4
Arquitetura

Este Capítulo apresenta a arquitetura proposta nesta tese que possibilita o projeto de ANA, ou seja, agentes que operam guiados pelo seus interesses enquanto levam em consideração as normas do sistema.

Na Seção 4.1 descreveremos os principais tipos primitivos abordados na arquitetura. Na Seção 4.2 descreveremos as operações que constituem a arquitetura proposta e utilizaremos um simples cenário no contexto de missões de resgate para exemplificar tais tipos primitivos e operações. Por fim, apresentaremos algumas discussões sobre a arquitetura proposta.

4.1
Tipos

No modelo arquitetural proposto tal como no modelo BDI existem primitivas que representam as atitudes mentais do agente. Nesta Seção, apresentaremos os tipos primitivos necessários para projetar ANA, como também algumas definições associadas a tais tipos.

A fim de exemplificar tais tipos primitivos e suas definições utilizaremos um cenário no contexto de missões de resgate reguladas por normas (Sycara et al. 2010). Em tal cenário temos dois grupos formando uma coalizão (Woolridge 2011) e suportados por ANA: (*Grupo H*) que é responsável por funções humanitárias, tais como, resgatar feridos que estão numa região potencialmente hostil e prestar assistência médica para os mesmos; e, (*Grupo M*) que é responsável por funções militares, onde o principal objetivo é atacar insurgentes. Mas, devido a coalizão formada por estes dois grupos, o *Grupo M* também pode adotar objetivos relacionados ao fornecimento de escoltas para *Grupo H* durante a realização de um regaste, como também assistir *Grupo H* no fornecimento de serviços médicos.

A fim de cumprir suas missões, os grupos supracitados têm disponível um conjunto de recursos limitados, por exemplo, *Grupo M* pode ter os seguintes recursos: helicópteros, helicópteros terrestres, jipes e soldados. Desta forma, é objetivo de cada grupo a obtenção de mais recursos.

Por fim, assumiremos que cada grupo tem associado uma reputação, logo, é objetivo de cada grupo aumentar ou manter sua reputação.

Neste cenário abordaremos como o comportamento do *Grupo M* pode ser influenciado a partir da definição de um conjunto de normas. Particularmente, as seguintes normas são consideradas:

(Norma 1): Se *Grupo H* está indo resgatar feridos em uma região hostil, *Grupo M* é obrigado a fornecer uma escolta para Grupo H.

(Recompensas): Se a escolta é provida:

(Recompensa 1): A reputação do *Grupo M* é aumentada.

(Recompensa 2): Mais soldados são disponibilizados para o *Grupo M*

(Punição 1): Se a escolta não é provida, a reputação do *Grupo M* é diminuída.

(Norma 2): Se *Grupo M* não é capaz de fornecer uma escolta com os recursos necessários para garantir a segurança mínima para *Grupo H*, *Grupo M* é proibido de realizar a escolta.

(Punição 1): Se a escolta é provida, a reputação do *Grupo M* é diminuída.

(Norma 3): Se durante o fornecimento de serviços médicos o número de feridos excede a capacidade de assistência do *Grupo H*, *Grupo M* é obrigado a assistir *Grupo H* na prestação do serviço.

(Recompensa 1): Se a assistência é fornecida, a reputação do *Grupo M* é aumentada.

(Punição 1): Se a assistência não é fornecida, a reputação do *Grupo M* é diminuída.

4.1.1 Comportamentos

Semelhante aos agentes autônomos tradicionais (Woolridge 2011), ANA são capazes de realizar comportamentos (ou seja, ações a ser executado e objetivos a ser alcançado)-ver definição *Behavior*-.

$$Behavior ::= actionbehavior\langle\langle Action\rangle\rangle \mid goalbehavior\langle\langle Goal\rangle\rangle$$

Onde uma ação do tipo *Action* é representada como um *átomo* do tipo *Atom*[1].

$$Action == Atom$$

No cenário descrito na Seção 4.1, foi definido que o *Grupo M* é capaz de utilizar um conjunto de recursos (por exemplo, helicópteros, helicópteros terrestres, jipes e soldados). Neste trabalho, assumiremos que tais recursos são utilizados a partir da execução de ações, por exemplo, o uso de helicópteros pode ser realizado a partir da ação: $use(helicopteros)$

Um objetivo é representado como um *átomo* prefixado com um identificador apropriado. Para um objetivo onde o desejo do agente é atingir um estado do mundo (Braubach et al. 2005), o identificador *achieve* é utilizado. Já para um objetivo onde o desejo do agente é, simplesmente, a execução de uma determinada ação (Braubach et al. 2005), o identificador do tipo *perform* é utilizado.

$$Goal ::= achieve\langle\langle Atom\rangle\rangle \mid perform\langle\langle Atom\rangle\rangle$$

Por exemplo, como descrito no cenário na Seção 4.1, os objetivos do *Grupo M* são representados como segue:

(*Atacar Insurgentes*):

$achieve\ ataca(insurgentes)$

(*Escoltar* Grupo H):

$achieve\ escolta(grupoh)$

[1]Um *átomo* do tipo *Atom* é um símbolo predicado e uma seqüência de atributos do tipo *Attribute*, conforme descrito em (López 2003)

(*Assistir* Grupo H *nos Serviços Médicos*):

$achieve\ assisti(grupoh)$

(*Mais Soldados Disponíveis*):

$achieve\ disponibiliza(grupom, soldados)$

(*Reputação Aumentada ou Diminuída*):

$perform\ atualizar\_reputacao(grupom, aumentada)$
$perform\ atualizar\_reputacao(grupom, diminuida)$

Dado que ANA são agentes autônomos, eles devem ser capazes de decidir qual comportamento deve ser realizado. Para tanto, eles consideram a *importância* de realizar um determinado comportamento antes de tomar qualquer decisão (Woolridge 2011). A fim de tornar mais preciso a *importância* de realizar um comportamento, permita-me descrever três definições básicas: *Importância* (ver **Def. 1.**), *Motivação* (ver **Def. 2**) e *Satisfação* (ver **Def. 3**).

Definição 1: (*Importância*) A *importância* de realizar um comportamento do tipo *Behavior* é definida pela função *behaviorImportance* que mapeia um comportamento do tipo *Behavior* para um inteiro representando quão importante é a realização de tal comportamento para o agente. A *importância* é avaliada levando em consideração a *motivação* do agente para atingir um objetivo (como definido em (Lopez and Marquez 2004) e formalizado em **Def. 2**), no caso do comportamento ser um objetivo, ou a satisfação do agente em executar uma ação (como definido em (Dam and Winikoff 2011) e formalizado em **Def. 3**), no caso do comportamento ser uma ação.

$$
\begin{array}{|l}
behaviorImportance : Behavior \rightarrow \mathbb{Z} \\
\hline
\forall\, behavior : Behavior \bullet \\
1 : (\ behavior \in \operatorname{ran} goalbehavior \Rightarrow \\
\quad behaviorImportance\ behavior = \\
\quad motivation(goalbehavior^{\sim}\ behavior)) \wedge \\
2 : (behavior \in \operatorname{ran} actionbehavior \Rightarrow \\
\quad behaviorImportance\ behavior = \\
\quad satisfaction(actionbehavior^{\sim}\ behavior))
\end{array}
$$

Definition 2: (*Motivação*) A motivação do agente para alcançar um

objetivo é definida pela função *motivation* que mapeia um objetivo do tipo *Goal* para um inteiro, que representa a intensidade do desejo do agente para alcançar tal objetivo. Tal função é definida em tempo de projeto e é inspirada na função *motivation* apresentada em (Lopez and Marquez 2004).

$$motivation : Goal \rightarrow \mathbb{Z}$$

Definition 3: (*Satisfação*) A *satisfação* do agente em realizar uma ação é definida com base nos ganhos e perdas ao executar uma ação. *Satisfação* é definida pela função *satisfaction* que mapeia uma ação do tipo *Action* para um inteiro, que representa a intensidade da satisfação do agente para executar tal ação. Consideramos que tal função é definida em tempo de projeto e é inspirada no trabalho apresentado em (Dam and Winikoff 2011).

$$satisfaction : Action \rightarrow \mathbb{Z}$$

A fim de esclarecer as definições supracitadas, considere os objetivos e ações do *Grupo M* descritas anteriormente. Uma possível configuração da importância em realizar os comportamentos, ou seja, *motivações* para alcançar os objetivos e *satisfação* para executar as ações, é como descrito abaixo[2]:

(*Atacar Insurgentes*): O objetivo principal do *Grupo M* é atacar os insurgentes, então assumiremos uma maior motivação associada a este objetivo.

$$motivation\,(achieve\ ataca(insurgentes)) = 10$$

(*Fornecimento de Escolta ou Assistência*): Dado que o atingimento dos objetivos resultantes da coalizão está além das funções tradicionais do *Grupo M*, assumiremos uma menor motivação para os objetivos relacionados ao fornecimento de escoltas ou assistência ao *Grupo H* nos serviços médicos.

(*Escoltar*): A motivação do *Grupo M* para escoltar *Grupo H* é:

$$motivation\,(achieve\ escortar(grupoh)) = 8$$

(*Assistir*): A motivação do *Grupo M* para assistir *Grupo H* na prestação de serviços médicos é:

[2]Neste cenário consideramos que o valor de uma *importância* varia de -10 à 10

$$motivation\,(achieve\ assisti(grupoh)) = 8$$

(*Mais Soldados Disponíveis*): Embora ter mais soldados disponíveis seja importante para realização das missões, este não é o principal objetivo do *Grupo M*. Portanto, assumiremos uma motivação igual a 5 para tal objetivo.

$$motivation\,(achieve\ disponibilizados(grupom, soldados)) = 5$$

(*Reputação Aumentada ou Diminuida*): Assumiremos que a reputação do *Grupo M* pode ser aumentada de 1 ou diminuída de -1. Desta forma, definiremos a motivação do *Grupo M* para ter sua reputação aumentada ou diminuída igual a variação ocorrida.

$$motivation\,(perform\ atualizar\_reputacao(grupom, aumentada)) = 1$$
$$motivation\,(perform\ atualizar\_reputacao(grupom, diminuida)) = -1$$

(*Utilizar recursos*): Embora os recursos tenham um custo associado à sua utilização, eles são fundamentais para *Grupo M* cumprir suas missões. Portanto, assumiremos uma satisfação de valor 2 para o *Grupo M* executar as ações para utilização dos recursos, por exemplo, a satisfação para executar uma ação para utilizar soldados é:

$$satisfaction\,(use(soldados)) = 2$$

4.1.2 Normas

Normas regulam o comportamento dos agentes, obrigando-os ou proibindo-os de alcançar um objetivo (Lopez and Marquez 2004) ou executar uma ação (da Silva 2008). Uma vez que uma norma é aplicada apenas em determinadas circunstâncias, ela também especifica as situações onde os agentes devem cumpri-la (da Silva 2008). Dado que os agentes são livres para decidir por cumprir ou violar as normas, recompensas podem ser definidas a fim de incentivá-los a cumprir e punições podem ser estabelecidas a fim de desencorajá-los a violar as normas (da Silva 2008). Com base nesta discussão,

adotamos o esquema *Norm* para representar uma norma.

