

Mô hình tri thức tích hợp theo sự mở rộng - thu hẹp và phương pháp suy luận giải vấn đề

Mai Trung Thành, Đỗ Văn Nhơn*
Trường Đại học Quốc tế Hồng Bàng

TÓM TẮT

Nghiên cứu về các phương pháp biểu diễn tri thức và suy luận trên máy tính có vai trò đặc biệt quan trọng trong thiết kế các hệ thống thông minh, đặc biệt là hệ giải bài toán thông minh trên tri thức hay tri thức tích hợp. Hiện nay các công trình nghiên cứu về phương pháp biểu diễn tri thức tích hợp còn nhiều hạn chế và cần phát triển, đặc biệt là chưa tập trung vào giải quyết các vấn đề tích hợp cho thiết kế các ứng dụng thông minh trong thực tế. Bài báo này sẽ trình bày một mô hình biểu diễn tri thức tích hợp theo tiếp cận mối quan hệ về mở rộng - thu hẹp giữa hai miền tri thức. Từ đó xem xét các bài toán suy luận giải quyết các vấn đề trên tri thức tích hợp. Đặc biệt là các nguyên lý thông minh trong việc suy luận giải quyết bài toán được đặt ra trên tri thức tích hợp. Bài báo cũng đã thực hiện cài đặt thử nghiệm trên miền tri thức Đại số tuyến tính. Lời giải của hệ thống được hiển thị và trình bày từng bước, tự nhiên như cách con người thường sử dụng trong việc giảng dạy và học tập trong giáo dục.

Từ khóa: Ontology COKB, phương pháp tích hợp tri thức, phương pháp suy luận, cơ sở tri thức tích hợp, hệ giải bài toán thông minh

1. ĐẶT VẤN ĐỀ

Trong khoa học về trí tuệ nhân tạo, biểu diễn tri thức và suy luận có vai trò quan trọng, quyết định trong quá trình thiết kế, xây dựng các hệ chuyên gia, các hệ giải bài toán thông minh dựa trên tri thức, đặc biệt là các hệ thống giải quyết vấn đề trên tri thức tích hợp [1, 2]. Việc xây dựng được các hệ thống có khả năng suy diễn để giải các vấn đề được đặt ra trên tri thức tích hợp, đòi hỏi các kỹ sư tri thức không những phải xây dựng các cơ sở tri phù hợp, các phương pháp tích hợp tri thức này một cách hiệu quả, mà còn phải thiết kế một động cơ suy diễn mô phỏng cách tư duy của con người trong việc giải quyết các vấn đề trên tri thức [3].

Vấn đề tích hợp tri thức đã được nhiều nhà nghiên cứu quan tâm [1], cũng như tầm quan trọng trong phát triển việc thiết kế các hệ thống thông minh hiện đại đòi hỏi cần sử dụng tích hợp tri thức [2]. Cùng với sự phát triển của nhiều ontologies, việc nghiên cứu về giải pháp tích hợp tri thức đến nay vẫn luôn một trong những thách thức lớn [4, 5]. Nhìn chung, các công trình hiện nay chỉ tập trung vào các ontologies đồng dạng về cấu trúc. Chưa quan tâm đến việc tích hợp các mô hình ontologies có khác nhau về mặt cấu trúc thành phần, đặc biệt là trên các khái niệm có cấu trúc trừu tượng. Việc nghiên cứu các phương pháp cho thiết kế hệ thống có khả năng suy luận giải vấn đề được đặt ra trên cơ sở tri thức [6]. Ta có thể điểm qua một số công trình liên quan như sau.

Từ các thách thức đặt ra trong việc tích hợp các ontologies [4] trên nền tảng ngôn ngữ đặc tả OWL, nhóm tác giả Inès OsMan trong công trình [7] đã đưa ra các kỹ thuật trong việc tích hợp ontology bao gồm như: Sự căn chỉnh (alignment), tái cấu trúc (refactoring), tích hợp (integration), trộn (merge). Kết quả của bài đã được triển khai vận dụng vào tích hợp cho các ontologies gồm LargeBio ontology: FMA (ontology về kiến thức giải phẫu trên con người), NCI (ontology về kiến thức Gene) and SNOMED-CT (ontology về kiến thức lâm sàng). Tuy nhiên, kết quả nghiên cứu của nhóm tác giả Inès OsMan chỉ quan tâm đến các vấn đề tích hợp trên ontologies có thành phần tri thức với cấu trúc khái đơn giản, cùng các vấn đề truy vấn và tìm kiếm trên cơ sở tri thức tích hợp, chưa quan tâm đến các khái niệm có cấu trúc trừu tượng và suy luận giải vấn đề trên cơ sở tri thức tích hợp.

