UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS QUIXADÁ
CURSO DE GRADUAÇÃO EM ENGENHARIA DE SOFTWARE

RODRIGO BARBOSA DE ALMEIDA

O USO DE RELEVÂNCIA NA MUDANÇA DO ESTADO INICIAL EM PROBLEMAS DE PLANEJAMENTO SEM SOLUÇÃO.

QUIXADÁ
2018

RODRIGO BARBOSA DE ALMEIDA

O USO DE RELEVÂNCIA NA MUDANÇA DO ESTADO INICIAL EM PROBLEMAS DE PLANEJAMENTO SEM SOLUÇÃO.

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Software do Campus Quixadá da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia de Software.

Orientadora: Profa. Dra. Maria Viviane de Menezes

QUIXADÁ

2018

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

A45u Almeida, Rodrigo Barbosa de.
O uso de relevância na mudança do estado inicial em problemas de planejamento sem solução / Rodrigo Barbosa de Almeida. – 2018.
41 f. : il.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá, Curso de Engenharia de Software, Quixadá, 2018.
Orientação: Profa. Dra. Maria Viviane de Menezes.

1. Inteligência artificial. 2. Planejamento automatizado. 3. Revisão de crenças. 4. Relevância. I. Título.
CDD 005.1

RODRIGO BARBOSA DE ALMEIDA

O USO DE RELEVÂNCIA NA MUDANÇA DO ESTADO INICIAL EM PROBLEMAS DE PLANEJAMENTO SEM SOLUÇÃO.

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Software do Campus Quixadá da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia de Software.

Aprovada em: __/__/__

BANCA EXAMINADORA

____

Profa. Dra. Maria Viviane de Menezes (Orientadora)
Universidade Federal do Ceará (UFC)

____

Prof. Dr. Paulo de Tarso Guerra Oliveira
Universidade Federal do Ceará (UFC)

____

Prof. Dr. Wladimir Araujo Tavares
Universidade Federal do Ceará (UFC)

À minha família, por sua capacidade de acreditar em mim e investir em mim. Mãe, seu cuidado e dedicação foi que deram, em alguns momentos, a esperança para seguir e a confiança necessaria para acreditar que não estou sozinho nessa caminhada.

AGRADECIMENTOS

A Prof.ª Dr.ª Maria Viviane de Menezes por me orientar neste trabalho de conclusão de curso.

Aos amigos de curso Samuel Alves, Davi Cedraz, Ernades Junior, Gleydson da Silva e Léo Jaimerson por toda ajuda e companheirismo durante estes quatro anos.

Aos amigos Dayse Campelo, André Davys e Raabe Campelo por se fazerem presentes durante esse périodo importante da minha vida.

Aos meus pais e avós, que sempre me fizeram entender que o futuro é feito a partir da constante dedicação no presente!

Ao meu tio Jéferson Henrique por toda ajuda que me foi servida, possibilitando a continuação dos meus estudos.

Agradeço a Prof.ª Ana Cristina Batista Sá e ao Prof. João Paulo Nogueira, como também a todos os professores por me proporcionar o conhecimento não apenas racional, mas a manifestação do caráter e afetividade da educação no processo de formação profissional e por tanto que se dedicaram a mim.

“Qualquer coisa que pensa logicamente pode ser enganada por outra coisa que pensa e faz logicamente da mesma forma”

(Douglas Adams)

RESUMO

Planejamento Automatizado estuda o processo de escolha de ações, para que um agente inteligente possa atingir seus objetivos. A dinâmica do ambiente em que o agente irá atuar é chamada de domínio de planejamento. Para cada domínio, é possível definir vários problemas de planejamento, cada um especificando o estado inicial do agente e uma propriedade que deve ser alcançada, chamada de meta de planejamento.Um plano é um conjunto de ações que quando executadas a partir do estado inicial,leva o agente a atingir a meta.

Quando não existe um plano, dizemos que o problema de planejamento é sem solução. Existem três razões pelas quais um problema de planejamento não tem uma solução: (1) má especificação de estado inicial; (2) um objetivo que não pode ser alcançado ou; (3) má especificação do conjunto de ações. Neste trabalho propomos a inclusão de informações de relevância das variáveis e seus valores para atingir a meta de planejamento no processo de mudança do estado inicial para problemas de planejamento sem solução.

Objetivo deste trabalho é propor uma métrica para o processo de mudança do estado inicial para problemas de planejamento sem solução, particularmente uma métrica capaz de escolher o novo estado inicial usando informações sobre relevância que as variáveis possuem para o alcance da meta de planejamento.

Palavras-chave: Inteligência Artificial. Planejamento Automatizado. Mudança do Estado Inicial. Revisão Crenças. Relevância.

ABSTRACT

Automated Planning for the process of choosing actions, so that it fits into your goals. The dynamics of the environment in which the agent will respond is called the domain of planning. To each domain, it is possible to multiple problems of planning, each to specifying the state-of-initial agent and the method that must execute the call of mettment of planning. leads the agent to achieve goal.

When there is no plan, we say that the planning problem is unsolvable. There are three reasons why a planning problem does not have a solution: (1) poor initial state specification; (2) an objective that can not be achieved or; (3) bad specification of the set of actions. In this work we propose the inclusion of information of relevance of the variables and their values to reach the planning goal in the process of change from initial state to problems of planning without solution.

The purpose of this paper is to propose a metric for the initial state change process for unsolvable planning problems, particularly a metric capable of choosing the new initial state using information on the relevance of the variables to the achievement of the planning goal.

Keywords: Artificial intelligence. Automated Planning. Change of the Initial State. Review Beliefs. Relevance.

LISTA DE FIGURAS

SUMÁRIO

1 INTRODUÇÃO

Planejar é determinar um conjunto de ações para atingir um dado objetivo. De fato, esta é uma das características essenciais do comportamento inteligente e uma atividade que executamos constantemente quando nos deparamos com a necessidade de obter uma sequência de ações para que nossos objetivos sejam alcançados.

Neste contexto *Planejamento Automatizado* (GHALLAB *et al.*, 2004) estuda o processo de escolha dessas ações de forma computacional para que um agente inteligente possa atingir suas metas. Um *planejador* propõe-se a ser um resolvedor geral de problemas das mais diversas naturezas: problemas de logística, problema de navegação de robôs, problemas de jogos, problema de operação de satélites, dentre outros.

1.1 Domínios, Problemas e Planos

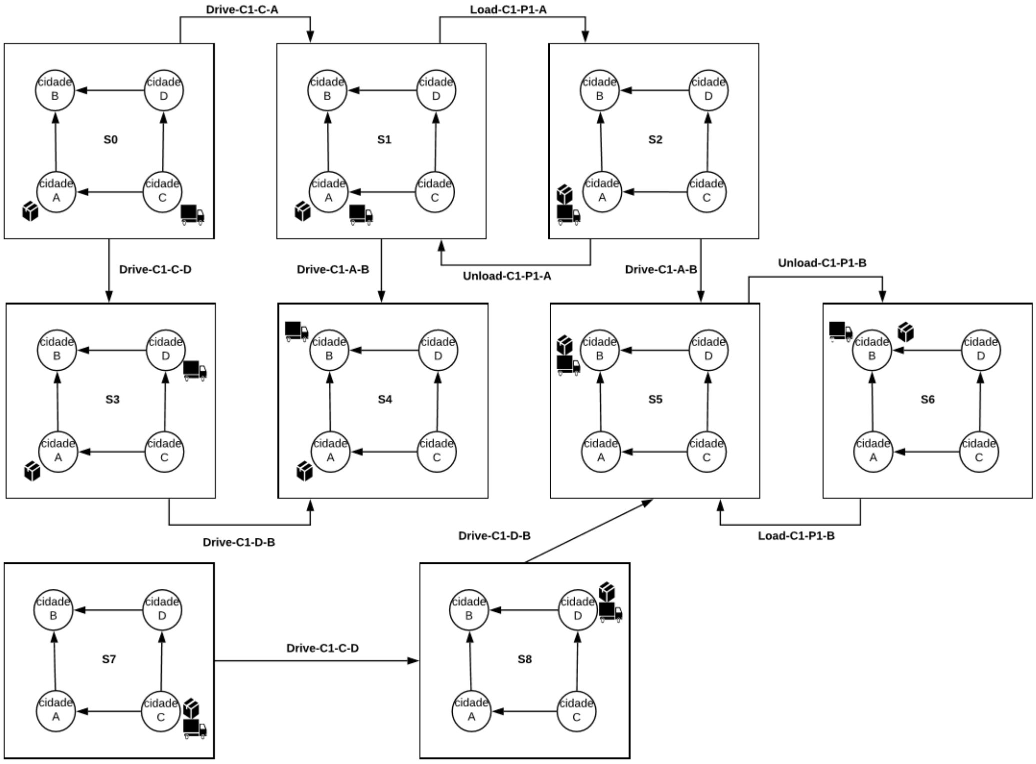
A dinâmica do ambiente em que o agente irá atuar é denominado *domínio de planejamento*. Dado um conjunto de *átomos proposicionais*, descrevendo as propriedades do ambiente e um conjunto de *ações* que o agente pode executar, um domínio de planejamento pode ser representado como um sistema de transição de estados em que os estados são rotulados por proposições que são verdadeiras e as transições são rotuladas pelas ações responsáveis pela evolução do ambiente de um estado para outro.

A Figura 1 ilustra um sistema de transição de estados para um domínio de logística. Neste domínio, temos: quatro cidades, um caminhão e um pacote. As proposições que representam as características deste ambiente são usadas para indicar a localização do pacote e do caminhão em cada instante de tempo. As ações possíveis são: *drive*, que permite que um caminhão vá de uma cidade a outra; *load*, que permite que um pacote seja carregado dentro do caminhão; e *unload*, que permite que um pacote seja descarregado do caminhão.

No entanto, a representação de domínios de planejamento por meio de grafos (*representação explicita*) não é escalável para representar grandes quantidades de estados. Assim, em planejamento costuma-se utilizar a representação implícita do domínio, dada por uma linguagem de ações. Na Figura 2 é apresentado a descrição do domínio de logística na linguagem PDDL (*Planning Domain Description Language*) (MCDERMOTT *et al.*, 1998).

Em PDDL, primeiramente, são definidos os predicados na lista `(:predicates)`. No domínio de logística apresentado temos os seguintes predicados: {`package p1`, `truck c1`,

Figura 1 – Sistema de transição de estados para um domínio de planejamento de logística.



Fonte: Elaborado pelo o autor.