Norm

$$
\begin{array}{l}
identifier : NormSym \\
addressees : Addressees \\
deonticconcept : DeonticConcept \\
activation : Context \\
deactivation : Context \\
behavior : Behavior \\
rewards : \mathbb{P}\, Sanction \\
punishments : \mathbb{P}\, Sanction \\
\hline
rewards \cap punishments = \emptyset
\end{array}
$$

Onde:

(*identifier*): indica o símbolo identificador da norma a partir do conjunto $[NormSym]$[3];

(*addressees*): indica os responsáveis por cumprir a norma, podendo ser agentes (cujo identificador é definido a partir do conjunto $[AgentName]$), papéis assumidos pelos agentes (onde um papel é definido a partir do conjunto $[Role]$) ou grupos de agentes (onde um grupo é definido a partir do conjunto $Group$), como definido pelo tipo $Addressees$;

$$
\begin{aligned}
Addressees ::= \; & name\langle\langle \mathbb{P}_1\, AgentName \rangle\rangle \mid \\
& role\langle\langle \mathbb{P}_1\, Role \rangle\rangle \mid \\
& group\langle\langle \mathbb{P}_1\, Group \rangle\rangle \\
& comp\langle\langle Addressees \times Addressees \rangle\rangle
\end{aligned}
$$

(*deonticconcept*): indica se uma norma é uma obrigação ou proibição, como apresentado na definição $DeonticConcept$;

$$DeonticConcept ::= OBLIGATION \mid PROHIBITION$$

(*activation*): descreve o contexto de ativação da norma. O contexto, como descrito na definição *Context*, pode ser uma condição lógica (representada por um conjunto de crenças - ver o tipo *Belief* descrito em (Lopez and Marquez 2004)) que deve ser satisfeito com base nas crenças

[3]Neste trabalho não estamos interessados na natureza do símbolo identificador da norm, então utilizamos o conjunto $[NormSym]$ para representá-los

do agente e uma condição normativa (formalizada pelo esquema *NormativeCondition*) que é definida a partir da especificação de uma norma e a sua situação atual (representada pelo tipo *NormativeState*). Por fim, um terceiro caso considera essas duas possibilidades e um quarto caso contempla uma situação onde o contexto é sempre satisfeito.

$$\begin{aligned} Context ::= \; & beliefcontext\langle\langle \mathbb{P}\, Belief \rangle\rangle \mid \\ & normativeconditioncontext\langle\langle \mathbb{P}\, NormativeCondition \rangle\rangle \mid \\ & and\langle\langle \mathbb{P}\, Belief \times \mathbb{P}\, NormativeCondition \rangle\rangle \mid \\ & true \end{aligned}$$

NormativeCondition

$normativestate : NormativeState$

$norm : Norm$

$$\begin{aligned} NormativeState ::= \; & ACTIVATED \mid \%A\ norma\ foi\ ativada \\ & DEACTIVATED \mid \%A\ norma\ foi\ desativada \\ & FULFILLED \mid \%A\ norma\ foi\ cumprida \\ & VIOLATED\, \%A\ norma\ foi\ violada \end{aligned}$$

A partir da definição de contexto supracitada conseguimos representar o *interlocking* entre normas, como discutido em (Lopez and Marquez 2004);

(*deactivation*): é o contexto de desativação da norma, especificado da mesma maneira que *activation*;

(*behavior*): descreve o comportamento regulado pela norma (ver definição *Behavior* na Seção 4.1.1);

(*rewards*): estabelece as recompensas a serem fornecidas ao agente após cumprir a norma. Uma recompensa pode ser a realização de um comportamento ou o estabelecimento de uma norma, tal como descrito na definição *Sanction*;

$$\begin{aligned} Sanction ::= \; & behaviorsanction\langle\langle Behavior \rangle\rangle \mid \\ & normsanction\langle\langle Norm \rangle\rangle \end{aligned}$$

(*punishments*): são as punições a serem fornecidas ao agente após violar a norma, assim como as recompensas, a punição pode ser um comportamento ou uma norma;

A fim de ilustrar a estrutura de uma norma, considere *Norma 1* apresentada no cenário descrito na Seção 4.1.

Norma 1: "Se *Grupo H* está indo resgatar feridos numa região hostil, *Grupo M* é obrigado a escoltá-lo".A partir disto, podemos produzir os seguintes componentes da *Norma 1*:

(identifier): Considere que o identificador da *Norma 1* é *norm1*;

$norm1$

(addressees): *Norma 1* é endereçada ao *Grupo M*:

$norm1(\underline{grupom}$

(deonticoncept): *Norma 1* é uma obrigação;

$norm1(grupom,$
$\underline{OBLIGATION},$

(activation): O contexto de ativação é satisfeito se *Grupo H* está indo resgatar feridos em uma região hostil, como apresentado abaixo:

$norm1(grupom,$
$OBLIGATION,$
$\underline{beliefcontext\,\{pos\ regiaohostil(grupoh)\}},$

(deactivation): Assumindo que *Norma 1* está desativada se *Grupo H* não está numa região hostil, o contexto de desativação é descrito como segue:

$norm1(grupom,$
$OBLIGATION,$
$beliefcontext\,\{pos\ regiaohostil(grupoh)\},$
$\underline{beliefcontext\,\{not\ regiaohostil(grupoh)\}},$

(behavior): *Norma 1* é uma obrigação para *Grupo M* escoltar *Grupo H*:

$norm1($ *grupom*,
OBLIGATION,
beliefcontext {*pos hostileregion*(*grupoh*)},
beliefcontext {*not hostileregion*(*grupoh*)},
goalbehavior achieve escolta(*grupoh*),

(recompensas): As recompensas da *Norma 1* são:

(Recompensa 1) O aumento da reputação do *Grupo M*:

$norm1($ *grupom*,
OBLIGATION,
beliefcontext {*pos hostileregion*(*grupoh*)},
beliefcontext {*not hostileregion*(*grupoh*)},
goalbehavior achieve escolta(*grupoh*),
{*behaviorsanction goalbeahvior perform*
atualizar_reputacao(*grupom*, *aumentada*),

(Recompensa 2) A disponibilização de mais soldados:

$norm1($ *grupom*,
OBLIGATION,
beliefcontext {*pos hostileregion*(*grupoh*)},
beliefcontext {*not hostileregion*(*grupoh*)},
goalbehavior achieve escolta(*grupoh*),
{*behaviorsanction goalbeahvior perform*
atualizar_reputacao(*grupom*, *aumentada*),
behaviorsanction goalbeahvior
achieve disponibilizados(*grupom*, *soldados*)}

(punições): A punição da *Norma 1* é a diminuição da reputação do *Grupo M*:

$norm1($ *grupom*,
OBLIGATION,
beliefcontext $\{$*pos hostileregion*(*grupoh*)$\}$,
beliefcontext $\{$*not hostileregion*(*grupoh*)$\}$,
goalbehavior achieve escolta(*grupoh*),
$\{$*behaviorsanction goalbeahvior perform*
atualizar_reputacao(*grupom*, *aumentada*),
behaviorsanction goalbeahvior achieve
disponibilizados(*grupom*, *soldados*)$\}$,
$\{$*behaviorsanction goalbeahvior perform*
atualizar_reputacao(*grupom*, *diminuida*)$\}$)

4.1.3 Planos

ANA são similares aos agentes BDI apresentados na Seção 2.2, onde eles possuem uma biblioteca de planos representando o conhecimento como atingir seus objetivos.

Planos em ANA consistem de três componentes (similar aos planos descritos na Seção 2.2):

(*Invocation*): Primeiro, a condição de invocação (representada pelo tipo *Invocation*) detalhando as circunstâncias em termos dos objetivos que levam um plano a ser invocado;

(*Context*): Segundo, o contexto (representado pelo tipo *Context* já definido na Seção 4.1.2) satisfeito a partir das crenças e condições normativas. Note que em ANA, é possível definir condições normativas, ou seja, condições definidas a partir do estado atual das normas do agente para que planos possam ser considerados aplicáveis, além de condições a serem satisfeitas a partir das crenças do agente, como já considerado pelo agente BDI apresentado na Seção 2.2;

(*Body*): Terceiro, o corpo do plano (representado pelo tipo *Body*) que especifica a sequência de comportamentos que o agente deve realizar para atingir o objetivo especificado na condição de invocação;

Plan

1 : *inv* : *Invocation*
2 : *context* : *Context*
3 : *body* : *Body*

$Invocation == Goal$

$Body == \text{seq}\, Behavior$

A fim de exemplificar a estrutura de planos supracitada, considere que o *Grupo M* tenha a sua disposição uma base de planos, tais planos guiam o *Grupo M* no atingimento de seus objetivos e no uso de seus recursos.

Por exemplo, considere os planos 1 e 2. Tais planos têm a mesma *condição de invocação*, ambos são acionados sempre que o *Grupo M* deseja escoltar o *Grupo H*. O *contexto* do plano 1 é satisfeito se a *Norma 3*, que obriga *Grupo M* a fornecer assistência médica, está ativada. O *contexto* do plano 2 é sempre verdadeiro. O corpo do Plano 1 realiza ações para utilizar soldados e helicópteros, e atingi o objetivo de assistir o *Grupo H* nos serviços médicos. O corpo do Plano 2 realiza ações para utilizar soldados e helicópteros terrestres.

Plano 1:

$inv = (achieve\ escolta(grupoh)))$
$context = normativecontext\,\{(ACTIVATED, norm3)\}$
$body = actionbehavior\ use(soldados);$
$actionbehavior\ use(helicopteros);$
$goalbehavior\ achieve\ assisti(grupoh).$

Plano 2:

$inv = (achieve\ escolta(grupoh)))$
$context = true$
$body = actionbehavior\ use(soldados);$
$actionbehavior\ use(helicopterosterrestres).$

4.1.4

Intenções

Similar a intenção adotada pelo modelo BDI apresentado na Seção 2.2, em ANA uma intenção é uma sequência não vazia de planos, com o primeiro elemento da sequência representando o topo da intenção.

$$Intention == \mathrm{seq}_1\ Plan$$

Por exemplo, considere que o *Grupo M* tem uma intenção, nomeada *Intenção*, composta pelo Plano 3 que tem uma condição de invocação que é acionada sempre que *Grupo M* deseja atacar insurgentes. Seu contexto é sempre verdadeiro. Seu corpo executa ações para utilizar soldados e jipes.

Plano 3:

$$inv = (achieve\ ataca(insurgentes)))$$
$$context = true$$
$$body = actionbehavior\ use(soldados);$$
$$\qquad actionbehavior\ use(jipes);$$

Então, a *Intenção* é igual a:

$$Intencao = \langle Plano3 \rangle$$

4.2 Estrutura ANA

Nesta seção descreveremos como os componentes e conceitos descritos na Seção anterior são combinados para definir ANA.

ANA consistem dos seguintes componentes (como descrito no esquema *AutonomousNormativeAgent*):

(*Nome do Agente*): Um símbolo identificador do agente (representado por *self*) definido a partir do conjunto [*AgentName*];

(*Plan Library*): ANA tem uma biblioteca de planos (representada pela variável *planlibrary*) que é um repositório contendo os planos disponíveis para o agente;

(*Motivação*): Uma motivação associada a cada objetivo, que é uma medida da intensidade do desejo do agente em atingir um objetivo. A motivação é representada pela definição *motivation* (ver Seção 4.1.1).

(*Satisfaction*): O grau de satisfação que o agente senti ao executar uma ação. *Satisfação* é representada pela definição *satisfaction* (ver Seção 4.1.1).

$$
\begin{array}{|l}
\hline
\textit{AutonomousNormativeAgent} \\
\textit{self} : \textit{AgentName} \\
\textit{planlibrary} : \mathbb{P}_1\ \textit{Plan} \\
\textit{motivation} : \textit{Goal} \rightarrow \mathbb{Z} \\
\textit{satisfaction} : \textit{Action} \rightarrow \mathbb{Z} \\
\hline
\end{array}
$$

Além dos componentes que constituem o estado de agentes BDI (ou seja, as crenças, objetivos, intenções e ações que o agente pode executar), ANA possuem outros componentes em tempo de execução, são eles (tal como descrito no esquema *AutonomousNormativeAgentMentalState*):

(*roles*): Papéis assumidos pelo agente;

(*groups*): Grupos que o agente faz parte;

(*adoptednorms*): Um conjunto de normas que o agente é responsável por cumprir mas, o contexto de ativação ainda não foi satisfeito, representado pela variável *adoptednorms*;

(*activatednorms*): Um conjunto de normas cujo contexto de ativação foi satisfeito, representado pela variável *activatednorms*;

(*deactivatednorms*): Um conjunto de normas cujo contexto de desativação foi satisfeito, representado pela variável *deactivatednorms*;

(*normselection*): Esta relação é utilizada para registrar quais normas foram selecionadas para serem cumpridas, violadas ou ainda não foram selecionadas (representada pela relação *normselection*, que associa uma norma a um tipo *Selection*);

$$
\begin{aligned}
\textit{Selection} ::= \ & \textit{tobe\_fulfilled} \mid \\
& \textit{tobe\_violated} \mid \\
& \textit{not\_evaluated}\ \%\textit{estado inicial}
\end{aligned}
$$

(*normfulfillment*): Esta relação é utilizada para registrar quais normas foram cumpridas, violadas ou o cumprimento ainda não foi verificado (representada pela relação *normfulfillment*, que associa uma norma a um tipo *Fulfillment*);

$$Fulfillment ::= fulfilled \mid violated \mid not\_verified \,\%\, estado\ inicial$$