Trên các ontologies có cấu trúc tương tự được xem xét bởi nhóm tác giả Inès OsMan, nhóm tác giả Hoàng Hữu Hạnh trong [8], đã có những nghiên cứu liên quan đến bài toán tích hợp ontologies. Kết quả đã đưa ra phương pháp đối sánh các ontologies về độ tương đồng ngữ nghĩa của các thực thể trong ánh xạ (mapping) các ontology. Mô hình được đề xuất được tác giả đánh giá cao về các độ bao phủ (recall) và độ chính xác (precision). Tuy nhiên, cũng như trong [4, 7], nhóm tác giả cũng chưa xét đến thành phần tri thức khái niệm có cấu trúc trừu

Tác giả liên hệ: Đỗ Văn Nhơn

Email: nhondv@hiu.vn

tượng, các thành phần tri thức phổ biến khác như tri thức toán tử, hàm, luật, sự kiện và các phương pháp suy luận giải vấn đề trên tri thức tích hợp.

Nhóm tác giả Nguyễn Ngọc Thành [2], Trương Hải Bằng [9] đề xuất phương pháp tiếp cận dựa trên lý thuyết đồng thuận [2], trong tích hợp các ontologies có dạng (C, R, I, Z) . Phương pháp này định nghĩa ontology mờ bao gồm mức các thuộc tính, các khái niệm và các quan hệ của chúng cùng với các khía cạnh không chắc chắn tiềm năng trong việc thể hiện những tri thức về chúng. Tuy nhiên, cũng như các công trình [7, 8], nhóm tác giả chưa quan tâm đến các khái niệm có cấu trúc trừu tượng, cũng như các tri thức phổ biến khác trong các trong ứng dụng thực tế như: Toán tử, hàm, luật, sự kiện và chưa đặt ra các vấn đề suy diễn tự động trên cơ sở tri thức tích hợp.

Trong công trình [10] đã trình bày một phương pháp phối hợp các cơ sở tri thức có dạng ontology COKB cho thiết kế hệ giải bài tập tự động trên miền tri thức Đại số tuyến tính. Với các mảnh tri thức con gồm Ma trận - Định thức, Hệ phương trình tuyến tính và Không gian vector. Việc phối hợp tri thức dựa vào các tri thức luật cho phép chuyển đổi trạng thái giữa các cơ sở tri thức trong việc suy luận giải các vấn đề trên miền tri thức phối hợp. Tuy nhiên, trong công trình này cũng còn hạn chế cần phải được phát triển và nghiên cứu. Đặc biệt là chưa làm rõ và tổng quát được các vấn đề về mối quan hệ giữa các cơ sở tri thức, cụ thể là quan hệ về sự mở rộng - sự thu hẹp. Cũng như chưa xét đến các vấn đề suy diễn một cách hiệu quả trên các miền tri thức mở rộng - thu hẹp.

Trong [11, 12] trình bày một cách tiếp cận tích hợp dựa trên sự bổ sung thành phần (toán tử, hàm) vào mô hình tri thức lõi (kernel ontology), từ đó bổ sung thêm các sự kiện, cùng các vấn đề trên trên mô hình tri thức được tích hợp theo kiểu bổ sung thêm thành phần mới. Kết quả nghiên cứu đã vận dụng vào thiết kế các hệ thống giải một số dạng bài tập trên miền tri thức như trong lý thuyết đồ thị, toán rời rạc. Tuy nhiên, trong các công trình này cũng chưa đề cập đến các vấn đề tổng thể trên tri thức tích hợp, cũng như chưa đề cập đến các cơ chế suy diễn phù hợp được đặt ra trên các cơ sở tri thức lõi, tri thức tích hợp.

Mô hình tri thức về các đối tượng tính toán, gọi tắt là mô hình tri thức COKB (Computational Objects Knowledge Base) [13, 14] là một ontology cho biểu diễn tri thức gồm có 6 thành phần $(C, H, R, Ops, Funcs, Rules)$. Trong đó C là tập hệ thống các khái niệm có sự phân cấp, H là tập các quan hệ đặc biệt

