



Ministério da
Ciência e Tecnologia



INPE-15666-TDI/1442

UMA ARQUITETURA MULTI-AGENTE DE PLANEJAMENTO DE CONTROLE DE SATÉLITES

Adriana Carniello

Tese de Doutorado do Curso de Pós-Graduação em Computação Aplicada,
orientada pelos Drs. Maurício Gonçalves Vieira Ferreira e José Demisio Simões da
Silva, aprovada em 25 de novembro de 2008.

Registro do documento original:

<<http://urlib.net/sid.inpe.br/mtc-m18@80/2008/11.17.15.15>>

INPE
São José dos Campos
2009

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3945-6911/6923

Fax: (012) 3945-6919

E-mail: pubtc@sid.inpe.br

CONSELHO DE EDITORAÇÃO:**Presidente:**

Dr. Gerald Jean Francis Banon - Coordenação Observação da Terra (OBT)

Membros:

Dr^a Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação

Dr. Haroldo Fraga de Campos Velho - Centro de Tecnologias Especiais (CTE)

Dr^a Inez Staciarini Batista - Coordenação Ciências Espaciais e Atmosféricas (CEA)

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Dr. Ralf Gielow - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Dr. Wilson Yamaguti - Coordenação Engenharia e Tecnologia Espacial (ETE)

BIBLIOTECA DIGITAL:

Dr. Gerald Jean Francis Banon - Coordenação de Observação da Terra (OBT)

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Jefferson Andrade Ancelmo - Serviço de Informação e Documentação (SID)

Simone A. Del-Ducca Barbedo - Serviço de Informação e Documentação (SID)

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Marilúcia Santos Melo Cid - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

EDITORAÇÃO ELETRÔNICA:

Viveca Sant´Ana Lemos - Serviço de Informação e Documentação (SID)



Ministério da
Ciência e Tecnologia



INPE-15666-TDI/1442

UMA ARQUITETURA MULTI-AGENTE DE PLANEJAMENTO DE CONTROLE DE SATÉLITES

Adriana Carniello

Tese de Doutorado do Curso de Pós-Graduação em Computação Aplicada,
orientada pelos Drs. Maurício Gonçalves Vieira Ferreira e José Demisio Simões da
Silva, aprovada em 25 de novembro de 2008.

Registro do documento original:

<<http://urlib.net/sid.inpe.br/mtc-m18@80/2008/11.17.15.15>>

INPE
São José dos Campos
2009

C217a Carniello, Adriana.

Uma arquitetura multi-agente de planejamento de controle de satélites / Adriana Carniello. – São José dos Campos: INPE, 2009.

198p. ; (INPE-15666-TDI/1442)

Tese (Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 24/11/2008.

1. Inteligência artificial. 2. Sistemas multi-agentes. 3. Planejamento. 4. Engenharia de software orientada a agentes. 5. Planos de voo. I.Título.

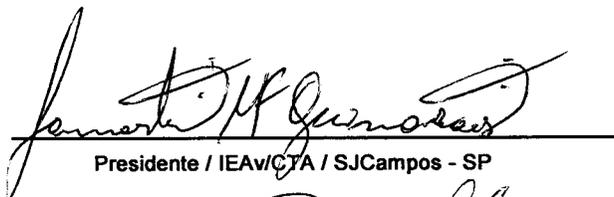
CDU 004.832

Copyright © 2009 do MCT/INPE. Nenhuma parte desta publicação pode ser reproduzida, armazenada em um sistema de recuperação, ou transmitida sob qualquer forma ou por qualquer meio, eletrônico, mecânico, fotográfico, microfílmico, reprográfico ou outros, sem a permissão escrita da Editora, com exceção de qualquer material fornecido especificamente no propósito de ser entrado e executado num sistema computacional, para o uso exclusivo do leitor da obra.

Copyright © 2009 by MCT/INPE. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, microfilming, recording or otherwise, without written permission from the Publisher, with the exception of any material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use of the reader of the work.

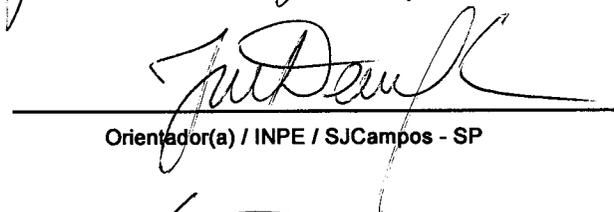
Aprovado (a) pela Banca Examinadora
em cumprimento ao requisito exigido para
obtenção do Título de Doutor(a) em
Computação Aplicada

Dr. Lamartine Nogueira Frutuoso
Guimarães



Presidente / IEAV/CTA / SJC Campos - SP

Dr. José Demisio Simões da Silva



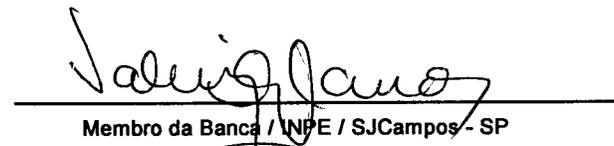
Orientador(a) / INPE / SJC Campos - SP

Dr. Mauricio Gonçalves Vieira Ferreira



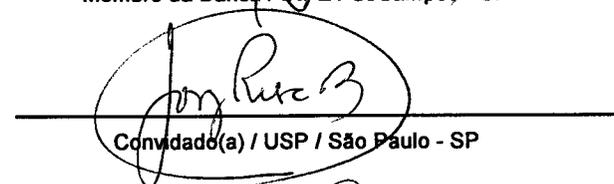
Orientador(a) / INPE / SJC Campos - SP

Dr. Valcir Orlando



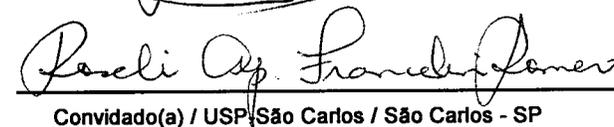
Membro da Banca / INPE / SJC Campos - SP

Dr. Jorge Luis Risco Becerra



Convidado(a) / USP / São Paulo - SP

Dra. Roseli Aparecida Francelin Romero



Convidado(a) / USP / São Carlos / São Carlos - SP

Aluno (a): Adriana Carniello

São José dos Campos, 25 de novembro de 2008

A Jesus, Deus amigo sempre presente.

A meus queridos pais e avós:

Antonio & Maria Aparecida, Ângelo & Palma,

que me dedicam tanto amor.

AGRADECIMENTOS

Agradeço a Deus por todas as graças recebidas, em especial pela graça de estar concluindo este trabalho de pesquisa com saúde e alegria.

Agradeço aos meus orientadores, prof. Dr. Maurício Gonçalves Vieira Ferreira e prof. Dr. José Demisio Simões da Silva, pelas contribuições, pelo apoio e incentivo, pela amizade e compreensão.

Agradeço ao Instituto Nacional de Pesquisas Espaciais (INPE), à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e ao Centro Federal de Educação Tecnológica de São Paulo (CEFET-SP) pela oportunidade de desenvolver este trabalho de pesquisa, pelas estruturas físicas concedidas e pelo apoio financeiro.

Agradeço, em especial, à equipe do Centro de Controle de Satélites (CCS) pelos conhecimentos fornecidos sobre o ambiente de controle de satélites do INPE.

Agradeço à minha família, em especial, a meus queridos pais e avós. Às minhas irmãs, cunhado e sobrinha também um imenso e forte “muito obrigada” pelo apoio e união, mesmo à distância.

Agradeço a cada um de meus familiares e a cada um de meus amigos. Aos amigos que conheci em Maringá, em Campinas, em São José dos Campos, em São João da Boa Vista e em Caraguatatuba, um “muito obrigada” a cada um pelo companheirismo em diferentes momentos da vida e, sobretudo, pela amizade sincera.

RESUMO

A comunidade da área espacial possui interesse na redução dos custos referentes ao controle de satélites em órbita, sobretudo diante da previsão de lançamento de novos satélites. No entanto, a redução de custos é uma meta de difícil obtenção devido a algumas características do ambiente de controle de satélites. Neste ambiente, as estações terrenas enviam operações aos satélites durante os períodos de tempo em que estes passam sobre as estações. No entanto, as passagens dos satélites sobre as estações terrenas possuem períodos de tempo restritos e, portanto, existe o risco destes períodos de tempo não serem suficientes para o envio das operações voltadas ao alcance dos objetivos primordiais dos rastreios. Além desta restrição, as estações terrenas podem controlar múltiplos satélites e, conseqüentemente, podem ocorrer conflitos de horário entre passagens de satélites distintos sobre uma mesma estação terrena. Diante destas restrições e da necessidade de reduzir os custos associados ao controle de satélites, este trabalho de pesquisa propõe uma arquitetura *Multi-Agente de Planejamento de controle de Satélites*, referenciada pela sigla MAPSat, constituída por agentes que gerenciam a alocação de recursos de solo para o rastreamento de múltiplos satélites e planejam as operações de controle destes satélites. A arquitetura MAPSat adota conceitos da área de Engenharia de Software Orientada a Agentes e utiliza a técnica de Planejamento da área de Inteligência Artificial para gerar planos de solução ao problema de controle de satélites. A implementação de um protótipo para a arquitetura MAPSat permitiu constatar que o processo de geração de planos da arquitetura confere configurabilidade à atividade de planejamento de controle de satélites. A característica de configurabilidade permite reduzir os custos associados ao controle de novos satélites, pois a atividade de planejamento se torna facilmente adaptável ao controle de múltiplos satélites, evitando, com isso, o crescimento do grupo de profissionais responsáveis pela atividade de planejamento de controle dos novos satélites.

A MULTI-AGENT SATELLITE CONTROL PLANNING ARCHITECTURE

ABSTRACT

Space Mission Control area has a strong interest in reducing the associated costs of the in-orbit satellite control process, mainly because of the prevision of new satellite launchings. Nevertheless, reducing these costs is a difficult task due to some satellite control environment features. In this environment, ground stations send operations to satellites during the periods that satellites pass over the groundstations. These time periods represent a restriction of time for sending the whole set of operations responsible for achieving the essential goals for the satellite control. Besides this restriction, ground stations can control multiple satellites and consequently there may occur time-conflicts between passes of distinct satellites over the same ground station. In order to provide a solution for managing these restrictions and achieving the goal of reducing the satellite control process cost, we propose a Multi-Agent Satellite Control Planning architecture (MAPSat) which is formed by a group of agents that allocate ground resources for multiple satellite trackings and plan these satellite control operations. MAPSat architecture adopts some concepts from the Agent Oriented Software Engineering area and employs the Automated Planning technique from the Artificial Intelligence area in order to generate plans that are solutions to the satellite control problem. The implementation of a prototype software for demonstrating MAPSat architecture made us verify that MAPSat plan generation process provides configurability to the satellite control planning activity. The configurability feature allows to reduce the costs associated to the control of new satellites as the planning activity becomes easily adaptable to the control of multiple satellites, avoiding to have to employ extra staff for the new satellite control planning activity.

SUMÁRIO

Pág.

LISTA DE FIGURAS

LISTA DE TABELAS

1	INTRODUÇÃO	21
1.1	Motivação do Trabalho de Pesquisa	23
1.2	Objetivos do Trabalho de Pesquisa	25
1.3	Metodologia de Desenvolvimento e Estrutura do Trabalho	26
2	AMBIENTE DE CONTROLE DE SATÉLITES	29
2.1	Definição de um Ambiente de Controle de Satélites	29
2.2	Telemetria e Telecomando	32
2.3	Planejamento da Operação de Controle de Satélites	35
2.4	Equipes de Controle de Missões Espaciais	37
3	PLANEJAMENTO AUTOMATIZADO DE MISSÕES ESPACIAIS	39
3.1	Motivação para a Automatização	39
3.2	Automatização em Solo	40
3.2.1	Solução SciSys em Parceria com o ESTEC/ESA	40
3.2.2	Solução do ESOC/ESA	42
3.2.3	Solução do ISTC/CNR em parceria com o ESOC/ESA	44
3.2.4	Solução do ISTC/CNR em parceria com o ESOC/ESA, NOVA Space Associates Ltd. and Scisys GmbH	47
3.3	Automatização em Bordo do Satélite	52
3.3.1	Solução de Bordo do INPE para Satélites Científicos Brasileiros	52
3.4	Análise das Soluções de Automatização de Missões Espaciais	54
4	PLANEJAMENTO EM INTELIGÊNCIA ARTIFICIAL	57
4.1	Linha do Tempo	57
4.2	Termos Fundamentais de Planejamento Automatizado	60
4.3	Planejador GRAPHPLAN	63
4.3.1	Fase de Geração do Grafo de Planejamento	64
4.3.2	Fase de Extração do Plano	67
4.4	Planejador LPG-TD	68
4.5	<i>Framework</i> de Planejamento Temporal	70
5	AGENTE DE PLANEJAMENTO	75
5.1	Definição de Agente	75
5.2	Definição de Agente de Planejamento	77
5.3	Ambientes de Planejamento	78
5.3.1	Propriedades dos Ambientes de Planejamento	78
5.3.1.1	Completamente observável versus parcialmente observável	78

5.3.1.2	Determinístico versus estocástico	79
5.3.1.3	Episódico versus seqüencial	79
5.3.1.4	Estático versus dinâmico.....	79
5.3.1.5	Discreto versus contínuo.....	79
5.3.1.6	Agente único versus multi-agente	80
5.3.2	Ambiente de Planejamento Clássico	81
5.3.3	Ambiente de Planejamento Não-Clássico.....	82
5.4	Linguagens de Representação Usadas por Agentes de Planejamento ...	83
5.4.1	Linguagem STRIPS	83
5.4.2	Linguagem PDDL	86
5.4.2.1	Estrutura da Linguagem PDDL	87
5.4.2.2	Descrição do Arquivo de Domínio.....	88
5.4.2.3	Descrição do Arquivo de Instância do Problema.....	92
5.5	Processo de Construção de Agentes de Planejamento	93
6	ENGENHARIA DE SOFTWARE ORIENTADA A AGENTES.....	95
6.1	Diretrizes para o Uso do Conceito de Agente	95
6.2	Definições de Engenharia de Software	98
6.3	Definição de Sistemas Multi-Agentes.....	99
6.3.1	Organização de Agentes	100
6.3.2	Planejamento em Sistemas Multi-Agentes	101
6.3.3	Projetos de Sistemas Multi-Agentes	103
6.4	Metodologias de Engenharia de Software Orientadas a Agentes.....	104
6.4.1	Genealogia das Metodologias Orientadas a Agentes.....	104
6.4.2	Metodologia MAS-CommonKADS	107
6.4.3	Metodologia Tropos	109
6.4.4	Metodologia MESSAGE.....	112
6.4.4.1	Conceitos da Metodologia MESSAGE	112
6.4.4.2	Visões do Modelo de Análise	115
6.5	<i>Frameworks</i> de Avaliação de Metodologias Orientadas a Agentes	116
6.6	<i>Framework</i> AMSisPlan.....	118
6.6.1	Crítérios de Avaliação de Metodologias do <i>Framework</i> AMSisPlan	119
6.7	Avaliação de Metodologias utilizando o Framework Proposto	121
7	ARQUITETURA MAPSAT	125
7.1	Estrutura e Organização de Agentes da Arquitetura MAPSat.....	125
7.2	Modelos da Arquitetura MAPSat.....	127
7.2.1	Visão Organizacional.....	127
7.2.2	Visão de Objetivo.....	133
7.2.3	Visão de Agente	136
7.2.4	Esquema do Agente Escalonador de Rastreios	139
7.2.5	Esquema do Agente Gerador de Contextos de Passagens.....	145
7.2.6	Esquema do Agente Gerador de Domínio	147
7.2.7	Esquema do Agente de Priorização de Objetivos.....	152
7.2.8	Esquema do Agente Gerador de Instâncias do Problema.....	159
7.2.9	Esquema do Agente Planejador de Operações de Vôo	162

7.3	Resultados do Protótipo da Arquitetura MAPSat	165
7.3.1	Escalonamento de Rastreios	166
7.3.2	Geração dos Arquivos de Instâncias do Problema	167
8	CONCLUSÕES	175
8.1	Contribuições	176
8.2	Trabalhos Futuros	178
	REFERÊNCIAS BIBLIOGRÁFICAS	181
	APÊNDICE A - EXEMPLO DE ARQUIVO DE DOMÍNIO	189
	APÊNDICE B - EXEMPLO ARQUIVO DE INSTÂNCIA DO PROBLEMA	195
	ANEXO A – PUBLICAÇÕES DESTE TRABALHO DE PESQUISA	197

LISTA DE FIGURAS

2.1 – Relacionamento entre um satélite, o segmento solo e os clientes	30
2.2 – Estrutura do segmento de solo.....	32
2.3 – Atividades típicas de uma passagem	36
4.3 – Representação de um grafo de planejamento.....	63
4.4 – Representação gráfica da definição de relação de exclusão mútua	65
4.5 – Componente “Planejar para Alcançar Objetivo”	71
5.1 – Exemplo de um esquema de ação na linguagem STRIPS.....	85
5.2 – Problemas de planejamento na linguagem PDDL.....	88
5.3 – Formato de um Arquivo de Domínio na linguagem PDDL.....	89
5.4 – Formato de um Arquivo de Instância do Problema em PDDL.....	92
6.1 – Genealogia de um grupo de metodologias orientadas a agentes	105
7.1 – Diagrama de relacionamentos estruturais da visão organizacional.....	128
7.2 – Diagrama de relacionamentos de interação da visão organizacional....	130
7.3 – Diagrama de decomposição de objetivos da visão de objetivo	134
7.4 – Diagrama de estrutura de delegação da visão de agente	138
7.5 – Resolução de conflitos para o problema de intersecção de passagens	142
7.6 – Redução do período de rastreamento de satélite de menor prioridade.....	167
7.7 – Arquivo de Instância do Problema para a passagem 60482-5.....	171
7.8 – Plano de Operações de Vôo (POV) para a passagem 60482-5.....	172

LISTA DE TABELAS

6.1 – Descrição dos critérios de avaliação do <i>framework</i> AMSisPlan	120
6.2 – Avaliação das metodologias usando o <i>framework</i> AMSisPlan	122
7.1 – Esquema do agente Escalonador de Rastreios	143
7.2 – Esquema do agente Gerador de Contextos de Passagens.....	146
7.3 – Esquema do agente Gerador de Domínio	149
7.4 – Descrições das situações do ambiente para as quais o agente atua....	154
7.5 – Esquema do agente de Priorização de Objetivos.....	155
7.6 – Esquema do agente Gerador de Instâncias do Problema.....	160
7.7 – Esquema do agente Planejador de Operações de Vôo	164

1 INTRODUÇÃO

O Instituto Nacional de Pesquisas Espaciais (INPE) concentra esforços no desenvolvimento tecnológico da área espacial. Dentre as atuações do INPE, destacam-se as ações de inovação tecnológica espacial, que buscam garantir o resultado sócio-econômico da atividade de pesquisa e do desenvolvimento espacial.

Assim, o INPE encontra-se em constante busca pelo desenvolvimento tecnológico da área espacial e do retorno deste desenvolvimento por meio de resultados sócio-econômicos. A inovação tecnológica espacial deve ocorrer, sobretudo, nas ações voltadas para novos programas de satélites.

Atualmente, o INPE controla em órbita os satélites SCD-1, SCD-2, CBERS-2 e CBERS-2B. Novos acordos foram estabelecidos para o desenvolvimento de novos satélites tecnologicamente mais avançados, como recentemente o acordo estabelecido com a China, que contempla a construção dos satélites CBERS-3 e CBERS-4. Estes novos satélites serão equipados com novas cargas úteis e o Brasil passará a contribuir com 50% dos recursos (nos primeiros satélites CBERS a divisão de investimentos de recursos foi de 70% para a China e 30% para o Brasil).

Em relação ao controle dos satélites brasileiros atualmente em órbita, o INPE contém em sua estrutura organizacional o Centro de Rastreamento e Controle de Satélites (CRC), responsável pelas atividades de rastreamento e controle em órbita dos satélites. O CRC é constituído pelo Centro de Controle de Satélites (CCS), localizado em São José dos Campos – SP, e pelas estações terrenas de rastreamento de Cuiabá – MT (CBA) e de Alcântara – MA (ALC). Estes três locais estão interligados por uma rede privada de comunicação de dados, que permite a troca de informações entre estas unidades.

O contato do sistema de controle de solo com um satélite é estabelecido por uma determinada estação terrena de rastreamento quando este passa sobre a região

de visibilidade da antena da estação. Durante os períodos de visibilidade do satélite, o CCS conecta-se à estação terrena de rastreo, através da rede de comunicação de dados, ficando habilitado a receber dados de telemetria do satélite e a transmitir telecomandos a este, em tempo real, além de comandar a execução de medidas de rastreo (distância e/ou velocidade).

Assim, durante o período de visibilidade de um satélite estabelece-se um canal de comunicação entre o satélite e a estação terrena. Os sinais transmitidos pelo satélite à estação terrena contêm informações de telemetria, que revelam o estado atual de funcionamento de seus subsistemas de bordo. Os sinais transmitidos pela estação terrena ao satélite, chamados de telecomandos, alteram os estados dos subsistemas do satélite em órbita, habilitam a execução de medidas de rastreo (medidas de distância e de velocidade) ou controlam a obtenção de dados desejados pelos usuários, como por exemplo, imagens e dados científicos. Por questões de confiabilidade, todas as operações a serem executadas a partir de solo durante os períodos de visibilidade de um satélite são previamente planejadas. Este planejamento de operações, realizado periodicamente, gera um documento chamado de Plano de Operações de Vôo (POV). O POV de um satélite resume todas as operações a serem executadas a partir de solo durante todas as passagens do satélite ocorridas em um determinado período.

O período de visibilidade entre um satélite e uma estação terrena recebe o nome de passagem. No entanto, uma passagem de um satélite pode não ser inteiramente rastreada por uma estação terrena. Assim, emprega-se o termo “período de rastreo” ao período de tempo de uma passagem em que há não somente visibilidade, mas também comunicação (período em que é possível o envio de telecomandos ao satélite, o recebimento de telemetrias e a execução de medidas de rastreo) entre o satélite e a estação terrena.

1.1 Motivação do Trabalho de Pesquisa

Nos satélites operados pelo INPE, classificados como de órbita baixa (os SCDs orbitam a cerca de 750 km de altitude e os CBERS a 778 km) e com duração média de órbita em cerca de 100 minutos, o tempo médio de passagem gira em torno de 10 minutos, ou seja, 10% do período orbital (KUCINSKIS, 2007). Este tempo de passagem restrito consiste em um delimitador para o processo de planejamento das operações a serem realizadas durante uma passagem.

Assim, a delimitação de tempo de uma passagem gera a necessidade de prover um mecanismo para garantir que um POV contemple todas as operações essenciais e mais relevantes durante o curto período de tempo de execução do plano.

Esta necessidade se torna ainda mais eminente quando surgem situações de conflito entre os horários das passagens de dois ou mais satélites distintos a serem rastreados por uma mesma estação terrena. Considerando a restrição de que uma estação terrena rastreia um único satélite em um determinado instante de tempo, as passagens simultâneas de dois satélites com horários que se sobrepõem passam, de fato, a competir pelo uso da estação terrena naquele horário. Desta competição, surge a necessidade de escolher a qual satélite conceder o uso exclusivo da estação terrena.

Quando esta situação de conflito ocorre, concede-se o uso exclusivo da estação terrena para o rastreamento de toda a passagem do satélite prioritário. Para isso, há a necessidade de escolher um satélite, de menor prioridade de rastreamento, que terá um trecho de sua passagem (o trecho em conflito de horário com a passagem do outro satélite) cancelado. Este cancelamento significa uma redução da janela de tempo durante a qual o satélite de menor prioridade será rastreado pela estação terrena.

Considerando o fato de que a passagem de um satélite pode sofrer uma redução no seu período de rastreamento, a necessidade de prover um mecanismo

para garantir que um POV contemple as operações essenciais durante o período de tempo de execução do plano torna-se ainda mais forte.

Assim, os fatores de complexidade do ambiente de controle de satélites a serem gerenciados podem ser resumidos em: (1) passagens com períodos de tempo restritos; e (2) existência de conflitos de horário entre passagens de satélites distintos sobre uma mesma estação terrena. Diante deste conjunto de fatores, a busca por uma técnica que facilite a geração de uma solução se faz necessária.

A técnica de Planejamento da área de Inteligência Artificial oferece recursos que permitem gerar planos de forma automatizada considerando muitas destas restrições de operabilidade do ambiente de controle de satélites. Na visão desta técnica, o ambiente de controle de satélites passa a ser tratado como um problema de planejamento, para o qual torna-se possível desvincular a definição geral do domínio do problema das instâncias do problema (problemas específicos de cada passagem de um satélite). Este modo de representação bem definida do problema de planejamento confere configurabilidade à tarefa de planejamento de controle de satélites.

O planejamento automatizado, além de conferir configurabilidade à tarefa de planejamento de operações, caminha ao encontro das necessidades emergentes com a atual expansão da quantidade de satélites a serem controlados pelo INPE. Estas necessidades são: manutenção do tamanho do Grupo de Controle de Vôo, constituído por especialistas responsáveis pelo planejamento e execução das operações de controle de satélites, frente ao aumento do número de satélites a serem controlados; e automatização do processo de execução dos POVs.

A técnica de Planejamento de IA satisfaz a estas necessidades devido à configurabilidade que oferece ao processo de planejamento, facilitando o esforço de geração de planos para vários satélites, e por prover a base para a futura automatização da execução dos planos gerados.

1.2 Objetivos do Trabalho de Pesquisa

Este trabalho de pesquisa visa, como objetivo macro, buscar um caminho de automatização para o controle de satélites do INPE frente à perspectiva de expansão do número de satélites a serem controlados pela Instituição. O caminho da automatização apresenta-se como uma alternativa promissora para a redução dos custos operacionais relacionados à manutenção de satélites em órbita uma vez que a tarefa de planejamento de controle de satélites, quando realizada segundo uma técnica adequada e de forma automatizada, se torna uma tarefa facilmente configurável ao rastreamento de diferentes satélites.

Para percorrer na direção deste objetivo macro, este trabalho de pesquisa visa definir uma arquitetura de software que permita realizar o planejamento automatizado, adotando a técnica de Planejamento de IA, das operações para o controle de satélites em órbita e dos recursos de solo para atender ao rastreamento de múltiplos satélites.

Assim, a necessidade de planejamento de operações de controle de satélites para vários satélites que compartilham recursos de solo motivou a definição de uma arquitetura de software. Esta arquitetura identifica períodos de rastreamento conflitantes e gera um plano de controle (Plano de Operações de Vôo – POV) para cada satélite no qual os conflitos são solucionados.

A definição desta arquitetura visa atingir um segundo objetivo macro, que consiste em colaborar com os anseios da comunidade de pesquisa de Planejamento em IA. Esta comunidade busca por aplicações de técnicas de planejamento em problemas do mundo real a fim de obter resultados que auxiliem o avanço da área de Planejamento no provimento de soluções aos desafios que os problemas reais apresentam.

Para percorrer na direção deste segundo objetivo macro, a arquitetura proposta visa contribuir com a área de pesquisa de Planejamento de IA por meio: da modelagem do problema de planejamento de controle de satélites utilizando o

conceito de agentes como elemento principal do projeto da arquitetura; e de um mecanismo de edição das entradas (priorização de objetivos de rastreios) para o processo de planeamento.

Um protótipo que implementa as funcionalidades da arquitetura proposta é construído para demonstrar o seu funcionamento.

1.3 Metodologia de Desenvolvimento e Estrutura do Trabalho

Este trabalho iniciou-se com o estudo de um ambiente de controle de satélites, em especial da atividade de planeamento para realizar o controle de satélites em órbita. O resultado deste estudo encontra-se descrito no Capítulo 2.

Em seguida, realizou-se um levantamento sobre o estado da arte dos trabalhos de automatização de ambientes de controle de satélites. Os conhecimentos adquiridos a partir deste levantamento e, sobretudo, a motivação em agregar valores, como configurabilidade, levaram à proposta de uma nova arquitetura de software para a automatização do processo de planeamento de controle de satélites. Os resultados deste levantamento encontram-se descritos no Capítulo 3.

O Capítulo 4 descreve alguns planeadores com a finalidade de apresentar um embasamento teórico sobre algoritmos de planeamento uma vez que a arquitetura proposta nesta tese é constituída por um planeador. Neste capítulo descreve-se também um *Framework* de Planeamento Temporal e as análises realizadas a respeito deste *framework* que culminaram em um dos aspectos diferenciais da arquitetura de planeamento automatizado proposta.

O segundo aspecto diferencial da arquitetura é relatado no Capítulo 5, que descreve o conceito de agentes. A arquitetura proposta é constituída por um grupo de agentes, dentre o qual se encontra um agente de planeamento. Assim, define-se o conceito de agente de planeamento e a linguagem de

representação adotada por este agente para descrever o problema de planejamento de controle de satélites.

O Capítulo 6 descreve os motivos pelos quais se concebeu a arquitetura proposta utilizando o conceito de agentes e o conjunto de critérios utilizado para a escolha de uma metodologia de engenharia de software orientada a agentes adequada à modelagem da arquitetura proposta. Este conjunto de critérios utilizado constitui um *framework* de avaliação de metodologias orientadas a agentes, proposto neste trabalho de pesquisa, voltado para a análise comparativa de metodologias utilizadas para a modelagem de sistemas de planejamento.

No Capítulo 7 apresentam-se os modelos, da metodologia orientada a agentes escolhida, construídos para o projeto da arquitetura proposta. Durante o detalhamento das características dos agentes que compõem a arquitetura, os comportamentos dos agentes são exemplificados. O mecanismo de priorização de objetivos de rastreamento, proposto neste trabalho, define o comportamento de um dos agentes da arquitetura.

Por fim, apresentam-se os resultados obtidos com a implementação de um protótipo da arquitetura MAPSat a fim de demonstrar o seu funcionamento.

O Capítulo 8 apresenta as conclusões, incluindo as contribuições deste trabalho e os trabalhos futuros vislumbrados.

2 AMBIENTE DE CONTROLE DE SATÉLITES

2.1 Definição de um Ambiente de Controle de Satélites

Todo satélite tem como objetivo realizar uma missão espacial, que consiste em desempenhar uma função específica da qual espera-se um determinado conjunto de produtos. Uma missão espacial pode ter ênfase, por exemplo, em telecomunicações, em sensoriamento remoto, em coleta de dados e em experimentos científicos e tecnológicos (ECSS-E-70 PART 1, 2000) (LARSON e WERTZ, 1999).

Por exemplo, uma missão espacial com ênfase em coleta de dados ambientais tem como função receber e retransmitir para a Terra os sinais emitidos por plataformas de coleta de dados espalhadas em um território. O produto esperado desse tipo de missão consiste nas informações ambientais coletadas pelas plataformas. Uma missão espacial com ênfase em sensoriamento remoto tem como produto esperado um conjunto de imagens de solo obtidas a partir do espaço. Uma missão espacial com ênfase em telecomunicações tem como produto esperado a retransmissão de dados e televisão direta.

Portanto, um satélite é desenvolvido para atender aos requisitos de uma missão espacial e é constituído por dois elementos: carga útil e plataforma de serviço. Estes elementos são descritos por Kucinskis (2007) conforme apresentado a seguir:

- A carga útil é um conjunto de equipamentos dedicados à aplicação da missão espacial, ou seja, ao motivo pelo qual o satélite foi lançado. A carga útil é responsável por gerar os dados da missão. Em um satélite de sensoriamento remoto, por exemplo, a carga útil é composta por câmeras e outros sensores. Em um satélite científico, por experimentos científicos e tecnológicos;

- A plataforma de serviço é um conjunto de subsistemas/equipamentos projetados para sustentar a operação da missão em órbita, ou seja, a operação das cargas úteis. É responsável pelo monitoramento, operação e manutenção do satélite em órbita, e pela comunicação com as estações terrenas de rastreamento e com a carga útil.

Para que um satélite possa cumprir sua missão faz-se necessário oferecer suporte em solo às suas atividades. Para atender esta necessidade, existe um segmento solo, constituído por recursos humanos, infra-estruturas de hardware e de software, e procedimentos sistematizados, que atua na preparação e execução da operação de controle de satélites. O satélite constitui o chamado segmento espacial. O segmento solo estabelece o elo entre um satélite e os clientes finais da missão, ou seja, os usuários que possuem interesse nos dados de carga útil transmitidos pelo satélite. A Figura 2.1 ilustra o relacionamento entre um satélite genérico, o segmento solo (que oferece suporte operacional ao satélite) e os clientes da missão.

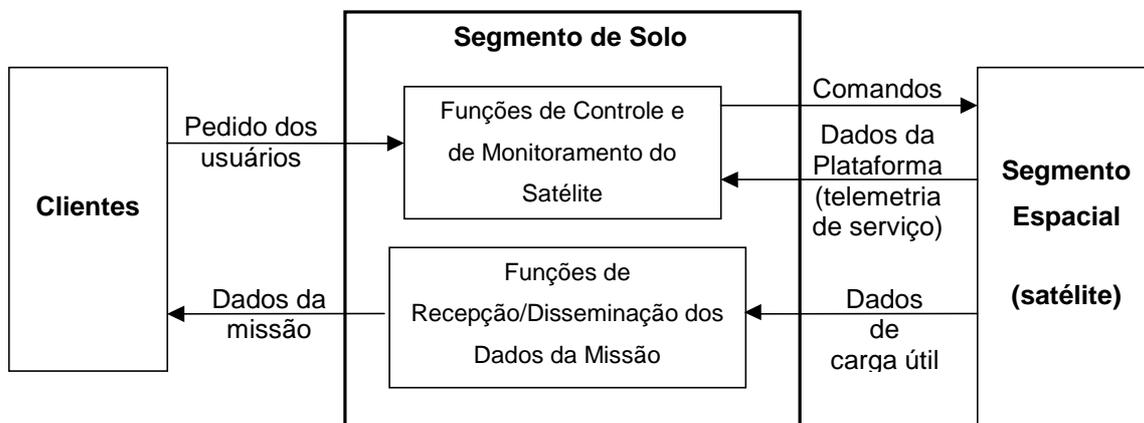


Figura 2.1 – Relacionamento entre um satélite, o segmento solo e os clientes

Fonte: Adaptada de Larson e Wertz (1999)

O segmento solo é composto de dois componentes principais: (1) a organização das operações de solo; e (2) os sistemas de solo. A organização das operações de solo compreende os recursos humanos responsáveis pela preparação dos dados de operação de uma missão e pela execução das

tarefas de operação de controle do satélite. Os sistemas de solo consistem em elementos de solo (hardware, software e instalações) que oferecem suporte ao controle e à monitoração do satélite e à exploração dos dados da missão. Os principais elementos do sistema de solo são:

- Sistema de Controle de Missão (SCM) – logicamente decomposto nos seguintes elementos funcionais: Sistema de Controle de Operações (SCO), Sistema de Controle de Carga Útil (SCCU) e Sistema de Exploração de Missão (SEM). Os dois primeiros são responsáveis por planejar, monitorar e controlar a plataforma e as cargas úteis, respectivamente. O último sistema é responsável por distribuir aos usuários os produtos da missão e dados extras necessários para o planejamento e a utilização dos dados de missão;

- Sistema de Estação Terrena (SET) – constitui a interface direta com o satélite. Esse sistema implementa o link de comunicação com o satélite durante os períodos em que este passa sobre a estação. Durante esses períodos são transmitidos comandos ao satélite (telecomandos) e recebidos dados da plataforma e das cargas úteis (telemetrias). O sistema SET pode ser classificado em duas instâncias lógicas: (1) Estação Terrena utilizada para o controle do satélite (ET-SE), que provê serviços de telemetria, telecomando e de rastreamento tanto para a plataforma como para a carga útil; e (2) Estação Terrena que suporta a exploração da missão (ET-EM), que provê serviços de recepção dos dados de carga útil tais como o recebimento de sinais de telecomunicações, de dados científicos e de imagens da Terra;

- Rede de comunicação de dados (NET) – responsável pela conexão entre todos os sistemas de solo permitindo a comunicação entre eles e a comunicação com os usuários da missão.

A Figura 2.2 apresenta a estrutura do segmento de solo com os principais elementos de solo descritos anteriormente. A decomposição apresentada é de natureza lógica e na prática, estas funções podem ser fisicamente agrupadas de acordo com o tipo da missão e da organização do segmento de solo.

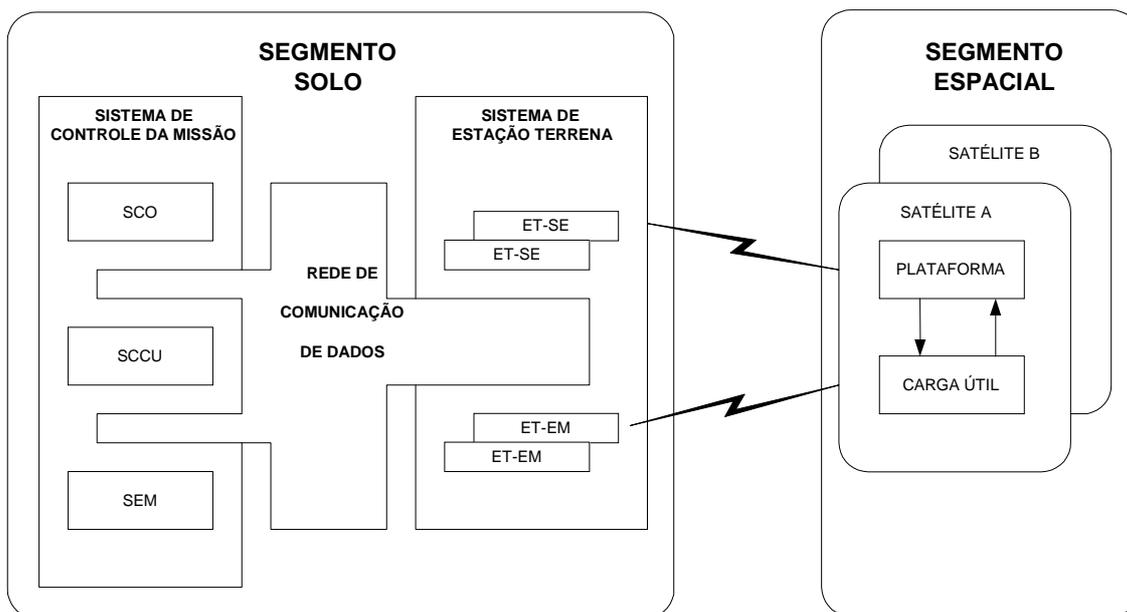


Figura 2.2 – Estrutura do segmento de solo

Fonte: Adaptada de ECSS-E-70 Part 1 (2000)

2.2 Telemetria e Telecomando

Kucinskis (2007) apresenta em seu documento de pesquisa definições sobre os sistemas de telemetrias e de telecomandos utilizadas nesta Seção. Portanto, conforme apresentado em Kucinskis (2007), o monitoramento, o controle e a comunicação com um satélite são realizados por meio dos sistemas de telemetrias e de telecomandos.

Um sistema de telemetrias tem como propósito transportar, de forma transparente e confiável, informações de medição de uma origem de dados remota para usuários distantes desta origem (CCSDS, 1987-a). Um sistema de telecomandos deve transportar informações de controle de um ponto de origem (como um operador humano) para um equipamento remotamente localizado (CCSDS, 1987-b). As unidades de telemetria e de telecomandos são chamadas de pacotes ou mensagens.

Os pacotes de telemetrias são enviados do satélite para a estação terrena. Eles podem conter: (1) dados sobre o estado do satélite, como temperatura, níveis de tensão elétrica, situação de seus subsistemas (se estão ligados ou não, seus modos de operação e parâmetros de funcionamento atuais); ou (2) dados de carga útil, como dados brutos de imagens, valores lidos por sensores de experimentos, etc. Pacotes de telemetrias carregam ainda relatos sobre o sucesso ou falha na execução de comandos, mensagens de erro e alertas sobre anomalias detectadas.

Existem dois tipos de telemetria: a telemetria de tempo real e a telemetria armazenada. A telemetria de tempo real é utilizada para monitorar, durante o período da passagem, o status dos principais subsistemas do satélite, como a bateria e o computador de bordo. Assim, caso seja detectada qualquer anomalia, comandos podem ser enviados para tentar corrigi-la durante a passagem, caso haja tempo hábil.

A telemetria armazenada pode conter dados sobre os estados dos equipamentos, relatos da execução de comandos, alertas, mensagens de erro e dados de carga útil. Estes dados são gerados durante a órbita e armazenados para envio em passagens futuras.

Os pacotes de telecomandos são enviados da estação terrena para o satélite. Eles contêm comandos gerados pela Equipe de Operação que serão executados pelo satélite. Estes comandos podem ser para ligar ou desligar equipamentos, mudar modos de operação de subsistemas, carregar novos parâmetros de funcionamento, etc. Os pacotes de telecomandos enviados durante uma passagem podem ser telecomandos de tempo real ou telecomandos temporizados, que geralmente são agrupados em arquivos.

Os telecomandos de tempo real consistem nos telecomandos executados em tempo real durante a passagem de um satélite sobre uma estação terrena, ou seja, eles devem ser executados pelo satélite assim que recebidos da estação terrena. Exemplos destes telecomandos são: a sincronização do relógio interno

do satélite com o de solo, o carregamento de novos programas, ou comandos emergenciais para a correção de anomalias ou recuperação de modos de erro.

Os telecomandos temporizados são telecomandos armazenados na memória de bordo do satélite e executados fora de passagem. Eles contêm um campo com uma referência temporal, indicando ao software do satélite quando eles devem ser executados. Ao serem recebidos, eles são armazenados numa estrutura de dados chamada fila de telecomandos temporizados. Esta fila é verificada regularmente e os comandos que ela contém são encaminhados para execução nos seus devidos momentos. O conjunto de telecomandos temporizados armazenados nesta fila compõe o plano de operações corrente do satélite.

A um conjunto de telecomandos temporizados voltados para a carga útil do satélite, ou seja, para a obtenção dos dados que constituem o objetivo da missão espacial, dá-se o nome de Plano de *Payload* (carga útil) (ELDER e PAYNE, 2004). A elaboração de um Plano de *Payload* para um satélite deve buscar atender às requisições dos diferentes usuários da missão. Por exemplo, um Plano de *Payload* para um satélite de sensoriamento remoto deve procurar obedecer às requisições de captura de imagens dos vários usuários da missão espacial.