`city a`, `city b`, `city c`, `city d`}, {`at-p1-a`, `at-p1-b`, `at-p1-c`, `at-p1-d`, `at-c1-a`, `at-c1-b`, `at-c1-c`, `at-c1-d`, `in-p1-c1`}.

As ações são definidas em termos de seus parâmetros, suas pré-condições e seus efeitos. Os parâmetros definem que objetos estarão envolvidos na ação. As pré-condições indicam o que deve ser verdadeiro antes que a ação possa ser aplicada, efeitos descrevem como o estado atual do mundo se altera se a ação for aplicada. Nesta versão do domínio de logística, as ações são *load*, *unload* e *drive*. Por exemplo, a ação `unload(p1 c1 a)`, utilizada para descarregar o objeto `p1` do caminhão `c1` na cidade `a`, possui como pré-condições (lista `(:precondition)`) que: `p1` seja um pacote `(package p1)`; que `c1` seja um caminhão `(truck c1)`; que `a` seja uma cidade; que o caminhão esteja na cidade `(at c1 a)` e que o pacote esteja dento do caminhão `(in p1 a)`. O uso do `and` na lista é para indicar que todas as pré-condições devem ser atendidas para que a ação possa ser aplicada em um estado. Os efeitos desta ação são: que o pacote `p1` não esteja mais dentro do caminhão `c1` (`(not (in p1 c1))` e; que o pacote `p1` esteja na cidade `a` (`(at p1 a)`).

Para cada domínio, é possível definir diversos *problemas de planejamento*, cada um especificando a situação inicial que o agente se encontra, denominado *estado inicial* e; uma propriedade que deve ser alcançada, chamada *meta de planejamento*.

Figura 2 – Domínio de Logística em PDDL.

```
(define (domain logistics)
    (:predicates (package p1) (truck c1) (city a) (city b) (city c)
                 (city d) (at p1 a) (at p1 b) (at p1 c) (at p1 d)
                 (at c1 a) (at c1 b) (at c1 c) (at c1 d) (in p1 c1))
    (:action load
        :parameters (p1 c1 a)
        :precondition (and (package p1) (truck c1) (city a)
                           (at c1 a) (at p1 a))
        :effect (and (not (at p1 a)) (in p1 c1)))

    (:action load
        :parameters (p1 c1 b)
        :precondition (and (package p1) (truck c1) (city b)
                           (at c1 b) (at p1 b))
        :effect (and (not (at p1 b)) (in p1 c1)))

    (:action unload
        :parameters (p1 c1 a)
        :precondition (and (package p1) (truck c1) (city a)
                           (at c1 a) (in p1 c1))
        :effect (and (not (in p1 c1)) (at p1 a)))

    (:action unload
        :parameters (p1 c1 b)
        :precondition (and (package p1) (truck c1) (city b)
                           (at c1 b) (in p1 c1))
        :effect (and (not (in p1 c1)) (at p1 b)))

    (:action drive
        :parameters (c1 a b)
        :precondition (and (truck c1) (city a) (city b)
                           (at c1 a))
        :effect (and (not (at c1 a)) (at c1 b)))
    ...
```

Fonte: Elaborado pelo o autor.

Exemplo 1 *(Problema de planejamento Solucionável) Seja um problema de planejamento na Figura 1, em que s_0 é o estado inicial e a meta é que o pacote chegue na cidade B, a saber estado s_6.*

A solução para um problema de planejamento é chamada de *plano*. Um plano consiste em um sequência de ações que o leva do estado inicial a um estado que satisfaz a meta. No Exemplo 1, um plano é a sequência de ações: `drive-c1-c-a`, `load-c1-p1-a`, `drive-c1-a-b` e `unload-c1-p1-b`.

No entanto nem sempre existe um plano solução para um problema de planejamento. Nestes casos, dizemos que o *problema de planejamento não possui solução*. O Exemplo 2 apresenta um problema de planejamento sem solução.

Exemplo 2 *(Problema sem Solução) Seja um problema de planejamento no domínio de logística da Figura 1, em que s_3 é o estado inicial e a meta de planejamento seja que o pacote $p1$ seja entregue na cidade b.*

O problema apresentado no Exemplo 2 é um problema de planejamento sem solução, uma vez que não existe um caminho que leve o agente do estado inicial s_3 a algum estado que satisfaz a meta, a saber: estado s_4.

Existem três motivos que fazem um problema de planejamento não possuir uma solução: (1) má especificação do estado inicial; (2) uma meta que não pode ser alcançada ou; (3) má especificação do conjunto de ações (MENEZES, 2014).

1.2 Trabalhos Correlatos

Para cada um dos motivos que tornam um problema de planejamento sem solução, existem abordagens que propõem possíveis modificações no problema a fim de torná-lo solucionável. As abordagens da literatura que realizam a mudança do estado inicial são (GÖBELBECKER *et al.*, 2010), (MENEZES *et al.*, 2012), (MENEZES, 2014) e (HERZIG *et al.*, 2014).

Em (GÖBELBECKER *et al.*, 2010) utiliza-se o formalismo $\mathtt{SAS}^+$ (HELMERT, 2004) com variáveis multi-valoradas para descrição do domínio de planejamento no lugar de utilizar variáveis proposicionais. No problema de logística, por exemplo, podemos indicar o conjunto dos possíveis valores que a variável de localização do pacote $p1$, `loc-p1`, pode assumir como o conjunto {a, b, c, d, c1}. Dizemos então que este {a, b, c, d, c1} é o domínio da variável `loc-p1`. Quando um problema de planejamento não possui solução, estruturas de dados denominadas *Grafos Causais* e *Grafos de Transição de Domínio* são utilizadas para obtenção do *domínio relevante* de cada variável multi-valorada. O domínio relevante de uma variável é o subconjunto do domínio desta variável que é *relevante* para o alcance da meta de planejamento. Por exemplo, para a variável `loc-p1` podemos ter {a, b, c1 } como domínio relevante. A partir das informações de relevância, são introduzidas no problema de planejamento um conjunto de ações fictícias, isto é, ações artificiais que modificam apenas os valores das variáveis para que alcancem os valores relevantes. Nesta abordagem, os autores propõe também uma classificação das possíveis mudanças do estado inicial em *boa*, *ótima* e *perfeita*.

A abordagem de (MENEZES *et al.*, 2012) propõe um método de obtenção do domínio relevante para um domínio de planejamento proposicional, como o descrito na Figura 2. Este método representa pré-condições e efeitos das ações do domínio de planejamento como fórmulas proposicionais e realiza a manipulação de tais fórmulas utilizando Diagramas de Decisão Binária (BRYANT, 1995). Também, assim como feito por (GÖBELBECKER *et al.*, 2010), a partir das informações de relevância, são introduzidas no problema de planejamento um conjunto de ações fictícias cuja tarefa é modificar o valor de uma variável proposicional para o valor relevante. Nesta abordagem, como as variáveis são proposicionais, as ações ficticias mudam os valores das variáveis de verdadeiro para falso ou de falso para verdadeiro.

O trabalho de (MENEZES, 2014) caracteriza a *mudança do estado inicial* como um problema de *revisão de crenças* (GÄRDENFORS, 2003). Nesta abordagem, procura-se modificar minimamente o estado inicial para que o problema de planejamento torne-se solucionável. A métrica de ordenação das possíveis mudanças utilizada é a *distância de Hamming* (FORBUS, 1988), uma métrica de diferença proposicional entre dois estados. Assim, todas as possíveis mudanças são ordenadas segundo essa métrica de forma que o novo estado inicial torne o problema solucionável e seja o mais semelhante possível ao estado inicial original do problema. A mudança do estado inicial foi testada em problemas sem solução dos seguintes domínios: logística, robô de marte e domínio das chaves.

A abordagem de (HERZIG *et al.*, 2014) caracteriza a mudança do estado inicial de um problema de planejamento sem solução como um problema de revisão de crenças na lógica dinâmica de atribuições proposicionais DL-PA (HERZIG *et al.*, 2014).

1.3 Motivação

A caracterização da mudança do estado inicial como um problema de revisão de crenças permite que se tenha estados com diferença minimal em relação ao estado inicial, isto é, garante-se que o estado inicial original do problema será alterado minimamente para que o problema de planejamento torne-se solucionável.

De um lado, se a ordenação baseada na métrica de distância proposicional (MENEZES, 2014) seleciona mais de um estado como opção para ser o novo estado inicial do problema, os possíveis novos estados iniciais são apresentados sem levar em consideração nenhum tipo de característica ou propriedade que os diferencie, são todos estados com *distância proposicional minimal em relação ao estado inicial original*. Por outro lado, a informação de *relevância* de uma variável para o alcance da meta têm sido utilizada na elaboração de heurísticas e na construção de planejadores estado-da-arte (HELMERT, 2006).

Como o cálculo do domínio relevante pode ser efetuado por meio de operações nos *Diagramas de Decisão Binária* que representam as ações de um domínio de planejamento (MENEZES *et al.*, 2012), a motivação deste trabalho é a utilização deste cálculo para elaborar uma nova métrica baseada em relevância no *arcabouço de mudanças de problemas sem solução* (MENEZES, 2014) para a modificação do estado inicial.

No problema de planejamento sem solução apresentado no Exemplo 2, temos s_3 como estado inicial e a meta é entregar o pacote 1 na cidade b. Efetuar a mudança do estado inicial neste problema, baseado numa abordagem de revisão de crenças com métrica de distância proposicional, nos levaria a escolher estados que diferem do estado s_3 por pequenas mudanças de proposições, como por exemplo, mudar caminhão de cidade, ou o pacote de lugar, mudanças estas que podem ser representadas nos estados s_0, s_1 e s_8 da Figura 1. Assim qualquer um destes estados poderia ser escolhido como um novo estado inicial. Porém, as ações da Figura 2 indicam, que o pacote só pode ser carregado e descarregado, nas cidades a e b. Assim, notamos que é *relevante para que a meta seja alcançada* que a carga esteja em uma destas cidades. Com esta informação podemos afirmar que os estados s_0 e s_1, são mais relevantes que o estado s_8.

1.4 Objetivos

1.4.1 Objetivo Geral

Definição e implementação de uma métrica baseada em relevância para a mudança do estado inicial em problemas de planejamento sem solução.

1.4.2 Objetivos Específicos

- Estudo da implementação do algoritmo de obtenção do domínio relevante utilizando diagramas de decisão binária.
- Inclusão do cálculo do domínio relevante utilizando diagramas de decisão binária no arcabouço de mudanças de problema de planejamento sem solução.

- Definição de uma métrica baseada em relevância para a mudança do estado inicial em problemas de planejamento sem solução.
- Implementação da métrica baseada em relevância para a mudança do estado inicial em problemas de planejamento sem solução.

