



Universidade Do Minho

Mestrado Integrado Em Engenharia Biomédica

Sistemas Inteligentes

COMPGUIDE

— Ontologia para Guias Clínicas Interpretáveis por Computador —

Braga, Maio 2013



Universidade Do Minho

Mestrado Integrado Em Engenharia Biomédica

Sistemas Inteligentes

COMPGUIDE

— Ontologia para Guias Clínicas Interpretáveis por Computador —

Docentes: Paulo Novais | Tiago Oliveira

Discente: Marta Paes Moreira (54459)

Conteúdo

1	Introdução	4
2	Representação do Algoritmo	5
2.1	Cuidados Pós-Morte	8
2.2	Cuidados Paliativos em Vida	10
2.2.1	Sintomas	11
2.2.2	Gestão, Planeamento e Apoio	16
3	Discussão e Conclusões	17
4	Referências	19

1 Introdução

As perspectivas futuras actualmente associadas à *web* apontam para a sua evolução no sentido da aquisição de uma dimensão iminentemente semântica, na qual a informação seja dotada de um significado explícito, facilitando o seu processamento e integração automáticos. Neste contexto, destaca-se a importância conjunta das linguagens XML (*eXtensible Markup Language*) e RDF (*Resource Description Framework*), devido à habilidade da primeira na definição e customização de esquemas de etiquetas, e à flexibilidade da abordagem à representação de dados da segunda. Um outro elemento fundamental neste processo diz respeito à existência de uma linguagem ontológica capaz de descrever formalmente a semântica das classes e propriedades utilizadas em documentos *web*, que permita a execução, sobre esses mesmos documentos, de tarefas computacionais de raciocínio úteis. Relativamente ao XML, ao RDF ou mesmo ao RDF-*Schema*, a linguagem OWL (*Web Ontology Language*) é apontada como apresentando uma maior capacidade de expressão semântica e, conseqüentemente, de representação de conteúdo *web* interpretável por computador, ao aliar um modelo lógico descritivo e conceptual a um conjunto de operadores consideravelmente mais rico, que inclui, a título de exemplo, a intersecção, a união e a negação. Assim, considera-se que o conceito de ontologia desempenha um papel fundamental na construção do ideal futuro de *web* semântica. Formalmente, é possível descrever uma ontologia como sendo uma entidade que define os termos utilizados para descrever e representar uma área de conhecimento, e que inclui definições computacionalmente viáveis de conceitos básicos no domínio específico considerado, bem como as relações que se estabelecem entre estes [1]. A nível sintático, uma ontologia OWL constitui simultaneamente um documento RDF válido e um documento XML bem formado, pelo que se torna possível processá-la utilizando uma grande quantidade de ferramentas RDF e XML já disponíveis [2]. A adopção de uma abordagem baseada em ontologias veicula não só a possibilidade de construção de conceitos complexos partindo da conjugação de instâncias mais simples, mas também a de incorporar a utilização de um *reasoner*, que permite, por um lado, verificar se as declarações e definições que lhes estão subjacentes são mutuamente consistentes, e, por outro, identificar a correcta adequação entre conceitos e definições, garantindo, em última análise, a manutenção de uma hierarquia apropriada [1, 3]. Em particular, esta abordagem pode ser aplicada no sentido de operacionalizar a utilização de guias clínicas, propondo-se o presente relatório a avaliar a adequação de um modelo de representação desenvolvido com recurso à linguagem OWL à acomodação específica deste tipo de documentos, explicitando as manipulações de informação e modificações consideradas no decorrer do processo de implementação, bem como a subsequente contraposição crítica dos benefícios e limitações da sua utilização, neste contexto.

2 Representação do Algoritmo

Avaliando o índice de conteúdos da guia clínica considerada, procurou-se, numa primeira abordagem, estabelecer uma segmentação lógica dos conjuntos de instruções patentes nas páginas a abranger, de forma a definir uma estrutura clara e objectiva do cerne da representação ontológica. Assim, designou-se o agrupamento das secções direccionadas à prestação de cuidados clínicos na presença de quadros sintomáticos específicos sob uma mesma designação, atribuindo-se a cada uma das restantes secções uma divisão própria. Identificou-se ainda a possibilidade de seccionar o enquadramento geral da guia clínica em dois domínios genéricos, nomeadamente referentes aos cenários em vida e pós-morte. O raciocínio estrutural resultante — ainda desprovido de quaisquer considerações de implementação computacional — encontra-se patente na Figura 1.

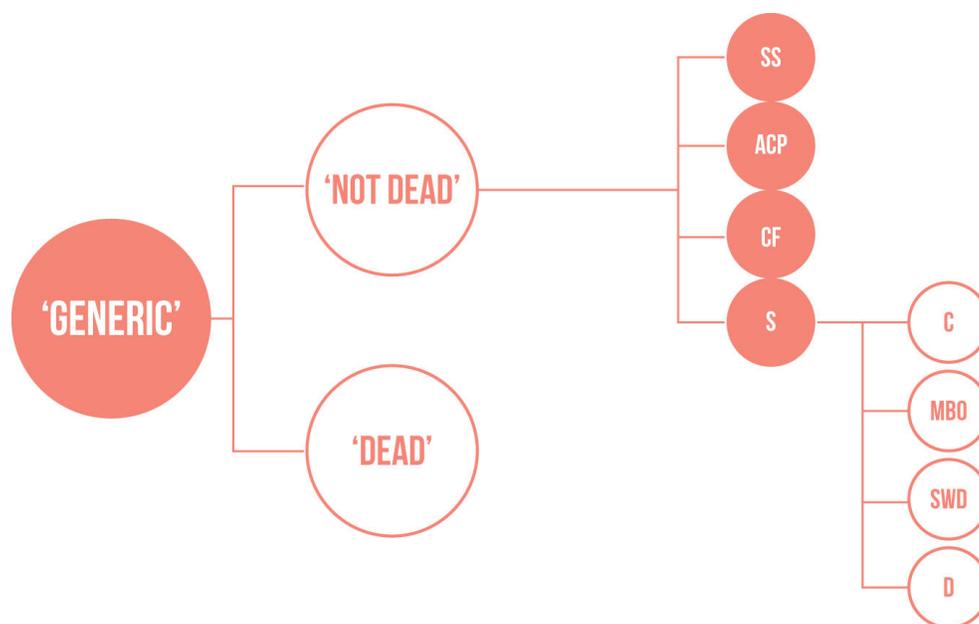


Figura 1: Representação geral da sequência de domínios a considerar na implementação.

