contextos. Essa é uma característica altamente desejável, tanto para o desenvolvimento teórico quanto para trabalho em aplicações reais [6].
- **Inferência do comportamento a partir da estrutura:** Muitos sistemas em RQ requerem como entrada a descrição estrutural de um sistema (em termos de objetos inter-relacionados), cujo comportamento é gerado automaticamente e descrito em uma simulação qualitativa, na qual estejam representados todos os comportamentos possíveis. Para o RQ, as simulações qualitativas, em que o usuário pode interagir com aspectos diversos do modelo, são chamadas simulações articuladas [24, 12, 11].
- **Qualidade das representações:** Em RQ são representados apenas comportamentos qualitativamente distintos, essenciais para a solução de um determinado problema [11].

- **Formalização de conhecimento:** Funções matemáticas são representadas em RQ por meio de álgebra qualitativa [32], ordem de magnitude, derivadas, relações monotônicas [26] e equações diferenciais qualitativas [39], entre outras.
- **Ontologias:** As ontologias contêm os elementos essenciais para a compreensão do mundo e são centrais para o RQ. Algumas famílias de ontologias são bem conhecidas, entre elas, a abordagem centrada em componentes [37], a abordagem centrada em processos [26] e a abordagem centrada em restrições [39].

Ontologias são modos de ver e descrever certos aspectos existentes em algo [6]. Nesse sentido, as ontologias contêm os elementos essenciais para uma representação do mundo. O uso de ontologias na computação tem origem em conceitos gerais da filosofia e da linguística, cujos fundamentos partem da premissa de que a formalização do conhecimento começa com a conceituação, que por sua vez consiste em um conjunto de entidades sobre a qual o conhecimento é expresso e a relação entre eles. Essencialmente uma ontologia consiste de termos, suas definições e seus axiomas [52].

2.2.1 Abordagens da teoria qualitativa dos processos

Forbus divide a teoria em três ontologias bem desenvolvidas [25]:

- Centrada em componentes [37],
- Centrada em restrições [39] e,
- Centrada em processos [25].

Comparada com as outras duas abordagens, a abordagem centrada em processos é considerada a mais promissora para a construção de modelos sobre sistemas, especialmente os ecológicos [53].

Abordagem centrada em componentes

Essa ontologia foi descrita inicialmente por Kleer e Brown [37]. A realidade é representada como sendo constituída de componentes que manipulam materiais e conexões através

das quais fluem os materiais, sendo o comportamento do sistema ditado pela estrutura dos componentes e conexões. Cada componente é descrito por determinados estados qualitativos, sendo que cada estado qualitativo descreve uma dada situação no tempo [20].

As principais características dessa abordagem são [6]:

- O mundo é modelado como componentes, que manipulam materiais, e conduítes que transportam esses materiais. O comportamento é explicitado na forma como os materiais são manipulados e transportados entre os componentes em uma biblioteca de modelos de componentes. Nessa descrição, um componente é associado com confluências, correspondentes às relações entre variáveis que equivalem às características dos materiais. O modelo de certo componente pode consistir de um número de estados qualitativos, cada um especificando um estado particular de comportamento.
- Variáveis qualitativas podem assumir apenas um número restrito de valores que têm significado para a descrição qualitativa dos fenômenos. Esse conjunto de valores é chamado espaço quantitativo da variável. Uma álgebra qualitativa é requerida para combinar esses valores qualitativos. Em uma abordagem baseada em componentes, um espaço quantitativo, formado por valores que representam, se a quantidade cresce, decresce ou se mantém constante é suficiente para muitas aplicações. A abordagem baseada em componentes foi desenvolvida na eletrônica, um domínio em que modelos são representações mais próximas dos sistemas reais.
- Os sistemas têm componentes e conduítes dispostos em topologias bem definidas, construídas para atingir comportamentos específicos. Uma vez definida, a topologia não pode ser alterada, o que dificulta representar coisas que aparecem e desaparecem durante o processo. O comportamento de seus componentes pode ser entendido pela aplicação de leis físicas bem estabelecidas. Esse trabalho deu origem a uma área importante de pesquisa, denominada diagnose baseada em modelos.

Abordagem centrada em processos

Na abordagem centrada em processos, os sistemas físicos podem ser pensados em termos de objetos (entidades) e suas propriedades. Nesse processo, as quantidades e relações mudam pela ação de processos [26].

A possibilidade de representar situações, descrever os mecanismos que causam as mudanças e de representar explicitamente as relações de causalidade são características que justificam a escolha dessa abordagem para a representação de problemas físicos [49].

Abordagem centrada em restrições

Nesta abordagem não há qualquer representação explícita de entidades do mundo real, ou biblioteca de fragmentos de modelo. O ponto de partida, de forma simplificada é a equação diferencial qualitativa (QDE), uma abstração de equações diferenciais ordinárias (ODE). Ou seja, sua hipótese baseia-se na premissa de que ODE's podem ser reescritas sob a forma de QDE [39]. Estas equações podem ser usadas para simulações qualitativas, que fornecerão descrições do comportamento do sistema no mundo real [6].

Na abordagem centrada em restrições, é a versão qualitativa dos conjuntos de equações diferenciais que descrevem um sistema, não havendo representação explícita das entidades do sistema. Parte-se do fato que muitos sistemas podem ser convenientemente descritos por um conjunto de parâmetros que variam, e funções que modelem essas variações, sendo essencial a definição dos valores iniciais, os limites e os pontos de descontinuidade, sendo também os espaços definidos em termos de pontos e intervalos entre esses pontos, com as mudanças ocorrendo através de descontinuidades entre intervalos ou vice-versa [20].

A abordagem centrada em restrições não é uma ontologia completa como as outras duas. É uma matemática qualitativa, formalizada para dar suporte à predição de comportamento a partir de equações qualitativas ou representações das restrições entre as variáveis [20].

Sozinha, essa abordagem é tão inadequada para construir sistemas tutoriais quanto o são os modelos numéricos. Não há representação explícita de relações causais. A única relação causal disponível é a sequência de saída de valores obtidos depois das restrições se-

rem satisfeitas. Essa abordagem pode ser combinada com outras abordagens para superar essas limitações [6].

Dentre as abordagens citadas, a abordagem centrada em processos deu origem a uma ontologia denominada teoria qualitativa dos processos. Essa teoria permite que processos sejam modelados usando relações causais entre objetos. Estes objetos são abstrações de elementos do mundo real, portanto, essa pode ser considerada uma abordagem eficiente para representar situações variadas, sem que o modelador tenha necessidade de aprender elementos muito específicos e/ou restritivos.

2.2.2 Teoria qualitativa dos processos

Na Teoria Qualitativa dos Processos (TQP), o objeto é a menor unidade do sistema que está sendo modelado [49]. Nos modelos, um sistema é uma coleção de objetos inter-relacionados [26]. Os objetos são representados de maneira estruturada e hierarquizada. Por consequência, as características definidas para objetos em níveis mais altos são herdadas por objetos em níveis inferiores [53].

A TQP é uma das ontologias mais desenvolvidas em RQ. Nessa abordagem, o mundo é modelado como consistindo de objetos cujas propriedades são descritas por quantidades. Embora, inicialmente, qualquer coisa possa ser representada como objeto, existe um compromisso nessa ontologia, de fazer com que a representação dos objetos seja a mais próxima possível de como as pessoas percebem o mundo. O comportamento do sistema é determinado quando os objetos são criados, destruídos ou são alterados. A hipótese central da TQP é que apenas processos causam mudanças nos objetos. Toda a mudança no sistema deve-se à ação de um processo. Esse pressuposto é conhecido como “mecanismo único” e deve-se assumir que esse mecanismo trata dos fenômenos na escala temporal de curto prazo. Esses fenômenos são perceptíveis por pessoas em trabalho de observação e reflexão [6].

Assume-se então que todas as mudanças do sistema são causadas direta ou indiretamente por processos. Uma situação é definida como uma coleção de objetos, suas propriedades, as relações entre elas e os processos que estão ocorrendo. Nessa ontologia, é

possível a descrição de novas entidades criadas por processos ao longo de uma simulação. Esse ponto contrasta com a abordagem centrada em componentes, em que a topologia do sistema pode ser complementada apenas no início da simulação [6]. Assim, a partir da teoria qualitativa de processos tem-se uma definição estrutural de como processos e quantidades interagem, permitindo construir modelos para quaisquer processos do mundo real [57].

Um sistema especialista é muitas vezes um modelo superficial do seu domínio de aplicação, no sentido de que as conclusões são tiradas diretamente de características observáveis da situação apresentada. Os pesquisadores afirmam que o desempenho do especialista deve estar atrelado ao conhecimento dos modelos de profundidade, em que um mecanismo subjacente, cujos estados variáveis podem não ser diretamente observáveis, explica os fatos observáveis.

Uma grande linha de pesquisa para a representação de modelos de profundidade é o estudo de modelos causais qualitativos. Pesquisas sobre modelos causais qualitativos diferem de trabalhos mais gerais sobre modelos de profundidade com foco em descrições qualitativas do mecanismo de profundidade, capazes de representar o conhecimento incompleto da estrutura e do comportamento do dispositivo. A manipulação simbólica de descrições qualitativas também parece ser um modelo plausível para a experiência humana [39].

O raciocínio causal qualitativo consiste de um número de diferentes operações. Um conjunto de equações de restrição descrevendo as relações estruturais relevantes em um sistema pode ser derivada através de um exame de sua estrutura física. Os possíveis comportamentos do sistema podem ser previstos por simulação qualitativa das equações de restrição e um estado inicial. A descrição de comportamento pode ser utilizada para explicar um conjunto de observações ou a forma onde um mecanismo produz este comportamento [39].

Os pesquisadores que trabalham em diferentes domínios de problemas têm levado muitas abordagens diferentes para a derivação de equações de restrição de estrutura física. Kleer e Brown descrevem um sistema físico em termos de componentes e co-

nexões. Equações de restrição são derivadas dos modelos de componentes e dos caminhos de interação fornecidas pelas conexões. Este ponto de vista levou a propor princípios da boa forma, como “não-função in-estrutura”, que afirma que os modelos de componentes devem ser formulados independentemente dos contextos dos dispositivo em que serão exibidos [37].

Estudando fundamentação física sobre situações cotidianas, Forbus determina o conjunto atual de processos ativos. As equações de restrição são obtidas a partir de todo o conjunto de processos ativos no momento [25]. Trabalhando principalmente em fisiologia médica, Kuipers trata equações de restrição como dado, ou por um livro de texto ou de aprendizagem experimental, mas fora do âmbito da resolução de problemas de causalidade imediata [39].

A inferência central dentro de todas estas abordagens é a simulação qualitativa: derivação de uma descrição do comportamento de um mecanismo a partir das equações de restrição qualitativa.

Equações diferenciais proporcionam uma analogia útil, como pode ser observado na Figura 2.2 [39]. A equação diferencial descreve um sistema físico em termos de um conjunto de variáveis de estado e restrições. A solução para a equação pode ser uma função que representa o comportamento do sistema ao longo do tempo.

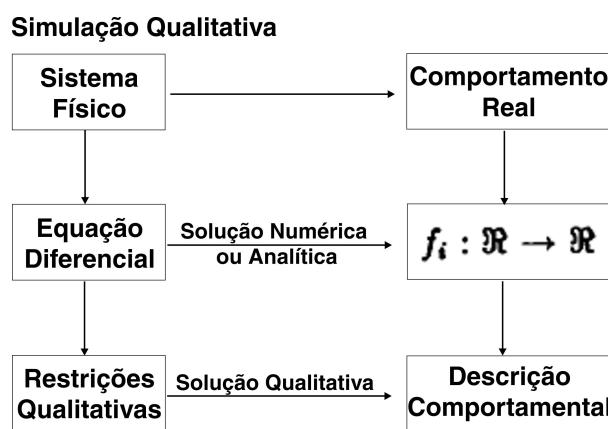


Figura 2.2: Comportamento da simulação qualitativa e das equações diferenciais

Uma descrição da estrutura em termos de equações de restrição é uma abstração adicional do mesmo sistema, e a simulação qualitativa destina-se a produzir uma abstração

correspondente do seu comportamento.

Todos os sistemas de simulação qualitativos descrevem quantidades em termos de suas relações ordinais com um pequeno conjunto de valores de referência. Os autores Kleer e Brown normalmente consideram um único ponto de referência a ser zero, e, assim, definem três valores qualitativos, $[+, 0, -]$ [37].

Embora esse valores permitam a possibilidade de espaços de quantidade mais complexos, as definições de adição e multiplicação como operadores ao longo de valores qualitativos não são úteis para uma situação mais complexa, e todos os seus resultados usam a semântica $[+, 0, -]$.

Um marco diferente de zero de uma quantidade de “x” pode ser acomodado através da definição de uma quantidade auxiliar “ $y = x-a$ ”, que refere-se a zero, e o valor “ $x = a$ ”. Valores que definem limites da região operante também podem ser usados, mas eles não são parte das operações de adição e multiplicação qualitativas.

Forbus e Kuipers definem uma quantidade de espaço como um conjunto parcialmente ordenado de valores de referência, de modo que uma quantidade é descrita em termos de suas relações ordinais com os marcos [25, 39].

A abordagem de Kuipers é diferente dos demais pesquisadores por permitir que novos marcos possam ser descobertos durante a simulação qualitativa, e usados para definir novas distinções qualitativas. O algoritmo denominado QSIM apresentado nos trabalhos de Kuipers descrevem quantidades em termos de um conjunto linearmente ordenado de pontos de referência, mas ainda permitindo que novos marcos possam ser descobertos e inseridos [39] .

Diferentes sistemas de simulação qualitativos tomam posições diferentes sobre se as quantidades que devem ser uma abstração da noção matemática padrão de números reais ou se um modelo fora do padrão deve ser usado, permitindo que dois pontos serem infinitamente separados.

Forbus, De Kleer e Brown adotam modelos diferentes do padrão de tempo em que “mítico” ou tempo “infinitesimal” podem separar estados qualitativos que correspondem ao mesmo ponto no tempo físico. Tais sequências de estados parecem ser necessárias

quando o ciclo de inferência computacional deve executar mais de uma vez para gerar um estado correspondente ao próximo estado físico [25, 37].

Os autores De Kleer e Brown adotam o modelo padrão para quantidades, mas são menos comprometidos com pontos e intervalos alternando no domínio do tempo, enquanto Kuipers segue o modelo padrão [37, 39].

2.2.3 Definições

A seguir serão definidos alguns conceitos básicos da TQP, como os elementos e os termos característicos dessa abordagem.

Objetos

O objeto é a menor unidade do sistema que está sendo modelado [49]. Nos modelos, um sistema é uma coleção de objetos inter-relacionados [26]. Os objetos são representados de maneira estruturada e hierarquizada. Por consequência, as características definidas para objetos em níveis mais altos são herdadas por objetos em níveis inferiores. A hierarquia de objetos fornece informações sobre as relações que são usadas para modelar a estrutura de dependência entre os objetos [6].

Quantidades

Os objetos têm diversas propriedades. As propriedades mais relevantes são representadas no modelo como quantidades, usadas para descrever como os objetos mudam ao longo do tempo. As quantidades são descritas com base na magnitude e na direção de mudança (denominada aqui de derivada) [6].

A magnitude refere-se à noção, no senso comum, do tamanho da quantidade. A derivada é uma noção requerida na dinâmica do modelo para indicar como a quantidade está mudando. Uma representação completa do valor de uma quantidade é feita por um par $\{magnitude, derivada\}$. Cada uma das duas partes é composta de valor e sinal. Por exemplo, para derivadas os sinais $\{-, 0, +\}$ indicam a direção de mudança da quantidade,

ou seja, $\{-\}$ significa que a quantidade está diminuindo ao longo do tempo; $\{+\}$ representa uma quantidade que está crescendo ao longo do tempo; quantidades constantes são indicadas com $\{0\}$ [6].

Espaços quantitativos

Os valores qualitativos que uma quantidade pode assumir são descritos em um conjunto denominado espaço quantitativo (EQ). O EQ pode incluir pontos, intervalos ou ambos. Mudanças de valores representam o comportamento de uma quantidade. O comportamento do sistema é uma sequência de estados qualitativos, observada ao longo do tempo, caracterizados por conjuntos de valores das variáveis, que aparecem simultaneamente [6].

Estados qualitativos

Os estados qualitativos do sistema são descritos por um conjunto de valores de algumas quantidades. Um estado qualitativo pode incluir um número infinito de valores quantitativos [26].

Processos

Os processos são os mecanismos que causam mudança, criando e destruindo objetos ou modificando suas propriedades. Os processos descrevem mecanismos de mudança. A descrição dos processos inclui conhecimentos sobre objetos e quantidades, condições para que os processos estejam ativos e as relações entre essas quantidades [26].

Visões individualizadas

As visões individualizadas, são definidas em termos de objetos, quantidades e relações quantitativas. Esses elementos são combinados para representar a estrutura do sistema. Na TQP, as visões e os processos formam blocos conhecidos como fragmentos de modelo, utilizados para a construção de modelos qualitativos.

Visões podem ser aplicadas em situações diferentes como cenários para simulações ou para descrever os efeitos das mudanças no sistema [26]. As visões individualizadas apresentam as relações entre objetos e quantidades que podem mudar [6].

As visões e processos incluem os elementos necessários para a representação de estruturas conceitual, causal e matemático [53] e proporcionam o vocabulário necessário para descrever o comportamento do sistema.

Uma visão consiste de quatro componentes [6]:

- **Indivíduos:** são listas de objetos, entidades e situações sob as quais visões e processos são aplicáveis. Podem ser instanciados para descrever uma situação específica.
- **Pré-condições:** são enunciados que representam indivíduos e suas relações, exceto aquelas estabelecidas pelas condições quantitativas. As pré-condições referem-se às condições externas necessárias para que visões e processos ocorram. Essas condições não são afetadas por processos e seus valores não podem ser deduzidos por inferências baseadas na TQP.
- **Condições quantitativas:** são declarações de desigualdades entre quantidades pertencentes aos indivíduos, ou ainda declarações sobre processos e visões individualizadas. Diferentemente das pré-condições, as restrições estabelecidas pelas condições quantitativas podem ser afetadas por processo.
- **Relações:** contêm enunciados que são verdadeiros, se e somente se, a totalidade da visão ou do processo é verdadeira. As relações são estabelecidas entre os parâmetros dos indivíduos e eventualmente são representadas novas entidades que são criadas. Entre as relações incluídas destacam-se as proporcionalidades qualitativas. Essas relações são usadas para representar como os efeitos dos processos se propagam pelo sistema.

Os processos são representados de forma similar às visões, exceto pela introdução de mais um componente: influências. Elas descrevem as mudanças que ocorrem quando os processos são ativos. Influências expressam os aspectos dinâmicos na TQP. Influências

diretas representam os efeitos dos processos e conferem os aspectos dinâmicos ao sistema. Esse campo não existe em visões, pois só os processos impõem influências diretas sobre as variáveis [6, 26].

Influências e proporcionalidades qualitativas

Na TQP, mudanças são sempre iniciadas por processos e se propagam para todo sistema mediante relações de dependência entre as quantidades. A TQP reconhece dois tipos diferentes de influências: influências diretas e influências indiretas, também denominadas proporcionalidades qualitativas – ambas possuem significado matemático e representam explicitamente o sentido das relações de dependência [26] :

- As influências diretas, ou simplesmente influências, aparecem nas definições de processos e são usadas para representar mudanças. As influências relacionam-se com a mudança de uma quantidade mediada por um processo e especificam como ou o que pode causar tal mudança na quantidade.
- As influências indiretas são usadas para calcular o valor da derivada de uma variável de estado. Ou seja, determinam a variação da derivada ao longo do tempo, também chamada de taxa de variação.

A taxa de variação é uma medida que refere-se a uma relação entre duas variáveis onde as restrições se estabelecem na derivada da quantidade ao invés da quantidade propriamente dita. As influências podem ser positivas ou negativas (I_+ , I_-), quando valores são adicionados ou deduzidos, respectivamente, da variável de estado [53].

A representação da mudança inicia-se pela influência e propaga-se por meio de proporcionalidades qualitativas. As influências indiretas ou proporcionalidades qualitativas (denotadas simplesmente pela letra P) descrevem como certa quantidade mudará pela sua dependência de outra quantidade. Elas expressam informação sobre uma dependência funcional entre dois parâmetros [26].

As proporcionalidades qualitativas formam o principal bloco para a construção de equações, representando funções monotônicas para as relações entre as derivadas de duas

quantidades. Essas relações monotônicas podem ser usadas para determinar mudanças em uma quantidade, com base em como outra quantidade do sistema muda [6].

Se uma função é negativamente proporcional, representa-se por “P-”, e se é positivamente proporcional, é representada por “P+”. A relação denotada apenas por P carrega a informação de que na relação entre duas quantidades quaisquer não é possível estabelecer que quantidade afeta a outra quantidade [26].

Causalidade aplicada

A formalização da noção de causalidade é central para a geração de explicação nos sistemas baseados em RQ [2]. Essa formalização significa que as explicações para o comportamento do sistema são obtidas em termos das relações de causa e efeito do tipo “A causa B”. Esta relação indica que quaisquer alterações em B ocorrem, se e somente se, A mudar. Desse mesmo modo, é possível inferir que “A causa C”, indiretamente, implicando que “A causa B” e “B causa C” [6].

Na teoria qualitativa de processos, as relações de causa e efeito são explicitadas por mudanças causadas direta e indiretamente por processos. As relações causais são explicitadas quando as mudanças decorrem de uma influência direta (influência) ou de propagação de efeitos a partir das proporcionalidades (influências indiretas). Uma quantidade que não sofre mudança em decorrência de quaisquer processos é uma constante. Esse conjunto de assertivas é denominado por Forbus de “hipótese de causalidade direta”. A noção de dependência entre quantidades e a causalidade que pode ser modelada por elas é uma importante característica dos modelos qualitativos [26].

Fragments de modelo

Uma das estratégias mais importantes na construção do modelo qualitativo é chamada modelagem composicional [21]. Nesse caso, modelos parciais, cada um chamado de fragmento de modelo (FM), que representam fenômenos ou aspectos mais ou menos independentes, são combinados para constituir modelos maiores [20].

Um FM contém informações sobre condições, consequências, relações entre as quantidades e entidades. Podem ainda ser organizados em uma hierarquia de subtipos. Esses subtipos são vinculados aos fragmentos de origem, dos quais herdam todas as propriedades inerentes [6]. Os FMs também especificam o comportamento causal em termos de influências, proporcionalidades, correspondências e desigualdades.

Os FM's podem ser de três tipos: a) fragmentos estáticos, b) fragmentos de processo, e c) modelos de agentes. Os FMs estáticos descrevem relações entre entidades e quantidades que não envolvem influências. Os fragmentos de processos descrevem comportamento causal em termos de influências, ou seja, descrevem os mecanismos naturais (processos físicos, químicos e biológicos) que promovem mudanças no sistema modelado. Os modelos de agentes representam os impactos exógenos no comportamento do sistema. Esses FMs referem-se, explicitamente, às mudanças que ocorrem no sistema em decorrência das ações antrópicas [8]. Fragmentos de modelo, armazenados em uma biblioteca, podem ser selecionados, reunidos, ativados e transformados em modelos de simulação [49].

Construir modelos qualitativos é, portanto, construir conjuntos de fragmentos de modelo (bibliotecas) a partir dos quais os modelos qualitativos podem ser formados. Essa abordagem dá grande flexibilidade ao trabalho de modelagem, visto que pode ser usado o mesmo conjunto de fragmentos de modelo para a construção de modelos sobre problemas de complexidade crescente [20].

2.2.4 Raciocínio qualitativo na educação

Os autores Bredeweg e Forbus discutem a modelagem qualitativa no processo educacional. A ideia é a de “aprender fazendo”, ou seja, novas possibilidades de experimentação que vão desde a definição da estrutura até a interação com os resultados da simulação, permitindo num processo iterativo e heurístico que o modelador amadureça no processo [9].

De acordo com os autores as abordagens em RQ proporcionam ferramentas básicas para a aquisição do conhecimento. Ou seja, permitem adquirir e articular ideias para uma representação formal de determinado campo do saber, induzindo o aprendizado [6].

A construção de modelos qualitativos é um processo complexo durante o qual uma

multiplicidade de aspectos deve ser administrada por um modelador [42]. Para criar um modelo é necessário não apenas identificar os objetos relevantes, suas interações, suas propriedades e quantidades importantes, mas também determinar o que é relevante e o não é. [11].

Ao desempenhar a tarefa de construir modelos qualitativos o modelador utiliza linguagem formal para expressar os elementos do modelo e desenvolver inferências sobre o sistema modelado [6].

2.3 Considerações finais

Este capítulo abordou sobre sistemas tutores inteligentes e sobre a teoria do raciocínio qualitativo. Os sistemas tutores inteligentes geralmente fazem o uso da IA tradicional como um meio de inferir comportamentos e estados possíveis para um estudante. No entanto, existem outras abordagens que também conseguem modelar estes comportamentos e estados. Uma abordagem considerada apropriada para esse processo é a teoria qualitativa de processos, que usualmente é destinada à sistemas físicos, no entanto pode ser aplicada à modelagens conceituais.

O raciocínio qualitativo possui um arcabouço vasto para a construção das modelagens conceituais ou físicas. Este material deve ser estudado atentamente para que os modelos sejam coerentes com o que se espera de um sistema real. Estabelecer quais serão os objetos usados, quais as influências e proporionalidades que existem entre eles e como as quantidades mudarão ao longo do tempo é algo importante e não deve ser negligenciado.

CAPÍTULO 3

ARQUITETURA DE UM STI BASEADO EM RACIOCÍNIO QUALITATIVO

Este capítulo apresenta uma proposta de arquitetura para um sistema tutor inteligente que usa a teoria do raciocínio qualitativo como base para a construção do modelo de domínio, descrevendo a visão geral da arquitetura proposta e seus componentes, e detalhando o principal componente dessa arquitetura, o PMA-TQP (Próxima Melhor Atividade baseada na Teoria Qualitativa de Processos) .

3.1 Visão geral

A arquitetura proposta neste trabalho usa a TQP (Teoria Qualitativa de Processos) como apoio para modelar um sistema tutor inteligente. Essa arquitetura está estruturada de acordo com os elementos que foram objetos de estudo da revisão bibliográfica. Optou-se por dividir a arquitetura proposta em dois níveis: um nível geral e um nível específico. A Figura 3.1 ilustra arquitetura proposta em um nível geral.

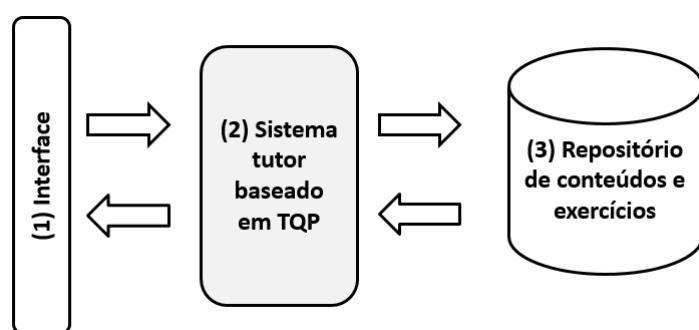


Figura 3.1: Arquitetura geral do sistema tutor

A Figura 3.1 ilustra os três componentes existentes no nível geral da arquitetura. O componente (1) representa a interface do sistema tutor, onde acontecerá a interação do aluno com o sistema. Nesta interface serão exibidos os conceitos a serem aprendidos e

as questões a serem respondidas pelo aluno. O componente (2) é denominado sistema tutor baseado em TQP e é o objeto de estudo principal neste trabalho. Este sistema está subdividido em componentes de software que representam o comportamento do sistema tutor. O componente (3) representa o repositório de conteúdos e exercícios que alimentará o sistema tutor. A seguir, serão descritos alguns detalhes sobre esses componentes.

- **Interface (1):** Responsável pela interação do aluno com o sistema, este módulo também realiza o monitoramento do estudante em relação ao seu histórico de atividades e em relação à sua sessão atual de trabalho, permitindo calcular e atualizar indicadores da evolução contínua do estudante.
- **Sistema tutor baseado em TQP (2):** Possui um conjunto de componentes internos que são capazes de eleger os conteúdos e exercícios a serem propostos. Essa recomendação é realizada baseada na resposta dos exercícios anteriores. Os componentes internos selecionam o próximo exercício (ou conteúdo) mais adequado para a continuidade do estudo. Este componente foi desenhado como um STI tradicional, mas acrescido de elementos da ontologia de raciocínio qualitativo.
- **Repositório de conteúdos e exercícios (3):** Compreende um conjunto de conteúdos e exercícios cadastrados por um professor no formato de questões de múltipla escolha; cada uma destas questões está relacionada com os conteúdos a serem apresentados (cada exercício pode estar ligado a vários conteúdos). Adicionalmente, as alternativas dos exercícios podem estar relacionadas diretamente com um ou mais conceitos de domínio do sistema tutor.