1.4.3 Organização

Este trabalho esta estruturado da seguinte forma. A seção 2, apresenta a fundamentação teórica, que inclui, a Busca por um Plano, Distância de Hamming, Grafos Causais e Grafos de Transição de Domínio e Diagramas de Decisão Binária. A seção 3 apresenta a nova métrica baseada em relevância proposta por este trabalho, a qual inclui a informação do domínio relevante no processo de mudança do estado inicial. A seção 4 apresenta as conclusões e trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Planejamento Automatizado

Planejamento Automatizado é o processo de escolha de um conjunto de ações a serem executadas por um agente inteligente em uma determinada sequência a fim de satisfazer um conjunto de objetivos (RUSSELL; NORVING, 2013).

Formalmente, um domínio de planejamento pode ser representado de uma forma *explícita* como um *sistema de transição de estados*, como mostrado na Definição 1.

Definição 1 ***(Domínio de Planejamento)*** *Dado um conjunto de proposições $\mathbb{P}$ e um conjunto de ações $\mathbb{A}$, um domínio de planejamento é definido por uma tupla de três elementos $\mathbb{D} = \langle S, L, T \rangle$ onde (GHALLAB et al., 2004):*

- *$S \neq \emptyset$ é um conjunto finito de estados possíveis do ambiente;*
- *L: $S \rightarrow 2^{\mathbb{P}}$ é a função de rotulação dos estados, qual descreve que proposições atômicas são verdadeiras em um estado e;*
- *$T : S \times \mathbb{A} \times S$ é uma função de transição de estados.*

Nesta representação, assume-se a suposição do mundo fechado, *closed world anssunption* (RUSSELL; NORVING, 2013), em que são representados nos estados apenas variáveis verdadeiras. As variáveis que não são representadas em um estado, são tidas como falsas.

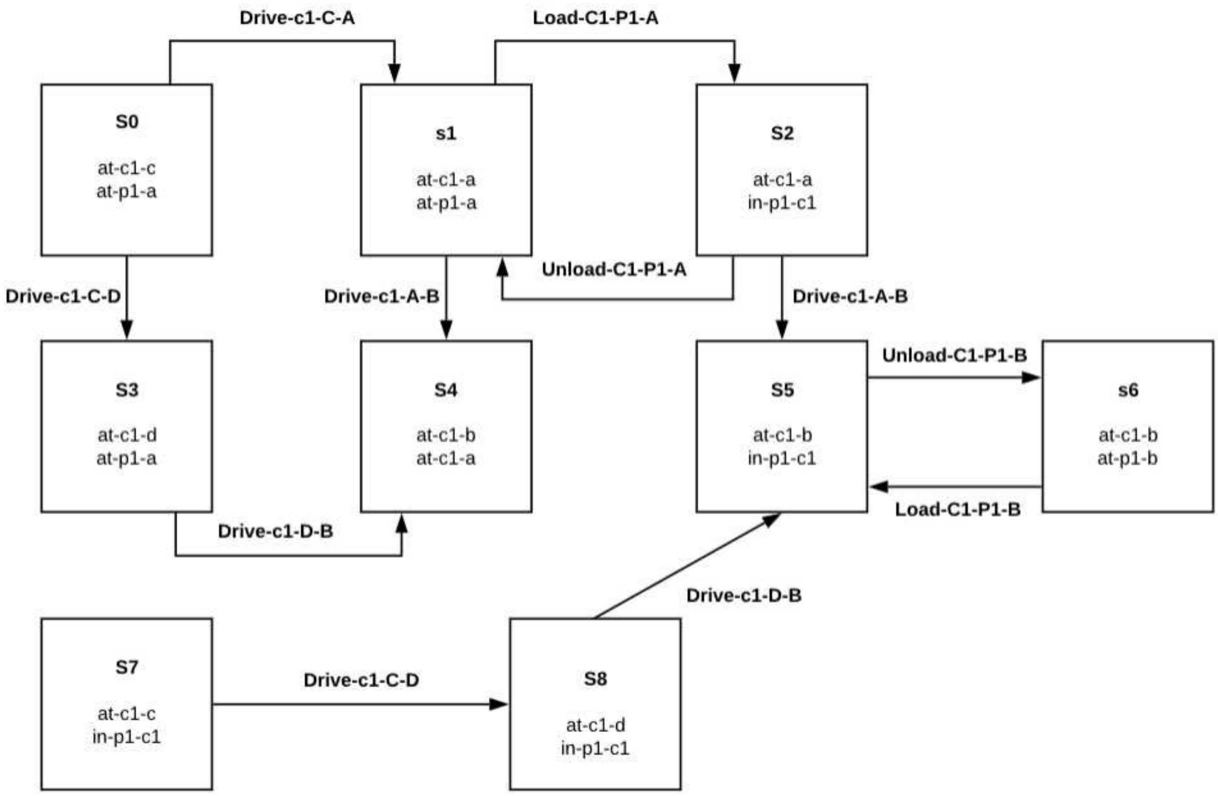
A Figura 3 apresenta o sistema de transição de estados para o domínio de Logística, onde os estados estão rotulados utilizando os atomos proposicionais definidos em PDDL.

A partir de um domínio de planejamento conseguimos definir diversos *problemas de planejamento*, quando informamos um *estado inicial* e uma meta φ. Formalmente, um *problema de planejamento* é definido como apresentado na Definição 2

Definição 2 ***(Problema de Planejamento)*** *Dado um conjunto finito de proposições atômicas $\mathbb{P}$ e um conjunto finto de ações $\mathbb{A}$, um problema de planejamento é definido por uma tupla $\Pi = \langle \mathbb{D}, s_0, \varphi \rangle$ onde (GHALLAB et al., 2004):*

- *$\mathbb{D}$ é um domínio de planejamento*
- *s_0 é o estado inicial do problema, e;*
- *φ é uma formula proposicional sobre $\mathbb{P}$ que especifica a meta de planejamento.*

Figura 3 – Sistema de Transição de Estados para o Domínio de Logística, onde os estados estão rotulados utilizando átomos proposicionais

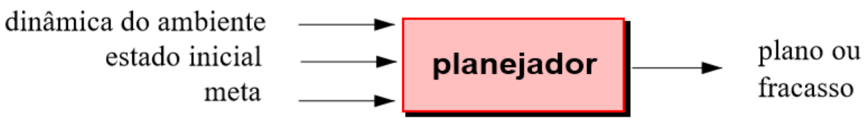


Fonte: Elaborado pelo o autor.

A solução para um problema de planejamento é denominado *plano*, uma sequencia de ações que quando executadas levam o agente do estado inicial pra um estado que satisfaz a meta.

Um *planejador* como o mostrado na Figura 4 é um algoritmo que recebe como entrada um problema de planejamento e retorna como saída um *plano*, se for possível encontrá-lo, ou *falha*, caso não seja possível encontrar um plano.

Figura 4 – Planejador



.

Fonte: (PEREIRA; BARROS, 2007)

2.2 Em Busca de uma Solução

Para encontrar um plano solução para um problema de planejamento, um planejador pode realizar uma *busca progressiva* ou uma *busca regressiva*. A busca progressiva é feita a partir do estado inicial do problema, gerando em cada etapa, estados sucessores até que a meta seja alcançada. A *busca regressiva* é feita a partir de todos os estados que satisfazem a meta de planejamento, gerando em cada passo predecessores, até que o estado inicial seja encontrado.

A busca progressiva tem como operação básica o cálculo da progressão de um estado por uma ação, denotado como, $prog^a(s)$. Esta operação permite que a partir de um estado s com a aplicação de uma ação a seja obtido um estado sucessor s'

A *progressão* de um estado s por uma dada ação a gera um único estado sucessor. Para que a progressão possa ser feita, é necessário primeiro verificamos se a ação a pode ser aplicada no estado s. Dizemos que a ação a pode ser aplicada em um estado s, quando as pré-condições de a são satisfeitas no estado s, ou seja, $precond(a) \subseteq L(s)$ (MENEZES, 2014).

Se uma ação a é aplicada no estado s, um estado sucessor é gerado por meio da fórmula $prog^a(s)$, como mostrado na Equação 2.2

$$prog^a(s) = s \cup add(a) \setminus del(a)$$

No domínio de logística apresentado na Figura 1 temos o estado s_3 definido por $\{$`at-c1-d, at-p1-a`$\}$, podemos construir os estados sucessores de s_3, utilizando o cálculo da progressão apresentado na Equação 2.2 e a linguagem de ações definida para o domínio de logística, apresentado na Figura 2.

Se observado, somente a ação `drive(c1-d-b)` pode ser aplicada em s_3 devido o fato de ser a única ação a possui suas pré-condições satisfeitas na rotulação do estado. Dessa forma podemos construir o estado sucessor de s_3 pela ação `drive(c1-d-b)`, denotado como $prog^{\text{drive(c1-d-b)}}(s_3)$, seguindo a Equação 2.2, da seguinte forma:

$$\begin{aligned} prog^{\text{drive(c1-d-b)}}(s_3) &= \{\texttt{at-c1-d, at-p1-a}\} \\ &= \{\texttt{at-c1-d, at-p1-a, at-c1-b}\} \\ &= \{\texttt{at-c1-b, at-p1-a}\} \end{aligned}$$

Assim o calculo de progressão a partir do estado s_3 com a ação `drive(c1-d-b)` nos leva até o estado s_4 que é rotulado por $\{$`at-c1-b, at-p1-a`$\}$.

A busca regressiva tem como operação básica o cálculo da regressão de um estado por uma ação, denotado por, $regr^a(s)$. Na regressão observamos que ações a que são relevantes ao estado s, isto é, que ações que contribuem para as propriedades satisfeitas em s, ou seja, os efeitos positivos de a, $add(a)$, devem esta contidos em na rotulação de s e os efeitos negativos de a, $del(a)$ não devem existir na rotulação de s (MENEZES, 2014).

A regressão de um estado meta s por uma ação a gera um conjunto de de estados predecessores $regr^a(s)$ que é definido como a seguir.

$$\mathrm{regr}^a(s) = (s \cup \mathrm{precond}(a)) \setminus \mathrm{add}(a) \tag{2.1}$$

Porém o calculo da regressão só pode acontecer para uma ação a caso $add(a) \subseteq s$ e $del(a) \cap x = \varnothing$, caso contrario não existe estado predecessor.

No domínio de logística apresentado na Figura 1 temos o estado s_3 definido por $\{$`at-c1-d, at-p1-a`$\}$, podemos construir os estados predecessores de s_3, utilizando o cálculo da regressão apresentado na Equação 2.1 e a linguagem de ações definida para o domínio de logística, apresentado na Figura 2.