hóa giữa các khái niệm trên C , R là tập các quan hệ khác trên C , Ops là tập các toán tử, $Funcs$ là tập các hàm, $Rules$ là tập các luật dẫn. Đi theo thành phần $Rules$, xét trên mô hình COKB ta có 12 loại sự kiện. COKB là mô hình tỏ ra hiệu quả cho việc thiết kế các hệ giải vấn đề thông minh dựa trên tri thức. Sức mạnh của ontology COKB nằm ở khả năng biểu diễn được đa dạng thành phần. Đồng thời trên COKB được trang bị các lớp bài toán tổng quát, cùng các phương pháp và kỹ thuật suy diễn hiệu quả giải các lớp bài toán tổng quát trên mô hình. Khai thác sức mạnh đó, trên nền tảng tri thức COKB, bài báo này sẽ trình bày một phương pháp tích hợp dựa trên xét mối quan hệ về sự mở rộng - thu hẹp thành phần khái niệm xét trong ngữ cảnh của mô hình COKB. Cụ thể là trên cơ sở tri thức về Ma trận - Định thức - Hệ phương trình tuyến tính, tri thức Không gian vector sẽ được hình thành thông qua việc mở rộng - thu hẹp từ cấu trúc của khái niệm trên miền tri thức có trước (trong phạm vi kiến thức về Đại số tuyến tính [15, 16]). Từ đó làm cơ sở đưa ra mô hình tri thức tích hợp và các lớp bài toán, cùng các phương pháp suy diễn hiệu quả giải các lớp bài toán được đặt ra trên các cơ sở tri thức, cơ sở tri thức tích hợp.

2. PHƯƠNG PHÁP BIỂU DIỄN TRI THỨC TÍCH HỢP CHO THIẾT KẾ CƠ SỞ TRI THỨC TOÁN ĐẠI SỐ TUYẾN TÍNH

2.1. Phương pháp biểu diễn tri thức cho miền tri thức về Ma trận - Định thức - Hệ phương trình tuyến tính

Định nghĩa 2.1 Cho K_1 là miền tri thức về Ma trận - Định thức - Hệ phương trình tuyến tính. Ta gọi mô hình biểu diễn cơ sở tri thức K_1 là mô hình có dạng COKB, gồm một bộ các thành phần sau đây:

$$(C_1, H_1, R_1, Ops_1, Funcs_1, Rules_1)$$

Trong đó:

- C_1 là một tập hợp hệ thống khái niệm trong miền tri thức, mỗi khái niệm $c \in C_1$ là một lớp các đối tượng tính toán. Ta gọi một đối tượng tính toán $o \in c$ trên mô hình COKB, có đặc trưng sau.

+ Một đối tượng tính toán gồm có 3 thành phần $(Attrs, Facts, Rules)$. $Attrs$ là tập các thuộc tính bên trong đối tượng o , mỗi thuộc tính của đối tượng o là một đối tượng tính toán. $Facts$ là tập các tính chất hay sự kiện vốn có của đối tượng. $Rules$ là thành tập các luật nội tại của đối tượng, trên đối tượng tính toán o ta có hai loại sau đây R_f và R_r . R_f là tập các quan hệ dạng đẳng thức, $R_f = \{f_1, f_2, \dots, f_n\}$. R_r là tập các quan hệ dạng luật dẫn, $R_r = \{r_1, r_2, \dots, r_m\}$.

+ Một số lớp bài toán trên một đối tượng tính toán

như sau: *Xác định bao đóng tập sự kiện; Xác định một sự kiện; Tính giá trị của một đối tượng hay thuộc tính của đối tượng; Kiểm tra sự giải được của bài toán GT $\rightarrow$ KL; Tìm lời giải cho bài toán GT $\rightarrow$ KL.* Mỗi lớp bài toán trên đối tượng được trang bị các thuật giải và các phương pháp suy luận giải quyết các bài toán này, các vấn đề này cũng đã được trình bày [13, 14].

Trên cấu trúc của đối tượng, ta có các khái niệm trong C_1 được phân cấp sau đây:

+ $C_{1(0)} = \{c \mid \text{dom}(c) \in \mathbb{R} \cup \text{Names} \cup \text{Exprs}\}$, $\text{dom}(c)$ miền giá trị của khái niệm c , Names là tập các không gian tên của miền tri thức, Exprs là tập các biểu thức có giá trị hằng.

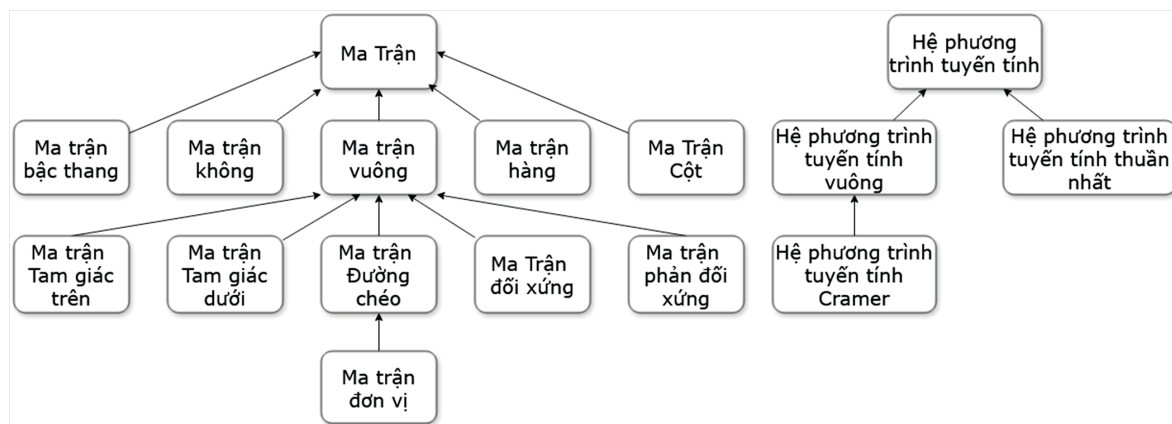
+ $C_{1(k)} = \{c \mid (\forall x \in \text{Attrs}(c), \exists i < k, \exists c' \in C_{1(i)}: x \in c') \wedge (\exists x \in \text{Attrs}(c), \exists c' \in C_{1(k-1)}: x \in c')\}, k > 0$.