(*behaviorrealization*): Esta relação é utilizada para registrar quais normas tiveram seu comportamento realizado (representada pela relação *behaviorealization*, que associa uma norma a um tipo *Realization*);

$$Realization ::= realized \mid not\_realized \,\%\, estado\ inicial$$

(*intentionstatus*): Esta relação é similar a descrita em (d'Inverno et al. 2004) e é utilizada para registrar quais intenções estão ativas, suspensas, ou ativas, mas em execução (representado pela relação *intentionstatus*, que associa uma intenção a um tipo *Status*);

$$Status ::= active \mid suspended \mid active\_int\_execution$$

AutonomousNormativeAgentMentalState

$$
\begin{array}{l}
AutonomousNormativeAgent \\
beliefs : \mathbb{P}_1\ Belief \\
goals : \mathbb{P}_1\ Goal \\
intentions : \mathbb{P}\ Intention \\
actions : \mathbb{P}\ Action \\
roles : \mathbb{P}\ Role \\
groups : \mathbb{P}\ Group \\
adoptednorms, activatednorms, deactivatednorms : \mathbb{P}\ Norm \\
normselection : Norm \rightarrow Selection \\
normfulfillment : Norm \rightarrow Fulfillment \\
behaviorrealization : Norm \rightarrow Realization \\
intentionstatus : Intention \rightarrow Status \\
\hline
activatednorms \cap deactivatednorms = \emptyset \\
\mathrm{dom}\ normselection = activatednorms \\
\mathrm{dom}\ normfulfillment = activatednorms \cup deactivatednorms \\
\mathrm{dom}\ behaviorrealization = activatednorms \\
\mathrm{dom}\ intentionstatus = intentions
\end{array}
$$

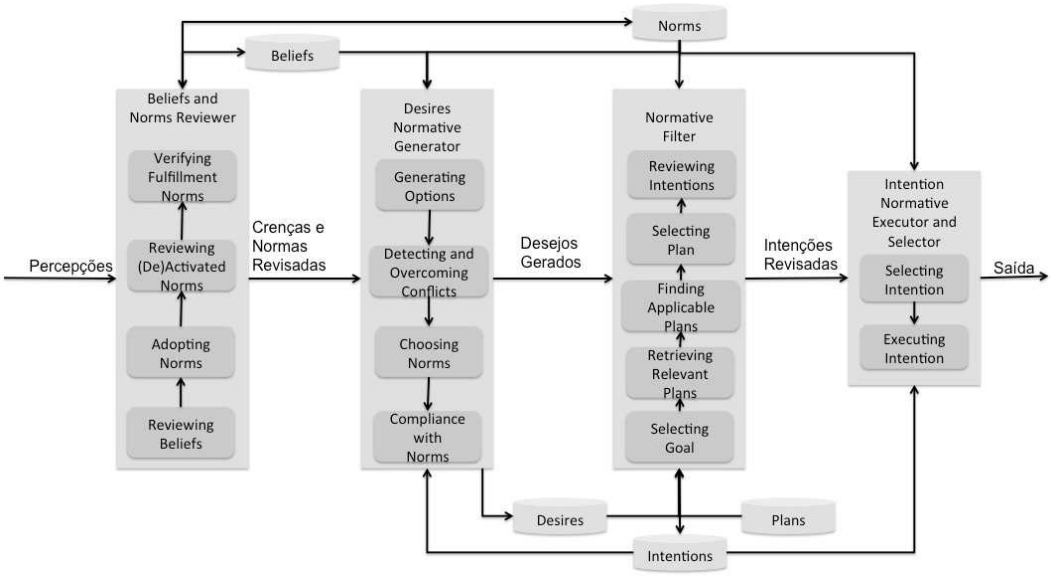

Figura 4.1: Arquitetura para projetar ANA

4.3 Visão Operacional de ANA

O modelo operacional de ANA estende o modelo BDI para que os agentes possam raciocinar sobre as normas do sistema. Figura 4.1 ilustra a arquitetura proposta neste trabalho que guia o funcionamento de ANA.

Em resumo, o agente percebe o mundo através de seus sensores, e revisa suas crenças e normas, levando em consideração as informações percebidas. A função *Belief and Norms Reviewer* (ver Seção 4.3.1) é responsável por: (*Reviewing Beliefs*) rever as crenças do agente, levando em consideração as informações percebidas e as crenças atuais. Esta tarefa funciona como a função *Beliefs Revision* descrita no modelo BDI apresentado na Seção 2.2; (*Adopting Norms*) verifica se a norma percebida é dirigida ao agente e armazena a nova norma no conjunto de normas *adotadas*, se a mesma ainda não existir; (*Reviewing (De)Activated Norms*) atualiza o conjunto de normas *ativadas*, considerando que algumas se tornam ativas e outras inativas devido às percepções, ativação, desativação, cumprimento ou violação de outras normas; e, (*Verifying Fulfillment Norms*) verifica o cumprimento ou violação das normas, considerando que algumas podem ter sido cumpridas e outras violadas devido à operação do agente.

Após rever as normas, a função *Desires Normative Generator* é executada, a fim de: (*Generating Options*) Gerar novos desejos do agente a partir de suas crenças. Esta tarefa funciona como a função *Options Generation* descrita no modelo BDI; (*Detecting and Overcoming Conflicts*) detectar e superar conflitos entre normas ativadas; (*Choosing Norms*) Selecionar as normas que

o agente tem intenção de cumprir e aqueles que pretende violar; e, (*Compliance with Norms*) Atualizar o conjunto de objetivos com base nas normas selecionadas para serem cumpridas ou violadas.

No *Normative Filter*, uma extensão da função *Filter* do modelo BDI, o objetivo com maior prioridade é selecionado a partir do conjunto de objetivos (ver tarefa *Selecting Goal*) e todos os planos cuja condição de invocação combina com o objetivo selecionado são recuperados a partir da biblioteca de planos (ver tarefa *Retrieving Relevant Plans*). Tais planos são chamados de *planos relevantes* e aqueles cujo contexto é satisfeito tomando como base as crenças do agente e as condições normativas são chamados *planos aplicáveis* (ver tarefa *Finding Applicable Plans*). O plano aplicável com maior importância é selecionado para se tornar uma intenção (veja tarefa *Selecting Plan*). Por fim, esta tarefa atualiza as intenções a partir do plano selecionado e levando em consideração que a importância das intenções já existentes podem ter sofrido alteração, dado que algumas normas tornaram-se ativas e outras inativas, como também cumpridas ou violadas (ver tarefa *Reviewing Intentions*).

Finalmente, a função (*Intention Normative Executor and Selector*), que é uma extensão da função *Action Selection* do modelo BDI, é executada a fim de: (*Selecting Intention*) selecionar uma intenção para execução com base na sua importância; e, (*Executing Intention*) executar a intenção selecionada e monitorar sua execução para verificar o cumprimento e violação das normas.

4.3.1 Beliefs and Norms Reviewer

A função *Beliefs and Norms Reviewer* executa três tarefas relacionadas as normas: (*Adopting Norms*) analisa o conjunto de normas *adotadas*; (*Reviewing (De)Activated Norms*) revisa o conjunto de normas ativadas e desativadas, e (*Verifying Fulfillment Norms*) verifica as normas cumpridas ou violadas. Nas próximas Seções apresentaremos detalhadamente tais tarefas. Note que a tarefa (*Reviewing Beliefs*) não é apresentada, dado que esta tarefa funciona como descrito no modelo BDI.

Adopting Norms

A tarefa *Adopting Norms*, formalizada no esquema *NormsAdoption*, recebe como entrada novas normas e atualiza o conjunto de normas adotadas (linha 3), verificando: *(i)* se a nova norma não existe na base de normas adotadas (linha 1); e *(ii)* se o agente, papel desempenhado pelo agente ou grupo que ele faz parte é o destinatário da norma[4] (linha 2).

[4]O destinatário da norma é verificado utilizando a função *isAddressee*

$$
\begin{array}{|l}
\hline
\textit{NormsAdoption} \\
\Delta \textit{AutonomousNormativeAgentMentalState} \\
\textit{newnorms}? : \mathbb{P}\, \textit{Norm} \\
\hline
\forall\, \textit{newnorm} : \textit{newnorms}? \mid \\
1 : \textit{newnorm} \notin \textit{adoptednorms} \land \\
2 : \textit{isAddressee}(\textit{self}, \textit{roles}, \textit{groups}, \textit{newnorm.addressees}) \bullet \\
3 : \quad \textit{adoptednorms}' = \textit{adoptednorms} \cup \{\textit{newnorm}\}) \\
\hline
\end{array}
$$

A fim de exemplificar esta tarefa, considere o cenário apresentado na Seção 4.1 onde temos um agente fornecendo suporte ao *Grupo M*. Se ele recebe informações sobre as normas do sistema, a tarefa *Adopting Norms* é executada para verificar se as normas já estão armazenadas no conjunto de normas adotadas pelo *Grupo M* e comparar as informações do destinatário da norma com *Grupo M*. Neste caso, as Normas 1, 2 e 3 descritas no cenário são dirigidas ao *Grupo M* e as mesmas são adicionadas ao conjunto de normas adotadas pelo *Grupo M*.

Reviewing (De)Activated Norms

Após analisar as crenças e normas, esta tarefa é executada para verificar quais normas foram ativadas ou desativadas. Estes passos são formalizados no esquema *ReviewDeActivatedNorms* e descritos a seguir:

1. Verifica as normas que se tornaram ativas, isto é, aquelas cujo contexto de ativação é satisfeito. Um contexto é verificado utilizando a função *verifyContext* que checa se a condição especificada é consequência lógica das crenças do agente (linha 1), se a condição normativa é satisfeita a partir do estado atual das normas do agente (linha 2) ou se ambas as condições são satisfeitas (linha 3). Se uma destas condições é satisfeita e a norma ainda não existi na base de normas ativadas, ela é adicionada a tal base;

2. Verifica as normas *ativadas* que se tornaram inativas, isto é, aquelas cujo contexto de desativação é satisfeito. Se o contexto de desativação é satisfeito, a norma é removida da base de normas ativadas e, se ela ainda não existi na base de normas desativadas, ela é adicionada a tal base.

$$
\begin{array}{|l}
\textit{ReviewDeActivatedNorms} \\
\textit{ReviewAdoptedNorms} \\
\hline
1 : \forall\, norm : adoptednorms \mid \\
\quad VerifyContext(norm.activation, beliefs, norms) \land \\
\quad norm \notin activatednorms \bullet \\
\qquad activatednorms' = activatednorms \cup \{norm\} \\
2 : \forall\, norm : activatednorms \mid \\
\quad VerifyContext(norm.deactivation, beliefs, norms) \bullet \\
\qquad activatednorms' = activatednorms \setminus \{norm\} \land \\
\qquad (\, norm \notin deactivatednorms \Rightarrow \\
\qquad\quad deactivatednorms' = deactivatednorms \cup \{norm\})
\end{array}
$$

$$
\begin{array}{|l}
VerifyContext : Context \times \mathbb{P}\, Belief \times \mathbb{P}\, Norm \times \mathbb{P}\, Norm \\
\hline
\forall\, bels, bs : \mathbb{P}\, Belief;\ acnorms, deacnorms : \mathbb{P}\, Norm; \\
\quad c1, c2 : Context;\ ncs : \mathbb{P}\, NormativeCondition \bullet \\
1 : (VerifyContext(beliefcontext\ bs, bels, acnorms, deacnorms) \iff \\
\quad LogicalConsequence(bs, bels)) \lor \\
2 : (VerifyContext(normativeconditioncontext\ ncs, \\
bels, acnorms, deacnorms) \iff \\
\quad NormativeConsequence(ncs, acnorms, deacnorms)) \lor \\
3 : (VerifyContext(and(c1, c2), bels, acnorms, deacnorms) \iff \\
\quad VerifyContext(c1, bels, acnorms, deacnorms) \land \\
\quad VerifyContext(c2, bels, acnorms, deacnorms))
\end{array}
$$

A verificação de uma consequência lógica entre uma condição de contexto, que é um conjunto de crenças, e o conjunto de crenças do agente é realizada utilizando o predicado *LogicalConsequence*, descrito em (Lopez and Marquez 2004), que verifica se a condição lógica é uma consequência das crenças do agente. E uma condição normativa é verificada utilizando o predicado *NormativeConsequence* que recebe o conjunto de condições normativas definidas no contexto e o conjunto de normas ativadas ou desativadas do agente como entrada e sua satisfação é dependente do tipo de condição normativa como segue:

(*ACTIVATED,norm*): O *estado normativo* da condição normativa é *ACTIVATED* e a norma especificada na condição normativa existe na base de normas ativadas (linha 1);

(*DEACTIVATED,norm*): O *estado normativo* da condição normativa é *DEACTIVATED* e a norma especificada na condição normativa existe na base de normas desativadas (linha 2);

(*FULFILLED,norm*): O *estado normativo* da condição normativa é *FULFILLED* e a relação *normfulfillment* da norma especificada na condição normativa está associando-a ao tipo *fulfilled* (linha 3);

(*VIOLATED,norm*): O *estado normativo* da condição normativa é *VIOLATED* e a relação *normfulfillment* da norma especificada na condição normativa está associando-a ao tipo *violated* (linha 4);

$$LogicalConsequence : (\mathbb{P}\, Belief \times \mathbb{P}\, Belief)$$

$$NormativeConsequence : \mathbb{P}\, NormativeCondition \times \mathbb{P}\, Norm \times \mathbb{P}\, Norm$$

$$\begin{aligned}
&\forall\, ncs : \mathbb{P}\, NormativeCondition;\ acnorms, deacnorms : \mathbb{P}\, Norm \bullet \\
&NormativeConsequence(ncs, acnorms, deacnorms) \iff \\
&(\forall\, nc : ncs \bullet \\
&1 : (nc.normativestate = ACTIVATED \land \\
&\quad nc.norm \in activatednorms) \lor \\
&2 : (nc.normativestate = DEACTIVATED \land \\
&\quad nc.norm \in deactivatednorms) \lor \\
&3 : (nc.normativestate = FULFILLED \land \\
&\quad normfulfillment\, nc.norm = fulfilled) \lor \\
&4 : (nc.normativestate = VIOLATED \land \\
&\quad normfulfillment\, nc.norm = violated)
\end{aligned}$$

Considere a seguinte situação. Se *Grupo H* vai resgatar feridos em uma região hostil, o contexto de ativação da *Norma 1* é consequência lógica das crenças do agente que fornece suporte ao *Grupo M*, portanto, a *Norma 1* é adicionada a base de normas ativadas e *Grupo M* é obrigado a fornecer uma escolta para *Grupo H*. Agora, se o *Grupo H* não está em uma região hostil, *Norma 1* torna-se inativa e é adicionada ao conjunto de normas desativadas.