Assim, temos que $regr^{\text{drive(c1-c-d)}}(s_3)$ é definido da seguinte forma:

$$\begin{aligned} regr^{\text{drive(c1-c-d)}}(s_3) &= \{\texttt{at-c1-d, at-p1-a}\} \\ &= \{\texttt{at-p1-a}\} \\ &= \{\texttt{at-c1-c, at-p1-a}\} \end{aligned}$$

De inicio, $\mathrm{regr}^{drive(c1-c-d)(s_3)}$ é igual a $\{$`at-c1-d, at-p1-a`$\}$, em seguida excluímos os efeitos positivos de `drive-c1-c-d`, a saber, `at-c1-d`, temos agora que $regr^{\text{drive(c1-c-d)}}(s_3)$ igual a $\{$`at-p1-a`$\}$, depois inserimos todos os efeitos negativos em $regr^{\text{drive(c1-d-b)}}(s_3)$, a saber, `at-c1-c`, assim, $regr^{\text{drive(c1-d-b)}}(s_3)$ é igual a $\{$`at-c1-c, at-p1-a`$\}$.

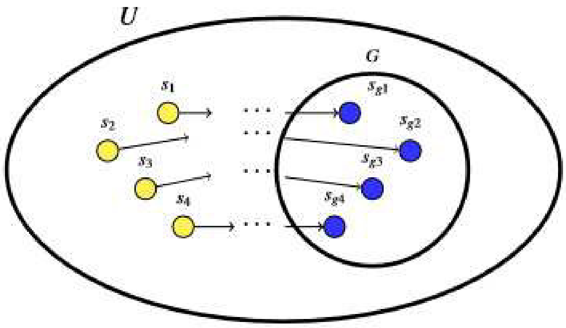
Logo o calculo da regressão a partir do estado s_3 nos leva até o estado s_0 que também é rotulado por $\{$`at-c1-c, at-p1-a`$\}$.

A busca regressiva quando executada até que nenhum novo estado seja encontrado, gera o conjunto de todos os estados que alcançam algum estado que satisfaz a meta, que chamaremos aqui de conjunto U, como representado na Figura 5.

2.3 Mudança do Estado Inicial com Revisão de Crenças

A mudança do estado inicial de um problema de planejamento pode ser caracterizada como um problema de revisão de crenças (GÄRDENFORS, 2003). A revisão de uma base

Figura 5 – Conjunto U dos estados que alcançam algum estado meta para um problema de planejamento $\Pi = \langle \mathbb{D}, s_0, \varphi \rangle$, sendo G o conjunto dos estados que satisfazem a formula φ

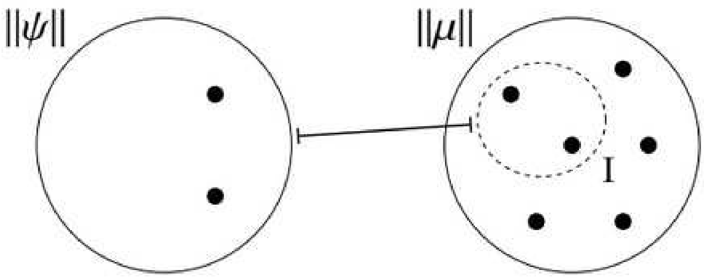


.

Fonte: (MENEZES, 2014)

conhecimento ψ por um novo fato μ, denotado por $\psi * \mu$, resulta em uma nova base de conhecimento que satisfaz este novo fato. O operador de revisão, escolhe os modelos da base $||\psi||$, que possuem uma *diferença minimal,* em relação aos modelos do fato $||\mu||$, para compor esta nova base de conhecimento. Ou seja, a revisão $\psi * \mu$ é o conjunto de modelos do novo fato μ mais próximos aos modelos da base. A Figura 6 ilustra o processo de revisão de crença de uma base de conhecimento ψ por um novo fato μ.

Figura 6 – Revisão de crenças de uma base de conhecimento ψ por um novo fato μ. $\psi * \mu$



.

Fonte: (MENEZES, 2014)

Na Figura 6 temos do lado esquerdo um conjunto de estados que satisfazem o a base de conhecimento ψ, do lado direito encontramos um conjunto com todos os estados que satisfazem o novo fato μ. Durante o processo de revisão de crenças, o operador escolhe um subconjunto dos estados que satisfazem μ, mais próximos do modelo da base de conhecimento ψ, que na imagem é representado pelo conjunto I.

Para que essa mudança possa ser feita de forma minimal é necessário definir métricas para que se possa selecionar quais são os modelos da base que são mais próximos aos modelos do fato. Uma métrica comumente utilizada é chamada *distância de Hamming* (FORBUS, 1988).

A *distância de Hamming* entre dois estados s_1 e s_2 é calculada a partir da distância proposicional entre os estados, conforme mostrado na Definição 3.

Definição 3 *(Distância de Hamming) Sejam s_1 e s_2 estados do domínio de planejamento, a distância de Hamming entre s_1 e s_2 é dada por: $h(s_1, s_2) = \mid L(s_1) - L(s_2) \mid$, em que $L(s_1)$ e $L(s_2)$ são, respectivamente, os rótulos dos estados s_1 e s_2. (FORBUS, 1988)*

O trabalho de (MENEZES, 2014) caracterizou a mudança do estado inicial como um problema de revisão de crenças de s_0 pelo o conjunto de estados que satisfazem a meta de planejamento, denotado por U. Isto é, desejamos encontrar dos modelos (estados) de U aqueles que são os mais próximos possíveis da meta. A garantia de que todos os estados em U finalmente alcançam a meta, faz com que os novos estados selecionados pela revisão torne o problema de planejamento solucionável.

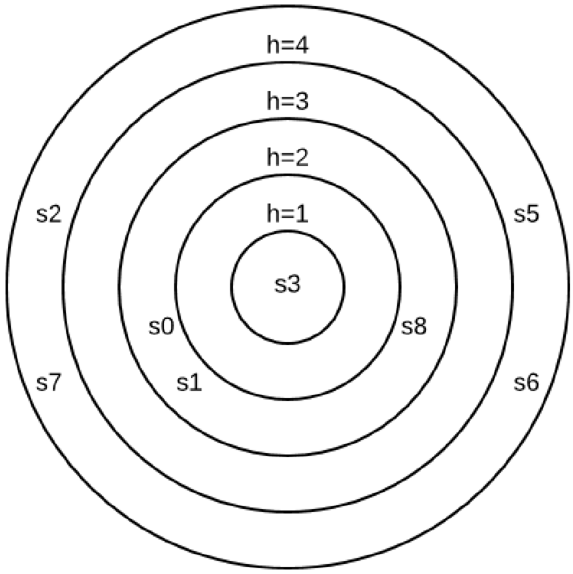
Considere, o problema de planejamento sem solução do Exemplo 2. Podemos efetuar o processo de *revisão de crenças* entre o *estado inicial* s_3 e o conjunto $U = \{s_0, s_1, s_2, s_5, s_6, s_7, s_8\}$. Para saber qual a distância entre cada estado de U e o estado s_3, utilizamos a métrica da *distância de Hamming*. Os rótulos dos estados pertencentes a U são: $L(s_0) = \{$`at-c1-c, at-p1-a`$\}$; $L(s_1) = \{$`at-c1-a, at-p1-a`$\}$; $L(s_2) = \{$`in-p1-c1,at-c1-a`$\}$;$L(s_5) = \{$`in-p1-c1,at-c1-b`$\}$; $L(s_6) = \{$`at-c1-b,at-p1-b`$\}$; $L(s_7) = \{$`at-c1-c,in-p1-c1`$\}$ e $L(s_8) = \{$`at-c1-d, in-p1-c1`$\}$. Por sua vez o estado inicial s_3 é rotulado com $L(s_3) = \{$`at-c1-d, at-p1-a`$\}$.

A Figura 7 mostra um gráfico em esfera, dividido em camadas, que apresenta a distância de todos os estados presente no conjunto U em relação ao estado inicial s_3. Como pode ser observado, os estados s_1, s_2 e s_8 possuem uma distância proposicional de apenas 2 elementos, fazendo assim eles serem os estados mais próximos do estado s_3, então, estes estados são o resultado da revisão de s_3 por U. Tanto o estado s_1 como o estado s_3 são igualmente apontados para serem opções de novo estado inicial do problema.

2.4 Mudança do Estado Inicial por Meio de Grafos Causais

O trabalho de (GÖBELBECKER *et al.*, 2010) utilizada *Grafos causais* e *Grafos de Transição de Domínio* para obter o *domínio relevante* das variáveis do problema a fim de auxiliar a mudança do estado inicial. Esta foi a primeira abordagem na literatura de planejamento a sugerir possíveis modificações em problema de planejamento sem solução.

Figura 7 – Distância proposicional dos estados no conjunto U em relação ao estado inicial s_3



Fonte: Elaborado pelo o autor.

Neste trabalho, utiliza-se a linguagem $\texttt{SAS}^+$ para descrever os domínios de planejamento. Nesta notação, as variáveis do domínio são *multi-valoradas*, o que difere um pouco das variáveis proposicionais que utilizamos na descrição dos domínios (Figura PDDL). No domínio de Logística, por exemplo, a variável multi-valorada `loc-c1`, que indica a localização do caminhão $c1$, pode assumir um dos seguintes valores: a, b, c, d. O conjunto $\{a,b,c,d\}$ é denotado como domínio de `loc-c1`, isto é, o conjunto de valores que a variável pode assumir. Quando estamos tratando de variáveis proposicionais, o domínio de cada variável é o conjunto de valores $\{verdadeiro, falso\}$

Assim, o *domínio relevante* de uma variável, seja ela proposicional ou multi-valorada, é o conjunto de valores desta variável que são *relevantes para o alcance da meta*. Suponha que a meta seja entregar o pacote $p1$ na cidade b e o caminhão esteja na cidade a, para a variável `loc-c1` é *relevante* o valor b (caminhão na cidade b), mas não temos tanta certeza que a variável assumir os valores c ou d é *relevante* para que a meta seja alcançada. Formalmente, o domínio relevante de uma variável é definido como na Definição 4.

Definição 4 *(Domínio Relevante) Seja x uma variável do problema de planejamento e seja dom(x) o conjunto de valores que x pode assumir, isto é, o domínio de x. O domínio relevante de*

Figura 8 – Relação de dependência entre as variáveis x e y

Fonte: Elaborado pelo o autor.

x, denotado por $dom_{rel}(x)$, é o conjunto de valores u pertencentes ao dom(x), tal que, o fato de x assumir o valor u, contribui para o alcance da meta. Ou seja, um dos passos necessário para o alcance da meta, é que a variável x possua o valor u. (GÖBELBECKER et al., 2010).