Ví dụ 2.1: Khái niệm Ma trận (MATRAN) là một khái niệm chỉ một lớp các đối tượng tính toán thuộc $C_{1(1)}$ (cấp 1), ta có cấu trúc Ma trận như sau: $\text{Attrs} = \{m, n, (a_{ij})_{m \times n}\}$; // m, n là các khái niệm cấp 0 ($m, n \in \mathbb{R}$); $(a_{ij})_{m \times n}$ là một tập các phần tử được biểu diễn dưới dạng bảng (mảng m dòng, n cột), mỗi phần tử a_{ij} là một khái niệm cấp 0 ($a_{ij} \in \text{Exprs}$); $\text{Facts} = \emptyset$; $\text{Rf} = \emptyset$;

$\text{Rr} = \{r1: \{\forall i, j, 1 \leq i \leq m, 1 \leq j \leq n: a_{ij} = 0\} \rightarrow \{this::\text{MATRANKHONG}\},$
 $r2: \{m = 1\} \rightarrow \{this::\text{MATRANDONG}\},$
 $r3: \{n = 1\} \rightarrow \{this::\text{MATRANCOT}\}, \dots\}$

Ví dụ 2.2: Khái niệm Ma trận vuông (MATRANVUONG) là một ma trận (cấp 1), ta có thể mô hình hóa cấu trúc của ma trận vuông từ mối quan hệ "là" như sau: $\text{Attrs} = \text{MATRAN}.\text{Attrs} \cup \{\text{inv}, \text{diag}, \text{det}\}$; $\text{Facts} = \text{MATRAN}.\text{Facts} \cup \{m = n\}$; $\text{Rf} = \text{MATRAN}.\text{Rf} \cup \emptyset$; $\text{Rr} = \text{MATRAN}.\text{Rr} \cup \{$
 $r1: \{\forall i, j, 1 \leq j < i \leq n: a_{ij} = 0\} \rightarrow \{this::\text{MATRANTAMGIACTREN}\},$
 $r2: \{\forall i, j, 1 \leq i < j \leq n: a_{ij} = 0\} \rightarrow \{this::\text{MATRANTAMGIACDUOI}\},$
 $r3: \{\forall i, j, 1 \leq i, j \leq n, i \neq j: a_{ij} = 0\} \rightarrow \{this::\text{MATRANDUONGCHEO}\},$
 $r4: \{\forall i, j, 1 \leq i, j \leq n, i \neq j: a_{ij} = 0, a_{ii} = 1\} \rightarrow \{this::\text{MATRANDONVI}\}, \dots\}$

- H_1 là tập các quan hệ đặc biệt hóa giữa các khái niệm trong tập C_1 , ta có các quan hệ mà theo đó có thể có những khái niệm là sự đặc biệt hoá của những khái niệm khác. Ví dụ: Các quan hệ phân cấp giữa các khái niệm như hình dưới đây.



Hình 1. Sơ đồ quan hệ đặc biệt hóa giữa các khái niệm trên miền tri thức K_1

- R_1 là tập các quan hệ hai ngôi khác trên các khái niệm C_1 . Mỗi quan hệ được xác định bởi <tên quan hệ> và các loại đối tượng có mối quan hệ. Ví dụ: Giữa các khái niệm ma trận, hệ phương trình tuyến tính ta có một số quan hệ như sau: *Quan hệ bằng nhau giữa hai ma trận, quan hệ tương đương dòng giữa hai ma trận, quan hệ tương đương giữa hai ma trận, quan hệ tương đương giữa hai hệ phương trình tuyến tính.*

- Ops_1 là tập các toán tử trên các khái niệm C_1 , $\text{Ops}_1 = \text{Ops}_1^1 \cup \text{Ops}_1^2$. Với Ops_1^1 là tập phép toán một ngôi, như *phép chuyển vị trên ma trận A^T , phép nghịch đảo ma trận A^{-1}* ; Ops_1^2 là tập các phép toán hai ngôi như: *Phép toán cộng hai ma trận, phép toán trừ trên ma trận, phép toán nhân hai ma trận, phép toán nhân giữa hệ số k ($k \in \mathbb{R}$) và một ma trận.*