A funcionalidade geral dessa arquitetura pode ser ilustrada através de um exemplo genérico, através dos seguintes passos:

1. O estudante acessa a interface e seleciona a disciplina que deseja estudar;
2. O sistema apresenta o primeiro conteúdo da disciplina;
3. O estudante visualiza o conteúdo apresentado e clica na opção para responder um questionário sobre o conteúdo visto;

4. O primeiro exercício é apresentado ao aluno e uma resposta é solicitada;
5. O aluno responde o exercício selecionando uma das alternativas apresentadas;
6. Caso o estudante acerte totalmente o exercício, ele será direcionado ao próximo assunto;
7. Caso o estudante não acerte totalmente o exercício, o módulo PMA-TQP estimará qual é a questão mais indicada para ser apresentada;
8. O sistema repetirá o processo de escolha de próximo melhor exercício até que o estudante seja considerado apto para seguir ao próximo conteúdo do domínio estudado;

3.2 Sistema tutor baseado em TQP

Este componente é responsável por fazer a escolha do melhor próximo exercício ou conteúdo a ser apresentado para o estudante dentro do sistema. A Figura 3.2, ilustra os componentes internos deste módulo.

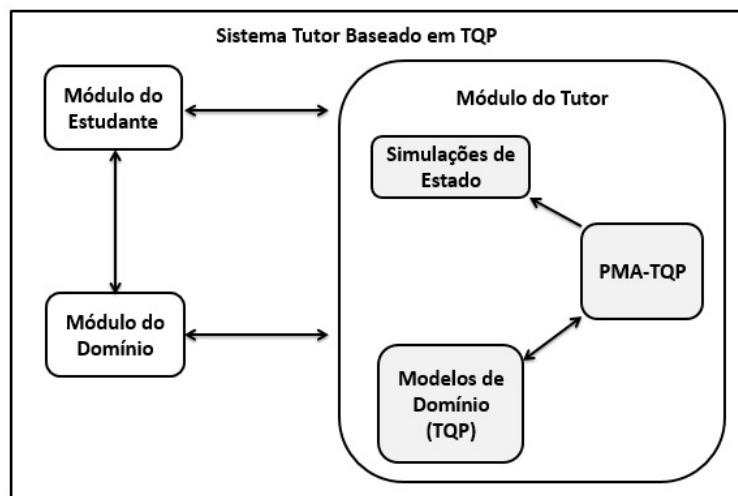


Figura 3.2: Arquitetura do sistema tutor baseado em TQP

Na Figura 3.2 nota-se que a arquitetura proposta possui os módulos clássicos dos sistemas tutores inteligentes: módulo do estudante, módulo do tutor e módulo do domínio (ou módulo especialista). O módulo do domínio está ligado ao módulo do tutor que possui o

componente denominado Modelos de Domínio (TQP), responsável por manter os modelos de domínio construídos de acordo com a teoria qualitativa de processos. Os modelos de domínio são usados pelo módulo de escolha da próxima melhor atividade (PMA-TQP). Esse módulo inicia as simulações de estado baseado em informações adquiridas do módulo do estudante. As simulações de estados são encaminhadas novamente ao PMA-TQP, que as analisará e indicará a melhor questão ou conteúdo ao estudante. A análise para recomendação de conteúdos e questões está detalhada nos próximos subcapítulos.

A escolha desses exercícios é realizada em duas etapas: a) simulação dos possíveis estados do estudante a partir de um estado inicial, e b) a proposição do próximo melhor exercício.

3.2.1 Simulação de estados

A arquitetura proposta precisa obter o estado de conhecimento do estudante para calcular e eleger as próximas melhores atividades. Ela obtém esses dados copiando para um cenário o estado atual do estudante. A partir disso, simulações de possíveis estados são geradas. Essas simulações são importantes para o sistema, pois a partir delas, o módulo PMA-TQP infere qual questão deve ser proposta.

Os cenários usam modelos de domínio baseados em TQP para gerar as inferências que são refletidas em grafos de estados. Estes modelos devem ser construídos por um especialista no domínio que conheça as premissas da TQP. A proposta do melhor exercício a ser apresentado será baseada na média geral obtida na simulação de estados do estudante.

Para que as simulações aconteçam, as entidades do sistema precisam estar identificadas. Os fragmentos de modelo representam as relações entre quantidades do modelo. As quantidades nesse modelo assumem valores qualitativos (zero, médio e alto). As relações descrevem a existência de proporcionalidades ou influências entre as quantidades que são descritas nos fragmentos de modelo.

Os modelos de domínio e os modelos de eleição de próximo melhor exercício apresentados nesse trabalho foram construídos no software GARP (*General Architecture for*

Reasoning in Physics)¹. O uso dessa ferramenta proporcionou a validação dos modelos. A implementação real desse módulo não foi objeto de estudo nessa dissertação, por esse motivo, optou-se por usar uma ferramenta que estivesse de acordo com a TQP para realizar as simulações necessárias.

A simulação de estados possíveis foi aplicada a dois modelos de estudo: um modelo geral, que demonstra como as proporcionalidades e as influências se comportam e um modelo de escolha do próximo melhor exercício. Estes modelos são necessários pois o módulo PMA-TQP, além de sugerir o melhor exercício, pode sugerir qual o próximo conteúdo a ser apresentado. Para que isso seja possível, os conteúdos também devem estar modelados segundo a TQP como um processo de aprendizagem.

3.2.2 Modelos de domínio com a teoria qualitativa de processos

Neste trabalho, os modelos de domínio baseados em TQP foram divididos em modelos para conteúdos e modelos para questões. Estes modelos estão descritos nos próximos subcapítulos.

3.2.3 Modelo de domínio para conteúdos

O modelo de domínio para conteúdos define como os conteúdos a serem aprendidos devem estar relacionados em uma cadeia causal. Os relacionamentos podem acontecer como influências diretas ou como proporcionalidades. Todo objeto nas relações da TQP tem um elemento quantidade, que possui como propriedades um valor e uma taxa de variação (derivada). No caso das influências, o valor da quantidade influenciada muda de acordo com o valor da quantidade influenciadora. No caso das proporcionalidades, o valor da quantidade influenciada sofre alteração de acordo com o valor da taxa de variação da quantidade influenciadora.

O início de uma modelagem TQP é a construção de um modelo conceitual que representa as relações entre os objetos do contexto a ser trabalhado. A Figura 3.3 ilustra o

¹Sistema especialista, escrito em Prolog, que possui uma metalinguagem associada a uma máquina de inferência. Este sistema possibilita automatizar os processos de raciocínio para diversos domínios do conhecimento [20]. O Apêndice A deste trabalho possui mais informações sobre o GARP.

modelo geral para conteúdos com TQP. Como esse é um modelo genérico, optou-se por chamar os conteúdos de A, B, C e D.

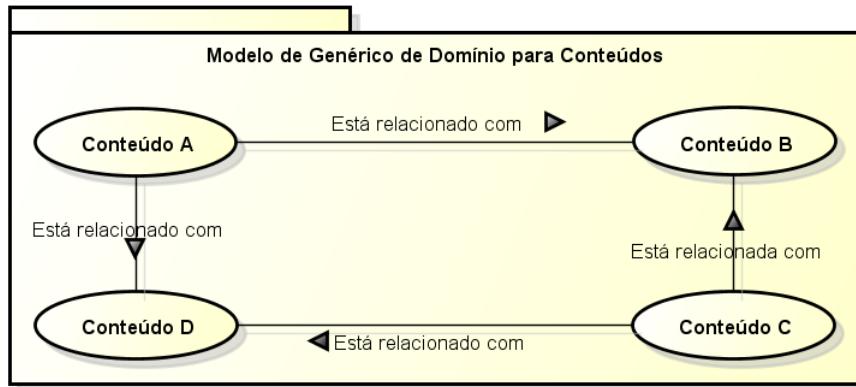


Figura 3.3: Modelo genérico conceitual para relacionamento entre conteúdos

O modelo conceitual da Figura 3.3 representa os conteúdos A, B, C e D e suas relações. Neste exemplo, o conteúdo A está relacionado com os conteúdos B e D. O conteúdo C está relacionado com os conteúdos B e D. Nessa etapa, não é necessário representar qual tipo de relação TQP (influência direta ou proporcionalidade) existe entre os elementos. Esta informação será representada no próximo passo, que é a contrução do modelo do processo em notação TQP. O modelo é construído com a combinação de elementos denominados fragmentos de modelo, gerados a partir de entidades que representam os objetos do relacionamento modelado.

O início da modelagem TQP na ferramenta GARP está representado na Figura 3.4. O elemento (a) da figura representa a criação das entidades do sistema, e o elemento (b) representa o conjunto de fragmentos de modelos criados para a situação modelada.

Para a criação desse modelo foi concebida uma situação onde existem quatro quantidades denominadas respectivamente de A, B, C e D, que representam os conteúdos de uma disciplina qualquer. Estas entidades estão relacionadas em uma cadeia de causalidade. Em notação da TQP, essa cadeia é representada da seguinte maneira: $I+(D, A)$, $P+(B, A)$, $P+(D, C)$ e $I+(B, C)$. A relação $I+(D, A)$ indica que A é uma taxa de mudança que determina a variação de D. Então se o valor de A é alto, D começa a crescer (taxa de

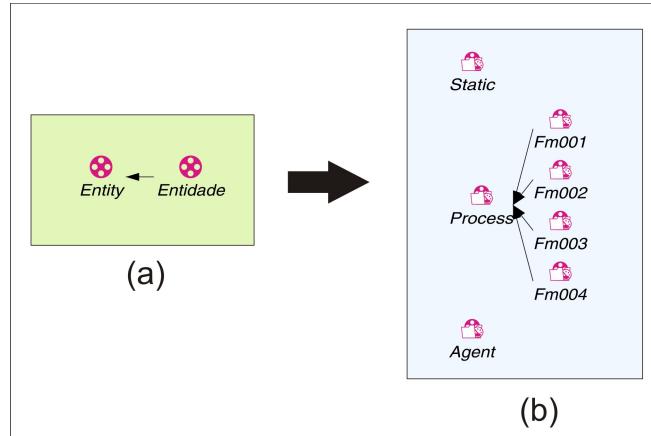


Figura 3.4: Modelo Geral de STI com RQ

variação sempre positiva). O valor $P+(B, A)$, indica que B muda no mesmo sentido de A (as taxas de variação de A e B são iguais). O mesmo raciocínio é aplicado as outras duas relações causais. A Figura 3.5 representa de forma ilustrada o conjunto de relações $I+(D, A)$, $P+(B, A)$, $P+(D, C)$ e $I+(B, C)$.

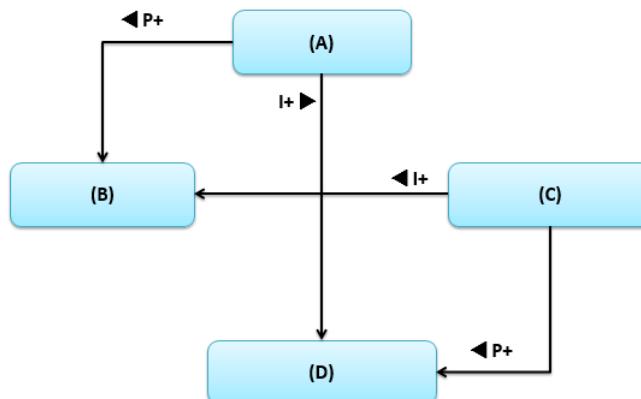


Figura 3.5: Modelo TQP para quantidades gerais

Nesse modelo, A e C são quantidades influenciadoras de B e D. Ou seja, quando os estados quantitativos de A e C são positivos, B e D terão suas derivadas positivas. Se os estados quantitativos de A e C forem menores que zero, as derivadas de B e D serão negativas. Os elementos B e D possuem derivadas idênticas as de A e C, respectivamente. Se A for positivo e C possuir derivada negativa, haverá uma invalidade no modelo, pois o valor qualitativo positivo de A indica que D possui por consequência derivada positiva.

3.2.3.1 Fragmentos de modelo

A Figura 3.6 ilustra cada um dos fragmentos de modelo criados para essa estrutura genérica, e que são descritos a seguir.

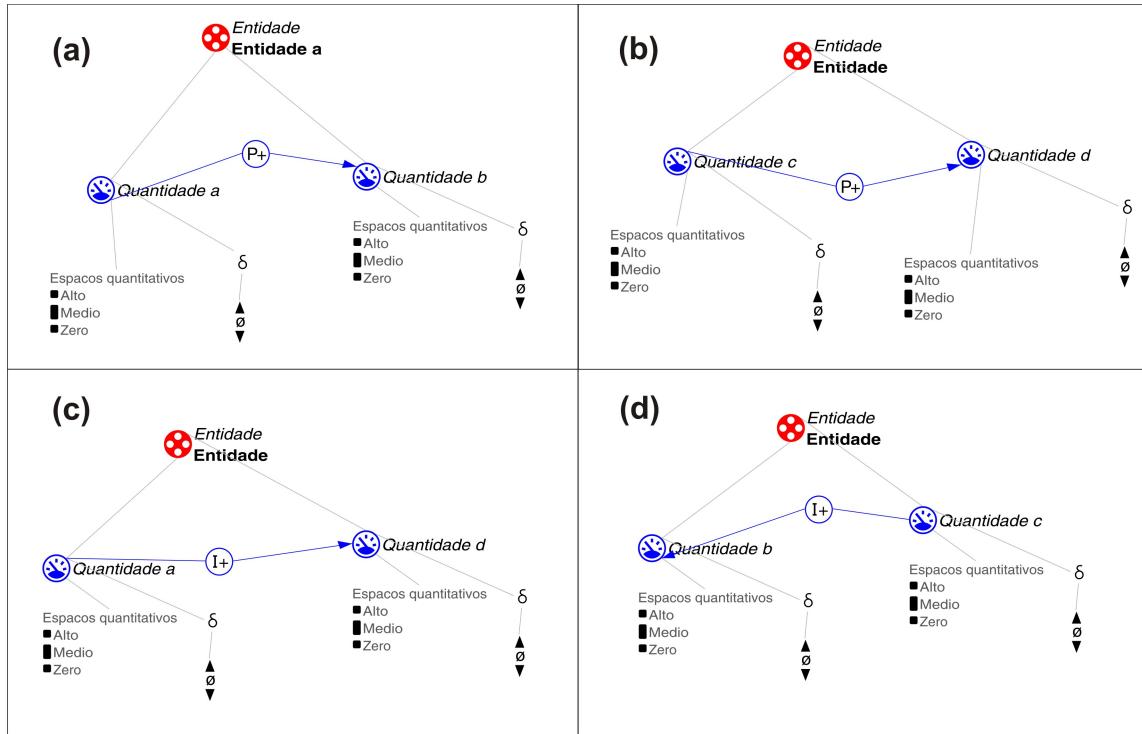


Figura 3.6: Fragmentos de modelo gerais para as relações $P+(B, A)$, $P+(D, C)$, $I+(D, A)$ e $I+(B, C)$

- Fragmento de modelo Fm001 (Figura 3.6 a): modela o relacionamento causal $P+(B, A)$, onde uma entidade possui duas quantidades associadas a ela. “Quantidade A” exerce uma proporcionalidade positiva ($P+$) sobre “Quantidade B”. Isso significa que, quando “Quantidade A” muda, ela exerce um efeito proporcional em “Quantidade B”. Esse efeito se espalhará no sistema caso “Quantidade B” esteja relacionada com outras quantidades.
- Fragmento de modelo Fm002 (Figura 3.6 b): modela o relacionamento causal $P+(D, C)$, onde “Quantidade C” exerce uma proporcionalidade positiva ($P+$) sobre a “Quantidade D”. Seu funcionamento é idêntico ao ilustrado no item (a).
- Fragmento de modelo Fm003 (Figura 3.6 c): modela o relacionamento causal $I+(D, A)$,

A), onde “Quantidade A” exerce uma influência positiva (I+) sobre “Quantidade D”. Essa influência faz com que, quando o valor da “Quantidade A” mudar, essa mudança exercerá uma influência proporcional sobre “Quantidade D” (as duas quantidades estão mudando na mesma direção). Essa influência denota que se “Quantidade D” estiver relacionada a outras quantidades, a relação de A para D não propagará as influências pelo restante do modelo.

- Fragmento de modelo Fm004 (Figura 3.6 d): modela o relacionamento causal I+(B, C), onde “Quantidade C” possui uma influência direta (I+) para com a “Quantidade D”. Essa relação segue a mesma lógica do fragmento de modelo Fm003 (Figura 3.6 c).

3.2.3.2 Análise dos cenários pelo PMA-TQP para conteúdos

Os fragmentos de modelo genérico descritos anteriormente trabalham com quatro quantidades que podem ser combinadas de várias formas em cenários positivistas, negativistas ou intermediários.

Um cenário positivista é aquele onde o estado inicial do conceito a ser aprendido é zero, mas o estudante começa a aprender, então seu valor qualitativo sobe para médio e depois para alto. Um cenário negativista é aquele no qual o estado inicial do conceito é alto, mas o estudante começa a errar as atividades propostas, indicando assim que seu nível de aprendizagem está decaindo. Neste caso, o espaço quantitativo do conceito deixa de ser alto para ser médio e zero. Um cenário intermediário é aquele onde as quantidades aumentam ou diminuem, mas não na mesma direção sempre. Este cenário pode ilustrar situações nas quais um estudante está aprendendo bem um conteúdo, mas, não está aprendendo outro satisfatoriamente.

Para o modelo geral de conteúdos, foram selecionados seis cenários considerados relevantes. O Cenário 1 demonstra os comportamentos de uma quantidade (conhecimento) isolada, sob a perspectiva positivista. O Cenário 2 apresenta o comportamento de uma única quantidade sob perspectiva negativista. O Cenário 3 apresenta o comportamento

de uma única quantidade sob perspectiva intermediária. O Cenário 4 apresenta o comportamento entre duas quantidades sob perspectiva positivista. O Cenário 5 apresenta o comportamento entre duas quantidades sob perspectiva negativista. O Cenário 6 apresenta o comportamento entre duas quantidades sob perspectiva intermediária.

Cenário 1: Comportamentos para quantidade única sob perspectiva positivista

A Figura 3.7 apresenta um exemplo de cenário positivista para uma quantidade (a), o grafo de estados gerados para esse cenário (b), e os estados resultantes (c).

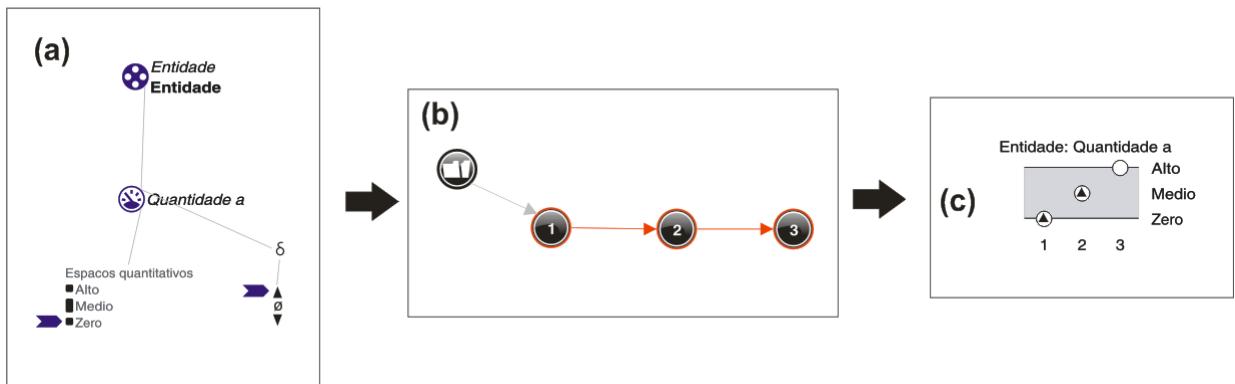


Figura 3.7: Cenário positivista para uma quantidade

No cenário ilustrado na Figura 3.7, apenas a “Quantidade A” faz parte do cenário. As setas azuis na ilustração indicam o estado inicial que se deseja aplicar a essa quantidade. No exemplo, (Figura 3.7 a), “Quantidade A” possui estado inicial zero e uma derivada positiva associada. Na TQP essa relação é sempre denotada por um par de elementos $\{magnitude, derivada\}$, conforme citado no Capítulo 2, e que nesse exemplo é escrito como $\{zero, +\}$.

O número de estados possíveis para as quantidades é proporcional ao número de quantidades e estados qualitativos existentes no cenário modelado. Em condições positivistas, num cenário com as quantidades variando entre os espaços quantitativos médio e alto pode-se estimar o total de estados usando um arranjo com repetição. Nesse arranjo, dois elementos (espaços quantitativos) são tomados um a um (número de quantidades

disponíveis), formando um total de 2 estados além do inicial ($1 + 2^1 = 3$ estados totais).

Essa modelagem resulta em três estados (Figura 3.7 b) porque soma o estado inicial do sistema com o número de variações quantitativas possíveis, elevado ao número de quantidades existentes no cenário. Ou seja, para esse cenário o total de estados é ($1 + 2^1$). Neste exemplo só existem duas variações possíveis para a situação indicada pela derivada, pois com a primeira quantidade sendo zero, restam apenas dois espaços quantitativos possíveis: médio e alto.

O elemento (Figura 3.7 c) mostra o gráfico de estados do sistema, onde cada estado é detalhado em relação aos seus espaços quantitativos. No exemplo ilustrado, “Quantidade A” inicia com o estado qualitativo zero. Como o exemplo é positivista (derivada é positiva), a “Quantidade A” subirá para o espaço quantitativo médio no estado dois. Esse mesmo movimento acontecerá no estado três, o que levará a “Quantidade A” atingir o espaço quantitativo alto.

Cenário 2: Comportamentos para quantidade única sob perspectiva negativista

Ao mudar os valores iniciais do cenário ilustrado na figura 3.7 para um cenário negativista, obtém-se o resultado ilustrado na Figura 3.8.

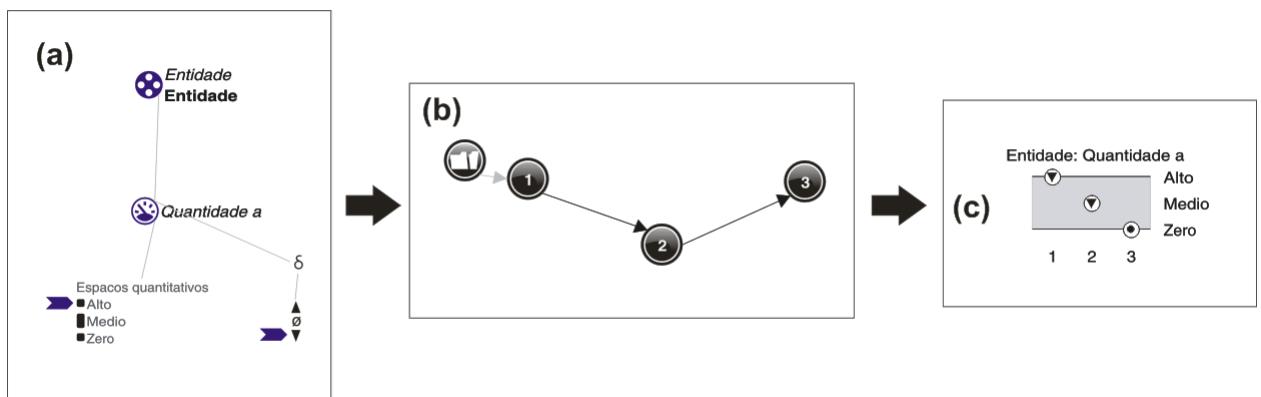


Figura 3.8: Cenário negativista para uma quantidade

O elemento (Figura 3.8 a) ilustra um cenário negativista. No exemplo, “Quantidade A” possui estado inicial alto e uma derivada negativa associada. Este cenário obedece às

mesmas leis do cenário positivista demonstrado anteriormente, mas começando em um espaço quantitativo alto e terminando em um estado quantitativo baixo (zero) (Figura 3.8 c).

Cenário 3: Comportamentos para quantidade única sob perspectiva intermediária

Ao mudar os valores iniciais do cenário ilustrado na figura 3.7 para um cenário neutro, obtém-se o resultado ilustrado na Figura 3.9.

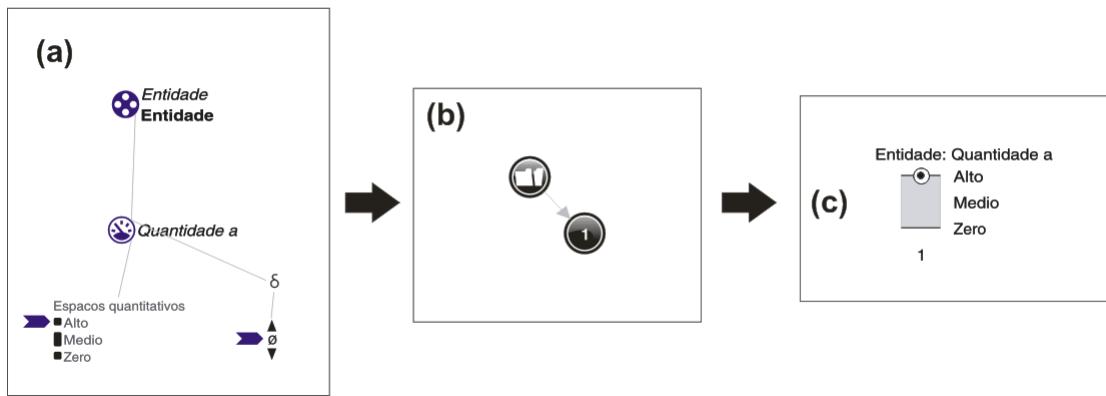


Figura 3.9: Cenário misto para uma quantidade

Na demonstração da Figura 3.9, a derivada possui valor zero e a magnitude possui o valor alto (Figura 3.9 a). Como a derivada é neutra (zero) o valor qualitativo para a quantidade em questão será seu valor inicial, pois a derivada é o indicativo de mudanças para as quantidades. O cenário possuirá apenas um estado (Figura 3.9 b) que está representado também no gráfico (Figura 3.9 c).

Cenário 4: Comportamentos entre duas quantidades sob perspectiva positivista

A cadeia de causalidade criada para esse exemplo influenciará todos os cenários criados. Conforme o número de quantidades dos cenários aumenta, a complexidade das simulações cresce exponencialmente. A Figura 3.10 ilustra um cenário com duas quantidades, A e D.

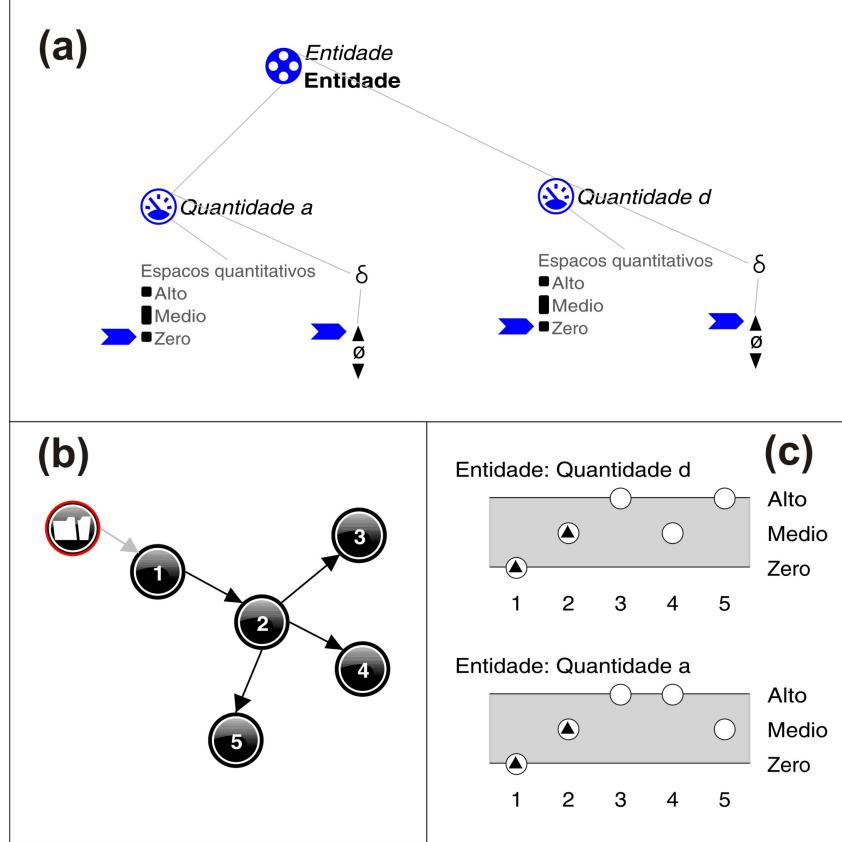


Figura 3.10: Cenário positivista para duas quantidade

A Figura 3.10 a, ilustra uma perspectiva positivista do criado. As quantidades A e D fazem parte dele. No exemplo, “Quantidade A” possui estado inicial zero e uma derivada positiva associada $\{zero, +\}$. O elemento “Quantidade D” possui as mesmas configurações $\{zero, +\}$.

Na mesma imagem, o quadro (Figura 3.10 b) indica o grafo de estados gerados para o cenário (Figura 3.10 a). Nesse arranjo, dois elementos (espaços quantitativos) são tomados dois a dois (número de quantidades disponíveis), formando um total de 4 estados além do inicial ($1 + 2^2 = 5$).