Verifying Fulfillment Norms

Após revisar as normas ativadas e desativadas, esta tarefa (especificada no esquema *VerifyFulfillmentNorms*) verifica as normas cumpridas ou violadas como descrito a seguir:

– Verifica as normas ativadas que foram cumpridas ou violadas como segue: (i) se a norma é uma obrigação e o comportamento regulado pela mesma foi realizado, tal norma foi cumprida e a relação *normfulfillment* da norma é atualizada para *fulfilled* (ver linha 2); e, (ii) se a norma é uma proibição e o comportamento regulado pela mesma foi realizado, tal norma foi violada e a relação *normfulfillment* da norma é atualizada para *violated* (ver linha 3).

– Verifica as normas desativadas que foram cumpridas ou violadas como segue: (i) se a norma é uma obrigação e o comportamento regulado pela mesma não foi realizado, tal norma foi violada e a relação *normfulfillment* da norma é atualizada para *violated* (ver linha 5); e, (ii) se a norma é uma proibição e o comportamento regulado pela mesma não foi realizado, tal norma foi cumprida e a relação *normfulfillment* da norma é atualizada para *fulfilled* (ver linha 6).

VerifyFulfillmentNorms

ReviewDeActivatedNorms

$$
\begin{aligned}
&1 : \forall\, norm : activatednorms \mid behaviorrealization\, norm = realized \bullet \\
&2 : \quad (norm.deonticconcept = OBLIGATION \Rightarrow \\
&\quad normfulfillment = normfulfillment \oplus \{(norm, fulfilled)\}) \wedge \\
&3 : \quad (norm.deonticconcept = PROHIBITION \Rightarrow \\
&\quad normfulfillment = normfulfillment \oplus \{(norm, violated)\}) \\
&4 : \forall\, norm : deactivatednorms \mid \\
&behaviorrealization\, norm = not\_realized \bullet \\
&5 : \quad (norm.deonticconcept = OBLIGATION \Rightarrow \\
&\quad normfulfillment = normfulfillment \oplus \{(norm, violated)\}) \wedge \\
&6 : \quad (norm.deonticconcept = PROHIBITION \Rightarrow \\
&\quad normfulfillment = normfulfillment \oplus \{(norm, fulfilled)\})
\end{aligned}
$$

Vamos exemplificar essa tarefa considerando uma situação apresentada na Seção anterior. Se durante a ativação da *Norma 1*, *Grupo M* escoltou *Grupo H*, o comportamento regulado pela *Norma 1* é satisfeito e a relação *normfulfillment* da *Norma 1* é atualizada para *fulfilled*. No entanto, se *Grupo M* não forneceu uma escolta durante a ativação da *Norma 1*, tal norma foi violada e a relação *normfulfillment* da *Norma 1* é atualizada para *violated*.

4.3.2 Desires Normative Generator

Os principais objetivos da função *Desires Normative Generator* é: (i) gerar objetivos de acordo com as crenças do agente tal como descrito na função *Options Generation* do BDI; (ii) selecionar as normas ativadas que o agente tem intenção de cumprir ou violar, e (iii) atualizar o conjunto de objetivos, levando em consideração tal seleção. A fim de alcançar os objetivos (ii) e (iii), esta função executa as seguintes tarefas: (i) *Detecting and Overcoming Conflicts*, (ii) *Choosing Norms*; e, (iii) *Compliance with Norms*.

Antes de descrever tais tarefas, apresentaremos como um agente avalia as normas do sistema, a fim de selecionar aquelas que devem ser cumpridas ou violadas. Em suma, os ganhos e perdas pelo cumprimento ou violação das normas do sistema podem ser medidos a partir da avaliação dos componentes das normas, ou seja, avaliando a influência (ou seja, obrigação ou proibição) deôntica da norma sobre o comportamento regulado pela mesma e a *importância* para o agente em receber as recompensas ou punições da norma.

A influência deôntica representa o impacto de uma obrigação ou proibição sobre o comportamento de um agente, como descrito na **Def. 4**.

Definição 4 (*Influência Deôntica*) A influência deôntica de uma norma sobre o comportamento de um agente é avaliada utilizando a seguinte estratégia (ver a função *deonticInfluence*): se a norma é uma obrigação, ela está influenciando o agente a realizar o comportamento regulado por ela, então a influência deôntica da obrigação é igual a importância do comportamento ser realizado (ver linha 1 e a **Def. 1** a fim de verificar como a importância em realizar um comportamento é avaliada), e, se a norma é uma proibição, ela está impedindo que o agente realize tal comportamento, então a influência deôntica é igual a menos a importância do comportamento ser realizado (ver linha 2).

$$
\begin{array}{|l}
deonticinfluence : DeonticConcept \times Behavior \ \rightarrow \mathbb{Z} \\
\hline
\forall\, dc : DeonticConcept;\ b : Behavior \bullet \\
1 : (\ dc = OBLIGATION \Rightarrow \\
\qquad deonticInfluence(dc, b) = behaviorImportance\ b\ ) \wedge \\
2 : (\ dc = PROHIBITION \Rightarrow \\
\qquad deonticInfluence(dc, b) = -behaviorImportance\ b)
\end{array}
$$

Por exemplo, considere *Norma 1* que obriga *Grupo M* a fornecer uma escolta para *Grupo H*. Dado que a motivação do *Grupo M* para fornecer uma escolta, como apresentado na Seção 4.1, é 8 e *Norma 1* é uma obrigação, a influência deôntica é avaliada da seguinte forma:

$$deonticinfluence(norm1.deonticconcept, norm1.behavior) =$$
$$deonticinfluence(OBLIGATION, goalbehavior\ achieve\ escolta(grupoh)) =$$
$$behaviorImportance(goalbehavior\ achieve\ escolta(grupoh)) =$$
$$motivation(achieve\ escolta(grupoh)) = 8$$

Agora, considere *Norma 2* que proíbe *Grupo M* de fornecer uma escolta. Novamente, dado que a motivação do *Grupo M* em fornecer uma escolta é 8 e *Norma 2* é uma proibição, a influência deôntica é avaliada como apresentado a seguir:

$$deonticinfluence(norm2.deonticconcept, norm2.behavior) =$$
$$deonticinfluence(PROHIBITION, goalbehavior\ achieve\ escolta(grupoh)) =$$
$$behaviorImportance(goalbehavior\ achieve\ escolta(grupoh)) =$$
$$-motivation(achieve\ escolta(grupoh)) = -8$$

A fim de avaliar a *importância* para o agente em receber as recompensas por cumprir ou punições por violar uma norma. Relembre da estrutura de normas apresentada na Seção 4.1.2 que ambas as recompensas e punições de uma norma é um conjunto de sanções. Desta forma, para avaliar a *importância* por receber as recompensas ou punições de uma norma basta avaliar as respectivas sanções associadas. Para tanto, apresentamos a **Def. 5**.

Definição 5 (*Importância das Sanções*): A *importância* por receber um conjunto de sanções (isto é, um conjunto de recompensas ou punições) é igual a soma da *importância* por receber cada sanção. A *importância* de uma sanção é avaliada da seguinte forma (ver função *sanctionsImportance*): (i) se a sanção é um comportamento, a realização da sanção, significa a realização do comportamento, então a importância da sanção é igual a importância do comportamento ser realizado (ver **Def. 1**); e, (ii) se a sanção é uma norma (ver função *evaluateNormSanction*), ANA adotam uma estratégia otimista onde é verificado o que é mais importante: cumprir a norma e receber as recompensas, ou violá-la e receber as punições. Tal avaliação é realizada verificando se a soma da influência deôntica da norma (ver **Def. 4**) e a importânca das recompensas é maior do que a importânca das punições da norma . Em caso afirmativo, a importância da sanção é igual à influência deôntica da norma mais importância das recompensas da norma. Em qualquer outro caso, a importância da sanção é igual importância das punições da norma.

$$
\begin{array}{|l}
sanctionsImportance : \mathbb{P}\, Sanction \times \mathbb{P}\, Norm \rightarrow \mathbb{Z} \\
\hline
\forall\, sanctions : \mathbb{P}\, Sanction;\ norms : \mathbb{P}\, Norm \bullet \\
\ \forall\, sanction : sanctions \bullet \\
1 : (sanction \in \mathrm{ran}\, behaviorsanction \Rightarrow \\
\quad sanctionsImportance\,(sanctions, norms) = \\
\quad behaviorImportance(behaviorsanction^{\sim}\, sanction)) \wedge \\
2 : (sanction \in \mathrm{ran}\, normsanction \Rightarrow \\
\quad sanctionsImportance\,(sanctions, norms) = \\
\quad evaluateNormSanction(normsanction^{\sim}\, sanction, norms)))
\end{array}
$$

$$
\begin{array}{|l}
evaluateNormSanction : Norm \times \mathbb{P}\, Norm \rightarrow \mathbb{Z} \\
\hline
\forall\, norm : Norm;\ norms : \mathbb{P}\, Norm \bullet \\
\ (deonticinfluence(norm.deonticconcept, norm.behavior) + \\
\ \ sanctionsImportance\,(norm.rewards, norms) \geq \\
\ \ sanctionsImportance\,(norm.punishments, norms) \Rightarrow \\
\ \ \ evaluateNormSanction\,(norm, norms) = \\
\ \ \ \ deonticinfluence(norm.deonticconcept, norm.behavior) + \\
\ \ \ \ sanctionsImportance\,(norm.rewards, norms)) \wedge \\
\ (\neg\, (deonticinfluence(norm.deonticconcept, norm.behavior) + \\
\ \ \ sanctionsImportance\,(norm.rewards, norms) \geq \\
\ \ \ sanctionsImportance\,(norm.punishments, norms)) \Rightarrow \\
\ \ \ \ evaluateNormSanction\,(norm, norms) = \\
\ \ \ \ sanctionsImportance\,(norm.punishments, norms)))
\end{array}
$$

Para exemplificar a **Def. 5**, considere *Norma 1*, apresentada na Seção 4.1. A recompensa pelo cumprimento da *Norma 1* é o aumento da reputação do *Grupo M* e a disponibilização de mais soldados. Como apresentado na Seção 4.1.1, a motivação do *Grupo M* de ter a sua reputação aumentada é 1 e para ter mais soldados disponíveis é 5. Finalmente, a importância das recompensas da *Norma 1* é avaliada como apresentado abaixo:

$$
\begin{aligned}
&sanctionsImportance(norm1.rewards, norms) = \\
&sanctionsImportance(\{behaviorsanction\ goalbehavior\ perform \\
&atualizar\_reputacao(grupom, aumentada), \\
&behaviorsanction\ goalbehavior\ achieve \\
&disponibilizados(grupom, soldados)\}) = \\
&behaviorImportance(goalbehavior\ perform \\
&atualizar\_reputacao(grupom, aumentada))+ \\
&behaviorImportance(goalbehavior\ achieve \\
&disponibilizados(grupom, soldados)) = \\
&motivation(perform\ atualizar\_reputacao(grupom, aumentada))+ \\
&motivation(achieve\ disponibilizados(grupom, soldados)) = \\
&1 + 5 = 6
\end{aligned}
$$

Então, a importância por receber as recompensas por cumprir *Norma 1* é 6.