- Funcs_1 là tập tri thức hàm trong miền tri thức. Mỗi hàm được xác định bởi <tên hàm> và các loại đối tượng của hàm. Hàm có một số tính chất sau đây như tính chất phản xạ, tính chất đối xứng, *tính chất phản xứng* và *tính chất bắc cầu*. Như hàm *rank* (hạng của ma trận), *det* (bậc của ma trận), *các hàm biến đổi sơ cấp trên ma trận* (hoán vị dòng, nhân dòng với một hệ số k , thay thế dòng).

- Rules_1 là tập các luật dạng luật dẫn trên miền tri thức K_1 . Các sự kiện trên các luật dẫn được xem xét gồm 12 loại sự kiện cùng các thuật giải xử lý trong việc hợp nhất sự kiện theo [13, 14].

2.2. Mô hình biểu diễn tri thức về Không gian Vector

Định nghĩa 2.2: Cho K_1 là mô hình biểu diễn cơ sở tri

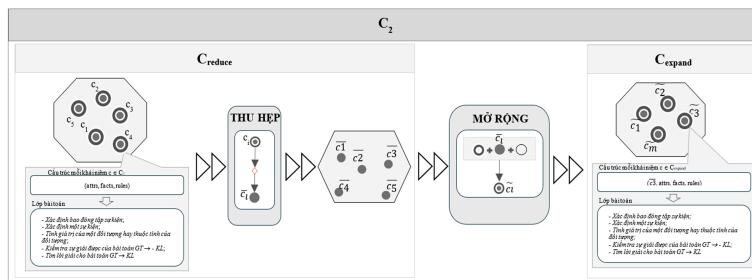
thức theo định nghĩa 2.1, ta gọi K_2 là *miền tri thức được mở rộng* từ tri thức K_1 . Mô hình cho biểu diễn tri thức về không gian vector là mô hình gồm các thành phần sau đây:

$$(C_2, H_2, R_2, Ops_2, Funcs_2, Rules_2)$$

Trong đó:

- C_2 là tập các khái niệm trên miền tri thức K_2 . Gồm các *khái niệm giản lược* và tập các *khái niệm mở rộng*, $C_2 = C_{\text{reduce}} \cup C_{\text{expand}}$, trong đó:
 - + C_{reduce} là tập các khái niệm được *giản lược* từ tập khái niệm trong C_1 , $C_{\text{reduce}} = \{\bar{C}_1, \bar{C}_2, \dots, \bar{C}_n\}$, mỗi *khái niệm giản lược* $\bar{C}_i \in C_{\text{reduce}}$ chỉ lớp *đối tượng giản lược* từ các đối tượng thuộc lớp khái niệm $c_i \in C_1$ (trong miền tri thức K_1). Mỗi *đối tượng giản lược* $\bar{O} \in \bar{C}_i$ có cấu trúc là gồm tập thuộc tính, ký hiệu $\bar{O}.Attrs = O.Attrs$. Ví dụ: Ta có khái niệm "*Tập thể hiện của các ma trận vuông cấp n*", ký hiệu $lc_{\text{MATRANVUONGCAPN}}$, ta có $lc_{\text{MATRANVUONGCAPN}} = \{O_1, O_2, \dots, O_m\}$, $\forall O_j \in lc_{\text{MATRANVUONGCAPN}}$ có cấu trúc $O_j.Attrs = O_j.Attrs = \{m, n, (a_{ij})_{m \times n}, inv, diag, det\}$, với $O_j \in C_j$ ($C_j \in \text{MATRANVUONG cấp } n$);
 - + C_{expand} là tập các khái niệm được *mở rộng* từ các