Ainda na Figura 3.10, o elemento (Figura 3.10 c) mostra o gráfico de estados do sistema, no qual cada estado é detalhado em relação aos seus espaços quantitativos. No exemplo ilustrado, “Quantidade A” inicia com o estado qualitativo zero. Como o exemplo é positivista (derivada é positiva), a “Quantidade A” subirá para o espaço quantitativo médio no estado dois. Esse mesmo movimento acontecerá no estado três, o que levará a

“Quantidade A” atingir o espaço quantitativo alto. Os demais estados demonstram as combinações possíveis entre médio e alto para as quantidades A e D.

Cenário 5: Comportamentos entre duas quantidades sob perspectiva negativista

O mesmo cenário pode ser visto de uma perspectiva negativa na Figura 3.11.

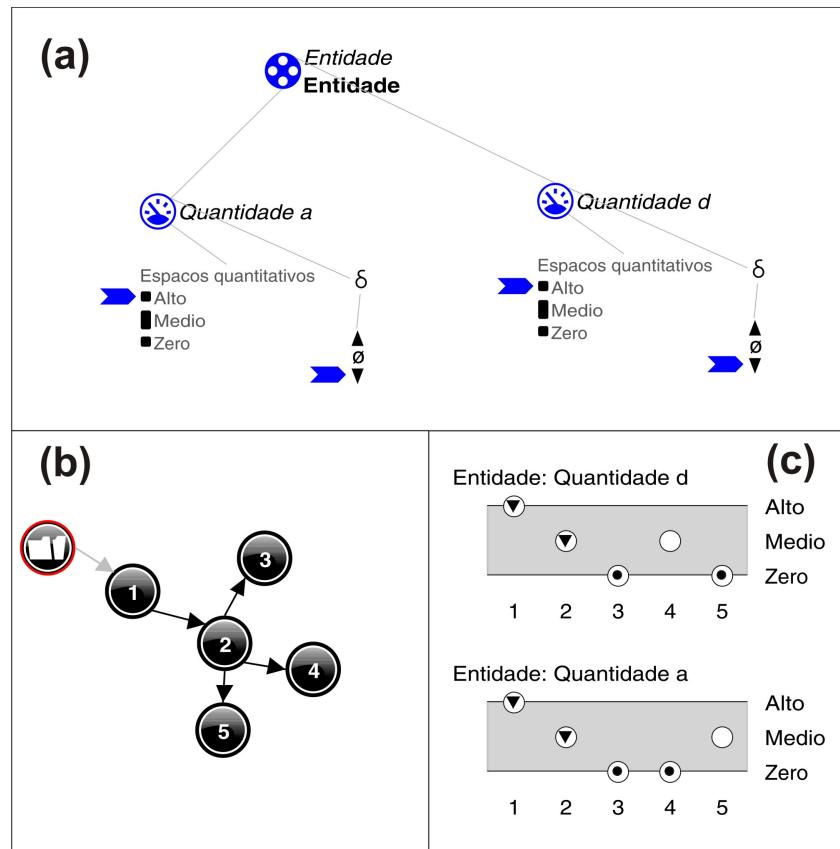


Figura 3.11: Cenário negativista para duas quantidades

No exemplo do quadro (Figura 3.11 a), “Quantidade A” possui estado inicial alto e uma derivada negativa associada $\{alto, -\}$. O elemento “Quantidade D” possui as mesmas configurações $\{alto, -\}$.

O quadro (Figura 3.11 b) indica o grafo de estados gerados para o cenário (a). Nesse arranjo, dois elementos (espaços quantitativos) são tomados dois a dois (número de quantidades disponíveis), formando um total de 4 estados além do inicial ($1 + 2^2 = 5$).

O elemento (Figura 3.11 c) mostra o gráfico de estados do sistema, no qual cada

estado é detalhado em relação aos seus espaços quantitativos. No exemplo ilustrado, “Quantidade A” inicia com o estado qualitativo alto. Como o exemplo é negativista (derivada é negativa), a “Quantidade A” descerá para o espaço quantitativo médio no estado dois. Esse mesmo movimento acontecerá no estado três, o que levará a “Quantidade A” atingir o espaço quantitativo zero. Os demais estados demonstram as combinações possíveis entre médio e zero para as quantidades A e D.

Cenário 6: Comportamentos entre duas quantidades sob perspectiva intermediária

O mesmo cenário pode ser visto de uma perspectiva mista, onde uma quantidade aumenta e a outra diminui. Essa situação é ilustrada na Figura 3.12.

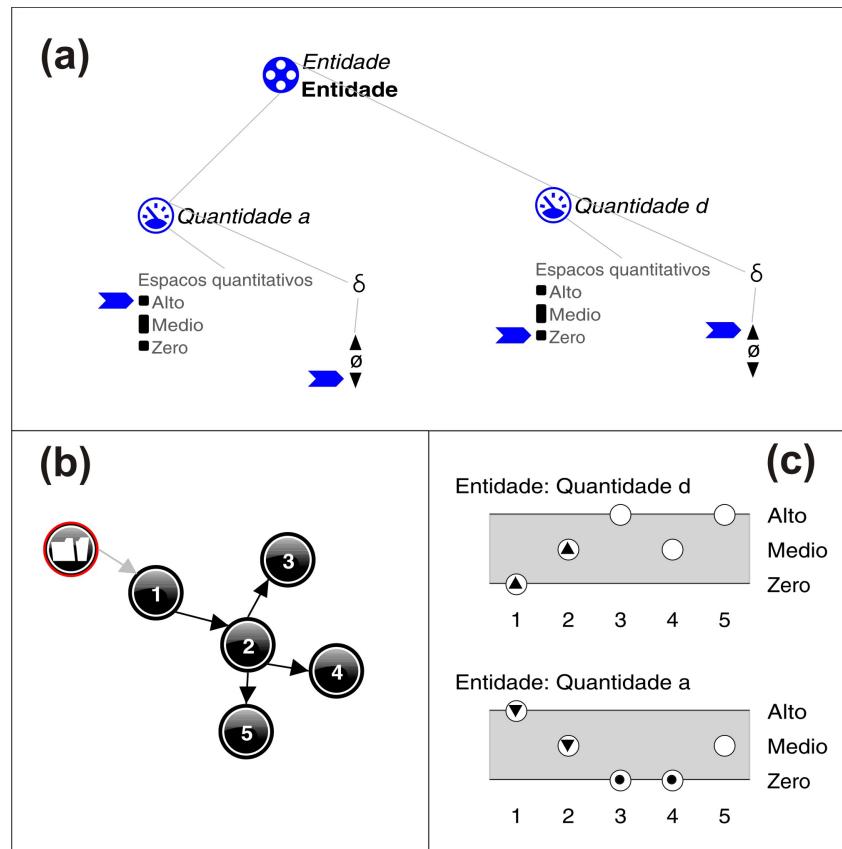


Figura 3.12: Cenário misto para duas quantidades

No exemplo do quadro (Figura 3.12 a), “Quantidade a” possui estado inicial alto e uma derivada negativa associada $\{alto, -\}$. O elemento “Quantidade D” possui estado

inicial zero e derivada positiva $\{zero, +\}$. O quadro (Figura 3.12 b) indica o grafo de estados gerados para o cenário (a). Nesse arranjo, dois elementos (espaços quantitativos) são tomados dois a dois (número de quantidades disponíveis), formando um total de 4 estados além do inicial ($1 + 2^2 = 5$).

Em seguida, o elemento (Figura 3.12 c) mostra o gráfico de estados do sistema, onde cada estado é detalhado em relação aos seus espaços quantitativos. No exemplo ilustrado, “Quantidade a” inicia com o estado qualitativo alto. Como o exemplo é mista (derivada diferentes), a “Quantidade A” descerá para o espaço quantitativo médio no estado dois. Esse mesmo movimento acontecerá no estado três, o que levará a “Quantidade A” atingir o espaço quantitativo zero. A “Quantidade D” subirá para o espaço quantitativo médio no estado dois. Esse mesmo movimento acontecerá no estado três, o que levará a “Quantidade D” atingir o espaço quantitativo alto. Os demais estados demonstram as combinações possíveis entre médio e zero para a quantidade A e entre médio e alto para a quantidade D.

3.2.4 Modelo de domínio para próximos melhores exercícios

O módulo PMA-TQP utiliza as informações obtidas das simulações de estudante para verificar se o aluno tem condições de responder uma questão. Ao identificar o estado do estudante, o sistema elege quais questões podem ser apresentadas a partir daquele momento.

Um exemplo disso seria a seguinte situação: um estudante está iniciando seus estudos, o sistema identifica que ele tem pouco conhecimento naquela disciplina e começa a apresentar os conteúdos de nível mais fácil. O sistema analisa e guarda o desempenho do estudante em cada questão. Caso o estudante tenha um comportamento positivo² em uma questão, o sistema assume como premissa que o estado inicial desse estudante corresponde a uma derivada positiva na TQP. Então, para as questões dos conteúdos relacionados que já foram aprendidos, são calculados os estados possíveis do estudante. O sistema calcula

²Comportamento no qual o estudante acerte o exercício totalmente. Neste modelo, isso significa que o estudante está no espaço quantitativo alto.

a média individual para cada cenário possível. O cenário que possui a média mais alta é apresentado ao estudante pois assume-se que será o cenário mais positivo para um acerto.

Usando os princípios da TQP (Teoria Qualitativa de Processos) aplicados em uma situação na qual deseja-se que um sistema indique qual é a melhor questão a ser apresentada a um estudante em um determinado contexto obtém-se uma modelagem de como o processo de indicação ou recomendação deve acontecer. Nesta arquitetura, definiu-se que as questões referentes aos conteúdos a serem aprendidos são modeladas como entidades do processo, assim como os conteúdos. Os conhecimentos, nessa arquitetura, são modelados como quantidades, que podem transitar entre valores qualitativos. Essa é uma associação válida para com a TQP, que modelada a interação entre entidades de um processo. Nesta arquitetura, o processo a ser modelado é a aprendizagem e as entidades são os conteúdos e suas questões. A Figura 3.13 a seguir, ilustra, como a entidade genérica foi criada, a partir da *Entity* do GARP, para ser usada por entidades filhas chamadas de questões.



Figura 3.13: Árvore de entidades para questões

A representação da Figura 3.13 indica que, a partir da entidade mais genérica disponível no GARP, foi instanciada uma nova entidade, que aqui possui apenas o nome de “Entidade”. As questões são derivadas dessa entidade e esse fato pode ser visto nos fragmentos de modelo.

3.2.4.1 Fragmentos de modelo para próximos melhores exercícios

Para o processo de recomendação de questões foram desenvolvidos cinco fragmentos de modelo, onde são representadas as influências e proporcionalidades entre as questões. A Figura 3.14 ilustra a árvore de fragmentos para elementos do tipo “questões”.

Os fragmentos desse processo são identificados por um nome, que para este trabalho consiste nas iniciais “F” e “m”, de fragmento de modelo, mais o nome da questão. Para este exemplo de funcionamento da arquitetura, foram criados dois grupos de conhecimen-

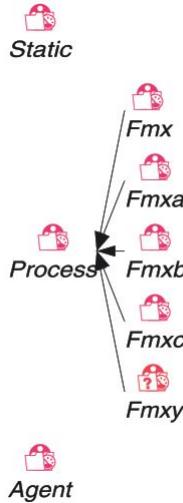


Figura 3.14: Árvore de fragmentos de modelos para questões

tos, cada um com três questões associadas a si. O grupo denominado “Conhecimento 1” possui as questões denominadas “xa”, “xb” e “xc”. O grupo denominado “Conhecimento 2” possui as questões denominadas “ya”, “yb” e “yc”. A representação usando esses nomes foi necessária por conta de algumas restrições que o software GARP apresenta. Nele não é possível nomear objetos quantidades com valores numéricos e nem com caracteres especiais.

Os elementos do Grupo X, foram associados a quantidades de acertos e erros. As quantidades, que representam os elementos de domínio, acertos e erros, podem transitar entre os valores qualitativos zero, médio e alto. A Figura 3.15 ilustra como as quantidades dos elementos de domínio foram associadas aos elementos de acerto e erro no GARP.

O quadro (Figura 3.15 a) ilustra a questão “x a” recebendo influência positiva (I+) da quantidade “Acerto” e recebendo influência negativa da quantidade “Erro”. Isso significa que quando a quantidade “Acerto” está no espaço quantitativo “Alto”, a derivada da questão “x a” será positiva. Quando a quantidade “Erro” está no espaço quantitativo “Zero”, a derivada da questão “x a” será negativa. Essa mesma situação se aplica nas ilustrações dos quadros (Figura 3.15 b e c). Essa abordagem foi adotada porque situações assertivas nesta arquitetura são vistas como a concretização do aprendizado do estudante e situações de erro são vistas como um fator de diminuição da quantidade de conhecimento do estudante naquela situação.

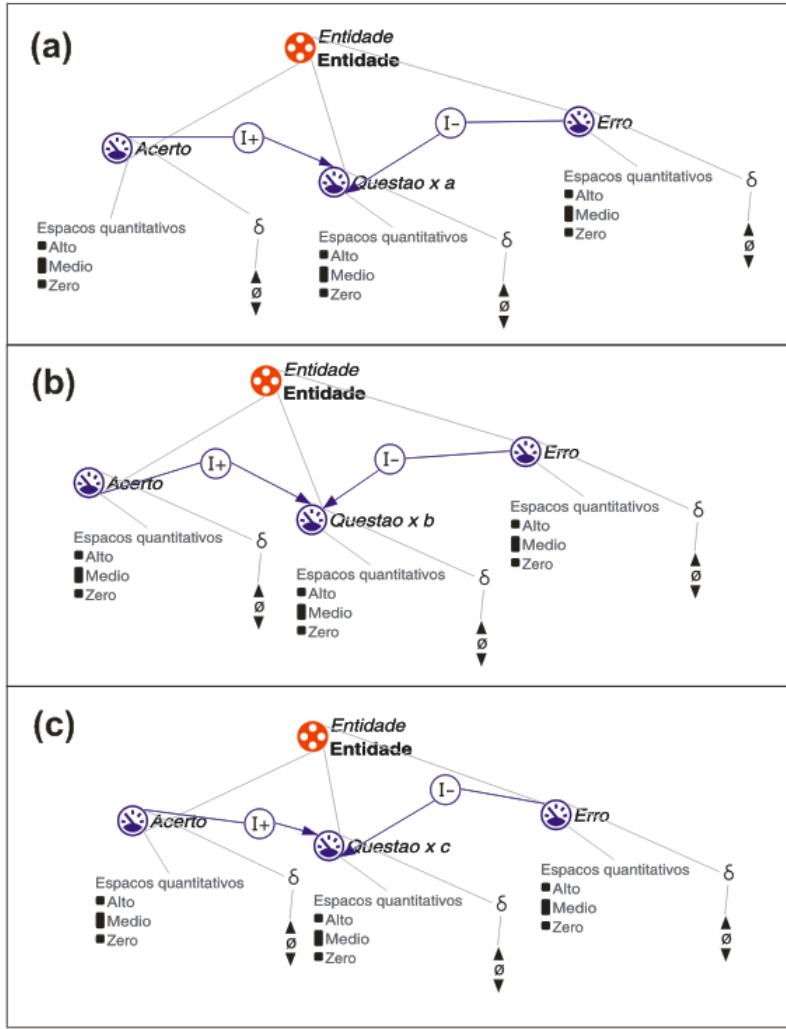


Figura 3.15: Fragmentos de modelo para o sistema de recomendação de questões

O conjunto de questões do Grupo Y não foi associado a quantidades de acertos e erros para demonstrar como o modelo se comporta com a falta dessas associações. Duas questões do Grupo X foram associadas entre si, conforme ilustrado na Figura 3.16.

Na Figura 3.16, está representado que a questão “x b” provoca uma proporcionalidade positiva ($P+$) sobre a questão “x c”. Para a TQP, isso significa que a derivada da questão “x c” se comporta como a derivada de questão “x b”. Ou seja, foi criada uma associação indicativa de que a questão “x c” se comporta como a questão “x b” durante as transições de espaço quantitativo. Em outros termos, essa modelagem indica que quando a questão “x b” estiver no espaço quantitativo alto, a questão “x c” pode ser recomendada pelo sistema como próxima questão a ser respondida pelo estudante. Nesse caso é provável que

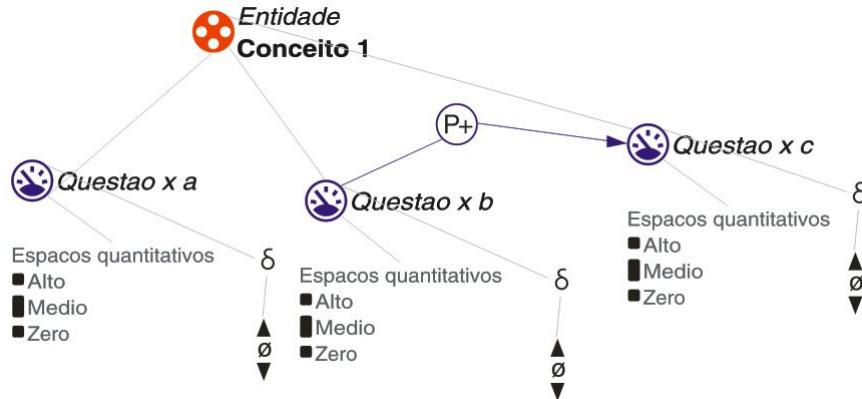


Figura 3.16: Associação entre as questões do Grupo X

esse estudante tenha um alto rendimento na questão “x c” assim como teve na questão “x b”. Caso o estudante acerte medianamente a questão “x b”, o resultado da recomendação será indicado pela média qualitativa das simulações que calcularão os estados do estudante. Se a média qualitativa obtida for maior que o valor do espaço qualitativo médio, então a questão “x c” é recomendada, caso contrário, não será recomendada. Para o caso de o estudante errar a questão “x b”, então a questão “x c” não será recomendada. As questões do sistema podem estar associadas dentro do mesmo grupo de conhecimento como citado nos parágrafos anteriores, mas também podem estar relacionadas a outros grupos de conhecimento. Essa situação está ilustrada na Figura 3.17.

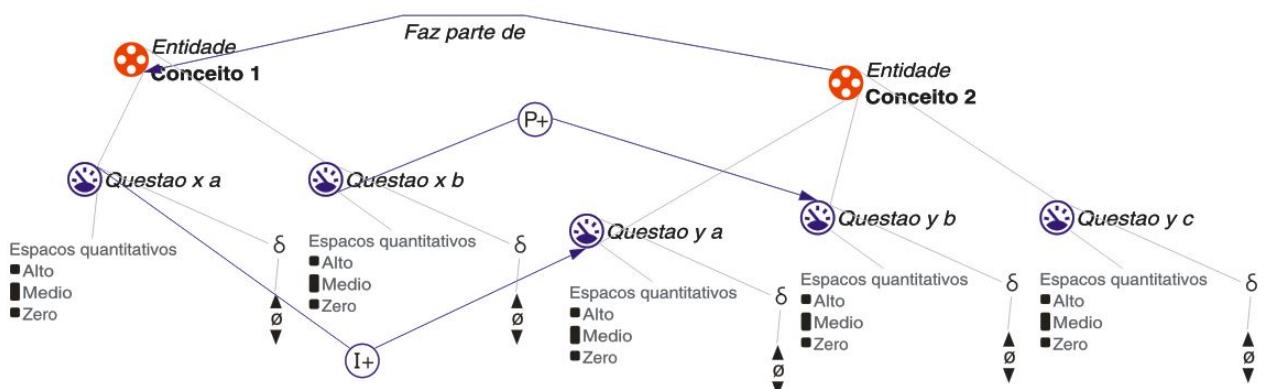


Figura 3.17: Associação entre as questões do Grupo X e do Grupo Y

Na Figura 3.17, a questão “x a” é vista como uma influência direta positiva (I+) sobre

a questão “y a”. Em TQP isso significa que a derivada da questão “y a” será positiva caso a questão “x a” esteja no espaço quantitativo “Alto”. Essa derivada será negativa caso a questão “x a” seja zero. Para a recomendação, isso é um indicativo de que quando o estudante acertou a questão “x a” (espaço quantitativo “alto”), questão “y a” pode ser recomendada. Caso o estudante acerte medianamente a questão “x a”, o resultado da recomendação será indicado pela média qualitativa das simulações que calcularão os estados do estudante. Se a média qualitativa obtida for maior que o valor médio, então a questão “y a” é recomendada, caso contrário, não será recomendada. Para o caso de o estudante errar a questão “x b”, então a questão “y a” não será recomendada.

Ainda na Figura 3.17, questão “x b” possui uma proporcionalidade positiva (P+) sobre a questão “y b”. Isso significa que a derivada da questão “y b” se comporta como a derivada da questão “x b”. Em termos de recomendação, isso significa que quando o estudante acerta a questão “x b”, ou tem uma tendência calculada para acertar questão “x b”, a questão “y b” pode ser recomendada. Caso o estudante acerte medianamente a questão “x b”, o resultado da recomendação será indicado pela média qualitativa das simulações que calcularão os estados do estudante. Se a média qualitativa obtida for maior que o valor médio, então a questão “y b” é recomendada, caso contrário, não será recomendada. Para o caso de o estudante errar a questão “x b”, então a questão “y b” não será recomendada se a derivada da questão “x b” for negativa. Pode acontecer um caso onde a derivada da questão “x b” é positiva, mas seu espaço quantitativo estar em zero. Nesse caso, a questão “y b” ainda será recomendada pelo sistema, visto que para os casos de proporcionalidade (P) o comportamento da quantidade influenciada é baseado na derivada do valor influenciador e não no seu valor real.

A abordagem citada anteriormente foi representada pois existem casos nos quais o estudante erra uma questão, mas no contexto geral ele está progredindo. O fato de ele estar obtendo progresso no caso geral indica uma derivada positiva naquela parte do sistema. Dessa forma, é possível recomendar uma questão baseada no contexto geral do estudante. Quando usamos apenas as influências diretas (I) não é possível propagar os efeitos da derivada geral do sistema, visto que para a TQP a influência é apenas pontual.

Para o sistema de recomendação, é importante que o modelador do sistema identifique onde deve usar influências ou proporcionalidades. Essas escolhas afetarão diretamente as simulações e as recomendações das questões.

Para o modelo de recomendação de questões foram geradas algumas simulações, como modo de validar a proposta. Estas simulações são descritas no próximo tópico.

3.2.4.2 Análise dos cenários pelo PMA-TQP para exercícios

Cenário Simples: Avaliando a transição de estados em questão única

Cenário 1: Perspectiva positivista

Um cenário no qual o estudante inicia o contato com o conteúdo a ser estudado, indica que ele está no espaço quantitativo zero. Ao responder corretamente a primeira questão, o modelo posiciona o estudante no valor alto do espaço quantitativo dos acertos e no valor zero do espaço quantitativo dos erros. Isto é considerado um cenário positivista. A intenção da simulação desse cenário sob a perspectiva positivista é validar a situação de influência positiva vinda do acerto em relação a questão “x a”. O cenário em questão está representado na Figura 3.18.

O quadro (Figura 3.18 a) ilustra a configuração do cenário, no qual está representado os valores iniciais das quantidades a serem observadas. Nessa perspectiva, a quantidade de acertos é alta, a quantidade de erros é zero e a quantidade inicial da questão “x a” é zero também. Nesse cenário, a quantidade de acertos e erros não variam, elas obedecem apenas a marcação inicial e se comportarão assim em todos os estados. Essa abordagem foi escolhida porque nesse cenário deseja-se apenas observar o comportamento da questão “x a” sob essa perspectiva.

Quando essa perspectiva é executada, são gerados os três estados ilustrados pelo grafo do quadro (Figura 3.18 b) e detalhados no quadro C. Observando o detalhamento do quadro (Figura 3.18 c), podemos observar que a questão “x a” tem valor zero no estado

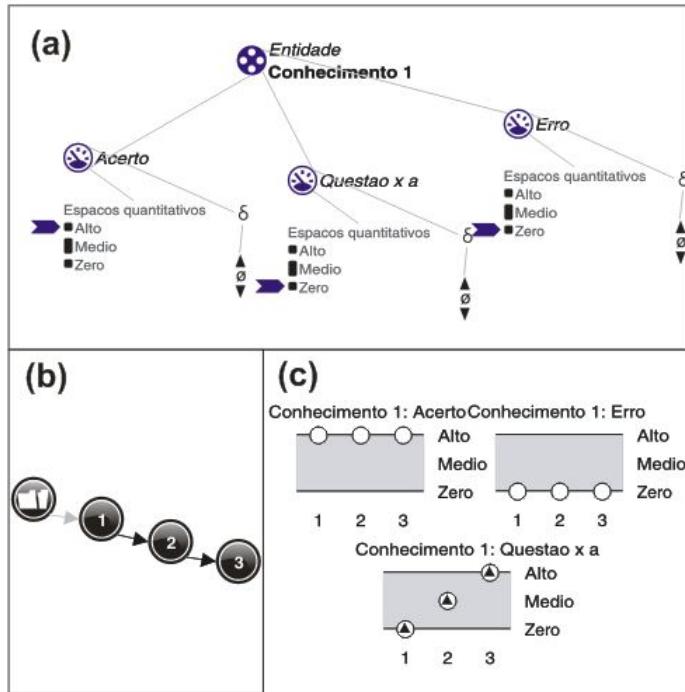


Figura 3.18: Cenário positivista para transição de estados em questão única

1, valor médio no estado 2 e valor alto no estado 3. Isso é possível porque a questão “x a” sofre influência ($I+$) da quantidade de acertos. Ou seja, como o valor de acerto é alto, questão “x a” mudará seus espaços quantitativos na direção do valor alto. Isso acontece porque a TQP infere que a derivada da questão “x a” será positiva, pois o acerto é alto. Do outro lado, a questão “x a” sofre influência negativa dos erros. Nessa situação, a quantidade de erro é sempre zero, o que indica que para a questão “x a” que a sua derivada será positiva, visto que a influência negativa ($I-$) entre erro e a questão “x a” denota que a derivada da questão “x a” crescerá na direção oposta do valor de erro.

Cenário 2: Perspectiva negativista

Um cenário no qual o estudante conhece o conteúdo a ser estudado (está no espaço quantitativo alto), mas possui um nível baixo de acertos e um nível alto de erros é considerado um cenário negativista. A intenção da simulação desse cenário sob a perspectiva negativista é validar a situação de influência negativa vinda do erro em relação a questão “x a”. O cenário está representado na Figura 3.19.

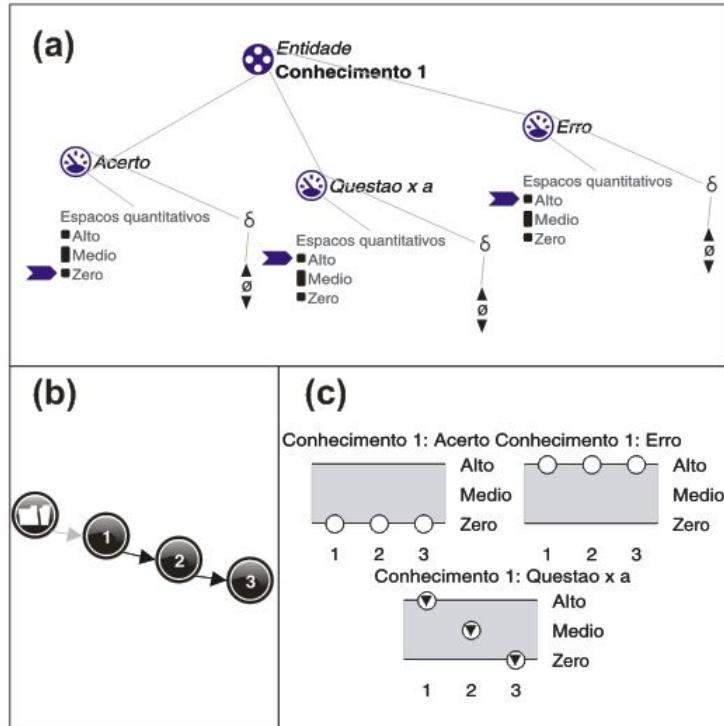


Figura 3.19: Cenário negativista para transição de estados em questão única

O quadro (Figura 3.19 a) ilustra a configuração do cenário, onde está representado os valores iniciais das quantidades a serem observadas. Nessa perspectiva. A quantidade de erros é alta, a quantidade de acertos é zero e a quantidade inicial da questão “x a” é alta. Nesse cenário, a quantidade de acertos e erros não variam, elas obedecem apenas a marcação inicial e se comportarão assim em todos os estados. Essa abordagem foi escolhida porque nesse cenário deseja-se apenas observar o comportamento da questão “x a” sob essa perspectiva.