A nível computacional, tornou-se necessário declarar, inicialmente, uma nova instância da classe `ClinicalPracticeGuideline`, denominada de `CPG_PalliativeCare`, destinada a desempenhar o papel de invólucro da efectiva representação da guia clínica, ou, por outras palavras, a corporizar a sua efectiva existência enquanto objecto computacional. De seguida, e de forma a completar a definição anterior, criaram-se duas novas instâncias, respectivamente das classes `Scope` e `Plan`, associadas ao objecto declarado por intermédio das propriedades `hasScope` e `hasPlan`, finalizando-se este primeiro processo através do acréscimo das propriedades de dados requeridas. Uma vez definido o ponto de partida para a construção do raciocínio ilustrado, traduziram-se os enunciados na criação de dois planos (`P_DEAD` e `P_NDEAD`) inerentes ao plano genérico considerado (`P_GENERIC`), respectivamente destinados a acolher o conjunto de procedimentos envolvidos na prestação de cuidados paliativos a pacientes em vida e o conjunto de procedimentos destinados às

intervenções após a sua morte. Partindo do plano genérico, definiu-se como primeira tarefa uma questão (Q_{Dead}) — implementada enquanto instância da classe `Question` —, que afere junto do utilizador a condição do paciente, com o intuito de encaminhá-lo no sentido do plano mais adequado à especificidade da situação. A implementação das opções e respectivas tarefas associadas à questão foi efectuada criando uma nova instância da classe `Decision` (D_{Dead}), na qual se definiram duas opções (Op_{Dead} e Op_{NDead}), correspondentes aos possíveis estados do paciente, e duas tarefas alternativas: os planos P_{DEAD} e P_{NDEAD} . Para efeitos de simplificação, apresenta-se a esquematização da referida implementação (Figura 2), contemplando não só as classes, mas também as propriedades envolvidas.

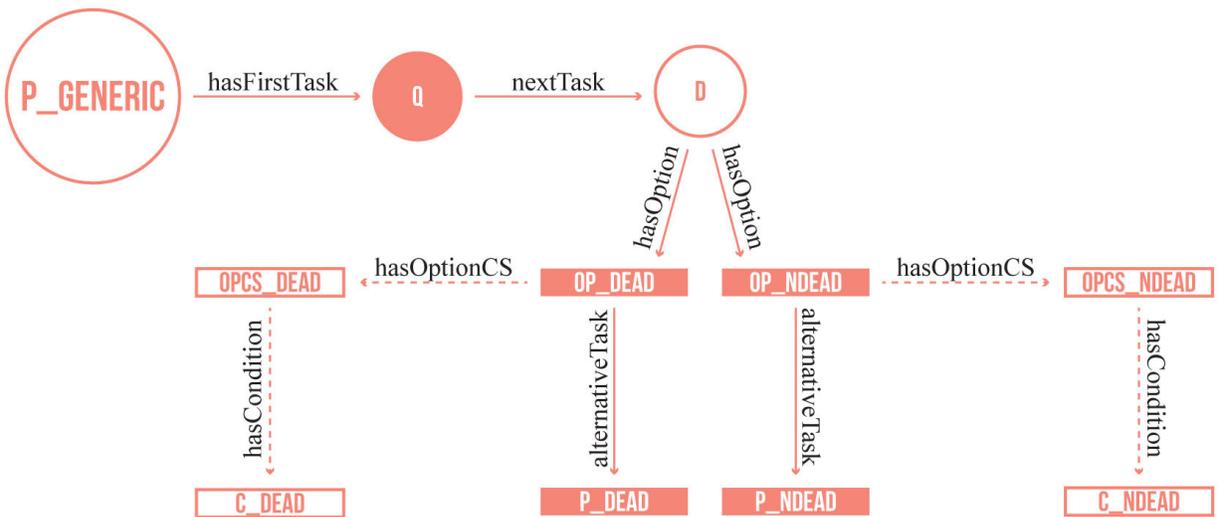


Figura 2: Exemplificação das entidades envolvidas na implementação de uma questão.

Conforme definido no campo das classes equivalentes da entidade `Option`, associaram-se a cada opção três propriedades de dados, nomeadamente um parâmetro (`optionParameter`), um valor (neste caso, `qualitativeValue`) e uma unidade (`unit`), e uma propriedade de objectos do tipo `hasOptionConditionSet`, que associa, por sua vez, uma instância da classe `optionConditionSet` à opção considerada. Note-se que a definição de opções implica necessariamente a criação de novas instâncias da classe `TriggerCondition`, uma vez que, de forma a efectivar a possibilidade de escolha, as tarefas alternativas inerentes à decisão requerem a especificação das condições que as despoletam. Esta particularidade será explicitada de seguida, aquando da exploração individual das diferentes divisões definidas na ontologia.

Alterações à Ontologia

Uma primeira alteração à ontologia pré-existente prendeu-se com a definição das expressões de restrição associadas à classe `Question`, uma vez que, originalmente, a existência de parâmetros se encontrava condicionada pela restrição existencial `some`, implicando a necessidade de definição

de, pelo menos, um parâmetro associado a cada instância da classe. Contudo, e contrariamente ao exemplo fornecido, nenhuma das questões desenvolvidas ao longo da construção da ontologia envolveu a consideração de parâmetros clínicos específicos, pelo que se optou por substituir a referida restrição existencial pela restrição universal *only*.