A descoberta do domínio relevante de cada variável do problema é feita de forma recursiva a partir das variáveis presentes na meta, utilizando estruturas de dados auxiliares de representação do problema de planejamento: o *grafo causal* e os *grafos de transição de domínio*.

Grafos causais (HELMERT, 2004) são estruturas de dados utilizadas para relacionar as variáveis de um domínio de planejamento e as suas dependências. Um grafo causal está associado a um problema de planejamento, onde os seus vértices são todas as variáveis do problema e suas arestas estabelecem as relações de dependências entre as variáveis.

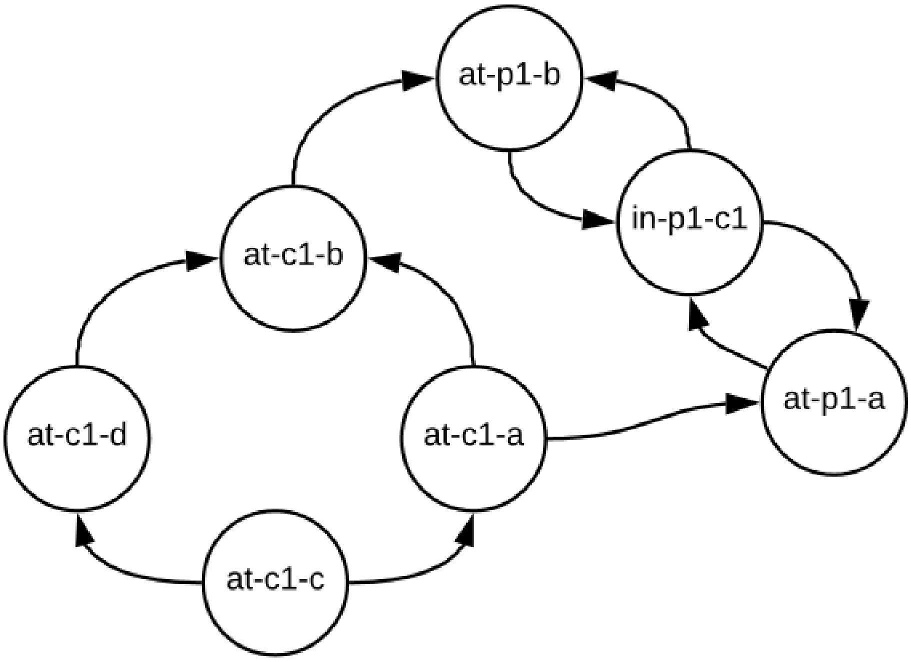
Definição 5 *(Relação de dependência entre x e y) Seja x e y duas variáveis proposicionais pertencentes a $\mathbb{P}$, dizemos que x depende de y se existe uma ação a $\in \mathbb{A}$, tal que y pertence as pré-condições de a e x pertencente aos efeitos de a. (HELMERT, 2004).*

Se uma variável *x* depende de uma variável *y*, haverá uma aresta partindo de *y* para *x*. A Figura 8, ilustra a relação de dependência entre as variáveis x e y, em um grafo causal.

Seguindo a Definição 5, podemos construir o grafo causal do domínio de logística apresentado na Figura 1, representando todas as dependências existente entre as variáveis presentes no domínio, que é apresentado na Figura 9.

Grafos de Transição de Domínio (HELMERT, 2004) representa as maneiras que uma variável *x* do domínio de planejamento pode alterar o seu valor e as condições para que isso ocorra. No grafo de transição de domínio, os vértices serão os valores do domínio da variável. E haverá uma aresta de um vértice *v*, para um vértice *u*, se houver alguma ação que altere o valor da variável *x* de *v* para *u*. Esta aresta possuirá um rótulo significando a condição para que esta

Figura 9 – Grafo Causal para o domínio de logística da Figura 1



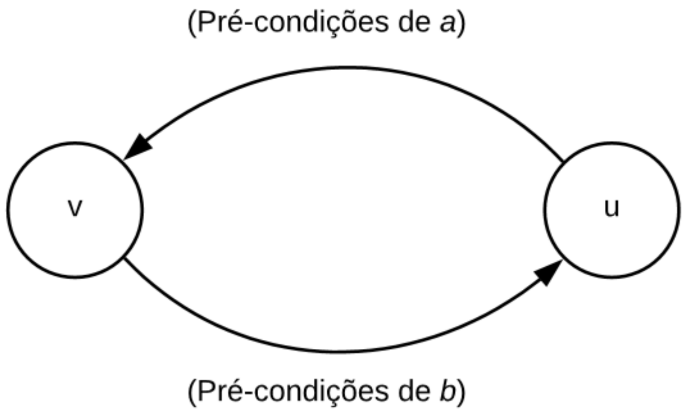
.

Fonte: Elaborado pelo o autor.

mudança de valor ocorra. Este rótulo provém das pre-condições da ação que causa a mudança de valor da variável *x* de *v* para *u*

A Figura 10 apresenta o grafo de transição de domínio de uma variável *x*, onde o domínio de *x* é o conjunto formado pelos valores *v* e *u*. A aresta (v, u) é rotulada pelas pré-condições da ação b, e a aresta (u, v) é rotulada pelas pré-condições da ação a.

Figura 10 – Grafo de Transição de domínio da variável *x*

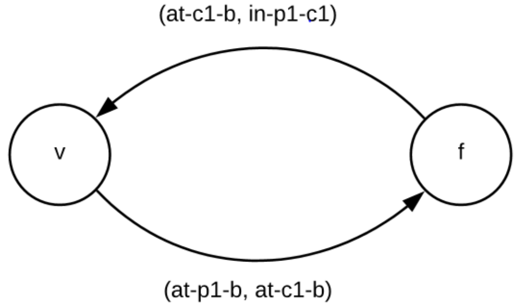


.

Fonte: Elaborado pelo o autor.

Observando na Figura 11 o grafo de transição de domínio da variável `at-p1-b` do dominío de lógistica apresentado na Figura 1. Como `at-p1-b` é uma variável proposicional,

Figura 11 – Grafo de Transição de domínio para a variável `at-p1-b`



Fonte: Elaborado pelo o autor.

vértices serão representados por $\top$, significando que a variável `at-p1-b` é verdadeida, e $\bot$ significando que a variável `at-p1-b` é falsa. A aresta ($\top$, $\bot$) indica que ocorreu uma mudança de valor verdade da variável `at-p1-b` quando a ação `load-p1-c1-b` 2 é executada. A aresta ($\bot$, $\top$) indica que a mudança do valor verdade da variável `at-p1-b` quando a ação `unload-p1-c1-b` 2 é executada.

O domínio relevante de cada variável do domínio de planejamento é obtido de formar recursiva por meio do grafo causal, iniciando a partir das variáveis presentes na meta. Observando o problema de planejamento apresentado no Exemplo 2, cuja a meta é entregar o pacote 1 na cidade b, queremos alcançar algum estado em que o valor de `at-p1-b` seja igual $\top$.

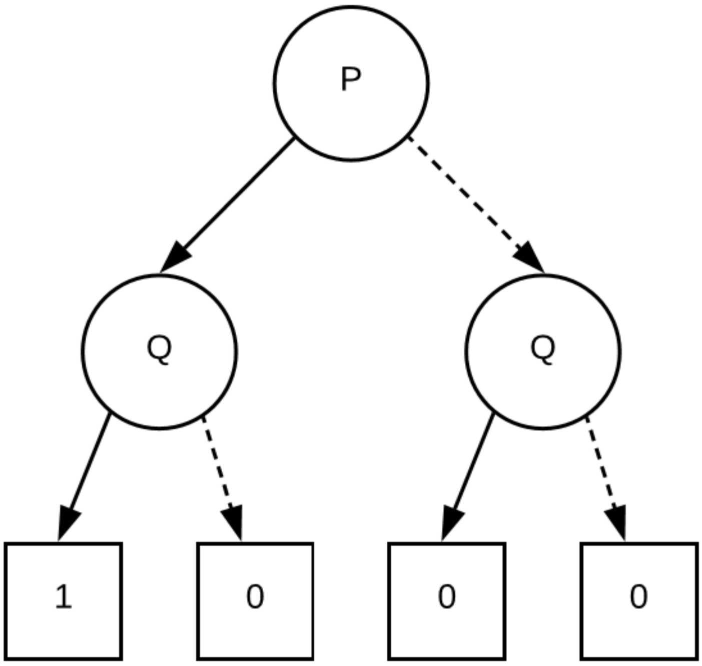
De inicio já notamos que `at-p1-b` possuir o valor $\top$ é relevante para o alcance da meta, isto porque, `at-p1-b` faz parte da meta, assim é inserido o valor $\top$ ao $dom_{\text{rel}}(\texttt{at-p1-b})$. Em seguida, observando o grafo causal do domínio de logística, descobrimos um conjunto de proposições da qual a `at-p1-b` depende, que aqui será denotado por $Dependentes^{at-p1-b}$ que a saber é formado por, $\{\texttt{at-c1-b}, \texttt{in-p1-c1}\}$. No grafo de transição de domínio de `at-p1-b`, construímos um conjunto formado por todas as rotulações de arestas que chegam ao vértice $\top$, que chamaremos de Rotulos, que é formado por, $\{\texttt{at-c1-b} = \top, \texttt{in-p1-c1} = \top\}$.

Em seguida, é verificado se alguma variável x pertencente a Dependentes$^{at-p1-b}$ também pertence a Rotulos. Caso isso seja verdade, iniciamos o processo recursivo com a variável x, e processo se repete até que não seja mais possível incorporar nenhum novo valor ao domínio relevante de qualquer variável.

2.5 Obtenção do Domínio Relevante com BDDS

O trabalho de (MENEZES *et al.*, 2012) propõe uma abordagem de obtenção do domínio relevante por meio da utilização de *Diagramas de Decisão Binaria*, BDDs (*Binary*

Figura 12 – Árvore de decisão Binária para a formula $p \wedge q$



Fonte: Elaborado pelo o autor.