As punições pela violação da *Norma 1* é a diminuição na reputação do *Grupo M*. Como apresentado na Seção 4.1.1, a motivação do *Grupo M* em ter a reputação diminuída é -1. Finalmente, a importância das punições da *Norma 1* é avaliada como apresentado abaixo:

$$
\begin{aligned}
&sanctionsImportance(norm1.punishments, norms) = \\
&sanctionsImportance(\{behaviorsanction\ goalbehavior\ perform \\
&atualizar\_reputacao(grupom, diminuida)\}) = \\
&behaviorImportance(goalbehavior\ perform \\
&atualizar\_reputacao(diminuida))) = \\
&motivation(perform\ atualizar\_reputacao(grupom, diminuida)) = -1
\end{aligned}
$$

Então, a importância por receber as punições por violar *Norma 1* é -1.

Agora, vamos considerar *Norma 2*. *Norma 2* que não tem nenhuma recompensa a ser aplicada no caso de cumprimento, mas tem a punição relacionada à diminuição da reputação do *Grupo M*. Assim, a *importância* para receber as recompensas da *Norma 2* é 0 e a importância por receber as punições da *Norma 2* é -1, conforme descrito abaixo:

$sanctionsImportance(norm2.punishments, norms)$
$sanctionsImportance(\{behaviorsanction\ goalbehavior\ perform$
$atualizar\_reputacao(grupom, diminuida)\}) =$
$behaviorImportance(goalbehavior\ perform$
$atualizar\_reputacao(grupom, diminuida))) =$
$motivation(perform\ atualizar\_reputacao(grupom, diminuida)) = -1$

Após descrever as definições básicas para avaliar uma norma, temos o *know-how* necessário para descrever o processo de seleção aplicado pelo agente a fim de decidir as normas a serem cumpridas ou violadas.

Detecting and Overcoming Conflicts

Se duas normas diferentes estão ativadas, sendo uma delas uma obrigação e a outra uma proibição, regulam o mesmo comportamento, elas estão em conflito. O agente não é capaz de cumprir (ou violar) ambas as normas, ao mesmo tempo.

Se o agente realiza o comportamento regulado pela norma, estará cumprindo a obrigação e violando a proibição e, conseqüentemente, ele receberá recompensas pelo cumprimento da obrigação e punições por violar a proibição. Mas, se não realizar o comportamento, ele estará violando a obrigação e cumprindo a proibição e, conseqüentemente, ele receberá as punições por violar a obrigação e recompensas por cumprir a proibição.

Tal situação de conflito é solucionada tomando a decisão que é mais importante para o agente: (i) Realizar o comportamento regulado pelas normas conflitantes, receber as recompensas da obrigação e as punições da proibição, ou (ii) não realizar o comportamento regulado pelas normas conflitantes, receber as recompensas da proibição e as punições da obrigação. Lembre-se da Seção 4.1 que a *importância* de realizar um comportamento é igual a motivação do agente para alcançar um objetivo, no caso onde o comportamento é um objetivo, ou a satisfação por executar uma ação, no caso onde o comportamento é uma ação. Então, quanto maior é a importância de uma decisão, maior é a motivação e a satisfação do agente ao tomar tal decisão. Portanto, na visão do agente, ele terá mais ganhos ao tomar tal decisão.

O esquema *DetectOvercomeConflicts* formaliza a nossa abordagem. O conflito entre duas normas é detectado na linha 1, utilizando a função *DetectingConflict*[5]. Um conflito é superado da seguinte forma (veja as linhas 2 e 3). Se a influência deôntica da primeira norma (lembre-se da **Def. 4** que a

[5]Esta função utiliza a função *VerifyBehavior* que verifica se dois comportamentos são similares

influência deôntica de uma norma é definida com base na importância para realizar o comportamento, no caso de uma obrigação, ou por não realizar o comportamento, no caso de uma proibição) mais a importância de receber as recompensas da primeira norma e as punições da segunda norma é maior ou igual que a importância para receber as punições da primeira norma e as recompensas da segunda norma, então a primeira norma é selecionada para ser cumprida, e a segunda para ser violada (ver linha 2). Quando acontece o contrário, a segunda norma é selecionada para ser cumprida e a primeira para ser violada (ver linha 3).

DetectOvercomeConflicts

ReviewDeActivatedNorms

$$
\begin{aligned}
&\forall\, n1 : activatednorms \bullet \\
&\quad \exists\, n2 : activatednorms \setminus \{n1\} \mid \\
&1 : DetectingConflict(n1, n2) \bullet \\
&2 : (\ deonticinfluence(n1.deonticconcept, n1.behavior) + \\
&\qquad rewardsImportance(n1, activatednorms) + \\
&\qquad punishmentsImportance(n2, activatednorms) \geq \\
&\qquad punishmentsImportance(n1, activatednorms) + \\
&\qquad rewardsImportance(n2, activatednorms)) \Rightarrow \\
&\qquad\quad normselection = normselection \oplus \{(n1, tobe\_fulfilled)\} \wedge \\
&\qquad\quad normselection = normselection \oplus \{(n2, tobe\_violated)\}) \wedge \\
&3 : (\neg\ (\ deonticinfluence(n1.deonticconcept, n1.behavior) + \\
&\qquad\quad rewardsImportance(n1, activatednorms) + \\
&\qquad\quad punishmentsImportance(n2.punishments, norms) \geq \\
&\qquad\quad rewardsImportance(n2, activatednorms) + \\
&\qquad\quad punishmentsImportance(n1.punishments, activatednorms)) \Rightarrow \\
&\qquad\quad normselection = normselection \oplus \{(n2, tobe\_fulfilled)\} \wedge \\
&\qquad\quad normselection = normselection \oplus \{(n1, tobe\_violated)\}
\end{aligned}
$$

$$
\begin{aligned}
&DetectingConflict : (Norm \times Norm) \\
&\forall\, n1 : Norm;\ n2 : Norm \bullet \\
&\quad (\ VerifyBehavior(n1.behavior, n2.behavior) \wedge \\
&\quad\ (\ n1.deonticconcept = OBLIGATION \wedge \\
&\quad\quad n2.deonticconcept = PROHIBITION) \vee \\
&\quad\ (\ n2.deonticconcept = OBLIGATION \wedge \\
&\quad\quad n1.deonticconcept = PROHIBITION)\ )
\end{aligned}
$$

Note que no esquema *DetectOvercomeConflicts* utilizamos as funções *rewardsImportance*, que recebe uma norma e um conjunto de normas ativadas e retorna a importância de receber as recompensas por cumprir tal norma, e *punishmentsImportance*, que recebe uma norma e um conjunto de normas ativadas e retorna a importância de receber as punições por violar tal norma.

A função *rewardsImportance* obtém as recompensas das normas similares[6] a norma de entrada (ver linha 1) e calcula a importância por receber as sanções associadas as recompensas da norma de entrada e das normas similares a mesma, como descrito na **Def. 5** (ver linha 2). Já a função *punishmentsImportance* obtém as punições das normas similares a norma de entrada (ver linha 1) e calcula a importância por receber as sanções associadas as punições da norma de entrada e das normas similares a mesma (ver linha 2). Duas normas são consideradas similares se elas estão regulando o mesmo comportamento e possuem o mesmo conceito deôntico. Note que é importante avaliar a importância por receber as recompensas ou punições de uma norma levando em consideração as recompensas e punições das normas similares a tal norma, pois, se uma norma é cumprida, as normas similares a tal norma também serão cumpridas, o mesmo pode ser dito em relação a violação, pois, se uma norma é violada, as normas similares a tal norma também serão violadas.

$$
\begin{array}{|l}
rewardsImportance : Norm \times \mathbb{P}\, Norm \rightarrow \mathbb{Z} \\
\hline
\forall\, norm : Norm;\ norms : \mathbb{P}\, Norm;\ rewards : \mathbb{P}\, Sanction; \\
importance : \mathbb{Z} \bullet \\
\quad rewardsImportance(norm, norms) = importance \iff \\
1 : (\forall\, n : similarNorms(norm, norms) \bullet \\
\qquad rewards = rewards \cup \{n.rewards\}) \wedge \\
2 : (\ importance = sanctionsImportance(rewards, norms))
\end{array}
$$

$$
\begin{array}{|l}
punishmentsImportance : Norm \times \mathbb{P}\, Norm \rightarrow \mathbb{Z} \\
\hline
\forall\, norm : Norm;\ norms : \mathbb{P}\, Norm;\ punishments : \mathbb{P}\, Sanction; \\
importance : \mathbb{Z} \bullet \\
\quad punishmentsImportance(norm, norms) = importance \iff \\
1 : (\forall\, n : similarNorms(norm, norms) \bullet \\
\qquad punishments = punishments \cup \{n.punishments\}) \wedge \\
2 : (\ importance = sanctionsImportance(punishments, norms))
\end{array}
$$

Vamos agora considerar as Normas 1 e 2 do cenário apresentado na Seção

[6]As normas similares a uma norma são encontradas utilizando a função *similarNorms*

4.1 para descrever a nossa abordagem. Se ambas as normas estão ativadas, elas estão em conflito. Assumindo que a influência deôntica, a importância por receber as recompensas e punições das normas 1 e 2 são como descrito na Seção anterior. A influência deôntica da *Norma 1* (ID1) é 8, a importância por receber suas recompensas (IR1) é 6 e por receber suas punições (IP1) é -1. A importância para receber as recompensas da *Norma 2* (IR2) é 0 e as suas punições é -1 (IP2). Assim, dado que:

$$ID1 + IR1 + IP2 > IR2 + IP1$$
$$8 + 6 + (-1) > 0 + (-1)$$

Norma 1 é selecionada para ser cumprida e *Norma 2* é selecionada para ser violada.

Choosing Norms

Assim como no processo de superação de conflitos entre normas, normas não conflitantes são selecionadas para serem violadas ou cumpridas levando em consideração o que é mais importante para o agente.

Por um lado, se o agente decide cumprir uma obrigação, ele irá realizar o comportamento regulado pela obrigação, receberá as recompensas e não receberá as punições. Isto significa que realizar o comportamento e receber as recompensas são mais importante para o agente do que receber as punições. Caso contrário, se o agente decide violar uma obrigação, ele não irá realizar o comportamento regulado pela obrigação, receberá as punições e não receberá as recompensas. Isso significa que receber as punições é mais importante para o agente do que realizar o comportamento e receber as recompensas.

No caso da proibição, se o agente decide cumprir uma proibição, ele não irá realizar o comportamento regulado pela proibição, receberá as recompensas e não receberá as punições. Isso significa que não realizar o comportamento e receber as recompensas são mais importante para o agente que receber as punições. Caso contrário, se o agente decide violar uma proibição, ele irá realizar o comportamento regulado pela proibição, receberá as punições e não receberá as recompensas. Isto significa que receber as punições é mais importante para o agente do que não realizar o comportamento e receber as recompensas.

Por exemplo, considere a situação onde *Norma 3* é ativada e o *Grupo M* é obrigado a assistir *Grupo H* nos serviços médicos. A influência deôntica da *Norma 3* é 8 (dado que a *Norma 3* é uma obrigação e a motivação do *Grupo M* para alcançar o objetivo "assisti(grupoh)"é 8). A importância para

receber recompensas da *Norma 3* é 1 (dado que a reputação do *Grupo M* será aumentada e a motivação do *Grupo M* para ter sua reputação aumentada é 1) e a importância de receber as punições da *Norma 3* é -1 (dado que a punição é a diminuição da reputação do *Grupo M* e a motivação do *Grupo M* para ter sua reputação diminuída é -1). Assim, *Norma 3* é selecionada para ser cumprida uma vez que a soma de sua influência deôntica, que é 8, mais a importância para receber as recompensas, que é 1, é maior do que a importância para receber as punições, que é -1. No entanto, se assumirmos que a motivação do *Grupo M* para assistir *Grupo H* nos serviços médicos é -3, *Norma 3* é selecionada para ser violada.

O esquema *ChooseNorms* resume o raciocínio do agente para decidir por cumprir ou violar as normas ativadas não conflitantes. O agente decide por cumprir uma norma se a sua influência deôntica mais a importância por receber suas recompensas é maior ou igual do que a importância por receber as punições da norma (linha 1). Caso contrário, o agente decide por violar a norma (linha 2).