khái niệm trong C_{reduce} , $C_{\text{expand}} = \{\tilde{C}_1, \tilde{C}_2, \dots, \tilde{C}_n\}$, với mỗi khái niệm $\tilde{C}_j \in C_{\text{expand}}$ chỉ lớp *một lớp các đối tượng mở rộng* từ tập các đối tượng trong $\bar{C}_j \in C_{\text{reduce}}$. Mỗi *đối tượng mở rộng* $\tilde{O} \in \tilde{C}_j$ ($\tilde{C}_j \in C_{\text{expand}}$) có cấu trúc gồm $(\tilde{C}_j, Attrs, Facts, Rules)$. Trong đó $\tilde{C}_j \in C_{\text{reduce}}$, $\tilde{C}_j = \{O_1, O_2, \dots, O_m\}$; $Attrs$ là tập thuộc tính của đối tượng $\tilde{O}$; $Facts$ là tập không rỗng các tính chất nội tại của đối tượng $\tilde{O}$; $Rules$ là tập các quy tắc/tính chất được trang bị nội tại trong đối tượng $\tilde{O}$. Bên cạnh cấu trúc, mỗi đối tượng cũng được bị các lớp bài toán cùng các thuật giải suy luận nội tại bên trong đối tượng tương tự như một lớp đối tượng tính toán theo COKB. Ví dụ: Khái niệm về "*không gian các ma trận vuông cấp n*", ta có cấu trúc một đối tượng thuộc lớp khái niệm không gian các ma trận vuông cấp n như sau: $lc_{\text{MATRANVUONGCAPN}} = \{O_1, O_2, \dots, O_m\}$; $Attrs = \{dim, rank, L\}$; // dim là số chiều, $rank$ là hạng của "*không gian các ma trận vuông cấp n*" ($dim, rank \in \mathbb{R}, L \subseteq lc_{\text{MATRANVUONGCAPN}}$); $Facts = \{u + v = v + u, u + (v + w) = (u + v) + w, k * (u + v) = k * u + k * v, (k + g) = k * u + g * u\}$, ($k * g = k * (g * u)$), " $u, v \in L, k \in \mathbb{R}; Rules = \emptyset$."



Hình 2. Minh họa cấu trúc khái niệm C_2

- H_2 là tập các quan hệ đặc biệt hóa giữa các khái niệm trong tập C_2 .
- R_2 là tập các quan hệ hai ngôi khác giữa các khái niệm trên C_2 . Ta có các quan hệ giữa hai không gian vector, như: *không gian con*, *cơ sở*, *độc lập tuyến tính*, *phụ thuộc tuyến tính*, *bao tuyến tính*, *hệ sinh*.
- Ops_2 là tập các toán tử, $Ops_2 = Ops_{\text{Vector}} \cup Ops_{\text{VectorSpace}}$. Trong đó Ops_{Vector} là tập các phép toán trên vector, $Ops_{\text{VectorSpace}}$ là tập các phép toán trên hai họ vector trong cùng không gian vector. Ví dụ: Cho V không gian các ma trận vuông cấp n , hai họ vector S, W trong không gian vector V , ta có các phép toán như giao ($\cap$), hợp ($\cup$), hiệu ($\setminus$), tổng tuyến tính (+).
- $Funcs_2$ là tập các hàm. Ta có $Funcs_2 = \{MaTranToaDo, ToaDo\}$
- $Rules_2$ là tập các luật của tri thức luật. Ta có, $Rules_2 = \{$
 - r1: $\{S: \text{HOVECTOR}, V: \text{KGVT}, [“COSO”, S, V]\} \rightarrow \{V.dim = |S|\}$,

- r2: $\{S: \text{HOVECTOR}, V: \text{KGVT}, |S| = V.dim, [“DLTT”, S, V]\} \rightarrow \{[“COSO”, V, S], \sum_{i=1}^{|S|} a_i * e_i = 0\}$,
- r3: $\{S: \text{HOVECTOR}, V: \text{KGVT}, u: \text{VECTOR}, [“THUOC”, u, V]\} \rightarrow \{u = \sum_{i=1}^{|S|} a_i * e_i\}$,
- r4: $\{S: \text{HOVECTOR}, V: \text{KGVT}, u: \text{VECTOR}, [“THUOC”, u, V], [“THTT”, u, S]\} \rightarrow \{u = \sum_{i=1}^{|S|} a_i * e_i\}$,
- r5: $\{S: \text{HOVECTOR}, V: \text{KGVT}, |S| > V.dim\} \rightarrow [“PTTT”, S, V]$
- r6: $\{S: \text{HOVECTOR}, V: \text{KGVT}, u: \text{VECTOR}, u = 0, [“THUOC”, u, S]\} \rightarrow [“PTTT”, S, V]$
- r7: $\{S: \text{HOVECTOR}, V: \text{KGVT}, \exists! u \neq 0 \mid [“THUOC”, u, S]\} \rightarrow [“DLTT”, S, V]$,
- r8: $\{S: \text{HOVECTOR}, V: \text{KGVT}, |S| > 1, [“THUOC”, u, S], u = \sum_{i=1}^{|S|} a_i * e_i\} \rightarrow [“PTTT”, S, V]$,
- r9: $\{S: \text{HOVECTOR}, W: \text{HOVECTOR}, V: \text{KGVT}, [“PTTT”, W, V], W \subseteq S\} \rightarrow [“PTTT”, S, V]$,
- r10: $\{S: \text{HOVECTOR}, W: \text{HOVECTOR}, V: \text{KGVT}, [“DLTT”, S, V], W \subseteq S\} \rightarrow [“DLTT”, W, V], ..\}$