Um satélite pode somente transmitir telemetrias e receber telecomandos durante o período em que há visibilidade entre sua antena e a antena da estação terrena, ou seja, no período em que o satélite se encontra posicionado sobre a estação terrena e, em decorrência deste posicionamento, torna-se possível a comunicação entre as antenas de solo e de bordo. A este período de visibilidade dá-se o nome de “passagem” do satélite (KUCINSKIS, 2007).

No entanto, durante a passagem de um satélite sobre uma estação terrena de rastreamento pode não haver comunicação com esta durante um intervalo de tempo da passagem. Assim, emprega-se o termo “período de rastreamento” ao período de tempo de uma passagem em que há não somente visibilidade, mas também

comunicação (envio de telemetrias e recebimento de telecomandos) entre o satélite e a estação terrena. O tempo de passagem de um satélite sobre uma estação terrena varia de uma órbita para outra, podendo não ocorrer em alguns casos.

Como já mencionado anteriormente, nos satélites operados pelo INPE, classificados como de órbita baixa (os SCDs orbitam a cerca de 750 km de altitude e os CBERS a 778 km) e com duração média de órbita em cerca de 100 minutos, o tempo médio de passagem gira em torno de 10 minutos, ou seja, 10% do período orbital (KUCINSKIS, 2007). Este tempo de passagem restrito consiste em um delimitador para o processo de planejamento das ações a serem realizadas durante uma passagem. Estas ações consistem na transmissão de telecomandos e na recepção de telemetrias pela estação terrena.

2.3 Planejamento da Operação de Controle de Satélites

Além de preparar fisicamente um satélite e o segmento solo responsável por oferecer suporte às suas atividades, faz-se necessário também planejar a operação de controle do satélite a fim de que a missão espacial obtenha sucesso.

A operação de controle de satélites é realizada por meio de dois tipos de operações: as operações de vôo e as operações de solo. Estes dois tipos de operações originam, respectivamente, os seguintes planos de controle: o Plano de Operações de Vôo (POV) e o Plano de Operações de Solo (POS) (ECSS-E-70 PART 1, 2000).

As operações de vôo de um satélite correspondem a todas as atividades relacionadas ao planejamento, à execução e à avaliação do controle do satélite quando em órbita. Já as operações de solo correspondem a todas as atividades relacionadas ao planejamento, à execução e à avaliação do controle

das facilidades de solo de suporte, tais como as estações terrenas e a rede de comunicação de dados de solo (ECSS-E-70 PART 1, 2000).

A Figura 2.3 apresenta as ações típicas de uma passagem, que podem ser classificadas de acordo com três fases: pré-passagem, passagem e pós-passagem.

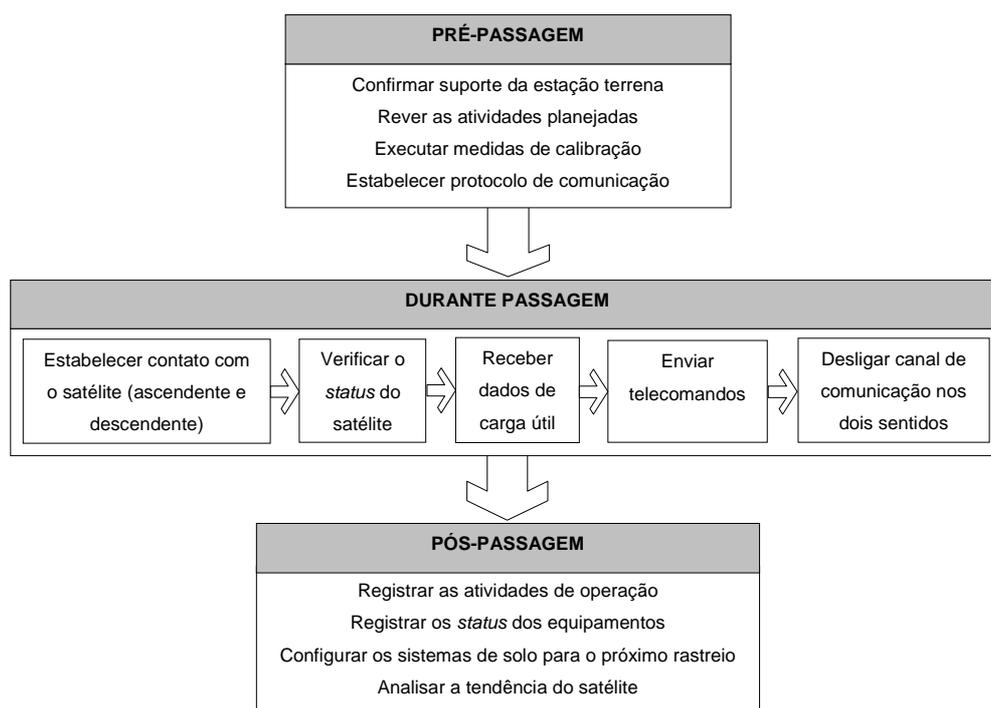


Figura 2.3 – Atividades típicas de uma passagem

Fonte: Adaptada de Larson e Wertz (1999)

Na fase de pré-passagem deve-se confirmar o suporte da estação terrena, rever as atividades planejadas, executar medidas de calibração de equipamentos da estação terrena a fim de ajustá-los com precisão e estabelecer o protocolo de comunicação com o satélite. Assim, para cada passagem de um satélite configuram-se os sistemas de solo de forma a suportar as necessidades de transmissão de telecomandos e de recepção de telemetrias da próxima fase.

Na fase subsequente, a fase de passagem do satélite, ou seja, quando este torna-se visível à estação terrena, deve-se primeiramente estabelecer contato de telecomunicação descendente (*downlink*) e, em seguida, estabelecer o contato ascendente (*uplink*). Ao final da passagem, a ligação de telecomunicação é perdida e o canal de comunicação deve ser desligado em ambos os sentidos.

Nesta fase recebem-se os pacotes de telemetrias e enviam-se os pacotes de telecomandos. Pacotes de telecomandos de tempo real e temporizados são enviados respeitando a lista dos telecomandos aceitos para o satélite considerado.

O Plano de Operações Solo (POS), em contraste com o POV, é um plano que orienta a operação das facilidades de solo para oferecer suporte ao controle de satélites em órbita. Pode-se ter um único POS para vários satélites que utilizam os mesmos recursos de solo, ou seja, que compartilham uma mesma estação terrena e rede de comunicação de dados (ECSS-E-70 PART 1, 2000).

2.4 Equipes de Controle de Missões Espaciais

Existem duas diferentes equipes para controlar uma missão espacial – a Equipe de Operação e a Equipe de Suporte.

A Equipe de Operação é estruturada da seguinte forma (ECSS-E-70 PART 1, 2000):

- Grupo de Controle de Vôo – responsável pelo controle da missão e do satélite;
- Grupo de Dinâmica de Vôo – responsável pela determinação e previsão de órbita e de atitude, cálculo e avaliação de manobras, e previsão de eventos orbitais e de dados de apontamento da antena da estação terrena. Este grupo oferece suporte à equipe de Controle de Vôo;

- Grupo de Operações de Solo – responsável por operar e manter as facilidades de suporte de solo, tais como as estações terrenas, as redes de comunicação de dados e o centro de controle de missão;
- Grupo de Exploração de Missão – responsável pela interface com os usuários de uma missão, arquivando os produtos e distribuindo os dados de carga útil.

A Equipe de Suporte, por sua vez, é constituída pelos seguintes grupos:

- Grupo de Suporte de Solo – composto por especialistas dos sistemas de solo que fornecem suporte às equipes de operações de solo;
- Grupo de Suporte de Projeto – composto por especialistas do satélite que fornecem suporte à equipe de controle de vôo.

As equipes apresentadas consistem em recursos humanos responsáveis pela preparação e execução das várias tarefas de operação de uma missão. No ambiente de controle de missões espaciais é ideal que essas equipes sejam subsidiadas por infra-estruturas de hardware e de software a fim de contornar a complexidade do ambiente e também reduzir os custos associados à sua operação.

Diante das atuais necessidades de redução de custos e de melhorias qualitativas na infra-estrutura de software do ambiente de controle de missões espaciais, observa-se um crescente interesse pela implantação de softwares que automatizem o processo de controle de missões espaciais. Assim, no Capítulo 3, a seguir, comentam-se trabalhos de pesquisadores e especialistas da área espacial voltados para essa automatização.

3 PLANEJAMENTO AUTOMATIZADO DE MISSÕES ESPACIAIS

O planejamento automatizado de missões espaciais envolve a geração de uma seqüência de ações a partir de objetivos de alto nível. Esta seqüência de ações codifica as restrições de operabilidade relacionadas às missões espaciais. O planejamento automatizado, aliado à execução automatizada das ações contidas nos planos gerados, permite manter o tamanho da equipe de Operação frente a um aumento do número de satélites a serem controlados pelo segmento de solo e, conseqüentemente, diminuir o custo de manutenção associado ao aumento do número de satélites em órbita a serem simultaneamente controlados.

3.1 Motivação para a Automatização

A preocupação atual consiste em encontrar soluções de automatização para diminuir os custos relacionados ao controle de missões espaciais e em oferecer maior configurabilidade às tarefas do ambiente de controle de missões espaciais para vários satélites.

Este ambiente contém fortes restrições de tempo e de recursos. Em relação às restrições de tempo, um satélite apresenta, por exemplo, um tempo de órbita e um período de visibilidade em relação a uma estação terrena. Em relação às restrições de recursos, existem restrições de orçamento para novos projetos e para a manutenção de projetos existentes, restrições de consumo de energia, uma quantidade limitada de estações terrenas para a captura de sinais, restrições de recursos humanos operacionais, dentre outras restrições.

Nas próximas seções são analisados trabalhos de pesquisa de automatização do ambiente de controle de missões espaciais. Alguns destes trabalhos empregam a técnica de Planejamento da área de Inteligência Artificial. Esta técnica, a ser detalhada no Capítulo 4, lida com as restrições de tempo e de recursos inerentes ao ambiente considerado e permite que sejam gerados

planos de operações, chamadas de ações de planejamento no contexto da área de IA, para o controle de missões espaciais.

Nos trabalhos analisados, a automatização do controle de missões espaciais é implementada investindo-se: (1) na automatização do segmento solo; ou (2) na autonomia do satélite (automatização de bordo), desenvolvendo-se sistemas de controle de bordo inteligentes com o objetivo de diminuir as tarefas de solo.

A Seção 3.2 descreve um conjunto de trabalhos voltados para a automatização do segmento de solo, nos quais os sistemas de automatização são executados em solo, ao passo que a Seção 3.3 apresenta trabalhos voltados para a automatização de bordo. Estes tipos de automatização constituem linhas de pesquisa distintas pelas características inerentes a cada ambiente de operação, mas complementares para o processo de automatização do ambiente de controle de missões espaciais como um todo.

3.2 Automatização em Solo

3.2.1 Solução SciSys em Parceria com o ESTEC/ESA

A empresa SciSys da Inglaterra em parceria com o ESTEC (*European Space Research & Technology Centre*) da ESA (*European Space Agency*) propõem um conjunto de ferramentas de gerenciamento de missões espaciais para o planejamento de missões e a execução automática de planos. Este conjunto de ferramentas, descrito em Harris et al. (2002), gera planos para satélites que são visíveis às estações terrenas por um tempo limitado e, por isso, precisam fazer o maior uso possível da estação terrena durante esse tempo.

Os componentes do sistema de gerenciamento de missões encontram-se descritos a seguir:

- Geração de Visibilidade – este componente baseia-se no conjunto de visibilidades. Uma visibilidade define quando uma estação terrena pode estabelecer contato com um determinado satélite da constelação;

- Planejador de Contatos – este componente gera um plano de contatos. Um contato consiste no período de tempo em que um satélite pode ser rastreado por uma estação terrena. O plano de contatos é gerado a partir das seguintes informações: (1) as visibilidades geradas da rede de estações terrenas para a constelação de satélites; (2) o conjunto de requisitos de contatos definidos pelos usuários; e (3) os tipos de contatos que as estações terrenas oferecem;
- Um requisito de contato pode ser do tipo *standing* ou *one-off*. O tipo *standing* significa uma demanda regular por contato, como por exemplo, o requisito de contato: “deve haver um contato para um determinado satélite no máximo a cada x segundos”. O tipo *one-off* significa uma demanda por contato única, como por exemplo: “deve haver no mínimo x segundos de contato entre o tempo de início de passagem de um satélite e o seu tempo de fim de passagem”;
- Um plano de contatos é criado relacionando-se as visibilidades aos requisitos de contatos dos usuários e às restrições de tipo de contato que as estações terrenas oferecem;
- Planejamento de Missões – esta parte efetua o planejamento das operações de um satélite, importando o plano de contatos. Após o processo de detecção e resolução de conflitos ter resolvido todos os conflitos, gera-se um plano executável. Esse plano é passado para o processo executor de planos;
- Execução de Planos – este processo tem como entrada um plano executável. Ele executa o plano no tempo correto, passando os comandos e os procedimentos (operações de bordo) para o processo de Execução de Procedimentos;
- Execução de Procedimentos – inicia um comando e a transmissão de uma fila de comandos (operações de bordo);

- Transmissor de Filas de Comandos – este é um processo específico para cada missão, que carrega os comandos de bordo no satélite. O transmissor de filas de comandos é uma função iniciada pelo processo de Execução de Procedimentos.

3.2.2 Solução do ESOC/ESA

O Centro de Operações Espaciais Europeu (ESOC - *European Space Operations Centre*) da ESA também propõe uma arquitetura conceitual de sistemas que oferece suporte à automação de operações de missões. Os papéis dos diferentes sistemas presentes nesta arquitetura, descrita em Ferri et al. (2004), são apresentados abaixo:

- Sistema de Preparação de Operações (OPS): oferece suporte ao controle *offline* de validação e configuração dos produtos envolvidos no processo de operação de satélites, principalmente os procedimentos e as bases de dados operacionais.
- Sistema de Planejamento de Missões (MPS): oferece suporte ao planejamento e ao escalonamento *offline* de todas as operações de missões, considerando as restrições de disponibilidade de recursos espaciais e de solo. A saída do MPS é um conjunto de planos escalonados livres de conflito para serem executados em solo ou em bordo.
- Sistema de Gerenciamento EXTRACT (EMS): oferece suporte ao gerenciamento centralizado dos recursos de solo multi-missão, incluindo as estações terrenas e as redes de comunicação. Esse sistema interage, durante a fase de planejamento, com os Sistemas de Planejamento de Missões específicos de cada missão. Durante a fase de execução, o EMS interage com os sistemas de solo que oferecem suporte à execução automatizada de operações, que constituem o Sistema de Automação de Missões e no Sistema de Computador de Estações.

- Sistema de Interface de Redes (NIS): oferece suporte à troca de telemetrias, de telecomandos e de mensagens administrativas com o equipamento final das estações terrenas.
- Sistema de Controle de Missões (MCS): oferece suporte às operações de controle e de monitoramento de operações. Este sistema troca dados operacionais, primeiramente dados de telemetria e de telecomando, com a estação terrena por meio do NIS. O MCS recebe do Sistema de Planejamento de Missões um Escalonamento de Operações de Bordo e o envia para o satélite para ser executado posteriormente.
- Sistema de Dinâmica de Vôo (FDS): oferece suporte à determinação, ao monitoramento, à previsão e ao controle de atitudes e órbitas de satélites.
- Computador de Estações (STC): oferece suporte ao controle e ao monitoramento remoto dos equipamentos das estações terrenas, baseando-se na execução automática dos escalonamentos recebidos do EMS.
- Sistema de Automação de Missões (MAS): oferece suporte à execução automatizada de procedimentos de controle. Este sistema acessa os serviços externos do MCS e do NIS: recebe do Sistema de Planejamento de Missões (MPS) um Escalonamento de Operações de Solo executável; e utiliza as definições de procedimentos operacionais produzidas pelo Sistema *offline* de Preparação de Operações (OPS).

Esta é uma arquitetura conceitual de automação do segmento de solo. Segundo os autores, o objetivo da automação não consiste em eliminar a necessidade do trabalho humano, mas sim em substituir as funções simples e repetitivas executadas manualmente por ferramentas automatizadas, deixando as tarefas mais complexas para os homens.

3.2.3 Solução do ISTC/CNR em parceria com o ESOC/ESA

O Instituto de Tecnologia e Ciência Cognitiva do Conselho Nacional de Pesquisas Italiano (ISTC/CNR – *Institute for Cognitive Science and Technology/National Research Council of Italy*) em parceria com o ESOC da ESA desenvolveram uma arquitetura denominada MEXAR2 – *Mars-EXpress Scheduling ARchitecture*. Esta arquitetura é utilizada na missão da sonda orbital *Mars-Express* desde fevereiro de 2005.

MEXAR2 foi proposta para tratar o problema de transmissão de dados para a Terra. Em uma missão de espaço profundo como a Mars Express, a espaçonave produz uma grande quantidade de dados de carga útil e de dados de controle do funcionamento dos subsistemas da espaçonave. Esta grande quantidade de dados deve ser enviada para a Terra durante janelas de tempo limitadas e pré-determinadas. Além da restrição de tempo para o envio de dados a Terra, a missão Mars Express possui o fator limitante de um sistema de apontamento único: ou a espaçonave aponta para Marte, para executar as suas operações de carga útil, ou para a Terra, para enviar os dados previamente obtidos. Conseqüentemente, requer-se que todos os tipos de dados sejam primeiramente armazenados em uma memória de bordo para serem posteriormente enviados.

Portanto, o problema principal que a arquitetura MEXAR2 soluciona consiste em gerar planos que contêm seqüências de operações para o envio de dados da memória de bordo da espaçonave Mars Express às estações terrenas durante as janelas de tempo pré-determinadas. Ressalta-se que quando não se consegue enviar a Terra a quantidade total de dados armazenados na memória de bordo, estes dados são sobrescritos para o armazenamento de novos dados, sendo necessário agendar novamente a atividade (experimento científico) quando se deseja obter dados específicos.

Para representar o problema de planejamento levantaram-se as entidades recurso e atividade. Os recursos representam subsistemas do domínio capazes

de oferecer serviços enquanto as atividades modelam tarefas que, ao serem executadas, consomem recursos ao longo do tempo.

Em relação aos recursos, identificaram-se dois tipos de recursos: os depósitos de dados e o canal de transmissão. Um depósito de dados representa uma subdivisão existente na memória de bordo, que apresenta uma capacidade de armazenamento fixa e um nível de prioridade. Um depósito de dados pode ser visto como um arquivo de tamanho máximo gerenciado de forma cíclica, ou seja, os dados previamente armazenados são sobrescritos (são perdidos) caso a nova quantidade de dados a ser armazenada exceda a capacidade limite do depósito de dados. O canal de transmissão, por sua vez, representa as conexões com a Terra disponíveis para transmissão de dados.

Em relação às atividades, cada uma delas possui um tempo de duração fixo, e um horário inicial e final. Dois tipos de atividades foram definidos: as operações de armazenamento e as operações de transmissão de dados. Uma operação de armazenamento armazena em um horário final x uma quantidade de dados y em um depósito de dados z . Este tipo de operação lida com dois tipos de dados: dados de carga útil ou dados de controle dos subsistemas da espaçonave. Esta diferenciação quanto ao tipo de dado gerenciado deve-se ao fato destes tipos de dados terem níveis de prioridade diferentes para os planejadores de missões, sendo, portanto, armazenados em depósitos de dados separados. Uma operação de transmissão de dados transmite, durante um intervalo de tempo i , uma quantidade de dados y armazenada no depósito de dados z para uma estação terrena específica.

Para oferecer soluções ao problema de planejamento apresentado, a ferramenta MEXAR2 apresenta uma arquitetura bimodular composta por dois módulos: o módulo Solucionador de Problemas e o módulo de Interação. O módulo Solucionador de Problemas responsabiliza-se por modelar o problema de planejamento e por prover planos de solução. O módulo de Interação responsabiliza-se pelo diálogo com o usuário, permitindo com que este se

informe e entenda as tarefas executadas pelo módulo Solucionador de Problemas, eleve o seu nível de confiabilidade em relação à atividade de geração de soluções automatizada e intervenha durante a resolução de problemas.

Para a modelagem do problema de planejamento, o módulo Solucionador de Problemas utiliza uma abordagem chamada de síntese na linha do tempo, também utilizada em outros trabalhos da literatura de Planejamento de IA, como em Jonsson et al. (2000), Chien et al. (2002), e Frank e Jonsson (2003). Empregando esta abordagem de modelagem, os autores sintetizam um modelo para os aspectos do problema considerados relevantes.

Em particular, os autores consideram a evolução temporal de um conjunto de depósitos de dados e o canal de transmissão. Para prover soluções às restrições dos problemas de capacidade dos depósitos de dados e da largura de banda do canal de transmissão, o horizonte temporal de planejamento é subdividido em intervalos contínuos e define-se que as operações de armazenamento devem ocorrer somente ao final destes intervalos. Para os depósitos de dados de cada intervalo são associadas variáveis de decisão que representam o volume de dados que poderá ser enviado a Terra por meio do canal de transmissão. O problema de planejamento é restringido de forma a considerar os perfis de recursos para cada depósito de dados e a disponibilidade do canal de transmissão para envio dos dados.

A segunda funcionalidade do módulo Solucionador de Problemas, que consiste em prover planos de solução, utiliza as abordagens de Problema de Satisfação de Restrições (CSP – *Constraint Satisfaction Problem*) e de Redução de Fluxo Máximo. Detalhes sobre estas abordagens podem ser obtidos em Oddi et al. (2003), Oddi e Policella (2004) e Oddi et al. (2005). O algoritmo deste módulo gera planos de solução em duas etapas. A primeira etapa tem como objetivo definir a quantidade de dados de cada pacote de dados a ser enviada a Terra durante cada janela de tempo do canal de comunicação. Na segunda etapa

definem-se, com base na alocação produzida na etapa anterior, os comandos de envio dos pacotes de dados alocados. Estes pacotes de dados são alocados nas janelas de tempo de forma não ordenada. A ordenação dos pacotes de dados, a serem enviados a Terra, considera os respectivos níveis de prioridade dos pacotes.

3.2.4 Solução do ISTC/CNR em parceria com o ESOC/ESA, NOVA Space Associates Ltd. and Scisys GmbH

A implementação da arquitetura Mexar2 e o seu uso operacional satisfatório desde 2005, oferecendo soluções ao problema de envio de dados a Terra da missão *Mars Express*, incentivaram a mesma equipe de desenvolvimento de MEXAR2 a estudar e desenvolver soluções para outro tipo de problema presente na mesma missão. Deste estudo desenvolveu-se uma nova ferramenta denominada RAXEM, descrita em Rabenau et al. (2008), em uso operacional desde o ano de 2007. O nome RAXEM originou-se de um trocadilho feito pela equipe PST do ISTC/CNR, os principais autores da ferramenta, e consiste no nome MEXAR escrito ao contrário.

A ferramenta RAXEM gera planos de solução para o problema de envio de telecomandos da estação terrena à sonda orbital Mars Express, que opera segundo um conceito de pré-armazenamento: primeiramente a espaçonave Mars Express armazena os dados na memória de bordo, tanto os dados recebidos como os dados a serem enviados, e depois trabalha com estes dados, liberando-os para execução (no caso de recebimento de telecomandos) ou enviando-os a Terra (no caso de dados de telemetria).

Os telecomandos, a serem enviados à Mars Express, são: gerados em solo, temporizados, ou seja, vincula-se a cada telecomando um horário de execução, e enviados à espaçonave durante uma passagem desta sobre uma estação terrena de rastreio. Os telecomandos enviados à espaçonave são armazenados na memória de bordo de forma ordenada pelos seus respectivos horários de execução e liberados para execução de acordo com estes horários.

O problema de envio de telecomandos à Mars Express apresenta as seguintes características:

- os telecomandos temporizados devem ser enviados de uma estação terrena à espaçonave Mars Express com margem de tempo suficiente para garantir que o armazenamento e a execução destes telecomandos ocorram de forma coerente e segura;
- as passagens com comunicação com a estação terrena ocorrem somente quando a antena da espaçonave aponta para a Terra. Quando a antena não aponta para a Terra, a passagem oferece suporte somente à obtenção de dados científicos ou de dados de manutenção. Portanto, as operações científicas e as operações de envio de telecomandos à espaçonave são operações que competem entre si. Devido a este conflito, pode ocorrer de não haver passagens com disponibilidade para envio de telecomandos por várias horas consecutivas, quando o apontamento da antena favorecer a execução de operações científicas. As estações terrenas alocadas para o rastreamento da espaçonave Mars Express pertencem: à rede de rastreios da ESA, a rede ESTRACK; e à rede de espaço profundo da NASA, a rede DSN. Cada uma destas redes encontra-se alocada para a missão Mars Express durante o total de oito horas por dia. No entanto, esta quantidade de horas pode variar, pois as estações terrenas de ambas as redes possuem uso compartilhado com outras missões espaciais;
- durante o começo da fase de rotina da missão espacial algumas restrições devem ser observadas para o envio de telecomandos a fim de preservar a integridade e a segurança da espaçonave. Estas restrições são: a espaçonave encontra-se em modo de apontamento para a Terra, o transmissor da espaçonave encontra-se ligado a fim de garantir a comunicação com a estação terrena (exceto em eclipse quando o transmissor deve estar desligado para economizar energia), não pode

estar ocorrendo nenhum envio de telemetrias a Terra e todos os instrumentos devem se encontrar desligados;

- A memória de bordo para o armazenamento de telecomandos, chamada de *Master Timeline Buffer* (MTL), armazena uma quantidade limitada de telecomandos, aproximadamente três mil;
- O Grupo de Planejamento da Missão Mars Express realiza o planejamento do conjunto de telecomandos a ser enviado e da ordem de envio destes para o período de uma semana. O processo de planejamento gera um arquivo chamado de *MTL Detailed Agenda Files* (MDAF), que geralmente contém uma grande quantidade de telecomandos. Este arquivo é dividido em segmentos, respeitando as restrições citadas anteriormente. O tamanho de cada segmento deve obedecer a dois critérios: ao tamanho da janela de passagem, que determina o volume de dados que pode ser enviado, e à quantidade de telecomandos que podem ser armazenados na MTL, que é determinada pela quantidade de execuções de telecomandos ocorrida entre o envio de dois grupos de telecomandos subseqüentes. A experiência obtida pelo Grupo de Planejamento da Missão torna possível a eles afirmar que um MDAF típico, já segmentado, contém aproximadamente de 250 a 300 telecomandos. Existe um processo automatizado responsável por realizar esta segmentação, gerando aproximadamente 30 MDAFs segmentados por semana, a serem enviados à espaçonave Mars Express.

Depois de gerados os arquivos MDAFs, estes são revisados pela Equipe de Operação, responsável pelo processo de envio dos arquivos à Mars Express. Para este processo, os autores propõem o conjunto de regras abaixo:

- realizar o envio de arquivos MDAFS de acordo com os seus respectivos horários de execução. Esta regra representa o caso padrão, garantindo

que os arquivos com horários de execução anteriores sejam enviados por primeiro;

- realizar o envio de arquivos MDAFs de acordo com os tipos dos arquivos (e de seus telecomandos), no caso da regra de envio por horário de execução apresentar-se ambígua. Para cada tipo é alocado um nível de prioridade, como por exemplo, os arquivos MDAFs de telecomandos de apontamento para a Terra e de configuração de transmissores têm prioridade maior do que os MDAFs de obtenção de dados científicos;
- realizar o envio de arquivos MDAFs com confirmação completa. Esta regra representa o caso padrão. Cada operação de envio de telecomandos é totalmente confirmada em solo após ter sido recebida pelos subsistemas da espaçonave, processada e inserida na MTL;
- realizar o envio de arquivos MDAFs sem confirmação completa. Em algumas situações, uma janela de passagem pode não apresentar duração suficiente para que a confirmação completa possa ocorrer. Por exemplo, isto ocorre quando a espaçonave Mars Express encontra-se em eclipse. No caso da duração de uma janela de passagem ser insuficiente para a confirmação completa, aceita-se que ao menos o recebimento da transferência do arquivo MDAF seja confirmado pelos subsistemas da espaçonave antes da perda do sinal de rastreamento;
- realizar o envio de arquivos MDAFs tendo uma janela de passagem de segurança, como *backup*, disponível em outra estação terrena. Esta regra assegura que os telecomandos serão enviados à espaçonave mesmo no caso de uma passagem não poder ser rastreada devido a problemas na estação terrena responsável pelo rastreamento.

Os objetivos deste processo de envio de telecomandos à Mars Express consistem em manter a MTL utilizada ao máximo durante a maior parte do tempo e enviar os telecomandos o mais breve possível a fim de evitar os

efeitos da ocorrência de possíveis problemas com as estações terrenas ou com o sistema de controle.

A ferramenta RAXEM, implementada para lidar com o problema de envio de telecomandos à Mars Express, auxilia o usuário a analisar o problema, realiza o planejamento de soluções e permite ao usuário interagir com decisões de planejamento. No entanto, apesar de ser uma ferramenta operacional desde 2007, ano em que foi projetada e logo depois operacionalizada, possui limitações quanto à implementação das regras citadas e encontra-se em fase de melhorias.

A ferramenta aceita como entrada: (a) os arquivos MDAFs do período de planejamento anterior para determinar o estado da memória MTL; (b) os arquivos MDAFs do período de planejamento atual; e (c) um arquivo que lista as janelas de passagem disponíveis para envio de telecomandos. O usuário seleciona os arquivos MDAFs que já se encontram a bordo da espaçonave a fim de desconsiderá-los para o processo de planejamento. Esta ação é necessária uma vez que, atualmente, a ferramenta ainda não é alimentada com os status dos envios dos arquivos. Em seguida, RAXEM gera uma solução. No caso da solução não contemplar o envio de todos os arquivos MDAFs, a ferramenta notifica o usuário, que pode tomar a decisão de dividir os arquivos em segmentos ainda menores de forma que se ajustem à janela de passagem e/ou a memória MTL. No pior caso, o usuário pode decidir desconsiderar um conjunto de arquivos MDAFs do plano de acordo com os níveis de prioridade definidos para os tipos de arquivos MDAFs.

O plano final de envio de telecomandos contém para cada arquivo MDAF o início da janela de passagem para envio dos telecomandos e a duração do envio. Mais de um arquivo MDAF pode ser enviado em uma única passagem, caso a duração da janela de passagem e a memória MTL comportem a demanda. Neste caso, os arquivos MDAFs enviados na mesma passagem possuem o mesmo horário de envio.

Existe uma ferramenta responsável por transformar o plano final em um formato legível para os Controladores de Vôo poderem executá-lo. A funcionalidade desta ferramenta não encontra-se incorporada à ferramenta RAXEM, mas os autores pretendem incorporá-la.

3.3 Automatização em Bordo do Satélite

3.3.1 Solução de Bordo do INPE para Satélites Científicos Brasileiros

Além dos programas de satélites de coleta de dados e de sensoriamento remoto, o Instituto Nacional de Pesquisas Espaciais (INPE) possui também o programa de Satélites Científicos e Tecnológicos, do qual faz parte o satélite de aplicação científica LATTES, atualmente em fase de projeto, para pesquisa da atmosfera equatorial, e monitoração e imageamento de raios-X.

Kucinskis (2007) descreve o cenário de atuação de satélites de aplicações científicas e propõem uma solução de bordo para um problema por eles identificado, conforme apresentado a seguir nesta seção.

Segundo Kucinskis (2007), os satélites de aplicações científicas possuem como carga útil um conjunto de experimentos científicos e tecnológicos, cujo controle se baseia em modos de operação pré-definidos e na execução de telecomandos temporizados. Estes telecomandos são gerados e validados em solo, e enviados para o satélite com agendamento de execução futura. Ao conjunto de telecomandos temporizados atribui-se o nome de plano de operações.

Qualquer anomalia relacionada à operação dos experimentos é relatada pelo satélite para a equipe de operação em solo via telemetria. Após analisar os dados sobre a anomalia e sobre o estado do satélite, a equipe (junto aos responsáveis pelos experimentos e outros envolvidos na missão) pode ou não modificar o plano de operações corrente. Como a comunicação ocorre apenas quando o satélite passa sobre sua estação terrena de rastreamento, o tempo

transcorrido entre a detecção da anomalia e a recepção e execução de um novo plano de operações para lidar com ela pode ser de dezenas de horas.

Esta forma de operação dos experimentos é perfeitamente adequada para observações científicas de longa duração. Existem, entretanto, fenômenos de curta duração cuja ocorrência é aleatória – uma perturbação ionosférica, por exemplo, pode ocorrer a qualquer momento e se manifestar por poucos minutos, ou por algumas horas.

Devido a curta duração e a dificuldade em saber quando um fenômeno deste tipo irá ocorrer, Kucinskis (2007) considera não ser suficiente deixar a cargo da equipe de operações em solo a geração de um novo plano de operações que reconfigure o sistema. Segundo os autores, o tempo necessário para que o fenômeno seja relatado e para que a equipe de operações gere e envie um novo plano ao satélite é em geral muito maior que a duração do fenômeno.

Os autores sugerem que os experimentos possam, ao detectarem a ocorrência de fenômenos de curta duração, solicitar diretamente ao computador de bordo do satélite a modificação do plano de operações corrente de forma a realocar temporariamente os recursos necessários para que sejam capazes de realizar uma análise mais detalhada do fenômeno. Esta realocação deve ocorrer de tal forma que afete o mínimo possível a operação de outros experimentos e do próprio satélite.

Neste contexto, os autores empregam a técnica de Planejamento da área de IA para os sistemas embarcados em satélites do INPE de forma a permitir o replanejamento autônomo de operações em resposta a eventos externos.

O algoritmo de planejamento criado por Kucinskis (2007) representa o problema apresentado como um Problema Baseado em Restrições (*Constraint Satisfaction Problem – CSP*) e gera soluções através de busca local guiada por restrições de escalonamento. Ao receber um esboço de plano, o planejador entra em um *loop* em busca da solução dos conflitos detectados. Para cada

conflito, diversas opções para a solução são testadas e avaliadas, e a melhor entre elas é selecionada. Este processo segue até que não haja mais conflitos no plano.

3.4 Análise das Soluções de Automatização de Missões Espaciais

As experiências de automatização do controle de missões espaciais citadas neste Capítulo, em centros de controles espaciais de âmbito internacional e também nacional, evidenciam a relevância e o interesse neste processo.

Em âmbito nacional, o trabalho de pesquisa de Kucinskis (2007) contribui no sentido de prover tecnologia de bordo para aumentar a autonomia de softwares de planejamento embarcados para missões espaciais brasileiras futuras. Por sua vez, a arquitetura proposta nesta tese tem como objetivo oferecer uma contribuição no sentido de prover automatização para o processo de planejamento do segmento de solo, segmento imprescindível para o controle de satélites.

Assim, este trabalho de pesquisa tem como propósito inicial investir em inovação tecnológica para o segmento solo a partir da análise do conjunto de aplicações de vanguarda da área de Controle de Missões Espaciais apresentadas nas Seções 3.2 e 3.3. A análise destas aplicações teve como objetivos adquirir conhecimentos específicos e práticos da área, e identificar aspectos de interesse para as missões espaciais brasileiras.

As soluções propostas pela empresa Scisys em parceria com o ESTEC (HARRIS et al., 2002) e pelo ESOC (FERRI et al., 2004) constituem arquiteturas voltadas para o controle de missões espaciais do ESOC, o centro de operações espaciais europeu. Por serem específicas ao gerenciamento de missões controladas pelo ESOC, estas arquiteturas consideram a infraestrutura de recursos de solo europeia, o que inviabiliza as suas aplicações em outros contextos. Além desta característica, faltam informações a respeito de

como os módulos desempenham suas atividades e como funcionam de forma integrada.

Os conhecimentos adquiridos a partir do estudo destas arquiteturas e, sobretudo, a motivação em agregar valores, como configurabilidade, levaram à proposta de uma nova arquitetura para a automatização do processo de planejamento de controle de missões espaciais brasileiras futuras. Esta arquitetura é apresentada em detalhes no Capítulo 7.

Os trabalhos voltados à automatização do controle da missão *Mars Express*, a missão de exploração ao planeta Marte da ESA, também contribuíram para a construção da proposta da nova arquitetura. Estes trabalhos focam os problemas de geração de planos de operações para os dois sentidos de envio de dados: o plano de operações de descarregamento de dados, ou seja, no sentido espaçonave-Terra, e o plano de operações de vôo a ser enviado a uma espaçonave, no sentido Terra-espaçonave.

A arquitetura MEXAR2, em Cesta et al. (2007), gera planos de solução para o primeiro caso, gerando planos que contêm seqüências de operações para o envio de dados da memória de bordo da espaçonave *Mars Express* às estações terrenas. Apesar da nova arquitetura proposta não focar esta direção de envio de dados, a arquitetura MEXAR2 trouxe o benefício da idéia de utilização de níveis de prioridade para os dados. MEXAR2, após alocar os pacotes de dados a serem enviados a Terra em uma passagem, ordena-os conforme os seus respectivos níveis de prioridade.

Em contraste com a arquitetura MEXAR2, a arquitetura RAXEM, em Rabenau et al. (2008), gera planos de solução para o caso de envio de operações de vôo no sentido estação terrena para espaçonave. A arquitetura RAXEM, gerada após a operacionalização da arquitetura MEXAR2, também atribui níveis de prioridade para os diferentes tipos de arquivos de telecomandos (arquivos MDAFs) e respeita estes níveis para ordenar os arquivos a serem enviados.

A arquitetura proposta nesta tese trabalha com a geração de planos para o envio de operações no sentido da Terra para a espaçonave e com níveis de prioridade, mas difere-se da arquitetura RAXEM ao considerar níveis de prioridade para objetivos de rastreamento.

Considera-se que para cada passagem de satélite, o Plano de Operações de Voo (POV) deve ser gerado de forma a atender aos objetivos do rastreamento. Um objetivo de rastreamento é satisfeito ao ser enviado um conjunto específico de telecomandos ao satélite. No caso do tempo de uma passagem ser insuficiente para o envio de telecomandos que satisfaçam ao conjunto de objetivos de rastreamento, propõe-se que o POV seja gerado de forma a atender aos objetivos de maior prioridade.

Na nova arquitetura proposta adotou-se a técnica de Planejamento da área de IA para propiciar a geração de planos que atendam às restrições de tempo e de recursos, e para conferir configurabilidade ao processo de planejamento frente às mudanças no ambiente de controle de satélites. O próximo capítulo apresenta uma visão geral desta técnica.

4 PLANEJAMENTO EM INTELIGÊNCIA ARTIFICIAL

Planejamento automatizado é uma linha de pesquisa da área de Inteligência Artificial que investe esforços para desenvolver solucionadores genéricos de problemas. Várias gerações de pesquisadores têm dedicado esforços neste sentido, deixando como legado uma vasta bibliografia. Este capítulo apresenta uma revisão sobre esta bibliografia a partir de um conciso histórico da área de Planejamento em IA e, em seqüência, descreve três planejadores existentes e um *Framework* de Planejamento Temporal.

As descrições dos planejadores têm como finalidade apresentar um embasamento teórico sobre algoritmos de planejamento uma vez que a arquitetura proposta nesta tese, a ser descrita no Capítulo 7, é constituída por um planejador. Por sua vez, a descrição do *Framework* de Planejamento Temporal, proposto em Coddington (2002) e Coddington e Luck (2004), deve-se ao fato deste *framework* ter-se apresentado como uma referência importante para as decisões de projeto da arquitetura de planejamento automatizado proposta.

4.1 Linha do Tempo

A descrição de problemas em termos de enumeração de passos para a satisfação de um objetivo apareceu no final dos anos cinquenta, com problemas clássicos da área de IA como a ida ao aeroporto e os missionários e canibais.

Nesta época, a dificuldade era centrada na correta formalização do problema por meio de premissas que pudessem ser usadas para a geração automática da solução. Quando existiam, os algoritmos que usavam estas premissas na satisfação de objetivos eram complexos e específicos para cada problema.

Alguns anos mais tarde, McCarthy propôs uma linguagem formal na tentativa de encapsular todos os componentes dos problemas temporais: o Cálculo

Situacional. Apesar da nova linguagem apresentar um avanço considerável em relação à representação de conhecimento, ainda faltava um algoritmo capaz de utilizar tal representação para a solução de problemas de planejamento. Foi então que, no começo dos anos setenta, Fikes e Nilsson lançaram o famoso planejador STRIPS (FIKES e NILSSON, 1971). De formulação simples, o STRIPS marca o início do estudo de planejamento automatizado a partir da idéia dos operadores ADD-DEL-PRE. Os operadores STRIPS permaneceram quase duas décadas como paradigma vigente, mas haviam problemas difíceis de serem resolvidos por eles. A complexidade alta de seu algoritmo - busca exaustiva no espaço de estados - fez com que muitas melhorias fossem propostas.

Os anos oitenta foram marcados por um certo desânimo quanto à capacidade da geração automática de planos, e alguns poucos resultados apareceram somente no final da década. O planejador mais famoso do final da década de oitenta foi o Prodigy (MINTON et al., 1989), que adotava uma engenhosa heurística de planejamento incluindo reordenação de operadores, aprendizagem automática e busca em paralelo de sub-objetivos.

Apesar destes esforços, os pesquisadores não conseguiam um mecanismo robusto o bastante para resolver problemas genéricos. A década de noventa começou com a busca pelo aumento da velocidade de geração de planos e alguns estudos sobre linguagens e técnicas alternativas ao STRIPS. Nessa época, surgiram alguns conceitos válidos ainda hoje, como planejamento em ordem parcial, uso de variáveis na definição do esquema de ações, efeitos condicionais e linguagens baseadas em causalidade entre os operadores. SNLP (MCALLESTER e ROSENBLITT, 1991) e UCPOP (PENBERTHY e WELD, 1992) completaram com o PRODIGY o grupo de planejadores baseados em STRIPS de maior sucesso.