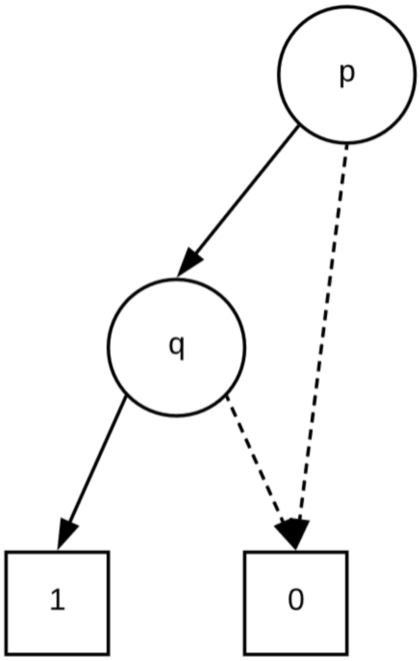
Decision Diagrams) (BRYANT, 1986). Diagramas de Decisão Binária são estruturas utilizadas para representação eficiente de formulas proposicionais como também a realização de operações entre elas.

2.5.1 Diagramas de Decisão Binaria

Podemos representar as possíveis valorações para uma fórmula proposicional por meio de tabelas verdade ou por meio de *Árvores de Decisão Binária*. Numa árvore de decisão binária as folhas são rotuladas pelas constantes 0 (falso) e 1 (verdadeiro). Os outros nós da árvore são rotulados por variáveis proposicionais. Todo nó p_i, que não seja um nó folha, possui 2 nós filhos. Um filho da esquerda ($p_i = 1$), que é indicado por uma linha contínua, e um filho direito ($p_i = 0$) indicado por uma uma linha pontilhada. Na Figura 12 é apresentado a árvore de decisão binária para a formula lógica $p \wedge q$.

Na árvore decisão binária apresentada na Figura 12 temos no seu primeiro nível um nó representando a variável proposicional p, como também temos os possíveis valores que p pode assumir, ($p = 1$) representado pela linha continua, como ($p = 0$) representado pela linha pontilhada. Se seguirmos o ramo da árvore em que $p = 1$, chegaremos ao vértice que representa a variável proposicional q no nível seguinte da arvore, em que q também podemos escolher seguir os caminhos 1 ou 0, se seguimos pela linha continua ($q = 1$), chegamos a um nó folha com o

Figura 13 – Diagrama de decisão Binária para a formula $p \wedge q$



Fonte: Elaborado pelo o autor.

valor 1, que indica que quando p e q possuem valor 1, toda a formula, $p \wedge q$ possui valor 1.

Como podemos notar, quando escolhemos seguir pelo caminho em que p possui o valor 0, não importa o caminho que escolhemos em q, sempre chegaremos a um nó folha que possui o valor 0, por esse motivo a árvore pode ser compactada. Chamamos de Diagrama de decisão a árvore de decisão quando ocorre:

- Eliminação de nós redundantes e.
- Compactação dos subgrafos isomorfos.

Essa estrutura de dados é capaz de representar espaços de estados de até 10^{20} estados (BRYANT, 1995). Podemos então representar o diagrama de decisão binaria para a fórmula $p \wedge q$, de acordo como na Figura 13.

2.5.2 Representação de Estados

Como apresentado na seção anterior, a utilização de BDDs é uma forma de representar as possíveis valorações de uma formula lógica. Assim, BDDs podem ser utilizados para representar estados de um domínio de planejamento, a partir das formulas lógicas que codificam estes estados. Neste trabalho, denotaremos ξ(s) como a formula lógica que codifica o estado s. Este processo também se aplica as transições existente entre estados, a representação da formula lógica de uma transição t = (s,s'), aqui denotaremos como, ξ(t).

Dado um conjunto de átomos proposicionais $\mathbb{P}$, um estado $s \in S$, de um domínio de planejamento 1, pode ser representado como uma conjunção dos átomos proposicionais que rotulam este estado e da negação dos átomos que não estão em L(s). A codificação proposicional de um estado é denotada pela formula:

$$\xi(s) = \bigwedge_{p \in \mathrm{L(s)}} p \wedge \bigwedge_{q \in \mathbb{P} \setminus \mathrm{L(s)}} \neg q$$

Baseado na Equação acima podemos representar o estado s_0 da Figura 1, como a formula: $\xi(s_0)$ = `at-c1-c` $\wedge$ `at-p1-a` $\wedge$ $\neg$`at-c1-a` $\wedge\neg$`at-c1-b` $\wedge\neg$`at-c1-d` $\wedge\neg$`at-p1-b` $\wedge$ $\neg$`at-p1-c` $\wedge\neg$`tat-p1-d` $\wedge\neg$`in-p1-c1`

2.5.3 Representação de Ações

As ações do domínio de planejamento são representadas por meio de fórmulas proposicionais. Considerando que as variáveis v_i são as variáveis presentes nas pré-condições de um ação e as variáveis v_i' são as variáveis presentes nos efeitos das ações, cada ação do domínio de planejamento pode ser representada pela fórmula proposicional apresentada na Definição 6.

Definição 6 *(Representação de Ações por meio de Fórmulas Proposicionais) Uma ação $a \in A$ pode ser representada na forma proposicional ψ_a (MENEZES et al., 2012).*

$$\psi_a = \bigwedge_{v \in precond(a)} p \wedge \bigwedge_{v' \in add(a)} p' \wedge \bigwedge_{v' \in del(a)} \neg p'$$

A partir da Definição 6 podemos representar as ações presentes na representação implícita do domínio de logística mostrado na Figura 2. Como exemplo, a ação, `load-p1-c1-a`, pode ser representada como, $\psi_{\text{load-p1-c1-a}} = \texttt{at-c1-a} \wedge \texttt{at-p1-a} \wedge \neg(\texttt{at-p1-a})' \wedge \texttt{in-p1-c1}'$.

2.5.4 Obtenção do Domínio Relevante

Nesta seção, utilizamos a representação das ações como formulas proposicionais para obtenção do domínio relevante das variáveis do domínio de planejamento. Esse método tem vantagens, pois não há necessidade de construir o grafo causal e os grafos de transição de domínio. Essa informação pode ser estruturada diretamente da representação das ações como formulas proposicionais. Para isso utilizaremos o operador existencial de lógica QBF

(*Quantifield Booleana Forumula*) (RINTANEN *et al.*, 2008). Este operador quantifica os valores verdades de uma variável x em uma formula proposicional ϕ como a seguir:

$$\exists x.\phi = \phi\,[v\backslash\bot] \vee \phi\,[v\backslash\top]$$

Onde $\phi\,[x\backslash\top]$ é substituição de x por $\top$ na formula ϕ, e $\phi\,[x\backslash\bot]$ é substituição de x por $\bot$ na formula ϕ. Intuitivamente, essa quantificação elimina a variável x da formula ϕ, chamaremos neste trabalho essa operação de *relaxamento* de x em ϕ.

2.5.4.1 Descobrindo o conjunto dependentes(x) a partir da representação proposicional das ações

Sendo $x \in \mathbb{P}$ uma variável do domínio de planejamento; $\mathbb{B} \subseteq \mathbb{A}$ o conjunto de ações que x aparece em seus efeitos. O conjunto *dependentes(x)* é obtido por:

$$dependentes(x) = \bigcup_{\alpha_i \in \mathbb{B}} dependentes(x)_{\alpha_i}$$

Em que $dependentes(x)_{\alpha_i}$ é o conjunto de variáveis que x depende, de acordo com a ação α_i. Esse conjunto é obtido pelo processo de relaxamento da variável x e sua respectiva variável x' na representação proposicional ψ_{α_i} da ação α_i, tal como a seguir.

$$\delta = \exists x.\exists x'.\psi_{\alpha_i}$$

$dependentes(x)_{\alpha_i}$ é constituído por toda variável presente em δ.

Considere como exemplo a variável `at-p1-b` e a ação `unload-p1-c1-b`, para obter o conjunto *dependentes(at-p1-b)*$_{\texttt{unload-c1-p1-b}}$ formado pelas as variável que `at-p1-b` depende na ação `unload-p1-c1-b`, temos que efetuar o processo de relaxamento de `at-p1-b` e `at-p1-b`' em `unload-c1-p1-b`.

Sabemos que a representação proposicional $\psi_{\texttt{unload-p1-c1-b}}$ da ação `unload-p1-c1-b` é dada por:

$$\psi_{\texttt{unload-p1-c1-b}} = \texttt{at-c1-b} \wedge \texttt{in-p1-c1} \wedge \neg(\texttt{in-p1-c1}') \wedge \texttt{at-p1-b}'$$

Assim, efetuamos o processo de relaxamento das variáveis `at-p1-b` e `at-p1-b`' em $\psi_{\texttt{unload-p1-c1-b}}$, talque:

$$\begin{aligned}\delta = (\texttt{at-c1-b} \wedge \texttt{in-p1-c1} \wedge \neg\texttt{in-p1-c1}' \wedge \bot) \vee (\texttt{at-c1-b} \wedge \texttt{in-p1-c1} \wedge \neg\texttt{in-p1-c1}' \wedge \top)\\ = (\bot) \vee (\texttt{at-c1-b} \wedge \texttt{in-p1-c1} \wedge \neg\texttt{in-p1-c1}' \wedge \top)\\ = \texttt{at-c1-b} \wedge \texttt{in-p1-c1} \wedge \neg\texttt{in-p1-c1}' \wedge \top)\\ = \texttt{at-c1-b} \wedge \texttt{in-p1-c1} \wedge \neg\texttt{in-p1-c1}')\end{aligned}$$

A partir de δ temos dependentes(at-p1-b)$_{\texttt{unload-p1-c1-b}}$ = $\{$`at-c1-b`, `in-p1-c1`, $\neg$`in-p1-c1`$\}$

2.5.4.2 *Obtenção de Ações Relevantes*

Dizemos que uma ação α é relevante para uma variável x, quando ele contribui para que x seja alcançada, ou seja, α possui x em seus efeitos. Para descobrir se uma ação α em sua representação proposicional ψ_α é relevantes para uma variável x, primeiro precisamos descobrir se ψ_α não nega nossa variável x. Para isso efetuamos uma conjunção entre $\psi_\alpha \wedge x'$, onde formula proposicional resultante chamaremos de σ, caso σ seja igual $\bot$, já podemos afirmar que ψ_α não contribui para x. Caso contrário, ainda precisamos efetuar uma segunda verificação, se σ é igual $\top$ já sabemos que ψ_α não nega a variável x, mas ainda precisamos saber se ψ_α se relaciona de alguma forma com a nossa variável. Para isso verificamos se σ é diferente ou não ψ_α, se são diferentes, essa diferença é dado somente pela variável x' que é o segundo lado de nossa conjunção, isso significa que a ação ψ_α não possuía em seus efeitos a variável x, ou seja, ela não possui nenhuma relação com a nossa variável. Caso σ seja igual a ψ_α significa que a variável x não teve nenhum impacto na construção de σ, assim podemos afirmar que ψ_α já possuía a variável x' em seus efeitos, por tanto ψ_α é relevante para x.