Quando essa perspectiva é executada, são gerados os três estados ilustrados pelo grafo do quadro (Figura 3.19 b) e detalhados no quadro (Figura 3.19 c). Observando o detalhamento do quadro (Figura 3.19 c), podemos observar que a questão “x a” tem valor alto no estado 1, valor médio no estado 2 e valor zero no estado 3. Isso é possível porque a questão “x a” sofre influência (I-) da quantidade de erros. Ou seja, como o valor de erros é alto, questão “x a” mudará seus espaços quantitativos na direção oposta do valor alto, que é o valor zero. Isso acontece porque a TQP infere que a derivada da questão “x a” será negativa, pois erro é alto. Por outro lado, a questão “x a” sofre influência positiva

dos acertos. Nessa situação, a quantidade de acertos é sempre zero, o que indica que para a questão “x a” que a sua derivada será negativa também, visto que a influência positiva (I_+) entre acerto e a questão “x a” denota que a derivada da questão “x a” crescerá na direção mesma direção do acerto, que nesse momento é zero.

Cenário 3: Perspectiva intermediária

Um cenário no qual o estudante não conhece o conteúdo a ser estudado (está no espaço quantitativo zero), mas possui um nível médio de acertos e um nível médio de erros é considerado um cenário intermediário. A intenção da simulação desse cenário sob a perspectiva intermediária é validar as influências de erros e acertos médios em relação a questão “x a”. O cenário está representado na Figura 3.20.

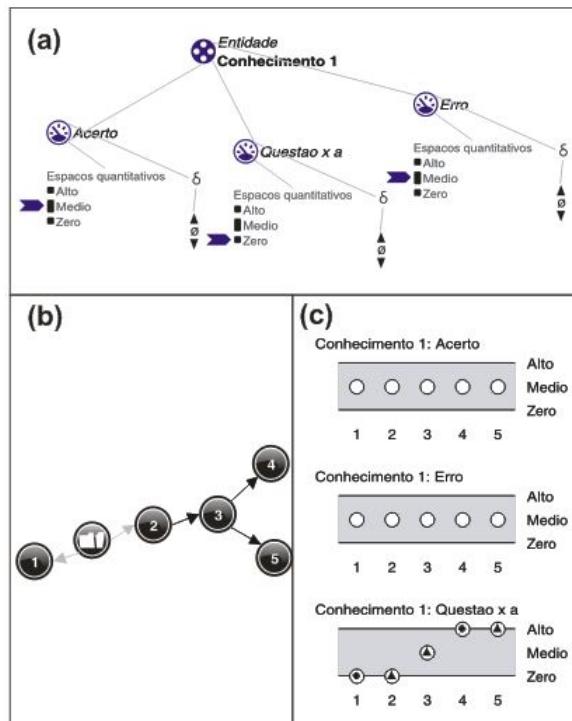


Figura 3.20: Cenário intermediário para transição de estados em questão única

O quadro (Figura 3.20 a) ilustra a configuração do cenário, no qual está representado os valores iniciais das quantidades a serem observadas. Nessa perspectiva. A quantidade de erros é média, a quantidade de acertos é média e a quantidade inicial da questão “x a” é zero. Nesse cenário, a quantidade de acertos e erros não variam, elas obedecem

apenas a marcação inicial e se comportarão assim em todos os estados. Essa abordagem foi escolhida porque nesse cenário deseja-se apenas observar o comportamento da questão “x a” sob essa perspectiva.

Quando essa perspectiva é executada, são gerados os cinco estados ilustrados pelo grafo do quadro (Figura 3.20 b) e detalhados no quadro (Figura 3.20 c). Observando o detalhamento do quadro C, podemos observar que a questão “x a” tem valor zero nos estados 1, 2, valor médio no estado 3 e valor alto nos estados 4,5. Isso é possível porque a questão “x a” sofre influência (I-) da quantidade de erros e influência (I+) da quantidade de acertos. Como nesse cenário está colocado que erros e acertos são medianos, os espaços quantitativos da questão “x a” são movidos para as três direções possíveis: zero, médio e alto. Isso acontece porque a derivada da questão “x a” deve se comportar de acordo com o valor de erros e acertos. Sendo o valor de erros e acertos médios, a derivada da questão “x a” ora será positiva, ora será negativa, e dará origem a quantidade de estados encontrados pela simulação.

Cenário Avançado: Avaliando a transição de estados em duas questões de um mesmo subdomínio de conhecimento

Este cenário ilustra uma situação possível na qual existem três questões relacionadas. Neste exemplo, as três questões pertencem a um mesmo subdomínio de conhecimento. As três questões desse exemplo são conhecidas como “x a”, “x b” e “x c” e o subdomínio recebeu o nome de Grupo X. Neste subdomínio cada questão é filha de uma entidade de conhecimento do Grupo X e, além disso, possui associada a si uma quantidade de erro e de acerto. Essa configuração pode ser vista na Figura 3.21 a seguir.

Na Figura 3.21 está ilustrada uma situação considerada um cenário positivista. Nesta perspectiva, todas as questões começam em nível de conhecimento zero, o nível de acerto aponta para o valor alto em todas elas e o nível de erros está apontando para o nível baixo também em todas as questões. Ao executar esse cenário, obtém-se o resultado ilustrado nas Figuras 3.22 e 3.23 a seguir.

O grafo da Figura 3.22 ilustra a existência de nove estados possíveis para o cenário.

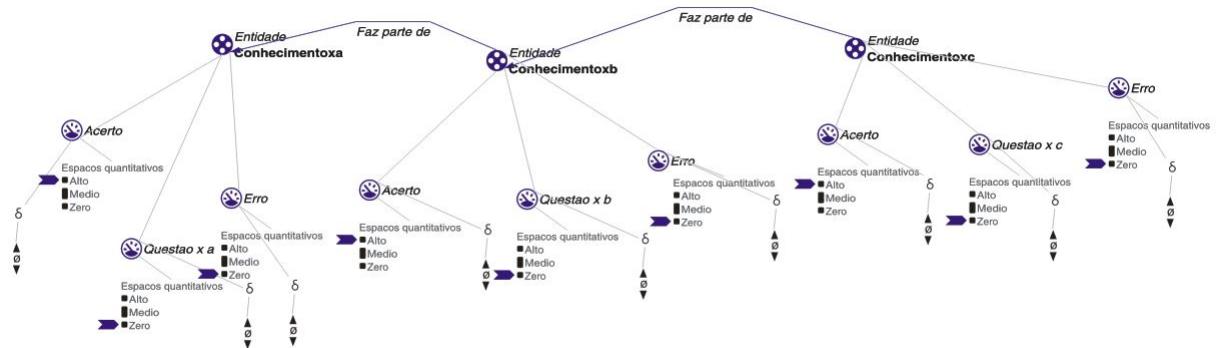


Figura 3.21: Cenário positivista para transição de estados com três questões

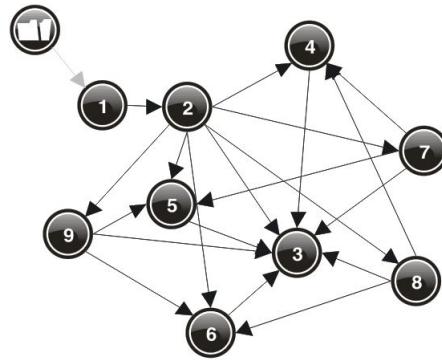


Figura 3.22: Grafo positivista para transição de estados com três questões

A Figura 3.23 expõe os valores dos estados.

Pode-se observar na Figura 3.23 que a configuração positivista faz com que as questões tenham comportamento de deslocar-se do estado qualitativo zero e subir até o estado qualitativo alto. Essa é uma influência causada pela quantidade de acertos. Como a quantidade de acertos é alta, a derivada das questões fará com que elas se desloquem no sentido do valor alto também. Esse cenário basicamente comprova que o comportamento das questões está ligado ao comportamento positivo do estudante.

Em uma perspectiva negativista, na qual o estudante está acertando as questões mas começa a ter um comportamento negativo³, o grafo deste cenário apontará outros estados, onde a tendência do estudante é começar a errar as questões. Esse comportamento pode ser verificado nas Figuras 3.24 e 3.25 a seguir.

³Comportamento no qual o estudante erra o exercício totalmente. Neste modelo, isso significa que o estudante está no espaço quantitativo zero.

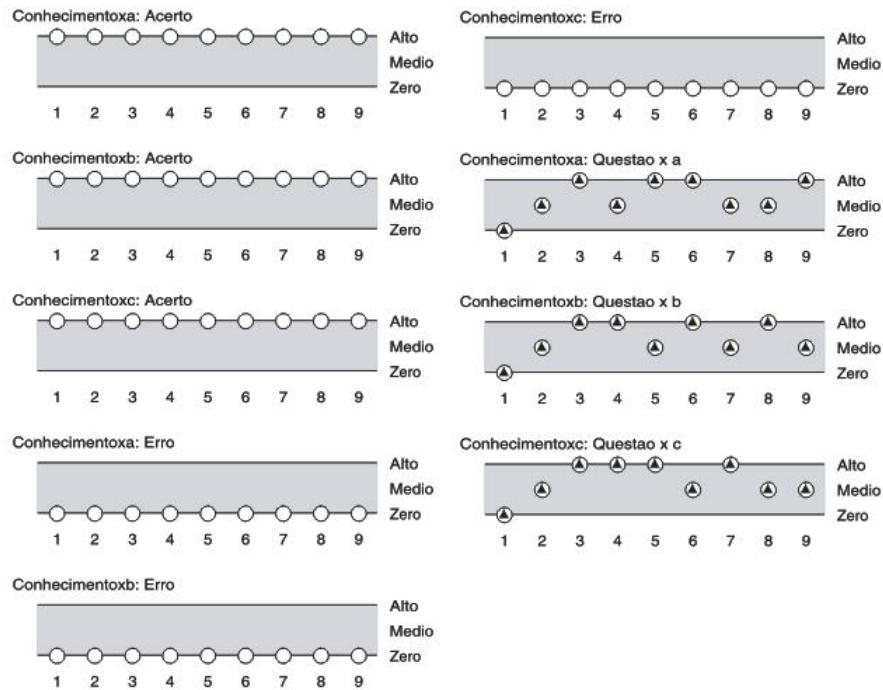


Figura 3.23: Gráfico positivista para transição de estados com três questões

O comportamento do cenário negativista exposto na 3.25 indica que se a quantidade de erros se mantém alta, então a tendência é a de que o estudante comece a errar as questões.

Estes dois cenários apresentados não indicam por si só uma melhor questão a ser apresentada. O que eles fazem é demonstrar o comportamento de erros e acertos de um estudante. No entanto, esse comportamento é considerado relevante para que a relação de indicação seja compreendida, visto que, se o estudante erra a questão “x a”, mas acerta a questão “x b”, a modelagem indica para o sistema que a questão a ser apresentada é “x c”. Esse fato acontece porque a questão “x b”, na modelagem proposta e ilustrada na Figura 3.15 , possui uma proporcionalidade positiva para com a questão “x c”. Este fato é comprovado pela simulação do cenário exposto na Figura 3.26.

No cenário da Figura 3.26, a questão “x a” está com uma tendência de erro e a questão “x b” está com uma tendência de acerto. Nessa situação, como a questão “x b” possui uma proporcionalidade positiva (P+) para com a questão “x c”, a tendência inicial de “x b” foi copiada para o estado inicial de “x c”. A questão “x c” por sua vez, apresentou

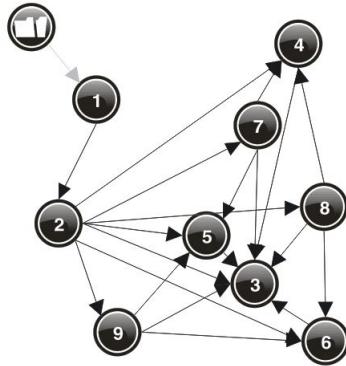


Figura 3.24: Grafo negativista para transição de estados com três questões

comportamento semelhante ao da questão “x b”, conforme pode ser observado no gráfico exposto na Figura 3.28, que é um detalhamento do grafo exposto na figura 3.27. Ambas as imagens são resultado da simulação do cenário exposto na Figura 3.26.

O gráfico exposto na Figura 3.28 mostra que a questão “x a” teve um comportamento de tendência à valoração zero. A questão “x b” obteve um comportamento com tendência a atingir os valores de conhecimento alto e médio, assim como a questão “x c”, que teve seu comportamento baseado na derivada da questão “x b”. Esse gráfico confirma para o sistema de recomendação, que a questão “x c” é uma boa candidata a ser exibida para o estudante como um próximo exercício.

Cenário Avançado: Avaliando a transição de estados entre duas questões de subdomínios de conhecimentos diferentes

Este cenário ilustra uma situação possível em que existem cinco questões relacionadas. Neste exemplo, duas questões pertencem ao subdomínio de conhecimento 1 e três outras questões pertencem ao subdomínio 2. As duas questões do Grupo 1 são conhecidas como “x a” e “x b”. As três questões do Grupo 2 são conhecidas como “y a”, “y b” e “y c”. Os dois subdomínios estão associados a quantidades de erros e acertos. Nessa abordagem, não foi acrescentado um erro ou acerto individual para cada questão porque adotou-se que para esse cenário, a quantidade de erros e acertos será a mesma para todos os elementos do subdomínio. A configuração do fragmento de modelo correspondente pode ser vista na

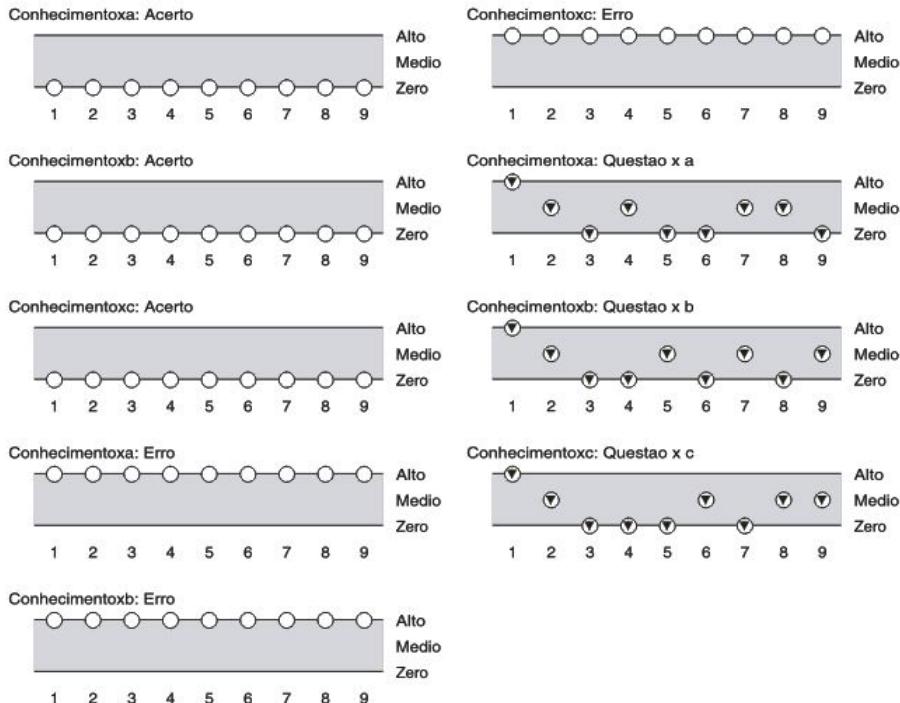


Figura 3.25: Gráfico negativista para transição de estados com três questões

Figura 3.17, onde a questão “x a” é uma influência positiva (I+) para a questão “y a” e a questão “x b” é uma proporcionalidade positiva (P+) para a questão “y b”. A Figura 3.29 exibe uma perspectiva positivista criada para a avaliação de transição de estados entre questões de subdomínios diferentes.

Na Figura 3.29 pode-se visualizar que as questões do domínio de conhecimento 1 estão associadas a um alto nível de acertos e a um baixo nível de erros. As questões “x a” e “x b” começam com espaço quantitativo igual a zero. Do outro lado da imagem está o grupo 2, no qual os acertos são altos e os erros baixos também. Aqui é importante lembrar que as questões do grupo 2 não foram associadas a erros e acertos nos fragmentos de modelo. Isso significa que o comportamento dessas questões dependem de uma derivada colocada manualmente no sistema ou de uma influência ou proporcionalidade vinda de outra quantidade (conhecimento). Essa situação foi proposital, para demonstrar como uma quantidade influencia outra quantidade em um domínio diferente. O resultado desse cenário está ilustrado nas Figuras 3.30 e 3.31.

A Figura 3.30 ilustra o grafo de estados com 18 nós obtidos na simulação gerada pelo

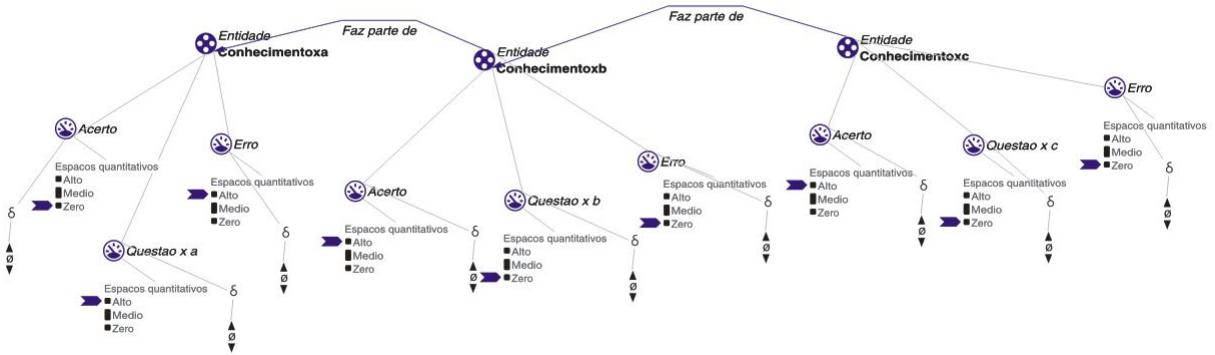


Figura 3.26: Cenário de recomendação para transição de estados com três questões

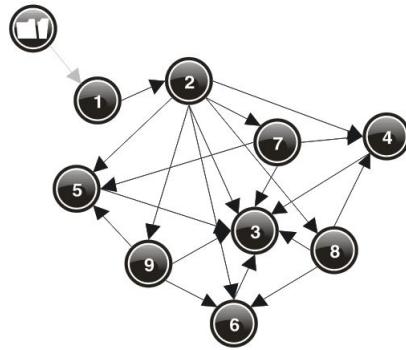


Figura 3.27: Grafo de recomendação para transição de estados com três questões

GARP. Todos os estados do grafo são representados no gráfico da Figura 3.31. Como a perspectiva é positiva para ambos os lados (subdomínios 1 e 2), a tendência mostra que o estudante nesse estado possui alta probabilidade de acertar as questões dos dois domínios. Caso a situação fosse negativista, ou seja, o estudante estivesse em um nível alto de conhecimento em ambos os conceitos, mas começasse a errar, sua tendência para as questões desses dois domínios é a de diminuir seu conhecimento até o nível zero. Essa situação é demonstrada no gráfico ilustrado na Figura 3.32.

Um fato interessante a ser observado no gráfico da Figura 3.32, que também pode ser notado na 3.30, está relacionado à questão “y c”. Nota-se que ela não muda de espaço quantitativo, mas simplesmente mantém a configuração inicial do cenário, seja ela qual for. Esse fato acontece porque a questão “y c” não sofre influência ou proporcionalidade de ninguém e também não está associada a acertos ou a erros. Essa situação foi demons-

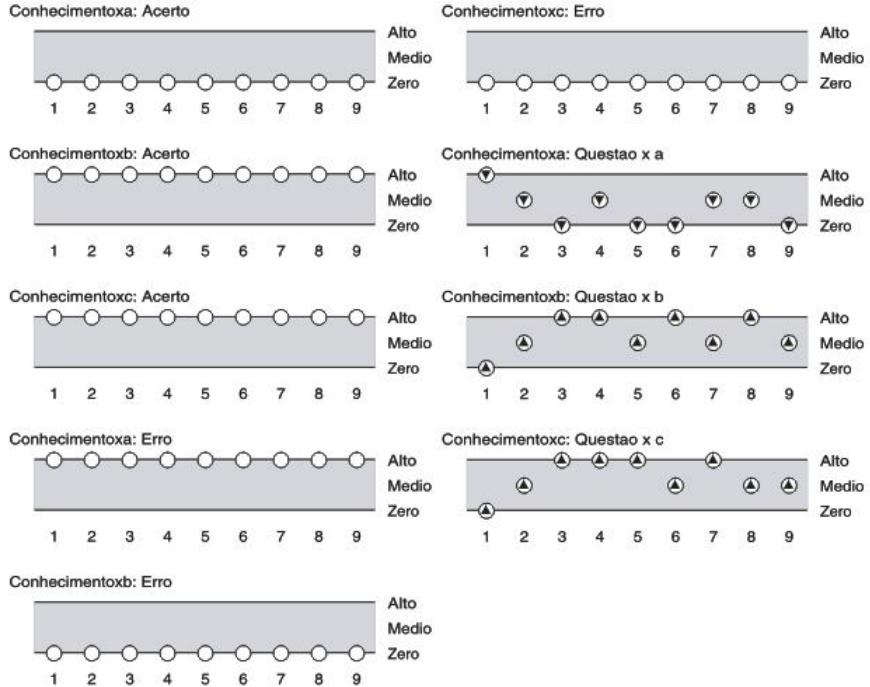


Figura 3.28: Gráfico de recomendação para transição de estados com três questões

trada nesse cenário para apresentar a importância das influências e proporcionalidades nas relações entre as quantidades (conhecimentos) dos subdomínios e domínios modelados. Quando essas relações não são representadas, não há como o sistema inferir algo sobre elas.

Nesta mesma perspectiva ainda, nota-se que a questão “y a” se comporta de forma linear também. Ela é uma questão que sofre influência positiva da questão “x a”, que está decrescendo seus valores até zero. No entanto, a questão “x a” move-se de alto até zero, mas a questão “y a” não. Esse fato aconteceu porque a influência negativa de erro sobre a questão “x a” não foi propagada para outros elementos, visto que a relação entre questão “x a” e questão “y a” também é uma influência (I+). Essa mudança seria propagada apenas se entre questão “x a” e questão “y a” ouvesse uma proporcionalidade. Esse fato é comprovado pelos valores das questões “x b” e “y b”. Na Figura 3.32 pode-se observar que a questão “x b” também sai do espaço quantitativo alto em direção ao estado quantivo zero. No entanto, a questão “y b” que possui uma proporcionalidade (P+) vinda de “x b” não mantém seus valores de forma linear apesar de não estar ligada a outros

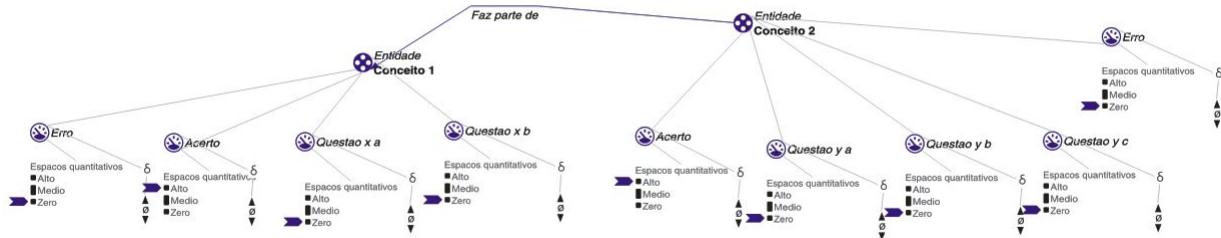


Figura 3.29: Cenário positivista para transição de estados com cinco questões

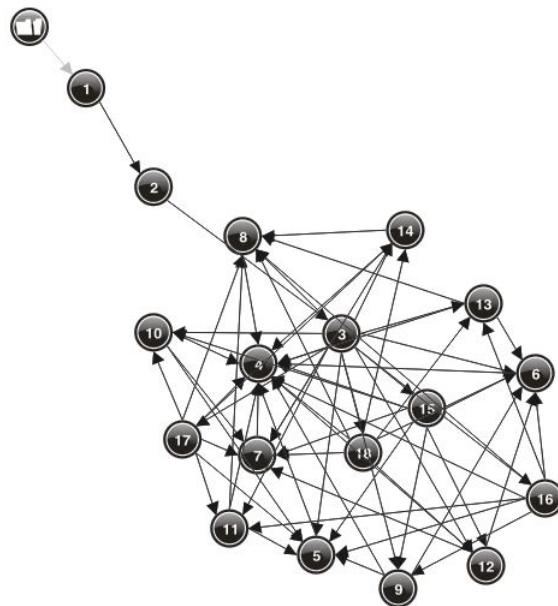


Figura 3.30: Grafo positivista para transição de estados com cinco questões

acertos e erros, mas sim, acompanha os valores de “x b”. Esse fato acontece porque a proporcionalidade entre “x b” e “y b” garante que os efeitos de “x b” serão propagados para “y b”. A execução desse cenário indica que a melhor questão a ser apresentada para o estudante, dado que o conceito 1 já foi aprendido, é a questão “y a”. A questão “y c” neste caso não seria considerada porque ela não está relacionada com as questões resolvidas anteriormente pelo estudante e apenas seria apresentada caso todas as outras questões possíveis já tivessem sido apresentadas. Ou seja, nessa simulação, a questão “y c” tem menor prioridade para ser escolhida dentre as demais “y a” e “y b”.

Este mesmo cenário pode ser visto sob uma perspectiva onde o estudante errou a

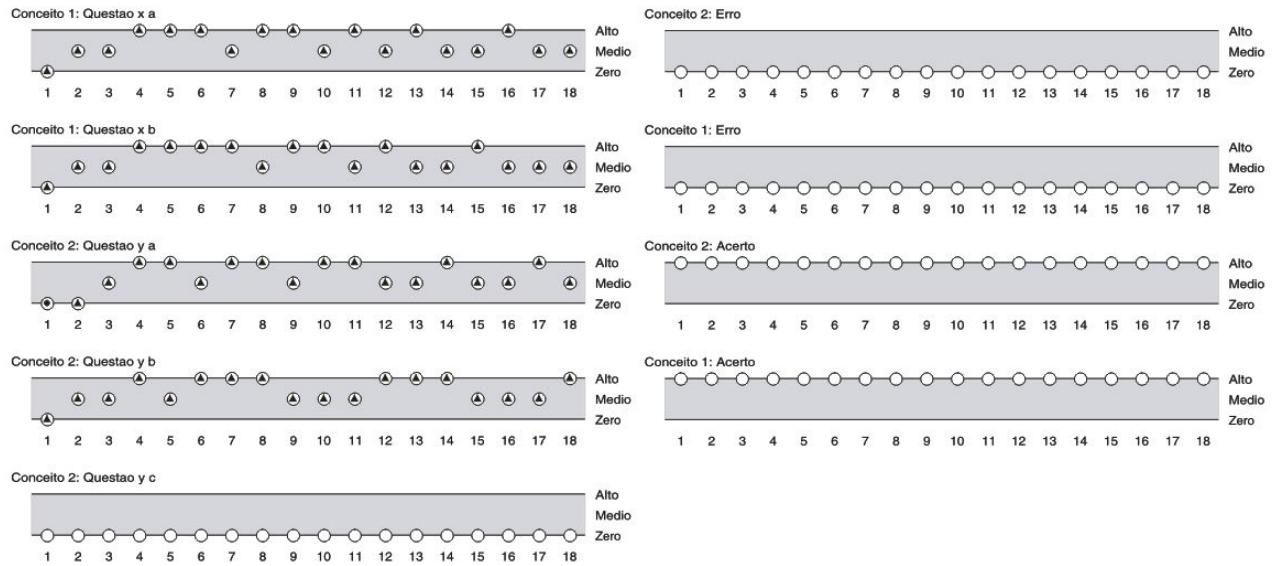


Figura 3.31: Gráfico positivista para transição de estados com cinco questões

questão “x a”, mas acertou “x b”. A Figura 3.33 ilustra essa perspectiva.

A execução do cenário apresentado na Figura 3.33 no GARP, resulta no grafo ilustrado na Figura 3.34 e no gráfico ilustrado na Figura 3.35.

A Figura 3.35 ilustra detalhadamente os estados obtidos e ilustrados na Figura 3.34. Essa simulação demonstra que as questões “x a” e “y a” estão no estado quantitativo zero e não alteram essa posição. Esse resultado acontece porque a quantidade de erros para esses conceitos é alta e a quantidade de acertos é zero. A questão “x b”, no entanto, tem um estado inicial no espaço quantitativo alto, o que indica acerto por parte do estudante. Como a inclinação do estudante é de cometer erros neste cenário, “x b” irá sair do estado qualitativo alto em direção ao estado qualitativo zero. A questão “y b” possui uma proporcionalidade positiva (P+) vinda de “x b”, e isso indica que seu comportamento será na mesma direção de “x b”. A simulação desse cenário indica que para essa situação, a melhor questão a ser proposta para o estudante é “y b”, porque a tendência dela é de um comportamento semelhante ao de “x b”, que não foi sempre bom, mas mesmo assim, obteve uma média maior quando comparado com as questões “x a” e “y a”.