2.3. Mô hình tri thức tích hợp cho thiết kế cơ sở tri thức Đại số tuyến tính

Định nghĩa 2.3: Ta gọi K là miền tri thức về đại số tuyến tính, tri thức K là mô hình được tích hợp từ hai miền tri thức K_1, K_2 (K_2 là tri thức được mở rộng từ K_1). Mô hình tri thức K là một bộ gồm 4 thành phần như sau.

$$(K_1, K_2, R, \text{Rules})$$

Trong đó:

- K_1 là mô hình cho cơ sở tri thức theo định nghĩa 2.1.
- K_2 là mô hình cơ sở tri thức được mở rộng từ cơ sở tri thức K_1 theo định nghĩa 2.2.
- R là tập các quan hệ hai ngôi giữa các khái niệm trong C_1 và các khái niệm trong C_2 . C_1 là thành phần khái niệm trong K_1 , C_2 là thành phần tri thức trong K_2 .
- Rules là tập tri thức luật trên mô hình tri thức tích hợp. Trong đó $\text{Rules} = \text{Rules}_B \cup \text{Rules}_H$. Với Rules_B là tập các tri thức luật gồm các tính chất, định lý, các tiền đề có trên miền tri thức tích hợp; Rules_H là tập các tri thức luật gồm các quy tắc heuristics.

3. CÁC LỚP BÀI TOÁN TRÊN MÔ HÌNH TRI THỨC TÍCH HỢP VÀ PHƯƠNG PHÁP SUY DIỄN GIẢI QUYẾT VẤN ĐỀ

3.1. Mô hình bài toán

Định nghĩa 3.1: Cho miền tri thức tích hợp K được biểu diễn theo mô hình tri thức tích hợp (định nghĩa 2.3). Ta gọi mô hình bài toán trên mô hình tri thức tích hợp K có dạng sau đây:

$$H \rightarrow G$$

Trong đó:

- H là tập sự kiện giả thiết. H gồm bộ (O, F). Trong đó (O, F) mô hình có dạng mạng các đối tượng tính toán trên mô hình COKB [13, 14], O là tập các đối tượng, F là tập các sự kiện.
- G là tập các sự kiện mục tiêu. Trên mô hình tri thức COKB, ta xét G gồm hai loại mục tiêu: *Tính toán giá trị của một thuộc tính hoặc hàm; Chứng minh sự xác định một sự kiện thuộc 12 loại sự kiện.*

Ví dụ 3.1: Cho hệ phương trình tuyến tính, tính nghiệm hệ phương trình sau:

$$\begin{cases} x - 3y - z = 15 \\ x + 4y + z = 16 \\ 2x - 13y - 6z = -71 \\ 3x + 5y + z = 17 \end{cases}$$

Ta có thể đặc tả bài toán trên theo yêu cầu như sau:

O := {"HEPHUONGTRINH TUYENTINH", E}; F := {E.Exprs[1] := $x - 3y - z = 15$, E.Exprs[2] := $x + 4y + z = 16$, E.Exprs[3] := $2x - 13y - 6z = -71$, E.Exprs[4] := $3x + 5y + z = 17$ }; G := {"TINH", E.Nghiem}.

Ví dụ 3.2: Trong không gian vector V trên trường số thực R^3 cho họ vector $S = \{u_1 = (-1, 5, 2), u_2 = (0, 3, -7), u_3 = (0, 0, 8)\}$, chứng minh rằng S là cơ sở của V.

Ta có thể đặc tả bài toán trên theo yêu cầu như sau: O := {"KGVT", V}, {"HOVECTOR", S}; F := {"THUOC", V.K, REAL, V.dim = 3, S.L[1] = [-1, 5, 2], S.L[2] = [0, 3, -7], S.L[3] = [0, 0, 8]} G := {"CHUNG_MINH", {"COSO", S, V}}.

3.2. Quy tắc suy luận

Việc giải quyết các vấn đề trên cơ sở tri thức tích hợp K, cũng như K_1 hay K_2 là việc mở rộng sự kiện đã biết (tập sự kiện H) dựa trên các quy tắc suy luận [14] để thay đổi giả thiết H (trạng thái) ban đầu dần đến trạng thái mà có chứa mục tiêu G của bài toán. Trên tri thức tích hợp, một quy tắc suy luận sẽ được bổ sung thêm các loại quy tắc có khả năng sinh ra sự kiện mới từ các tri thức K_1 hay K_2 .