2.1 Cuidados Pós-Morte

Por apresentar uma complexidade comparativamente inferior, optou-se por abordar primeiramente o plano referente aos cuidados pós-morte definidos na guia clínica. No seguimento das considerações anteriores, uma primeira acção consistiu na criação de uma nova instância da classe `TriggerCondition` (`TC_Dead`), associada, através da propriedade `hasTriggerConditionSet`, ao conjunto `TCS_Dead`, no qual se encerrou a condição de morte do paciente (`C_Dead`). A definição da cadeia de tarefas constituinte do plano foi iniciada através da propriedade `hasFirstTask`, que aponta para a tarefa `A_Dead1`, definida enquanto instância da classe `Action`. Neste domínio, urge destacar que, ao longo de todo o processo de implementação, se considerou que as tarefas descritas constituem uma sucessão ordenada, sendo que, relativamente à guia clínica não informatizada — na qual não é especificado qualquer tipo de ordenação —, se perde alguma agilidade, uma vez que é necessário percorrer sequencialmente todas as tarefas. Por outro lado, adquire-se a capacidade de adaptabilidade, que evita a consideração de tarefas desenquadradas da especificidade das situações, como seja, a título de exemplo, a acção `A_Dead2`, que apenas deve ser executada caso não esteja prevista a realização de uma autópsia. Em situações deste género, que envolvam exposições condicionais, a abordagem utilizada passa pela inclusão de uma questão na dinâmica de trabalho, de forma a estabelecer uma diferenciação entre as acções a executar, conforme patente na Figura 3.

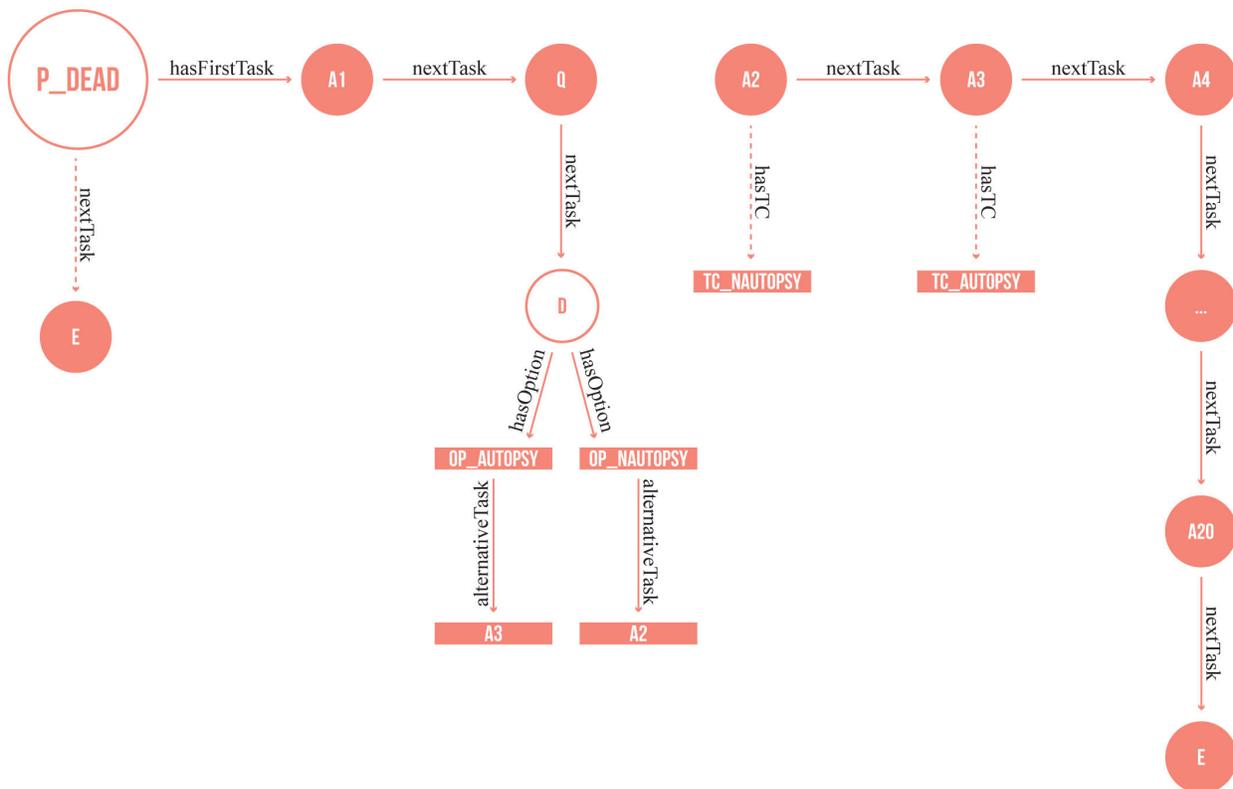


Figura 3: Processo de introdução de uma questão na dinâmica de trabalho.

Individualmente, associou-se a cada tarefa um tipo de acção clínica (`ClinicalActionType`), por intermédio da propriedade `hasClinicalActionType`, assumindo-se que quaisquer acções levadas a cabo pela equipa de cuidados paliativos pertenceriam à classe `Procedure`, enquanto que acções relacionadas com a administração de fármacos se incluíam na classe `MedicationRecommendation`. A classe `NonMedicationRecommendation` foi reservada a acções que se reportassem a outras acções presentes na guia clínica, sendo ainda empregue sempre que uma instrução sugerisse a consulta de entidades externas à equipa ou indicasse recomendações não farmacológicas dirigidas ao paciente. Por último, a classe `Exam` foi apenas contemplada com as acções que explicitamente indicassem a necessidade de realização de exames médicos. Relativamente à definição das propriedades de dados associadas a cada um dos tipos de acção, revelou-se novamente necessário aplicar algumas alterações à ontologia pré-existente, de forma a acomodar a informação veiculada pela guia clínica.