A partir de 1995, a história do planejamento automatizado sofreu novo impulso quando Avrim Blum lançou o GRAPHPLAN (BLUM e FURST, 1995) - um

mecanismo de representação de intervalos de tempo e extração de planos por meio de grafos. A simplicidade deste novo planejador, aliado ao desempenho superior aos planejadores da época, estimulou o desenvolvimento de muitos outros planejadores a partir de sua idéia central. Tal sucesso permitiu a recuperação de idéias associadas a planejamento por satisfatibilidade.

Um ano mais tarde, Henry Kautz e Bart Selman lançaram o SATPLAN (KAUTZ e SELMAN, 1992) (KAUTZ et al., 1996), um planejador que traduzia o conhecimento do GRAPHPLAN ao problema da satisfatibilidade. Tal transformação na forma de representar os problemas se mostrou mais eficiente que o GRAPHPLAN na maioria dos problemas em que os dois sistemas foram comparados.

Em 1997, Bonet e Geffner propuseram uma técnica de planejamento baseada em busca heurística denominada *Heuristic Search Planning* – HSP (BONET et al., 1997). Esta técnica define uma função de custo que permite verificar a distância dos estados intermediários ao estado objetivo e conferiu maior agilidade ao processo de geração de planos.

A comunidade de pesquisa na área de Planejamento em IA também investiu esforços no problema de planejamento temporal a fim de gerar planos cujas ações apresentam durações. Na técnica de planejamento temporal, associa-se a cada ação uma restrição temporal de início e fim de ação.

As seções a seguir descrevem os termos fundamentais da área de Planejamento Automatizado em IA e, em seguida, detalham os planejadores: (a) GRAPHPLAN, pelo fato do planejador utilizado no protótipo deste trabalho de pesquisa, chamado planejador LPG-TD, ter como base a técnica de planejamento em grafos utilizada pelo algoritmo GRAPHPLAN; e (b) LPG-TD, por ser o planejador adotado pela arquitetura de software proposta nesta tese, responsável pela automatização da atividade de planejamento da arquitetura.

4.2 Termos Fundamentais de Planejamento Automatizado

Apesar da não existência de uma nomenclatura padrão na área de pesquisa em planejamento automatizado, alguns termos são compartilhados pela comunidade de pesquisa desta área, o que demonstra um consenso sobre os conceitos em planejamento. Os termos fundamentais da área são apresentados a seguir:

- **Fluentes:** na lógica clássica, caracterizamos os objetos que compõem um mundo por meio de fatos. No planejamento automatizado, entretanto, estas informações são associadas a instantes de tempo, ou seja, o que é válido em um momento pode ser inválido no momento seguinte. Essa não-monotonicidade é representada nos problemas de planejamento pelo termo fluente, ou seja, um fluente é uma proposição cujo valor verdade é dado em função do tempo;
- **Estados:** sendo L uma linguagem da Lógica Clássica de 1ª ordem, um estado S_t é um conjunto finito de fluentes pertencentes a L e que são verdadeiros no instante de tempo t . Note-se que esta definição assume um mundo fechado, ou seja, todo fluente que não pertence ao conjunto S_t é considerado falso no instante de tempo t ;
- **Operadores:** também chamados de ações. As ações são esquemas que representam regras de inferência não instanciadas. Um esquema de ação é constituído pelo nome da ação e lista de parâmetros, uma pré-condição que deve ser válida para que a ação seja executada e o efeito resultante de sua execução. Durante o processo de planejamento, o algoritmo planejador deve instanciar variáveis com fluentes selecionados a partir de algum critério - este processo de instanciação e seleção das ações é o cerne da complexidade dos problemas de planejamento;
- **Domínio:** o domínio é composto pela enumeração dos predicados e do conjunto de operadores aplicáveis a um determinado ambiente;

- Problema: o problema é composto pela descrição do estado inicial e do estado objetivo associado a um ambiente, ou seja, descreve-se o estado atual de um mundo e o estado esperado após a aplicação de um plano;
- Plano: é a lista ordenada de operadores instanciados por um planejador e que permite a transição do estado atual ao estado objetivo de um problema.

A partir dos termos citados acima, podemos conceituar o termo planejamento automatizado por meio da definição abaixo:

- Definição de Planejamento Automatizado: é a enumeração do conjunto de instâncias de operadores aplicáveis ao domínio de um problema de planejamento e que permitem a transição do estado inicial, dito S_0 , para o estado objetivo S_n , que representa o estado desejado do problema de planejamento.

A definição acima pressupõe o uso de lógica clássica e, apesar de ser genérica, é considerada válida para a maioria dos sistemas planejadores conhecidos.

Formas de Busca

Em relação à forma de busca dos algoritmos de planejamento, é possível buscar em qualquer um dos sentidos, ou seja, para a frente a partir do estado inicial ou para trás a partir do objetivo (RUSSELL e NORVIG, 2004). A busca nestes dois sentidos é possível devido ao fato de esquemas de ações conterem tanto pré-condições quanto efeitos.

A busca para frente no espaço de estados é também chamada de planejamento por progressão. Neste tipo de busca, começa-se no estado inicial do problema e considera-se seqüências de ações até que seja encontrada uma seqüência que alcance o estado objetivo.

No entanto, a busca para a frente no espaço de estados considera todas as ações aplicáveis a partir de cada estado. Uma ação é aplicável a um estado quando a sua pré-condição for satisfeita. Devido à esta característica, este tipo de busca necessita de uma heurística muito precisa para se tornar eficiente.

A busca para trás no espaço de estados é também chamada de planejamento por regressão. Esse tipo de busca começa no(s) estado(s) objetivo e utiliza o inverso das ações para buscar pelo estado inicial. A principal vantagem da busca para trás é que ela permite considerar apenas ações relevantes. Uma ação é relevante para um objetivo conjuntivo se ela alcança um dos elementos da conjunção do objetivo. Por exemplo, em um problema de carga aérea o objetivo é mover n itens de carga C_n do aeroporto A para o aeroporto B , sendo $n = 20$. Este objetivo é mostrado mais precisamente abaixo, onde o símbolo \wedge representa uma conjunção de elementos:

$$Em(C1, B) \wedge Em(C2, B) \wedge \dots \wedge Em(C20, B)$$

Considerando o elemento da conjunção $Em(C1, B)$ e trabalhando no sentido para trás, pode-se buscar ações que têm este elemento como um efeito. Assim, a busca para trás geralmente tem um fator de ramificação mais baixo que a busca para frente.

Espaço de Busca

Em relação ao espaço de busca, pode-se ter um espaço de estados do ambiente ou um espaço de planos.

A busca no espaço de estados consiste em escolher ações aplicáveis a um estado de forma a alterar o ambiente para um estado desejado, que representa a solução do problema.

A busca no espaço de planos consiste em uma busca que inicia-se com um plano vazio e aprimora o plano até a obtenção de um plano completo que resolva o problema. As ações nesta busca não são ações em estados de um

ambiente, mas sim ações sobre planos, como adicionar um passo a um plano e impor uma ordenação que coloque uma ação antes de outra. Assim, os estados do problema de busca passam a ser planos e não situações.

4.3 Planejador GRAPHPLAN

Blum e Furst propuseram uma técnica de planejamento automatizado baseada na análise de grafos de planejamento, desenvolvendo o algoritmo denominado GRAPHPLAN (BLUM e FURST, 1995).

Um grafo de planejamento consiste em uma seqüência de níveis que correspondem a períodos de tempo no plano, sendo que o nível 0 representa o estado inicial. Cada nível é constituído por um único tipo de nodo: nodos literais (proposicionais) ou nodos de ações. O grafo alterna entre níveis de nodos literais e níveis de nodos de ações (RUSSELL e NORVIG, 2004) (CARDOSO, 2006). A Figura 4.3 apresenta uma ilustração de um grafo de planejamento.

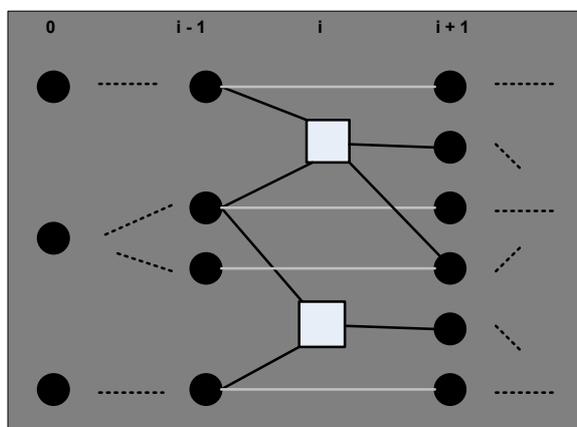


Figura 4.3 – Representação de um grafo de planejamento

Fonte: Retirada de Weld (1999)

Os nodos literais (representados pelos círculos) representam as proposições sobre o conhecimento do mundo, que podem ser verdadeiras dependendo das ações executadas nos níveis anteriores. Os nodos das ações (representados

pelos quadrados) representam as ações, que podem ter suas pré-condições satisfeitas em um período de tempo, dependendo de quais literais são válidos.

Existe um nodo para cada instância de ação cujas precondições estão presentes (e são mutuamente consistentes) no nível anterior. As linhas que conectam os literais com as instâncias de ações do próximo nível significam que as precondições das ações referenciam tais literais. Adicionalmente, as linhas que conectam nodos de ações aos literais subseqüentes significam que os mesmos se tornam verdadeiros pelos efeitos das ações. As linhas cinzas e horizontais entre os nodos literais representam “ações de persistência”, que indicam que um nodo literal permanece verdadeiro ao passar de níveis de literais, se nenhuma ação o alterar. Uma ação de persistência consiste em uma ação que permite a um literal não ser alterado após passar pela ação.

O algoritmo GRAPHPLAN funciona em duas fases: (1) geração do grafo de planejamento, representando os estados e as ações; e (2) extração do plano a partir deste grafo.

4.3.1 Fase de Geração do Grafo de Planejamento

Na geração do grafo de planejamento utiliza-se um processo de expansão. Em cada expansão do grafo adicionam-se as ações cujas pré-condições existem no conjunto de nodos literais do nível atual. Adicionam-se também ações de persistência para todos os literais deste nível. Os efeitos de todas as ações adicionadas são inseridos no conjunto de nodos literais do nível sucessor.

No processo de expansão, o próximo passo consiste em procurar por relações de exclusão mútua para as ações e os literais, também chamadas de relações mutex, e acrescentar estas relações ao grafo. Duas ações em um determinado nível A_i são consideradas de exclusão mútua caso satisfaçam aos seguintes critérios:

- Efeitos inconsistentes: o efeito de uma ação é a negação do efeito da outra ação;

- Interferência: um dos efeitos de uma ação é a negação de uma pré-condição da outra ação;
- Necessidades concorrentes: uma das pré-condições de uma ação é mutuamente exclusiva com uma pré-condição da outra ação.

Os vínculos de exclusão mútua entre duas ações registram conflitos que impedem que ambas as ações ocorram no mesmo nível.

Uma relação de exclusão mútua é válida para dois literais de um mesmo nível se um for a negação do outro ou se as ações geradoras destes literais forem mutuamente exclusivas. A relação de exclusão mútua entre dois literais é chamada de “suporte inconsistente” e também indica que ambos os literais não podem ocorrer no mesmo nível, independente da escolha das ações. Assim, o suporte inconsistente é uma restrição capaz de definir o conjunto de estados possíveis.

A Figura 4.4 ilustra as relações de exclusão mútua. Os círculos denotam literais, os quadrados representam ações, e as linhas curvas e finas representam as relações de exclusão mútua. As três primeiras partes ilustram as relações de exclusão mútua entre ações (quadrados pretos) e a última parte, a relação de exclusão mútua entre literais (círculos pretos).

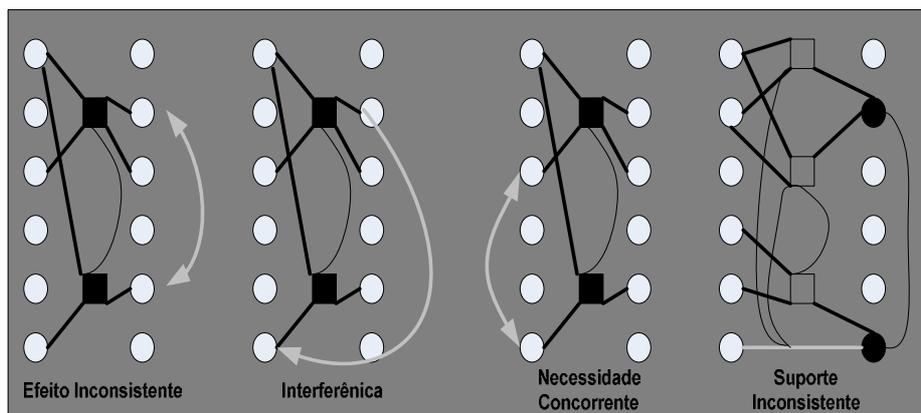


Figura 4.4 – Representação gráfica da definição de relação de exclusão mútua

Fonte: Retirada de Weld (1999)

O processo de expansão estende-se até chegar o momento em que dois conjuntos de nodos literais de níveis consecutivos sejam idênticos (S_{i-1} e S_i). Este momento é denominado nivelamento do grafo.

O processo de construção do grafo resulta em uma estrutura na qual:

- todo nível A_i contém todas as ações que são aplicáveis em S_i , juntamente com as relações de exclusão mútua que informam quais são os pares de ações que não podem ser ambas executadas;
- todo nível S_i contém todos os literais que poderiam resultar de qualquer escolha possível de ações no nível A_{i-1} , juntamente com relações de exclusão mútua que informam quais pares de literais não podem se tornar ambos verdadeiros no nível S_i .

É importante ressaltar que o processo de construção do grafo não requer a escolha entre ações, o que exigiria uma busca combinatória. Ele apenas registra a impossibilidade de certas escolhas com a utilização de relações de exclusão mútua.

Quando o nivelamento do grafo de planejamento ocorre, o algoritmo verifica se o último nível do grafo contém todos os literais do objetivo e se estes literais do objetivo são mutuamente exclusivos. Se o último nível do grafo omitir um dos literais do objetivo do problema ou se dois dos literais do objetivo forem mutuamente exclusivos, o problema não pode ser resolvido. Conseqüentemente, o algoritmo é interrompido e retorna-se uma falha.

Se o último nível tiver todos os literais do objetivo e estes literais não forem mutuamente exclusivos, então é possível que o grafo contenha um plano válido para o problema em questão. Esta condição é necessária, mas insuficiente para afirmar que o plano existe. Para tal, é necessário que não hajam relações de exclusão mútua entre as ações que geraram os literais do objetivo.

4.3.2 Fase de Extração do Plano

A verificação da ausência de relações de exclusão mútua, ou seja, de conflitos, entre as ações que geraram os literais do objetivo é denominada extração da solução.

A extração da solução consiste em um problema de busca com as seguintes características:

- Parte do nível mais alto do grafo ao nível mais baixo;
- O estado inicial é representado pelo conjunto de literais do objetivo do problema de planejamento e encontra-se no último nível do grafo de planejamento (nível S_i , onde $i=n$);
- Para cada literal do nível S_i , o algoritmo seleciona uma ação geradora do nível A_{i-1} e verifica se esta ação não é mutuamente exclusiva com as outras ações geradoras selecionadas no mesmo nível A_{i-1} ;
- Caso não haja conflito, o algoritmo repete o processo com as pré-condições destas ações selecionadas como literais do objetivo e, assim por diante, até chegar ao nível 0;
- Caso chegue ao nível 0, o plano existe.

Note-se que um literal do objetivo pode ser gerado por mais de uma ação, obrigando o algoritmo a considerar todas as combinações de ações geradoras.

Em resumo, o algoritmo de planejamento em grafo GRAPHPLAN constrói um grafo de forma incremental, partindo do estado inicial. Neste grafo, cada nível (camada) contém um conjunto de todos os literais ou ações que poderiam ocorrer em certo período de tempo e codifica relações de exclusão mútua entre literais ou entre ações que não podem ocorrer concomitantemente. O grafo gerado é posteriormente processado pelo algoritmo GRAPHPLAN, utilizando busca para trás para extrair um plano.

4.4 Planejador LPG-TD

O planejador LPG-TD (GEREVINI et al., 2005) é uma extensão do planejador LPG (GEREVINI et al., 2003), sendo utilizado no trabalho de pesquisa de Cardoso (2006) e também na arquitetura de planejamento automatizado proposta neste trabalho.

Conforme descrito em Cardoso (2006), o planejador LPG-TD é um planejador automático e incremental no processo de geração de planos para problemas que envolvam os seguintes elementos: ações STRIPS; ações com duração; ações e objetivos com expressões numéricas; operadores com efeitos de quantificação universal; operadores com precondições de quantificação existencial; operadores com precondições disjuntivas; operadores com precondições implicativas; literais iniciais temporizados (eventos exógenos incondicionais determinísticos); e predicados derivados por axiomas do domínio.

Assim como o LPG, o LPG-TD também baseia-se em pesquisa local estocástica em um espaço particular de grafos de ações, derivado da especificação do problema de planejamento. O espaço de busca do planejador é formado por subgrafos particulares do grafo de planejamento, chamados de grafos de ações, que representam planos parciais.

Dado um grafo de planejamento G , o processo de busca local começa em um grafo de ação inicial A de G (isto é, um grafo parcial), e transforma-o em um grafo de solução por meio da aplicação iterativa de algumas modificações no grafo A , melhorando o plano parcial corrente. Estas modificações podem ser realizadas de duas formas: (i) adicionando um novo nodo de ação no grafo de ação corrente; ou (ii) removendo um nodo de ação do grafo juntamente com os níveis correspondentes de nodos de precondições e efeitos do grafo de planejamento.

Em qualquer passo no processo de busca, o qual produz um novo grafo de ação, o conjunto de ações que podem ser adicionadas ou removidas é determinado pelas violações de restrições que existem no grafo de ação corrente. A violação de restrição (ou de inconsistência) é qualquer relação mutex que envolva nodos de ação no grafo de ação corrente ou um literal não suportado que seja condição de uma ação no plano parcial corrente.

O esquema geral para encontrar um grafo solução (isto é, um estado final da busca) é constituído por dois passos principais. O primeiro passo é a inicialização da busca, na qual é construído o grafo de ação inicial. O segundo passo é o processo de busca local no espaço de todos os grafos de ação, iniciando pelo grafo de ação inicial.

Neste planejador, as técnicas de busca local básicas foram melhoradas por meio do uso de uma função parametrizada de avaliação de ação que permite especificar diferentes tipos de heurísticas. Esta função especifica o custo de inserir e de remover uma ação do plano parcial, representado pelo grafo de ação corrente. Várias técnicas são usadas para definir e avaliar a busca na vizinhança de modo a explorar a estrutura de um grafo subjacente. Estas técnicas incluem os multiplicadores de Lagrange, a propagação Noop, a ordenação de ações e o tamanho do grafo incremental, e os custos de condições heurísticas.

O enfoque dado pelo planejador LPG-TD para o tratamento de problemas de planejamento temporal, nos quais as ações têm durações e devem ser executadas em uma determinada janela de tempo, consiste na combinação de duas técnicas: (i) algoritmo polinomial para raciocínio temporal baseado em restrições durante a busca; e (ii) técnicas de busca local para planejamento que exploram informação temporal.

4.5 *Framework* de Planejamento Temporal

A técnica de Planejamento Temporal ganhou espaço na comunidade de Planejamento de IA devido à necessidade de se oferecer planos, a problemas do mundo real, cujas ações apresentem durações. Esta necessidade foi retratada na 3ª Competição Internacional de Planejamento, evento conhecido pela sigla IPC (*International Planning Competition*) e ocorrido no ano de 2002. Neste evento, passou-se a desenvolver domínios de planejamento que incluíam ações com duração, chamadas de *durative actions* em inglês.

Na direção do planejamento temporal, Coddington (2002) apresenta à comunidade de Planejamento de IA um *Framework* de Planejamento que, além de considerar que as ações apresentam durações, acrescenta prioridades aos objetivos e prazos finais para que estes sejam alcançados. A priorização de objetivos torna o sistema capaz de abandonar o alcance de um subconjunto de objetivos a favor de outros, no caso do tempo do plano ser insuficiente para o alcance de todos os objetivos dentro dos seus respectivos prazos.

Coddington projetou o *Framework* de Planejamento para ser empregado por um agente de planejamento. O termo “agente de planejamento” será definido no Capítulo 5. No *Framework* proposto, novos objetivos podem ser apresentados ao agente de planejamento mesmo enquanto este estiver planejando. Em decorrência deste dinamismo do ambiente, faz-se necessário intercalar as atividades de geração e execução do plano. Para realizar esta intercalação, Coddington utiliza a técnica de planejamento contínuo, uma técnica descrita em detalhes em Russell e Norvig (2004), voltada para ambientes dinâmicos.

O *Framework* de Planejamento proposto é constituído por um conjunto de componentes. Dentre este conjunto de componentes, um componente chamado “Planejar para Alcançar Objetivo” gera um plano para alcançar um objetivo, que lhe é previamente fornecido por outro componente do *Framework*.

O componente “Planejar para Alcançar Objetivo” apresenta em sua estrutura três subcomponentes, ilustrados na Figura 4.5.

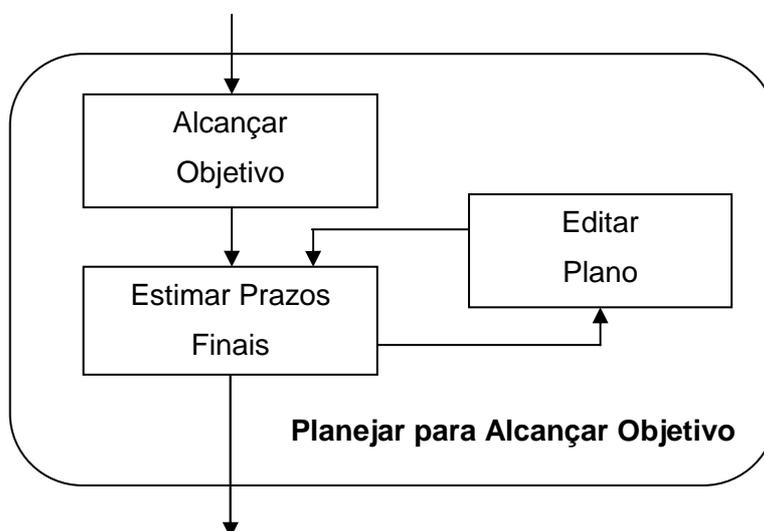


Figura 4.5 – Componente “Planejar para Alcançar Objetivo”

Fonte: Adaptada de Coddington (2002)

O subcomponente “Alcançar Objetivo” representa o planejador, que responsabiliza-se por gerar um plano utilizando o paradigma de Planejamento de Ordem Parcial de forma estendida. Este paradigma encontra-se descrito em detalhes em McAllester e Rosenblitt (1991).

Uma vez gerado o plano, o subcomponente “Estimar Prazos Finais” responsabiliza-se por determinar se existe tempo suficiente disponível para alcançar todos os objetivos no plano nos seus respectivos prazos finais. Caso este processo fracasse, ou seja, caso não exista tempo suficiente para alcançar todos os objetivos no plano nos seus prazos finais, o plano é fornecido ao subcomponente “Editar Plano”.

O subcomponente “Editar Plano” responsabiliza-se por editar o plano por meio da remoção de um objetivo de menor prioridade e das ações e restrições associadas a este. Após o plano ter sido editado, o subcomponente “Estimar

Prazos Finais” volta a estimar os prazos finais. Caso ainda o tempo no plano seja insuficiente, o plano é novamente editado. Este processo continua até que o tempo seja suficiente para executar o plano editado, ou seja, reduzido.

Para implementar o subcomponente “Alcançar Objetivo” o paradigma de Planejamento de Ordem Parcial foi estendido para oferecer suporte ao fato de que os objetivos possuem prazos finais e valores que indicam seus respectivos níveis de prioridade, e as ações apresentam durações. Esta extensão inclui armazenar as dependências existentes entre ações e objetivos durante o processo de planejamento para tornar possível a posterior edição do plano. Cada ação contém uma lista de objetivos para os quais ela contribui.

O *Framework* de Planejamento proposto em Coddington (2002) apresenta-se como uma abordagem interessante para tratar problemas de planejamento que envolvam prazos para o alcance de objetivos. O problema de geração de planos para o controle de múltiplos satélites, objeto de estudo deste trabalho de pesquisa, consiste em um problema que apresenta um prazo final de execução para o objetivo “rastrear satélite x na estação terrena y ”, representado pelo período da janela de passagem do satélite. Desta forma, considerou-se a possibilidade de aplicação do *Framework* de Planejamento proposto em Coddington (2002) para este problema em estudo.

No entanto, em análises a respeito da abordagem do *framework* verificou-se que o armazenamento das dependências existentes entre ações e objetivos durante a geração do plano representa um custo adicional durante o processo de planejamento. Esta verificação originou a intenção de diminuir ou até eliminar este referido custo. Com esta intenção, o mecanismo de edição de planos adotado pelo *framework* de Coddington foi descartado como possível abordagem a ser usada na arquitetura de planejamento proposta neste trabalho e propôs-se uma nova abordagem.

Nesta nova abordagem para a geração de planos considerando objetivos com prazos de execução e prioridades, quando o período da janela de passagem de

um satélite for insuficiente para o alcance do conjunto de objetivos de rastreo, não gera-se um plano para posteriormente editá-lo, ou seja, para posteriormente remover os objetivos de menor prioridade e as ações associadas a estes. Ao contrário, primeiramente remove-se um objetivo de menor prioridade para posteriormente gerar-se o plano de forma a evitar a necessidade de edição do plano e o decorrente custo adicional deste mecanismo.

Portanto, propõe-se que, em caso de tempo insuficiente para o alcance de um conjunto de objetivos de rastreo, a arquitetura de planejamento proposta remova objetivos de menor prioridade a partir das definições de entrada para o processo de planejamento e planeje para o subconjunto de objetivos obtido a partir do conjunto de objetivos original. O ciclo remoção de objetivo e planejamento segue até ser possível a geração de um plano que alcance um subconjunto do conjunto de objetivos originais.

A possibilidade de desconsiderar objetivos possibilita a geração de planos ajustáveis às limitações de tempo das janelas de passagens dos satélites e que atendam aos principais objetivos de rastreo.

5 AGENTE DE PLANEJAMENTO

Com o propósito de fornecer subsídios para a compreensão da arquitetura de planejamento proposta, este capítulo apresenta uma fundamentação teórica sobre o conceito de agentes. A arquitetura proposta é constituída por um grupo de agentes, dentre o qual se encontra um agente de planejamento.

Portanto, apresenta-se também neste capítulo o conceito de agente de planejamento e descrevem-se alguns elementos essenciais para a construção deste tipo de agente: (1) a classificação dos ambientes em que este pode atuar segundo um conjunto de propriedades; e (2) a adoção de uma linguagem de representação a fim de descrever o problema de planejamento para o qual o agente proverá soluções.

Em relação às linguagens de representação, ao final deste capítulo apresenta-se a evolução do conjunto de linguagens de representação de problemas de planejamento baseadas na formulação STRIPS, descrevendo de forma mais detalhada a notação da linguagem PDDL, a linguagem adotada pelo agente de planejamento da arquitetura proposta e também pela comunidade de Planejamento em IA em seus trabalhos de pesquisa.

5.1 Definição de Agente

A definição de agente não é consensual (LIEBERMAN, 1997). O maior problema na definição do termo agente é que se trata de um vocábulo extensamente usado por diversos pesquisadores que trabalham em áreas estreitamente relacionadas (SYCARA, 1998). Assim, existem várias definições vinculadas a pesquisadores empenhados no estudo e no desenvolvimento de agentes. A seguir, apresentam-se algumas destas definições.

Para Franklin e Graesser (1996), um agente é um sistema situado dentro de um ambiente e como parte deste ambiente, sente e age sobre o mesmo, ao longo do tempo, em busca de cumprir sua própria agenda e de maneira que afete o seu próprio comportamento no futuro.

Para Ferber (1999), um agente é uma entidade física ou virtual com as seguintes características: é capaz de agir em um ambiente, pode comunicar-se diretamente com outros agentes, é dirigido por um conjunto de tendências, possui recursos próprios, é capaz de perceber o ambiente com uma extensão limitada, tem somente uma representação parcial do seu ambiente, oferece serviços, seu comportamento tende a satisfazer os seus objetivos, e depende de suas percepções, de suas representações e da comunicação que ele estabelece com o ambiente.

Por entidade física, entende-se algo que possui uma existência física e age no mundo real, como um robô, uma espaçonave ou um carro. Por entidade virtual entende-se algo que não existe na forma física, mas existe na forma lógica, como um componente de software ou um modelo computacional.

Para Russell e Norvig (2004), um agente pode ser visto como algo que observa o ambiente por meio de sensores e age neste ambiente por intermédio de atuadores. Para exemplificar este conceito de agente, pode-se fazer um comparativo entre agentes humanos, agentes robôs e agentes de software.

Um agente humano possui olhos, ouvidos e outros órgãos que funcionam como sensores no ambiente, e possui mãos, pernas, boca e outras partes do corpo que agem como atuadores. Um agente robô, por sua vez, pode possuir câmeras de vídeo e outros sensores para observar o ambiente, e pode ter vários motores como atuadores. Um agente de software recebe seqüências de digitações, conteúdo de arquivos e/ou pacotes de rede como entradas sensórias e atua sobre o ambiente exibindo algo na tela, gravando arquivos e/ou enviando pacotes de rede.

Em termos matemáticos, Russell e Norvig afirmam que o comportamento de um agente é descrito por uma função que mapeia qualquer seqüência de percepções específica para uma ação. Para um agente de software, esta função é implementada por um programa de agente, que consiste em uma implementação concreta.

Caire et al. (2002) definem um agente como sendo uma entidade autônoma que é capaz de desempenhar alguma função específica. Eles consideram esta capacidade como sendo os serviços do agente. Um serviço possui o nível de conhecimento análogo a uma operação de um objeto.

A característica de autonomia significa que as ações de um agente não são somente ditadas por interações ou eventos externos, mas também pela própria motivação do agente. Os autores capturam a motivação de um agente em um atributo denominado propósito. O propósito de um agente determina se um agente concordará ou não em desempenhar um serviço requisitado e também em como este serviço será oferecido. Um agente pode ser tanto um agente humano quanto um agente de software, sendo ambos considerados especializações de agente.

Neste trabalho de pesquisa adota-se a definição de agente proposta por Caire et al. (2002), uma vez que esta definição representa um dos conceitos fundamentais de uma metodologia de engenharia de software orientada a agentes proposta pelos mesmos autores. Esta metodologia encontra-se descrita em detalhes no Capítulo 6 e a sua adoção oferece o benefício de se obter primitivas para descrever estruturas de alto nível presentes nos agentes. Estas estruturas de alto nível encontram-se presentes nos agentes somente quando estes são visualizados em um nível mais abstrato do que simplesmente visualizá-los como objetos complexos.

5.2 Definição de Agente de Planejamento

Um agente de software pode ser classificado de acordo com a sua funcionalidade principal. Segundo Russell e Norvig (2004), um agente de software capaz de construir planos de ações para um determinado problema é denominado agente de planejamento ou agente planejador.

Dado um objetivo, um agente de planejamento é capaz de construir um plano de ações para um determinado problema a partir de suas percepções do ambiente, executar estas ações e avaliar a satisfação do objetivo, modificando o seu plano,

se necessário. Neste trabalho de pesquisa adota-se esta definição de agente de planeamento proposta por Russell e Norvig (2004).

O problema de planeamento é tratado como um problema de inferência lógica, que utiliza lógica de primeira ordem como linguagem de representação. Dessa forma, o agente de planeamento representa o ambiente em que opera e é capaz de deduzir quais ações executar.

5.3 Ambientes de Planeamento

Um ambiente de planeamento é essencialmente um problema para o qual um agente de planeamento apresenta um plano para a sua solução.

Os ambientes de planeamento podem ser classificados de acordo com as propriedades que apresentam. Assim, um ambiente de planeamento pode ser classificado como clássico e não-clássico (RUSSELL e NORVIG, 2004).

Um ambiente de planeamento clássico é um ambiente que apresenta as seguintes propriedades - completamente observável, determinístico, estático (uma mudança ocorre somente quando o agente age) e discreto. Em contrapartida, um ambiente de planeamento não-clássico possui as propriedades - parcialmente observável e/ou estocástico. Para cada um destes tipos de ambientes de planeamento existe um conjunto diferente de algoritmos e projetos de agentes.

Estas propriedades, que classificam os ambientes de planeamento em clássicos e não-clássicos, encontram-se definidas na seção a seguir.

5.3.1 Propriedades dos Ambientes de Planeamento

5.3.1.1 Completamente observável versus parcialmente observável

Um ambiente é completamente observável se os sensores do agente de planeamento permitem acesso ao estado completo do ambiente. Os sensores devem ser capazes de detectar todos os aspectos que são relevantes para a

escolha das ações. Em ambientes com essa propriedade o agente não precisa manter um estado interno para ser capaz de agir. Um ambiente é parcialmente observável devido a sensores imprecisos ou porque partes do estado estão ausentes nos dados do sensor.

5.3.1.2 Determinístico versus estocástico

Um ambiente é considerado determinístico se o seu próximo estado pode ser completamente determinado pelo estado atual e pela ação executada pelo agente. Caso contrário, o ambiente é estocástico.

5.3.1.3 Episódico versus seqüencial

Um ambiente é considerado episódico quando a escolha da ação em cada episódio depende somente do próprio episódio e o agente não precisa pensar para frente. Cada episódio consiste na percepção do agente e depois na execução de uma única ação. Caso contrário, o ambiente é seqüencial, no qual a decisão atual pode afetar decisões futuras.

5.3.1.4 Estático versus dinâmico

Um ambiente é dinâmico para um agente se puder se alterar enquanto o agente estiver deliberando sobre a sua próxima ação. Caso contrário, o ambiente é estático. Os ambientes estáticos são mais fáceis de se manipular, porque o agente não precisa continuar a observar o mundo enquanto está decidindo sobre a realização de uma ação e não precisa se preocupar com a passagem do tempo.

5.3.1.5 Discreto versus contínuo

A distinção entre discreto e contínuo se aplica ao estado do ambiente, ao modo como o tempo é tratado, e às percepções e ações do agente. Por exemplo, um jogo de xadrez é considerado como um ambiente de tarefas discreto, pois apresenta um número finito de estados distintos e um conjunto discreto (valores separados) de percepções e ações. Dirigir um táxi é considerado como um ambiente de tarefas contínuo, pois o estado e o tempo são contínuos, ou

seja, ininterruptos. A velocidade e a posição do táxi e dos outros veículos passam por um intervalo de valores contínuos ao longo do tempo. As ações de dirigir um táxi também são contínuas considerando os ângulos de rotação do volante.

Dentro da classificação de ambientes de planejamento clássico e não-clássico, um ambiente de planejamento pode ainda ser classificado como de agente único ou multi-agente.

5.3.1.6 Agente único versus multi-agente

A distinção entre um ambiente de planejamento de agente único e um ambiente de planejamento multi-agente está relacionada à quantidade de agentes em um ambiente. Por exemplo, um agente que resolve um jogo de palavras cruzadas sozinho está claramente em um ambiente de planejamento de agente único, enquanto um agente que joga xadrez está em um ambiente de dois agentes (multi-agente).

Em um ambiente de planejamento multi-agente, surgem outras questões que devem ser consideradas: estes ambientes podem ser cooperativos ou competitivos. A cooperação entre agentes ocorre quando estes possuem objetivos em conjunto, enquanto que a competição assume objetivos conflitantes. Os planos gerados para ambientes multi-agente cooperativos são planos conjuntos, enquanto que os planos para ambientes competitivos são planos que visam os sucessos individuais dos agentes.

A combinação das propriedades listadas acima em um ambiente de planejamento determina o grau de dificuldade na implementação de um agente de planejamento. O ambiente mais complexo de se tratar consiste no ambiente que agrega as propriedades parcialmente observável, estocástico, seqüencial, dinâmico, contínuo e multi-agente.

5.3.2 Ambiente de Planejamento Clássico

O ambiente de planejamento clássico trata o tipo de problema para o qual são fornecidos o estado inicial do ambiente, o conjunto de ações e um objetivo a ser atingido. A solução é uma seqüência de ações que quando executada começa em um estado inicial e leva para uma situação na qual o objetivo é atingido. Assume-se que o planejador conhece tudo o que é verdade na situação inicial e também conhece os efeitos de cada ação.

No problema de planejamento temporal levam-se em consideração outros fatores, além dos elementos do planejamento clássico. Estes fatores são: o tempo gasto para a execução do plano, e o início e a duração de cada ação.

Segundo Russell e Norvig (2004), para os tipos de problemas de planejamento clássico e temporal, um agente pode primeiro planejar e depois executar inteiramente o plano. Estes autores comparam agentes de software a seres humanos afirmando que os agentes para estes tipos de problemas podem executar os planos gerados “com os olhos fechados”.

Para tratar este tipo de problema existem várias técnicas de planejamento, como a técnica de planejamento de ordem parcial (*Partial Order Planning - POP*), descrita por Russell e Norvig (2004), e a técnica de planejamento baseado em grafos, descrita no Capítulo 4.

Técnica de Planejamento de Ordem Parcial – um agente de planejamento de ordem parcial é capaz de inserir duas ações em um plano sem especificar qual delas deve ser executada por primeiro. Dessa forma, esta técnica de planejamento não se compromete com a geração de um plano com uma seqüência totalmente ordenada de ações. Um agente de planejamento de ordem parcial atua de forma regressiva, ou seja, no sentido para trás a partir do objetivo, acrescentando ações ao plano para alcançar cada subobjetivo pertencente ao objetivo final. Esta técnica de planejamento é aplicada de forma

efetiva em problemas onde a abordagem dividir e conquistar pode ser empregada.

5.3.3 Ambiente de Planejamento Não-Clássico

Em um ambiente de planejamento não-clássico, o agente não pode simplesmente executar o plano inteiro. Neste tipo de ambiente, o agente deve usar suas percepções para descobrir o que está acontecendo enquanto o plano é executado e, possivelmente, modificar ou substituir o plano se algo inesperado acontecer.

Para tratar este tipo de problema existem várias técnicas de planejamento, como as técnicas de planejamento condicional e de planejamento contínuo, descritas de forma resumida a seguir (RUSSELL e NORVIG, 2004).

Técnica de Planejamento Condicional – esta técnica de planejamento lida com a incerteza por meio da verificação do que realmente está acontecendo no ambiente em pontos pré-determinados no plano. Um agente de planejamento condicional constrói no plano, em tempo de planejamento, ações condicionais que verificam o estado do ambiente, em tempo de execução, com o objetivo de decidir qual a próxima ação a ser executada. Uma ação condicional é uma ação que representa um ponto de ramificação no plano final.

Técnica de Planejamento Contínuo – um agente de planejamento contínuo realiza o planejamento e o monitoramento de execução de forma contínua, alternando-os à medida que as ações de um plano se tornam prontas para serem executadas. O agente de planejamento contínuo constrói um plano de maneira incremental, obedecendo ao ciclo – perceber o ambiente, remover falha e executar ações. Após a execução de uma ação do plano, o novo estado do ambiente é refletido na ação “Iniciar” do plano, que é sempre usada para se obter o estado atual do ambiente.

De forma geral, o tratamento de um problema de planejamento do tipo não-clássico depende dos conjuntos de sensores e de atuadores disponíveis para o agente.

As técnicas de planejamento apresentadas nesta seção não serão detalhadas neste trabalho de pesquisa pelo fato de não serem empregadas na arquitetura de planejamento proposta. O agente de planejamento da arquitetura emprega uma técnica de planejamento baseada em grafos para ambientes clássicos, descrita no Capítulo 4. Os motivos que levaram à escolha desta técnica encontram-se descritos no Capítulo 7 junto à apresentação da arquitetura proposta.

No entanto, qualquer tipo de problema de planejamento, seja ele clássico ou não-clássico, inclui a representação de estados, ações e objetivos em uma linguagem apropriada. Esta linguagem de representação deve tornar possível a elaboração de algoritmos de planejamento que tirem proveito da estrutura lógica do problema em análise. A chave é usar uma linguagem de representação que seja suficientemente expressiva para descrever o problema, mas restritiva o bastante para permitir que algoritmos eficientes operem sobre ela.

5.4 Linguagens de Representação Usadas por Agentes de Planejamento

A representação de problemas de planejamento inclui a representação de estados, ações e objetivos, e deve tornar possível que um agente de planejamento desempenhe a sua tarefa de planejar conforme a estrutura lógica do problema em análise.

5.4.1 Linguagem STRIPS

Uma abordagem clássica de representação de problemas de planejamento consiste em descrever os elementos do problema de acordo com uma linguagem denominada STRIPS – STanford Research Institute Problem

Solver. Esta forma de representação foi implementada pela primeira vez pelo planejador STRIPS (FIKES e NILSSON, 1971).

A linguagem STRIPS, criada no começo dos anos setenta, é uma linguagem básica de representação de problemas de planejamento clássicos, constituída pelas seguintes representações (RUSSELL e NORVIG, 2004):

- Representação de estados: os planejadores decompõem o mundo em condições lógicas e representam um estado como uma conjunção de literais positivos. Os literais em descrições de estados de primeira ordem devem ser básicos e livres de funções, como por exemplo, os literais: $Em(\text{Avião1}, \text{Campinas}) \wedge Em(\text{Avião2}, \text{Brasília})$, que significam que o Avião1 encontra-se localizado na cidade de Campinas e o Avião2 em Brasília. Assim, literais como $Em(x, y)$ não são permitidos. A descrição de um estado não precisa ser completa. Quando um agente obtém uma descrição de estado incompleta, provavelmente de um ambiente inacessível, utiliza a hipótese de mundo fechado. Esta hipótese significa que quaisquer condições não mencionadas em um estado são consideradas falsas;
- Representação de objetivos: um objetivo é um estado parcialmente especificado, representado como uma conjunção de literais básicos positivos. Um estado proposicional s satisfaz a um objetivo g se s contém todos os átomos em g (e possivelmente outros). Por exemplo, o estado $(\text{Rico} \wedge \text{Famoso} \wedge \text{Íntegro})$ satisfaz ao objetivo $(\text{Rico} \wedge \text{Famoso})$. Os objetivos podem conter variáveis, como por exemplo, o objetivo de estar em uma loja x que vende leite: $Em(x) \wedge Vende(x, \text{Leite})$;
- Representação de ações: uma ação é especificada em termos das pré-condições que devem ser válidas para que a ação seja executada e dos efeitos resultantes de sua execução. Por exemplo, uma ação equivalente a voar em um avião de um local para outro pode ser representada na linguagem STRIPS conforme ilustra a Figura 5.1.

```
Ação( Voar(p, de, para),  
  PRECOND: Em(p, de)  $\wedge$  Avião(p)  $\wedge$  Aeroporto(de)  $\wedge$  Aeroporto(para)  
  EFEITO:  $\neg$ Em(p, de)  $\wedge$  Em(p, para)  
)
```

Figura 5.1 – Exemplo de um esquema de ação na linguagem STRIPS

Fonte: Retirada de Russell e Norvig (2004)

A forma de representação da Figura 5.1 é denominada esquema de ação, pois representa várias ações diferentes que podem ser derivadas pela instanciação das variáveis com constantes diferentes. Em geral, um esquema de ação é constituído de três partes:

- O nome da ação e uma lista de parâmetros. Por exemplo, `Voar(p, de, para)` identifica a ação;
- A pré-condição é uma conjunção de literais positivos que declara o que deve ser verdadeiro em um estado antes da ação ser executada. Quaisquer variáveis na pré-condição devem constar na lista de parâmetros da ação;
- O efeito é uma conjunção de literais positivos ou negativos que descreve como o estado se altera após a ação ter sido executada. As variáveis no efeito também devem constar na lista de parâmetros da ação.