Para explicação vamos utilizar um conjunto $\mathbb{B}$ formado pelas ações `load-p1-c1-b`, `unload-p1-c1-b` e `drive-c1-a-c` e desejamos saber quais dessa ações é relevante para a variável `at-p1-b`, como explicado acima primeiro efetuamos uma conjução da representação proposicional das ações com a variável `at-p1-b`'. Assim, teremos as seguintes formulas proposicionais:

$$\begin{aligned}\sigma_{load-p1-c1-b} = \texttt{at-c1-b} \wedge \texttt{at-p1-b} \wedge \neg\texttt{at-p1-b}' \wedge \texttt{in-p1-c1}' \wedge \texttt{at-p1-b}'\\ \sigma_{unload-p1-c1-b} = \texttt{at-c1-b} \wedge \texttt{in-p1-c1} \wedge \neg\texttt{in-p1-c1}' \wedge \texttt{at-p1-b}' \wedge \texttt{at-p1-b}'\\ = \texttt{at-c1-b} \wedge \texttt{in-p1-c1} \wedge \neg\texttt{in-p1-c1}' \wedge \texttt{at-p1-b}'\\ \sigma_{drive-a-c} = \texttt{at-c1-a} \wedge \neg\texttt{at-c1-a}' \wedge \texttt{at-c1-c}' \wedge \texttt{at-p1-b}\end{aligned}$$

Em $\sigma_{load-p1-c1-b}$ temos os elementos ¬ `at-p1-b`' e `at-p1-b`' que nos gera uma contradição, logo, $\sigma_{load-p1-c1-b} = \perp$ e assim $\sigma_{load-p1-c1-b}$ não é relevante para a variável `at-p1-b`. Em $\sigma_{drive-a-c}$ não encontramos uma contradição, porém apos a conjunção $\sigma_{drive-a-c}$ se tornou diferente de $\psi_{drive-a-c}$ e como explicado, isso mostra que `at-p1-b`' não existia nos efeitos de $\psi_{drive-a-c}$, logo a ação `drive-a-c` não se relaciona com a variável `at-p1-b`. Mas em $\sigma_{unload-p1-c1-b}$ temos a situação onde $\sigma_{unload-p1-c1-b}$ é igual a $\top$ e também é igual a $\psi_{unload-p1-c1-b}$, significando assim que `unload-p1-c1-b` já possuía a variável `at-p1-b`' em seus efeitos, e por esse motivo dizemos que `unload-p1-c1-b` é relevante para a variável `at-p1-b`.

2.5.4.3 Algorítimo Recursivo

O algoritmo recebe um conjunto de submetas, representando a meta de planejamento, $G = \{p_1 = v(p_1), p_2 = v(p_2), \cdots, p_n = v(p_n)\}$, em que p_i é um átomo proposicional e $v(p_i)$ é uma possível valoração, isto é, verdadeiro (1) ou falso (0). Dizemos que cada par $p_i = v(p_i)$ é uma submeta. Assim, seja $p \wedge q$ a meta ϕ de um problema de planejamento $\Pi =< D, s_0, \varphi >$, dizemos que $G = \{p = 1, q = 1\}$ (MENEZES *et al.*, 2012).

A partir de cada submeta $p_i = v(p_i)$, o algoritmo constrói, de forma recursiva, o domínio relevante das proposições que contribuem para o alcance desta submeta. De inicio inserimos ao domínio relevante de p_i o valor $v(p_i)$. Em seguida encontramos o conjunto de ações relevantes para a nossa submeta, da forma como apresentado na seção anterior.

Seja $\mathbb{B}$ um conjunto formado pelas ações relevantes para a nossa submeta, tal que $\mathbb{B} \subseteq \mathbb{A}$. Descobrimos para cada ação $b \in \mathbb{B}$ o conjunto de variáveis que a nossa submeta depende, $dependentes(submeta)_b$, efetuando o processo de relaxamento.

Por ultimo precisamos descobrir para cada dependente $d \in dependentes(submeta)_b$ qual o valor v(d) precisa assumir para que a ação b seja verdadeira. Caso b seja igual a $\top$ quando d assumir o valor $\top$, efetuamos a chamada recursiva passando os valores $d = v(1)$, caso b seja igual a $\top$ quando d assumir o valor $\perp$, efetuamos a chamada recursiva passando os valores $d = v(0)$, e o processo se repete até que todo o domínio relevantes seja computado. O pseudo código do algoritmo é apresentado a seguir:

Fonte: Elaborado pelo o autor.

```
Entrada: submeta, valorSubmeta
if valorSubmeta já pertence ao domínio relevante de submeta then
    Return
end
submeta.dominio_relevante.adiciona(valorSubemeta);
acoesRelevantes = calcular_acoesRelevantes(submeta);
for Cada ação a em acoesRelevantes do
    dependentesA = calcular_dependentes(a, submeta);
    for Cada dependente d em dependentesA do
        if a == ⊤ quando d == ⊤ then
            calc_dominio_relevantes(d, 1) ;
        end
        calc_dominio_relevantes(d, 0) ;
    end
end
```

3 O USO DE RELEVÂNCIA NA MUDANÇA DO ESTADO INICIAL

A mudança de um estado inicial de um problema de planejamento sem solução, baseada na abordagem de revisão de crenças com a utilização da métrica de distância *hamming*, seleciona estados que possuem distância minimal em relação ao conjunto de proposições que descrevem o estado inicial.

Dentre os diversos estados proveniente do método de revisão de crenças com *distância de hamming*, alguns são mais relevantes que outros. O estudo de relevância no de planejamento automatizado tem sido usado na construção de heurísticas para planejadores tais como a heurística de grafos causais do planejador *Fast Downward* (HELMERT, 2006)

Neste trabalho, propomos uma nova métrica para mudança do estado inicial baseado nas informações das proposições e seus valores relevantes ao alcance da meta de planejamento. Como mostrado no Capítulo 3, os valores relevantes de cada proposição do domínio de planejamento podem ser obtidos por meio de uma função recursiva que extrai estas informações do diagrama de decisão binaria que representam simbolicamente as ações do domínio de planejamento.

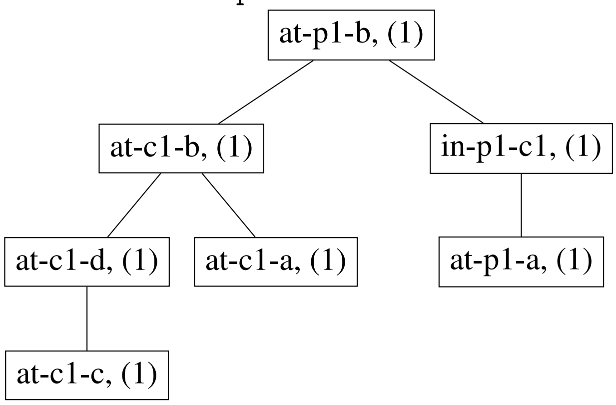
Dentre os estados selecionados como modificações minimais em relação ao estado inicial s_0 pela métrica de distância de *Hamming*, desejamos selecionar aqueles que são *mais relevantes* para o alcance da meta. No entanto, assim como o trabalho de (GÖBELBECKER *et al.*, 2010), queremos *interferir* o mínimo possível nas modificações do estado preferindo mudanças que torne o problema solucionável mas que ainda assim não acabe por executar trabalho que o agente seria capaz de fazer. Imagine a situação onde desejamos que um robô entre em uma sala, porém esta sala está trancada e o robô não tem a chave. Para que o problema torne-se solucionável temos duas modificações minimais em relação a distância de Hamming: (i) destrancar a porta da sala ou; (ii) entregar a chave ao robô. Uma modificação com interferência mínima prefere escolher entregar a chave ao robô para que, assim, ele mesmo possa executar a ação *destrancar a porta*. Esta interferência mínima também é apontada nas abordagens em que robôs pedem ajuda a humanos para executar suas tarefas (*robots asking for helping*) (ROSENTHAL *et al.*, 2012).

Para determinar o valor de relevância de um estado $s \in S$ do domínio de planejamento temos que determinar os valores de relevância das variáveis que o rotulam, isto é, os valores de relevância de cada $p \in L(s)$. Este valor está relacionado com o momento em que descobrimos que uma variável p assumir um dado valor x é relevante para o alcance da meta.

Como apresentado nas seções anteriores, a informação do domínio relevante é descoberto por meio de uma função recursiva que é iniciada a partir de cada uma das sub-metas do problema de planejamento. Assim, podemos visualizar as chamadas recursivas dessa função como uma árvore enraizada em uma dada submeta, a partir da qual a função se inicia. Os nós filhos são construídos a partir das chamadas recursivas do procedimento para cada uma da, variáveis e seus valores das quais a submeta depende. O processo termina quando todos os elementos do domínio relevante tenham sido computados.

A Figura 14 ilustra a árvore da execução da função recursiva, responsável por encontrar o domínio relevante das variáveis do problema de Logística, para a submeta `at-p1-b`.

Figura 14 – Visualização em arvore da função recursiva, que obtém o domínio relevante das variáveis para a submeta `at-p1-b`.



Fonte: Elaborado pelo o autor.

Como mostrado na árvore da Figura 14, na primeira iteração da função o valor (1) é adicionado ao domínio relevantes da variável at-p1-b, que é a submeta em questão. Em seguida, são feitas chamadas recursivas para todas as variáveis que at-p1-b depende, que são as variáveis `at-c1-b` e `in-p1-c1`, nas quais também é inserido o valor (1) aos domínios relevantes destas variáveis. O processo se repete até que todos os domínios relevantes para esta submeta tenham sido computados.

O objetivo da métrica é associar o nível de relevância de cada um dos elementos do domínio relevante ao maior nível este elemento recebe que ele se encontra na árvore, onde o valor mínimo é 1.