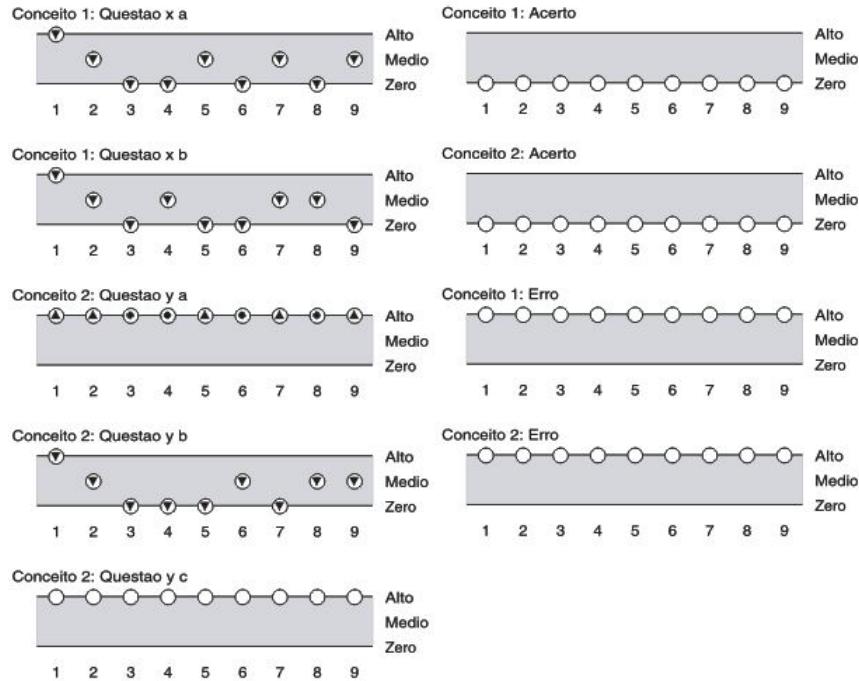


Figura 3.32: Gráfico negativista para transição de estados com cinco questões

3.2.5 Considerações finais

Cenários maiores podem ser gerados a partir do acréscimo de quantidades. Exemplos maiores que os citados não serão exemplificados neste trabalho, visto que, eles crescem exponencialmente, o que torna suas reproduções e acompanhamentos extensos e que podem não acrescentar informações relevantes. Ainda sobre as simulações realizadas, é importante ressaltar que todas respeitam o uso das influências diretas e proporcionalidades para indicar os estados em que as quantidades (conhecimentos) podem transitar.

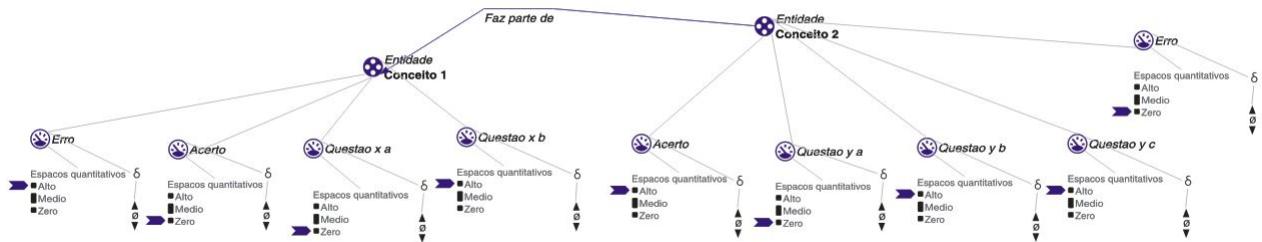


Figura 3.33: Cenário de indicação para transição de estados com cinco questões

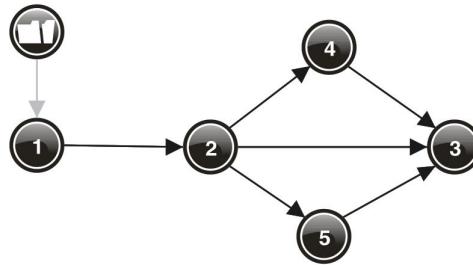


Figura 3.34: Grafo de indicação para transição de estados com cinco questões

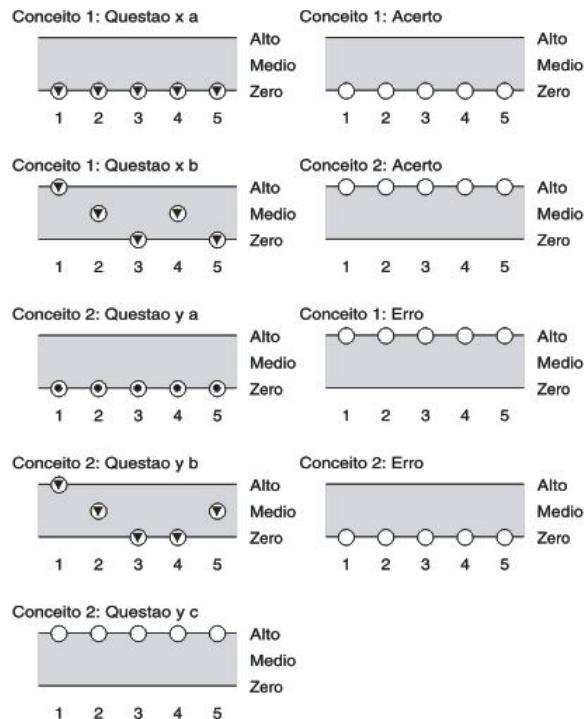


Figura 3.35: Gráfico de indicação para transição de estados com cinco questões

CAPÍTULO 4

ESTUDO DE CASO

4.1 Introdução

A TQP (Teoria do Raciocínio Qualitativo) geralmente é usada para modelar processos físicos, tais como o processo da fotossíntese e o comportamento de um ecossistema, por exemplo. Neste estudo de caso, no entanto, usou-se a TQP para modelar o processo de aprendizagem em um sistema tutor inteligente.

A simulação dos estados de um estudante é possível através da execução de cenários nos quais são refletidos os modelos baseados em TQP para um domínio dos conteúdos. Para modelar uma disciplina ou um conjunto de conhecimentos usando a TQP é necessário seguir os mesmos passos citados no Capítulo 3.

Este capítulo apresenta um estudo de caso em que foi aplicada a proposta de utilizar o raciocínio qualitativo na construção de um STI (Sistema Tutor Inteligente). Neste estudo de caso, a disciplina de orientação a objetos foi modelada segundo as especificações da teoria de raciocínio qualitativo de processos. O trabalho exposto neste capítulo apresenta as simulações geradas no software GARP (*General Architecture for Reasoning in Physics*) para a validação do módulo PMA-TQP, apresentado no Capítulo 4. Os próximos capítulos apresentam a aplicação da TQP à modelagem do processo de ensino para a disciplina de orientação a objetos.

4.2 A disciplina de orientação a objetos

Os conceitos de objetos começaram a ser utilizados na construção de sistemas de software nos anos 60, a partir de uma analogia com a construção de um sistema mecânico onde são utilizados objetos concretos [19].

O paradigma da orientação a objetos (OO) vem se desenvolvendo com a criação de

padrões de projeto. Com seu crescente uso os padrões de projeto vêm se afirmando como uma forma apropriada de organização de sistemas. Muitas linguagens antecederam a linguagem Java¹ (uma das mais usadas atualmente) no paradigma da orientação a objetos, sendo a mais conhecida a linguagem denominada SmallTalk². As linguagens orientadas a objeto, apesar de pertencerem ao mesmo paradigma, podem ser divididas em dois grandes grupos: as linguagens baseadas em classes e as linguagens baseadas em objetos [19].

Linguagens baseadas em classes têm como característica o conceito de classe o qual fornece a descrição de objetos gerados por ela. São exemplos de linguagens baseadas em classe o Java, o C++³ e o SmallTalk. Os objetos das linguagens baseadas em classes podem possuir métodos embutidos no próprio objeto, ou podem acessá-los em um conjunto de métodos da classe através de delegação [19].

Experiências acadêmicas apontam problemas que podem ser classificados em três grandes grupos: a) problemas relativos ao processo de ensino-aprendizagem, b) problemas relativos à experiência prévia dos alunos, e c) experiência no paradigma estruturado e problemas em relação às linguagens, ambientes de programação e metodologias OO existentes [61] .

Cabe ressaltar que não existe um consenso nem na identificação dos problemas de ensino e adoção do paradigma, nem nas respostas e conclusões dos diversos autores em relação aos mesmos. Aspectos considerados importantes e benéficos para alguns autores são considerados por outros como sendo prejudiciais no processo como um todo [61] .

Os problemas referentes ao processo de ensino-aprendizagem são aqueles que estão relacionados com dificuldades no entendimento dos conceitos do paradigma OO e na sua aplicação. Como consequência existe a perda das vantagens que o paradigma pode proporcionar em termos de qualidade e produtividade no processo de desenvolvimento de software como um todo. Como efeito, ao passar por este tipo de dificuldades, os indivíduos não conseguem perceber e aproveitar os benefícios do paradigma, não sendo justificada então sua adoção. Em alguns casos mais graves, indivíduos passam até mesmo

¹https://www.java.com/pt_BR/

²<http://www.smalltalk.org/main/>

³<http://www.cplusplus.com/>

a considerar o paradigma OO como inadequado ao desenvolvimento de software. Entre os fatores que causam e agravam problemas de ensino aprendizagem estão [61]:

- Falta de material bibliográfico adequado de apoio;
- Falta de experiência de professores/instrutores;
- Expectativa de retorno imediato por parte dos alunos do investimento de aprendizado;
- Influência do paradigma estruturado;
- Falta de metodologias de apoio ao aprendizado de linguagens e ambientes de programação .

Trabalhos discutem os conceitos a serem apresentados em cursos básicos e avançados. De uma maneira geral eles concordam que conceitos como abstração, classes, métodos, encapsulamento, *information hiding*(ocultamento de informações), herança, redefinição, polimorfismo, devem ser apresentados primeiro por serem básicos e por solidificarem as vantagens associadas ao paradigma. A herança múltipla, que faz parte do paradigma, é considerada desnecessária e até mesmo dispensável por estes autores no contexto de um curso introdutório, que consideram este tipo de herança como um conceito avançado e que deve ser ensinado mais tarde [3, 38].

4.3 O processo de modelagem

O início do processo de modelagem começa ao formalizar os conceitos da disciplina de orientação a objetos e representá-los segundo a TQP. Ao todo, foram formalizados vinte conceitos e a relação entre eles, conforme ilustrado na Figura 4.1.

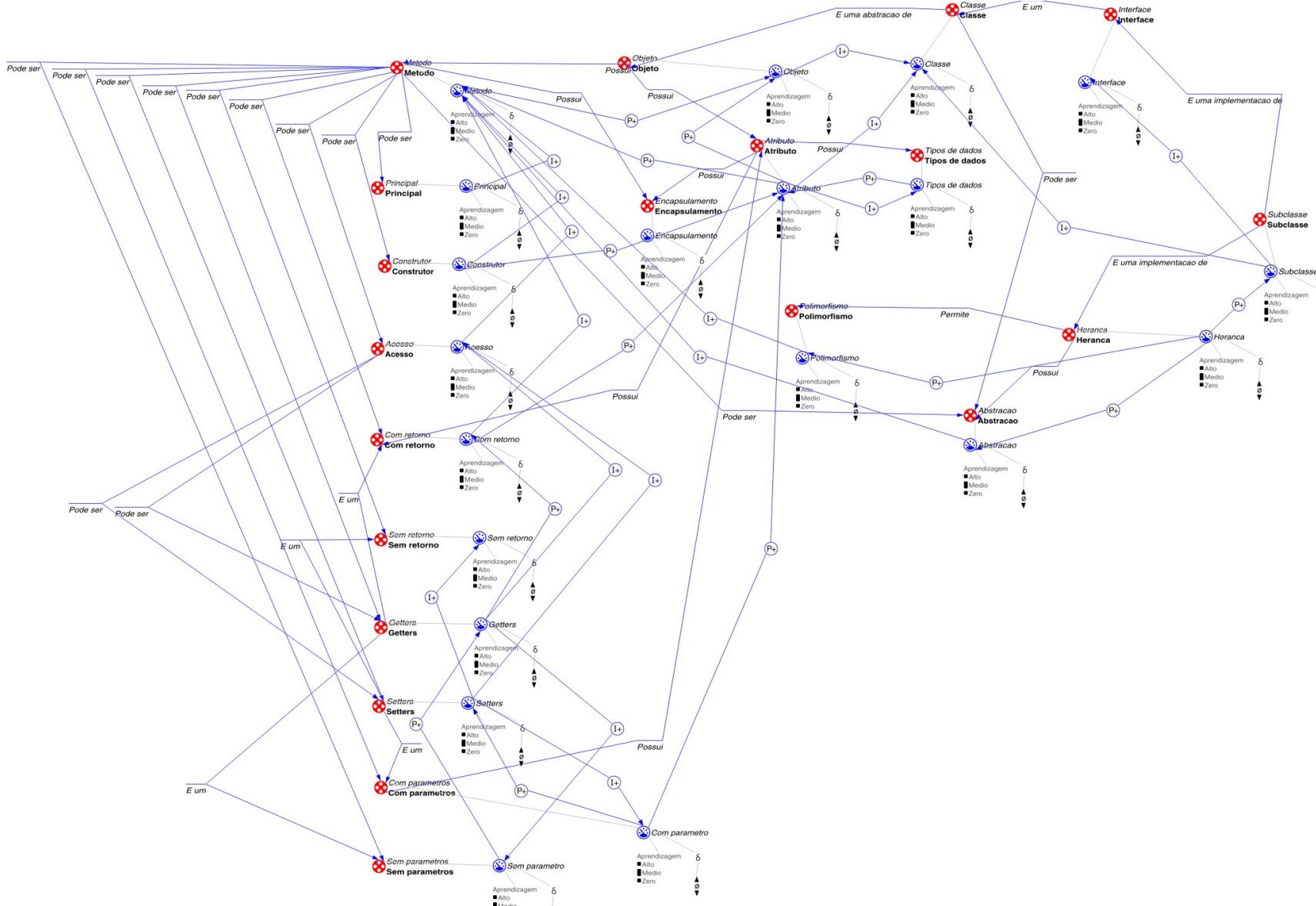


Figura 4.1: Fragmento de modelo contendo as entidades do domínio para orientação a objetos

As entidades possuem elementos denominados quantidades. Estas, por sua vez, estão atreladas a espaços quantitativos chamados de aprendizagem. Os espaços quantitativos podem ser zero, médio ou alto. Essa definição permite representar influências e proporcionalidades positivas para os acertos e influências e proporcionalidades negativas para os erros cometidos pelo estudante.

O fragmento de modelo geral criado para a disciplina de orientação a objetos ilustra os 20 conceitos formalizados em entidades. Cada entidade possui um elemento quantidade associado a si. Esse elemento contém um conjunto de espaços quantitativos possíveis, denominados de aprendizagem nesse modelo. Foi considerado que o aprendizado pode ser zero (quando o estudante não aprendeu), médio (para uma aprendizagem mediana) e alto (para representar que o estudante aprendeu satisfatoriamente o conteúdo).

O processo de modelagem de um domínio de conhecimento começa na criação dos fragmentos de modelo. A Figura 4.2 ilustra uma árvore de fragmentos para esse domínio.

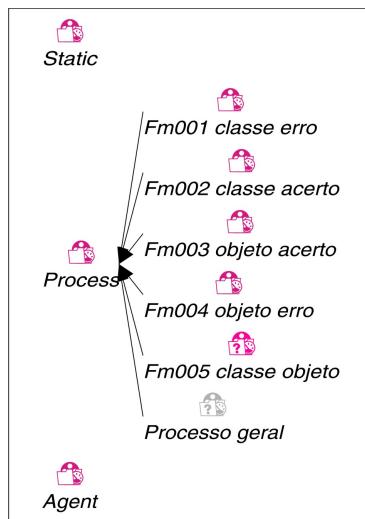


Figura 4.2: Árvore de fragmentos de modelo para as entidades classe e objeto

Cada fragmento de modelo, identificado por um código (*Fm001*, *Fm002...*), representa uma parte das relações causais formadas para o conjunto de conceitos do domínio. A Figura 4.3 ilustra o conteúdo de cada fragmento de modelo para os elementos “classe” e “objeto”, suas relações com os acertos e erros e suas inter-relações.

Na abordagem adotada, todo conceito possui uma quantidade de erro e uma quan-

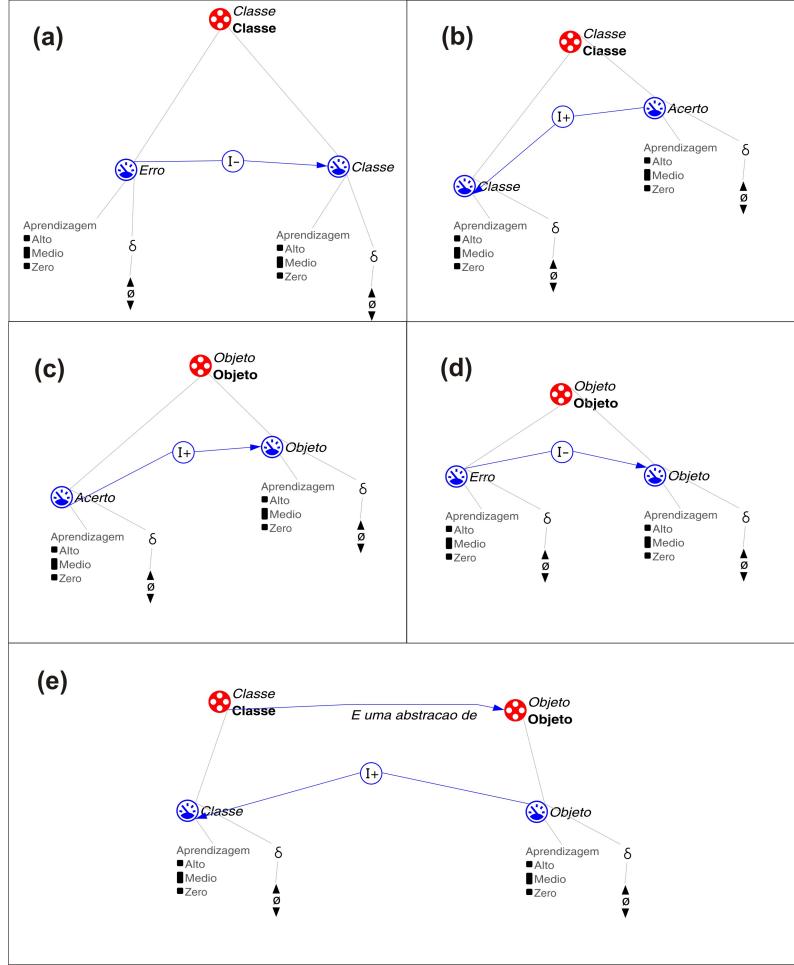


Figura 4.3: Fragmentos de modelo para as entidades classe e objeto

tidade de acerto que são associadas a ele. O objetivo dessa abordagem é indicar nas simulações, através das quantidades de acertos e erros, se um estudante está aprendendo ou não.

A relação causal criada para representar as inter-relações de aprendizagem entre os elementos de domínio e as quantidades de erros e acertos é determinada pelo conjunto:

$$C = \{I+(entidadeDeDominio, acerto), I-(entidadeDeDominio, erro)\}$$

Os acertos foram considerados influências positivas porque a entidade de domínio mudará de espaço quantitativo na mesma direção do valor do acerto. Ou seja, se o acerto é alto, a derivada da entidade de domínio indicará à simulação que esta entidade deverá seguir na direção do espaço quantitativo alto. Isso significa, em termos de aprendizagem, que o estudante está conseguindo assimilar o conteúdo relacionado à entidade de domínio

em foco.

Os erros foram denotados como uma influência negativa (I-) porque, nesse caso, a entidade de domínio mudará no sentido inverso do valor do erro. Ou seja, quando valor do erro é alto, a derivada da entidade de domínio seguirá na direção de zero. Isso significa, em termos de aprendizagem, que o estudante não está aprendendo o conteúdo relacionado à entidade de domínio em foco.

Além do relacionamento com erros e acertos, as entidades de domínio podem estar relacionadas com outras entidades de domínio. Neste caso, a relação geral é denotada por $C = \{R(\text{entidadeM}, \text{entidadeN})\}$, onde R é o relacionamento de influência direta (I) ou proporcionalidade (P), ambas positivas ou negativas. A escolha entre I ou P depende da interpretação que o modelador do domínio tem sobre a relação entre as entidades. Essa escolha deve ser feita de acordo com as definições de I e P definidas na TQP.

4.4 Modelo de domínio com inserção de erros e acertos

Cada elemento criado para o domínio de orientação a objetos pode ser representado isoladamente ou relacionado a outros elementos em um cenário. Quando um elemento de domínio é representado isolado, considerando apenas suas ligações com as quantidades de acertos e erros, as simulações geradas pelo GARP resultam num volume de dados menor e mais fácil de analisar se comparado a modelos mais complexos, que envolvam dois ou mais elementos da orientação a objetos.

Para ilustrar todos os casos possíveis em uma simulação com três elementos, sendo um principal para o domínio e os outros dois correspondentes aos acertos e erros, cada um com três espaços quantitativos possíveis, poderiam ser gerados 27 cenários diferentes. A reprodução de todos estes cenários é muito extensa e não muito relevante, visto que conhecendo os cenários principais é possível delinear todas as possibilidades pelo comportamento demonstrado pelos elementos. Em cenários maiores, com a representação entre dois ou mais elementos, o tamanho de possíveis cenários é maior ainda. Com dois elementos de domínio, classe e objeto, por exemplo, considerando que cada um possui uma quantidade de erros e uma quantidade de acertos, é possível elaborar 729 cenários.

Por este motivo, este trabalho abordará somente os cenários principais, que correspondem a um cenário positivista, negativista e mediano (ou intermediário entre as opções positivista, negativista) para representações com apenas um elemento de domínio. Para cenários com mais de um elemento de domínio, serão expostas algumas situações mais relevantes, como por exemplo, quando o estudante está acertando muito em um conteúdo mas começa a errar em outro conteúdo que está relacionado nos fragmentos de modelo. Com essa representação, por exemplo, é possível verificar as influências diretas ou indiretas agindo sobre o comportamento do estudante.

Neste trabalho, um cenário positivista representa a situação na qual o estudante está num estado de conhecimento zero, mas seu nível de acerto é alto e o nível de erro é baixo. Um cenário negativista expõe a situação em que o estado do estudante é de conhecimento alto, mas seu nível de acerto é baixo e o nível de erro é alto. Um cenário mediano configura o estudante com o estado médio e atribui que seus erros e acertos são medianos também. É possível que um estudante esteja em um estado no qual seu conhecimento é alto e seu nível de acerto também seja alto.

Quando existe apenas um elemento a ser aprendido, e este elemento não possui níveis de dificuldade, essa situação acaba evoluindo para um cenário em que o estudante sempre acerta tudo, de acordo com o raciocínio qualitativo. Isso acontece porque a tendência é que a quantidade de acertos influencie positivamente no aprendizado, como foi modelado no fragmento de modelo exposto na Figura 4.3. O caso de um estudante que está em um estado de conhecimento zero, com acertos em zero e muitos erros, resulta em uma situação em que o estudante evolui para um cenário no qual sempre erra todas as questões. Isso é justificado pela influência negativa que o erro possui sobre os elementos de domínio a serem aprendidos. Essas situações extremas não são consideradas relevantes para serem expostas neste trabalho, uma vez que seus comportamentos serão sempre iguais. O objetivo de executar testes em cenários fornecidos pelo GARP é verificar como as quantidades se alteram ao longo do processo de aprendizagem. Por este motivo, os cenários apresentados neste trabalho são aqueles que realmente demonstram o acontecimento dessas mudanças de estados qualitativos.

A modelagem total da disciplina de orientação a objetos, seguindo o fragmento de modelo geral da Figura 4.1 seria muito extensa. Outra questão é que a inserção de erros e acertos para os conceitos da orientação a objetos implicaria modelar cada entidade com mais duas outras quantidades, tornando o fragmento modelo maior ainda.

No entanto, é interessante que o modelo geral seja exposto, uma vez que para comprovar o estudo de caso, não é necessário demonstrar todos os fragmentos de modelos para os grupos de conceitos da orientação a objetos. A apresentação do modelo geral de fragmentos expõe a formalização geral e total para esse domínio de aprendizagem.

Os próximos capítulos são destinados a apresentar alguns dos modelos conceituais e dos modelos em TQP para o domínio de orientação a objetos.

4.5 Modelos conceituais de domínio

Foram selecionados quatro elementos da orientação a objetos para serem utilizados na modelagem do estudo de caso deste trabalho. Os elementos escolhidos foram: Objeto, Classe, Atributo e Método.

O primeiro modelo conceitual de domínio para os elementos selecionados no estudo de caso é ilustrado pela Figura 4.4. Nesse exemplo, cada conteúdo possui quatro questões associadas a si (A, B, C e D). As questões podem estar relacionadas entre si por uma sinalização de recomendação, o que indica, de acordo com os acertos do estudante, qual outra questão do domínio pode ser recomendada.

O segundo modelo do estudo de caso representa a relação de recomendação de questões em caso de erro. Esse modelo está ilustrado na Figura 4.5.

O modelo da Figura 4.4 deve ser sobreposto ao modelo da Figura 4.5. Dessa maneira, pode-se criar caminhos por onde o estudante passará, seja quando ele acerta ou quando ele erra alguma das questões.