Apresentada a sintaxe da linguagem STRIPS para representações de problemas de planejamento, a semântica pode ser descrita em termos de como as ações afetam os estados. Uma ação é aplicável a um estado se este satisfaz à sua pré-condição. Caso contrário, a ação não apresenta nenhum efeito. Para um esquema de ação de primeira ordem, o estabelecimento da aplicabilidade envolve uma substituição para as variáveis da pré-condição.

Considerando o exemplo mostrado acima de voar de um lugar para outro, suponha que se deseja voar de JFK para SFO, a ação `Voar(P1, JFK, SFO)` é aplicável se o estado atual satisfizer a pré-condição desta ação.

Considerando que o estado atual seja: $Em(P1, JFK) \wedge Em(P2, SFO) \wedge$
 $Avião(P1) \wedge Avião(P2) \wedge Aeroporto(JFK) \wedge Aeroporto(SFO)$.
Este estado satisfaz à pré-condição: $PRECOND\ Em(p, de) \wedge Avião(p) \wedge$
 $Aeroporto(de) \wedge Aeroporto(para)$ com substituição $\{p/P1, de/JFK,$
 $para/SFO\}$. Deste modo, a ação concreta $Voar(P1, JFK, SFO)$ é aplicável.

5.4.2 Linguagem PDDL

A linguagem *Planning Domain Definition Language* – PDDL, baseada na linguagem STRIPS e em extensões desta linguagem, foi criada com o objetivo de obter um modelo comum de testes para que os sistemas planejadores, que atuam como mecanismos de raciocínio dos agentes de planejamento, pudessem ser comparados entre si. Portanto, a PDDL é uma linguagem de descrição de problemas de planejamento que permite aos sistemas planejadores trocarem problemas de *benchmark* e compararem seus resultados.

Mcdermott (1998) e o comitê organizador da primeira edição da *International Planning Competition* (IPC) foram os responsáveis pelo desenvolvimento da primeira versão da linguagem PDDL, utilizada por todos os sistemas planejadores participantes desta competição. A linguagem PDDL tornou-se a linguagem padrão das competições bi-anuais de planejamento automático (IPCs) e apresentou-se, em cada competição, com uma versão diferente.

As várias versões da linguagem devem-se ao constante esforço da comunidade de pesquisa de Planejamento em IA em tornar a linguagem cada vez mais capaz de tratar problemas de planejamento reais, diminuindo a distância entre os requisitos de modelagem destes problemas e o que pode ser, de fato, expresso na linguagem PDDL (FOX e LONG, 2003).

A linguagem PDDL na versão 2.1, apresentada para ser utilizada na competição 3º IPC, propôs que fossem vinculadas durações às ações de planejamento (FOX e LONG, 2003).

Na versão 2.2, apresentada para ser utilizada na competição 4^o IPC, a linguagem PDDL passou a implementar literais que permitem a representação de eventos exógenos incondicionais determinísticos e predicados derivados. Um evento exógeno incondicional determinístico consiste em um fato que se torna verdadeiro ou falso em um determinado ponto de tempo, sendo que este ponto de tempo é de conhecimento prévio do planejador, e a sua ocorrência independe das ações escolhidas pelo planejador durante o processo de planejamento.

Por exemplo, em um plano para controlar o expediente de uma loja de departamentos a abertura da loja e o seu fechamento ao final do dia são eventos exógenos, pois são eventos que simplesmente ocorrem independente de outras ações no plano, como as ações de venda e/ou troca de produtos realizadas durante o expediente.

Um predicado derivado consiste em um predicado que não é afetado por nenhuma das ações disponíveis no plano, mas cujo valor verdade é derivado de um conjunto de regras (EDELKAMP e HOFFMANN, 2004).

5.4.2.1 Estrutura da Linguagem PDDL

A linguagem PDDL separa a descrição de ações parametrizadas, que caracterizam o comportamento do Domínio de planejamento, da descrição das condições iniciais do ambiente e dos objetivos que se deseja alcançar, que caracterizam a Instância do Problema de planejamento. Dessa forma, a linguagem distingue a descrição do Domínio de planejamento da descrição da Instância do Problema de planejamento, que juntos definem um Problema de planejamento. Com esta separação, a linguagem permite que uma mesma descrição de Domínio seja utilizada por várias descrições de Instâncias de Problemas a fim de produzir diferentes Problemas de planejamento para um mesmo Domínio.

A Figura 5.2 ilustra diferentes Problemas de planejamento, ou seja, pares de arquivos de descrição do Domínio e da Instância do Problema, sendo fornecidos como entradas a um Agente de Planejamento, responsável pela geração de Planos de solução para Problemas de planejamento.

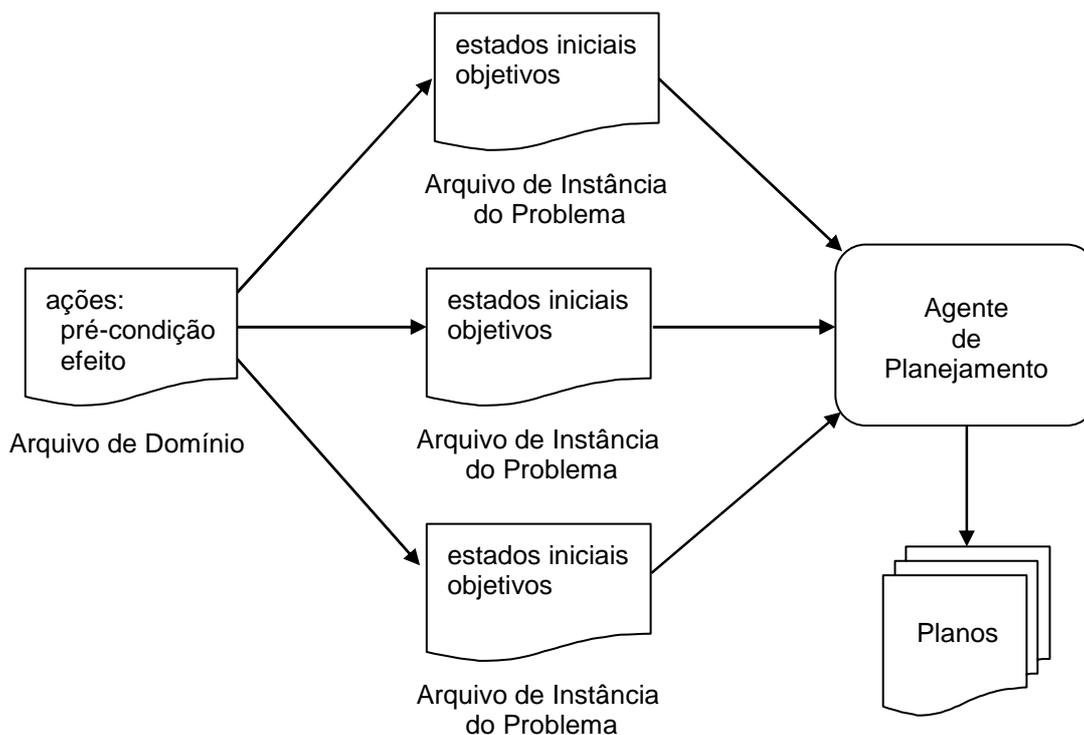


Figura 5.2 – Problemas de planejamento na linguagem PDDL

5.4.2.2 Descrição do Arquivo de Domínio

O Arquivo de Domínio modela o comportamento do domínio de planejamento por meio do seguinte conjunto de elementos:

- Tipos – representam as classes dos objetos que fazem parte do modelo. São definidos após a instrução *types*;
- Predicados – representam atributos que definem os estados dos objetos instanciados. São definidos após a instrução *predicates*;
- Funções – representam valores numéricos obtidos com a associação entre objetos. São definidas após a instrução *functions*;

- Ações – representam regras de inferência não instanciadas. Uma ação possui parâmetros, uma pré-condição para que seja executada e um efeito que ocorre após a sua execução. Para ações com duração, faz-se necessário definir também a duração da execução da ação. As ações são definidas após a instrução *action* para ações pontuais ou após a instrução *durative-action* para ações com duração associada.

A Figura 5.3 apresenta o formato de um Arquivo de Domínio em uma notação semi-formal. Nesta notação, os elementos que aparecem entre os símbolos de colchetes são elementos opcionais e o símbolo de reticências é usado para omitir predicados intermediários ou declarações de parâmetros intermediários. As linhas encontram-se numeradas simplesmente para facilitar a explicação do formato do arquivo.

```

1  (define (domain DOMAIN_NAME)
2
3    (:requirements [:equality][:typing][:fluents][:durative-actions][:timed-initial-literals])
4
5    (:types TYPE_1_NAME ... TYPE_N_NAME)
6
7    (:predicates (PREDICATE_1_NAME ?PAR1 - TYPE_OF_PAR1 ... ?PARN - TYPE_OF_PARN)
8                  . . .
9                  (PREDICATE_N_NAME ?PAR1 - TYPE_OF_PAR1 ... ?PARN - TYPE_OF_PARN)
10 )
11
12    (:functions (FUNCTION_1_NAME ?PAR1 - TYPE_OF_PAR1 ... ?PARN - TYPE_OF_PARN)
13                ...
14                (FUNCTION_N_NAME ?PAR1 - TYPE_OF_PAR1 ... ?PARAM_N - TYPE_OF_PARN)
15 )
16
17    (:durative-action ACTION_1_NAME
18      [:parameters (?PAR1 - TYPE_OF_PAR1 ... ?PARN - TYPE_OF_PARN)]
19      [:duration(= ?duration (FUNCTION_NAME ?PAR1 - TYPE_OF_PAR1 ... ?PARN - TYPE_OF_PARN))]
20      [:condition (CONDITION_FORMULA)]
21      [:effect (EFFECT_FORMULA)]
22 )
23
24    (:durative-action ACTION_N_NAME
25      ...
26 )
27
28 ) ;:Comentary - end of domain definition

```

Figura 5.3 – Formato de um Arquivo de Domínio na linguagem PDDL

Na primeira linha da Figura 5.3, declara-se em `DOMAIN_NAME` um nome para o domínio de planejamento.

Na segunda linha, declara-se o conjunto de requisitos para o domínio. As declarações dos requisitos têm como propósito estabelecer as características da linguagem PDDL com as quais o planejador está habilitado a trabalhar. Com isso, evita-se a utilização de características mais avançadas da linguagem para as quais o planejador escolhido não oferece suporte.

A seguir, encontra-se listado o possível conjunto de requisitos para o domínio:

- requisito `:equality`, significa que o domínio pode utilizar o predicado `=`, interpretado como igualdade;
- requisito `:typing`, significa que o domínio utiliza tipos;
- requisito `:fluents`, permite o uso de fluentes numéricos;
- requisito `:durative-actions`, significa que o domínio aceita ações com duração;
- requisito `:timed-initial-literals`, significa que o domínio aceita eventos exógenos incondicionais determinísticos.

Na terceira linha, declaram-se os nomes dos tipos. Estes tipos são posteriormente utilizados nas instruções `:predicates` e `:functions`, presentes nas linhas 04 e 08 da Figura 5.3, no momento das declarações das variáveis que as compõem.

A sintaxe para declarações de variáveis em PDDL consiste em um símbolo que representa o nome da variável, precedido de um ponto de interrogação (?), seguido de um hífen (-) e do seu tipo. Por exemplo, na linha 04 declara-se a variável de nome `PAR1` da seguinte forma: `?PAR1 - TYPE_OF_PAR1`, onde

`TYPE_OF_PAR1` é um tipo que deve ser previamente declarado na instrução `:types` da linha 03.

Após as declarações dos predicados e das funções, a partir da linha 12 declaram-se as ações do domínio. As definições das ações utilizam os tipos, os predicados e as funções, definidas nas instruções anteriores do Arquivo de Domínio, para mapear os seus comportamentos. Durante o processo de planejamento, nem todas as ações são inseridas no plano: o planejador insere no plano somente as ações que são aplicáveis ao estado do ambiente, ou seja, as ações cujas pré-condições forem satisfeitas no instante de tempo do processo de planejamento.

Na linha 12, declarou-se uma ação com tempo associado por meio da instrução `:durative-action`. A duração desta ação consta na instrução `:duration` da linha 14 e consiste no valor numérico de retorno de uma função, cujo nome é `FUNCTION_NAME` e deve ter sido previamente declarada na instrução `:functions`. Atribui-se o valor de retorno da função a uma variável chamada `duration`.

Cada ação apresenta uma pré-condição e um efeito. Na linha 15, o elemento `CONDITION_FORMULA` pode ser: (1) uma fórmula atômica: `(PREDICATE_NAME ARG1 ... ARGN)`, onde os argumentos do predicado devem ser parâmetros da ação, ou seja, serem previamente declarados na instrução `:parameters` (linha 13) da ação; ou (2) uma conjunção de fórmulas atômicas: `(and ATOM1 ... ATOMN)`. Na linha 16, uma `EFFECT_FORMULA` pode ser: (1) a adição de um átomo: `(PREDICATE_NAME ARG1 ... ARGN)`, onde os argumentos do predicado devem ser parâmetros da ação; (2) a eliminação de um átomo: `(not (PREDICATE_NAME ARG1 ... ARGN))`; ou (3) uma conjunção de efeitos atômicos: `(and ATOM1 ... ATOMN)`.

5.4.2.3 Descrição do Arquivo de Instância do Problema

O Arquivo de Instância do Problema modela uma instância do problema de planejamento por meio do seguinte conjunto de elementos:

- Objetos – representam as instanciações dos tipos definidos no domínio. São definidos após a instrução *objects*;
- Estados iniciais – representam as descrições dos estados iniciais ao iniciar a execução do planejador. São definidos após a instrução *init*;
- Objetivo – representa o estado que se deseja alcançar após a execução do plano. É definido após a instrução *goal*.

A Figura 5.4 apresenta o formato de um Arquivo de Instância do Problema na notação semi-formal.

```
1 (define (problem PROBLEM_NAME))
2   (:domain DOMAIN_NAME)
3   (:objects OBJ1 OBJ2 ... OBJN)
4   (:init ATOM1 ATOM2 ... ATOMN)
5   (:goal CONDITION_FORMULA)
6 )
```

Figura 5.4 – Formato de um Arquivo de Instância do Problema em PDDL

Na primeira linha da Figura 5.4, declara-se em `PROBLEM_NAME` um nome para a instância do problema de planejamento.

Na segunda linha, declara-se o nome do domínio de planejamento que será utilizado pelo Arquivo de Instância do Problema.

Na terceira linha, declaram-se os objetos, que são instanciações dos tipos definidos na instrução `:types` do Arquivo de Domínio.

Na linha 04, declaram-se os estados iniciais. Os estados iniciais consistem em uma lista com todos os átomos que são verdadeiros no estado inicial do

processo de planejamento. Qualquer outro átomo não declarado é considerado falso, por definição.

Na linha 05, representa-se a descrição do objetivo por meio da fórmula `CONDITION_FORMULA`, que apresenta a mesma forma da fórmula `CONDITION_FORMULA` utilizada para descrever a condição de uma ação no Arquivo de Domínio. A única diferença é que quando `CONDITION_FORMULA` é empregada no Arquivo de Instância do Problema, os argumentos dos predicados devem ser nomes de objetos ao invés de parâmetros.

A construção da linguagem PDDL exige que todos os predicados utilizados para a descrição do objetivo sejam previamente declarados na instrução `:predicates` do Arquivo de Domínio.

5.5 Processo de Construção de Agentes de Planejamento

Conforme apresentado anteriormente, Russell e Norvig (2004) definem um agente de planejamento como sendo um agente de software capaz de construir planos de ações para um determinado problema. Este problema para o qual o agente propõe soluções consiste no cerne do processo de definição de agentes de planejamento.

Em decorrência desta importância, o processo de definição do agente de planejamento da arquitetura de planejamento proposta neste trabalho teve como foco o problema para o qual se deseja gerar planos de solução. Este processo de definição ocorreu segundo as fases:

- análise do problema de controle de satélites (levantamento dos estados e ações possíveis, objetivos, etc.);
- identificação das propriedades do ambiente de planejamento;
- escolha de uma linguagem de representação suficientemente expressiva para o tipo de ambiente de planejamento identificado;

- identificação da técnica de planejamento adequada ao tipo de ambiente de planejamento identificado;
- escolha de um algoritmo de planejamento, para atuar como mecanismo de raciocínio do agente de planejamento, que ofereça suporte às representações de conhecimento especificadas na linguagem de representação escolhida.

Em especial, a técnica de planejamento utilizada unida à linguagem de representação de problemas de planejamento escolhida formaram a base para a criação do agente de planejamento da arquitetura proposta.

Apesar do agente de planejamento ser um agente de fundamental importância para a arquitetura proposta, uma vez que ele é quem de fato gera os planos para os rastreios de satélites, o funcionamento da arquitetura não ocorre unicamente por meio da atuação deste agente, mas sim por meio do trabalho cooperativo de um grupo de agentes de software.

O Capítulo 6, a seguir, descreve os motivos pelos quais modelou-se a arquitetura proposta utilizando o conceito de agente e aborda as características relacionadas a sistemas multi-agentes, visto que a arquitetura proposta não é constituída unicamente por um agente de planejamento, mas sim por um grupo de agentes. O Capítulo 6 também descreve e compara um conjunto de metodologias de engenharia de software orientadas a agentes utilizadas para o projeto de sistemas multi-agentes e apresenta a proposta de um *framework* de avaliação destas metodologias.

6 ENGENHARIA DE SOFTWARE ORIENTADA A AGENTES

Este capítulo focaliza uma subárea da área de Engenharia de Software, chamada de Engenharia de Software Orientada a Agentes, voltada especificamente para a modelagem de sistemas multi-agentes.

Neste capítulo apresentam-se aspectos relevantes da arquitetura multi-agente de planejamento proposta, como: (1) os motivos pelos quais se concebeu a arquitetura proposta utilizando o conceito de agente; (2) as características gerais de um sistema multi-agente, e em especial, de um sistema multi-agente de planejamento, caracterizando a arquitetura proposta; e (3) o conjunto de critérios utilizado para a escolha de uma metodologia de engenharia de software orientada a agentes adequada para a modelagem da arquitetura proposta.

Este conjunto de critérios utilizado constitui um *framework* de avaliação de metodologias orientadas a agentes, proposto neste trabalho de pesquisa, voltado para a análise comparativa de metodologias utilizadas para a modelagem de sistemas de planejamento. O *framework* proposto provê critérios de avaliação essenciais à escolha de uma metodologia orientada a agentes adequada para a modelagem de sistemas multi-agentes de planejamento, contemplando os requisitos deste tipo de sistema.

6.1 Diretrizes para o Uso do Conceito de Agente

Na arquitetura proposta adotou-se o conceito de agente, um conceito que originou-se dentro da comunidade de IA nos anos 70 e infiltrou-se na comunidade de Engenharia de Software no começo dos anos 90. A adoção deste conceito embasou-se em um conjunto de diretrizes, proposto pelos participantes do projeto P907 do EURESCOM – European Institute for Research and Strategic Studies in Telecommunications (MILGROM, 2001), para identificar tipos de problemas para os quais os autores consideram

adequada, ou até mesmo vantajosa, a concepção de um sistema de software de forma orientada a agentes.

Estas diretrizes identificam propriedades existentes em softwares complexos, que caracterizam situações nas quais a adoção do conceito de agente é promissora.

O que de fato motivou a adoção do conceito de agente foi a constatação de que, não apenas uma diretriz, mas um conjunto de diretrizes propostas pelo projeto P907 do EURESCOM (MILGROM, 2001) são simultaneamente válidas para o problema de controle de satélites em análise. O conjunto de diretrizes considerado aplicável ao problema de controle de satélites encontra-se listado a seguir.

Diretriz 1 – Necessidade de Obter Múltiplos Objetivos

Existem sistemas de software que executam em situações onde não é possível especificar os seus comportamentos de forma detalhada devido à dificuldade de se enumerar, exaustivamente na fase de projeto, todas as situações que podem ocorrer durante os seus ciclos de vida. Para estes sistemas, torna-se mais fácil especificar uma série de objetivos a serem alcançados e um conjunto de restrições a serem obedecidas.

As soluções baseadas em agentes são candidatas a este tipo de sistema, pois são especificadas em termos de comportamentos orientados a objetivos. O comportamento de um agente de software não precisa ser especificado em termos do mapeamento de uma entrada para uma saída, ou seja, em termos do que ele deve fazer. Ele pode ser especificado em termos de como decidir o que fazer. Esta forma de especificar o comportamento de um agente mostra-se útil quando existem modos alternativos de se alcançar um objetivo.

A abordagem orientada a agentes leva ao desenvolvimento de sistemas mais flexíveis, que podem adaptar o seu comportamento para satisfazer objetivos em circunstâncias de mudança.

Diretriz 2 – Necessidade de Comportamento Autônomo

Os sistemas de software podem ser projetados de acordo com uma abordagem puramente reativa. Segundo esta abordagem, o sistema toma ações somente quando estas são solicitadas por meio de pedidos explícitos. Uma maneira mais flexível de projeto de sistemas consiste na abordagem direcionada a objetivos, que considera que os sistemas devem tomar ações para tentar satisfazer os seus objetivos mesmo na ausência de pedidos explícitos.

Os sistemas baseados em agentes podem se comportar mais facilmente de forma orientada a objetivos, o que lhes confere um comportamento autônomo.

Diretriz 3 – Necessidade de Alta Flexibilidade e Adaptação

Muitas vezes um sistema de software terá que ser expandido ou modificado durante a sua vida de operação ou até mesmo o propósito do sistema pode evoluir. Quando esta questão for relevante, a utilização do conceito de agente de software é aplicável devido a sua característica de modularidade. Os agentes podem ser facilmente adicionados ou substituídos, o que reduz o custo de manutenção de um sistema.

Diretriz 4 – Necessidade de Interoperabilidade

O desenvolvimento de um sistema de software orientado a agente pode ser considerado principalmente quando um problema requer que um software interaja, possivelmente no futuro, com outro software que seja ainda desconhecido durante o projeto do primeiro.

Um sistema multi-agente pode interconectar agentes que foram desenvolvidos separadamente e que são independentes de outros, permitindo que juntos

ofereçam um serviço cuja abrangência supera as capacidades individuais dos agentes.

Os sistemas legados podem ser encapsulados em agentes de software para poderem ser integrados a outros sistemas. Esta habilidade dos agentes interoperarem com outros sistemas, para os quais os agentes não tinham sido projetados antes, significa que não se precisa oferecer uma solução específica para cada nova combinação de agentes.

6.2 Definições de Engenharia de Software

Uma metodologia consiste em um conjunto de métodos empregados de forma sistemática. Um método consiste em um procedimento para se obter algo desejado.

Segundo Henderson-Sellers e Giorgini (2005), existem muitos sistemas orientados a agentes desenvolvidos por meio da aplicação de métodos, mas que carecem de uma metodologia. Estes sistemas são desenvolvidos segundo uma abordagem *ad-hoc*, que apresenta escassez de diretrizes e de uma terminologia uniforme.

A abordagem *ad-hoc* oferece flexibilidade, mas em contrapartida, torna a qualidade do sistema de software questionável, uma vez que o conhecimento e a experiência adquiridos no desenvolvimento de um sistema de software não podem ser facilmente transferidos a outros projetos. A combinação das características – presença de flexibilidade e ausência de controle – faz com que dificilmente seja possível definir uma metodologia, pois falta uma abordagem coordenada e sistemática para estabelecer métodos de trabalho.

Diante da aleatoriedade, algumas metodologias de engenharia de software orientadas a agentes foram propostas na literatura com os objetivos de: (1) prover auxílio aos desenvolvedores de software por meio da sugestão de diretrizes, de técnicas, de notações e de uma terminologia uniforme; e (2)

assegurar qualidade aos sistemas desenvolvidos por meio da garantia da aplicação coordenada de métodos.

6.3 Definição de Sistemas Multi-Agentes

Um sistema multi-agente (SMA) consiste em um sistema constituído por um grupo de agentes que atuam em conjunto no sentido de resolver problemas que estão além de suas capacidades individuais. Para prover soluções aos problemas, os agentes de um SMA interagem entre si de modo cooperativo ou competitivo para atingir objetivos comuns ao grupo ou objetivos individuais.

O termo cooperação pode ser utilizado na literatura de SMA com significados distintos. Neste trabalho de pesquisa, assim como em Rezende (2003), adota-se este termo com o seguinte significado: dois ou mais agentes que almejam atingir um mesmo objetivo em conjunto e são conscientes deste fato.

Segundo Henderson-Sellers e Giorgini (2005), muitas metodologias de engenharia de software orientadas a agentes utilizam uma metáfora da organização humana para o projeto de sistemas multi-agentes. Nesta metáfora, utilizam-se estruturas e modelos da organização humana para se projetar os SMAs e define-se que estes sistemas sejam constituídos por agentes que devem desempenhar um ou mais papéis e interagir entre si.

Assim, para satisfazer a idéia de metáfora da organização humana, muitas metodologias orientadas a agentes utilizam os conceitos de papel, de dependência social e de regras organizacionais para a modelagem de SMAs. As definições destes conceitos encontram-se especificadas, de forma particular, em cada metodologia orientada a agentes adotada.

Uma das atividades importantes destas metodologias consiste em definir os modelos de cooperação e de interação entre os agentes. Estes modelos capturam as dependências e os relacionamentos sociais entre os agentes e os papéis que estes desempenham no sistema. Estes modelos são abstratos e

podem ser concretizados por meio da implementação de protocolos de interação.

Segundo Milgrom (2001), um protocolo de interação é um padrão de comunicação estabelecido entre os agentes de um SMA. Este padrão permite que os agentes compartilhem um conjunto limitado de tipos de mensagens, cujas semânticas são bem definidas. Para estabelecer este protocolo de interação, os agentes de um SMA podem adotar uma linguagem de Comunicação de Agentes, que é constituída por ações de comunicação, uma linguagem de conteúdo e uma ontologia.

Russell e Norvig (2004) definem o conceito de ontologia como uma especificação de um conjunto de termos de um vocabulário especializado capaz de descrever um domínio. Assim, uma ontologia consiste no vocabulário de predicados, de funções e de constantes que compõem um determinado domínio. Os agentes de um domínio devem procurar “falar” a mesma linguagem e “entender” a mesma ontologia.

Existe uma organização denominada FIPA – *Foundation for Intelligent Physical Agents*, que busca desenvolver padrões para a interoperabilidade entre a comunicação de agentes. As especificações destes padrões podem ser obtidas em (FIPA, 2002).

6.3.1 Organização de Agentes

Reforçando a idéia de metáfora da organização humana, um SMA pode ser caracterizado segundo a área de estudo da Teoria da Organização, definida por Prietula et al. (1998) como: “A Teoria Organizacional pode ser caracterizada como o estudo de como múltiplos fatores influenciam o comportamento de uma organização, de pessoas e das tecnologias que a compõem”.

A literatura de sistemas multi-agentes apresenta várias definições para o conceito organização. Um estudo comparativo de um grupo de definições

importantes pode ser obtido em (HÜBNER e SICHTMAN, 2000). Rezende et al. (2003) define de forma simplificada o conceito organização de um sistema multi-agente como sendo um conjunto de restrições adotadas por um grupo de agentes para que possam atingir seus objetivos globais mais facilmente.

Do ponto de vista da psicologia social, as organizações de agentes devem ser analisadas por meio da ocorrência de interações sociais. A análise destas interações permite prever a formação dinâmica destas organizações. Os pesquisadores Conte e Castelfranchi (1992) propõem duas classes de modelos para as interações sociais e, portanto, para as organizações de agentes:

- Modelos estáticos ou descendentes (*top-down*): os agentes já possuem um problema prévio a resolver. Neste caso, uma cooperação é preestabelecida como uma hipótese de partida. As interações sociais são limitadas por uma organização preexistente, que guia os agentes para atingir o objetivo para o qual o sistema foi concebido;

- Modelos dinâmicos ou ascendentes (*bottom-up*): neste caso, os agentes não têm necessariamente um objetivo comum a atingir. As interações sociais, assim como as organizações, são estabelecidas dinamicamente como forma de atingirem seus próprios objetivos. Para estes modelos torna-se mais comum o uso dos termos coalisão ou coligação em vez do termo organização.

6.3.2 Planejamento em Sistemas Multi-Agentes

Em um sistema multi-agente de planejamento as ações de planejamento podem ser divididas em três fases – construção de planos, coordenação de planos e a sua execução. Um sistema de planejamento em um ambiente multi-agente constitui-se, portanto, de um grupo de agentes capaz de planejar, coordenar e executar planos, sendo que um agente pode executar uma ou mais destas tarefas (FERBER, 1999).

O primeiro passo consiste em gerar os planos. Esta tarefa pode ser delegada a um único agente ou ser particionada e distribuída entre um grupo de agentes.

Estas duas formas de geração de planos em ambientes multi-agentes são denominadas planejamento centralizado e planejamento distribuído, respectivamente.

Na realidade, o termo planejamento distribuído pode estar vinculado tanto à forma de geração de planos como à forma de coordenação ou execução dos planos gerados, ou a ambos (WEISS, 1999).

Relacionando-se os modos de classificação dos pesquisadores Ferber (1999) e Weiss (1999) para organização de sistemas multi-agentes de planejamento obtém-se as seguintes categorias de sistemas:

- Sistemas com planejamento e coordenação centralizados;
- Sistemas com planejamento distribuído e coordenação centralizada;
- Sistemas com planejamento e coordenação distribuídos.

Em relação à execução dos planos gerados, para cada categoria acima pode-se ter uma execução de planos de forma centralizada ou de forma distribuída entre um grupo de agentes. O caso de um sistema com planejamento, coordenação e execução centralizados representa um sistema de planejamento clássico, desprovido da característica de distribuição.

A arquitetura multi-agente de planejamento proposta neste trabalho de pesquisa caracteriza-se como um sistema de planejamento clássico.

Assim, um único agente de software da arquitetura responsabiliza-se pela tarefa de planejamento, construindo planos de ações para alcançar objetivos de rastreios. Os demais agentes que constituem a arquitetura são coordenados por um único agente, chamado de agente coordenador, e cooperam com o agente de planejamento no sentido de prover informações essenciais ao processo de planejamento. Em relação à execução dos planos gerados, a arquitetura proposta não contempla a automatização da execução de planos.

6.3.3 Projetos de Sistemas Multi-Agentes

De acordo com os participantes do projeto P907 do EURESCOM (MILGROM, 2001), os projetos de engenharia de software de sistemas multi-agentes existentes se dividem em duas categorias:

- projetos que aplicam metodologias de engenharia de software convencionais a sistemas multi-agentes, modificando-as quando necessário;
- projetos que apresentam como conceito central o conceito de agentes. Estes projetos aplicam novas metodologias de engenharia de software orientadas a agentes para cobrir as fases de análise e projeto de software.

Os projetos que se encaixam na primeira categoria buscam meios adicionais para suprir a ausência de recursos das metodologias de engenharia de software convencionais para a modelagem de agentes.

Odell et al. (2000) propõem um recurso útil a esta primeira categoria de projetos de sistemas multi-agentes. Este recurso consiste em uma notação, denominada AUML – *Agent UML*, que provê extensões à linguagem de modelagem UML (*Unified Modeling Language*). Estas extensões são capazes de representar as características essenciais aos agentes e têm como principal objetivo propor um meio de representar protocolos de interação de agentes. A linguagem AUML é empregada por muitas metodologias de engenharia de software, tanto convencionais como totalmente orientadas a agentes.

Neste trabalho de pesquisa não será descrita a linguagem UML por se tratar de uma linguagem que se tornou um padrão para a modelagem de sistemas orientados a objeto. As notações e diagramas propostos pela linguagem UML podem ser consultados em Rumbaugh et al. (2004).

A Seção 6.4 contempla os projetos de sistemas multi-agentes pertencentes à segunda categoria, ou seja, projetos que apresentam como conceito central o conceito de agentes por meio da aplicação de metodologias de engenharia de software inteiramente orientadas a agentes.

6.4 Metodologias de Engenharia de Software Orientadas a Agentes

Os agentes de software requerem um tratamento adequado para o seu projeto. As metodologias de engenharia de software convencionais não são suficientes para capturar algumas características importantes e inerentes aos agentes. Estas características incluem o comportamento autônomo, as interações interagentes e as estruturas organizacionais complexas dos sistemas de agentes.

O fato de as metodologias de engenharia de software convencionais não terem sido projetadas para representar tais características motivou o estudo de Metodologias de Engenharia de Software Orientadas a Agentes. Estas metodologias consistem em uma ramificação da área de Engenharia de Software voltada especialmente para o desenvolvimento de sistemas multi-agentes, que, portanto, contemplam as características inerentes a estes tipos de sistemas.

6.4.1 Genealogia das Metodologias Orientadas a Agentes

As metodologias orientadas a agentes possuem diversas origens. Algumas baseiam-se em idéias da área de Inteligência Artificial, outras descendem diretamente de metodologias orientadas a objetos, outras ainda baseiam-se em uma combinação destas abordagens, e outras não se fundamentam em metodologias divulgadas na literatura.

Algumas metodologias orientadas a objeto de renome serviram de base para projetos de metodologias orientadas a agentes. A Figura 6.1 ilustra as influências diretas e indiretas de um grupo de metodologias orientadas a

objetos em um grupo representativo de dez metodologias orientadas a agentes existentes na literatura, citadas em Henderson-Sellers e Giorgini (2005).

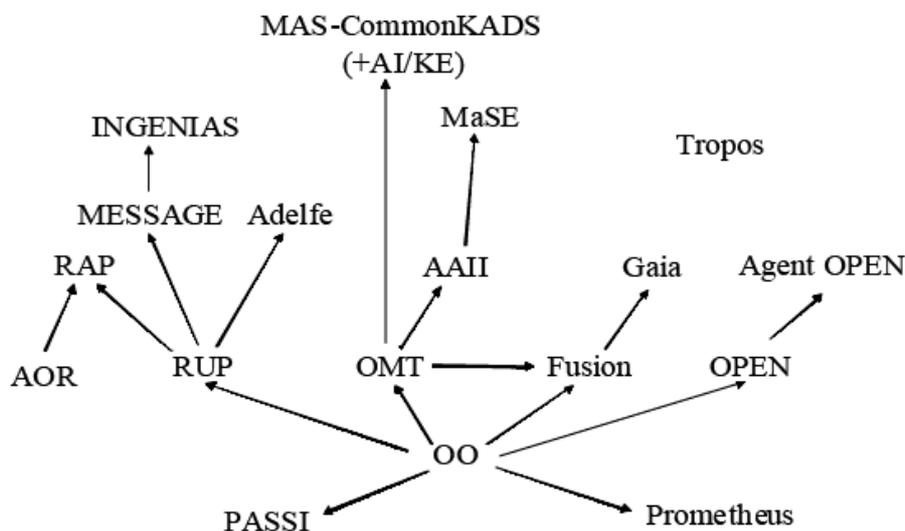


Figura 6.1 – Genealogia de um grupo de metodologias orientadas a agentes

Fonte: Retirada de Henderson-Sellers e Giorgini (2005)

A seguir, apresenta-se um resumo do histórico das influências das metodologias orientadas a objeto apresentadas na Figura 6.1:

- A metodologia orientada a agentes MaSE (WOOD e DELOACH, 2001) recebe influências da metodologia AAll de Kinny et al. (1996), que por sua vez recebeu fortes influências da metodologia orientada a objeto de Rumbaugh e amigos, a metodologia OMT (RUMBAUGH et al., 1991);
- De forma similar, a metodologia orientada a objeto Fusion (COLEMAN et al., 1994) teve forte influência no projeto da metodologia orientada a agentes Gaia (WOOLDRIDGE et al., 2000) (ZAMBONELLI et al., 2003);
- A metodologia orientada a objeto RUP formou as bases para as metodologias orientadas a agentes ADELFE (BERNON et al, 2002) e MESSAGE (CAIRE et al., 2002), que por sua vez é a base da

metodologia INGENIAS (PAVON et al., 2005). Posteriormente, a metodologia RUP juntamente com a metodologia AOR de Wagner (2003) constituíram a base para a metodologia orientada a agentes RAP (TRAVETER e WAGNER, 2005);

- A metodologia orientada a objeto OPEN foi estendida para oferecer suporte ao desenvolvimento de sistemas multi-agentes, sendo denominada a partir desta extensão de Agent OPEN (DEBENHAM e HENDERSON-SELLERS, 2003);
- As metodologias PASSI (COSSENTINO, 2005) e Prometheus (PADGHAM e WINIKOFF, 2003) não descendem de uma metodologia orientada a objetos específica, mas recebem influências do paradigma de orientação a objetos, sugerindo inclusive a utilização de diagramas deste paradigma na notação da linguagem UML;
- A metodologia MAS-CommonKADS (IGLESIAS et al., 1998) é uma metodologia baseada na área de IA, mas também é influenciada por metodologias orientadas a objeto, notavelmente a metodologia OMT;
- Existem metodologias que não possuem nenhuma ligação genealógica direta com outras metodologias, sejam estas orientadas a objeto ou orientadas a agentes. Este é o caso da metodologia Tropos (PERINI et al., 2002), (BRESCIANI et al., 2004), que pelo fato de utilizar a linguagem de modelagem i* (YU, 1995) apresenta uma aparência que lhe é peculiar, diferente das metodologias que utilizam a linguagem de modelagem AUML (ODELL et al., 2000).

De forma geral, todas as metodologias orientadas a agentes citadas acima oferecem contribuições valiosas que auxiliam o projeto de sistemas baseados em agentes. No entanto, cada metodologia possui uma perspectiva única para o desenvolvimento deste tipo de sistema, enquanto ao mesmo tempo possui características que se sobrepõem às características de outras metodologias.

Na verdade, nenhuma metodologia em especial pode ser considerada ideal para ser aplicada em toda situação de desenvolvimento de sistemas.

Assim, para a concepção da arquitetura proposta neste trabalho, realizou-se um estudo detalhado de um conjunto de três das metodologias orientadas a agentes apresentadas acima a fim de identificar uma metodologia mais adequada ao projeto da arquitetura do sistema de planejamento para o ambiente de controle de satélites.

Buscou-se analisar um conjunto de metodologias de origens distintas para ser possível verificar os pontos fracos e fortes de cada tipo de metodologia. O conjunto de metodologias analisadas, apresentadas nas seções a seguir, é constituído pelas metodologias MAS-CommonKADS, baseada na área de IA; Tropos, não descendente de uma metodologia específica; e MESSAGE, baseada na metodologia orientada a objeto RUP.

6.4.2 Metodologia MAS-CommonKADS

A metodologia MAS-CommonKADS (IGLESIAS et al., 1998), (IGLESIAS e GARIJO, 2005) é uma extensão da metodologia CommonKADS que engloba aspectos relevantes aos sistemas multi-agentes.

CommonKADS tornou-se, principalmente na Europa, uma referência no desenvolvimento de sistemas baseados em conhecimento (WERNECK et al., 2006). Na metodologia CommonKADS são definidos vários modelos, sendo o Modelo de Experiência o modelo central da metodologia.

Este modelo tem como objetivo modelar o conhecimento de resolução de problemas empregado por um agente para realizar uma tarefa, dividindo o conhecimento da aplicação em três níveis: nível do domínio (conhecimento declarativo sobre o domínio), nível de inferência (uma biblioteca de estruturas genéricas de inferência) e nível de tarefa (ordem das inferências). O Modelo de Experiência descreve o conhecimento necessário aos agentes para estes atingirem seus objetivos.

Esta metodologia define outros seis modelos (IGLESIAS et al., 1998), (IGLESIAS e GARIJO, 2005): Modelo de Agentes, Modelo de Organização, Modelo de Tarefas, Modelo de Comunicação, Modelo de Coordenação e Modelo de Projeto.

O Modelo de Agentes especifica as características de um agente: sua capacidade de raciocínio, habilidades, serviços, sensores, grupo de agentes ao qual pertence e classe de agente. Um agente pode ser um agente humano, de software, ou qualquer entidade capaz de empregar uma linguagem de comunicação de agentes.

O Modelo de Organização é uma ferramenta para analisar a organização humana em que o sistema multi-agente vai ser introduzido e para descrever a organização dos agentes de software e sua relação com o meio. O Modelo de Tarefas descreve as tarefas que os agentes podem realizar, os objetivos de cada tarefa, sua decomposição e os métodos de resolução de problemas para atingir cada objetivo.

O Modelo de Comunicação descreve as interações entre um agente humano e um agente de software. Centra-se na consideração de fatores humanos para as interações. O Modelo de Coordenação descreve as interações entre agentes de software. Finalmente, o Modelo de Projeto descreve a arquitetura e o projeto do sistema multi-agente como passo prévio a sua implementação, sendo o único modelo a não tratar da fase de Análise.