Alterações à Ontologia

Ao nível das propriedades de dados, verificou-se a inexistência de uma forma de representação capaz de traduzir a fragmentação das intervenções enunciadas com base nos respectivos destinatários preferenciais. No caso específico das intervenções pós-morte, são considerados dois agrupamentos distintos, que englobam, respectivamente, 'Family and Caregiver(s)' e 'Health Care Professionals'. De forma a contornar esta situação, criou-se uma nova propriedade de dados (`procedureTarget`), devidamente associada ao domínio `Procedure` e declarada nas classes equivalentes com uma restrição de cardinalidade (`procedureTarget max 1 string`). Duas outras questões de representação levaram ao acréscimo de propriedades de dados à classe `Procedure`, nomeadamente a classificação genérica atribuída a conjuntos de acções específicos e a inclusão de instruções ou indicações adicionais alojadas no interior de outras instruções. No primeiro caso, explorou-se a divisão das tarefas pelos grupos 'Immediate After-Death Care', 'Bereavement Support', 'General Support' e 'After-Death Support' através da propriedade `procedureClass`, associada a uma restrição de cardinalidade semelhante à anterior. No segundo caso, considerou-se a criação da propriedade `procedureDetail`, com a finalidade de adicionar ao procedimento principal quaisquer considerações adicionais que lhe sejam inerentes. Por não se pretender limitar a cardinalidade da utilização desta propriedade, recorreu-se a uma restrição universal aquando da respectiva declaração nas classes equivalentes da classe `Procedure`.

2.2 Cuidados Paliativos em Vida

Naturalmente, o plano que engloba as instruções referentes à prestação de cuidados paliativos a pacientes oncológicos vivos apresenta uma complexidade significativamente superior, sobretudo no que concerne a articulação e a representação das secções de tarefas definidas na fracção da guia clínica considerada. Consoante esquematizado na Figura 1, a divisão considerada lógica passou por definir quatro áreas passíveis de intervenção: 'Symptoms' (S), 'Social Support/Resource Management' (SSupport), 'Goals and Expectations, Educational and Informational Needs and Cultural Factors Affecting Care For the Patient and Family' (CFactors), 'Advance Care Planning' (ACPlanning). Assim, a acção subsequente à escolha do plano P_NDEAD passou pela implementação de uma questão (Q_INTarget) e da respectiva decisão (D_INTarget), no sentido de aferir a área de interesse da consulta, que apresenta como acções alternativas duas outras questões (Q_LifeExpectancy e Q_Symptoms), respectivamente destinadas a avaliar a esperança de vida do paciente e a especificar o sintoma de interesse, conforme ilustrado na Figura 4.

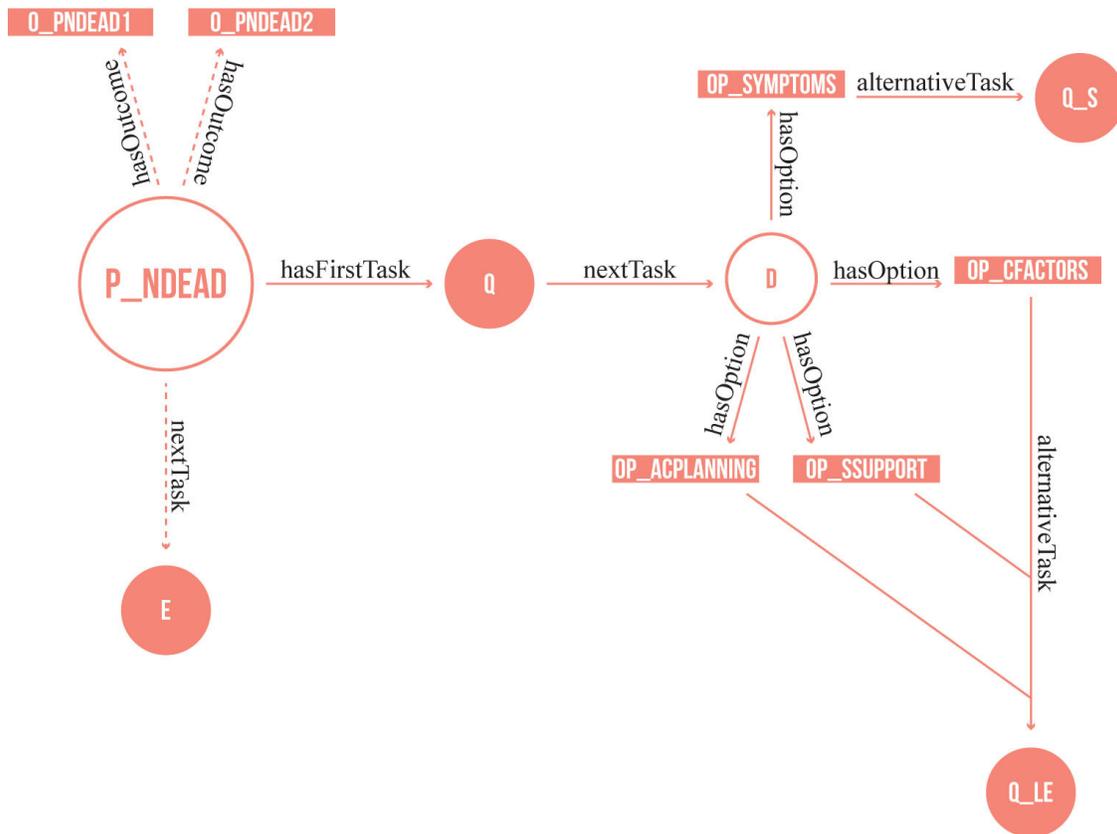


Figura 4: Sequência inicial de acções envolvidas no plano P_NDEad.