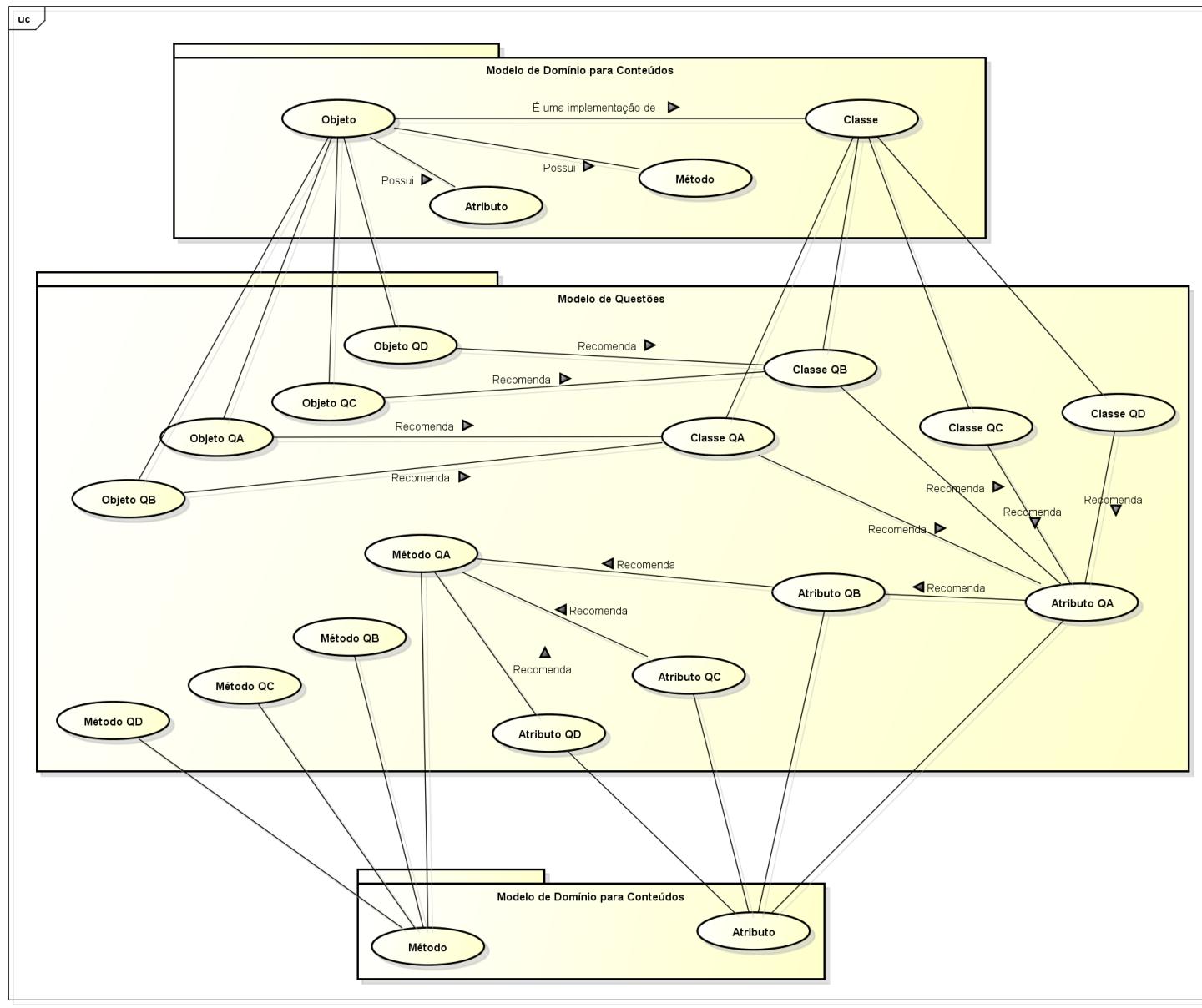


Figura 4.4: Modelo de domínio para conteúdos e questões - Recomendação para acertos

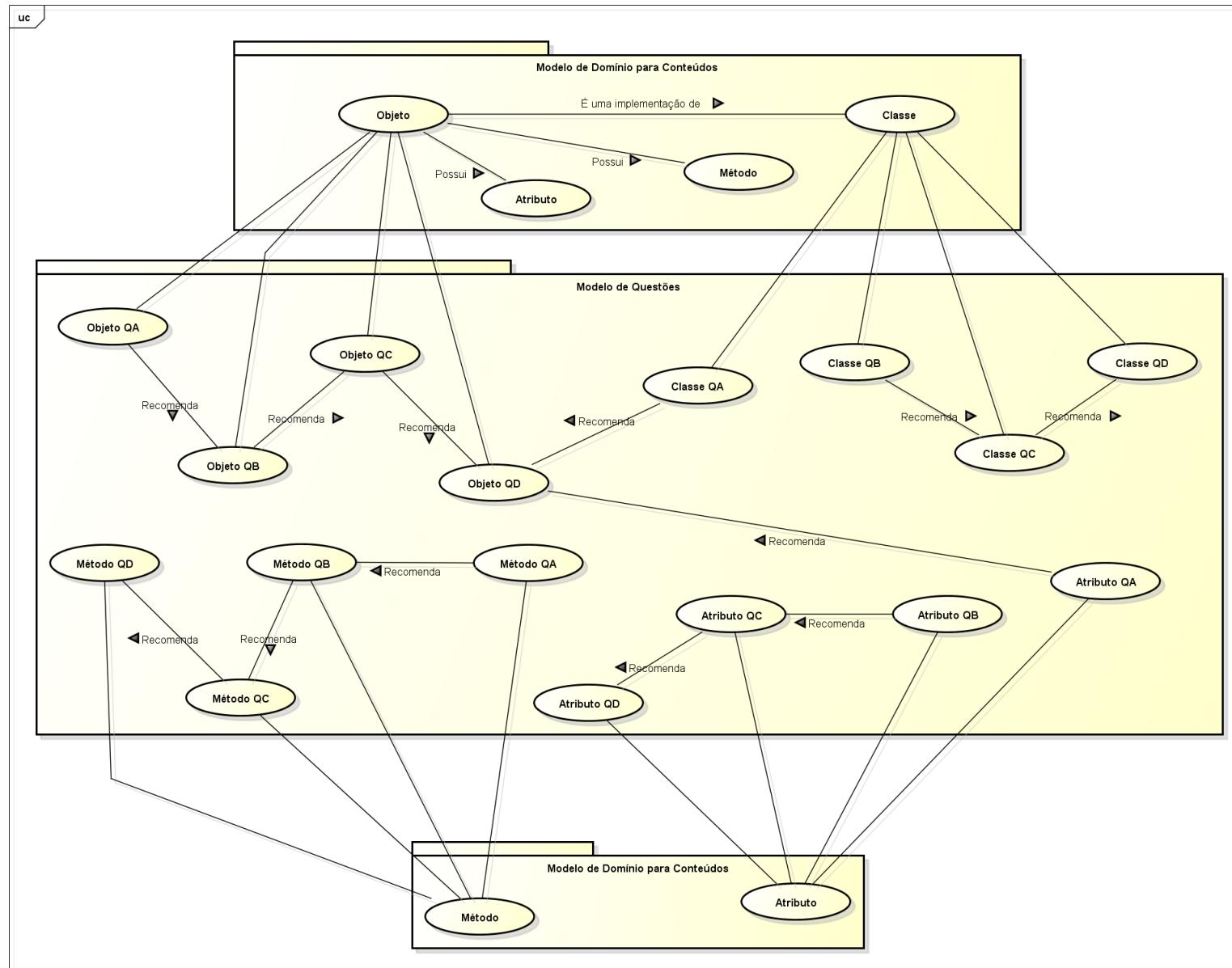


Figura 4.5: Modelo de domínio para conteúdos e questões - Recomendação para erros

Um exemplo de recomendação de questões seria o caso no qual o estudante visualiza o conteúdo sobre o elemento Objeto e responde de maneira correta uma questão do domínio. Este exemplo está ilustrado na Figura 4.6.

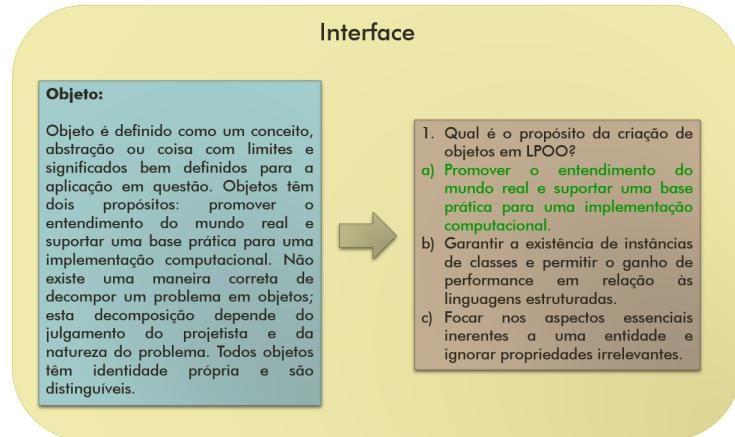


Figura 4.6: Exemplo de questão para o domínio Objeto

Neste caso, o estudante será direcionado para o conteúdo do elemento Classe, onde ele visualiza o conteúdo e em seguida responde a uma questão. Este exemplo está ilustrado na Figura 4.7.

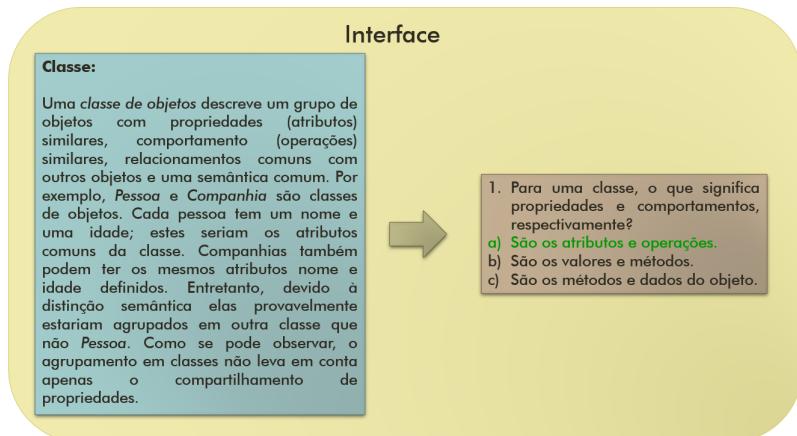


Figura 4.7: Exemplo de questão para o domínio Classe

A situação ilustrada nos exemplos é análoga ao modelo conceitual exposto na Figura 4.4 do estudo de caso.

Os próximos subcapítulos apresentarão os modelos em TQP para alguns elementos do domínio selecionado no estudo de caso.

4.5.1 Modelo de domínio para classe

Neste estudo de caso, cada conceito de domínio está relacionado com uma quantidade de erros e acertos. A Figura 4.8 ilustra como os as quantidades de acerto e erros estão relacionadas ao elemento de domínio classe.

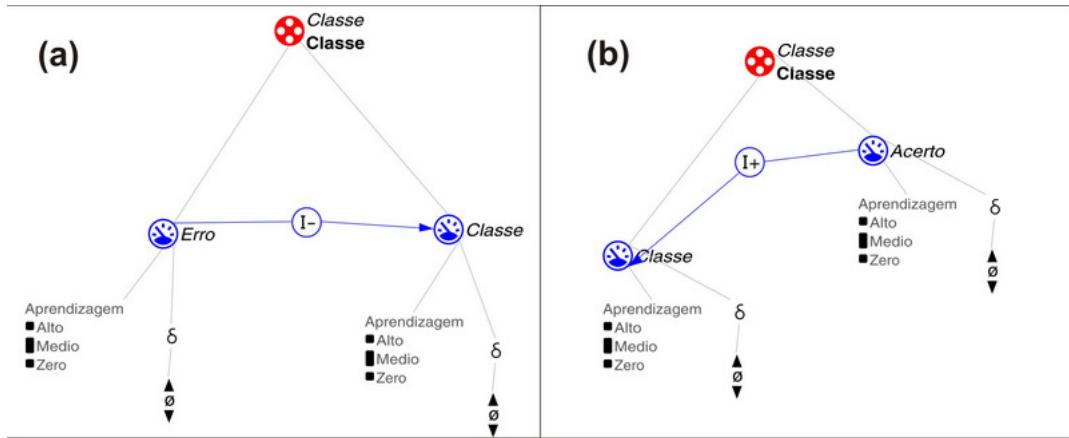


Figura 4.8: Fragmento de modelo para classe com acerto e erro

De acordo com a Figura 4.8, acertos fazem uma influência positiva sobre classe e erros exercem uma influência negativa sobre a mesma. A modelagem do conceito de classe associado com quantidades de acerto e erro pode ser vista sob diversas perspectivas. Para este domínio de conhecimento, foram analisadas três perspectivas: positivista, negativista e mediana, que estão representadas a seguir.

Perspectiva positivista

Essa perspectiva representa um modelo de domínio para classe, no qual o estudante tem um nível de conhecimento baixo sobre classe, um nível alto de acertos e um nível baixo de erros. A simulação dessa perspectiva de cenário está ilustrada na Figura 4.9.

O quadro (Figura 4.9 a) ilustra a configuração do cenário. O quadro (Figura 4.9 b) ilustra o grafo de possíveis estados para a simulação. O quadro (Figura 4.9 c) ilustra os estados em formato de gráfico. A quantidade de acerto foi marcada com espaço quantitativo alto. Nessa perspectiva, o GARP calcula os estados possíveis considerando as informações do fragmento de modelo de classe exposto na Figura 4.8. O estado (1) mostra que classe

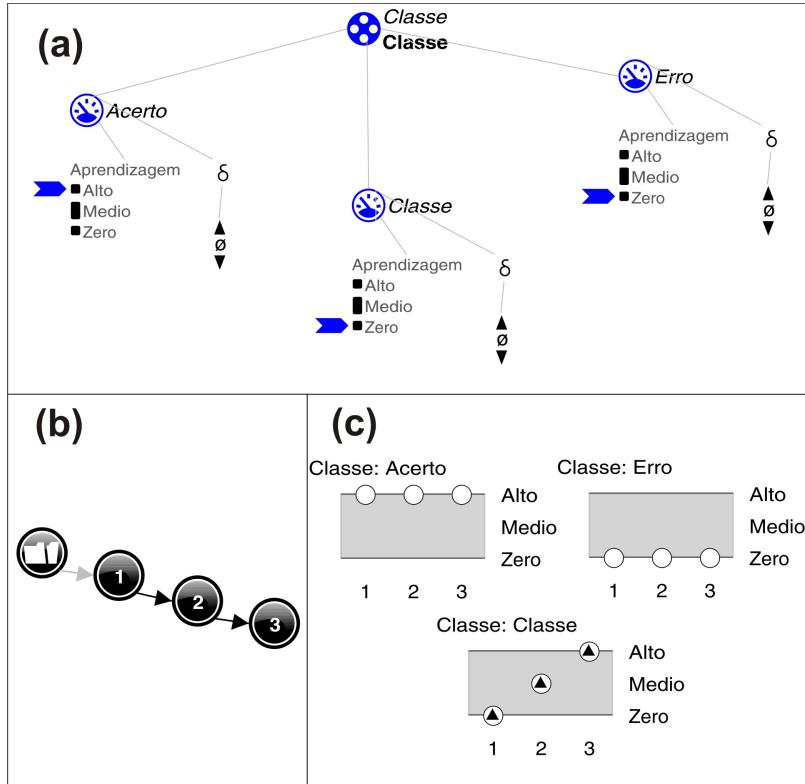


Figura 4.9: Simulação positivista para classe com acerto e erro

está no espaço quantitativo zero. Este é o estado demonstrado pela configuração inicial do cenário. O estado (2) apresenta que classe está no espaço quantitativo médio. Isso acontece por causa da influência positiva que a quantidade de acerto possui sobre classe. Por ser uma influência, para a TQP, isso significa que classe muda na direção de acerto. Por esse motivo, o estado (3) muda o valor de classe de médio para alto.

Perspectiva negativista

Essa perspectiva representa um modelo de domínio para classe, no qual o estudante tem um nível de conhecimento alto sobre classe, um nível baixo de acertos e um nível alto de erros. A simulação dessa perspectiva de cenário está ilustrada na Figura 4.10.

O quadro (Figura 4.10 a) ilustra a configuração do cenário. O quadro (Figura 4.10 b) ilustra o grafo de possíveis estados para a simulação. O quadro (Figura 4.10 c) ilustra os estados em formato de gráfico. A quantidade de erros foi marcada com espaço quantitativo alto. Nessa perspectiva, o GARP calcula os estados possíveis considerando

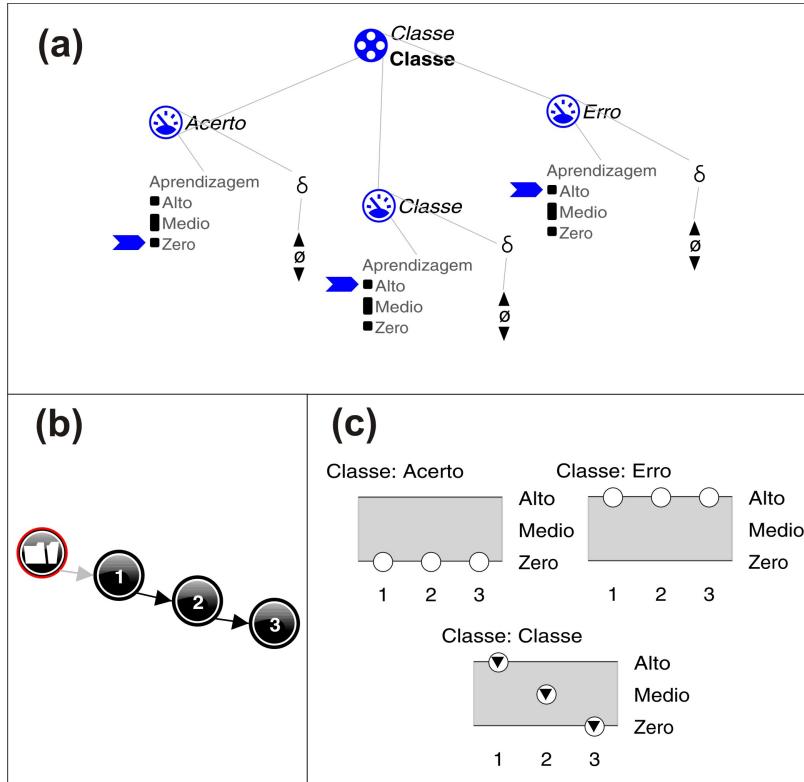


Figura 4.10: Simulação negativista para classe com acerto e erro

as informações do fragmento de modelo de classe. O estado (1) mostra que classe está no espaço quantitativo alto. Este é o estado demonstrado pela configuração inicial do cenário. O estado (2) apresenta que classe está no espaço quantitativo médio. Isso acontece por causa da influência negativa que a quantidade de erro possui sobre classe. Por ser uma influência, para a TQP, a classe muda na direção oposta do valor da quantidade de erro. Por esse motivo, o estado (3) muda o valor de classe de médio para baixo.

Perspectiva intermediária

Essa perspectiva representa um modelo de domínio para classe, em que o estudante tem um nível de conhecimento médio sobre classe, um nível médio de acertos e um nível médio de erros. A simulação dessa perspectiva de cenário está ilustrada na Figura 4.11.

O quadro (Figura 4.11 a) ilustra a configuração do cenário. O quadro (Figura 4.11 b) ilustra o grafo de possíveis estados para a simulação. O quadro (Figura 4.11 c) ilustra os estados em formato de gráfico. A quantidade de erros foi marcada com espaço quantitativo

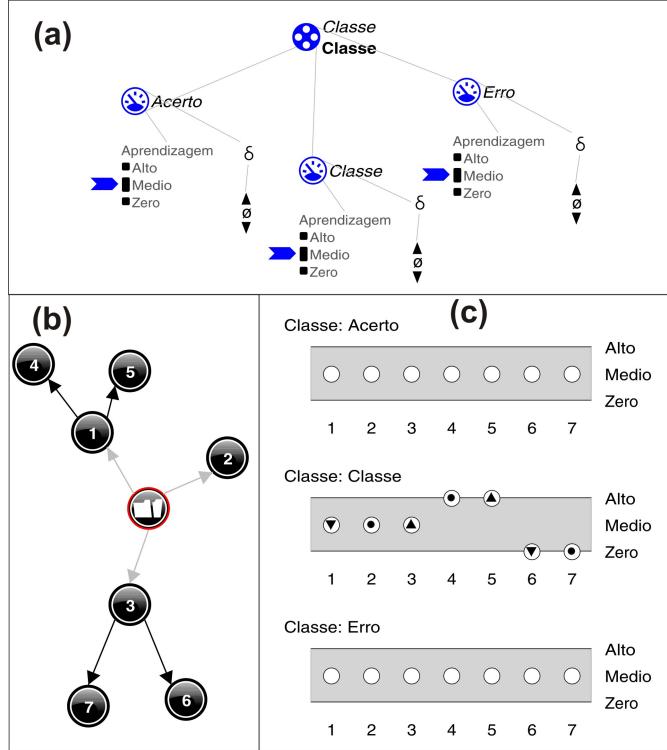


Figura 4.11: Simulação mediana para classe com acerto e erro

médio, assim como a quantidade de acertos. Nessa perspectiva, o GARP calcula os estados possíveis considerando as informações do fragmento de modelo de classe exposto na Figura 4.8. Os estados (1), (2), (3) mostram que o elemento classe está no espaço quantitativo médio. Este é o estado demonstrado pela configuração inicial do cenário. Os estados (4) e (5) apresentam que classe está no espaço quantitativo alto. Os estados (6) e (7) apresentam que classe está no espaço quantitativo zero.

Esses estados aconteceram por causa das influências positivas e negativas que as quantidades de acerto e de erro possuem sobre classe. Como o valor inicial de classe é médio e as duas quantidades de erros são altas, a simulação gera todos os caminhos possíveis iniciados a partir do espaço quantitativo médio da classe. Estando nesse ponto, o valor de classe pode tornar-se alto por influência do acerto ou zero por influência do erro. Por esse motivo, a simulação mostra todos os valores que classe pode assumir a partir da configuração do cenário em níveis médios.

4.5.2 Modelo de domínio para objeto e classe

Os conceitos de classe e objeto estão relacionados entre si. Adotando que o aprendizado de classe influencia o aprendizado de objeto é possível construir uma relação causal I+(classe, objeto). Essa relação exprime que o aprendizado de classe cresce no mesmo sentido que o valor de objeto. Em outras palavras, quanto mais o estudante assimila os conceitos sobre objeto, mais o conhecimento de classe que ele possui crescerá.

As relações causais entre as entidades dos domínios classe e objeto com as quantidades de acertos e erros são denotadas por I-(objeto, erro), I+(objeto, acerto), I-(classe, erro) e I+(classe, acerto). Todas as ilustrações das relações causais desse conjunto de domínio são ilustradas na Figura 4.12.

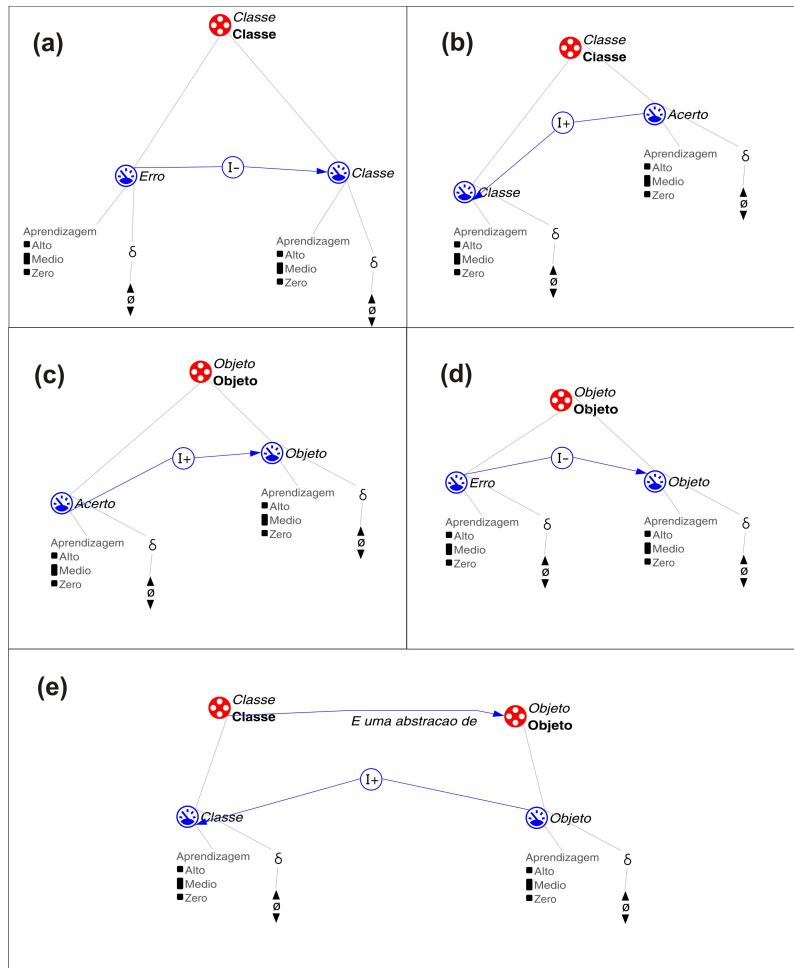


Figura 4.12: Fragmentos de modelo para a orientação a objetos que envolvem as entidades classe e objeto

O quadro (Figura 4.12 a) indica que a quantidade erro exerce uma influência negativa

sobre a quantidade classe. Isso significa que a relação entre classe e o erro é inversamente proporcional, ou seja, quando uma quantidade se eleva a outra decresce. O quadro (Figura 4.12 b) indica que um acerto exerce uma influência positiva sobre a quantidade de conhecimento de classe, ou seja, quando o valor quantitativo de acertos aumenta, o conhecimento de classe aumenta na mesma direção.

A combinação desses cinco fragmentos é realizada em um cenário com três perspectivas: positivista, negativista e intermediária.

Perspectiva positivista

Uma perspectiva positivista para esse modelo de domínio está ilustrada nas figuras 4.13, 4.14 e 4.13.

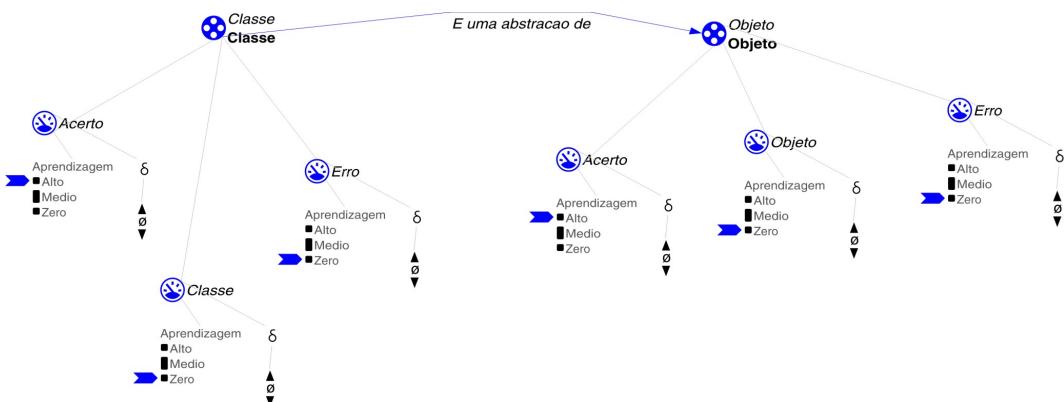


Figura 4.13: Cenário positivista que envolvem as entidades classe e objeto

Na perspectiva de cenário ilustrada na Figura 4.13, as quantidades de acertos são altas e as quantidades de erros são baixas. Como ponto inicial, classe e objeto possuem quantidade zero. Para esse cenário, chegou-se em um grafo com cinco estados, conforme ilustrados na Figura 4.14, e os estados possíveis são ilustrados na figura 4.15.

Os cinco estados gerados representam as possibilidades dos elementos classe e objeto dadas as perspectivas de acertos em alta e de erros em baixa. A média de aprendizado nessas condições pode ser dada qualitativamente pela soma dos estados dividido pela quantidade de estados. Convertendo hipoteticamente os espaços quantitativos de zero, médio e alto para 0, 5 e 10, obtém-se o resultado de $(30/5 = 6)$, um valor entre os espaços

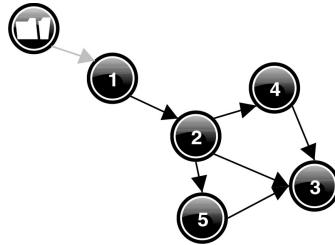


Figura 4.14: Grafo positivista que envolvem as entidades classe e objeto

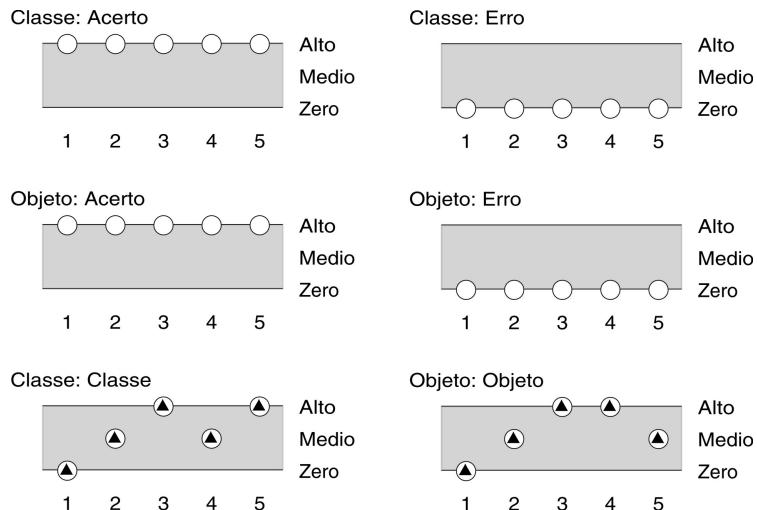


Figura 4.15: Gráfico positivista que envolvem as entidades classe e objeto

quantitativos médio e alto.

Perspectiva negativista

O mesmo cenário exposto da Figura 4.13 pode adquirir uma perspectiva negativista, na qual o estudante não está acertando, e sim cometendo muitos erros. Essa perspectiva é ilustrada na Figura 4.16.

Pela perspectiva negativista apresentada na Figura 4.16, a quantidade de acertos para ambas as entidades classe e objeto é igual a zero. As quantidades classe e objeto iniciam seus espaços quantitativos em alto, simulando um estudante que estava aprendendo mas começa a errar. A Figura 4.17 ilustra o grafo de estados para essa perspectiva, e os 8 caminhos encontrados são ilustrados no gráfico da Figura 4.18.

O grafo exposto na Figura 4.17 indica que essa perspectiva de cenário possui três estados iniciais igualmente prováveis (os estados 1, 2 e 3). O elemento classe recebe uma

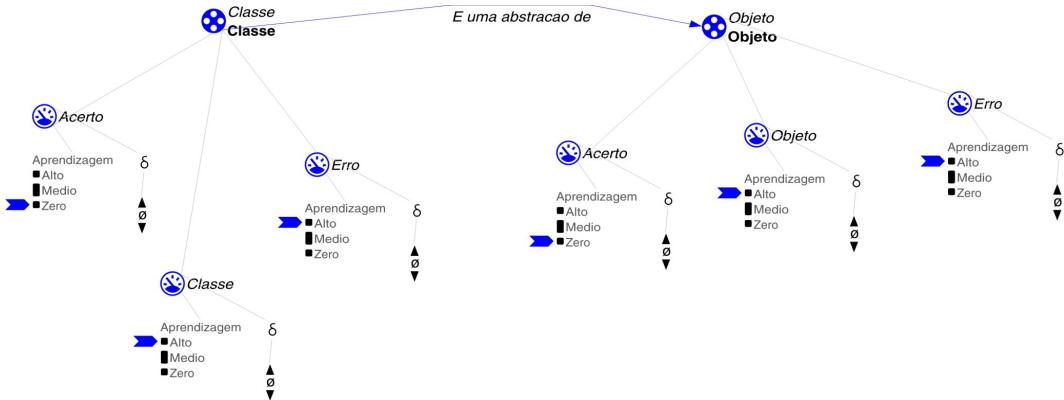


Figura 4.16: Cenário negativista que envolvem as entidades classe e objeto

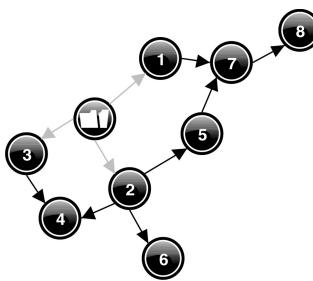


Figura 4.17: Grafo negativista que envolvem as entidades classe e objeto

influência positiva do elemento objeto, o que significa que sua derivada irá se comportar no sentido da mudança do elemento objeto. Por este motivo, os estados iniciais desse modelo, para o elemento classe são todos no espaço quantitativo alto, mas com as três derivadas possíveis associadas, pois nesse momento, a simulação não estabeleceu ainda se o elemento objeto será estável, crescente ou decrescente.