O modelo de ciclo de vida para o desenvolvimento de sistemas multi-agentes (IGLESIAS et al., 1998), (IGLESIAS e GARIJO, 2005) segue a gestão de projetos de Common-KADS. Este modelo engloba as seguintes fases: Conceituação, Análise, Projeto, Codificação, Integração, e Operação e Manutenção.

A fase de Conceituação consiste na tarefa de elicitação para obter uma primeira descrição do problema e a determinação dos casos de uso que podem

ajudar a entender os requisitos informais e a testar o sistema. A fase de Análise determina os requisitos do sistema partindo do enunciado do problema. Durante esta fase se desenvolvem os seguintes modelos: Organização, Tarefas, Agentes, Comunicação, Coordenação e Experiência.

Na fase de Projeto define-se como os requisitos da fase de Análise podem ser realizados, determinando no Modelo de Projeto as arquiteturas tanto da rede multi-agente como de cada agente. Na fase de Codificação cada agente é implementado e testado. Por fim, na fase de Integração testa-se o sistema completo.

6.4.3 Metodologia Tropos

Perini et al. (2002) acreditam que para a construção de sistemas multi-agentes, que interagem com agentes humanos e de software, é necessário modelar os processos de coordenação que ocorrem na organização social, estabelecendo claramente onde o sistema deve ser introduzido.

Segundo estes autores, as especificações de coordenação devem estar embasadas em uma análise profunda das dependências intencionais dos agentes, que podem ser modeladas em termos de objetivos, planos e dependências de recursos entre pares de agentes.

Assim, Perini et al. (2002) propõem uma metodologia de engenharia de software orientada a agentes, denominada Tropos, que tem como base duas idéias principais:

- 1) a coordenação entre agentes de software e agentes humanos deve ser projetada em passos, especificando-se os requisitos;
- 2) esta coordenação deve ser modelada como uma dependência entre atores. Dois atores dependem um do outro para atingir objetivos, adquirir recursos ou executar um plano.

A metodologia Tropos trabalha com os conceitos de agentes, objetivos, planos, e atividades de especificação de requisitos e projeto. Baseia-se na construção de um Modelo Conceitual, que é refinado e estendido a partir de um Modelo de Requisitos até a obtenção de artefatos executáveis, ao longo de cinco fases de desenvolvimento: Análise de Requisitos Prévia, Análise de Requisitos, Projeto Arquitetural, Projeto Detalhado e Implementação.

Para a construção do Modelo Conceitual, usando uma linguagem visual própria que oferece suporte aos conceitos de ator, de objetivo e de dependência, a metodologia Tropos propõe o uso de dois diagramas:

- Diagrama de Atores para descrever a rede de relacionamentos de dependência entre atores;
- Diagramas de Objetivos para ilustrar a análise de planos e objetivos a partir da visão de um ator específico.

No Diagrama de Atores, um ator chamado de atendente (*dependee*), depende de outro ator, chamado de requisitante (*dependor*), para:

- obter um objetivo. Por meio do relacionamento de Dependência de Objetivos um objetivo é delegado pelo ator requisitante para o ator atendente, que por sua vez decide de forma autônoma como satisfazer o objetivo;
- executar um plano. O relacionamento de Dependência de Planos especifica que o ator requisitante solicita ao ator atendente para executar um plano específico. O controle de execução é delegado ao ator atendente;
- explorar um recurso. O relacionamento de Dependência de Recursos modela o pedido de um ator requisitante a um ator atendente de um recurso específico.

O Diagrama de Atores é construído na fase de Análise de Requisitos Prévia e, em seguida, refinado na fase de Análise de Requisitos, considerando-se todos os atores e os seus objetivos. Esse refinamento dá origem ao Diagrama de Objetivos, construído para cada ator especificado no Diagrama de Atores.

Na fase de Análise de Requisitos, busca-se descobrir novos processos de coordenação e adicionar novos atores e dependências ao modelo. Nesta fase, procura-se identificar possíveis subsistemas de suporte à decisão, introduzindo-os como novos atores no Modelo Conceitual.

Na fase de Projeto Arquitetural define-se a arquitetura global do sistema em termos de subsistemas representados como atores. As dependências entre atores descrevem os processos de coordenação entre os componentes do sistema.

Na fase de Projeto Detalhado os agentes são minuciosamente especificados em termos de seus planos, de suas capacidades e das interações entre eles. Nesta fase, os autores Perini et al. (2002) exploram as especificações de protocolos de comunicação de agentes existentes na literatura sobre sistemas multi-agentes e sugerem o uso de um conjunto de diagramas da notação AUML (ODELL et al., 2000).

Para as fases de Análise de Requisitos, a metodologia Tropos adota o *framework* de modelagem i^* (YU, 1995). Na realidade, a idéia de representar os clientes de um sistema como atores, que dependem uns dos outros para alcançar objetivos, para realizar tarefas, e para consumir e produzir recursos é obtida deste *framework*. O *framework* i^* também inclui um modelo de dependência estratégica para descrever a rede de interdependências entre atores, que é utilizado na metodologia Tropos com o nome de Diagrama de Atores.

6.4.4 Metodologia MESSAGE

O instituto europeu de pesquisas na área de telecomunicações EURESCOM – European Institute for Research and Strategic Studies in Telecommunications – direcionou um dos seus projetos, o P907 (MILGROM, 2001) (CAIRE et al., 2002), para a definição de uma metodologia de análise e projeto baseada em agentes com o objetivo de aplicar esta metodologia no ambiente de telecomunicações. A metodologia definida recebeu o nome de MESSAGE – Methodology for Engineering Systems of Software Agents.

MESSAGE (CAIRE et al., 2002) utiliza a linguagem UML – *Unified Modeling Language* (RUMBAUGH et al., 2004) como ponto de partida, adicionando conceitos e relacionamentos necessários à modelagem orientada a agentes. Esta modelagem busca descrever a forma como um sistema multi-agente funciona para a realização de um objetivo coletivo, similarmente às organizações humanas e sociedades, e o comportamento cognitivo dos agentes, com o auxílio da área de Inteligência Artificial.

Na realidade, a metodologia MESSAGE propõe extensões ao MOF, a linguagem de meta-modelo da UML que lhe confere extensibilidade. MESSAGE estende o meta-modelo da UML com novos conceitos de entidades orientadas a agentes. A maioria destes conceitos de entidades podem ser classificados em três categorias: categoria de Entidade Concreta, categoria de Atividade e categoria de Estado Mental. A primeira categoria inclui os conceitos de Agente, Organização, Papel e Recurso. A segunda categoria contém os conceitos de Tarefa, Interação e Protocolo de Interação. A terceira tem um foco maior no conceito de Objetivo.

6.4.4.1 Conceitos da Metodologia MESSAGE

Descrição dos Conceitos da Categoria Entidade Concreta

~~Conceito de Agente~~ – consiste em uma entidade autônoma atômica, capaz de realizar alguma função útil. Esta competência funcional é representada pelos

serviços do agente, que são análogos às operações de um objeto. A autonomia significa que as ações do agente não são somente determinadas por eventos externos ou interações, mas também por sua própria motivação. Esta motivação é referenciada como um atributo chamado propósito, que influencia em como um agente reagirá a um pedido para a execução de um determinado serviço e em como este será provido. Neste âmbito, os agentes de software e os agentes humanos são especializações do conceito de Agente.

Conceito de Organização – consiste em um grupo de agentes que trabalham em conjunto para o alcance de um propósito comum. Trata-se de uma entidade virtual no sentido de que não possui uma entidade computacional individual que a codifique. Os agentes que a constitui provêem os serviços da Organização e alcançam os seus objetivos de forma coletiva. Estes agentes conectam-se entre si por meio de relacionamentos organizacionais, como os existentes em uma organização humana, tal como o de superior-subordinado, e por meio de mecanismos de coordenação, expressos pelas interações entre agentes.

Conceito de Papel – um Papel descreve as características externas de um agente em um contexto particular. A distinção entre os conceitos de Papel e Agente é análoga a existente entre a Classe de um objeto e sua Interface. Um agente pode desempenhar diferentes papéis e vários agentes podem desempenhar um mesmo papel, visto que este descreve as características externas de um agente em um contexto particular.

Conceito de Recurso – representa entidades que não são autônomas, como bases de dados e programas externos utilizados pelos agentes. A metodologia MESSAGE considera adequados os conceitos padronizados da orientação a objetos para a modelagem de recursos.

Descrição dos Conceitos da Categoria Atividade

Conceito de Tarefa – consiste em uma unidade de atividade, que possui um único executor primário e estados definidos por pré-condições e pós-condições. Para uma tarefa ser realizada, suas pré-condições devem ser válidas e pode-se esperar, após o término de mesma, que as pós-condições sejam satisfeitas.

Conceitos de Interação e Protocolo de Interação – o conceito de Interação está fortemente relacionado ao introduzido pela metodologia Gaia (ZAMBONELLI et al., 2003). Uma interação possui um ou mais participantes e um propósito que estes pretendem atingir em conjunto. Este propósito geralmente consiste na obtenção de uma visão consistente de algum aspecto do domínio do problema, no estabelecimento dos termos de um serviço ou na troca de resultados de um ou mais serviços. Um protocolo de interação define um padrão para a troca de mensagens existente em uma interação.

Descrição do Conceito da Categoria Estado Mental

Conceito de Objetivo – este conceito associa o agente a um estado de forma que este pretende causar o estado relacionado ao objetivo. Alguns objetivos são derivados do propósito do agente e são intrínsecos a sua identidade, sendo persistentes por toda a vida útil deste.

Descrição de Outros Conceitos Importantes da MESSAGE

Conceito de Mensagem – representa um objeto comunicado de um agente a outro. A transmissão de uma mensagem tem um tempo finito de duração e requer que uma ação seja executada entre um agente emissor e um agente receptor. Uma Mensagem é composta por atributos que especificam o emissor, o receptor, a categorização conforme o plano do emissor (que categoriza a Mensagem em termos da intenção do emissor) e o conteúdo (uma Entidade de Informação).

Conceito de Entidade de Informação – consiste simplesmente em um objeto que encapsula um conjunto de informações.

6.4.4.2 Visões do Modelo de Análise

O modelo de análise consiste em uma complexa rede de classes e instâncias inter-relacionadas, derivadas dos conceitos da metodologia. A metodologia MESSAGE (CAIRE et al., 2002) propõe que este modelo seja estruturado em visões a fim de oferecer uma técnica que possibilite ao analista focalizar nos conceitos da metodologia e, conseqüentemente, descrever os vários aspectos de um sistema.

A metodologia propõe a utilização de um conjunto de visões, dentre as quais se destacam: a visão Organizacional, a visão de Objetivo e a visão de Agente. Os autores da MESSAGE recomendam que se assegure a consistência entre as diferentes visões para que a técnica de estruturação em visões seja bem sucedida.

A Visão Organizacional representa as entidades concretas no sistema e no seu ambiente, e os relacionamentos entre estas entidades, como os relacionamentos de agregação e de conhecimento. O relacionamento de conhecimento indica que há pelo menos uma interação entre as entidades. As entidades podem ser agentes, organizações, papéis ou recursos.

A Visão de Objetivo representa um objetivo e as dependências entre seus sub-objetivos. É possível representar um objetivo como a composição de um conjunto de sub-objetivos, em um nível inferior do diagrama de representação desta visão.

A Visão de Agente descreve, por meio de um esquema de agente, as seguintes características de cada agente: os objetivos, os eventos percebidos, os recursos controlados, as regras de comportamento e as tarefas que o agente sabe realizar.

Na metodologia MESSAGE considera-se uma abordagem *top-down* para a definição da estrutura do sistema multi-agente. Para verificar se os comportamentos destes agentes não apresentam desvios prejudiciais ao sistema como um todo, deve-se fazer uma análise *bottom-up*. Assim, o comportamento do sistema será avaliado em relação aos seus requisitos, de forma que, caso aspectos indesejáveis sejam encontrados, a análise *top-down* deva ser revisada.

6.5 Frameworks de Avaliação de Metodologias Orientadas a Agentes

Quando se deseja adotar uma metodologia de engenharia de software orientada a agentes deve-se estar ciente dos potenciais benefícios desta adoção e de o quão adequável é a abordagem escolhida para os propósitos pretendidos. Desta forma, para a concepção da arquitetura proposta neste trabalho buscou-se analisar cuidadosamente as características das metodologias existentes a fim de identificar a metodologia mais adequada para a modelagem do problema de controle de satélites em análise.

Na literatura sobre Engenharia de Software Orientada a Agentes existem modelos, chamados de *frameworks* de avaliação, que oferecem suporte à avaliação de metodologias orientadas a agentes. Um *framework* de avaliação define critérios que permitem: (1) analisar as características das metodologias; e (2) compará-las entre si por meio da métrica de satisfação ou não-satisfação dos requisitos definidos pelos critérios.

Os seguintes autores propõem *frameworks* de avaliação de metodologias orientadas a agentes:

- Sturm e Shehory (2003) definem um *framework* de avaliação quantitativa e qualitativa, e o aplica na avaliação das metodologias Gaia, Adept e Desire por meio da modelagem de um estudo de caso de Leilão;

- Dam e Winikoff (2003) utilizam um *framework* de avaliação baseado em atributos e em um questionário respondido por pesquisadores da área, especificamente pelos autores das metodologias analisadas e por alunos que modelaram um estudo de caso de Planejamento de Itinerário Pessoal. A avaliação comparativa utilizando este *framework* foi realizada para as metodologias Gaia, MaSE, MESSAGE, Prometheus e Tropos;
- Tran et al. (2003), propõem um *framework* de avaliação que contém critérios que analisam categorias de características presentes nas metodologias de forma a possibilitar a análise dos três componentes de uma metodologia – processo, modelos e técnicas.

Este último *framework* de avaliação de metodologias, proposto por Tran et al. (2003), foi constituído a partir da integração de critérios de avaliação de vários *frameworks* existentes. Em especial, este *framework* inclui os critérios de avaliação propostos por Sturm e Shehory (2003), além de adicionar novos critérios, como por exemplo os critérios: “a abordagem de desenvolvimento de sistemas multi-agentes que a metodologia oferece suporte”, “suporte a agentes móveis” e “suporte a ontologia”.

Os critérios propostos pelo *framework* de Tran e co-autores dividem-se em quatro categorias:

- critérios relacionados ao processo. Analisam os seguintes aspectos de uma metodologia: sua aplicabilidade, os passos existentes para o seu processo de desenvolvimento e a abordagem de desenvolvimento adotada;
- critérios relacionados às técnicas. Avaliam a usabilidade das técnicas oferecidas por uma metodologia para a execução dos passos do seu processo de desenvolvimento e/ou para o desenvolvimento dos seus modelos;

- critérios relacionados aos modelos. Avaliam os seguintes aspectos dos modelos de uma metodologia: os conceitos representados pelos modelos, a qualidade das notações e as características de agentes que os modelos oferecem suporte;
- critérios relacionados às características de suporte. Avaliam características suplementares de alto nível de uma metodologia orientada a agentes, como por exemplo, as características de disponibilidade de ferramentas de software e de suporte a ontologia.

Como este framework propõe um vasto conjunto de critérios para cada categoria, estes não serão listados, mas encontram-se disponíveis para consulta em Tran et al. (2003).

6.6 Framework AMSisPlan

Em virtude do *framework* de Tran et al. (2003) oferecer uma vasta abrangência de critérios, escolheu-se este framework para identificar a metodologia orientada a agente mais adequada para a modelagem do sistema multi-agente proposto neste trabalho.

Outros autores também consideraram abrangente o *framework* de Tran et al. (2003) e o utilizaram como suporte à avaliação de metodologias orientadas a agentes. Henderson-Sellers e Giorgini (2005) realizaram uma análise comparativa interessante, por meio do *framework* de Tran et al. (2003), de um grupo de metodologias orientadas a agentes, incluindo as metodologias Tropos, MAS-CommonKADS e MESSAGE. O resultado desta análise comparativa foi a proposta de um *framework* voltado a auxiliar projetistas de sistemas multi-agentes na seleção, entre o grupo de metodologias consideradas pelos autores, da metodologia orientada a agentes mais apropriada para qualquer tipo de aplicação.

No entanto, os próprios autores Henderson-Sellers e Giorgini (2005) afirmam:

- o *framework* por eles proposto auxilia na escolha da metodologia mais adequada, mas a escolha da melhor metodologia depende da aplicação;
- cada aplicação apresenta um determinado conjunto de requisitos, que indicam quais são os critérios de avaliação mais relevantes do *framework* de Tran et al. (2003) para a aplicação;
- a escolha da melhor metodologia para uma determinada aplicação deve preferencialmente satisfazer a estes critérios.

Com base nas afirmações de Henderson-Sellers e Giorgini (2005), para a escolha de uma metodologia para o projeto de um sistema multi-agente de planejamento, objeto de estudo deste trabalho, verificou-se a necessidade de restringir o conjunto de critérios de avaliação de metodologias de Tran et al. (2003) a um subconjunto que retrate diretamente sistemas de planejamento.

Assim, tendo como base os requisitos de sistemas multi-agentes de planejamento, propõe-se neste trabalho de pesquisa um subconjunto de critérios, extraídos do conjunto de critérios do *framework* de Tran et al. (2003), que possibilite uma avaliação de metodologias orientadas a agentes voltada aos requisitos dos sistemas de planejamento. Ao subconjunto de critérios de avaliação propostos neste trabalho dá-se o nome de *framework* de Avaliação de Metodologias para sistemas de Planejamento, referenciado pela sigla *framework* AMSisPlan.

6.6.1 Critérios de Avaliação de Metodologias do *Framework* AMSisPlan

Os critérios de avaliação de metodologias orientadas a agentes agrupados na Tabela 6.1, extraídos do conjunto de critérios do *framework* de Tran et al. (2003), constituem o *framework* AMSisPlan. O objetivo do *framework* AMSisPlan consiste em sintetizar um conjunto de critérios, considerados relevantes, para a escolha de uma metodologia orientada a agentes para a modelagem de sistemas de planejamento.

Tabela 6.1 – Descrição dos critérios de avaliação do *framework* AMSisPlan

Critério	Descrição do Critério
01. Suporte ao conceito de papel	A metodologia emprega o conceito de “papel” na análise e projeto de sistemas multi-agentes? Caso afirmativo, qual é a abordagem adotada para a identificação de papéis: abordagem orientada a organização, abordagem orientada a comportamento ou abordagem orientada a objetivos?
02. Suporte às atitudes mentais dos agentes	A metodologia oferece suporte à definição das atitudes mentais dos agentes, ou seja, seus objetivos, compromissos, etc.?
03. Suporte à arquitetura do sistema multi-agente	A metodologia oferece suporte à especificação da arquitetura do sistema, permitindo esboçar uma visão geral dos componentes do sistema e suas conexões?
04. Suporte à estrutura organizacional	A metodologia contempla a especificação da estrutura organizacional do sistema ou dos relacionamentos sociais entre agentes?
05. Suporte a objetos	A metodologia permite a utilização ou a integração de objetos convencionais em um sistema multi-agente?
06. Suporte à ontologia	A metodologia oferece suporte à utilização e à especificação de uma ontologia para um problema de planejamento?
07. Suporte à habilidade de comunicação	A metodologia oferece suporte à modelagem do comportamento cooperativo ou competitivo de um grupo de agentes, realizado por meio da comunicação entre eles, para alcançar objetivos de planejamento?
08. Suporte à característica de autonomia	A metodologia oferece suporte à característica de autonomia de um agente?
09. Fornecimento de exemplos	A metodologia oferece exemplos de aplicação das técnicas que propõe?
10. Suporte de software	Existem softwares que oferecem suporte à construção dos modelos que a metodologia propõe?

O critério 01 analisa a abordagem adotada por uma metodologia para a identificação de papéis. Esta abordagem pode ser: orientada a organização, orientada a comportamento ou orientada a objetivos.

A abordagem orientada a organização focaliza na análise de propriedades gerais, como: estrutura do sistema, serviços oferecidos, objetivos e tarefas gerais, papéis principais e recursos. Durante o processo de refinamento, identificam-se os agentes necessários à realização dos objetivos e analisam-se as cooperações, e os possíveis conflitos e suas resoluções.

A abordagem orientada a comportamento focaliza no refinamento progressivo de cenários de interação, que caracterizam os comportamentos internos e externos dos agentes do sistema. Estes cenários comportam-se como fontes para a caracterização de tarefas, objetivos, mensagens, protocolos e entidades do domínio.

A abordagem orientada a objetivos baseia-se na decomposição funcional. Esta abordagem tem como foco a definição dos agentes necessários ao provimento das funcionalidades do sistema. Os papéis e objetivos do sistema são sistematicamente analisados para que sejam determinadas as condições de decisão e os métodos para resolução de problemas.

6.7 Avaliação de Metodologias utilizando o Framework Proposto

As metodologias avaliadas consistem nas três metodologias apresentadas nas Seções 6.4.2, 6.4.3 e 6.4.4: metodologia MAS-CommonKADS, metodologia Tropos e metodologia MESSAGE, respectivamente. A escolha destas metodologias deve-se ao fato de apresentarem origens distintas, o que lhes confere características diversificadas.

Em decorrência desta diversidade, a avaliação das metodologias parte do pressuposto de que, para um determinado tipo de aplicação, uma metodologia possa ser mais adequada em relação às outras. Portanto, o objetivo da avaliação, efetuada por meio de uma análise comparativa entre as metodologias, consiste em identificar a metodologia mais facilmente aplicável a sistemas de planejamento, objeto de estudo deste trabalho.

A Tabela 6.2 apresenta os resultados da avaliação das três metodologias aplicando-se o *framework* AMSisPlan. As palavras “Sim” e “Não” indicam satisfação e não-satisfação, respectivamente, de um determinado critério pela metodologia avaliada. Define-se que uma metodologia satisfaz a um critério quando esta contempla os requisitos de avaliação, ou seja, as características desejadas para a modelagem de sistemas, definidos pelo critério. As palavras “Baixo”, “Médio” e “Alto” indicam os níveis de intensidade com que uma metodologia satisfaz um determinado critério.

Tabela 6.2 – Avaliação das metodologias usando o *framework* AMSisPlan

Critério	MAS-CommonKADS	Tropos	MESSAGE
01. Suporte ao conceito de papel	Não	Não	Sim Orientada a objetivos e a comportamento
02. Suporte às atitudes mentais dos agentes	Médio	Médio	Alto
03. Suporte à arquitetura do sistema multi-agente	Não	Médio	Alto
04. Suporte à estrutura organizacional	Alto	Baixo	Alto
05. Suporte a objetos	Não	Não	Sim
06. Suporte à ontologia	Não	Sim	Sim
07. Suporte à habilidade de comunicação	Sim	Sim	Sim
08. Suporte à característica de autonomia	Sim	Sim	Sim
09. Fornecimento de exemplos	Sim	Sim	Sim
10. Suporte de software	Não	Não	Sim

Conforme apresentado na Tabela 6.2, as três metodologias avaliadas equiparam-se quanto à satisfação dos critérios 07, 08 e 09. Em relação à satisfação do critério 06, a metodologia MAS-CommonKADS fica em desvantagem em relação às outras duas metodologias por não satisfazer a este critério. No entanto, em relação ao critério 04, a desvantagem fica para a metodologia Tropos. Observa-se que a metodologia MESSAGE satisfaz tanto o

critério 06 quanto o critério 04. Em relação aos critérios 01, 02, 03, 05 e 10, a metodologia MESSAGE satisfaz a estes critérios enquanto as outras metodologias não os satisfazem ou satisfazem-nos em níveis de intensidade inferiores.

Assim, os resultados da avaliação das metodologias MAS-CommonKADS, Tropos e MESSAGE, por meio do *framework* AMSisPlan, indicam a metodologia MESSAGE como a mais adequada à modelagem de sistemas multi-agentes de planejamento.

Esta verificação conduziu a escolha da metodologia MESSAGE para o projeto da arquitetura multi-agente de planejamento proposta neste trabalho de pesquisa. O Capítulo 7, a seguir, apresenta a arquitetura proposta, voltada para o problema de planejamento de controle de satélites.

7 ARQUITETURA MAPSAT

Este capítulo apresenta a arquitetura multi-agente de planejamento proposta neste trabalho de pesquisa para o problema de planejamento de controle de múltiplos satélites. A arquitetura de software proposta, chamada de arquitetura *Multi-Agente de Planejamento de Controle de Satélites* (arquitetura MAPSat), possui o conceito de agente como o elemento principal de seu projeto arquitetural.

Neste capítulo apresentam-se a estrutura e a organização de agentes da arquitetura MAPSat e, em seqüência, apresentam-se os modelos da metodologia MESSAGE (CAIRE et al., 2001) construídos para o projeto orientado a agentes da arquitetura. Durante as definições dos agentes que compõem a arquitetura e do detalhamento de suas características, na Visão de Agente da metodologia MESSAGE, os comportamentos dos agentes são exemplificados.

Na última seção apresentam-se os resultados obtidos com a construção de um protótipo da arquitetura MAPSat a fim de demonstrar o funcionamento da arquitetura. Em resumo, os agentes da arquitetura MAPSat atuam, de forma cooperativa, para planejar operações de vôo para um grupo de satélites, considerando o uso compartilhado de estações terrenas de rastreamento.

7.1 Estrutura e Organização de Agentes da Arquitetura MAPSat

A arquitetura MAPSat considera o conceito de agente como o elemento principal de seu projeto arquitetural. A adoção do conceito de agente embasou-se no conjunto de diretrizes proposto pelos participantes do projeto P907 do EURESCOM, detalhados no Capítulo 6. Em síntese, as principais razões para se conceber a arquitetura MAPSat utilizando o conceito de agente são as capacidades que os agentes possuem de exibir comportamento direcionado a

objetivos, de atuação autônoma, e de se adaptar flexivelmente a mudanças de requisitos ou no ambiente (MILGROM, 2001).

Os agentes da arquitetura são considerados agentes especialistas, pois têm como objetivo o provimento de serviços específicos. Além disso, apresentam relacionamentos de dependência funcional entre si. Um relacionamento de dependência funcional caracteriza-se pela necessidade de um agente em se beneficiar do provimento de uma funcionalidade (serviço) de outro agente. A arquitetura MAPSat apresenta uma estrutura hierárquica baseada nos relacionamentos de dependência funcional dos agentes que a constitui.

A estrutura hierárquica da arquitetura MAPSat é constituída por:

- um agente de planejamento, provê o serviço de planejamento, construindo planos de ações para alcançar objetivos de rastreios;
- agentes humanos e agentes de software, cooperam com o agente de planejamento no sentido de gerar e prover informações essenciais ao processo de planejamento e;
- um agente coordenador, coordena as interações entre os agentes da arquitetura.

Em relação à organização de agentes da arquitetura MAPSat, esta adota o modelo estático, sendo caracterizada, portanto, como uma Organização de Agentes Estática. O modelo estático de organização de agentes encontra-se definido no Capítulo 6.

Esta adoção deve-se ao fato de os agentes da arquitetura possuírem um problema prévio a resolver, que consiste no planejamento de controle de múltiplos satélites com períodos de rastreios que podem ser determinados previamente em relação ao processo de planejamento. Apesar da possibilidade de existência de conflitos entre os períodos de rastreios de dois ou mais satélites, estes podem ser detectados previamente, o que possibilita aos

agentes da arquitetura estabelecerem cooperações pré-determinadas como ponto de partida para a geração das informações de entrada para o processo de planejamento.

Assim, na arquitetura MAPSat as interações sociais entre os agentes são limitadas por uma organização pré-existente, que guia os agentes para atingir o objetivo para o qual a arquitetura foi concebida.

7.2 Modelos da Arquitetura MAPSat

A arquitetura MAPSat foi modelada adotando-se o processo de desenvolvimento de sistemas orientado a agentes proposto pela metodologia MESSAGE (CAIRE et al., 2001). Esta metodologia sugere que os modelos de análise de um sistema sejam produzidos por meio de refinamentos sucessivos. Segundo MESSAGE, o processo de modelagem inicia-se com a construção das visões Organizacional e de Objetivos, sendo que ambas provêm a base para a construção da Visão de Agente. As seções subseqüentes apresentam os modelos de análise construídos para a arquitetura MAPSat.

7.2.1 Visão Organizacional

A Visão Organizacional considera o sistema a ser desenvolvido como uma caixa preta, focalizando os relacionamentos do sistema com as entidades do ambiente no qual o sistema está inserido. Estes relacionamentos podem ser expressos por meio de dois diagramas: um diagrama que contempla relacionamentos estruturais e outro que contempla relacionamentos de interação.

A Figura 7.1 apresenta o diagrama de Relacionamentos Estruturais da Visão Organizacional para o ambiente de controle de satélites, ambiente no qual a arquitetura MAPSat encontra-se inserida.

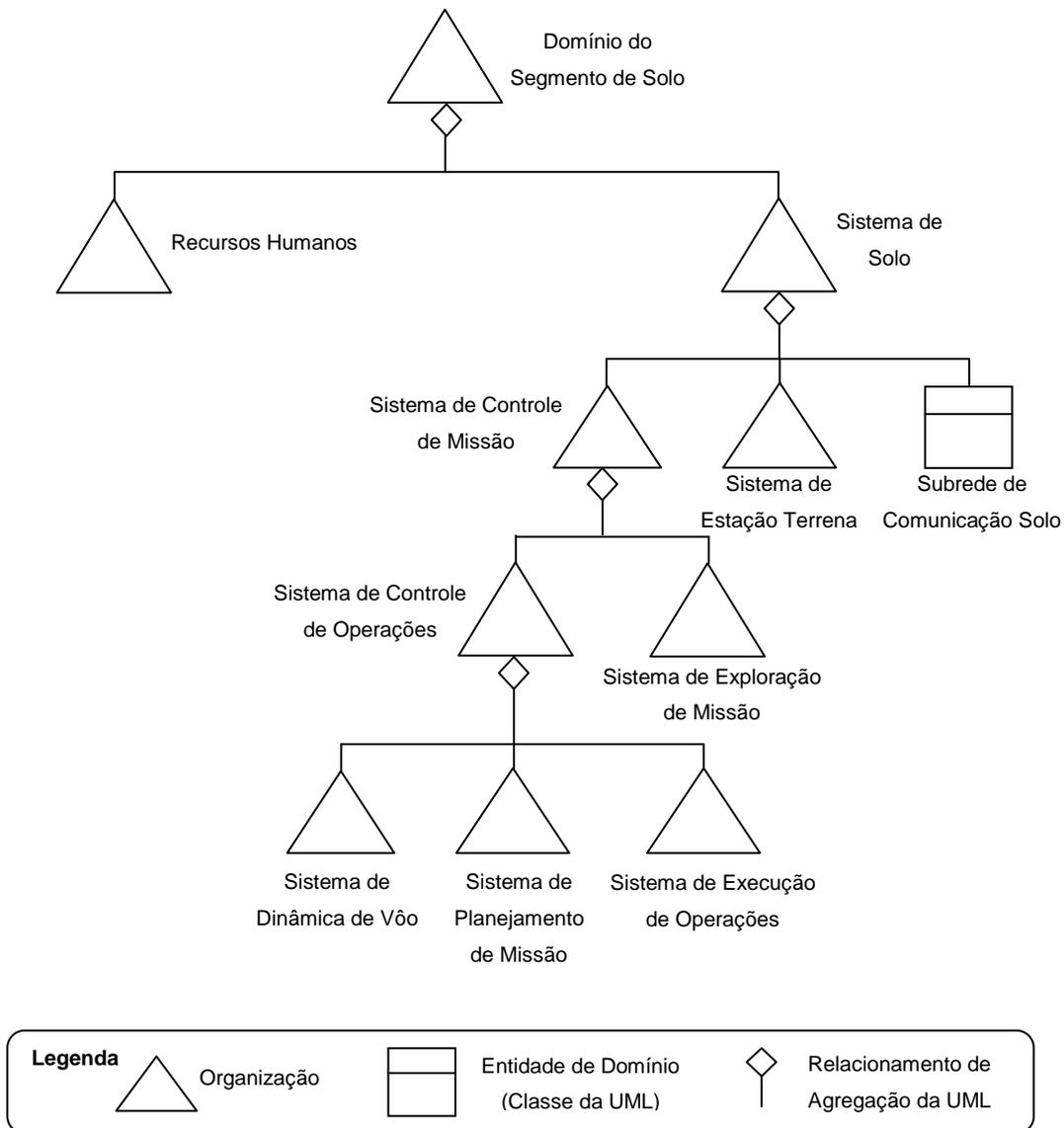


Figura 7.1 – Diagrama de relacionamentos estruturais da visão organizacional

Os relacionamentos estruturais presentes no diagrama da Figura 7.1, modelados por meio de relacionamentos de agregação da UML, demonstram a estrutura do Segmento de Solo, constituído por: (1) Recursos Humanos especialistas em operações para o controle de missões espaciais; e (2) um Sistema de Solo, constituído por hardware, software e instalações, que oferece suporte às atividades dos especialistas humanos.

O Sistema de Solo é constituído por três tipos de sistemas:

- Sistema de Controle de Missão, planeja, monitora e controla a plataforma e a carga útil de um satélite;
- Sistema de Estação Terrena, presente em cada estação terrena de rastreamento, implementa o *link* de comunicação com um satélite durante os períodos em que este passa sobre uma estação. Durante estes períodos são transmitidos comandos ao satélite (telecomandos) e recebidos dados da plataforma de serviço e das cargas úteis (telemetrias);
- Subrede de Comunicação Solo, conecta todos os sistemas de solo permitindo a comunicação entre eles.

Por sua vez, o Sistema de Controle de Missão é constituído pelos sistemas:

- Sistema de Controle de Operações, realiza as seguintes atividades por meio dos três sistemas que o constitui: (1) gera informações, que englobam a órbita e a atitude dos satélites em órbita, expressas nos arquivos de Previsão de Visibilidade de Passagens – PVPs (Sistema de Dinâmica de Vôo); (2) planeja as operações de vôo para estes satélites considerando o uso compartilhado dos mesmos recursos de solo, quando for o caso (Sistema de Planejamento de Missão); e (3) executa os planos de operações de vôo gerados (Sistema de Execução de Operações);
- Sistema de Exploração de Missão, efetua procedimentos extras para tratar os produtos da missão, quando necessário, e distribui estes produtos aos usuários finais. Este sistema se comunica com os usuários finais de uma missão, que são os solicitantes dos serviços que um satélite oferece. Uma solicitação de usuário pode ser, por exemplo, a obtenção de imagens de coordenadas específicas.

A arquitetura MAPSat, proposta neste trabalho de pesquisa, focaliza a organização de agentes do Sistema de Planejamento de Missão, que interage

com outras organizações do ambiente de controle de satélites, além de interagir com papéis e utilizar recursos. Assim, a Figura 7.2 apresenta o diagrama de Relacionamentos de Interação da Visão Organizacional para a arquitetura MAPSat.

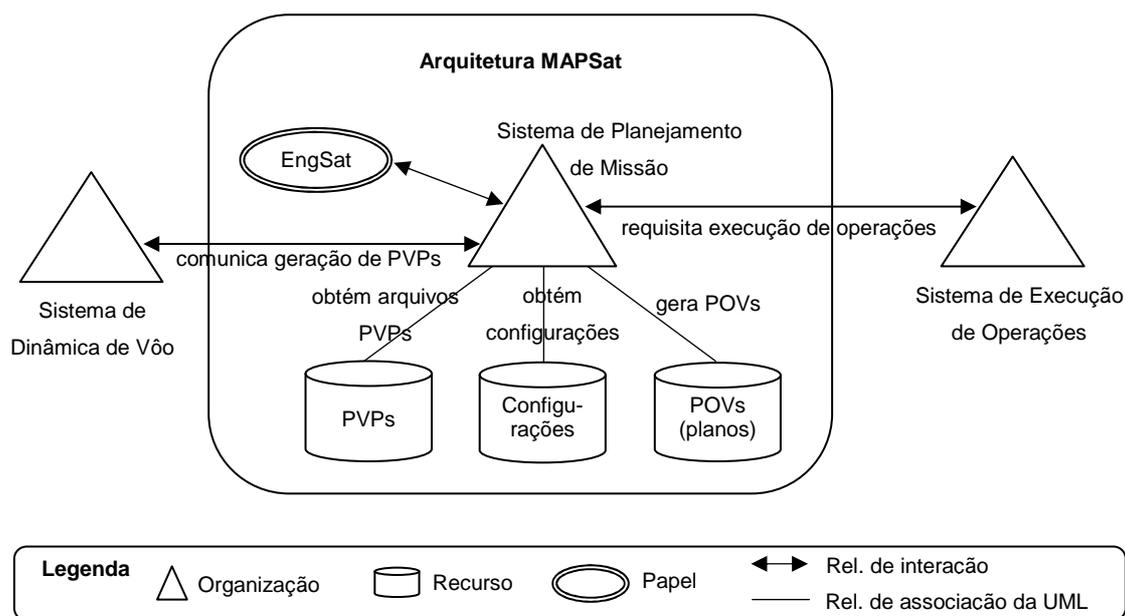


Figura 7.2 – Diagrama de relacionamentos de interação da visão organizacional

Como ilustra a Figura 7.2, a organização de Sistema de Planejamento de Missão interage com um papel (EngSat) pertencente à organização de Recursos Humanos da Figura 7.1, duas organizações do ambiente de controle de satélites (Sistema de Dinâmica de Vôo e Sistema de Execução de Operações), utiliza recursos das bases de dados (PVPs e Configurações) e alimenta a base de dados de planos (POVs).

O papel EngSat refere-se ao cargo de Engenheiro de Satélites, especialista no controle de um satélite específico ou de uma família de satélites, ou seja, satélites que apresentam estruturas internas e missões similares. O EngSat é responsável pelas operações de um determinado satélite em órbita. Por exemplo, este papel oferece os serviços de definição do conjunto de possíveis

operações para a fase de rotina de um satélite e as prioridades iniciais destas operações.

A organização de Sistema de Dinâmica de Vôo gera arquivos de Previsão de Visibilidade de Passagens, conhecidos pela sigla PVPs. Um arquivo PVP contém dados sobre as futuras passagens de um determinado satélite sobre uma estação terrena específica por um período de sete dias consecutivos. Por exemplo, o conteúdo deste arquivo provê para cada passagem do satélite considerado: valores dos ângulos de azimute e de elevação do satélite para o futuro apontamento da antena da estação terrena em direção do satélite; a data da passagem; os horários de aquisição de sinal (AOS – *Acquisition of Signal*) e perda de sinal (LOS – *Loss of Signal*) do satélite pela estação terrena de rastreo.

A organização de Sistema de Execução de Operações oferece o serviço de execução de operações. Estas operações encontram-se ordenadas por tempo de execução no Plano de Operações de Vôo (POV) e consistem em instâncias de ações de planejamento. A cada instante de tempo de execução especificado no POV, o sistema de Execução de Operações dispara a execução da operação correspondente por meio do serviço de Envio de Telecomandos. Este sistema também oferece o serviço de Recebimento de Telemetrias.

Em relação aos recursos utilizados pela organização de Sistema de Planejamento de Missão: a base de dados PVPs armazena os arquivos PVPs gerados pela organização de Sistema de Dinâmica de Vôo; a base de dados Configurações armazena dados gerais sobre os satélites e as estações solo de rastreo que fazem parte do ambiente de controle de satélites considerado; e, por fim, a base de dados POVs armazena os POVs gerados pela organização de Sistema de Planejamento de Missão.

Os dados armazenados na base de dados Configurações não se referem a uma única passagem de um satélite, mas sim ao conjunto de passagens de um determinado satélite sobre uma estação terrena específica. Estes dados podem

ser, por exemplo: a duração mínima de uma passagem para que esta seja rastreada; a duração da operação de calibração; a elevação mínima da antena solo para realizar medida de distância; a elevação mínima para realizar medida de velocidade; a lista completa de telecomandos que o satélite aceita; e a existência de um período de zona de silêncio, se aplicável.

Os termos citados acima, relacionados à área de controle de satélites, são definidos como (CARDOSO, 2006):

- duração mínima (DurMin): duração mínima de uma passagem para que esta seja rastreada por uma estação terrena, ou seja, para que se torne possível enviar telecomandos e/ou executar medidas de distância e de velocidade;
- operação de calibração: medida de calibração do equipamento de medida de distância da estação terrena de rastreo, que consiste na medida do atraso do sinal na estação. Esta medida é utilizada para corrigir as medidas realizadas. Esta operação pode ou não ser realizada dependendo de estarem ou não programadas execuções de sessões de medidas de distância na passagem. Quando realizada, ocorre antes do início do rastreo de uma passagem (operação de pré-passagem);
- medida de distância: medida do tempo de propagação de um sinal eletromagnético emitido por um equipamento instalado, por exemplo, em uma estação terrena de rastreo, no percurso total de ida até um *transponder* instalado, por exemplo, em um satélite, e retorno até o local de onde foi emitido, após ser retransmitido de volta pelo *transponder*. Como o sinal eletromagnético se propaga a velocidade da luz, uma vez obtida a medida do tempo de propagação do sinal no percurso entre a estação terrena e o satélite, em um dado instante, basta multiplicar o valor desta medida pela velocidade da luz para transformá-lo no valor de uma medida da distância da estação terrena ao satélite;

- medida de velocidade: o processo de geração baseia-se na medição do desvio *Doppler* de frequência sofrido por um sinal eletromagnético no percurso entre um transmissor e um receptor, que é proporcional à velocidade radial relativa entre o transmissor e o receptor. Diz-se que este tipo de medida é de um caminho quando o transmissor se encontra no satélite e o receptor em solo, onde é feita a medição. Quando tanto o transmissor quanto o receptor se encontram em solo, e o satélite é equipado com um *transponder* que apenas recebe o sinal enviado de solo e o retransmite de volta (o que implica em uma medida de velocidade baseada na medição do desvio *Doppler* total sofrido pelo sinal nos percursos de subida ao satélite e retorno ao solo), a medida é dita ser de dois caminhos;
- período de zona de silêncio: período da passagem durante o qual a comunicação do satélite com a estação terrena de rastreo é interrompida.

7.2.2 Visão de Objetivo

O objetivo principal da arquitetura MAPSat consiste em planejar operações de vôo para satélites, considerando o uso compartilhado de estações terrenas de rastreo. Este objetivo macro é decomposto em subobjetivos. O diagrama da Figura 7.3, um diagrama da Visão de Objetivo chamado de diagrama de Decomposição de Objetivos, ilustra a decomposição de objetivos para o objetivo principal da arquitetura de planejamento proposta.