ChooseNorms

DetectOvercomeConflicts

$$
\begin{array}{l}
\forall\, norm : activatednorms \mid \\
\quad \neg\, (\exists\, n : activatednorms \bullet DetectingConflict(norm, n)) \bullet \\
1: \quad (\, deonticinfluence(norm.deonticconcept, norm.behavior) + \\
\qquad rewardsImportance(norm, activatednorms) \geq \\
\qquad punishmentsImportance(norm, activatednorms) \Rightarrow \\
\qquad\quad normselection = normselection \oplus \{(norm, tobe\_fulfilled)\}) \wedge \\
2: \ (\neg\, (\, deonticinfluence(norm, norm.behavior) + \\
\qquad\quad rewardsImportance(norm, activatednorms) \geq \\
\qquad\quad punishmentsImportance(norm, activatednorms)) \Rightarrow \\
\qquad\qquad normselection = normselection \oplus \{(norm, tobe\_violated)\}))
\end{array}
$$

Compliance with Norms

Após selecionar as normas a serem cumpridas ou violadas, esta tarefa rever os objetivos do agente (através da geração de novos objetivos) visando tornar o agente ciente das decisões tomadas. Para tanto, o processo descrito no esquema *ComplianceNorms* é executado, levando em consideração que se um comportamento regulado por uma norma ativada ainda não existe no conjunto de objetivos ou intenções do agente (ver linha 1) , e uma norma

de obrigação foi selecionada para ser cumprida (ver linha 2) ou uma norma de proibição foi selecionada para ser violada (ver linha 3). Então, o agente decidiu por realizar o comportamento regulado pela norma e um novo objetivo representando o desejo do agente em realizar tal comportamento é adicionado a base de objetivos do agente (ver linha 4).

ComplianceNorms

ChooseNorm

$$
\begin{aligned}
1: & \forall norm : activatednorms \mid \\
& \neg ExistBehavior(norm.behavior, goals, intentions) \bullet \\
2: & ((norm.deonticconcept = OBLIGATION \wedge \\
& normselection\, norm = tobe\_fulfilled) \vee \\
3: & (norm.deonticconcept = PROHIBITION \wedge \\
& normselection\, norm = tobe\_violated)) \wedge \\
4: & norm.behavior \in \operatorname{ran} goalbehavior \Rightarrow \\
& goals' = goals \cup \{goalbehavior^{\sim}\, norm.behavior\} \wedge \\
& norm.behavior \in \operatorname{ran} actionbehavior \Rightarrow \\
& goals' = goals \cup \{perform(actionbehavior^{\sim}\, norm.behavior)\}
\end{aligned}
$$

A fim de esclarecer o passo relacionado ao cumprimento da norma, relembre que a *Norma 1* que obriga *Grupo M* a atingir o objetivo "escolta(grupoh)", foi selecionada para ser cumprida e a *Norma 2* que proíbe *Grupo M* de atingir tal objetivo foi selecionada para ser violada. A partir de tal decisão um novo objetivo do tipo "escolta(grupoh)"é adicionado a base de objetivos do agente.

O mesmo raciocínio é aplicado para *Norma 3* que obriga o agente a atingir o objetivo "assisti(grupoh)"e foi selecionada para ser cumprida. Tal objetivo é adicionado a base de objetivos do agente.

Após efetivar as decisões normativas tomadas, gerando novos objetivos a partir das normas selecionadas para serem cumpridas ou violadas, a função *Normative Filter* (descrita na próxima Seção) seleciona o objetivo com maior prioridade para ser atingido, encontra os planos relevantes e aplicáveis, escolhe aquele com maior importância para se tornar uma intenção e revisa o conjunto de intenções.

4.3.3 Normative Filter

Os principais objetivos da função *Normative Filter* são selecionar um objetivo para ser alcançado, recuperar os planos relevantes para atingir tal

objetivo, verificar os planos aplicáveis, selecionar um deles para se tornar uma intenção, e, finalmente, rever o conjunto de intenções. Para tanto, esta função executa cinco tarefas: (i) *Selecting Goal*; (ii) *Retrieving Relevant Plans*; (iii) *Finding Applicable Plans*; (iv) *Selecting Plan*; e, (v) *Reviewing Intentions*.

Selecting Goal

Semelhante ao modelo BDI, ANA tem um conjunto de objetivos que eles desejam alcançar, mas é necessário escolher um deles para ser atingido. Para tanto, ANA consideram que cada objetivo tem associado a ele uma prioridade, que é definida levando em consideração a motivação do agente para alcançar o objetivo e que a realização de qualquer comportamento pode implicar no cumprimento e violação de um conjunto de normas, conseqüentemente, no recebimento de um conjunto de recompensas e punições.

Baseado nesta discussão a prioridade de um objetivo é definda como descrito na **Def. 6**

Definição 6(*Prioridade de um Objetivo*): A prioridade de um objetivo é definida a partir da função *goalPriority* que mapeia um objetivo a um número inteiro. A prioridade do objetivo é a soma da motivação do agente para atingir o objetivo (ver **Def. 1**) e a influência normativa das normas sobre tal objetivo. A influência normativa avalia, no caso de um comportamento regulado por normas de obrigação, a importância para o agente em realizar o comportamento, receber as recompensas e não receber as punições, ou, no caso de um comportamento regulado por normas de proibição, a importância para o agente em realizar o comportamento, receber as punições e não receber as recompensas, como descrito no **Def. 7**.

$$
\begin{array}{|l}
goalPriority : Goal \times \mathbb{P}\, Norm \rightarrow \mathbb{Z} \\
\hline
\forall\, g : Goal;\ norms : \mathbb{P}\, Norm \bullet \\
\quad goalPriority(g, norms) = motivation\ g + \\
\qquad\qquad\qquad\qquad behaviorNormativeInfluence(goalbehavior\ g, norms)
\end{array}
$$

Definição 7 (*Influência Normativa sobre um Comportamento*): A *influência normativa* das normas sobre um comportamento é definida pela função *behaviorNormativeInfluence* que mapeia um comportamento e o conjunto de normas endereçadas ao agente a um número inteiro, que representa o nível de influência das normas sobre o comportamento. A influência normativa sobre um comportamento é avaliada utilizando a seguinte estratégia: (1) se o comportamento é regulado por uma norma de obrigação, a realização de tal comportamento implicará no cumprimento da norma, portanto, o agente rece-

berá as recompensas e não receberá as punições. Assim, a influência normativa é igual a importância para receber as recompensas, avaliada através da função *rewardsImportance*, e não receber as punições. A importância das punições é avaliada utilizando a função *punishmentsImportance*; e, (2) se o comportamento é regulado por uma norma de proibição, a realização de tal comportamento implica na violação da norma e, conseqüentemente, o agente receberá as punições e não receberá as recompensas. Assim, a *influência normativa* é igual a importância para receber as punições, e não receber as recompensas.

$$
\begin{array}{|l}
behaviorNormativeInfluence : Behavior \times \mathbb{P}\, Norm \rightarrow \mathbb{Z} \\
\hline
\forall\, behavior : Behavior;\ norms : \mathbb{P}\, Norm \bullet \\
\quad \exists\, norm : norms \mid VerifyBehavior(behavior, norm.behavior) \bullet \\
1: \ (norm.deonticconcept = OBLIGATION \Rightarrow \\
\quad behaviorNormativeInfluence(behavior, norms) = \\
\quad rewardsImportance(norm, norms) - \\
\quad punishmentsImportance(norm, norms)) \wedge \\
2: \ (norm.deonticconcept = PROHIBITION \Rightarrow \\
\quad behaviorNormativeInfluence(behavior, norms) = \\
\quad punishmentsImportance(norm, norms) - \\
\quad rewardsImportance(norm, norms))
\end{array}
$$

A fim de explificar as duas últimas definições, relembre que *Grupo M* tem os objetivos "escolta(grupoh)"e "assisti(grupoh)"na sua base de objetivos. A prioridade de tais objetivos são avaliadas como segue:

(*escolta(grupoh)*): Conforme descrito na Seção 4.1.1, a motivação do *Grupo M* para fornecer uma escolta para o *Grupo H* é 8. No entanto, tal objetivo é influenciado pelas Normas 1 e 2. *Norma 1* é uma obrigação e a realização do objetivo "escolta(grupoh)"implica no cumprimento da *Norma 1*, isto é, receber as recompensas da *Norma 1* cuja importância é igual a 6 e não receber as punições da *Norma 1* cuja importância é igual a -1, como descrito na Seção 4.1.1. *Norma 2* é uma proibição e a realização do objetivo "escolta(grupoh) implica na sua violação, isto é, receber as punições da *Norma 2* cuja importância é igual a -1 e não receber as recompensas, cuja importância é igual a 0, dado que *Norma 2* não fornece recompensas, como descrito na Secção 4.1.1. Assim, a influência normativa das Normas 1 e 2 sobre o comportamento "escolta(grupoh)"é avaliada da seguinte forma:

$$
\begin{aligned}
&behaviorNormativeInfluence(goalbehavior\ achieve\\
&escort(grupoh), norms) =\\
&rewardsImportance(norm1, norms) -\\
&punishmentsImportance(norm1, norms) +\\
&punishmentsImportance(norm2, norms) -\\
&rewardsImportance(norm2, norms) =\\
&6 - (-1) + (-1) + 0 = 6
\end{aligned}
$$

Como resultado, a prioridade do objetio "escolta(partyh)"é:

$$
\begin{aligned}
&goalPriority(achieve\ escolta(grupoh)) =\\
&motivation(achieve\ escolta(grupoh)) +\\
&behaviorNormativeInfluence(goalbehavior^{\sim}\\
&achieve\ escolta(grupoh)) =\\
&8 + 6 = 14
\end{aligned}
$$

(*assisti(grupoh)*): Conforme descrito na Seção 4.1.1, a motivação do *Grupo M* para assistir *Grupo H* é 8. No entanto, este objetivo é influenciado pela *Norma 3*. *Norma 3* é uma obrigação e a realização do objetivo "assisti(grupoh)"implica no cumprimento da mesma, isto é, receber as recompensas da *Norma 3* cuja importância é igual a 1 e não receber as punições cuja importância é igual a -1. Assim, a influência normativa da Norma 3 sobre o objetivo "assisti(grupoh)"é avaliada da seguinte forma:

$$
\begin{aligned}
&behaviorNormativeInfluence(goalbehavior\ achieve\\
&assisti(grupoh), norms) =\\
&rewardsImportance(norm3, norms) -\\
&punishmentsImportance(norm3, norms) =\\
&1 - (-1) = 2
\end{aligned}
$$

Como resultado, a prioridade do objetivo "assisti(grupoh)"é:

$$
\begin{aligned}
&goalPriority(achieve\ escort(partyh)) =\\
&motivation(achieve\ assisti(grupoh)) +\\
&behaviorNormativeInfluence(goalbehavior^{\sim}\ achieve\\
&assisti(grupoh)) =\\
&8 + 2 = 10
\end{aligned}
$$

Após avaliar a prioridade dos obejtivos, o agente seleciona aquele com

maior prioridade, tal como formalizado pela função *selectGoal*.

$$
\begin{array}{|l}
selectGoal : \mathbb{P}\, Goal \times \mathbb{P}\, Norm \rightarrow Goal \\
\hline
\forall\, goals : \mathbb{P}\, Goal;\ norms : \mathbb{P}\, Norm;\ goal : Goal \bullet \\
\quad selectGoal(goals, norms) = goal \iff \\
\quad (\exists\, g1 : goals \bullet \forall\, g2 : goals \mid \\
\qquad goalPriority(g1, norms) \geq goalPriority(g2, norms) \bullet goal = g1)
\end{array}
$$

Dado que a prioridade do objetivo "escolta(grupoh)"é maior que a prioridade do objetivo "assisti(grupoh)", o objetivo "escolta(grupoh)"é selecionado para ser atingido.

Finding Relevant and Applicable Plans

Após selecionar o objetivo com maior prioridade, é necessário encontrar os planos relevantes e aplicáveis. Um plano é considerado por ANA para se tornar uma intenção quando eles são relevantes para o objetivo selecionado (pela função *selectGoal* descrita na Seção anterior), e aplicável em relação as crenças atuais e estado das normas do agente.

A verificação dos planos relevantes é realizada utilizando a função *genrelplans* que recebe como entrada um objetivo g e um conjunto de planos ps, e retorna os planos relevantes, ou seja, os planos cujo o objetivo que constitui a condição de invocação é similar ao objetivo selecionado[7] . Este passo funciona de maneira similar ao descrito no BDI apresentado na Seção 2.2.

$$
\begin{array}{|l}
genrelplans : Goal \rightarrow \mathbb{P}\, Plan \rightarrow \mathbb{P}\, Plan \\
\hline
\forall\, g : Goal;\ ps : \mathbb{P}\, Plan \bullet \\
\quad genrelplans\ g\ ps = \{p : ps \mid VerifyGoal(g, p.inv) \bullet p\}
\end{array}
$$

Agora vamos voltar ao nosso exemplo dos planos descritos na Seção 4.1.3 cujas condições de invocação são:

$$
\begin{array}{l}
Plan1 : \\
\quad inv = (achieve\ escolta(grupoh)) \\
Plan2 : \\
\quad inv = (achieve\ escolta(grupoh))
\end{array}
$$

Dado que o objetivo “escolta(grupoh)"foi selecionado para ser alcançado, os planos 1 e 2 são retornados pela função *genrelplans*, dado que suas condições

[7]Tal similaridade é verificada utilizando a função *VerifyGoal*

de invocação são relevantes para lidar com o objetivo “escolta(grupoh)".