Após a criação dos três estados iniciais possíveis, cada um seguirá por um caminho. Os detalhamentos desses caminhos estão representados na Figura 4.18. O estado (1) seguirá para (7), (8), indicando o comportamento de classe e objeto: classe [alto, medio, zero] e objeto [alto, medio, medio]. O estado (2) seguirá para (6), em que o comportamento de classe e objeto são: classe [alto] e objeto [medio]. Outra alternativa do estado (2) é seguir para (4), no qual os valores de classe e objeto serão: classe [alto] e objeto [medio] também. O estado (2) pode seguir também para (5), (7) e (8), cujos os valores serão respectivamente: classe [alto, alto, medio, zero] e objeto [alto, medio, medio, medio]. O estado (3) segue para o estado (4) em que o comportamento de classe e objeto são: classe

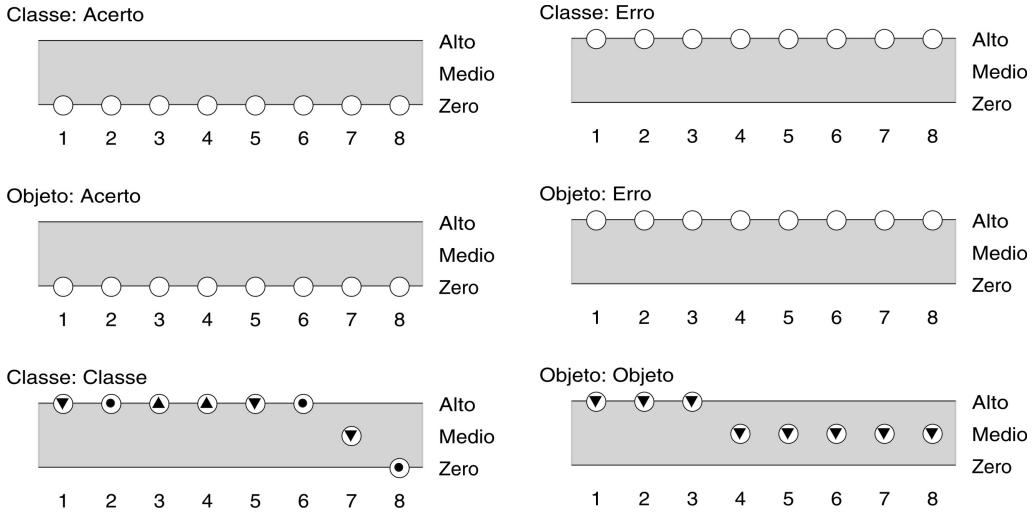


Figura 4.18: Gráfico negativista que envolvem as entidades classe e objeto

[*alto*] e objeto [*medio*].

Observando todos os estados e caminhos possíveis, conclui-se que, se um estudante está com um nível alto de conhecimento em classe e objeto, mas começa a errar, sua tendência é chegar em um estado no qual seu conhecimento seja baixo, o que indica para o sistema que o estudante precisa de intervenções, seja uma revisão de conteúdo e/ou a aplicação de novos exercícios, com níveis de dificuldades mais baixos, para que ele recupere o que não conseguiu aprender anteriormente.

Os comportamentos acima demonstrados são simulações, ou seja, o estudante não percorreu estes caminhos ainda. Para tanto, o sistema tutor inteligente identificando esses estados previamente pode intervir na tentativa de evitar que o estudante chegue no pior caso. Essa abordagem evita que o estudante se sinta desanimado, pois o sistema apresenta para ele a possibilidade de rever os conteúdos e exercitar novas questões em caso de baixo rendimento na aprendizagem.

Perspectiva intermediária

Outra perspectiva possível representa a possibilidade de erros e acertos serem médios, assim como o conhecimento do estudante. Esse cenário está ilustrado na Figura 4.19.

Neste cenário, adotou-se que o estudante estava com o conhecimento baixo sobre classe

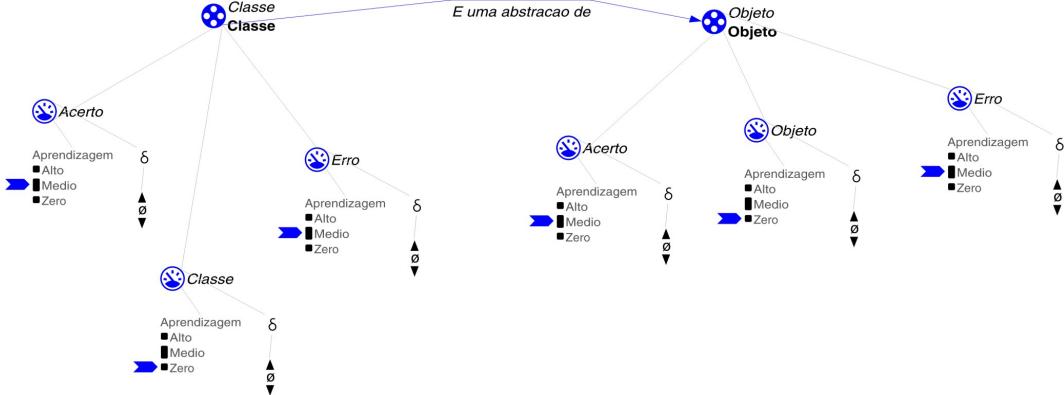


Figura 4.19: Cenário médio que envolvem as entidades classe e objeto

e objeto e se comporta ora acertando, ora errando. Para esse comportamento, a simulação qualitativa gerou um grafo que pode ser visualizado na Figura 4.20.

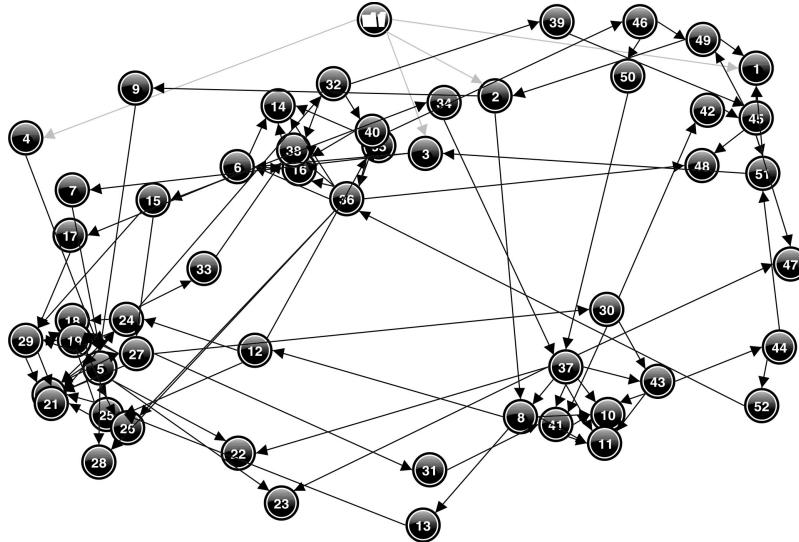


Figura 4.20: Grafo médio que envolvem as entidades classe e objeto

Era esperado, e foi confirmado pelo grafo de estados, que esse cenário possuísse muitos estados porque a perspectiva de erros e acertos médios leva a crer que o estudante pode estar errando totalmente, acertando totalmente ou ainda acertando mediamente. Então para cada conteúdo, o estudante pode se comportar das três maneiras citadas. O gráfico da Figura 4.21 ilustra o comportamento de todos os caminhos do grafo de estados.

Os estados para os elementos objeto e classe, conforme a Figura 4.21 ilustra, podem variar entre os espaços quantitativos alto, médio ou zero, conforme suposto para essa

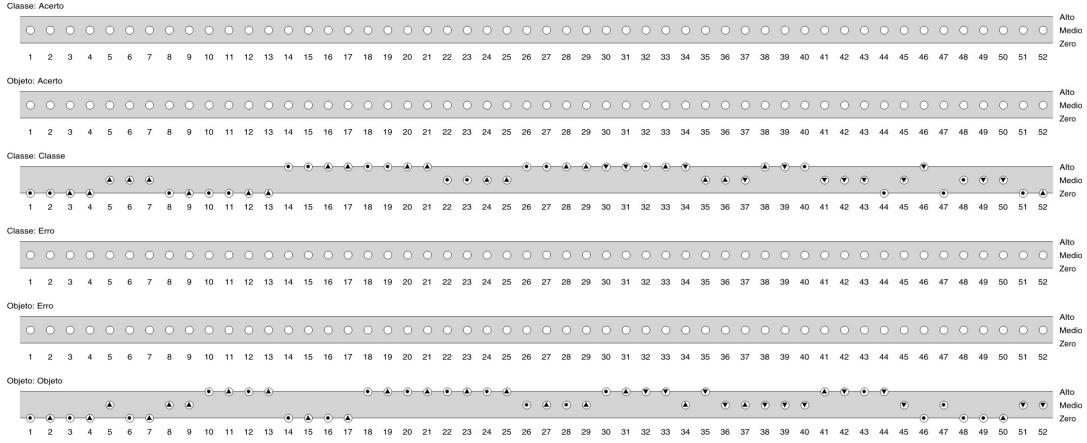


Figura 4.21: Gráfico médio que envolvem as entidades classe e objeto

configuração.

Foi avaliada uma outra perspectiva que mostra o exemplo de um aluno com estado inicial de aprendizagem em zero, e que possui o comportamento tendenciado para o erro, mas acerta medianamente também. A suposição nesse caso é de que o estudante estará no espaço qualitativo zero muitas vezes, assim como no espaço mediano. O gráfico da Figura 4.22 ilustra a perspectiva criada.

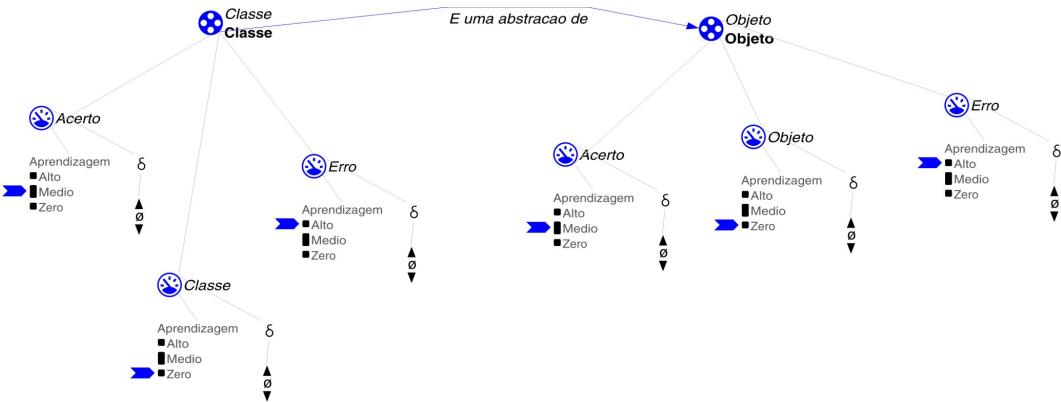


Figura 4.22: Cenário com acertos médios e erros altos que envolvem as entidades classe e objeto

A situação ilustrada na Figura 4.22 gera os estados ilustrados na Figura 4.23.

A descrição detalhada dos valores de cada estado está descrita na Figura 4.24 (Última página deste capítulo).

Apesar dos gráficos das Figuras 4.21 e 4.24 serem semelhantes, pois ambos demonstram o mesmo número de estados para cada espaço quantitativo (14 zeros, 17 médios, 21

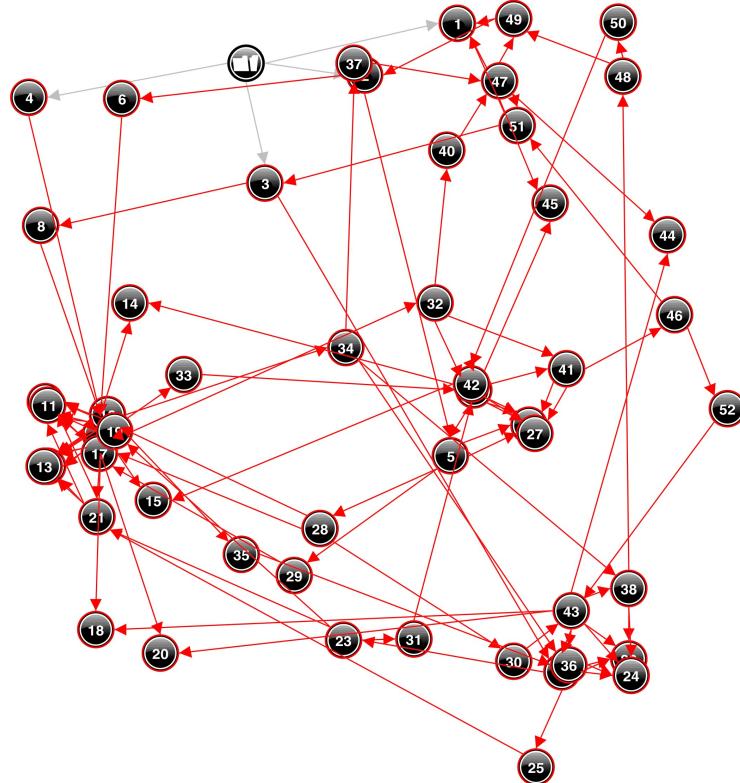


Figura 4.23: Grafo com acertos médios e erros altos que envolvem as entidades classe e objeto

altos para classe, assim como as mesmas medidas para objetos), os caminhos obtidos são diferentes.

4.6 Considerações finais

Este capítulo apresentou um estudo de caso que envolvem a os conceitos da disciplina de orientação a objetos. Foram apresentados cenários sob três perspectivas: positivista, negativista e intermediária.

Para esse estudo de caso, conclui-se que perspectivas positivistas ou negativistas são mais fáceis de analisar, visto que não produzem uma saída tão grande quanto a perspectiva intermediária. Por serem perspectivas com resultados mais breves, fazem com que a recomendação seja mais rápida e mais objetiva. Talvez as perspectivas intermediárias possam ser divididas em duas ou mais, diminuindo assim o volume de informações da saída.

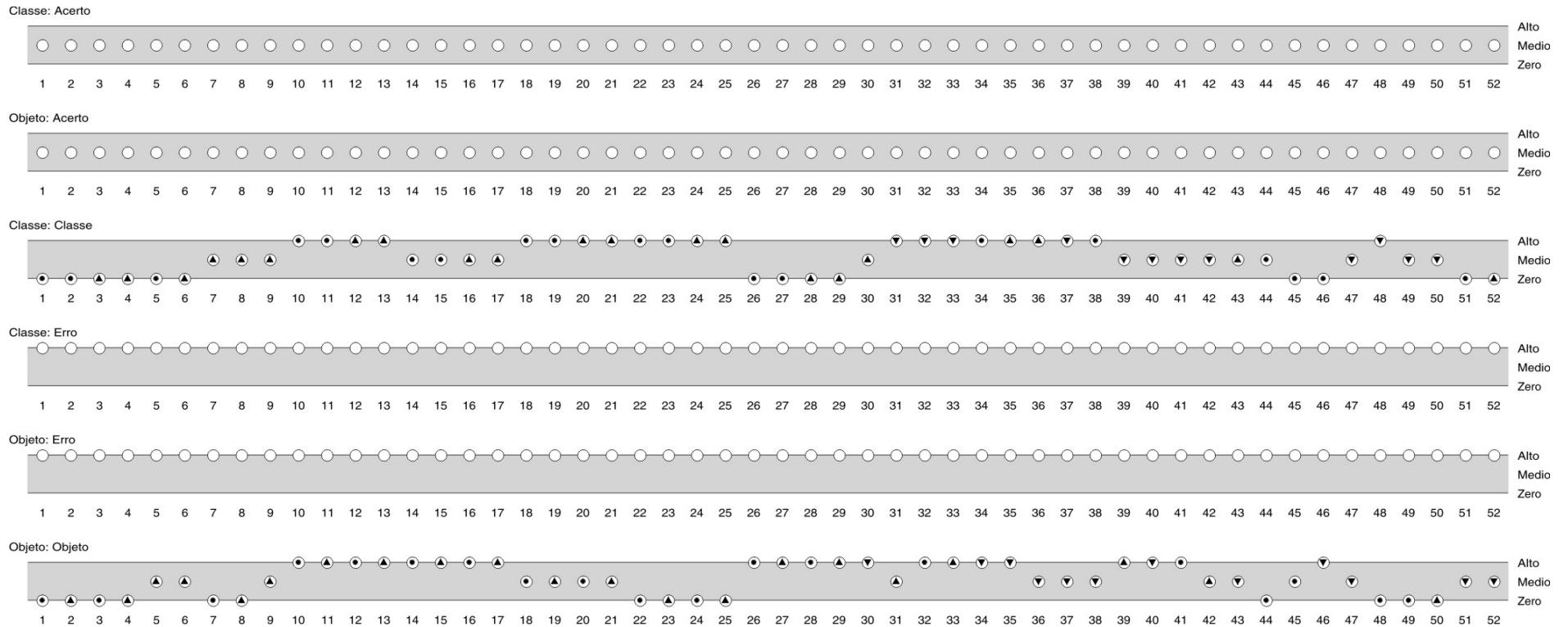


Figura 4.24: Gráfico com acertos médios e erros altos que envolvem as entidades classe e objeto

CAPÍTULO 5

CONSIDERAÇÕES FINAIS

5.1 Conclusão

O objetivo principal deste trabalho foi utilizar a teoria qualitativa de processos (TQP) para definir uma arquitetura de recomendação de próximas melhores atividades (conteúdos e exercícios), e modelar a disciplina de orientação a objetos como um estudo de caso.

A revisão bibliográfica apresentou um estudo sobre as principais abordagens para sistemas tutores inteligentes e também, um estudo sobre a TQP.

A modelagem do processo de aprendizagem com a TQP é uma forma de trabalhar o modelo de domínio de um sistema tutor inteligente sem usar as bases de regras e inferências, típicas da inteligência artificial (IA), de forma direta. É uma alternativa para modelar os conceitos atrelados a uma disciplina.

A arquitetura contruída, define que conteúdos, exercícios, acertos e erros fazem parte da aprendizagem e são inter-relacionados pelas definições de causalidades advindas da TQP. Os exercícios inseridos como parte da modelagem, continham acertos e erros, tornando a abordagem satisfatória, garantindo a indicação de próximas melhores atividades de uma forma coerente com o esperado para as situações modeladas.

A aplicação da arquitetura proposta no estudo de caso, permitiu o que as simulações fossem geradas a partir de três perspectivas: positivista, negativista e intermediária. Estas perspectivas representaram estados onde o estudante estava aprendendo ou não, sempre, na maioria ou em algumas vezes.

A análise dos grafos de estados e dos gráficos de situação permitiu o entendimento e a interpretação dos resultados obtidos sob as perspectivas existentes. Com as informações obtidas nas simulações, foi possível inferir qual o melhor próximo exercício ou conteúdo a ser apresentado a um estudante.

Para que a inferência de melhores próximas atividades aconteça, a modelagem do

objeto a ser aprendido precisa ser realizada conforme os princípios de proporcionalidades e influências que existem na teoria qualitativa dos processos. São essas informações de influências e proporcionalidades que atuam no comportamento do sistema.

Usar a teoria qualitativa dos processos para modelar uma disciplina proporcionou que fatores novos surgissem durante a fase de modelagem. Os espaços quantitativos são um exemplo disso. Foi necessário adaptar o que antes era usado para medir qualitativamente propriedades físicas para algo que fosse capaz de mensurar o valor qualitativo do aprendizado. Considera-se que foi necessária uma atenção especial em virtude da existência de espaços quantitativos que são valores fundamentais para o sistema. São eles que mostram o deslocamento das quantidades, que na modelagem para o ensino foram chamadas de conhecimentos, questões, acertos e erros.

Na literatura pesquisada, existem poucos trabalhos relacionados a TQP que demonstram totalmente seus experimentos, mas a soma dos trabalhos encontrados contribuiu positivamente para a construção de exemplos adequados à arquitetura proposta.

Sistemas de recomendação de próximas melhores atividades também não possuem pesquisa vasta, no entanto, foi possível, com o estudos de modelos de comportamentos desenvolvidos para entidades físicas da TQP, construir modelos de comportamento para situações de aprendizagem. Estes modelos usam relações onde as influências positivas ou negativas representam a maneira como a quantidade de conhecimento de um estudante pode ser medida e alterada qualitativamente.

A relação e o uso da TQP na informática não é objeto de estudo de muitos pesquisadores, no entanto, existem vários trabalhos direcionados a outras áreas do conhecimento, fato que permite conhecer os procedimentos técnicos para a elaboração de modelos de processos qualitativos.

5.2 Trabalhos futuros

Com base no desenvolvimento desta pesquisa, que contempla a modelagem de uma disciplina com a TQP para a recomendação de próximos melhores exercícios, é possível que os demais elementos da arquitetura sejam desenvolvidos também. São eles: a interface, o

repositório de conteúdos e exercícios e os módulos do STI clássicos, que não foram objetos de desenvolvimento deste estudo.

Sugere-se que as análise das simulações sejam automatizadas, partindo do princípio que foi utilizado o conhecimento humano para diagnosticar e discorrer acerca dos resultados.

Estudar a utilização da arquitetura proposta para modelar outras disciplinas, não somente da área de informática, é possível e interessante. Por fim, pode-se sugerir que é factível modelar a cognição humana considerando outros fatores, como, conhecimentos prévios, faixa etária, competências e habilidades.

APÊNDICE A

FERRAMENTA GARP

O modelo conceitual de aprendizagem proposto neste trabalho está desenhado no simulador qualitativo GARP, do inglês *General Architecture for Reasoning in Physics*. Esse sistema especialista, escrito em Prolog, possui uma metalinguagem associada a uma máquina de inferência que possibilita um processo de raciocínio automatizado com sistemas de diversos domínios do conhecimento [20, 8].

A.1 Principais funcionalidades

O GARP utiliza blocos de construção para criar modelos de processos. Os blocos são classificados em: entidades físicas simples e relações estruturais entre elas; quantidades e espaços quantitativos (e as restrições), e sua variação no tempo; dependências entre as quantidades e seus valores [6]. Esses blocos de construção de modelos, junto com os cenários iniciais, são especificados em uma biblioteca de fragmentos de modelo. Os cenários consistem usualmente de uma descrição estrutural de um sistema e da definição dos valores iniciais de algumas quantidades [12].

A simulação de um cenário é realizada por meio de um motor de raciocínio. O GARP usa a estratégia “estado por estado”, para construir o grafo de estados. A Figura A.1 apresenta uma visão geral das principais inferências do GARP.

Fragmentos de modelo e cenários são as entradas para o mecanismo de raciocínio do GARP. A partir dessas entradas, um ou mais estados são construídos pelo procedimento “Procurar estados”. Os estados encontrados são os nós iniciais no grafo de estado (ou grafo de comportamento). Em seguida, as possíveis mudanças são determinadas pelo procedimento “Buscar transações” para cada novo estado encontrado. Essas mudanças formam os “Cenários de transição”, que são novamente a entrada para o procedimento “Procurar estados”.

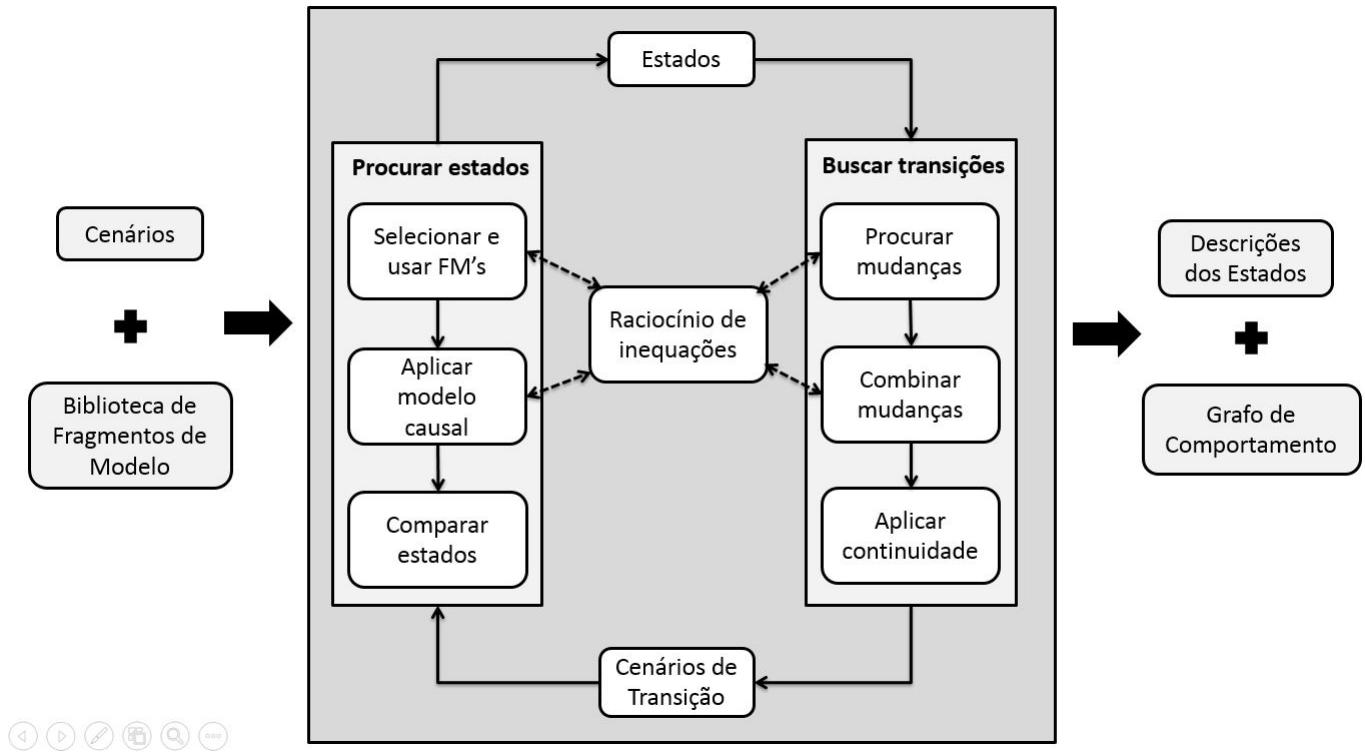


Figura A.1: Motor de raciocínio da ferramenta GARP

Cenários e cenários de transição podem levar a estados já existentes ou para novos estados. No primeiro caso, uma ligação de transição é adicionada ao grafo de estados (entre os nós já existentes). No segundo caso, uma ligação e um nó são adicionados. Esse processo se repete até que todos os estados sejam analisados em relação a possíveis mudanças. As mudanças de estados em um cenário são construídas com base no raciocínio de inequações [10].

A abordagem usada no GARP de procura de estado por estado tem duas vantagens. Em primeiro lugar, o usuário tem a possibilidade de controlar a simulação e decidir qual o comportamento do sistema pode ser mais investigado. Esse fato é particularmente útil como um instrumento para a construção de modelos grandes em geral. Em segundo lugar, o raciocínio do motor é mais “humanóide” (traduzido de *HumanLike*), no sentido de que considera a situação e determina como ela está mudando. Este ponto em particular fornece suporte para a usabilidade do software no contexto educacional.

A interface da ferramenta GARP é sensitiva ao contexto e restringe a possibilidade a erros. Isso permite ao usuário estar livre de erros sintáticos e alguns erros semânticos. Um

exemplo disso, acontece quando o modelador tenta apagar uma definição de quantidade, a ferramenta avisa que a quantidade está sendo utilizada em determinados fragmentos de modelo [6].

A.2 Principais componentes

O software GARP possui vários componentes que podem ser explorados durante a modelagem de processos. A seguir são descritos componentes do GARP que foram usados neste trabalho.

Entidades e atributos

Entidades são os objetos físicos ou conceitos abstratos que desempenham um papel dentro do sistema. Estas entidades estão dispostas numa hierarquia de subtipos. No GARP, a definição das entidades acontece de maneira estruturada, usando a hierarquia “*is a*” (“é um”). Isso significa que entidades mais específicas são definidas a partir de entidades mais genéricas. O GARP define uma entidade padrão geral denominada “*Entity*”.

Um exemplo de entidades no GARP está ilustrado na Figura A.2.

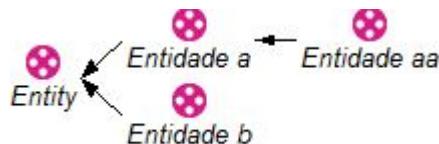


Figura A.2: Entidades genéricas no GARP

O conjunto de entidades ilustrado na Figura A.2, indica que a partir da entidade geral foram criadas duas outras entidades: “Entidade a” e “Entidade b”. A entidade “a” ainda define uma entidade filha, denominada de “Entidade aa”. Esta representação está em formato horizontal.