Definição 7 *(Caminho em uma Árvore) Definimos o caminho de uma árvore a sequência nós distintos* $\{v1, v2, \cdots, vk\}$*, tal que não existe outro nó entre nós consecutivos (isto é, não existe um nó entre v1 e v2, entre v2 e v3,* $\cdots$*, v(k-1) e vk) as relações "é filho de" ou "é pai de"é denominado um caminho na árvore (NETTO et al., 2004).*

Definição 8 *(Nível de um Nó) Um nível de um nó n em uma árvore é igual a números de nós existentes entre o caminho da raiz até o nó n (NETTO et al., 2004).*

Definição 9 *(Nivel de relevância do par (p_i , v(p_i)) Seja $\mathbb{D} = \langle S, L, T \rangle$ um domínio de planejamento e $\mathbb{P}$ um conjunto de átomos proposicionais, tal que, $\mathbb{P}$ = $\{p_1 \cdots p_n\}$ e $\Pi = \langle \mathbb{D}, s_0, \{g_1 \cdot \cdot g_n\} \rangle$ um problema de planejamento em que $\{g_1 \cdots g_n\}$ são as submetas do problema, e um função de valoração V(p), onde V(p) é igual a 1, caso a variável p possua valor verdade igual a $\top$ e V(p) é igual a 0, caso a variável p possua valor verdade igual a $\perp$. Definimos o nível de relevância de uma variável $p \in \mathbb{P}$ e um valor $v(p)$ pertencente ao domínio de p, denotado por rel(p, v(p)), como:*

$$rel(p, V(p)) = \max_{t^i \in \mathbb{F}} \{Nivel[p, V(p), t_{g^i}]\}$$

Onde $\mathbb{F}$ é um conjunto, tal que $\mathbb{F} = \{t_{g^1}, \cdots, t_{g^n}\}$, em que t_{g^i} é a arvore para a submeta g_i.

Observando a Figura 14 encontramos os seguintes níveis de relevância paras as variáveis do domínio de logística. O par [`at-p1-b`, (1)] = 1, e segue crescendo de forma incremental. Assim o nível de relevância dos pares, [`at-c1-b`, (1)] e [`in-p1-c1`, (1)] é igual a 2, os pares, [`at-c1-d`, (1)], [`at-c1-`, (1)] e [`at-p1-a`, (1)] seria igual a 3, e por fim teríamos o par [`at-c1-c`, (1)] com o nível de relevância igual a 4.

Uma característica importante a ser ressaltada é que devido o fato das chamadas recursivas só serem feitas para elementos que o nó pai depende, significa que todo estado rotulado por nós filhos consegue alcançar um estado rotulado pelo o nó pai com a execução de uma única ação. Assim, um estado que tenha em sua rotulação o elemento [`at-p1-b`, (1)], pode ser alcançado por um estado rotulado pelos elementos filhos, `at-c1-b`, (1) e [`in-p1-c1`, (1)] pode ser alcançado por uma única ação, onde nesse caso especifico seria a ação `unload-c1-p1-b`, assim podemos afirmar intuitivamente que um estado rotulado somente por elementos que possuem um nível de relevância igual a 3, precisa de no mínimo a execução de 2 ações para alcançar um estado que esteja rotulado pela nossa meta.

Baseado no nível de relevância dos elementos do domínio relevantes conseguimos obter uma nova informação, a *Relevância de um estado*, que pode ser definido da seguinte forma:

Definição 10 *(Relevância de um Estados) Dado um domínio de planejamento* $\mathbb{D} = \langle S, L, T \rangle$ *e um estado* $s \in S$*, definimos a relevância de s, denotado por rel(s), como:*

$$\sum_{p \in L(s)} rel(p,1) + \sum_{p \notin L(s)} rel(p,0)$$

Baseado na Definição 10 conseguimos obter a relevância dos estado do Domínio de logística apresentado no Exemplo 2. Assim o estado s_1 que é rotulado pelos elementos, `at-c1-a` = $\top$ e `at-p1-a` = $\top$, possuiria um nível de relevância = 6, isto por quer ambos os elementos `at-c1-a` = $\top$ e `at-p1-a` = $\top$ se encontram no terceiro nível da árvore. Assim conseguimos afirmar que o estado s_1 precisa de no mínimo duas ações para que possa alcançar um estado que satisfaça a meta at-1-b = $\top$

No problema de planejamento sem solução apresentado no Exemplo 2, temos como estado inicial o estado s_3 e desejamos entregar o pacote p1 na cidade b, que é representado pelo estado s_4, porém não conseguimos executar uma sequência de ações que nos faça alcançar este estado a partir de s_3. Por esse motivo o método de revisão calcula a distância de hamming entre o estado s_3 e todo os estados presente no conjunto **U**, selecionando os três estados com a menor distância proposicional em relação a s_3, que são os estados s_0, s_1 e s_8, conforme apresentado na Figura 7.

Com a relevância dos estados temos uma nova informação, a partir dela conseguimos saber qual entres os estados s_0, s_1 e s_8 é o estado que é rotulado com o mínimo de informação necessário para alcançar um estado que satisfaça a meta de planejamento. Como sabemos o estado s_0 é rotulado por $L(s_0) = \{$at-c1-c, at-p1-a$\}$, o estado s_1 tem como rotulação, $L(s_1) = \{$at-c1-a, at-p1-a$\}$ e por fim o estado s_8 é rotulado por $L(s_8) = \{at - c1 - d, in - p1 - c1\}$. Por meio das rotulações conseguimos calcular a de relevância de cada estado, onde temos que a relevância do estado s_0 é igual 7, o estado s_1 é igual a 6 e o estado s_8 possui relevância igual a 5. Assim o estado s_0 seria indicado como o novo estado inicial, por possuir o maior nível de relevância entre os estados selecionados pela distância de hamming, significando assim ser o estado que possui a menor interferência no problema em relação aos outros.

4 METODOLOGIA

O desenvolvimento deste trabalho conforme o seguinte procedimento metodológico:

- Estudo do método de obtenção de domínio relevante por meio da representação simbólica das ações do domínio de planejamento utilizando Diagramas de Decisão Binária (MENEZES *et al.*, 2012);
- Inclusão do cálculo do valor de relevância de uma variável e seu valor no arcabouço de mudanças de problemas de planejamento sem solução;
- Definição da métrica baseada em relevância para a mudança de estado inicial em problemas de planejamento sem solução;
- Implementação da métrica baseada em relevância para a mudança de estado inicial em problemas de planejamento sem solução.

5 CONCLUSÕES E TRABALHOS FUTUROS

Existem três motivos que fazem um problema de planejamento não possuir uma solução: (1) má especificação do estado inicial; (2) uma meta que não pode ser alcançada ou; (3) má especificação do conjunto de ações.

Na literatura existem alguns trabalhos que se propõem a trabalhar com problemas de planejamento sem solução devido a problemas na especificação do estado inicial. Como o trabalho de (GÖBELBECKER *et al.*, 2010) que efetua uma analise dos grafos causais de os grafos de transição de domínio para obter informações sobre a relevância das variáveis e seus valores para o alcance da meta de planejamento. O trabalho de (MENEZES *et al.*, 2012), obtém a informação de relevância a partir da representação proposicional das ações do domínio com Diagramas de Decisão Binária. O trabalho de (MENEZES, 2014) propõe um método baseado em *Revisão de Crenças* com a métrica de Distancia de Hamming, como arcabouço teórico para a mudança do estado inicial.

Neste trabalho definimos uma nova métrica capaz de incorporar a informação de relevância das proposições e seus valores no processo de revisão do estado inicial do problema de planejamento sem solução. Para isso:

- Definimos o nível de relevância de uma variável e seu valor para o alcance da meta de planejamento.
- Definimos o conceito relevância de um estado.
- Implementamos este conceito no arcabouço de mudanças de problemas sem solução (MENEZES, 2014).

Como trabalhos futuros, iremos: Testar a mudança do estado inicial com a nova métrica nos domínios benchmarks da UIPC *Unsolvable Planning Competition*, verificando-se quais os estados sugeridos pela métrica com distância proposicional e quais os sugeridos pela métrica baseada em relevância. Comparação das mudanças sugeridas pela métrica baseada em relevância com as mudanças sugeridas pelo algoritmo de criação de ações fictícias de (GÖBELBECKER *et al.*, 2010).

REFERÊNCIAS

BRYANT, R. E. Graph-based algorithms for boolean function manipulation. **Computers, IEEE Transactions on**, IEEE, v. 100, n. 8, p. 677–691, 1986.

BRYANT, R. E. Binary decision diagrams and beyond: Enabling technologies for formal verification. In: IEEE. **Computer-Aided Design, 1995.** [S.l.], 1995. p. 236–243.

FORBUS, K. D. **Introducing actions into qualitative simulation**. [S.l.], 1988.

GÄRDENFORS, P. **Belief revision**. [S.l.]: Cambridge University Press, 2003. v. 29.

GHALLAB, M.; NAU, D.; TRAVERSO, P. **Automated Planning**. [S.l.]: Elsevier, 2004.

GÖBELBECKER, M.; KELLER, T.; EYERICH, P.; BRENNER, M.; NEBEL, B. Coming up with good excuses: What to do when no plan can be found. **Cognitive Robotics**, v. 10081, 2010.

HELMERT, M. A planning heuristic based on causal graph analysis. In: **ICAPS**. [S.l.: s.n.], 2004. v. 4.

HELMERT, M. The fast downward planning system. **Journal of Artificial Intelligence Research**, v. 26, p. 191–246, 2006.

HERZIG, A.; MENEZES, V.; BARROS, L. N. de; WASSERMANN, R. On the revision of planning tasks. In: IOS PRESS. [S.l.], 2014.

MCDERMOTT, D.; GHALLAB, M.; HOWE, A.; KNOBLOCK, C.; RAM, A.; VELOSO, M.; WELD, D.; WILKINS, D. **PDDL-the planning domain definition language**. 1998.

MENEZES, M. V. **Mudanças em Problemas de Planejamento sem Solução**. Tese (Doutorado) — Tese de Doutorado, Universidade de Sao Paulo, Instituto de Matemática e Estatıstica, 2014.

MENEZES, M. V.; BARROS, L. N. de; PEREIRA, S. do L. Planning task validation. In: **Proc. of the ICAPS Workshop on Scheduling and Planning Applications**. [S.l.: s.n.], 2012. p. 48–55.

NETTO, J. R.; CERQUEIRA, R. d. G.; FILHO, W. C. **Introdução a estrutura de dados: com técnicas de programação em C**. [S.l.]: Rio de Janeiro: Editora Campus, 2004.

PEREIRA, S. do L.; BARROS, L. N. de. **Planejamento sob incerteza para metas de alcançabilidade estendidas**. Tese (Doutorado) — Tese de Doutorado, Universidade de Sao Paulo, Instituto de Matemática e Estatıstica, 2007.

RINTANEN, J. *et al.* Regression for classical and nondeterministic planning. In: **ECAI**. [S.l.: s.n.], 2008. v. 178, p. 568–572.

ROSENTHAL, S.; VELOSO, M.; DEY, A. K. Is someone in this office available to help me? **Journal of Intelligent & Robotic Systems**, Springer, v. 66, n. 1-2, p. 205–221, 2012.

RUSSELL, S.; NORVING, P. **Inteligência Artificial**. 3th. ed. [S.l.]: Elsevier, 2013.