Um plano relevante é aplicável se o seu *contexto* é satisfatível. Relembre da Seção 4.3.1, que a satisfação de um contexto pode ser verificada utilizando a função *VerifiContext*.

A função *genapplplans* recebe os planos relevantes, as atuais crenças e normas do agente, e retorna os *planos aplicáveis*.

$$genapplplans : \mathbb{P}\, Plan \rightarrow \mathbb{P}\, Belief \rightarrow \mathbb{P}\, Norm \rightarrow \mathbb{P}\, Norm \rightarrow \mathbb{P}\, Plan$$

$$\begin{aligned}
&\forall\, relplans : \mathbb{P}\, Plan;\ bels : \mathbb{P}\, Belief;\ acnorms, deacnorms : \mathbb{P}\, Norm \bullet \\
&\quad genapplplans\ relplans\ bels\ acnorms\ deacnorms = \\
&\qquad \{rel : Plan \mid \\
&\quad VerifyContext(rel.context, bels, acnorms, deacnorms) \bullet rel\}
\end{aligned}$$

Em resumo, a função *genapplplans* recebe o conjunto de planos relevantes e, para cada plano, tenta encontrar os planos cujo contexto é satisfeito realizando uma verificação das crenças e estados atuais das normas com o contexto de cada plano.

Por exemplo, considere que a Norma 3 está ativada. Nesse caso, nota-se que o contexto do *Plano 1* cuja condição é a ativação da *Norma 3* é satisfeito, e dado que o contexto do *Plano 2* é sempre verdade. Então, os planos 1 e 2 são planos aplicáveis.

Selecting Plan

Assim como no BDI apresentado na Seção 2.2, após encontrar os planos relevantes e aplicáveis, é necessário escolher um deles para se tornar uma intenção dado que um conjunto de planos podem ser capazes de lidar com o objetivo selecionado. Embora o BDI não defina como tal plano é selecionado, consideramos que cada plano tem uma importância associada a ele. Tal importância é definida avaliando a condição de invocação e o corpo do plano.

Em suma, conforme descrito na **Def. 8**, para avaliar a importância de um plano é necessário levar em consideração a prioridade do objetivo que compõe a *condição de invocação* do plano e a *importância* de realizar os comportamentos que constituem o corpo do plano. Dado que a realização dos comportamentos que compõem o plano pode implicar no cumprimento e violação de um conjunto de normas, também é necessário considerar o recebimento de um conjunto de recompensas e punições.

Definição 8(*Importância do Plano*) A importância de um plano é definida pela função *planImportance*, que mapeia um plano para um número inteiro. Importância do plano é a soma da prioridade do objetivo (veja **Def.**

6) que compõe a *condição de invocação*, a sua importância principal (ver **Def 9.**) e a influência normativa das normas sobre os comportamentos do corpo do plano (ver **Def 10.**).

$$
\begin{array}{|l}
planImportance : Body \times \mathbb{P}\, Norm \rightarrow \mathbb{Z} \\
\hline
\forall\, b : Body;\ norms : \mathbb{P}\, Norm;\ imp : \mathbb{Z} \bullet \\
\quad planImportance(b) = goalPriority(p.inv, norms) + \\
\qquad\qquad\qquad\qquad bodyMainImportance\ p.body + \\
\qquad\qquad\qquad\qquad bodyNormativeInfluence(p.body, norms)
\end{array}
$$

Definição 9 (*Importância Principal*): A *importância principal* de um plano é avaliada considerando os comportamentos no corpo do plano, como discutido em (Dam and Winikoff 2011). Tal importância é definida pela função, *bodyMainImportance*, que mapeia um plano para um inteiro. Conforme definido em **Def. 1**, a importância de um comportamento avalia as motivações do agente em atingir um objetivo ou a satisfação em realizar uma ação. Portanto, a *importância principal* de um plano, é a soma da importância para o agente em realizar todos os comportamentos que constituem o corpo do plano.

$$
\begin{array}{|l}
bodyMainImportance : Body \rightarrow \mathbb{Z} \\
\hline
\forall\, b : Body;\ norms : \mathbb{P}\, Norm;\ imp : \mathbb{Z} \bullet \\
\quad bodyMainImportance(b) = imp \iff \\
\qquad (\forall\, i : 1 \,..\, \#b \bullet imp = imp + behaviorImportance(b\, i))
\end{array}
$$

Definição 10(*Influência Normativa sobre o Corpo do Plano*): A *influência normativa* sobre o corpo de um plano é definida pela função, *bodyNormativeInfluence*, que mapeia o corpo de um plano e um conjunto de normas para um número inteiro. Em outras palavras, *bodyNormativeInfluence* retorna a soma da *influência normativa* das normas sobre os comportamentos do corpo de um plano.

$$
\begin{array}{|l}
bodyNormativeInfluence : Body \times \mathbb{P}\, Norm \rightarrow \mathbb{Z} \\
\hline
\forall\, b : Body;\ norms : \mathbb{P}\, Norm;\ imp : \mathbb{Z} \bullet \\
\quad bodyNormativeInfluence(b) = imp \iff \\
\quad (\forall\, i : 1 \,..\, \#b \bullet imp = imp + \\
behaviorNormativeInfluence(b\, i, norms))
\end{array}
$$

Como exemplo, podemos avaliar os planos 1 e 2, descritos na Seção 4.1.3:

(Plano 1) Como apresentado na Seção 4.1.3, a condição de invocação do *Plano* 1 é o objetivo "escolta(grupoh)"cuja prioridade para atingir é igual a 14 (como avaliado anteriormente na Seção *Selecting Goal*), o corpo do plano 1 é composto pelas ações "use(soldados)"e "use(helicopteros)", e um objetivo "assisti(grupoh)". A satisfação para realizar cada uma destas ações é igual 2 e a motivação para atingir o objetivo "assisti(grupoh)"é igual a 8 (como descrito na Seção 4.1.1). Sendo assim, a importância principal do *Plano 1* é igual a 12. Agora, é necessário levar em consideração que o *Plano 1* sofre uma influência normativa da *Norma 3*.

(Influência Normativa da Norm 3) A *Norma 3* obriga o atingimento do objetivo "assisti(grupoh)", se tal norma é cumprida a importância por receber as recompensas é igual 1, dado que se *Norma 3* é cumprida a reputação do *Grupo M* é aumentada e a motivação do *Grupo M* para ter sua reputação aumentada é igual a 1. Entretanto, se tal norma é violada a importância por receber as punições é igual a -1, dado que a reputação do *Grupo M* é diminuída e a motivação do *Grupo M* para ter sua reputação diminuída é igual a -1. Como resultado, a influência normativa da *Norma 3* sobre o objetivo "assisti(grupoh)"é igual 2. Logo, a importância do *Plano 1* é igual 28.

$$
\begin{aligned}
&planImportance(plan1, norms) = goalPriority(plan1.inv, norms)+ \\
&\qquad mainImportance(plan1.body, norms)+ \\
&\qquad bodyNormativeInfluence(plan1.body, norms) = \\
&goalPriority(achieve\ escolta(grupoh), norms)+ \\
&mainImportance(\langle use(soldados);\ use(helicopters); \\
&assisti(grupoh)\rangle, norms)+ \\
&rewardsImportance(norm3, norms)- \\
&punishmentsImportance(norm3, norms) = \\
&14 + 12 + (1) - (-1) = 28
\end{aligned}
$$

(Plano 2) Como apresentado na Seção 4.1.3, a condição de invocação do *Plano* 2 é o objetivo "escolta(grupoh)"cuja prioridade para atingir é igual a 14, o corpo do plano 2 é composto pelas ações "use(soldados)"e "use(helicopterosterrestres)". A satisfação para realizar cada uma destas ações é igual a 2. Sendo assim, a importância principal do *Plano 2* é igual a 4. Dado que o *Plano 2* não sofre influência normativa, a importância do *Plano 2* é igual 18.

$$planImportance(plan1, norms) = goalPriority(plan1.inv, norms) + mainImportance(plan1.body, norms) + bodyNormativeInfluence(plan1.body, norms) =$$
$$goalPriority(achieve\ escolta(grupoh), norms) + mainImportance(\langle use(soldados);\ use(helicoptersterrestres)\rangle, norms)$$
$$14 + 4 = 18$$

Agora consideraremos como selecionar um plano aplicável para se tornar uma intenção. Isto é obtido utilizando a função *planselect*, que recebe um conjunto de planos aplicáveis como entrada e retorna o plano com maior importância (ver **Def. 12**).

$$planselect : \mathbb{P}\, Plan \rightarrow Plan$$

$$\forall\, appplans : \mathbb{P}\, Plan;\ plans : \mathbb{P}\, Plan;\ plan : Plan \bullet$$
$$planselect(appplans) = plan \iff$$
$$(\exists\, p1 : appplans \bullet \forall\, p2 : appplans \mid$$
$$planimportance\ p1.body \geq planimportance\ p2.body \bullet$$
$$plan = p1)$$

Para ilustrar tal função, lembre-se que a importância do *Plano 1* é igual a 28 e a importância do *Plano 2* é igual 18. Então, *Plano 1* é selecionado para tornar-se uma intenção.

Reviewing Intentions

Embora o agente tenha um conjunto de intenções, é necessário levar em consideração que o interesse do agente em manter tais intenções pode mudar dado que a motivação do agente para atingir alguns objetivos e satisfação para executar algumas ações que compõem uma intenção pode ter mudado. Além disto, algumas normas que regulam comportamentos que compõem as intenções também podem ter se tornado ativas e outros inativas. Desta forma, tais intenções precisam ser revistas.

A fim de capacitar o agente com a habilidade de rever suas intenções consideramos que cada intenção tem uma importância associada, que é definida como descrito em **Def. 11**.

Definition 11 (*Importância das Intenções*): A importância de uma intenção é definida por uma função, *intentionImportance*, que mapeia uma intenção a um número inteiro. Conforme descrito na Seção 4.1.4, uma intenção é uma sequência de planos que devem ser executados, por isto, definimos a

importância de uma intenção como a soma da importância dos planos que a compõe (ver **Def 8.** como a importância de um plano é avaliada).

$$
\begin{array}{|l}
intentionImportance : Intention \times \mathbb{P}\, Norm \rightarrow \mathbb{Z} \\
\hline
\forall\, int : Intention;\ norms : \mathbb{P}\, Norm;\ imp : \mathbb{Z} \bullet \\
\quad intentionImportance(int, norms) = imp \iff \\
\quad\quad (\forall\, i : 1 \,..\, \#int \bullet imp = imp + planImportance(int(i), norms)) \\
\end{array}
$$

Por exemplo, considere a *Intenção*, descrita na Seção 4.1.4, que é composta pelo *Plano 3*. A importância da *Intencao* é avaliada como segue:

(**Plano 3**) Como apresentado na Seção 4.1.3, a condição de invocação do *Plano* 3 é o objetivo "ataca(insurgentes)"cuja prioridade para atingir é igual a 10 (dado que a motivação do *Grupo M* para atingir tal objetivo é igual a 10 e não existe nenhuma norma influenciando tal objetivo), o corpo do plano 3 é composto pelas ações "use(soldados)"e "use(jipes)". A satisfação para realizar cada uma destas ações é igual 2 (como descrito na Seção 4.1.1). Sendo assim, a importância principal do *Plano 3* é igual a 4. Dado que o *Plano 3* não sofre influência de nenhuma norma. Então, a importância do *Plano 3* é igual 14.

$$
\begin{aligned}
&planImportance(plan3, norms) = goalPriority(plan3.inv, norms) + \\
&\qquad\qquad mainImportance(plan3.body, norms) + \\
&\qquad\qquad bodyNormativeInfluence(plan3.body, norms) = \\
&goalPriority(achieve\ ataca(insurgentes), norms) + \\
&mainImportance(\langle use(soldados);\ use(helicopters)\rangle, norms) \\
&10 + 4 = 14
\end{aligned}
$$

Antes de analisarmos o conjunto de intenções lembre-se, conforme definido na Seção 4.2, que a intenção pode assumir um *status* de três: *active*, *suspended* e *ative_in_execution*. Se a intenção é *suspended* ou *ative_in_execution* não pode ser revista dado que o seu processo de execução já foi iniciado. No entanto, se a intenção é *active*, a mesma ainda não foi selecionada para ser executada e pode ser revista. Desta forma, o primeiro passo do processo de revisão de intenções (ver linha 1 no esquema *ReviewIntentions*) é rever as intenções ativa levando em consideração a *importância* associada a cada uma.