O uso de entidades no GARP segue a premissa da TQP que afirma que entidades são representações de objetos do mundo real. Qualquer processo que seja modelado seguindo os princípios da TQP deverá transformar seus conceitos em objetos (entidades) ao iniciar

a modelagem do processo. As entidades são usadas em fragmentos de modelo e podem originar diversas instâncias de si mesmas como elementos a serem relacionados entre si. Essa cadeia de relacionamento é responsável pelo comportamento do sistema. A estruturação utilizada permite ainda a extensão do modelo e o seu reúso, que são características fundamentais do processo de modelagem na TQP.

No estudo de caso apresentado nesse trabalho, foi construída uma árvore de entidades com os conceitos da disciplina de orientação a objetos. A Figura A.3 é uma representação do estudo de caso em formato vertical.

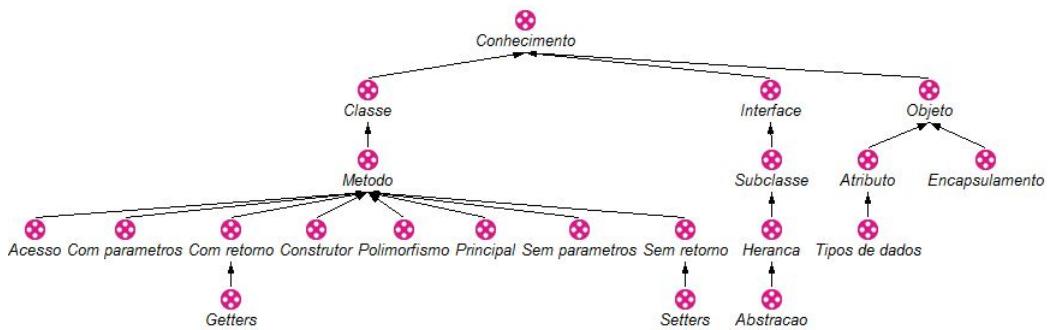


Figura A.3: Entidades criadas a partir do paradigma de orientação a objetos

A estrutura definida e apresentada na A.3 propõe que existe uma entidade base denominada “Conhecimento”. Essa entidade possui entidades filhas, “Classe”, “Objeto” e “Interface”. Cada classe filha pode possuir suas próprias entidades filhas. No exemplo da Figura A.3, a entidade “Classe” possui “Método” como filha. A entidade “Método” possui, por sua vez, outras entidades filhas.

No caso deste trabalho, optou-se por modelar cada conceito como uma entidade pois dessa forma será possível criar relações entre eles, tanto de forma interna, quanto de forma externa. Isso significa que quando o modelador estiver construindo questões sobre “Classe”, elas deverão ser derivadas da entidade “Classe”, mas poderão estar relacionadas com questões derivadas de outras entidades.

Configurações

As configurações são utilizadas para representar as relações estruturais entre as entidades. Inicialmente, uma configuração tem de ser definida por meio da atribuição de um nome a ela. A configuração pode ser posteriormente utilizada para especificar uma ligação entre os casos de entidades em cenários ou fragmentos de modelo. A Figura A.4, ilustra a criação de uma configuração no software GARP.

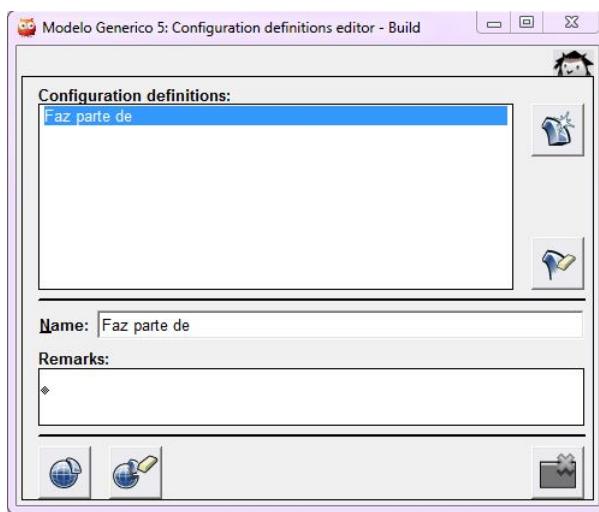


Figura A.4: Exemplo de configuração criada no GARP

No exemplo mostrado na Figura A.4, foi criada a configuração denominada “Faz parte de”, e ela indicará quando uma entidade (conhecimento) fará parte de outra entidade (conhecimento). Essa relação poderá ser construída em fragmentos de modelo ou então em um cenário.

No estudo de caso foram construídas algumas configurações que representam as relações que as entidades do domínio possuem entre si. São elas: “é um”, “é uma abstração de”, “é uma implementação de”, “Permite”, “Pode ser” e “Possui”. Estas configurações podem ser visualizadas no grafo geral de entidades do domínio, representado pela Figura 4.1, do Capítulo 5.

Espaços quantitativos e quantidades

No software GARP, os espaços quantitativos constituem-se em um conjunto ordenado de pontos e intervalos que representam os valores possíveis para magnitude e derivada das quantidades. Valores de espaços quantitativos de variáveis distintas podem ser relacionados via desigualdades e correspondências [6].

Os espaços quantitativos são relacionados nos fragmentos de modelo e apontados como caminhos previstos no cenário criado. Quando o cenário for colocado em execução, o GARP constrói a árvore de estados com base nos espaços quantitativos marcados como possíveis soluções ou caminhos.

As quantidades representam recursos de entidades que, em princípio, podem mudar de um estado para outro. Toda quantidade deve estar associada a um espaço quantitativo, que é um intervalo, entre zero e o infinito, e que indica os estados pelos quais o sistema modelado poderá passar. A A.5 ilustra um exemplo de criação de quantidades no GARP.

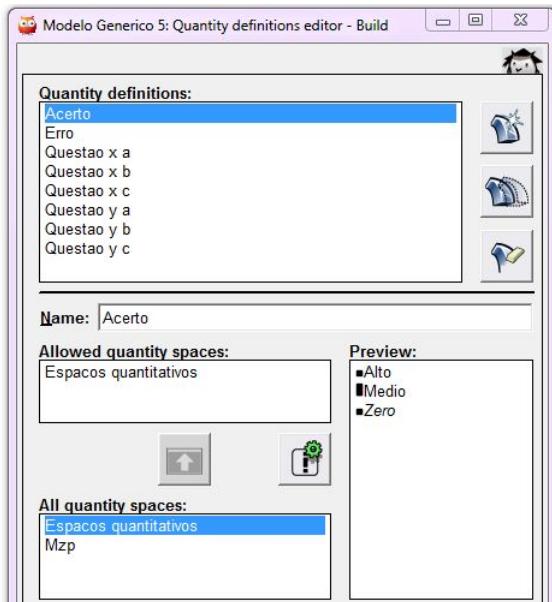


Figura A.5: Exemplo de espaços quantitativos criados no GARP

No exemplo mostrado na Figura A.5 as quantidades são definidas como sendo as questões do modelo mais erros e acertos. Logo “Questao x a” refere-se a uma questão de um domínio “X” denominada de “A”¹. Cada quantidade deve estar atrelada a um

¹A notação para quantidades no GARP não permite o uso de valores numéricos e nem de caracteres

espaço quantitativo, que no exemplo é denominado de “Espaços quantitativos” e possui três valores definidos: “Zero”², “Medio” e “Alto”.

No estudo de caso apresentado no Capítulo 5 foram criadas vinte quantidades que representam os conteúdos, duas quantidades por conteúdo representando questões e duas quantidades representando os acertos e os erros dos estudantes. As quantidades que representam os conteúdos são usadas para fazer a recomendação de melhor conteúdo a ser exibido ao estudante. As entidades de questões são usadas para realizar a recomendação de melhor questão a ser apresentada aos estudantes. A entidades de erros e acertos são auxiliares das demais quantidades.

Para a modelagem do processo de aprendizagem de conteúdos, adotou-se como espaços quantitativos, neste trabalho, as medidas qualitativas zero, médio e alto. Quando o estudante está no espaço quantitativo zero, não aprendeu nada do conceito apresentado. Quando o estudante está no espaço quantitativo médio, aprendeu medianamente o conteúdo apresentado. Quando o estudante está no espaço quantitativo alto, aprendeu satisfatoriamente o conteúdo apresentado.

Dependências causais e fragmentos de modelo

Dependências causais são usadas para modelar como os processos afetam o sistema. Os efeitos diretos são modelados como influências, os efeitos indiretos são modelados como proporcionalidades. Ambos podem assumir valores positivos ou negativos e são representados em fragmentos de modelo no GARP.

Para adicionar uma influência, o fragmento modelo deve conter, no mínimo, duas quantidades. Um exemplo de fragmento de modelo com dependências causais está ilustrado na Figura A.6.

O exemplo ilustrado na Figura A.6, representa a existência de dois conceitos (entidades) diferentes denominados respectivamente de Conceito 1 e Conceito 2. O elemento Conceito 1 possui duas quantidades ligadas a si: “Questao x a” e “Questao x b”. O

especiais

²O valor denominado “Zero” é obrigatório para a modelagem no GARP e seu nome não pode ser alterado.

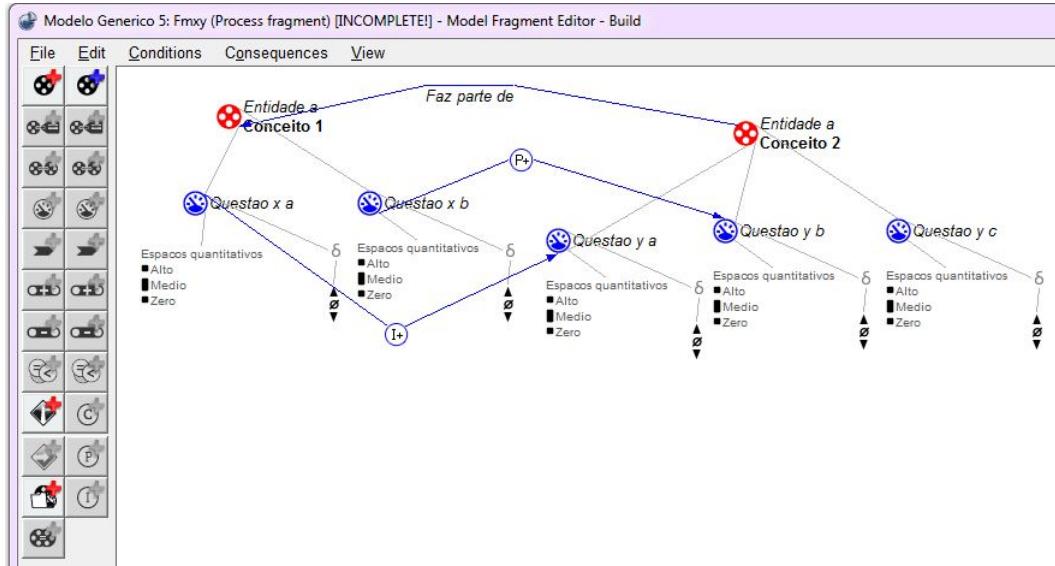


Figura A.6: Exemplo de relações causais criadas no GARP

elemento Conceito 2 possui três quantidades ligadas a si: “Questao y a”, “Questao y b” e “Questao y c”. Todas as quantidades possuem o mesmo espaço quantitativo, com as medidas zero, médio e alto.

O fragmento de modelo exibido na Figura A.6 representa a modelagem gráfica para as relações causais: [Conceito1, Conceito2], I+(Questao y a, Questao x a), P+(Questao y b, Questao x b).

Outras relações como as ilustradas no exemplo são abordadas nos Capítulos 4 e 5.

Cenários e Simulações

No GARP, cenários são ambientes nos quais os fragmentos de modelo são combinados (não necessariamente todos) para construir o processo que está sendo modelado. Neste cenário é possível dizer quais são os valores iniciais das quantidades que fazem parte do modelo. Também é possível, mas não necessário, influenciar manualmente os valores das derivadas, e assim, alterar o comportamento do sistema temporariamente sem usar os fragmentos de modelo.

A Figura A.7 ilustra a construção de um cenário.

O cenário ilustrado na Figura A.7 representa a combinação de vários fragmentos de modelo criados anteriormente. Fazem parte dele, por exemplo, as questões “x a”, “x

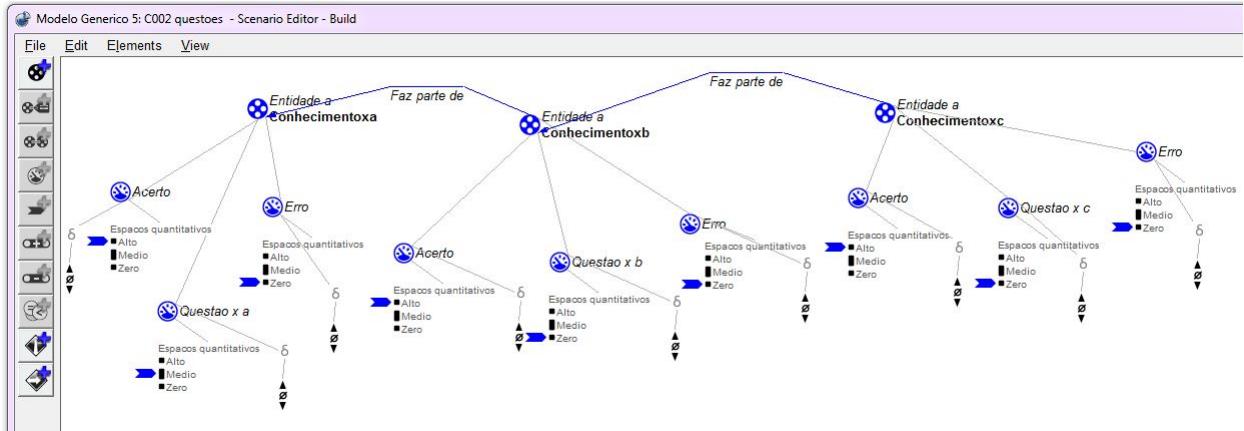


Figura A.7: Exemplo de cenário criado no GARP

b” e “x c”, onde “x b” e “x c” possuem a relação causal $P+(Questao\ x\ c, Questao\ x\ b)$. Todas as questões possuem relações causais com as quantidades de acertos e erros. As notações que definem as demais relações causais representadas em fragmentos de modelo não expostos nesse exemplo são: $I+(Questao\ x\ a, acerto)$, $I-(Questao\ x\ a, erro)$, $I+(Questao\ x\ b, acerto)$, $I-(Questao\ x\ b, erro)$, $I+(Questao\ x\ c, acerto)$, $I-(Questao\ c\ a, erro)$.

Na Figura A.7 pode-se observar que em cada espaço quantitativo existe uma marcação (seta azul). Os valores marcados correspondem ao estado (1) do sistema modelado. A partir dele serão gerados os próximos estados que são exibidos em formato de grafo de comportamento e também em gráficos com descrições dos estados. A Figura A.8 ilustra um grafo de estados e a Figura A.9 ilustra um gráfico com as descrições de comportamentos obtidas após a execução do cenário presente na Figura A.7.

No gráfico ilustrado pela A.9 é possível identificar os valores que as quantidades “Questao x a”, “Questao x b” e “Questao x c” assumem durante a execução da simulação. Ao combinar essas informações com o grafo de estados exibido na Figura A.8, podemos identificar a mudança dos espaços quantitativos ao longo dos caminhos possíveis. Pode-se, por exemplo, eleger o caminho $(1, 2, 6)$ e nele apresentar os valores das quantidades do cenário. A quantidade “Questao x a”, no caminho $(1, 2, 6)$, assumiu os valores $[medio, medio, alto]$. A quantidade “Questao x b”, no caminho $(1, 2, 6)$, assumiu os valores $[zero, medio, alto]$. A quantidade “Questao x c”, no caminho $(1, 2, 6)$, assumiu os valores $[zero, medio, zero]$.

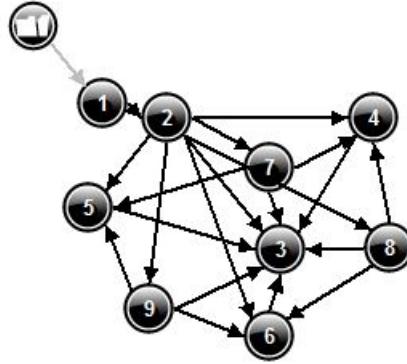


Figura A.8: Exemplo de grafo de estados criado no GARP

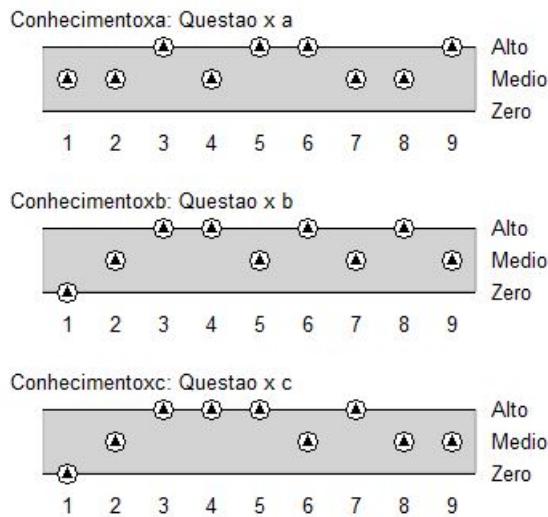


Figura A.9: Exemplo de gráfico com descrições de estados criado no GARP

A interpretação das mudanças de valores das quantidades deve ser realizada de acordo com o contexto onde elas estão aplicadas. Este exemplo apresentado está modelado para o contexto do processo de ensino e resultado apresentado pelo gráfico indica que o estudante está conseguindo apresnder e que a “Questao x c” é uma boa candidata a ser exibida como próximo melhor exercício.

O objetivo deste capítulo foi apresentar algumas funcionalidades do GARP. Este software foi usado para construir os modelos de aprendizagem apresentados no Capítulo 4, assim como, o modelo de domínio que é objeto de estudo do Capítulo 5.

BIBLIOGRAFIA

- [1] Front matter. Etienne Wenger, editor, *Artificial Intelligence and Tutoring Systems*. Morgan Kaufmann, 1987.
- [2] *Ecological informatics : understanding ecology by biologically-inspired computation*. 2004.
- [3] Joel C. Adams. Object-centered design: a five-phase introduction to object-oriented programming, 1996.
- [4] John R Anderson. Act-r, 1998.
- [5] John R. Anderson, C.Franklin Boyle, Albert T. Corbett, e Matthew W. Lewis. Cognitive modeling and intelligent tutoring. *Artificial Intelligence*, 42(1):7 – 49, 1990.
- [6] Symone Christine De Santana Araújo. *Modelos De Simulação Baseados Em Raciocínio Qualitativo Para Avaliação Da Qualidade Da Água Em Bacias Hidrográficas*. Tese de Doutorado, Universidade de Brasília, Brasília, 2005.
- [7] Giorgio Brajnik, Udine Italy, e Marji Lines. Qualitative modeling and simulation of socio-economic phenomena. *Journal of Artificial and Social Simulation*, 1997.
- [8] Bert Bredeweg. *Expertise in Qualitative Prediction of Behaviour*. Tese de Doutorado, University of Amsterdam, Amsterdam, The Netherlands, 1992.
- [9] Bert Bredeweg e Kenneth D. Forbus. Qualitative modeling in education. *AI Magazine*, 24(4):35–46, 2004.
- [10] Bert Bredeweg, Floris Linnebank, Anders Bouwer, e Jochem Liem. Garp3: Workbench for qualitative modelling and simulation. *Ecological Informatics*, 4(5–6):263–281, 2009. Special Issue: Qualitative models of ecological systems.
- [11] Bert Bredeweg e Peter Struss. Current topics in qualitative reasoning. *AI Magazine*, 24(4):13–16, 2004.

- [12] Bert Bredeweg e Radboud Winkels. Qualitative models in interactive learning environments: An introduction, 1998.
- [13] Janet Carter, Kirsti Ala-Mutka, Ursula Fuller, Martin Dick, John English, William Fone, e Judy Sheard. How shall we assess this? *SIGCSE Bull.*, 35(4):107–123, junho de 2003.
- [14] William J. Clancey. Diagnosis, teaching, and learning: An overview of guidon2 research, 1987.
- [15] Mihaly Csikszentmihalyi. *Flow: The Psychology of Optimal Experience*. Harper Perennial, New York, NY, March de 1991.
- [16] Mihaly Csikszentmihalyi. *Finding Flow: The Psychology of Engagement with Every-day Life*. Basic Books, 1997.
- [17] T. Orehovacki Danijel Radosevic e Zlatko Stapic. Automatic on-line generation of student's exercises in teaching programming. *Artificial Intelligence*, páginas 22–24, 2010.
- [18] Teresa del Soldato e Benedict du Boulay. Implementation of motivational tactics in tutoring systems. *J. Artif. Intell. Educ.*, 6(4):337–378, janeiro de 1995.
- [19] Fabrício Doi. Objetos adaptativos: aplicação da tecnologia adaptativa á orientação a objetos. Dissertação de Mestrado, Universidade de São Paulo, São Paulo. BR, 2007.
- [20] Bruno Versiani dos Anjos. Modelagem em raciocínio qualitativo sobre degradação de vegetação ripária na microbacia do riacho fundo. *REMEA - Revista Eletrônica do Mestrado de Educação Ambiental*, 17, 2006. Universidade de Brasília. Brasília.
- [21] Brian Falkenhainer e Kennethd Forbus. Compositional modeling: Finding the right model for the job, 1991.
- [22] Dave W. Farthing, Dave M. Jones, e Duncan McPhee. Permutational multiple-choice questions: An objective and efficient alternative to essay-type examination questions.

Proceedings of the 6th Annual Conference on the Teaching of Computing and the 3rd Annual Conference on Integrating Technology into Computer Science Education: Changing the Delivery of Computer Science Education, ITiCSE '98, páginas 81–85, New York, NY, USA, 1998. ACM.

- [23] Márcio Ferrari. B. f. skinner, o cientista do comportamento e do aprendizado. *Revista Nova Escola*, 2008. [Acesso em: 16 jul. 2014].
- [24] K. D. Forbus. Exploring artificial intelligence. capítulo Qualitative Physics: Past, Present, and Future, páginas 239–296. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988.
- [25] Kenneth D. Forbus. Qualitative reasoning about physical processes. páginas 326–330, 1981.
- [26] Kenneth D. Forbus. Qualitative process theory. *Artif. Intell.*, 24(1-3):85–168, dezembro de 1984.
- [27] Kim Willian Forest. Protótipo de um sistema tutor de orientação a objetos, 2004.
- [28] Mariane F. Galhardo e Luciana A. M. Zaina. Simulação para ensino de conceitos da orientação a objetos. 2004.
- [29] Joanna Golinska-Pilarek. On decidability of a logic for order of magnitude qualitative reasoning with bidirectional negligibility. *Logics in Artificial Intelligence - 13th European Conference, JELIA 2012, Toulouse, France, September 26-28, 2012. Proceedings*, páginas 255–266, 2012.
- [30] Simone Salvador Gomes. Quando o jogo flui: uma investigação sobre a teoria do fluxo no voleibol. Dissertação de Mestrado, Universidade Federal de Juiz de Fora, Juiz de Fora, MG, 2010.
- [31] Marco A. Gómez-Martín, Pedro P. Gómez-Martín, e Pedro A. González-Calero. Aprendizaje activo en simulaciones interactivas. *Revista Iberoamericana de Inteligencia Artificial*, 11(33):25–36, 2007.

- [32] F. Guerrin e J Dumas. Knowledge representation and qualitative simulation of salmon reed functioning. *BioSystems*, páginas 75–84, 2001.
- [33] J. R. Hartley e D. H. Sleeman. Towards more intelligent teaching systems. *Int. J. of Man-Machine Studies*, (5):215–236, 1973.
- [34] Michael Heilman. *Automatic factual question generation from text*. Tese de Doutorado, 2011.
- [35] R. Kaplan e D. Rock. New directions for intelligent tutoring, 1995.
- [36] Terry King e Emma Duke-Williams. Using computer-aided assessment to test higher level learning outcomes. *5th Annual CAA Conference*, 2001.
- [37] Johan De Kleer e John Seely Brown. A qualitative physics based on confluences. *Artificial Intelligence*, 24(1-3):7–83, 1984.
- [38] Michael Kolling, Bett Koch, e John Rosenberg. Requirements for a first year object-oriented teaching language. *ACM SIGCSE Bulletin*, 27:173–177, 1995.
- [39] Benjamin Kuipers. Qualitative simulation. *Artificial Intelligence*, 29:289–338, 1986.
- [40] Maici Duarte Leite. *Arquitetura Para Remediação de Erros Baseada em Múltiplas Representações Externas*. Tese de Doutorado, Universidade Federal do Paraná, Curitiba, PR, 2013.
- [41] Raymond Lister. Objectives and objective assessment in cs1. *Proceedings of the Thirty-second SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '01, páginas 292–296, New York, NY, USA, 2001. ACM.
- [42] Vania Bessa Machado e Bert Bredeweg. Building qualitative models with homer: A study in usability and support. *Proceedings of the 17th International workshop on Qualitative Reasoning*, QR'03, P. s and B. Bredeweg (Eds, páginas 39–46. Morgan Kaufmann Publishers, 2003.

- [43] Pedro Gómez-Martín Marco Gómez-Martín e e Pedro González-Calero. Game-based learning as a new domain for case-based reasoning. *Workshop on Computer Gaming and Simulation Environments*, 2005.
- [44] J. McTaggart. Intelligent tutoring systems and education for the future, 2001.
- [45] Antonija Mitrovic e Brent Martin. Evaluating the effects of open student models on learning. Paul De Bra, Peter Brusilovsky, e Ricardo Conejo, editors, *AH*, volume 2347 of *Lecture Notes in Computer Science*, páginas 296–305. Springer, 2002.
- [46] Antonija Mitrovic e Brent Martin. Evaluating adaptive problem selection. Paul De Bra e Wolfgang Nejdl, editors, *AH*, volume 3137 of *Lecture Notes in Computer Science*, páginas 185–194. Springer, 2004.
- [47] HyacinthS. Nwana. Intelligent tutoring systems: an overview. *Artificial Intelligence Review*, 4(4):251–277, 1990.
- [48] Jenifer Satie Vaz Ogasawara. O conceito de aprendizagem de skinner e vygotsky: Um diálogo possível, 2009.
- [49] Robert I. Muetzelfeldt Paulo S.B.A. Salles e Helen Pain. Qualitative models in ecology and their use in intelligent tutoring systems. *International Workshop of QR*, 1996.
- [50] Andrey Ricardo Pimentel. Medidas cognitivas para o ensino de conceitos visuais com sistemas tutores inteligentes. Dissertação de Mestrado, Universidade Federal do Paraná, Curitiba, PR, 1997.
- [51] Andrey Ricardo Pimentel e Alexandre Ibrahim Direne. Medidas cognitivas para o ensino de conceitos visuais com sistemas tutores inteligentes. *Revista Brasileira de Informática na Educação (RBIE)*, páginas 17–24, 1998.
- [52] Edson Pinheiro Pimentel. *Um modelo para avaliação e acompanhamento contínuo do nível de aquisição de conhecimento do aprendiz*. Tese de Doutorado, ITA, São José dos Campos, SP, 2006.

- [53] Paulo Salles. *Qualitative models in ecology and their use in learning environments.* Tese de Doutorado, Universidade de Edimburgo, Edimburg, 1997.
- [54] Roger Schank e Chip Cleary. *Engines for Education.* Routledge, 1995.
- [55] John Self. The defining characteristics of intelligent tutoring systems research: ITSs care, precisely. *International Journal of Artificial Intelligence in Education (IJAIED)*, 10:350–364, 1998. The following papers are extended versions of some of the invited papers presented at ITS 98.
- [56] Jose Armando Valente Seymour Papert e Beatriz Bitelman. *Logo: computadores e educação.* Brasiliense, 1980.
- [57] Neumann M. Tullos, D.D. e J.J.A. Sanchez. Development of a qualitative model for investigation benthic community response to anthropogenic activities. *IProceedings of the 18th International Workshop on Qualitative Reasoning (QR04)*, páginas 179 – 185, 2004.
- [58] José Armando Valente. *O computador na sociedade do conhecimento.* NIED, 1999.
- [59] Rosa Maria Vicari. *Um tutor inteligente para a programação em Lógica: Idealização, Projeto e Desenvolvimento.* Tese de Doutorado, Faculdade de Ciencias e Tecnologia, Universidade de Coimbra, Coimbra, PT, 1989.
- [60] Dinesha Samanthi Weragama. *Intelligent tutoring system for learning PHP.* Tese de Doutorado, Queensland University of Technology, 2013.
- [61] Ana Lucia Zanella. *Um Ambiente Colaborativo para Apoio a um Curso de Projeto de Software Orientado a Objetos.* Tese de Doutorado, Faculdade de Informática, Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, RS, Brasil, 1999.

MARCELA TURIM KOSCHEVIC

**ARQUITETURA PARA SISTEMAS TUTORES
INTELIGENTES QUE UTILIZA A TEORIA QUALITATIVA
DE PROCESSOS PARA RECOMENDAÇÃO DE PRÓXIMOS
MELHORES EXERCÍCIOS**

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre. Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, Universidade Federal do Paraná.

Orientador: Prof. Dr. Andrey Ricardo Pimentel

CURITIBA

2015