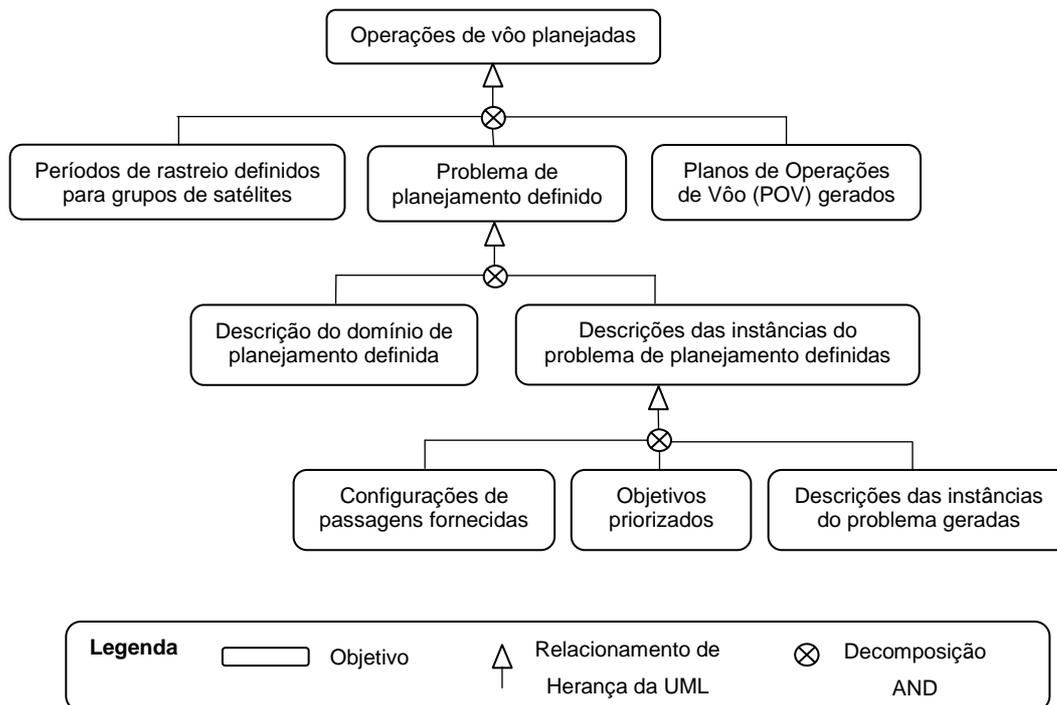


Figura 7.3 – Diagrama de decomposição de objetivos da visão de objetivo

Conforme ilustra o diagrama da Figura 7.3, o objetivo principal da arquitetura MAPSat “Operações de voo planejadas” é decomposto nos subobjetivos: “Períodos de rastreo definidos para grupos de satélites”, “Problema de planejamento definido” e “Planos de Operações de Voo (POV) gerados”, sendo satisfeito somente quando os três subobjetivos forem satisfeitos.

Por sua vez, estes subobjetivos são satisfeitos quando:

- Subobjeto “Períodos de rastreo definidos para grupos de satélites”: os períodos de rastreo dos satélites considerados forem definidos após a resolução de possíveis conflitos em relação às intersecções de passagens (horários conflitantes) de satélites distintos sobre uma mesma estação terrena de rastreo. Enfatiza-se o fato de que uma estação terrena, na qual está localizada a antena solo de rastreo, captura o sinal de um único satélite em um dado instante de tempo;

- Subobjetivo “Problema de planejamento definido”: o problema de planejamento for definido, o que inclui definir na linguagem de representação de problemas de planejamento PDDL os arquivos referentes à definição do problema de planejamento, que consistem nos arquivos de descrição de Domínio e de descrição das Instâncias do Problema;
- Subobjetivo “Planos de Operações de Vôo (POV) gerados”: os planos (POVs) forem gerados por um planejador, que utiliza os arquivos de descrição do problema de planejamento como entrada para o processo de geração de planos.

Em um nível inferior do diagrama, verifica-se que o objetivo “Problema de planejamento definido” é decomposto nos subobjetivos “Descrição do domínio de planejamento definida” e “Descrições das instâncias do problema de planejamento definidas”, satisfeitos quando:

- Subobjetivo “Descrição do domínio de planejamento definida”: o arquivo de descrição do Domínio do problema de planejamento for definido. Este arquivo contém os tipos, os predicados, as funções e as ações de planejamento para o problema de planejamento de controle de satélites;
- Subobjetivo “Descrições das instâncias do problema de planejamento definidas”: os arquivos de descrição das Instâncias do Problema forem definidos. Estes arquivos contêm os objetos, os estados iniciais, os eventos exógenos e os objetivos de cada instância do problema de planejamento de controle de satélites, ou seja, de cada rastreio a ocorrer no ambiente considerado.

No último nível do diagrama verifica-se que o objetivo “Descrições das instâncias do problema de planejamento definidas” é decomposto em três subobjetivos, satisfeitos quando:

- Subobjetivo “Configurações de passagens fornecidas”: informações que definem o contexto de cada passagem forem geradas para todas as passagens de um determinado satélite sobre uma estação terrena e para todos os satélites e estações terrenas de rastreamento pertencentes ao ambiente de controle de satélites considerado;
- Subobjetivo “Objetivos prioritizados”: forem estabelecidos níveis de prioridade para os objetivos a serem alcançados durante o rastreamento de uma passagem. Esta priorização de objetivos se faz necessária para passagens que sofreram redução no seu período de rastreamento devido a conflitos de horário com passagens de outros satélites;
- Subobjetivo “Descrições das instâncias do problema geradas”: os arquivos de descrição das Instâncias do Problema de planejamento forem gerados, utilizando os dados referentes às configurações de passagens e os níveis de prioridade para objetivos no momento de definição dos objetivos de rastreamento.

7.2.3 Visão de Agente

A Visão de Agente relaciona-se diretamente com as Visões Organizacional e de Objetivos. Esta visão considera as organizações, definidas na Visão Organizacional, e a decomposição de objetivos para estas organizações, definida na Visão de Objetivo.

A arquitetura MAPSat é constituída por uma única organização, a organização de Sistema de Planejamento de Missão, e por um conjunto de papéis e recursos. Assim, a organização de Sistema de Planejamento de Missão é responsável por atingir o objetivo principal da arquitetura MAPSat e seus subobjetivos, representados no diagrama da Figura 7.3.

A Visão de Agente da arquitetura MAPSat define como estes objetivos podem ser alcançados por meio da delegação de agentes responsáveis pelo alcance destes objetivos. A Figura 7.4 apresenta o diagrama de Estrutura de Delegação

da Visão de Agente para a arquitetura MAPSat, no qual os subobjetivos obtidos da decomposição do objetivo principal da arquitetura são atribuídos para agentes que constituem a organização de Sistema de Planejamento de Missão.

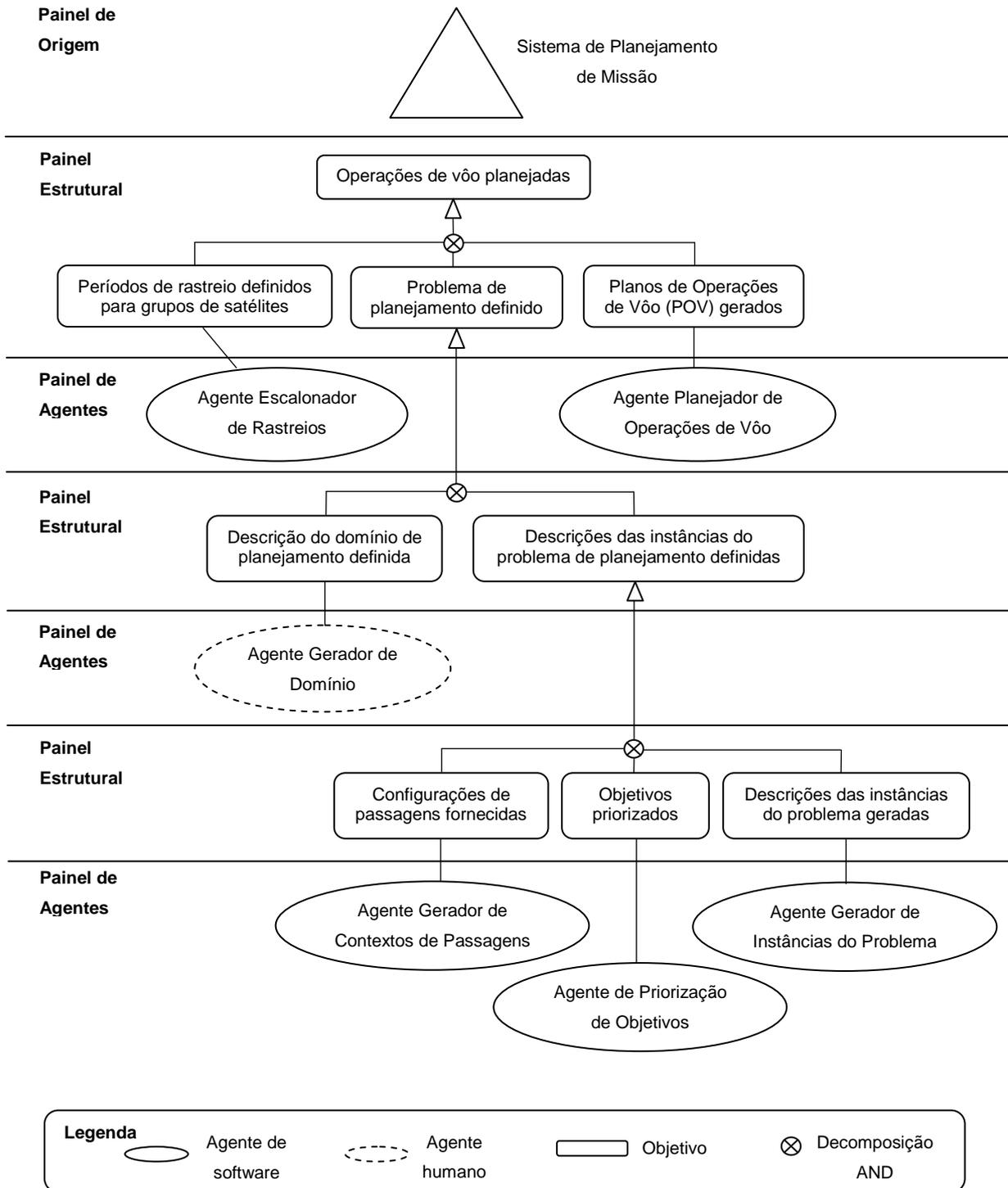


Figura 7.4 – Diagrama de estrutura de delegação da visão de agente

Como pode-se observar no diagrama da Figura 7.4, identificou-se os agentes que constituem a arquitetura MAPSat, especificamente a organização de Sistema de Planejamento de Missão, por meio da necessidade de atingir os objetivos previamente identificados. Assim, foram levantados agentes para a arquitetura MAPSat que realizam tarefas para o alcance de cada um dos objetivos, considerados essenciais para a geração de soluções para o problema de controle de múltiplos satélites.

Na Visão de Agente, a metodologia MESSAGE propõe a construção do diagrama de Estrutura de Delegação apresentado na Figura 7.4 e também a construção de um Esquema de Agente para cada agente identificado. A construção de um Esquema de Agente tem como finalidade detalhar o comportamento do agente, descrevendo as suas características principais. As próximas seções apresentam Esquemas para os agentes que constituem a arquitetura MAPSat.

7.2.4 Esquema do Agente Escalonador de Rastreios

O agente Escalonador de Rastreios define quais satélites serão rastreados por uma estação terrena, a ordem dos rastreios e as suas respectivas durações.

Como definido no Capítulo 2, que apresenta definições importantes sobre o ambiente de controle de satélites, um satélite pode somente transmitir telemetrias e receber telecomandos, ou seja, ser rastreado, durante o período em que existe visibilidade entre sua antena de bordo e a antena da estação terrena. Este período de visibilidade entre o satélite e a estação terrena recebe o nome de passagem.

No entanto, uma passagem pode não ser inteiramente rastreada por uma estação terrena. Assim, emprega-se o termo “período de rastreio” ao período de tempo de uma passagem em que há não somente visibilidade, mas também comunicação (envio de telemetrias e recebimento de telecomandos) entre o satélite e uma estação terrena.

O agente Escalonador de Rastreios tem como objetivo gerenciar o uso compartilhado de uma determinada estação terrena por um grupo de satélites, visto que a antena de uma estação terrena possui a capacidade de rastrear apenas um único satélite a cada instante de tempo. Assim, este agente busca: (1) identificar, para uma determinada estação terrena, a possível existência de conflitos de horário entre as passagens de dois ou mais satélites; e (2) resolver os conflitos identificados.

A técnica de resolução de conflitos utilizada consiste em reduzir a parte conflitante do período de rastreo do satélite de menor prioridade ou cancelar o rastreo deste satélite. O cancelamento ocorre quando, após a redução no período de rastreo do satélite de menor prioridade, o tempo resultante da passagem não satisfaz ao requisito de tempo $DurMin$, que define o valor da duração mínima de uma passagem para que esta seja rastreada por uma estação terrena.

A Figura 7.5 ilustra como o agente Escalonador de Rastreios soluciona o problema de passagens de satélites, sobre uma mesma estação terrena, com horários conflitantes. A Figura 7.5 (a) apresenta o caso típico de rastreo das passagens de dois satélites, satélite A e satélite B, por uma estação terrena e três variáveis importantes para o rastreo de múltiplos satélites, as variáveis $LOSCal$, $CalAOS$ e $LOS-AOS$.

A variável $LOSCal$ (*Loss Of Signal – Calibration*) representa o intervalo de tempo mínimo entre o final do período de rastreo do satélite A e o início da operação de calibração para o rastreo do satélite B.

A variável $CalAOS$ (*Calibration – Acquisition Of Signal*) representa o intervalo de tempo mínimo entre o início da operação de calibração para o rastreo do satélite B e o início do período de rastreo deste satélite (aquisição de sinal do satélite).

A variável LOS-AOS (*Loss Of Signal - Acquisition Of Signal*) representa o intervalo de tempo mínimo entre o final do período de rastreamento do satélite A e o início do período de rastreamento do satélite B.

Considerando que o satélite B apresenta prioridade menor de rastreamento em relação ao satélite A, as Figuras 7.5 (b) e 7.5 (c) ilustram os casos de cancelamento de uma parte do início do período de rastreamento de um satélite (de menor prioridade) e de uma parte ao final do período, respectivamente. A operação de calibração é somente realizada antes do início do período que contém as passagens simultâneas de dois satélites. Assim, no caso de passagens simultâneas sobre uma estação terrena, não é necessário executar a operação de calibração antes do período de rastreamento do segundo satélite, pois a operação de calibração executada antes do início do rastreamento do primeiro satélite ainda permanece válida para o rastreamento do segundo.

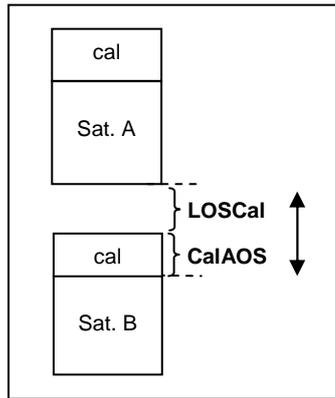


Figura 7.5 (a)

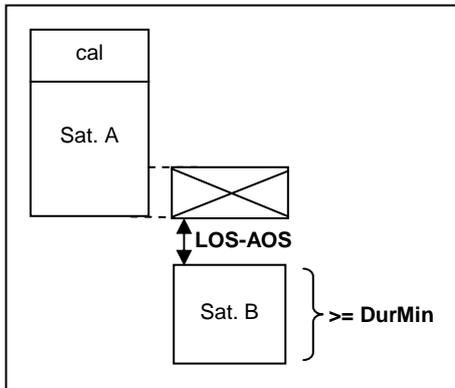


Figura 7.5 (b)

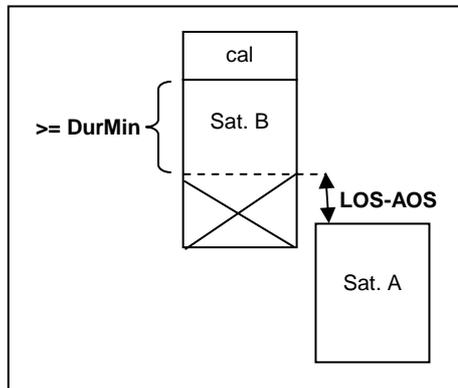


Figura 7.5 (c)

Figura 7.5 – Resolução de conflitos para o problema de intersecção de passagens

Uma vez definido o problema de conflitos de horário entre passagens, para o qual o agente Escalonador de Rastreios busca gerar soluções, apresenta-se a Tabela 7.1 que detalha o propósito e o objetivo deste agente, os eventos que ele sente, os recursos que utiliza e o seu comportamento, definido por meio do seu fluxo de tarefas.

Tabela 7.1 – Esquema do agente Escalonador de Rastreios

Esquema do Agente	Agente Escalonador de Rastreios.
Propósito	Escalonar os rastreios de um conjunto de satélites em relação a uma estação terrena.
Objetivo	Períodos de rastreio definidos para grupos de satélites.
Eventos	Mensagem da organização de Sistema de Dinâmica de Vôo comunicando o término da geração de arquivos PVPs para o conjunto de satélites e a estação terrena considerados.
Recursos	<ul style="list-style-type: none"> - Base de Dados Arquivos PVPs (<i>Previsão de Visibilidade de Passagens</i>). Um arquivo PVP é gerado para um período de 07 dias consecutivos para um determinado satélite e uma estação terrena específica. Este arquivo provê ao agente Escalonador de Rastreios a data, e os horários de AOS (<i>Acquisition Of Signal</i>) e EOS (<i>End Of Signal</i>) das passagens do satélite sobre a estação terrena durante o período de 01 semana. - Base de Dados Configurações. Esta base de dados provê ao agente Escalonador de Rastreios as prioridades dos satélites e a duração mínima de uma passagem para que esta seja rastreada (variável <i>DurMin</i>). - Base de Dados Escalonamentos de Rastreios. Esta base de dados armazena a Agenda de Rastreios gerada pelo agente Escalonador de Rastreios para o conjunto de satélites e a estação terrena considerados.
Fluxo de Tarefas	<ul style="list-style-type: none"> - Obtém da Base de Dados PVPs os arquivos PVPs, de períodos de tempo equivalentes (mesma semana), de dois satélites distintos em relação a uma mesma estação terrena. - Para cada arquivo PVP, calcula as durações de todas as passagens do respectivo satélite. As passagens de um satélite são identificadas por números de órbita únicos. - Compara as durações das passagens de cada satélite com o requisito de tempo <i>DurMin</i>, obtido da Base de Dados Configurações. Este requisito define o valor de duração mínima de uma passagem para que esta seja rastreada por uma estação terrena.

(continua)

Tabela 7.1 – Conclusão

<p>Fluxo de Tarefas (continuação)</p>	<ul style="list-style-type: none"> - Caso a duração da passagem de um satélite seja menor do que o valor do requisito de tempo DurMin, cancela o rastreo desta passagem. O cancelamento do rastreo da passagem deve-se ao fato da sua duração ser insuficiente para estabelecer e manter comunicação com o satélite. Caso contrário (período da passagem \geq DurMin), escalona a passagem do satélite para rastreo futuro. - Obtem as prioridades de rastreo dos satélites da Base de Dados Configurações. - Compara dois arquivos PVPs de satélites distintos em relação a uma mesma estação terrena e procede com esta comparação, obtendo arquivos PVPs aos pares, até que todos sejam comparados entre si a fim de verificar a possível existência de conflitos de horário entre duas passagens. - Caso ocorra um conflito de horário entre duas passagens de satélites distintos, cancela a parte conflitante da passagem do satélite de menor prioridade. - Para uma passagem que sofreu redução em seu período de rastreo, compara o novo período de rastreo da passagem com o requisito de tempo DurMin, obtido da Base de Dados Configurações. - Caso a duração do novo período de rastreo da passagem seja menor do que o valor DurMin, cancela o rastreo desta passagem. Caso contrário (período de rastreo \geq DurMin), agenda a passagem do satélite de rastreo reduzido para rastreo futuro. - Armazena na Base de Dados Escalonamentos de Rastreios a Agenda de Rastreios gerada para o conjunto de satélites e a estação terrena considerados. Uma Agenda de Rastreios inclui os seguintes dados: a ordem de rastreo do conjunto de satélites considerado para a estação terrena, as datas dos rastreios de cada passagem e dos horários de início (AOS – Acquisition Of Signal) e fim (LOS – Loss Of Signal) de rastreo, e os números de órbita das passagens cujos rastreios foram reduzidos pelo agente Escalonador de Rastreios.
---------------------------------------	---

7.2.5 Esquema do Agente Gerador de Contextos de Passagens

O agente Gerador de Contextos de Passagens (CARDOSO, 2006) provê dois serviços essenciais à futura geração de planos de operações de vôo (POVs): gerencia a Base de Dados Configurações, permitindo a inserção, a remoção e a edição de dados gerais relacionados ao ambiente de controle de satélites; e gera informações, chamadas de informações de configuração, que definem o contexto de cada passagem de satélite a ser rastreada. As informações de configuração variam de acordo com cada passagem a ser rastreada e são geradas a partir da análise dos dados gerais de rastreamento previamente armazenados na Base de Dados Configurações e dos dados listados nos arquivos PVPs.

Estas informações provêm a base para a futura geração automatizada do estado inicial de uma passagem e dos objetivos a serem atingidos durante o seu rastreamento por outro agente da arquitetura MAPSat. Portanto, as informações de configuração permitem a geração futura dos Arquivos de Instâncias do Problema de Planejamento da linguagem PDDL, por outro agente da arquitetura MAPSat, chamado de agente Gerador de Instâncias do Problema.

A Tabela 7.2 apresenta o esquema do agente Gerador de Contextos de Passagens.

Tabela 7.2 – Esquema do agente Gerador de Contextos de Passagens

Esquema do Agente	Agente Gerador de Contextos de Passagens.
Propósito	Gerar informações de configuração que definem o contexto de cada passagem de um satélite.
Objetivo	Configurações de passagens fornecidas.
Eventos	- Mensagens do agente Escalonador de Rastreios comunicando o término da geração da Agenda de Rastreios para cada conjunto de satélites e a estação terrena considerados.
Recursos	<p>- Base de Dados Arquivos PVPs (<i>Previsão de Visibilidade de Passagens</i>). Um arquivo PVP, gerado para um período de 07 dias consecutivos para um determinado satélite e uma estação terrena específica, provê ao agente Gerador de Contextos de Passagens previsões de dados de apontamento da antena da estação terrena para o rastreio do satélite considerado durante as suas passagens sobre a estação terrena.</p> <p>- Base de Dados Configurações. Esta base de dados provê ao agente Gerador de Contextos de Passagens informações gerais sobre o ambiente de controle de satélites.</p> <p>- Base de Dados Escalonamentos de Rastreios. Esta base de dados provê ao agente Gerador de Contextos de Passagens as passagens dos satélites que serão de fato rastreadas pelas estações terrenas consideradas (Agendas de Rastreios).</p>
Fluxo de Tarefas	<p>- Obtém da Base de Dados Escalonamentos de Rastreios as passagens (números de órbita) dos satélites que foram agendadas para rastreio pelas estações terrenas consideradas.</p> <p>- Gera, para cada passagem fornecida pelo agente Planejador de Rastreios, informações que estabelecem o contexto da passagem do satélite. Estas informações definem o contexto da passagem do satélite dentro do ambiente de controle de satélites uma vez que representam informações específicas referentes a uma determinada passagem.</p>

Por exemplo, uma informação gerada pelo agente Gerador de Contextos de Passagens para uma determinada passagem consiste na geração do horário de envio do último telecomando durante um rastreo. Esta informação não possui um valor fixo para todas as passagens de satélites, pois o horário de envio do último telecomando de uma determinada passagem depende do último horário em que se prevê, no arquivo PVP correspondente, que o satélite terá um ângulo de elevação mínima que permita o envio de um telecomando de forma segura ao final de uma passagem.

O agente Gerador de Contextos de Passagens gera a informação de horário de envio de último telecomando, para cada passagem de um satélite sobre uma estação terrena, de acordo com as seguintes regras de comportamento:

- Obtém da Base de Dados Configurações o valor de elevação mínima de fim de passagem para o envio de telecomandos da estação terrena ao satélite, chamado de restrição ElevMinTC;
- extrai do arquivo PVP correspondente (estação terrena e satélite considerados) o último horário da passagem que atende à restrição ElevMinTC;
- atribui este horário obtido a uma variável, chamada HorUltimoTC, que representa o horário de envio do último telecomando da respectiva passagem.

Esta variável pertence ao conjunto de variáveis que definem o Contexto da Passagem. Este Contexto da Passagem será posteriormente utilizado para a geração do Plano de Operações de Vôo (POV) da respectiva passagem.

7.2.6 Esquema do Agente Gerador de Domínio

O agente Gerador de Domínio gera um domínio para um problema de planejamento. Na arquitetura MAPSat, devido à adoção da linguagem de representação de problemas PDDL, este agente gera o arquivo de Domínio da

linguagem PDDL para o problema de planejamento de controle de satélites. A geração deste arquivo PDDL implica em definir os tipos, os predicados, as funções e as ações existentes no problema de planejamento.

Uma vez definidos os predicados do domínio do problema de planejamento, segundo Cardoso (2006), estes devem ser classificados em quatro categorias para possibilitar a futura geração automática dos arquivos de Instâncias do Problema em PDDL. Estas categorias são:

- categoria estados iniciais do ambiente. Esta categoria inclui os predicados que informam o estado inicial do ambiente;
- categoria estados intermediários. Esta categoria inclui os predicados que representam estados que se tornam verdadeiros durante o processo de planejamento, mas não são estados objetivos do problema de planejamento. Dessa forma, estes estados, chamados de estados intermediários, não são mapeados nos arquivos de Instâncias do Problema;
- categoria eventos exógenos. Esta categoria inclui os predicados que representam eventos que se tornam verdadeiros ou falsos em determinados instantes de tempo independentemente das ações executadas pelo planejador;
- categoria estados objetivos. Esta categoria inclui os predicados que constituem o estado objetivo final do processo de planejamento. Portanto, o estado objetivo final de um plano pode ser composto por subobjetivos, ou seja, o predicado do estado objetivo final passa a ser uma conjunção de predicados de estados objetivos. Os predicados de estados objetivos norteiam o processo de planejamento uma vez que o planejador insere ações de planejamento no plano, em processo de construção, quando estas levam à satisfação destes predicados.

A Tabela 7.3 apresenta o esquema do agente Gerador de Domínio.

Tabela 7.3 – Esquema do agente Gerador de Domínio

Esquema do Agente	Agente Gerador de Domínio.
Propósito	Definir o domínio do problema de planejamento de controle de satélites adotando uma ontologia para o domínio.
Objetivo	Descrição do domínio de planejamento definida.
Eventos	- Mensagem do papel EngSat, que refere-se ao cargo de Engenheiro de Satélites, autorizando o início da tarefa de geração de um arquivo de Domínio na linguagem PDDL para o problema de planejamento de controle de satélites.
Fluxo de Tarefas	<p>- Escolhe uma linguagem para representar o problema de planejamento que seja suficientemente expressiva para abranger as características do problema. Para o problema de planejamento de controle de satélites o agente Gerador de Domínio escolheu a linguagem PDDL (versão 2.2).</p> <p>- Define uma ontologia, ou seja, um vocabulário, para o domínio do problema de planejamento de controle de satélites. A definição da ontologia procede de acordo com os elementos existentes na estrutura da linguagem PDDL: tipos, predicados, funções e ações.</p> <p>- Classifica os predicados gerados em uma das quatro categorias de predicados propostas por Cardoso (2006): categoria estados iniciais do ambiente, categoria estados intermediários, categoria eventos exógenos e categoria estados objetivos. Esta classificação é realizada para possibilitar a futura geração automática dos arquivos de Instâncias do Problema em PDDL.</p> <p>- Codifica o arquivo de Domínio na linguagem PDDL.</p>

Para o problema de planejamento de controle de satélites, o agente Gerador de Domínio escolheu a linguagem PDDL, versão 2.2, devido ao fato desta versão da linguagem ser capaz de representar eventos exógenos incondicionais determinísticos. Este tipo de evento ocorre com frequência no problema de

planejamento de controle de satélites, como por exemplo: o início e o fim de uma passagem são eventos exógenos incondicionais determinísticos, pois são eventos externos que ocorrem independentemente da ocorrência de outros eventos ou ações escolhidas pelo planejador.

Após a definição da ontologia, o agente realiza a tarefa de codificação do arquivo de Domínio na linguagem PDDL. Nesta etapa, todos os termos definidos na ontologia do domínio são escritos de acordo com a estrutura da linguagem e os predicados definidos para o problema de planejamento de controle de satélites são classificados em uma das quatro categorias de predicados propostas por Cardoso (2006).

A seguir, apresentam-se exemplos de classificação de alguns dos predicados definidos para o problema de planejamento de controle de satélites nas quatro categorias (CARDOSO, 2006):

- categoria estados iniciais do ambiente. Exemplos de predicados desta categoria são: predicado (GSAvailable), significa que a estação terrena encontra-se disponível para o rastreamento da passagem; predicado (PassTrack), significa que a passagem deve ser rastreada; e predicado (RangeAvailable), significa que medidas de distância devem ser executadas durante a passagem;
- categoria estados intermediários. Um exemplo de predicado classificado na categoria estado intermediário consiste no predicado (CalibrationOK), que significa predicado de Calibração Executada, um estado atingido como efeito da execução da ação (Calibrate) durante o processo de planejamento. O alcance do estado intermediário (CalibrationOK) faz-se necessário para a posterior execução da ação (Range), que significa ação Executar Medida de Distância;
- categoria eventos exógenos. Um exemplo de predicado desta categoria consiste no predicado (SatSilentZoneAvailable), que informa a

ocorrência de zona de silêncio durante o rastreamento de uma passagem, ou seja, a ocorrência de uma janela de tempo durante a qual nenhuma ação de comunicação entre o satélite e a estação terrena pode ser executada;

- categoria estados objetivos. Alguns exemplos de predicados classificados na categoria estados objetivos são: predicado (RangeOK), que significa Medida de Distância Executada; predicado (PassEliminated), que significa Passagem Eliminada devido ao período da passagem não atender ao requisito de tempo mínimo DurMin para rastreamento de uma passagem; e predicado (TCCondicionalOK), que significa Telecomando Condicional Enviado. Um telecomando do tipo condicionado depende da execução prévia de outro telecomando, chamado de telecomando condição, para que ele possa ser executado. Quando o estado objetivo (TCCondicionalOK) for verdadeiro significa que ambos os telecomandos foram executados.

Uma vez definidos os predicados, o agente Gerador de Domínio faz uso dos predicados definidos, além de tipos e funções, para codificar as ações de planejamento. Para cada ação, o agente define: os tipos utilizados do domínio (parameters); os predicados de pré-condição para a inserção da ação no plano (condition); os predicados de efeito, que informam o que a ação gera ao ser executada (effect); e se a ação tiver uma duração, o tempo necessário para a execução da mesma (duration), representado como um valor fixo ou como um valor variável obtido pelo valor de retorno de uma função. Os valores de retorno de funções devem ser definidos nos arquivos de Instâncias do Problema.

A seguir são apresentados dois exemplos de ações codificadas em PDDL 2.2 para o problema de planejamento de controle de satélites. O arquivo de Domínio completo, escrito na linguagem PDDL 2.2, para este problema de planejamento encontra-se no Apêndice A.

01. Ação WaitSilentZone (ou seja, esperar por Zona de Silêncio)

```

(:durative-action WaitSilentZone
:parameters (?pass - PassNumber ?sat - SatCode ?gs - GSCode ?ZS - ZSNumber)
:duration (= ?duration (ZoneSilentTime ?pass ?ZS))
:condition (and (at start (GSAvailable ?gs))
                (at start (SatSilentConeAvailable ?pass ?sat ?gs ?ZS))
              )
:effect (and (at end (SilentZone ?pass ?sat ?gs ?ZS))
            (at start (not (GSAvailable ?gs)))
            (at end (GSAvailable ?gs))
          )
)
)

```

02. Ação SendFirstTC (ou seja, enviar primeiro telecomando da passagem)

```

(:durative-action SendFirstTC
:parameters (?pass - PassNumber ?sat - SatCode ?tc - TCCode ?gs - GSCode)
:duration (= ?duration (TCTime ?tc))
:condition (and (at start (GSAvailable ?gs))
                (at start (FirstTc ?pass ?sat ?tc))
                (over all (PassTrack ?pass ?sat ?gs))
                (over all (not (TCNotAvailable ?pass ?sat)))
              )
:effect (and (at end (FirstTcOK ?pass ?sat ?tc))
            (at end (TcSent ?pass ?sat))
            (at start (not (GSAvailable ?gs)))
            (at end (GSAvailable ?gs))
          )
)
)

```

Nos exemplos 01 e 02 apresentados pode-se observar que para cada ação foram definidas as pré-condições (condition) para que a ação seja inserida no plano durante o processo de planejamento, o tempo de execução (duration) da ação e os efeitos (effect) que ela produz ao ser executada.

7.2.7 Esquema do Agente de Priorização de Objetivos

O agente de Priorização de Objetivos age quando situações excepcionais ocorrem no ambiente de controle de satélites. Este agente prioriza os

predicados do arquivo de Domínio em PDDL classificados na categoria “estados objetivos” para gerenciar três situações de exceção: (1) a redução do período de rastreio de uma passagem, realizada pelo agente Escalonador de Rastreios para solucionar o conflito de intersecção de passagens (passagens de dois satélites distintos sobre uma mesma estação terrena com horários conflitantes); (2) a primeira passagem de um Grupo de Passagens Consecutivas; e (3) a última passagem de um Grupo de Passagens Consecutivas.

Um Grupo de Passagens Consecutivas consiste em um conjunto de passagens seqüenciais de um satélite sobre uma estação terrena que são todas visíveis à estação. No ambiente de controle de satélites, existem satélites que durante um período de 24 horas se tornam visíveis a uma estação terrena durante Passagens Consecutivas (órbitas consecutivas) e depois permanecem fora de visibilidade da estação nas demais passagens do satélite durante o restante do dia.

A Tabela 7.4, a seguir, apresenta as descrições das situações de exceção do ambiente de controle de satélites para as quais o agente de Priorização de Objetivos provê o serviço de estabelecimento de prioridades aos objetivos.

Tabela 7.4 – Descrições das situações do ambiente para as quais o agente atua

Situação	Descrição
Redução do Período de Rastreo de uma Passagem	No caso da redução do período de rastreo de uma passagem, o novo período de rastreo resultante pode não ser suficiente para que todos os objetivos sejam alcançados. O agente de Priorização de Objetivos, por meio da atribuição de prioridades aos predicados objetivos do arquivo de Domínio em PDDL, permite que somente os predicados objetivos de maior prioridade sejam considerados durante o processo de planejamento. O propósito deste agente consiste em privilegiar objetivos, frente à restrição de tempo de um período de rastreo reduzido, a fim de garantir que o Plano de Operações de Vôo a ser gerado inclua operações para o alcance dos objetivos prioritários.
Primeira e Última Passagens de um Grupo de Passagens Consecutivas	No caso da passagem, para a qual deseja-se gerar um Plano de Operações de Vôo (POV), ser a primeira ou a última passagem de um Grupo de Passagens Consecutivas, o agente de Priorização de Objetivos aumenta ou diminui os níveis de prioridade de alguns predicados objetivos do arquivo de Domínio a fim de garantir que o POV a ser gerado inclua operações para o alcance destes objetivos ou não considere os objetivos, respectivamente.

Para implementar esta solução de priorização de objetivos sugere-se vincular a cada predicado objetivo do arquivo de Domínio PDDL um valor numérico que simboliza o nível de prioridade de execução do objetivo, que pode ser “baixo”, “normal” ou “alto”. Os valores propostos para representar os níveis de prioridade “baixo”, “normal” e “alto” são: 0 – objetivo de prioridade baixa; 1 – objetivo de prioridade normal; e 2 – objetivo de alta prioridade.

Estes níveis de prioridade são posteriormente utilizados por outro agente da arquitetura MAPSat, chamado de agente Gerador de Instâncias do Problema, responsável por gerar os objetivos de rastreo específicos a cada passagem de um satélite. Este agente beneficia-se do mecanismo de priorização de objetivos à medida que, no momento de sua atuação, analisa somente os predicados objetivos de prioridades normal e alta do arquivo de Domínio PDDL durante o

processo de geração dos objetivos de rastreo de uma passagem, desconsiderando os predicados objetivos de prioridade baixa.

A Tabela 7.5 apresenta o esquema do agente de Priorização de Objetivos.

Tabela 7.5 – Esquema do agente de Priorização de Objetivos

Esquema do Agente	Agente de Priorização de Objetivos.
Propósito	Priorizar objetivos a fim de garantir que os Planos de Operações de Vôo (POVs), a serem gerados, incluam operações para o alcance dos objetivos prioritários.
Objetivo	Objetivos priorizados.
Eventos	- Mensagens do agente Escalonador de Rastreios comunicando o término da geração da Agenda de Rastreios para cada conjunto de satélites e a estação terrena considerados.
Recursos	- Base de Dados Configurações. Esta base de dados provê ao agente Gerador de Priorização de Objetivos as prioridades iniciais dos objetivos. - Base de Dados Escalonamentos de Rastreios. Esta base de dados provê ao agente de Priorização de Objetivos: as passagens (números de órbita) que sofreram redução nos seus períodos de rastreios para resolver conflitos de horário entre passagens (Agendas de Rastreios); e as indicações de primeira e última passagem de um Grupo de Passagens Consecutivas.
Fluxo de Tarefas	- Obtém da Base de Dados Escalonamentos de Rastreios: as passagens (números de órbita) que sofreram redução nos seus períodos de rastreios; e as passagens indicadas como primeiras ou últimas passagens de Grupos de Passagens Consecutivas.

(continua)

Tabela 7.5 – Conclusão

<p>Fluxo de Tarefas (continuação)</p>	<ul style="list-style-type: none"> - Para cada uma destas passagens obtidas, realiza as seguintes tarefas: <ul style="list-style-type: none"> - aumenta os níveis de prioridade de predicados objetivos do arquivo de Domínio PDDL (previamente gerados pelo agente Gerador de Domínio) para o nível 2, que significa prioridade alta. O aumento do nível de prioridade de um predicado objetivo faz com que este seja analisado durante o processo de geração do arquivo PDDL de Instância do Problema, ou seja, durante o processo de geração dos objetivos de rastreo de uma passagem, realizado pelo agente Gerador de Instâncias do Problema; - diminui os níveis de prioridade de predicados objetivos do arquivo de Domínio PDDL para o nível 0, que significa prioridade baixa. A diminuição do nível de prioridade de um predicado objetivo faz com que este não seja analisado durante o processo de geração do arquivo PDDL de Instância do Problema pelo agente Gerador de Instâncias do Problema.
---------------------------------------	--

A seguir, apresentam-se exemplos do comportamento do agente de Priorização de Objetivos para tratar as três situações excepcionais do ambiente de controle de satélites que este agente tem como propósito lidar.

Exemplo 01 de Comportamento – para a situação de Redução do Período de Rastreo de uma Passagem, o agente de Priorização de Objetivos:

- Diminui a prioridade do predicado objetivo “Executar Operação de Calibração” para 0 (nível baixo) para que este predicado objetivo não seja analisado durante o posterior processo de geração dos objetivos de rastreo da passagem, realizado pelo agente Gerador de Instâncias do Problema. A operação de calibração consiste em efetuar uma medida de calibração, ou seja, uma medida do atraso sofrido pelo sinal de medição da distância no equipamento da estação, atraso este que deve ser subtraído do valor do tempo de propagação do sinal da estação ao satélite medido, já que se constitui em um erro de medida. Esta

operação é realizada antes do início do rastreamento da passagem, não conflitando, portanto, com as operações de tempo real;

- Aumenta a prioridade do predicado objetivo “Reservar Janela de Tempo LOS-AOS” para 2 (nível alto) para que este predicado objetivo seja considerado durante o posterior processo de geração dos objetivos de rastreamento da passagem. Este predicado objetivo garante a reserva de um intervalo de tempo mínimo, chamado de tempo LOS-AOS, entre o final do período de rastreamento do satélite anterior (LOS - Loss Of Signal), agendado para rastreamento pela estação terrena, e o início do período de rastreamento da passagem sob análise (AOS - Acquisition Of Signal), para a qual deseja-se gerar o Plano de Operações de Voo.

Exemplo 02 de Comportamento – para a situação de Última Passagem de um Grupo de Passagens Consecutivas, o agente de Priorização de Objetivos:

- Aumenta a prioridade do predicado objetivo “Reduzir Taxa de Aquisição de Telemetrias do Computador de Bordo” para o nível 2 (nível alto) para que este predicado objetivo seja considerado durante o posterior processo de geração dos objetivos de rastreamento da passagem, realizado pelo agente Gerador de Instâncias do Problema. Este predicado objetivo garante a redução da taxa de aquisição de telemetrias do computador de bordo do satélite e a consequente redução do volume de telemetrias obtidas e armazenadas pelo computador de bordo nas próximas órbitas do satélite não-visíveis a estações terrenas.

Um satélite possui um computador de bordo que, enquanto fora de visibilidade de antenas terrenas, armazena telemetrias (dados sobre o estado do satélite) segundo uma taxa ou frequência de aquisição. No entanto, este computador de bordo possui uma capacidade de armazenamento limitada, o que faz com que seja necessário reduzir o valor da taxa de aquisição de telemetrias, para as órbitas consecutivas do satélite não-visíveis a estações terrenas, a fim de gerar

um intervalo maior de tempo entre duas aquisições de telemetrias subseqüentes e, com isso, obter e armazenar um volume menor de telemetrias.

O problema fundamental consiste no fato de que para órbitas consecutivas não-visíveis a estações terrenas, o computador de bordo do satélite armazenará, de forma acumulativa, as telemetrias obtidas durante estas órbitas até que uma órbita seja visível e rastreada por uma estação terrena, momento no qual, as telemetrias poderão ser enviadas a Terra e a memória do computador de bordo será conseqüentemente descarregada.