Em particular, a questão Q_LifeExpectancy assume um papel de importância transversal à totalidade das áreas de intervenção, já que, consoante a esperança de vida do paciente, o tipo de actuação pode ou não sofrer variações. Neste ponto, uma das possibilidades seria a de agrupar

os períodos de tempo considerados de acordo com um padrão aparentemente dominante, considerando um primeiro grupo formado pelos períodos 'Years', 'Years to Months' e 'Months to Weeks', e um segundo pelo período 'Weeks to Days'. Contudo, tendo em consideração que a divisão de uma das secções não obedece a este padrão, e antevendo futuras alterações à guia clínica, optou-se por encarar cada um dos períodos como condições individuais (C_LEY, C_LEYM, C_LEMW e C_LEWD), forçando-se assim a criação de quatro instâncias distintas das classes *Option* e *TriggerCondition*. Por último, torna-se necessário destacar a criação de duas instâncias da classe *Outcome* (O_PNDead1 e O_PNDead2), que traduzem os dois objectivos implícitos à execução do plano: a gestão eficaz da dor e, em casos terminais, a capacidade de proporcionar ao paciente uma morte tranquila, dentro dos parâmetros definidos. Respeitando as características da classe, cada uma das instâncias encontra-se associada a um *ConditionSet* (OCS1 e OCS2), por intermédio da propriedade *hasOutcomeConditionSet*, composto pelas respectivas condições (C_Outcome1 e C_Outcome2).

2.2.1 Sintomas

No confronto com a questão que averigua a área de intervenção de interesse, a escolha da opção *Op_TSymptoms* desencadeia, conforme mencionado anteriormente, o disparo de uma nova questão, com a qual se pretende particular o sintoma a abordar. Neste caso específico, não se considerou a existência de tarefas alternativas, uma vez que, independentemente da opção contemplada, a tarefa que sucede a decisão é necessariamente a questão relativa à esperança de vida do paciente. O raciocínio básico subjacente a esta abordagem encontra-se ilustrado na Figura 5.

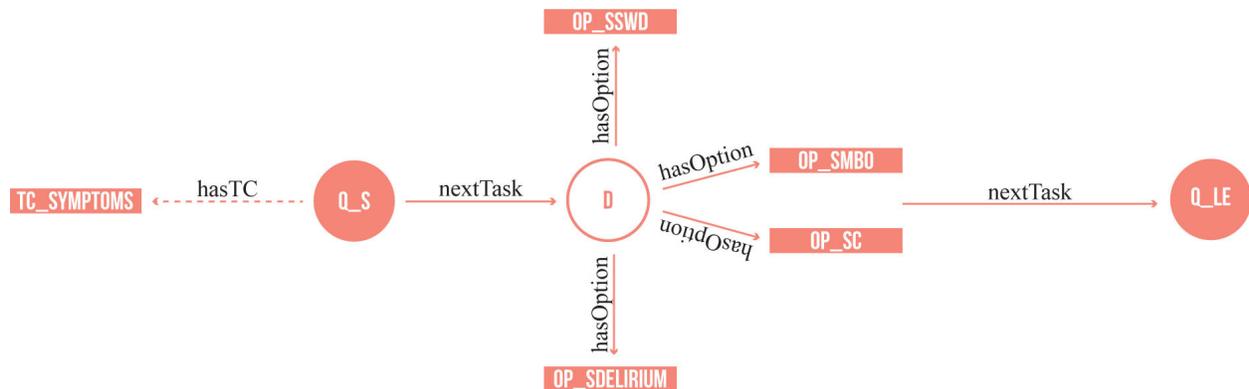


Figura 5: Processo de escolha do sintoma específico a considerar, no interior da respectiva área de intervenção.

Ao nível da implementação individual das sequências de tarefas envolvidas nos métodos de actuação de cada sintoma, urge relevar, numa primeira análise, a inconsistência e variação das representações patentes na guia clínica, que impossibilitaram a construção de um modelo uniforme e coeso, dificultando a adaptação da mesma à ontologia pré-existente. A título de exemplo,

considere-se a declaração de recomendações farmacêuticas, que apenas em casos pontuais compreende na totalidade as propriedades de dados básicas requeridas na construção de uma instância da classe `MedicationRecommendation` — princípio activo, dosagem, forma farmacêutica e posologia. Neste sentido, a metodologia de implementação de cada um dos sintomas será explicitada individualmente, de forma a destacar as referidas discrepâncias, pese embora a resolução dos conflitos encontrados na primeira instância de implementação tenha simplificado de forma significativa as implementações subsequentes.

3.2.1.1 Obstipação

Uma vez que, neste caso em particular, o método de actuação é transversal a todos os períodos de esperança média de vida, a primeira tarefa da sequência (`A_SC1`) encontra-se associada a quatro condições de disparo distintas, considerando-se, no seu seguimento, a inclusão de uma questão (`Q_SC1`) na dinâmica de trabalho, devido à interposição de uma tarefa de cariz condicional. Uma das dificuldades que surgiu, neste ponto, diz respeito à existência de uma tarefa desta natureza que não especifica quaisquer condições de execução, deixando ao livre arbítrio do executante a execução ou não da mesma. Optou-se por não aplicar uma quebra ao ritmo da representação aquando da presença deste tipo de tarefas, já que não faria sentido forçar uma decisão para a qual não são explicitamente fornecidas as condições necessárias. De forma semelhante, a ambiguidade da representação da última tarefa do bloco de medidas preventivas suscitou dúvidas relativamente à abordagem a utilizar, considerando que é apresentada uma recomendação farmacêutica fragmentada em três partes distintas. A solução encontrada para esta situação passou pela criação de uma nova propriedade de objectos, denominada de `complementaryTask`, com a qual se pretendia associar a uma acção principal (neste caso, `A_SC4`) outras acções que lhe fossem intrínsecas ou complementares (neste caso, `A_SC4A` e `A_SC4B`). Considerou-se fundamental, no seguimento desta decisão, a definição de uma nova propriedade no domínio `Annotations`, designada de `SeeAlso`, cujo objectivo se prende com a declaração das acções e tipos de acção clínica que devem ser complementarmente consultados em cada caso específicos. Para além destas alterações, revelou-se necessário acrescentar duas outras à ontologia pré-existente, nomeadamente ao nível da expressão das classes `Condition` e `MedicationRecommendation`. A primeira alteração surge da necessidade de traduzir a existência de um ciclo associado a uma condição (`C_OSC5`) de `Outcome`, estando a segunda destinada a lidar com a limitação da guia clínica previamente exposta, relacionada com a incompletude das recomendações farmacêuticas fornecidas. A acção `A_SC4B` implicou ainda a modificação das propriedades de dados associadas ao domínio `Periodicity`, uma vez que a periodicidade da referida condição não se encontra bem delimitada, devendo esta verificar-se, de acordo com a guia clínica, 'a cada 1-2 dias'. Assim, criaram-se duas novas propriedades (`periodicityMinValue` e `periodicityMaxValue`), podendo a conjunção de ambas ser empregue em alternativa à propriedade pré-existente (`periodicityValue`), em casos que assim o requeiram.