Para tanto, uma nova *intenção* é encontrada utilizando a função *getNewIntention* que recebe um objetivo, a biblioteca de planos, o conjunto de crenças e o conjunto de normas e retorna o plano com maior importância para alcançar o objetivo. A fim de encontrar os planos com maior importância, a função *getNewIntention* usa as funções *genrelplans*, *genapplplans* e *selectPlans*

(apresentadas na Seção 4.3.3).

Por exemplo, ao executar o passo descrito acima, a *Intenção* descrita na Seção 4.1.4 cujo *status* é igual a *active* não sofre mudanças dado que não existe outro plano na base de planos do agente que possa substituir o plano que compõe a intenção.

Depois de revisar as intenções *active*, as intenções ANA são atualizadas com base no plano selecionado para se tornar uma intenção pela função *planselect*. Para realizar este passo, tomamos como base a formalização apresentada em (Georgeff and Lansky 1987), onde é ressaltado que para adicionar uma nova intenção é necessário considerar que o objetivo associado a condição de invocação do plano selecionado, que pode ser um novo objetivo adotado pelo agente ou um objetivo que foi gerado como conseqüência da execução de uma intenção.

No primeiro caso, o objetivo não possui uma intenção associada, tais tipos de objetivos são conhecidos como *indefinidos*(ver função *undefined*), e uma nova intenção (encontrada utilizando a função *getNewIntention* a partir do objetivo selecionado) é gerada e seu *status* definido para *active* (ver linha 2). Já no segundo caso, o objetivo possui uma intenção associada, tais tipos de objetivos são conhecidos como *definidos*(ver função *defined*), e o plano escolhido para atingir este objetivo é adicionado a intenção que gerou o objetivo definido e, por fim, a intenção que gerou tal objetivo atualiza seu *status* para *active_in_execution*(ver linha 3).

A fim de exemplificar o caso onde uma nova intenção é adicionada a base de intenções, lembre-se que *Plano 1* foi escolhido para se tornar uma intenção, portanto, uma nova intenção composta pelo *Plano 1* é criada, o seu *status* é definido como *active* e ela é adicionada ao conjunto de intenções.

ReviewIntentions

$\Delta AutonomousNormativeAgentState$

$1 : (\forall\, int : intentions \mid status\ int = active \bullet$
$int = (\langle getNewIntention((head\ int).inv, planlibrary, beliefs,$
$acnorms, deacnorms)\rangle)$
let $selectedgoal == selectGoal(goals, activatednorms) \bullet$
let $newint == getNewIntention(selectedgoal, planlibrary, beliefs,$
$acnorms, deacnorms) \bullet$
$2 : (undefined\ selectedgoal \Rightarrow$
$intentions' = intentions \cup \{(newint)\}) \wedge$
$intentionstatus' = intentionstatus \cup \{(newint, active)\}) \wedge$
$3 : (defined\ selectedgoal \Rightarrow$
let $oldintention == (the\ selectedgoal) \bullet$
$intentions = intentions \setminus \{oldintention\}$
$newintention.body = (newint) \frown (oldintention)) \wedge$
$intentions = intentions \cup \{newint\} \wedge$
$intentionstatus' = (\{oldintention\} \lhd intentionstatus) \cup$
$\{(newint, active\_in\_execution)\})$

$getNewIntention : Goal \times \mathbb{P}\, Plan \times \mathbb{P}\, Belief \times \mathbb{P}\, Norm \times \mathbb{P}\, Norm \rightarrow Plan$

$\forall\, selectedgoal : Goal;\ planlibrary : \mathbb{P}\, Plan;\ beliefs : \mathbb{P}\, Belief;$
$anorms, dnorms : \mathbb{P}\, Norm \bullet$
$getNewIntention(selectedgoal, planlibrary, beliefs, anorms, dnorms) =$
let $relevantplans == genRelPlans(selectedgoal, planlibrary) \bullet$
let $applicableplans ==$
$genApplPlans\ relevantplans\ beliefs\ anorms\ dnorms \bullet$
$selectPlan(applicableplans, anorms))$

4.3.4 Intention Normative Executor and Selector

Após atualizar as intenções, ANA iniciam a segunda fase do seu ciclo de operação, que diz respeito à execução das suas intenções. O esquema *SelectIntention* representa o início de tal processo, incialmente uma intenção é selecionada utilizando a função *selectIntention*. A variável, *selectedintention*, representa a intenção selecionada. Em seguida, é confirmado que a intenção selecionada deve está *active*. A variável *executingplan* representa o plano em

execução (ou seja, primeiro plano) da intenção e a variável *executingbehavior* o comportamento em execução (ou seja, o primeiro comportamento) neste plano.

SelectIntention

$$\Xi NormativeAgentSpeakAgentState$$
$$selectedintention : Intention$$
$$executingplan : Plan$$
$$executingbehavior : Behavior$$

$$selectedintention = selectIntention\,(intentions, activatednorms)$$
$$executingplan = head\ selectedintention$$
$$status\ selectedintention = active$$
$$executingbehavior = head\ executingplan.body$$

Uma intenção é selecionada com base em sua importância (ver **Def 11.**), Conforme descrito na função *selectintention* abaixo:

$$selectIntention : \mathbb{P}\, Intention \times \mathbb{P}\, Norm \rightarrow Intention$$

$$\forall\, intentions : \mathbb{P}\, Intention;\ norms : \mathbb{P}\, Norm;\ intention : Intention \bullet$$
$$selectIntention(intentions, norms) = intention \iff$$
$$(\exists\, i1 : intentions \bullet \forall\, i2 : intentions \mid$$
$$\quad intentionImportance(i1, norms) \geq intentionImportance(i2, norms) \bullet$$
$$\quad intention = i1)$$

Durante a execução de uma intenção duas possibilidades podem surgir devido aos diferentes comportamentos no topo da intenção. Estas variam dependendo se o comportamento é um objectivo ou uma ação.

Primeiro, se o comportamento é um objetivo, um novo objetivo é gerado, adicionado ao conjunto de objetivos a serem alcançados pelo agente e é registrado que tal objetivo está associado a está intenção.

PostGoal

$$\Delta SelectIntention$$

$$executingbehavior \in \mathrm{ran}\ goalbehavior$$
$$\textbf{let}\ goal == goalbehavior^{\sim}\ executingbehavior \bullet$$
$$\quad goals' = goals \cup \{goal\} \wedge$$
$$\quad the' = the \oplus \{(goal, selectedintention)\}$$

Neste caso, o estado da intenção em executação deve ser definido como *suspended*. Processo similar foi aplicado em (d'Inverno et al. 2004).

SuspendedIntention

$$\Delta SelectIntention$$

$$intentionstatus' = intentionstatus \oplus \{(selectedintention, suspended)\}$$

Segundo, se o comportamento é uma ação, ela é adicionada ao conjunto de ações a serem executadas. Neste trabalho, assumimos que cada ação é realizada com sucesso e se existe uma norma regulando tal ação. Então, o comportamento regulado pela norma foi realizado e a relação *behaviorrealization* de tal norma é atualizada para "*realized*"(ver Seção 4.2). Este passo é importante para determinar quais normas de ação foram cumpridas ou violadas, conforme apresentado na Seção 4.3.1.

PostAction

$$\Delta SelectIntention$$

$$executingbehavior \in \text{ran } actionbehavior$$
$$actions' = actions \cup \{action^{\sim}\ executingbehavior\})$$
$$(\forall\, norm : activatednorms \mid$$
$$VerifyBehavior(norm.behavior, executingbehavior) \bullet$$
$$behaviorrealization = behaviorrealization \cup \{norm, realized\})$$

Agora, o comportamento que foi executado é explicitamente removido do corpo do plano (ver esquema *RemoveBehavior*). Assim como definido em (d'Inverno et al. 2004).

RemoveBehavior

$$\Delta SelectIntention$$

$$executingplan'.body = tail(executingplan.body)$$
$$selectedintention' = (\langle executingplan' \rangle) \frown tail(selectedintention)$$
$$intentions' = (intentions \setminus \{selectedintention\}) \cup \{selectedintention'\}$$

Conforme definido no esquema *AchievePlanOnly*, se, após o comportamento é removido, não existem mais comportamentos no plano atual (ou seja, # executingplan.body = <>), mas ainda há mais planos na pilha de intenções (#selectedintetion > 1), o plano atual está terminado e o próximo plano

(representado pela variável *nextplan*) na intenção está pronto para ser executado. Note que um plano foi finalizado, logo, um objetivo foi atingido. Desta forma, se existe uma norma regulando tal objetivo. Então, o comportamento regulado pela norma foi realizado e a relação *behaviorrealization* de tal norma é atualizada para "*realized*".

AchievePlanOnly

$\Delta SelectIntention$

$executingplan.body = \langle\rangle \land$
$\#selectedintention > 1 \land$
let $nextplan == selectedintention\,2 \bullet$
$\quad selectedintention' = (\langle nextplan \rangle) \frown (tail\ selectedintention) \land$
$\quad intentions' = (intentions \setminus \{selectedintention\}) \cup \{selectedintention'\})$
$(\forall\, norm : activatednorms \mid$
$\quad VerifyBehavior(norm.behavior, goalbehavior\ executingplan.inv) \bullet$
$\quad\ behaviorrealization = behaviorrealization \cup \{norm, realized\})$

Finalmente, tal como discutido em (d'Inverno et al. 2004) e formalizado no esquema *AchieveIntention*, se após a realização de um comportamento, o plano foi executado completamente e não há mais planos na intenção (ou seja, *executingplan.body* =<> e # *selectedintention = 1*), a intenção foi realizada e pode ser removida do conjunto de intenções do agente. Além disso, se a intenção é completamente executada, o objetivo da *condição invocação* do plano no topo da intenção foi atingido, isto significa que algumas normas podem ter seus comportamentos realizados e tais normas são adicionadas ao conjunto de normas cujo comportamento foi realizado.

AchieveIntention

$\Delta AgentIntExecutionState$

$executingplan.body = \langle\rangle \land$
$\#selectedintention = 1$
$(\forall\, norm : activatednorms \mid$
$\quad VerifyBehavior(norm.behavior, goalbehavior\ executingplan.inv) \bullet$
$\quad\ behaviorrealization = behaviorrealization \cup \{norm, realized\})$
$intentions = intentions \setminus \{selectedintention\}$

A fim de ilustrar as etapas anteriores, lembre-se que na base de intenções

do agente existem duas intenções. A primeira composta pelo plano 1, cuja importância é igual a 28 e a segunda composta pelo plano 3 cuja importância é igual a 14 . Sendo assim, a primeira intenção é selecionada para ser executada.

Lembre-se que o Plano 1 é composto pelas ações "use(soldados)", "use(helicopteros)"e o objetivo "assisti(grupoh)". Dado que a intenção selecionada para ser executada é composta pelo *Plano 1*, durante a execução deste plano quando o objetivo "assisti(grupoh)"é atingido o comportamento regulado pela *Norma 3* é realizado, então a relação *behaviorrealization* de tal norma é atualizada para "REALIZED".

4.4 Considerações Finais

Este Capítulo apresentou uma arquitetura para projetar agentes capazes de lidar com normas autonomamente (também conhecidos como ANA). Um agente projetado a partir da arquitetura proposta é capaz de:(i) verificar quais normas ele é responsável (ver esquema *AdoptingNorms*);(ii) atualizar as normas ativadas ou desativadas ((ver esquema *ReviewDeActivatedNorms*);(ii) verificar as normas que foram cumpridas ou violadas; (iv) dectar e superar conflitos entre normas levando em consideração as normas do sistema (ver esquema *DetectOvercomeConflicts*); (v) selecionar as normas não conflitantes que devem ser cumpridas ou violadas (ver esquema *ChoosingNorms*) ; (vi) gerar novos objetivos para serem atingidos a partir das decisões normativas (ver esquema *CompliancewithNorms*); (vii) selecionar um objetivo para ser atingido (ver esquema *selectGoal*); (viii) encontrar os planos relevantes para atingir o objetivo selecionado (ver função *genrelplans*); (ix) verificar quais dos planos relevantes são aplicáveis (ver função *genapplplans*); (x) selecionar um dos planos aplicáveis para se tornar uma intenção (ver função *selectPlan*); (xi) atualizar as intenções do agente (ver esquema *ReviewIntentions*); e, finalmente, (xii) selecionar, executar e monitorar a execução de uma intenção (ver esquemas apresentados na Seção 4.3.3).

Por fim, note que as decisões tomadas durante o raciocínio normativo de ANA são baseadas na importância do agente em realizar seus comportamentos, ou seja, na motivação em atingir seus objetivos e na satisfação em executar suas ações.