Assim, a redução da taxa de aquisição de telemetrias tem como finalidade assegurar que o volume de telemetrias armazenadas não exceda a capacidade de armazenamento limite do computador de bordo durante as órbitas consecutivas do satélite não-visíveis a estações terrenas.

Em contrapartida, o aumento da taxa de aquisição de telemetrias é efetuada para gerar o efeito contrário, ou seja, para gerar um intervalo menor de tempo entre duas aquisições de telemetrias subseqüentes e, com isso, obter e armazenar um volume maior de telemetrias. Esta prática é adotada para passagens consecutivas de um satélite (todas visíveis a estações terrenas), uma vez que, neste caso, pode-se armazenar um maior volume de telemetrias no computador de bordo, pois estas telemetrias poderão ser enviadas a Terra em passagens subseqüentes, o que significa que a memória do computador de bordo será constantemente descarregada.

Exemplo 03 de Comportamento – para a situação de Primeira Passagem de um Grupo de Passagens Consecutivas, o agente de Priorização de Objetivos:

- Aumenta a prioridade do predicado objetivo “Aumentar Taxa de Aquisição de Telemetrias do Computador de Bordo” para o nível 2 (nível alto) para que este predicado objetivo seja considerado durante o posterior processo de geração dos objetivos de rastreo da passagem, realizado pelo agente Gerador de Instâncias do Problema. Este predicado objetivo

garante o aumento da taxa de aquisição de telemetrias do computador de bordo do satélite a fim de que um volume maior de telemetrias possa ser obtido e armazenado no computador de bordo, visto que a memória do computador de bordo já poderá ser descarregada na próxima passagem.

7.2.8 Esquema do Agente Gerador de Instâncias do Problema

O agente Gerador de Instâncias do Problema gera, para cada passagem de satélite, o arquivo de Instância do Problema na linguagem PDDL (CARDOSO, 2006). Um arquivo deste tipo contém os objetos, os estados iniciais, os eventos exógenos incondicionais determinísticos e os objetivos referentes ao período de rastreamento de uma determinada passagem.

Este agente possui como percepções de entrada as Informações do Contexto da Passagem em análise, geradas pelo agente Gerador de Contextos de Passagens, e o arquivo de Domínio em PDDL, gerado pelo agente Gerador de Domínio para todo o ambiente de controle de satélites.

Para a geração de forma automatizada dos objetivos de rastreamento de uma determinada passagem, este agente analisa os predicados objetivos definidos no arquivo de Domínio em PDDL. Na realidade, somente os predicados objetivos de níveis de prioridade alta e normal são analisados, por ordem de relevância. Esta análise consiste em verificar se cada predicado objetivo satisfaz às restrições impostas pelas Informações do Contexto da Passagem, geradas pelo agente Gerador de Contextos de Passagens, referentes à passagem considerada. Quando as restrições são satisfeitas, o predicado objetivo é inserido no arquivo de Instância do Problema. Esta inserção significa que o predicado objetivo passa a ser considerado um objetivo a ser atingido durante o rastreamento específico da passagem analisada.

A Tabela 7.6 apresenta o esquema do agente Gerador de Instâncias do Problema.

Tabela 7.6 – Esquema do agente Gerador de Instâncias do Problema

Esquema do Agente	Agente Gerador de Instâncias do Problema.
Propósito	Gerar um arquivo de Instância do Problema para cada passagem do ambiente de controle de satélites.
Objetivo	Descrições das instâncias do problema de planejamento definidas.
Eventos	<ul style="list-style-type: none"> - Mensagem do agente Gerador de Contextos de Passagens comunicando o término da geração das Informações dos Contextos das Passagens a serem rastreadas. - Mensagem do agente de Priorização de Objetivos comunicando o término do processo de priorização de predicados objetivos do arquivo de Domínio PDDL para as passagens de rastreios reduzidos, ou para as passagens que são as primeiras ou as últimas de um Grupo de Passagens Consecutivas.
Recursos	<ul style="list-style-type: none"> - Informações dos Contextos das Passagens. Estas informações, geradas pelo agente Gerador de Contextos de Passagens, provêm ao agente Gerador de Instâncias do Problema restrições específicas a respeito de cada passagem dos satélites considerados. - Arquivo de Domínio PDDL. Este arquivo, gerado pelo agente Gerador de Domínio, provê ao agente Gerador de Instâncias do Problema as descrições dos seguintes elementos que compõem o domínio do problema de planejamento de controle de satélites: os tipos, as funções, os predicados e as ações de planejamento. Em particular, os predicados são classificados em quatro categorias, propostas por Cardoso (2006), para permitir a geração de forma automatizada dos arquivos de Instâncias do Problema.
Fluxo de Tarefas	<ul style="list-style-type: none"> - Para cada categoria de predicados declarada no arquivo de Domínio em PDDL, obtém o conjunto de predicados pertencente à categoria.

(continua)

Tabela 7.6 – Conclusão

<p>Fluxo de Tarefas (continuação)</p>	<ul style="list-style-type: none"> - Considerando o conjunto de predicados obtido, verifica as Informações do Contexto da Passagem referentes a cada predicado. Estas informações são geradas pelo agente Gerador de Contextos de Passagens para a passagem para a qual se deseja gerar um arquivo de Instância do Problema (posteriormente, será gerado o Plano de Operações de Voo para esta passagem). - Insere um predicado no arquivo de Instância do Problema, para a passagem considerada, somente caso as restrições impostas pelas Informações do Contexto da Passagem referentes àquele predicado sejam satisfeitas. - Para os predicados pertencentes à categoria Estados Objetivos, verifica não somente as Informações do Contexto da Passagem referentes a cada predicado objetivo, mas também os valores que representam os níveis de prioridades destes predicados objetivos. Estes valores de níveis de prioridades foram previamente atualizados pelo agente de Priorização de Objetivos para as situações de exceção do ambiente de controle de satélites. - Insere um predicado objetivo no arquivo de Instância do Problema, para a passagem considerada, somente caso as restrições impostas pelas Informações do Contexto da Passagem referentes àquele predicado objetivo sejam satisfeitas e o seu nível de prioridade seja normal ou alto (níveis 1 ou 2, respectivamente).
---------------------------------------	---

A seguir, apresenta-se um exemplo do comportamento do agente Gerador de Instâncias do Problema para realizar a tarefa de geração de um objetivo de rastreamento para uma determinada passagem:

- O agente Gerador de Instâncias do Problema primeiramente obtém do arquivo de Domínio PDDL o conjunto de predicados que pertence à categoria de predicados objetivos;

- Por exemplo, para o predicado objetivo “Executar Medida de Velocidade”, pertencente ao conjunto de predicados objetivos obtido, o agente verifica

as Informações do Contexto da Passagem referentes a este predicado objetivo. Estas informações foram previamente geradas pelo agente Gerador de Contextos de Passagens. Portanto, o agente verifica o valor de uma variável de informação do contexto, chamada `MedidaVelocidadeDisponível`, que armazena os valores: “verdadeiro”, quando prevê-se no arquivo PVP (arquivo de Previsão de Visibilidade de Passagens) que o satélite apresentará durante a passagem a elevação mínima requerida para o envio/recebimento do sinal; ou “falso”, quando prevê-se no arquivo PVP que o satélite não apresentará a elevação mínima requerida;

- Caso o valor da variável `MedidaVelocidadeDisponível` seja “verdadeiro”, insere o predicado objetivo “Executar Medida de Velocidade” no arquivo de Instância do Problema como um objetivo de rastreamento da passagem. Caso contrário, não realiza a inserção do predicado objetivo.

7.2.9 Esquema do Agente Planejador de Operações de Vôo

O agente Planejador de Operações de Vôo gera um Plano de Operações de Vôo (POV) para cada passagem de satélite a ser rastreada por uma estação terrena. Um POV contém as operações necessárias para o controle em órbita de um satélite. Considerando a geração de POVs como um problema de planejamento da área de IA, as operações contidas em um POV são instâncias de ações de planejamento, cujos esquemas de ações encontram-se descritos no arquivo de Domínio PDDL definido para o problema de planejamento de controle de satélites.

Assim, o agente Planejador de Operações de Vôo pode ser classificado como um agente de planejamento, pois é um agente de software capaz de construir planos de ações para um determinado problema.

O processo de definição deste agente teve como foco o problema para o qual se deseja gerar planos de solução, ou seja, o problema de planejamento de

controle de satélites. Por meio da análise deste problema de planejamento, buscou-se identificar:

- as propriedades do ambiente de planejamento. O ambiente de planejamento foi classificado como um ambiente de planejamento clássico com restrições de tempo e recurso;
- uma técnica de planejamento adequada para o tipo de ambiente de planejamento identificado. Uma técnica de planejamento considerada adequada ao ambiente de planejamento clássico com restrições de tempo e recurso é a técnica de planejamento Temporal Baseada em Grafos.

Em especial, a técnica de planejamento Temporal Baseada em Grafos, originada do algoritmo GRAPHPLAN (BLUM e FURST, 1995), unida à linguagem de representação de problemas de planejamento PDDL, escolhida pelo agente Gerador de Domínio, formaram a base para a criação do agente Planejador de Operações de Vôo.

Por sua vez, a escolha do planejador (algoritmo de planejamento), para atuar como mecanismo de raciocínio do agente Planejador, foi guiada por dois fatores: (1) suporte à técnica de planejamento Temporal Baseada em Grafos; e (2) suporte às representações de conhecimento da linguagem PDDL. O planejador LPG-TD (GEREVINI et al., 2005) foi selecionado por oferecer suporte à técnica de planejamento Temporal Baseada em Grafos e também ser capaz de interpretar as características da linguagem PDDL 2.2.

No planejador LPG-TD, as representações de grafos foram estendidas para lidar com as novas características que a versão 2.2 da linguagem PDDL oferece (predicados derivados e literais iniciais temporizados), bem como para melhorar o gerenciamento de ações com duração. Este planejador encontra-se descrito em detalhes no Capítulo 4.

Assim, o planejador LPG-TD é o mecanismo de raciocínio do agente Planejador de Operações de Vôo e as percepções de entrada deste agente são todos os arquivos da linguagem PDDL necessários para a descrição do problema de planejamento de controle de satélites – o arquivo de Domínio e os arquivos de Instâncias do Problema.

A Tabela 7.7 apresenta o esquema do agente Planejador de Operações de Vôo.

Tabela 7.7 – Esquema do agente Planejador de Operações de Vôo

Esquema do Agente	Agente Planejador de Operações de Vôo.
Propósito	Gerar um Plano de Operações de Vôo (POV) para cada passagem do ambiente de controle de satélites.
Objetivo	Planos de Operações de Vôo (POVs) gerados.
Eventos	<ul style="list-style-type: none"> - Mensagem do agente Gerador de Domínio comunicando o término da geração do arquivo de Domínio PDDL para o problema de planejamento de controle de satélites. - Mensagem do agente Gerador de Instâncias do Problema comunicando o término da geração dos arquivos de Instâncias do Problema para as passagens consideradas.
Recursos	- Arquivo de Domínio PDDL. Este arquivo, gerado pelo agente Gerador de Domínio, provê ao agente Planejador de Operações de Vôo as descrições dos seguintes elementos que compõem o domínio do problema de planejamento de controle de satélites: os tipos, as funções, os predicados e as ações de planejamento.

(continua)

Tabela 7.7 – Conclusão

Recursos (continuação)	<p>- Arquivos de Instâncias do Problema. Um arquivo de Instância do Problema, gerado pelo agente Gerador de Instâncias do Problema para cada passagem de satélite, provê ao agente Planejador de Operações de Vôo: os objetos, os estados iniciais, os eventos exógenos incondicionais determinísticos e os objetivos referentes ao período de rastreamento de uma determinada passagem.</p>
Fluxo de Tarefas	<p>- Obtém do agente Gerador de Domínio o nome do arquivo de Domínio em PDDL gerado por este agente para o problema de planejamento de controle de satélites.</p> <p>- Obtém do agente Gerador de Instâncias do Problema os nomes dos arquivos de Instâncias do Problema em PDDL gerados por este agente para as passagens para as quais se deseja gerar Planos de Operações de Vôo.</p> <p>- Invoca o planejador LPG-TD, o algoritmo de planejamento, para a geração de Planos de Operações de Vôo (POVs) para um conjunto de passagens. Na realidade, repete-se a invocação do planejador LPG-TD para a geração individual de um POV para cada passagem, fornecendo como entrada o arquivo de Domínio e o respectivo arquivo de Instância do Problema referente à passagem considerada.</p>

7.3 Resultados do Protótipo da Arquitetura MAPSat

Nesta seção apresentam-se os resultados gerados por um protótipo implementado para demonstrar o funcionamento da arquitetura MAPSat.

Este protótipo utiliza as seguintes ferramentas e linguagens: gerenciador de base de dados Microsoft Access 2003; Microsoft Visual Studio 6.0; linguagem C++; linguagem de representação de problemas de planejamento PDDL 2.2; e planejador LPG-TD-1.0.exe, codificado por seus autores na linguagem C.

Nas subseções a seguir apresentam-se os arquivos gerados pelo protótipo para a geração de um Plano de Operações de Vôo (POV) para uma passagem

que teve o seu período de rastreo reduzido devido a uma intersecção desta passagem com a passagem de outro satélite de maior prioridade de rastreo.

7.3.1 Escalonamento de Rastreios

Durante o processo de escalonamento de rastreios de passagens de múltiplos satélites em relação a uma mesma estação terrena, realizado pelo agente Escalonador de Rastreios, podem existir conflitos de horário entre duas ou mais passagens de diferentes satélites para o uso exclusivo de uma determinada estação terrena.

A Figura 7.6 apresenta uma ilustração para o caso de redução do período de rastreo de uma passagem do satélite SCD1, pertencente à família de satélites Brasileiros de Coleta de Dados, devido a um conflito de horário entre a passagem deste satélite (identificada pelo número de órbita 60482-5) e uma passagem do satélite SCD2, pertencente à mesma família de satélites. Ressalta-se que estas passagens dos satélites SCD1 e SCD2 competem pelo uso exclusivo da estação terrena de Cuiabá.

Como o satélite SCD1 possui prioridade menor de rastreo em relação ao satélite SCD2, o trecho de sua passagem em conflito de horário com a passagem do satélite SCD2 é cancelado, o que implica na redução do período de rastreo do satélite SCD1. Em contrapartida, mantém-se intacta a passagem do satélite SCD2, ou seja, a passagem do satélite SCD2 é inteiramente rastreada pela estação terrena de Cuiabá.

A Figura 7.6 apresenta parte do arquivo PVP correspondente às previsões de passagens do satélite SCD1 para a estação terrena de Cuiabá e ilustra o cancelamento do trecho da passagem do satélite SCD1 em conflito de horário com a passagem do satélite SCD2.

A passagem original do satélite SCD1 inicia-se às 04:23:00 (h:min:seg), mas após o cancelamento de um trecho desta passagem passa a ter o novo horário de início de 04:29:00. Assim, esta passagem sofreu uma redução no seu

período de rastreamento, originalmente de 13 min. e 30 seg. (810 seg.) para 7 min. e 30 seg. (450 seg.), totalizando uma redução de 6 min. no período de rastreamento global da passagem.

SCP - Satellite Contact Prediction Pg. 4

Satellite Identification - SCD1
 Ground Station Identification - CBA
 Orbit Number - 60482.5 related to 29/07/04 04:23:00.000
 E - Eclipse (1 = Satellite in Earth Shadow)
 ZP - Zenith pass (1 = Zenith passage)
 SP - Sun pointing (1 = Antenna pointing to Sun)
 SZ - Silent Zone (0 = Contact; <>0 = Contact angle)

Date	Time (UTC)	Azimuth	Elevat	Range	Doppler	D-Rate	E	ZP	SP	SZ
DD/MM/YY	HH:MM:SS.SSS	(deg)	(deg)	(km)	(kHz)	(kHz/s)				
29/07/04	04:23:00.000	277.900	4.146	2815.102	44.870	-0.0071	1	0	0	176
29/07/04	04:23:30.000	277.049	6.143	2632.867	44.614	-0.0100	1	0	0	179
29/07/04	04:24:00.000	276.069	8.308	2451.868	44.258	-0.0138	1	0	0	179
29/07/04	04:24:30.000	274.925	10.677	2272.572	43.773	-0.0187	1	0	0	176
29/07/04	04:25:00.000	273.570	13.296	2095.583	43.117	-0.0253	1	0	0	173
29/07/04	04:25:30.000	271.936	16.226	1921.708	42.232	-0.0341	1	0	0	170
29/07/04	04:26:00.000	269.926	19.543	1752.039	41.034	-0.0464	1	0	0	166
29/07/04	04:26:30.000	267.390	23.341	1588.090	39.398	-0.0636	1	0	0	161
29/07/04	04:27:00.000	264.098	27.731	1431.985	37.145	-0.0880	1	0	0	156
29/07/04	04:27:30.000	259.669	32.820	1286.735	34.017	-0.1225	1	0	0	149
29/07/04	04:28:00.000	253.463	38.654	1156.578	29.669	-0.1697	1	0	0	142
29/07/04	04:28:30.000	244.388	45.062	1047.281	23.709	-0.2295	1	0	0	132
29/07/04	04:29:00.000	230.761	51.322	966.054	15.863	-0.2930	1	0	0	121
29/07/04	04:29:30.000	211.117	55.757	920.429	6.310	-0.3390	1	0	0	109
29/07/04	04:30:00.000	187.614	56.254	915.780	-4.054	-0.3442	1	0	0	96
29/07/04	04:30:30.000	166.740	52.512	952.708	-13.887	-0.3055	1	0	0	83
29/07/04	04:31:00.000	151.874	46.466	1026.704	-22.144	-0.2436	1	0	0	72
29/07/04	04:31:30.000	141.968	39.996	1130.455	-28.502	-0.1819	1	0	0	62
29/07/04	04:32:00.000	135.246	34.007	1256.536	-33.168	-0.1317	1	0	0	54
29/07/04	04:32:30.000	130.490	28.753	1398.830	-36.530	-0.0946	1	0	0	47
29/07/04	04:33:00.000	126.982	24.218	1552.780	-38.947	-0.0683	1	0	0	42
29/07/04	04:33:30.000	124.295	20.300	1715.139	-40.698	-0.0497	1	0	0	38
29/07/04	04:34:00.000	122.174	16.886	1883.613	-41.977	-0.0365	1	0	0	34
29/07/04	04:34:30.000	120.456	13.878	2056.574	-42.920	-0.0270	1	0	0	31
29/07/04	04:35:00.000	119.032	11.197	2232.849	-43.618	-0.0200	1	0	0	28
29/07/04	04:35:30.000	117.831	8.778	2411.580	-44.134	-0.0148	1	0	0	26
29/07/04	04:36:00.000	116.801	6.572	2592.122	-44.513	-0.0108	1	0	0	24
29/07/04	04:36:30.000	115.905	4.541	2773.982	-44.786	-0.0077	1	0	0	22

Figura 7.6 – Redução do período de rastreamento de satélite de menor prioridade

7.3.2 Geração dos Arquivos de Instâncias do Problema

O agente Gerador de Instâncias do Problema gera automaticamente os arquivos de Instâncias do Problema em PDDL 2.2 para as passagens escalonadas para rastreamento por uma determinada estação terrena.

Assim, para a passagem da Figura 7.6, que sofreu redução em seu período de rastreamento, este agente gera um arquivo de Instância do Problema em PDDL. O processo de geração do arquivo de Instância do Problema para esta passagem utiliza: o arquivo de Domínio escrito em PDDL 2.2, previamente criado pelo agente Gerador de Domínio, que encontra-se listado no Apêndice A; Informações do Contexto da Passagem, geradas pelo agente Gerador de Contextos de Passagens; e os níveis de prioridade dos predicados objetivos do domínio de planejamento, redefinidos pelo agente de Priorização de Objetivos para o caso excepcional de passagens que sofreram redução de tempo nos seus períodos de rastreamento.

O resultado do processo de geração do arquivo de Instância do Problema para a passagem da Figura 7.6 consiste no arquivo de Instância do Problema apresentado de forma completa no Apêndice B. A Figura 7.7, a seguir, apresenta um trecho particular deste arquivo referente às definições dos estados iniciais, dos eventos exógenos e do objetivo de planejamento, que é composto por um conjunto de subobjetivos.

Na Figura 7.7, as linhas que contêm o símbolo ;; são linhas de comentário, ou seja, não são linhas de código válidas. Existem algumas observações em negrito, escritas na língua portuguesa, que visam apenas esclarecer o código e, portanto, não fazem parte do código original. Alguns predicados do código PDDL também encontram-se em evidência (negrito) a fim de enfatizá-los, uma vez que estes predicados foram inseridos no arquivo de Instância do Problema da passagem considerada em decorrência das atuações dos agentes da arquitetura MAPSat para tratar o problema de planejamento de controle de múltiplos satélites. A seguir, listam-se as características destes predicados:

- predicado (**ReducedTrack** `PASS60482-5` `SCD1` `CBA`). Este predicado compõe o estado inicial da passagem e informa que a passagem teve o seu período de rastreamento reduzido;

- predicado (`LOSAOSTimeOk PASS60482-5 SCD1 CBA`). Este predicado compõe o objetivo de rastreo da passagem e a sua inserção no arquivo de Instância do Problema deve-se à elevação do nível de prioridade do predicado objetivo “Reservar Janela de Tempo LOS-AOS” do arquivo de Domínio, realizada pelo agente de Priorização de Objetivos. O alcance deste objetivo é essencial para o caso de uma passagem de rastreo reduzido e consiste em reservar o período de tempo mínimo LOS-AOS, ou seja, um período de tempo entre o fim do rastreo da passagem do satélite anterior (SCD2), de maior prioridade de rastreo, e o início do rastreo da passagem do satélite atual (SCD1), de menor prioridade;
- predicados (`at 0.0 (LOSAOSPeriod PASS60482-5 SCD1 CBA)`) e (`at 120.0 (not (LOSAOSPeriod PASS60482-5 SCD1 CBA))`). Estes predicados foram definidos no arquivo de Instância do Problema como eventos exógenos e, em conjunto, estabelecem uma janela de tempo reservada para o período LOS-AOS, que inicia-se no começo da passagem do satélite SCD1 e abrange 120 segundos. Estes eventos exógenos forçam o agente Planejador de Operações de Vão a respeitar esta janela de tempo, ou seja, a não inserir ações de planejamento (operações) durante este período. Os eventos exógenos são eventos que ocorrem em determinados instantes de tempo independentemente das ações escolhidas pelo planejador, ou seja, são eventos com tempos pré-determinados no plano;
- predicados (`TCCondicionadoSendOK PASS60482-5 SCD1 TC160 TC158`), (`TCCondicionadoSendOK PASS60482-5 SCD1 TC159 TC160`) e (`TCCondicionadoSendOK PASS60482-5 SCD1 TC138 TC159`). Estes predicados compõem o objetivo de rastreo da passagem e a inserção destes três predicados, em conjunto, no arquivo de Instância do Problema representa o subobjetivo de rastreo “Executar Manobra de Eixo de Rotação”. Uma manobra de eixo de rotação é realizada para

alterar a atitude de um satélite a fim de que raios solares não incidam em certos locais físicos do satélite. O subobjetivo de rastreo “Executar Manobra de Eixo de Rotação” é alcançado com o envio de um conjunto de telecomandos do tipo condicionado. Um telecomando do tipo condicionado depende da execução prévia de outro telecomando, chamado de telecomando condição, para que ele possa ser executado. Assim, para o predicado de subobjetivo de rastreo (TCCondicionadoSendOK PASS60482-5 SCD1 TC160 TC158) o parâmetro real TC158 representa o telecomando condicionado e o parâmetro real TC160 representa o telecomando condição.

```

;;Definição do Estado inicial do problema de planejamento

(GSAvailable CBA) ← estação terrena CBA disponível para rastreo
(PassTrack PASS60482-5 SCD1 CBA) ← passagem 60482-5 a ser rastreada

(RangeAvailable PASS60482-5 SCD1 CBA) ← medida de distância disponível
para esta passagem
(ReducedTrack PASS60482-5 SCD1 CBA) ← passagem de rastreo reduzido

;;Definição dos Eventos Exógenos da passagem

(at 126.0 (FirstRange PASS60482-5 SCD1 CBA)) ← 1ª medida de distância
(at 216.0 (not (FirstRange PASS60482-5 SCD1 CBA)))

(at 121.0 (FirstTc PASS60482-5 SCD1 TC137)) ← 1º telecomando
(at 123.0 (not (FirstTc PASS60482-5 SCD1 TC137)))

(at 0.0 (LOSAOSPeriod PASS60482-5 SCD1 CBA)) ← período LOS-AOS
(at 120.0 (not (LOSAOSPeriod PASS60482-5 SCD1 CBA)))

(at 390.0 (LastTC PASS60482-5 SCD1 TC138)) ← último telecomando
(at 392.0 (not (LastTC PASS60482-5 SCD1 TC138)))

(at 124.0 (RRateAvailable PASS60482-5 SCD1 CBA)) ← medida de velocidade
(at 125.0 (not (RRateAvailable PASS60482-5 SCD1 CBA)))

;;Definição dos Objetivos a serem atingidos no Plano

(:goal (and
    ← objetivo de planejamento composto por subobjetivos
    (LOSAOSTimeOk PASS60482-5 SCD1 CBA) ← reservar período LOS-AOS
    (LastTCOK PASS60482-5 SCD1 TC138) ← enviar último telecomando
    (RRateOK PASS60482-5 SCD1 CBA RGroup04) ← executar medida de
    velocidade
    (RangeOk PASS60482-5 RGroup03 Range02) ← executar medida de
    distância
    (TCCondicionadoSendOK PASS60482-5 SCD1 TC160 TC158)
    (TCCondicionadoSendOK PASS60482-5 SCD1 TC159 TC160)
    (TCCondicionadoSendOK PASS60482-5 SCD1 TC138 TC159) } executar
    manobra de
    eixo de
    rotação
)
)

```

Figura 7.7 – Arquivo de Instância do Problema para a passagem 60482-5

A Figura 7.8 apresenta o plano (POV) gerado pela arquitetura MAPSat, em especial pelo agente Planejador de Operações de Vôo, para a passagem 60482-5 do satélite SCD1, de período de rastreo reduzido, sobre a estação terrena de Cuiabá. Cada linha do plano gerado é constituída pelos seguintes itens: (i) tempo relativo ao início da passagem, no qual a ação deve ser

executada; (ii) a ação a ser executada, acompanhada dos parâmetros que a identificam; e (iii) tempo necessário para a execução da ação (duração de execução em segundos).

```
; Version LPG-td-1.0
; Seed 2008
; Command line: lpg-td-1.0 -o POVMEDZSRRTCDDOMAIN.PDDL -f
SCD1CBAProblemPass60482-5.PDDL -speed -out SCD1CBAPOVPass60482-5.dat
-seed 2008 -cputime 4
; Problem SCD1CBAProblemPass60482-5.PDDL

0.0003:      (LOSAOSTIMEWINDOW PASS60482-5 SCD1 CBA) [120.0000]
121.0005:    (SENDFIRSTTC PASS60482-5 SCD1 TC137 CBA) [2.0000]
124.0007:    (RANGERATE PASS60482-5 SCD1 CBA RRGROUP04) [1.0000]
125.0010:    (SENDTCCOND PASS60482-5 SCD1 TC138 CBA) [2.0000]
127.0013:    (SENDTCCOND PASS60482-5 SCD1 TC159 CBA) [2.0000]
129.0015:    (SENDTCCOND PASS60482-5 SCD1 TC160 CBA) [2.0000]
131.0021:    (FIRSTRANGE PASS60482-5 SCD1 CBA RGROUP03) [90.0000]
221.0023:    (SENDTCCOND PASS60482-5 SCD1 TC158 CBA) [2.0000]
223.0029:    (RANGE PASS60482-5 SCD1 CBA RGROUP03 RANGE02) [90.0000]
390.0032:    (SENDLASTTC PASS60482-5 SCD1 TC138 CBA) [2.0000]
```

Figura 7.8 – Plano de Operações de Vôo (POV) para a passagem 60482-5

Na Figura 7.8, observa-se que a primeira ação do plano gerado consiste na ação (LOSAOSTIMEWINDOW PASS60482-5 SCD1 CBA) com duração de execução de 120 seg. Esta ação refere-se à reserva de tempo do período LOS-AOS no início da passagem e ocorre exclusivamente em passagens que sofreram redução em seus períodos de rastreo devido a conflitos de horário com outras passagens.

Observa-se que o plano gerado para esta passagem, de rastreo reduzido, não inclui uma ação para realizar medida de calibração antes do início do rastreo. Como comentado anteriormente, esta ação deve ser executada apenas antes do início do período que contém as passagens simultâneas de dois satélites. Assim, no caso de passagens simultâneas sobre uma estação terrena, não é necessário executar a operação de calibração antes do período de rastreo do segundo satélite, pois a operação de calibração executada antes do início do

rastreio do primeiro satélite ainda permanece válida para o rastreio do segundo. Em decorrência disto, considerando que a passagem 60482-5, para a qual gerou-se o plano da Figura 7.8, é a segunda passagem de um período de duas passagens simultâneas (passagens com conflitos de horário) sobre a estação terrena de Cuiabá, não se faz necessário realizar uma medida de distância antes do início do seu rastreio.

A ausência de uma ação para realizar medida de calibração no plano da Figura 7.8 ocorreu por meio da atuação dos agentes da arquitetura MAPSat, em especial do agente de Priorização de Objetivos, que diminuiu o nível de prioridade do predicado objetivo que leva à geração do objetivo de rastreio “Executar Medida de Calibração” e elevou o nível de prioridade do predicado objetivo “Reservar Janela de Tempo LOS-AOS”.

No plano da Figura 7.8, em **negrito e sobre tarjas cinzas**, têm-se as ações referentes ao início e ao término do rastreio, ou seja, a ação de envio do primeiro telecomando da passagem (telecomando 137), para ligar o canal de comunicação no sentido estação terrena e satélite e vice-versa, e a ação de envio do último telecomando (telecomando 138), para desligar o canal de comunicação em ambos os sentidos.

A ação (`RANGERATE PASS60482-5 SCD1 CBA RRGROUP04`) refere-se à execução de uma medida de velocidade, enquanto a ação (`FIRSTRANGE PASS60482-5 SCD1 CBA RGROUP03`) refere-se à execução da primeira medida de distância.

A ação de primeira medida de distância indica que a partir do instante de tempo de sua execução o plano pode conter outras medidas de distância. No caso da passagem 60482-5, uma única outra ação de medida de distância é executada no instante de tempo 223.0029, a ação (`RANGE PASS60482-5 SCD1 CBA RGROUP03 RANGE02`). Caso esta passagem não tivesse sofrido redução no seu período de rastreio, outras medidas de distância seriam realizadas, pois em

uma passagem procura-se realizar a maior quantidade de medidas de distância possível. As medidas coletadas são utilizadas para a geração do próximo arquivo PVP (arquivo de Previsão de Visibilidade de Passagens).

Por fim, têm-se um conjunto de ações referente ao alcance do subobjetivo de rastreo “Executar Manobra de Eixo de Rotação”. Este conjunto de ações é constituído pelas ações relativas ao envio dos telecomandos 138, 159, 160 e 158, inseridas no plano na ordem correta de envio. Estas ações são: (SENDTCCOND PASS60482-5 SCD1 **TC138** CBA), (SENDTCCOND PASS60482-5 SCD1 **TC159** CBA), (SENDTCCOND PASS60482-5 SCD1 **TC160** CBA) e (SENDTCCOND PASS60482-5 SCD1 **TC158** CBA).

8 CONCLUSÕES

Este trabalho propõe uma arquitetura multi-agente de planejamento para o controle de múltiplos satélites.

A arquitetura proposta, referenciada pela sigla MAPSat, provê os seguintes serviços: automaticamente planeja as operações a serem enviadas aos satélites durante os seus respectivos períodos de rastreo, considerando a restrição de tempo destes períodos; e soluciona o problema de satélites com períodos de rastreo com horários conflitantes em relação a uma mesma estação terrena. Estes serviços são providos por meio da atuação de um grupo de agentes, que trabalham de forma cooperativa.

Para o projeto dos agentes que compõem a arquitetura MAPSat empregou-se a metodologia de Engenharia de Software Orientada a Agentes MESSAGE, que conduziu o processo de levantamento e detalhamento dos agentes que compõem a arquitetura. Para a escolha desta metodologia utilizou-se um *framework* de avaliação de metodologias orientadas a agentes, proposto neste trabalho de pesquisa, denominado *framework* AMSisPlan, que a partir dos resultados da avaliação das metodologias MAS-CommonKADS, Tropos e MESSAGE indicou esta última metodologia como a mais adequada à modelagem do sistema multi-agente de planejamento desejado.

A atividade de planejamento da arquitetura MAPSat utiliza a técnica de Planejamento de IA para a geração dos planos de controle de satélites. O uso desta técnica possibilita: empregar planejadores de renome, eficientes e bem testados, disponibilizados pela comunidade de Planejamento de IA; e adotar uma linguagem de representação específica para problemas de planejamento a fim de representar o ambiente de forma bem definida, facilitando a tarefa de planejamento. A arquitetura MAPSat adota o planejador LPG-TD e a linguagem de representação de problemas de planejamento PDDL 2.2.

A seguir, são apresentadas as principais contribuições deste trabalho de pesquisa e os trabalhos futuros vislumbrados.

8.1 Contribuições

Este trabalho oferece contribuições significativas para os dois contextos de um trabalho de pesquisa: o contexto teórico, definido por técnicas fundamentadas, e o contexto da aplicação, definido pela aplicação destas técnicas.

No contexto teórico esta pesquisa propõe um mecanismo inovador de priorização de objetivos de planejamento que considera as prioridades dos objetivos durante o processo de geração automática das entradas para o processo de planejamento. Este mecanismo é representado na arquitetura MAPSat pelos fluxos de tarefas do agente de Priorização de Objetivos e do agente Gerador de Instâncias do Problema, que atuam em conjunto para a geração dos arquivos de Instâncias do Problema em PDDL. Estes agentes inserem nestes arquivos PDDL somente os objetivos de rastreios relevantes para o processo de geração de planos a ser iniciado (geração de POVs).

Na literatura da área de Planejamento em IA verificou-se a existência da proposta de Coddington (2002) que considera prioridades para objetivos de planejamento. No entanto, nesta proposta gera-se um plano para posteriormente editá-lo, ou seja, para posteriormente remover as ações que permitem o alcance dos objetivos de menor prioridade. Diferencialmente, o mecanismo proposto neste trabalho de pesquisa impede que um objetivo de prioridade baixa pertença às definições de entrada para o processo de planejamento, evitando, com isso, a necessidade de edição do plano gerado e o decorrente custo desta edição.

Este mecanismo de desconsiderar objetivos por nível de prioridade permite a geração de planos que são ajustáveis às limitações de tempo dos períodos de rastreio dos satélites e que atendem aos principais objetivos de rastreio.

Uma segunda contribuição no contexto teórico desta pesquisa é a proposta do *framework* AMSisPlan, um *framework* de Avaliação de Metodologias para Sistemas de Planejamento. A proposta deste *framework* baseou-se nas seguintes motivações: a tarefa de comparação entre metodologias é considerada uma tarefa complexa devido à heterogeneidade das metodologias atuais; e a não-existência de critérios de avaliação de metodologias orientadas a agentes voltados aos requisitos dos sistemas de planejamento. Assim, o *framework* AMSisPlan contribui no sentido de oferecer um conjunto de critérios para a escolha de uma metodologia orientada a agentes especialmente adequada à modelagem de sistemas de planejamento.

No contexto da aplicação deste trabalho de pesquisa, a proposta da arquitetura MAPSat e o desenvolvimento de um protótipo da arquitetura representam contribuições à atividade de controle de satélites do INPE por proverem automatização à esta atividade. Quando operacionalizada, espera-se que a arquitetura MAPSat seja, de fato, uma alternativa para a redução dos custos operacionais relacionados ao controle de satélites em órbita.

Esta expectativa baseia-se nas contribuições reais obtidas com a implementação do protótipo da arquitetura MAPSat. A implementação do protótipo permitiu verificar que o uso da técnica de Planejamento de IA e, em especial, da representação do problema de planejamento de controle de satélites por meio da linguagem PDDL, conferem configurabilidade à atividade de planejamento de controle de satélites.

A característica de configurabilidade constitui uma forte contribuição ao problema de controle de satélites. Frente ao projeto de aumento do número de satélites a serem controlados pelo INPE e das restrições orçamentárias para expansão da infra-estrutura de apoio ao controle destes satélites, a arquitetura MAPSat torna possível rastrear novos e diferentes satélites reutilizando um único domínio de planejamento, gerado pelo agente Gerador de Domínio, e gerando as seqüências de operações particulares a cada rastreio (POVs) de

forma automática. Dessa forma, facilitam-se as atividades dos profissionais responsáveis pelo planejamento de controle de satélites a fim de que possam atender à necessidade de controle de um maior número de satélites.

Uma segunda contribuição, no contexto da aplicação deste trabalho de pesquisa, é a característica de escalabilidade. A arquitetura MAPSat também confere escalabilidade à atividade de controle de satélites por permitir gerar planos para o controle de múltiplos satélites, independente da quantidade, alterando automaticamente somente as informações que tratam as particularidades de cada rastreo (arquivos de Instâncias do Problema).

Uma última contribuição, inovadora no contexto da aplicação, é a aplicação em si de um conjunto de técnicas a um problema do mundo real, ou seja, ao problema de planejamento de controle de satélites do INPE. O processo de concepção e implementação da arquitetura MAPSat empregou o seguinte conjunto de técnicas: o conceito de agentes como elemento principal do projeto arquitetural, a modelagem orientada a agentes segundo a metodologia MESSAGE, a representação do problema de planejamento segundo as definições da linguagem PDDL e a geração de planos segundo a técnica de Planejamento de IA (planejador LPG-TD).

Em relação a publicações, este trabalho de pesquisa gerou publicações em periódicos e anais de congressos nacionais e internacionais. Estas publicações encontram-se listadas no Apêndice C.

8.2 Trabalhos Futuros

A arquitetura MAPSat gera planos (POVs) para o controle de satélites em órbita, incluindo o envio de telecomandos de tempo real e de telecomandos temporizados (*time-tagged telecommands*). Ao contrário dos telecomandos de tempo real, os telecomandos temporizados são executados quando um satélite encontra-se fora da visibilidade de estações terrenas. Estes telecomandos são armazenados na memória de bordo do satélite e contêm referências temporais,

que indicam ao computador de bordo do satélite quando eles devem ser executados.

A arquitetura MAPSat planeja o envio de telecomandos temporizados, mas não considera a restrição de armazenamento destes telecomandos nas memórias de bordo dos satélites. Assim, podem ser obtidos avanços no sentido de prover planos que atendam à capacidade de armazenamento das memórias de bordo.

Um segundo aspecto observado refere-se à atuação do agente Escalonador de Rastreios da arquitetura MAPSat. Este agente gerencia o uso compartilhado de uma estação terrena para o rastreo de múltiplos satélites. No entanto, o comportamento deste agente poderia ser estendido para também atuar na resolução do conflito de visibilidade de uma passagem de satélite, ou parte desta, a duas ou mais estações terrenas simultaneamente.

Outro aspecto a ser considerado refere-se ao mecanismo de prioridade de objetivos de planejamento utilizado pelos agentes da arquitetura MAPSat. Pretende-se analisar se existem casos, no ambiente de controle de satélites, em que a satisfação de um objetivo de alta prioridade depende da satisfação prévia de um objetivo de menor prioridade. Caso afirmativo, seria necessário incorporar à arquitetura MAPSat um processo de análise de relacionamentos de dependência entre objetivos. A arquitetura MAPSat contempla o planejamento de telecomandos condicionados, ou seja, de telecomandos que para serem executados necessitam da execução prévia de outros telecomandos. Este recurso deve contribuir para a futura análise de dependência entre objetivos de rastreo.

Por fim, o último aspecto a ser explorado constitui a proposta de um agente Executor, pertencente à organização de Sistema de Execução de Operações, que provenha o serviço de execução das operações contidas nos planos (POVs), gerados pela arquitetura MAPSat.

REFERÊNCIAS BIBLIOGRÁFICAS

BERNON, C.; GLEIZES, M. P.; PICARD, G.; GLIZE, P. The ADELFE methodology for an intranet system design. In: International Bi-Conference Workshop on Agent-Oriented Information Systems (AOIS), 4., 27-28 May 2002, Toronto, Ontario, Canada. **Proceedings...** [s.n.], 2002, p. 1-15.

BONET, B.; LOERINCS, G.; GEFNER, H. A robust and fast action selection mechanism for planning. In: National Conference on Artificial Intelligence (AAAI-97), 14., 27-31 Jul 1997, Providence, RI, USA. **Proceedings...** Providence: AAAI/MIT Press, 1997. p. 714-719.

BLUM, A.; FURST, M. Fast planning through planning graph analysis. In: International Joint Conferences on Artificial Intelligence (IJCAI-95), 14., 20-25 Aug 1995, Montreal, Canada. **Proceedings...** Montreal: Morgan Kaufmann, 1995. p. 1636–1642.

BRESCIANI, P.; GIORGINI, P.; GIUNCHIGLIA, F.; MYLOPOLOUS, J.; PERINI, A. Tropos: An agent-oriented software development methodology. **Autonomous Agents and Multi-Agent Systems**, v. 8, n. 3, p. 203-236, 2004.

CAIRE, G.; COULIER, W.; GARIJO, F.; GOMEZ, J.; PAVON, J.; LEAL, F.; CHAINHO, P.; KEARNEY, P.; STARK, J.; EVANS, R.; MASSONET, P. Agent-oriented analysis using MESSAGE/UML. **Lecture Notes in Computer Science**. v. 2222, p. 119-135, 2002. Agent-oriented software engineering II (AOSE-2001), Montreal, Canada, 29 May 2001.

CARDOSO, L. S. **Aplicação da tecnologia de agentes de planejamento em operações de satélites**. 2006. 167 p. (INPE-14092-TDI/1075). Dissertação (Mestrado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos. 2006. Disponível em: <<http://urlib.net/sid.inpe.br/MTC-m13@80/2006/05.04.17.36>>. Acesso em: 25 jan. 2009.