Previamente à implementação do segundo bloco de acções, pressupôs-se que a existência de um primeiro bloco de medidas preventivas deixasse em aberto a possibilidade de o sintoma não

vir a manifestar-se no paciente, pelo que se incluiu nas tarefas alternativas das decisões consideradas uma tarefa do tipo End (E_ASC5). A diferenciação das condições de presença do sintoma e persistência do sintoma, que conduzem a métodos de actuação distintos, foi assegurada através da associação às respectivas condições (C_SCNPersistent e C_SCPersistent) dos operadores temporais `Never_in_the_past` — criado propositadamente com o intuito de traduzir as situações em que, embora o paciente manifeste um dado sintoma no decorrer do período de observação, este nunca tenha apresentado esse mesmo sintoma em observações anteriores — e `Always_in_the_past`, respectivamente. Por último, as restantes acções foram implementadas recorrendo aos métodos explicitados até este ponto e seguindo a linha de raciocínio já mecanizada para a abordagem à especificidade de cada tarefa ou situação enunciadas, sendo que a fase de reavaliação, presente nos métodos de actuação da totalidade dos sintomas e das restantes áreas de intervenção com uma base de construção semelhante, será explorada na secção seguinte.

3.2.1.2 Obstrução Intestinal Maligna

Contrariamente à representação anterior, os métodos de actuação referentes a este sintoma encontram-se fragmentados em dois blocos, distinguindo as acções clínicas direccionadas a pacientes com uma esperança média de vida reduzida das restantes. Neste sentido, identificaram-se, numa primeira fase, as acções iniciais de cada um dos blocos (A_SMB01 e A_SMB04), de forma a garantir a correcta atribuição das respectivas condições de disparo. A evolução da implementação deu-se primeiramente no bloco superior de tarefas, uma vez que o bloco inferior referencia e utiliza alguns métodos de actuação deste último, nomeadamente ao nível de intervenções específicas e tarefas de reavaliação. As acções individuais da sequência de tarefas referentes à avaliação do paciente foram representadas enquanto procedimentos clínicos, sendo que se recorreu à recém-criada propriedade de dados `procedureDetail` para acomodar as indicações específicas inerentes a cada procedimento. Porém, as subseqüentes acções de intervenção representaram um primeiro obstáculo à dinâmica da implementação do bloco, devido ao formato adoptado para a exposição da informação na guia clínica, demonstrando, mais uma vez, que a não linearidade associada a este tipo de documento constitui um sério entrave à sua informatização. Optou-se por forçar uma representação semelhante às anteriores, pese embora se reconheça que este caso poderia ser abordado de uma forma mais específica. Assim, cada instrução de intervenção foi devida e subjectivamente classificada como sendo uma acção principal, complementar de uma outra acção principal, um detalhe de procedimento ou simplesmente um comentário, sendo que as últimas três instâncias foram individualmente precedidas de uma questão, por serem expressamente dependentes da verificação de determinadas condições. Partindo da representação e dinâmicas assim criadas, a implementação do bloco inferior beneficiou da possibilidade de reutilização de algumas das instâncias das classes inerentes à superclasse `ClinicalActionType` previamente criadas, seguindo uma linha de raciocínio semelhante àquela adoptada até este ponto.

Relativamente à implementação da fase de reavaliação (Figura 6), as indicações que verificam os impactos da aplicação da metodologia de actuação sugerida pela guia clínica como

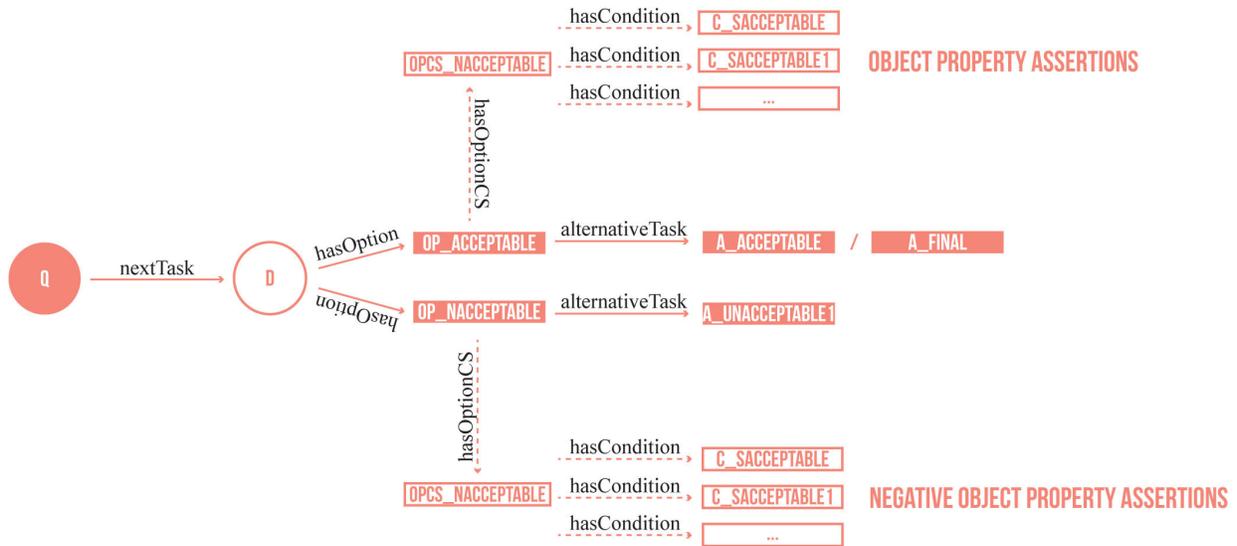


Figura 6: Esquema ilustrativo da fase de reavaliação, transversal à totalidade das áreas de intervenção consideradas.