CONSULTATIVE COMMITTEE FOR SPACE DATA SYSTEMS (CCSDS). **Telemetry – summary of concept and rationale**. Washington, D.C., USA, December 1987. 56 p. Report Concerning Space Data Systems Standards, CCSDS 100.0-G-1. Green Book. Issue 1. Disponível em: <<http://public.ccsds.org/publications/archive/100x0g1.pdf>>. Acesso em: 26 out 2008.

CONSULTATIVE COMMITTEE FOR SPACE DATA SYSTEMS (CCSDS). **Telecommand – summary of concept and rationale**. Washington, D.C., USA, December 1987. Report Concerning Space Data Systems Standards, CCSDS 200.0-G-6. Green Book. Issue 6. Disponível em: <<http://public.ccsds.org/publications/archive/200x0g6.pdf>>. Acesso em: 26 out 2008.

CESTA, A.; CORTELLESA, G.; ODDI, A.; POLICELLA, N. Studying decision support for MARS EXPRESS planning tasks: a report from the MEXAR experience. In: International Workshop on Planning and Scheduling for Space, 4., 2004, Darmstadt, Germany. **Proceedings...** Darmstadt: ESA-ESOC, 2004.

CESTA, A.; CORTELESSA, G.; FRATINI, S.; ODDI, A.; DENIS, M.; DONATI, A.; POLICELLA, N.; RABENAU, E.; SCHULSTER, J. Mexar2: AI Solves Mission Planner Problems. **IEEE Intelligent Systems**, v. 22, n. 4, p. 12-19, 2007.

CHIEN, S.; SHERWOOD, R.; TRAN, D.; CASTANO, R.; CICHY, B.; DAVIES, A.; RABIDEAU, G.; TANG, N.; BURL, M.; MANDL, D.; FRYE, S.; HENGEMIHLE, J.; D'AGOSTINO, J.; BOTE, R.; TROUT, B.; SHULMAN, S.; UNGAR, S.; VAN GAASBECK, J.; BOYER, D.; GRIFFIN, M.; BURKE, H.; GREELEY, R.; DOGGETT, T.; WILLIAMS, K.; BAKER, V.; DOHM, J. Autonomous science on the EO-1 mission. In: International Symposium on Artificial Intelligence Robotics and Automation in Space (I-SAIRAS), 7., 2003, Nara, Japan. **Proceedings...** Nara: JAXA, 2003.

CHIEN, S.; SHERWOOD, R.; TRAN, D.; CICHY, B.; RABIDEAU, G.; CASTANO, R.; DAVIES, A.; LEE, R.; MANDL, D.; FRYE, S.; TROUT, B.; D'AGOSTINO, J.; SHULMAN, S.; BOYER, D.; HAYDEN, S.; SWEET, A.; CHRISTA, S. Lessons learned from autonomous sciencecraft experiment. In: International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2005), 4., 2005, Utrecht, Netherlands. **Proceedings...** Utrecht: ACM Press, 2005.

CODDINGTON, A. A continuous planning framework with durative actions. In: International Symposium on Temporal Representation and Reasoning, 9., Jul 2002, Manchester, UK. **Proceedings...** Manchester, 2002. p. 108-114.

CODDINGTON, A. M.; LUCK, M. A motivation based planning and execution framework. **International Journal on Artificial Intelligence Tools**, v. 13, n. 1, p. 5-25, 2004.

CONTE, R.; CASTELFRANCHI, C. Mind is not enough: precognitive bases of social interaction. In: Symposium on Simulating Societies, 1992, Guildford, UK. **Proceedings...** 1992, p. 93-110.

COSENTINO, M. From requirements to code with the PASSI methodology. In: B. Henderson-Sellers & P. Giorgini (Eds.), **Agent-oriented methodologies**, chapter 4. Hershey, PA: Idea Group, 2005.

COLEMAN, D.; ARNOLD, P.; BODOFF, S.; DOLLIN, C.; GILCHRIST, H. **Object oriented development: the fusion method**. Englewood Cliffs, New Jersey: Prentice Hall, 1994.

DAM, K. H.; WINIKOFF, M. Comparing agent-oriented methodologies. In: International Bi-Conference Workshop on Agent-Oriented Information Systems, 5., 2003, Melbourne, Australia. **Proceedings...** Melbourne, Australia, 2003. p. 52-59.

DEBENHAM, J.; HENDERSON-SELLERS, B. Designing agent-based process systems - Extending the OPEN Process Framework. In: **Intelligent agent software engineering**, chapter VIII. Hershey, PA: Idea Group Publishing, 2003. p. 160-190.

EUROPEAN COOPERATION FOR SPACE STANDARDIZATION. **ECSS-E-70: Space Engineering – Ground Systems and Operations – Part 1: Principles and Requirements**, ESA-ESTEC, Requirements & Standards Division. Noordwijk, The Netherlands, Apr 2000.

EDELKAMP, S.; HOFFMANN, J. **PDDL 2.2: the language for the classical part of the 4th International Planning Competition**. Freiburg, Germany, Albert Ludwigs Universität Institut für Informatik: Technical Report n.195, Jan 2004.

ELDER, C. E.; PAYNE, A. V. RADARSAT-2 planning system. In: The Space Operations Conference (SpaceOps 2004), May 2004, Montreal, Canada. **Proceedings...** Montreal: AIAA, 2004.

FEBER, J. **Multi-agent systems: an introduction to distributed artificial intelligence**. London: Addison-Wesley (ed). 1999. 509 p. (ISBN 0-201-36048-9)

FERRI, P.; PECCHIOLI, M.; PENA, A. Introducing automation in ground segment operations. In: International Conference on Space Operations (SpaceOps 2004), 8., 17 – 21 May 2004, Montreal, Quebec, Canada. **Proceedings...** Montreal, Canada: American Institute of Aeronautics and Astronautics, 2004.

FIKES, R. E.; NILSSON, N. J. STRIPS: a new approach to the application of theorem proving to problem-solving, **Artificial Intelligence**, v. 2, n. 3 - 4, p. 189-208, 1971.

FOUNDATION FOR INTELLIGENT PHYSICAL AGENTS (FIPA). **FIPA Abstract Architecture Specification**, 2002. Disponível em: <<http://www.fipa.org/specs/fipa00001/index.html>>. Acesso em 27 out 2008.

FOX, M.; LONG, D. PDDL2.1: An extension to PDDL for expressing temporal planning domains, **Journal of Artificial Intelligence Research**, v.20, p. 61-124, Dec 2003. (ISSN-1076-9757).

FRANK, J.; JONSSON, A. K. Constraint-based attribute and interval planning. **Journal of Constraints**, v. 8, n. 4, p. 339-364, Oct 2003.

FRANKLIN, S.; GRAESSER, A. Is it an agent, or just a program? A taxonomy for autonomous agents. In: International Workshop on Agent Theories, Architectures, and Languages (ATAL), 3., 1996, Berlin, Germany. **Proceedings...** Springer-Verlag, 1996. (ISBN 3-540-62507-0).

FRATINI, S.; CESTA, A. A planning and scheduling tool to automate and support mission planning. In: International Conference on Space Operations (SpaceOps), 9., Jun 2006, Rome, Italy. **Proceedings...** Rome: AIAA, 2006.

GEREVINI, A.; SAETTI, A.; SERINIA, I. Planning through stochastic local search and temporal action graphs in LPG. **Journal of Artificial Intelligence Research**, v.20, p. 239-290, Dec 2003. (ISSN-1076-9757).

GEREVINI, A.; SAETTI, A.; SERINA, I. Integrating planning and temporal reasoning for domains with durations and time windows. In: International Joint Conference on Artificial Intelligence (IJCAI), 19., 30 Jul – 05 Aug 2005, Edinburgh, Scotland, UK. **Proceedings...** Edinburgh, Scotland, UK: IJCAI-Inc, 2005.

HARRIS, W.; BLAKE, R.; WOODS, D.; THOMPSON, R. Mission planning for constellations. In: The Space Operations Conference (SpaceOps), Oct 2002, Houston, Texas. **Proceedings...** Texas: AIAA, 2002.

HENDERSON-SELLERS, B.; GIORGINI, P. **Agent-oriented methodologies**. London, United Kingdom: Idea Group Publishing, 2005. 413 p. (ISBN 1-59140-587-4).

HÜBNER, J. F.; SICHMAN, J. S. A. Um modelo de organização para SMA direcionado à reorganização. In: Workshop of the International Joint Conference IBERAMIA-SBIA, 2000, São Carlos, ICMC/USP. **Proceedings...** São Carlos, 2000. p. 191-199.

IGLESIAS, C. A.; GARIJO, M. The agent oriented methodology MASCommonKADS. Brian Henderson-Sellers e Paolo Giorgini (ed.). In: **Agent-oriented methodologies**. IDEA Group Publishing, 2005. p. 46-78.

IGLESIAS, C. A.; GARIJO, M.; GONZALEZ, J. C.; VELASCO, J. R. Analysis and design of multi-agent systems using MAS-CommonKADS. **Lecture Notes in Artificial Intelligence**, v. 1365, p. 313-326, 1998. In: Intelligent Agents: Agent theories, architectures, and languages, 4., 1998, Berlin: Springer-Verlag.

JONSSON, A.; MORRIS, P.; MUSCETTOLA, N.; RAJAN, K. Planning in interplanetary space: theory and practice. In: International Conference on Artificial Intelligence Planning and Scheduling (AIPS), 5., 2000, Breckenridge, CO, USA. **Proceedings...** 2000. p. 177-186.

KAUTZ, H.; SELMAN, B. Planning as satisfiability. In: European Conference on Artificial Intelligence (ECAI), 10., 1992, Vienna, Austria. **Proceedings...** Chichester: John Wiley and Sons, 1992. p. 359–363.

KAUTZ, H.; MCALLESTER, D.; SELMAN, B. Encoding plans in propositional logic. In: International Conference on Principles of Knowledge Representation and Reasoning, 5., 5 – 8 Nov 1996, Cambridge, MA, USA. **Proceedings...** San Francisco: Morgan Kaufmann, 1996, p. 374-385. (ISBN 1-55860-421-9).

KINNY, D.; GEORGEFF, M.; RAO, A. A methodology and modelling techniques for systems of BDI agents. In: European Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW), 7., 1996. **Proceedings...** Springer-Verlag, 1996. p. 56-71.

KUCINSKIS, F. N. Alocação dinâmica de recursos computacionais para experimentos científicos com replanejamento automatizado a bordo de satélites. 2007. 165 p. (INPE-14798-TDI/1241). Dissertação (Mestrado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos. 2007. Disponível em: <<http://urlib.net/sid.inpe.br/mtc-m17@80/2007/04.23.11.42>>. Acesso em: 25 jan. 2009.

LARSON, W. J.; WERTZ, J. R. **Space mission analysis and design**. 3a. Ed. Torrance, and Dordrecht: Microcosm, Inc. and Kluwer Academic Publishers. 1999. 827 p. (ISBN 1-881883-01-9).

LIEBERMAN, H. Autonomous interface agents. In: Conference on Human Factors in Computing Systems, 1997, Atlanta, U.S.A. **Proceedings...** Atlanta: ACM/SIGCHI, 1997. Disponível em: <<http://www.sigchi.org/chi97/proceedings/paper/hl.htm>>. Acesso em: 01 nov 2008.

MCALLESTER, D.; ROSENBLITT, D. Systematic nonlinear planning. In: National Conference on Artificial Intelligence (AAAI-91), 9., 1991, Anaheim, CA, USA. **Proceedings...** Anaheim: AAAI Press, 1991. p. 634-639.

MCDERMOTT, D. **PDDL - the planning domain definition language**. Yale, CT, USA. 1998. Technical Report CVC TR-98-003/DCS, Yale Center for Computational Vision and Control.

MILGROM, E. **MESSAGE: Methodology for Engineering Systems of Software Agents**. Project P907 Report, EDIN 0215-0907, EURESCOM – European Institute for Research and Strategic Studies in Telecommunications GmbH, Sept., 2001. Disponível em: <http://www.eurescom.de/~public-webspace/P900-series/P907/index.htm>. Acesso em: 01 nov 2008.

MINTON, S.; KNOBLOCK, C.; KUOKKA, D.; GIL, Y.; JOSEPH, R.; CARBONELL, J. **Prodigy 2.0: the manual and tutorial**. Pittsburg, PA, USA, 1989. Technical Report CMU-CS-89-146, University of Carnegie Mellon.

ODDI, A.; POLICELLA, N.; CESTA, A.; CORTELLESSA, G. Generating high quality schedules for a spacecraft memory downlink problem. **Lecture Notes in Computer Science (LNCS)**, v. 2833, 2003.

ODDI, A.; POLICELLA, N. A max-flow approach for improving robustness in a spacecraft downlink schedule. In: International Workshop on Planning and Scheduling for Space, 4., Jun 2004, Darmstadt, Germany. **Proceedings...** 2004.

ODDI, A.; POLICELLA, N.; CESTA, A.; CORTELLESSA, G. Constraint-based random search for solving spacecraft downlink scheduling problems. In: KENDALL, G.; BURKE, E. K.; PETROVIC, S.; GENDREAU, M. (Ed.). **Multidisciplinary Scheduling: Theory and Applications**. Springer, 2005. p. 133-162. ISBN (978-0-387-25266-7).

ODELL, J.; PARUNAK, H. V. D.; BAUER, B. Extending UML for agents. In: National Conference on Artificial Intelligence, 17., 2000. **Proceedings...** 2000.

PADGHAM, L.; WINIKOFF, M. Prometheus: A methodology for developing intelligent agents. **Lecture Notes in Computer Science (LNCS)**, v. 2585, p. 174-185, 2003. Special issue on the Third International Workshop on Agent-Oriented Software Engineering (AOSE'02). ISBN 978-3-540-00713-5.

PAVÓN, J.; GOMEZ-SANZ, J.; FUENTES, R. **The INGENIAS methodology and tools**. In: Agent-oriented methodologies (Chapter 4). Hershey, PA: Idea Group, 2005.

PENBERTHY, J. S.; WELD, D. S. UCPOP: A sound, complete, partial-order planner for ADL. In: International Conference on Principles of Knowledge Representation and Reasoning (KR-92), 3., 1992, Cambridge, MA, USA. **Proceedings...** San Francisco: Morgan Kaufmann Publishers Inc., 1992.

PERINI A.; SUSI A.; GIUNCHIGLIA F. Coordination specification in multi-agent systems: from requirements to architecture with the Tropos methodology. In: International Conference on Software Engineering and Knowledge Engineering (SEKE), 14., Jul 2002, Ischia, Italy. **Proceedings...** Ischia, Italy: ACM Press, 2002. p. 51-54.

PRIETULA, M. L.; CARLEY, K. M.; GASSER, L. A computational approach to organizations and organizing. In: _____ (Ed.). **Simulating Organizations: Computational Models of Institutions and Groups**, chapter 1. MA, USA: MIT Press Cambridge, 1998. p. xiii-xix.

RABENAU, E.; DONATI, A.; DENIS, M.; POLICELLA, N.; SCHULSTER, J.; CESTA, A.; CORTELLESA, G.; ODDI, A.; FRATINI, S. The RAXEM tool on mars express – uplink planning optimisation and scheduling using AI constraint resolution. In: Space Operations Conference (SpaceOps), 10, 2008, Heidelberg, Germany. **Proceedings...** Heidelberg, Germany: AIAA, 2008.

REZENDE, S. O. **Sistemas inteligentes: fundamentos e aplicações**. Barueri, SP: editora Manole, 2003. ISBN 85-204-1683-7.

RUMBAUGH, J.; BLAHA, M.; PREMERLANI, W.; EDDY, F.; LORENSEN, W. **Object-oriented modeling and design**. Englewood Cliffs, New Jersey: Prentice-Hall, 1991.

RUMBAUGH, J.; JACOBSON, I.; BOOCH, G. **The unified modeling language reference manual**. 2. ed. Addison-Wesley, 2004.

RUSSELL, S.; NORVIG, P. **Inteligência artificial**. 2. ed. São Paulo: Editora Campus, 2004. 1040 p. ISBN (8535211772).

SHERWOOD, R.; CHIEN, S.; TRAN, D.; CICHY, B.; CASTANO, R.; DAVIES, A.; RABIDEAU, G. Operating the autonomous sciencecraft experiment. In: International Conference on Space Operations (SPACEOPS), 8., 17 – 21 May 2004, Montreal, Quebec, Canada. **Proceedings...** Montreal, Canada: American Institute of Aeronautics and Astronautics, 2004.

STURM, A.; SHEHORY, O. A framework for evaluating agent-oriented methodologies. In: International Bi-Conference Workshop on Agent-Oriented Information Systems (AOIS), 5., 14 Jul 2003, Melbourne, Australia. **Proceedings...** Melbourne, Australia, 2003. p. 60-67.

SYCARA, K. P. The many faces of agents. **Artificial Intelligence Magazine**, v. 19, n. 2, p. 11-12, 1998.

TAVETER, K.; WAGNER, G. Towards radical agent-oriented software engineering processes based on AOR modelling. In: HENDERSON-SELLERS, B.; GIORGINI, P. (Ed.). **Agent-oriented methodologies**. Chapter 10. Hershey, PA: Idea Group, 2005.

TRAN, Q. N.; LOW, G.; WILLIAMS, M. A. A feature analysis framework for evaluating multi-agent system development methodologies. In: International Symposium on Methodologies for Intelligent Systems (ISMIS), 14., 28-31 Oct. 2003, Maebashi, Japan. **Proceedings...** Maebashi, Japan: Springer-Verlag, 2003. p. 613-617.

WAGNER, G. The agent-object relationship metamodel: Towards a unified view of state and behaviour. **Information Systems**, v. 28, n. 5, p. 475-504, 2003.

WEISS, G. **Multiagent systems: a modern approach to distributed modern approach to artificial intelligence**. Cambridge, Massachusetts: The MIT Press, 1999. 619 p. ISBN 0-262-23203-0.

WELD, D. S. Recent Advances in AI Planning. **AI Magazine**, v. 20, n. 2, p. 93 - 123, Oct 1999.

WERNECK, V. M. B.; PEREIRA, L. F.; SILVA, T. S.; ALMENTERO, E. K.; CYSNEIROS, L. M. Uma avaliação da metodologia MAS-CommonKADS. In: Workshop on Software Engineering for Agent-oriented Systems (SEAS), 3., Oct. 2006, Florianópolis, Brasil. **Proceedings...** Florianópolis: Sociedade Brasileira de Computação, 2006. p. 25-36.

WOOD, M.; DELOACH, S. A. An overview of the multiagent systems engineering methodology. **Lecture Notes in Computer Science (LNCS)**, v. 1957, p. 207-222, Springer-Verlag, 2001. Special Issue on the First International Workshop on Agent-Oriented Software Engineering (AOSE 2000).

WOOLDRIDGE, M.; JENNINGS, N. R.; KINNY, D. The Gaia methodology for agent-oriented analysis and design. **Journal Autonomous Agents and Multi-Agent Systems**, v. 3, p. 285-312, 2000.

YU, E. **Modelling strategic relationships for process reengineering**. PhD Thesis - University of Toronto, Department of Computer Science, Toronto, 1995.

ZAMBONELLI, F.; JENNINGS, N.; WOOLDRIDGE, M. Developing multiagent systems: The Gaia methodology. **ACM Transactions on Software Engineering and Methodology**, v. 12, n. 3, p. 317-370, 2003.

APÊNDICE A - EXEMPLO DE ARQUIVO DE DOMÍNIO

A seguir, apresenta-se o arquivo de Domínio codificado na linguagem PDDL 2.2, gerado pelo agente Gerador de Domínio da arquitetura MAPSat, para o problema de planejamento de controle de satélites.

```
;; Domínio - Controle de Satélites

(define (domain POVMEDZSRRTC)

  (:requirements :equality :typing :fluents :durative-actions :timed-initial-
    literals)

  (:types PassNumber SatCode GSCode RangeGroupNumber RangeNumber ZSNumber
    RRateGroupNumber TCCode)

  (:predicates
    ;;Internal Events
    (TCSent ?pass - PassNumber ?sat - SatCode)
    (FirstTcOK ?pass - PassNumber ?sat - SatCode ?tc - TCCode)
    (TCCondSendOK ?pass - PassNumber ?sat - SatCode ?tcCond - TCCode)
    ;;Initial Environment States
    (GSAvailable ?gs - GSCode)
    (PassTrack ?pass - PassNumber ?sat - SatCode ?gs - GSCode)
    (NotPassTrack ?pass - PassNumber ?sat - SatCode ?gs - GSCode)
    (RangeAvailable ?pass - PassNumber ?sat - SatCode ?gs - GSCode)
    (TCNotAvailable ?pass - PassNumber ?sat - SatCode)
    (ReducedTrack ?pass - PassNumber ?sat - SatCode ?gs - GSCode)
    ;;Exogenous Events
    (LOSAOSPeriod ?pass - PassNumber ?sat - SatCode ?gs - GSCode)
    (CalibrationAvailable ?pass - PassNumber ?gs - GSCode)
    (FirstRange ?pass - PassNumber ?sat - SatCode ?gs - GSCode)
    (RRateAvailable ?pass - PassNumber ?sat - SatCode ?gs - GSCode)
    (SatSilentConeAvailable ?pass - PassNumber ?sat - SatCode ?gs - GSCode ?ZS -
      ZSNumber)
    (FirstTc ?pass - PassNumber ?sat - SatCode ?tc - TCCode)
    (LastTC ?pass - PassNumber ?sat - SatCode ?tc - TCCode)
    (TimeTaggedTc ?pass - PassNumber ?sat - SatCode ?tc - TCCode)
    ;;Goals
    (LOSAOSTimeOk ?pass - PassNumber ?sat - SatCode ?gs - GSCode)
    (CalibrationOk ?pass - PassNumber ?gs - GSCode)
    (FirstRangeOk ?pass - PassNumber ?rgroup - RangeGroupNumber)
    (RangeOk ?pass - PassNumber ?rgroup - RangeGroupNumber ?rnum - RangeNumber)
    (RRateOK ?pass - PassNumber ?sat - SatCode ?gs - GSCode ?RRGroup -
      RRateGroupNumber)
    (SilentZone ?pass - PassNumber ?sat - SatCode ?gs - GSCode ?ZS - ZSNumber)
    (TimeTaggedTCOK ?pass - PassNumber ?sat - SatCode ?tc - TCCode)
    (TCSendOK ?pass - PassNumber ?sat - SatCode ?tc - TCCode)
```

```

(TCCondicionadoSendOK ?pass - PassNumber ?sat - SatCode ?tcCond - TCCode ?tc
    - TCCode)
(PassEliminated ?pass - PassNumber ?sat - SatCode ?gs - GSCode)
(LastTCOK ?pass - PassNumber ?sat - SatCode ?tc - TCCode)
)

(:functions
  (LOSAOSTime ?pass - PassNumber ?sat - SatCode ?gs - GSCode)
  (CalibrationTime ?gs - GSCode ?cgroup - CalibGroupNumber)
  (RangeTime ?gs - GSCode ?rgroup - RangeGroupNumber)
  (PassTime ?pass - PassNumber)
  (ZoneSilentTime ?pass - PassNumber ?ZS - ZSNumber)
  (TcTime ?tc - TCCode)
)

;; Eliminação de passagem
(:durative-action PassEliminate
  :parameters (?pass - PassNumber ?sat - SatCode ?gs - GSCode)
  :duration (= ?duration (PassTime ?pass))
  :condition (and (over all (NotPassTrack ?pass ?sat ?gs))
    (at start (GSAvailable ?gs))
  )
  :effect (and (at end (PassEliminated ?pass ?sat ?gs))
    (at start (not (GSAvailable ?gs)))
    (at end (GSAvailable ?gs))
  )
)

;; Calibração antes da passagem
(:durative-action Calibrate
  :parameters (?pass - PassNumber ?sat - SatCode ?gs - GSCode ?cgroup -
    CalibGroupNumber)
  :duration (= ?duration (CalibrationTime ?gs ?cgroup))
  :condition (and (at start (GSAvailable ?gs))
    (at start (CalibrationAvailable ?pass ?gs))
    (over all (PassTrack ?pass ?sat ?gs))
    (over all (RangeAvailable ?pass ?sat ?gs))
  )
  :effect (and (at end (CalibrationOK ?pass ?cgroup))
    (at start (not (GSAvailable ?gs)))
    (at end (GSAvailable ?gs))
    (at end (not (CalibrationAvailable ?pass ?gs)))
  )
)

;; Período de tempo LOS-AOS antes do início do rastreo da passagem
(:durative-action LOSAOSTimeWindow
  :parameters (?pass - PassNumber ?sat - SatCode ?gs - GSCode)
  :duration (= ?duration (LOSAOSTime ?pass ?sat ?gs))
  :condition ( and (at start (GSAvailable ?gs))
    (at start (ReducedTrack ?pass ?sat ?gs))
    (at start (LOSAOSPeriod ?pass ?sat ?gs))
    (over all (PassTrack ?pass ?sat ?gs))
  )
  :effect (and (at start (not (GSAvailable ?gs)))

```

```

        (at end (GSAvailable ?gs))
        (at end (not (LOSAOSPeriod ?pass ?sat ?gs)))
        (at end (LOSAOSTimeOk ?pass ?sat ?gs))
    )
)

;; Primeira medida de distância
(:durative-action FirstRange
 :parameters (?pass - PassNumber ?sat - SatCode ?gs - GSCode ?rgroup -
              RangeGroupNumber)
 :duration (= ?duration (RangeTime ?gs ?rgroup))
 :condition (and (at start (GSAvailable ?gs))
                 (at start (FirstRange ?pass ?sat ?gs))
                 (over all (PassTrack ?pass ?sat ?gs))
                 (over all (RangeAvailable ?pass ?sat ?gs))
                )
 :effect (and (at end (FirstRangeOK ?pass ?rgroup))
              (at start (not (GSAvailable ?gs)))
              (at end (GSAvailable ?gs))
             )
)

;; Próxima medida de distância
(:durative-action Range
 :parameters (?pass - PassNumber ?sat - SatCode ?gs - GSCode ?rgroup -
              RangeGroupNumber ?rnum - RangeNumber)
 :duration (= ?duration (RangeTime ?gs ?rgroup))
 :condition (and (at start (GSAvailable ?gs))
                 (over all (PassTrack ?pass ?sat ?gs))
                 (over all (RangeAvailable ?pass ?sat ?gs))
                 (over all (FirstRangeOK ?pass ?rgroup))
                )
 :effect (and (at end (RangeOk ?pass ?rgroup ?rnum))
              (at start (not (GSAvailable ?gs)))
              (at end (GSAvailable ?gs))
             )
)

;; Esperar por zona de silêncio
(:durative-action WaitSilentZone
 :parameters (?pass - PassNumber ?sat - SatCode ?gs - GSCode ?ZS - ZSNumber )
 :duration (= ?duration (ZoneSilentTime ?pass ?ZS))
 :condition (and (at start (GSAvailable ?gs))
                 (at start (SatSilentConeAvailable ?pass ?sat ?gs ?ZS))
                )
 :effect (and (at end (SilentZone ?pass ?sat ?gs ?ZS))
              (at start (not (GSAvailable ?gs)))
              (at end (GSAvailable ?gs))
             )
)

;; Medida de Velocidade
(:durative-action RangeRate
 :parameters (?pass - PassNumber ?sat - SatCode ?gs - GSCode ?RRgroup -
              RRateGroupNumber)

```

```

:duration (= ?duration 1.0)
:condition (and (at start (GSAvailable ?gs))
                (at start (RRateAvailable ?pass ?sat ?gs ))
                (over all (PassTrack ?pass ?sat ?gs))
              )
:effect (and (at end (RRateOK ?pass ?sat ?gs ?RRGroup))
             (at end (not(RRateAvailable ?pass ?sat ?gs)))
             (at start (not (GSAvailable ?gs)))
             (at end (GSAvailable ?gs))
           )
)

;; Enviar primeiro telecomando da passagem
(:durative-action SendFirstTC
 :parameters (?pass - PassNumber ?sat - SatCode ?tc - TCCode ?gs - GSCode)
 :duration (= ?duration (TCTime ?tc))
 :condition (and (at start (GSAvailable ?gs))
                 (at start (FirstTc ?pass ?sat ?tc))
                 (at start (LOSAOSTimeOk ?pass ?sat ?gs))
                 (over all (PassTrack ?pass ?sat ?gs))
                 (over all (not (TCNotAvailable ?pass ?sat))))
             )
 :effect (and (at end (FirstTcOK ?pass ?sat ?tc))
             (at end (TcSent ?pass ?sat))
             (at start (not (GSAvailable ?gs)))
             (at end (GSAvailable ?gs))
           )
)

;; Enviar último telecomando da passagem
(:durative-action SendLastTC
 :parameters (?pass - PassNumber ?sat - SatCode ?tc - TCCode ?gs - GSCode)
 :duration (= ?duration (TCTime ?tc))
 :condition (and (at start (GSAvailable ?gs))
                 (at start (LastTc ?pass ?sat ?tc))
                 (over all (PassTrack ?pass ?sat ?gs))
                 (over all (TcSent ?pass ?sat))
             )
 :effect (and (at end (LastTcOK ?pass ?sat ?tc))
             (at start (not (GSAvailable ?gs)))
             (at end (GSAvailable ?gs))
             (at end (not (TCSent ?pass ?sat)))
             (at end (not (PassTrack ?pass ?sat ?gs))))
           )
)

;; Enviar telecomando temporizado em solo
(:durative-action SendTimeTaggedTC
 :parameters (?pass - PassNumber ?sat - SatCode ?tc - TCCode ?gs - GSCode)
 :duration (= ?duration (TCTime ?tc))
 :condition (and (at start (GSAvailable ?gs))
                 (at start (TimeTaggedTC ?pass ?sat ?tc))
                 (over all (PassTrack ?pass ?sat ?gs))
             )
 :effect (and (at end (TimeTaggedTCOK ?pass ?sat ?tc))

```

```

        (at start (not (GSAvailable ?gs)))
        (at end (GSAvailable ?gs))
    )
)

;; Enviar telecomando não temporizado
(:durative-action SendTC
 :parameters (?pass - PassNumber ?sat - SatCode ?tc - TCCode ?gs - GSCode)
 :duration (= ?duration (TCTime ?tc))
 :condition (and (at start (GSAvailable ?gs))
                (over all (PassTrack ?pass ?sat ?gs))
                (over all (TCSent ?pass ?sat))
               )
 :effect (and (at start (not (GSAvailable ?gs)))
              (at end (GSAvailable ?gs))
              (at end (TCSendOK ?pass ?sat ?tc))
             )
)

;; Enviar telecomando condição
(:durative-action SendTCCond
 :parameters (?pass - PassNumber ?sat - SatCode ?tcCond - TCCode ?gs -
             GSCode)
 :duration (= ?duration (TCTime ?tcCond))
 :condition (and (at start (GSAvailable ?gs))
                (over all (PassTrack ?pass ?sat ?gs))
                (over all (TCSent ?pass ?sat))
               )
 :effect (and (at start (not (GSAvailable ?gs)))
              (at end (GSAvailable ?gs))
              (at end (TCCondSendOK ?pass ?sat ?tcCond))
             )
)

;; Enviar telecomando condicionado
(:durative-action WASendTCCond
 :parameters (?pass - PassNumber ?sat - SatCode ?tcCond - TCCode ?tc - TCCode
             ?gs - GSCode)
 :duration (= ?duration 0.0001)
 :condition (and (at start (GSAvailable ?gs))
                (over all (TCCondSendOK ?pass ?sat ?tcCond))
                (over all (TCCondSendOK ?pass ?sat ?tc))
                (over all (PassTrack ?pass ?sat ?gs))
               )
 :effect (and (at start (not (GSAvailable ?gs)))
              (at end (GSAvailable ?gs))
              (at end (TCCondicionadoSendOK ?pass ?sat ?tcCond ?tc))
             )
)

) ;; Fim do Domínio de Controle de Satélites

```


APÊNDICE B - EXEMPLO DE ARQUIVO DE INSTÂNCIA DO PROBLEMA

A seguir, apresenta-se o arquivo de Instância do Problema codificado na linguagem PDDL 2.2, gerado pela arquitetura MAPSat (em especial pelo agente Gerador de Instâncias do Problema), para a passagem identificada pelo número de órbita 60482-5 do satélite brasileiro SCD1 sobre a estação terrena de Cuiabá.

```
;;Arquivo de Instância do Problema para a passagem: SCD1CBAProblemPass60482-5
;;
(define (problem SCD1CBAProblemPass60482-5)
  (:domain POVMEDZSRRTC)
  ;;
  ;;Definição dos objetos
  (:objects
    PASS60482-5 - PassNumber
    SCD1 - SatCode
    CBA - GSCode
    RGroup03 - RangeGroupNumber
    Range02 - RangeNumber
    Range00 - RangeNumber
    ZS00 - ZSNumber
    RGroup04 - RRateGroupNumber
    TC137 - TCCode
    TC118 - TCCode
    TC138 - TCCode
    TT140 - TCCode
    TT150 - TCCode
    TC158 - TCCode
    TC160 - TCCode
    TC159 - TCCode
    TC138 - TCCode
  )
  ;;
  (:init
    ;;Definição das funções
    (= (LOSAOSTime PASS60482-5 SCD1 CBA) 120.0)
    (= (PassTime PASS60482-5) 450.0)
    (= (RangeTime CBA RGroup03) 90.0)
    (= (TcTime TC137) 2.0)
    (= (TcTime TC118) 2.0)
    (= (TcTime TC138) 2.0)
    (= (TcTime TT140) 3.0)
    (= (TcTime TT150) 2.0)
    (= (TcTime TC158) 2.0)
    (= (TcTime TC160) 2.0)
```

```

(= (TcTime TC159) 2.0)
(= (TcTime TC138) 2.0)
;;
;;Definição do estado inicial do problema de planejamento
(GSAvailable CBA)
(PassTrack PASS60482-5 SCD1 CBA)
(RangeAvailable PASS60482-5 SCD1 CBA)
(ReducedTrack PASS60482-5 SCD1 CBA)
;;
;;Definição dos eventos exógenos
(at 126.0 (FirstRange PASS60482-5 SCD1 CBA))
(at 216.0 (not (FirstRange PASS60482-5 SCD1 CBA)))
;;
(at 121.0 (FirstTc PASS60482-5 SCD1 TC137))
(at 123.0 (not (FirstTc PASS60482-5 SCD1 TC137)))
;;
(at 0.0 (LOSAOSPeriod PASS60482-5 SCD1 CBA))
(at 120.0 (not (LOSAOSPeriod PASS60482-5 SCD1 CBA)))
;;
(at 390.0 (LastTC PASS60482-5 SCD1 TC138))
(at 392.0 (not (LastTC PASS60482-5 SCD1 TC138)))
;;
(at 124.0 (RRateAvailable PASS60482-5 SCD1 CBA))
(at 125.0 (not (RRateAvailable PASS60482-5 SCD1 CBA)))
)
;;
;;Definição dos objetivos a serem atingidos no plano
;;
(:goal (and
  (LOSAOSTimeOk PASS60482-5 SCD1 CBA)
  (LastTCOK PASS60482-5 SCD1 TC138)
  (RRateOK PASS60482-5 SCD1 CBA RRGroup04)
  (RangeOk PASS60482-5 RGroup03 Range02)
  (TCCondicionadoSendOK PASS60482-5 SCD1 TC160 TC158)
  (TCCondicionadoSendOK PASS60482-5 SCD1 TC159 TC160)
  (TCCondicionadoSendOK PASS60482-5 SCD1 TC138 TC159)
)
)
)

```

ANEXO A – PUBLICAÇÕES DESTE TRABALHO DE PESQUISA

A seguir, apresentam-se as publicações, realizadas em periódicos e anais de congressos nacionais e internacionais, que foram geradas no decorrer do desenvolvimento deste trabalho de pesquisa.

CARNIELLO, A.; CARNIELLO, A.; CARDOSO, L. S.; FERREIRA, M. G. V.; SILVA, J. D. S. Multi-agent planning and execution approach for controlling multiple satellites. **Journal of Aerospace Computing, Information, and Communication (JACIC)**, v. 5, Jul 2008.

CARNIELLO, A.; AQUINO, A. C.; FERREIRA, M. G. V.; SILVA, J. D. S.; CARDOSO, L. S. Multi-agent architecture for automating satellite control operations planning and execution. **Lecture Notes in Computer Science (LNCS)**, v. 4251, p. 268-275, 2006.

CARNIELLO, A.; CARNIELLO, A.; FERREIRA, M. G. V.; SILVA, J. D. S.; CARDOSO, L. S. Dynamical assignment of priorities to operations for planning multiple satellite control. In: International Conference on Space Operations (SpaceOps 2008), 10., 12-16 May 2008, Heidelberg, Germany. **Proceedings of the International Conference on Space Operations (SpaceOps 2008)**. Heidelberg: AIAA, 2008.

CARNIELLO, A.; SILVA, J. D. S.; FERREIRA, M. G. V. Uma arquitetura multi-agente de planejamento de controle de satélites. In: Encontro Regional de Matemática Aplicada e Computacional (ERMAC), 5., 26-28 ago 2008, INPE, São José dos Campos, SP. **Anais do Encontro Regional de Matemática Aplicada e Computacional (ERMAC 2008)**. São José dos Campos: INPE, 2008.

CARNIELLO, A. ; AQUINO, A. C. ; FERREIRA, M. G. V. ; SILVA, J. D. S. ; CARDOSO, L. S. Software agents for multiple satellite automated planning and control. In: International Communications Satellite Systems Conference (ICSSC), 24., 2006, San Diego, California. **Proceedings of the 24th AIAA International Communications Satellite Systems Conference**. Reston, Virginia : American Institute of Aeronautics and Astronautics, Inc., 2006.

CARNIELLO, A. ; AQUINO, A. C. ; FERREIRA, M. G. V. ; SILVA, J. D. S. ; CARDOSO, L. S. A Multi-agent based ground-operations automation architecture. In:

International Conference on Space Operations (SpaceOps), 9., 2006, Roma. **Proceedings of the 9th International Conference on Space Operations**. Reston, Virginia : American Institute of Aeronautics and Astronautics, Inc., 2006.

CARNIELLO, A. ; AQUINO, A. C. ; FERREIRA, M. G. V. ; SILVA, J. D. S. ; CARDOSO, L. S. . Applying software agents to the satellite control engineering. In: Iberian Latin American Congress on Computational Methods in Engineering (CILAMCE), 27., 2006, Belém - PA. **Proceedings of the XXVII Iberian Latin American Congress on Computational Methods in Engineering**, 2006.

CARNIELLO, A. ; CARNIELLO, A. ; FERREIRA, M. G. V. ; SILVA, J. D. S. ; CARDOSO, L. S. . Multi-agent ground-operations automation architecture. In: International Astronautical Congress (IAC), 56., 2005, Fukuoka. **Proceedings of the 56th International Astronautical Congress**, 2005.

CARDOSO, L.; FERREIRA, M.; ORLANDO, V.; CARNIELLO, A. Aplicação da tecnologia de agentes de planejamento em operações de satélites. In: Simpósio Brasileiro de Automação Inteligente (SBAI), 7., 20 – 23 set 2005, São Luís, Maranhão, Brasil. **Anais do Simpósio Brasileiro de Automação Inteligente (SBAI 2005)**. São Luis, 2005. (ISBN 85-85048-55-7).

CARNIELLO, A. ; FERREIRA, M. G. V. ; SILVA, J. D. S. . Uma arquitetura de automação de operações solo multi-agente. In: Workshop dos Cursos de Computação Aplicada do INPE (WORCAP), 5., 2005, São José dos Campos - SP. **Anais do V Workshop dos Cursos de Computação Aplicada do INPE**, 2005.

CARNIELLO, A. ; CARNIELLO, A. ; FERREIRA, M. G. V. ; SILVA, J. D. S. ; CARDOSO, L. S. . Uma arquitetura de automação de operações solo multi-agente. In: Encontro Nacional de Inteligência Artificial (V ENIA) - XXV Congresso da Sociedade Brasileira de Computação (SBC), 5., 2005, São Leopoldo, RS. **Anais do V Encontro Nacional de Inteligência Artificial (V ENIA)**, 2005. p. 1146-1149.

CARNIELLO, A. ; FERREIRA, M. G. V. ; SILVA, J. D. S. Uma arquitetura de automação de operações solo multi-agente. In: Workshop dos Cursos de Computação Aplicada do INPE (WORCAP), 4., 2004, São José dos Campos, SP. **Anais do IV Workshop dos Cursos de Computação Aplicada do INPE**, 2004.

PUBLICAÇÕES TÉCNICO-CIENTÍFICAS EDITADAS PELO INPE

Teses e Dissertações (TDI)

Teses e Dissertações apresentadas nos Cursos de Pós-Graduação do INPE.

Manuais Técnicos (MAN)

São publicações de caráter técnico que incluem normas, procedimentos, instruções e orientações.

Notas Técnico-Científicas (NTC)

Incluem resultados preliminares de pesquisa, descrição de equipamentos, descrição e ou documentação de programa de computador, descrição de sistemas e experimentos, apresentação de testes, dados, atlas, e documentação de projetos de engenharia.

Relatórios de Pesquisa (RPQ)

Reportam resultados ou progressos de pesquisas tanto de natureza técnica quanto científica, cujo nível seja compatível com o de uma publicação em periódico nacional ou internacional.

Propostas e Relatórios de Projetos (PRP)

São propostas de projetos técnico-científicos e relatórios de acompanhamento de projetos, atividades e convênios.

Publicações Didáticas (PUD)

Incluem apostilas, notas de aula e manuais didáticos.

Publicações Seriadas

São os seriados técnico-científicos: boletins, periódicos, anuários e anais de eventos (simpósios e congressos). Constam destas publicações o Internacional Standard Serial Number (ISSN), que é um código único e definitivo para identificação de títulos de seriados.

Programas de Computador (PDC)

São a seqüência de instruções ou códigos, expressos em uma linguagem de programação compilada ou interpretada, a ser executada por um computador para alcançar um determinado objetivo. São aceitos tanto programas fonte quanto executáveis.

Pré-publicações (PRE)

Todos os artigos publicados em periódicos, anais e como capítulos de livros.