sendo aceitável foram traduzidas em instâncias da classe `Condition` e encapsuladas no interior de `OptionConditionSets`, para posterior associação aos respectivos elementos de opção. Devido ao facto de se verificar que, ao longo de todos os domínios de intervenção, apenas uma das condições é variável, criou-se um núcleo de condições consideradas básicas (`C_Acceptable2` a `C_Acceptable7`) e, atendendo à particularidade de cada situação, adicionou-se a este núcleo uma condição específica (neste caso, `C_SMBOAcceptable`). Por outro lado, um cenário de reavaliação não aceitável não explicitam quaisquer condições de verificação, pelo que se assumiu a negação das condições anteriores nos conjuntos correspondentes. A topologia da sequência de acções a executar após concluído o processo de decisão depende directamente do domínio de reavaliação considerado, sendo que, no caso de opção pela ramificação correspondente a uma reavaliação favorável, a condição de disparo `TC_Acceptable` pode, dependendo da área de intervenção em questão, estar associada a uma acção intermédia ou directamente a uma acção final de reavaliação contínua. Porém, a ramificação alternativa implica necessária e logicamente a execução de um conjunto de tarefas direccionadas à correcção do cenário de reavaliação, previamente à execução da acção final. No que concerne a implementação da acção que dá por terminada cada uma das sequências de acções, tornou-se inevitável proceder à alteração da ontologia pré-existente, de forma a traduzir a ideia de continuidade que lhe está associada, na descrição da guia clínica. A solução implementada consistiu na criação de uma nova instância da classe `TemporalOperator`, denominada de `Ongoing`, associada por intermédio da propriedade `hasTemporalRestriction` às acções respectivas.

3.2.1.3 Transtornos do Ciclo Vigília-Sono

As instruções de actuação relativas aos transtornos do ciclo vigília-sono subdividem-se, de forma semelhante à situação anteriormente descrita, em dois blocos distintos, que confluem na fase de reavaliação, imediatamente após a sequência de acções correspondentes. Verificou-se, uma vez mais, uma flutuação no molde de representação, nomeadamente ao nível da fragmentação hierárquica das indicações para três (até este ponto, havia-se considerado apenas acções de, no máximo, dois níveis). Contudo, as indicações de segunda ordem, neste caso específico, reportam-se alternativamente a detalhes de procedimento ou a recomendações farmacêuticas, pelo que se optou por representar as primeiras através da propriedade `procedureDetail` e implementar as segundas enquanto tarefas complementares de uma acção principal de cariz farmacêutico, conforme exemplificado na Figura 7. Uma alteração introduzida à ontologia pré-existente prendeu-se com a presença de uma referência bibliográfica associada a uma das recomendações farmacêuticas, que foi representada por intermédio da criação de uma nova propriedade no domínio `Annotations`, designada de `reference`. A implementação do bloco inferior e de todas as sequências de acções subsequentes processou-se de forma idêntica àquelas descritas nas secções anteriores, uma vez que a guia clínica se apresenta homogénea ao nível da representação da fase final de reavaliação.

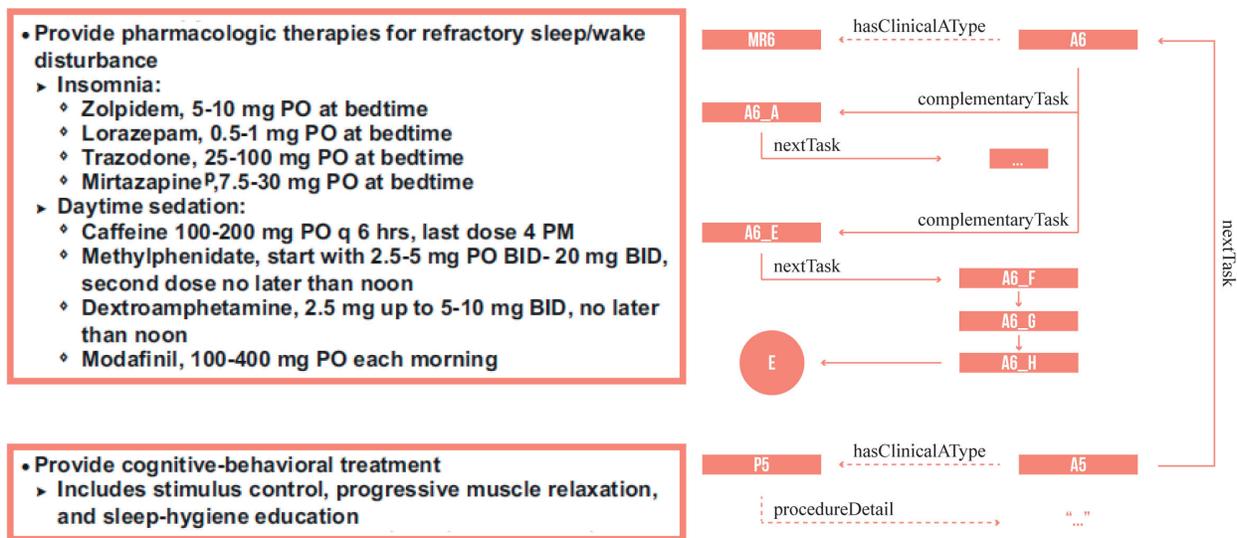


Figura 7: Exemplo de adaptação da fragmentação hierárquica de informação à ontologia.

3.2.1.4 Delírio

De entre a totalidade de cenários sintomáticos considerados, apenas neste caso é considerado como critério adicional de ponderação de escolha do método de actuação a seguir o estágio de manifestação da condição no paciente. Esta particularidade implica a existência de uma fase de pré-avaliação, à qual se segue necessariamente uma questão que pretende direccionar a actuação no sentido da sequência de acções de intervenção mais adequadas à condição específica avaliada. A restante dinâmica fez uso das valências criadas até este ponto, verificando-se apenas a necessidade de consideração de uma pré-condição (`PC_SDAgitation`) — a primeira e única utilizada no de-

correr da implementação —, originada enquanto nova instância da classe `PreCondition`, por sua vez associada a um objecto do tipo `PreConditionSet`. Por intermédio deste objecto, pretendia-se implicar que uma dada acção (`A_SD7`) apenas seria despoletada caso o paciente revelasse ser sensível a uma determinada dose de fármaco.

2.2.2 Gestão, Planeamento e Apoio

Na transição do domínio sintomático para as áreas de intervenção relacionadas com a gestão de recursos, o planeamento de cuidados e a prestação de apoio social e psicológico ao paciente, familiares e outros indivíduos envolvidos no processo de cuidados paliativos, evidenciou-se uma discrepância significativa ao nível da complexidade das sequências de acções a implementar, sendo sobretudo notórios, por um lado, a diminuição da variabilidade dos tipos de acção clínica presentes e, por outro, o aumento do número de recomendações e acções de aconselhamento relativos à consulta de entidades externas à equipa de cuidados paliativos. Por se tratar de um conjunto de secções de teor evidentemente menos pragmático, não foram sentidas quaisquer dificuldades ao nível da implementação — contrariamente ao cenário descrito até este ponto —, que se considera ter sido positivamente homogénea e transversal à totalidade dos domínios considerados. Uma vez que, nesta fase, não se revelou necessário efectuar quaisquer alterações adicionais à ontologia utilizada, procurou-se, sobretudo, retirar deste conjunto de implementações bases de comparação que permitissem construir a discussão dos resultados obtidos e, conseqüentemente, derivar ilações acerca da adequação da representação ontológica à guia clínica fornecida.

3 Discussão e Conclusões

Conforme explicitado no decorrer do presente relatório, a abordagem aplicada à implementação da guia clínica atribuída não possibilitou a sua acomodação directa na ontologia originalmente construída, muito em parte devido à existência de flutuações consideráveis ao nível das metodologias de representação da informação clínica. Assim, crê-se, numa primeira análise, que a subjectividade inerente à redacção deste tipo de documentos — quer a nível estrutural, quer semântico — constitui um sério entrave à sua modelação utilizando uma estrutura rígida e hierarquizada como a de uma ontologia. Relativamente às entidades responsáveis pelo processo de construção dos documentos, apontam-se ainda outras lacunas, nomeadamente a aparente inexistência de padrões concretos relativamente a indicações que, idealmente, se pretendem concisas e completas, em contexto clínico, como seja o caso das recomendações farmacêuticas. Urge destacar, neste ponto, que a alteração à descrição desta sub-classe foi efectuada única e exclusivamente no sentido de tornar a implementação coerente com a informação veiculada, sendo que se considera que a definição original se apresentava como clinicamente correcta e, portanto, preferível à utilizada.

Uma das primeiras vantagens que sobressai da utilização do modelo ontológico prende-se com a possibilidade de criação de sequências de acções selectivas, através da implementação de questões ou mesmo pré-condições que restringem a execução de acordo com determinados critérios. Esta particularidade agiliza significativamente o processo de consulta, uma vez que, na guia clínica física, as expressões condicionais são declaradas no interior das acções que pretendem restringir, forçando o utilizador a uma leitura extensiva da mesma. Contudo, detectou-se, neste contexto, uma situação que não se considerou passível de implementação, relacionada com a existência de momentos em que a decisão é deixada ao juízo do utilizador, não existindo quaisquer condições que ditem explicitamente ou sequer forneçam conselhos específicos acerca do método desejável de actuação. Esta situação é extensível a outros contextos e constitui, possivelmente, uma das limitações principais das ontologias clínicas: a incapacidade de lidar com a não objectividade e a abertura a juízos pessoais, muitas vezes indissociáveis da prática clínica. Porém, há que ter em linha de consideração que também o agente responsável pela adaptação do documento à ontologia afecta a respectiva construção da leitura pessoal que faz das situações representadas. Apraz então concluir, neste ponto, que ambas as representações são consideravelmente afectadas, em maior ou menor escala, pela não objectividade.

No geral, embora não seja necessariamente válido sobrepor uma forma de representação a outra, conclui-se que a fraca adopção da utilização de guias clínicas em ambientes hospitalares é justificável não só pela extensão e dispersão da informação contida nas mesmas, mas também devido ao facto de não existir uma linha de raciocínio que torne a sua aplicação imediata, sistematizada e eficaz. Contudo, destaca-se também pela negativa a mecanização inerente ao modelo ontológico, que conduz à relativização e dessubstanciação de quaisquer variações inerentes aos objectos de actuação. Naturalmente, e considerando o ideal semântico no sentido do qual se pretende evoluir, a representação de guias clínicas por intermédio de ontologias apresenta-se como desejável, pese

embora pareçam existir sérios impedimentos à obtenção de modelos dotados de universalidade.

4 Referências

- [1] W3C OWL Working Group. *OWL Web Ontology Language: Document Overview*. W3C Recommendation, 2004.
- [2] J. Heflin. An Introduction to the OWL Web Ontology Language. *Lecture Notes*, Department of Computer Science and Engineering, Lehigh University.
- [3] M. Horridge, H. Knublauch, A. Rector, R. Stevens, and C. Wroe. A Practical Guide To Building OWL Ontologies Using The Protégé-OWL Plugin and CO-ODE Tools. 27:0–117, 2004.