Định nghĩa 3.2: Tập các quy tắc suy luận RCN (rule of reasoning on computational objects networks) được trang bị cho thiết kế phương pháp suy luận giải quyết vấn đề trên K gồm:

- (RCN1) Loại quy tắc "Tự động sinh ra các sự kiện mặc nhiên từ tập sự kiện đã biết trước"
- (RCN2) Loại quy tắc "Sinh ra sự kiện mới dựa trên việc áp dụng một luật dẫn"
- (RCN3) Loại quy tắc "Sinh ra sự kiện mới dựa trên hành vi của một đối tượng"
- (RCN4) Loại quy tắc "Sinh ra sự kiện mới dựa trên việc sử dụng các tính chất, hay thực hiện các thủ tục của một hàm"
- (RCN5) Loại quy tắc "Sinh ra sự kiện mới dựa trên việc giải một hệ phương trình được thành lập bằng cách kết hợp nhiều dạng quan hệ tính toán"
- (RCN6) Loại quy tắc "Sinh ra đối tượng mới dựa trên việc áp dụng một luật dẫn có phát sinh đối tượng".
- (RCN7) Loại quy tắc "Sinh ra sự kiện mới dựa trên việc sử dụng cơ sở tri thức K_1 "
- (RCN8) Loại quy tắc "Sinh ra sự kiện mới dựa trên việc sử dụng cơ sở tri thức K_2 "

3.3. Bước giải và lời giải

Định nghĩa 3.3. Xét bài toán $H \rightarrow G$ trên mô hình tri thức tích hợp K, ta có:

- Một bước giải trên cơ sở tri thức tích hợp K là một bộ ba (rf, kf, nf). rf là một quy tắc suy luận $rf \in \text{RCN}$; kf là tập các sự kiện cần để có thể áp dụng để áp dụng được rf; nf là tập sự kiện mới được sinh ra dựa vào rf và các sự kiện kf.

- Danh sách các bước giải $S = [s_1, s_2, s_3, \dots, s_m]$, ta ký hiệu $H_0 = H, H_1 = s_1(H_0), H_2 = s_2(H_1), H_3 = s_3(H_2), \dots$, và $S(GT) = H_m$. Ta nói bài toán $H \rightarrow G$ là giải được khi và chỉ khi $G \subseteq S(H)$, và ta nói S là một *lời giải* của bài toán.

- S được gọi là *lời giải* không có bước giải thừa khi và chỉ khi không tồn tại $S' \subset S$ sao cho $G \subseteq S'(H)$. Từ S ta có thể tìm được một lời giải không có chứa bước thừa với thuật giải 3.1 sau đây.

Thuật giải 3.1: Thuật giải tìm lời giải không có bước thừa từ một lời giải S của bài toán $H \rightarrow G$.

```
// Đầu vào: S, G: một danh sách các bước giải, G tập mục tiêu.
// Đầu ra: S': một lời giải không có bước giải thừa.
1. NewVar := G;
2. S' := [];
3. for i from |S| down to 1 do
4.     if nf(S[i]) ⊆ NewVar then
5.         NewVars := NewVars - nf(S[i]);
6.         NewVars := NewVars ∪ kf(S[i]);
7.         S' := [op(S'), S[i]];
8.     end if;
9. end do;
10. S' là một lời giải không có bước thừa;
```

Hình 3. Thuật giải 3.1

3.4. Phương pháp suy diễn giải quyết bài toán trên mô hình tri thức tích hợp K

Việc suy diễn giải quyết bài toán $H \rightarrow G$ trên tri thức tích hợp K được thiết kế dựa trên các chiến lược suy diễn tiến, suy diễn lùi cùng bằng cách sử dụng *loại quy tắc suy luận* để sinh ra các sự kiện mới, cùng các quy tắc heuristics phù hợp với cách suy luận của con người trong việc giải quyết vấn đề. Tri thức K là tri thức được tích hợp từ K_1 và K_2 . Vì vậy, việc thiết kế thuật toán suy diễn giải bài toán trên K, ta cần phải có các phương pháp suy diễn cho việc giải bài toán trên miền tri thức K_1 , tri thức K_2 (được mở rộng từ K_1). Chi tiết được trình bày dưới đây.

3.4.1. Phương pháp suy diễn giải quyết bài toán trên cơ sở tri thức K_1

Cơ sở tri thức K_1 được mô hình hóa theo cơ sở tri thức COKB, quá trình suy diễn giải vấn đề $H \rightarrow G$ là quá trình suy diễn tiến bằng cách sử dụng các *loại quy tắc suy luận* trên K_1 để sinh ra các sự kiện mới từ giả thiết H, quá trình này sẽ được lặp cho đến khi trạng thái các sự kiện đạt được chứa các sự kiện mục tiêu trong G. Quá trình này được diễn đạt thông qua thuật giải 3.2 dưới đây.

Thuật giải 3.2: Thuật giải tìm lời giải cho bài toán H G, trên mô hình tri thức K_1

Cho trước: Cơ sở tri thức K_1

Đầu vào: Giả thiết H, mục tiêu G

Đầu ra: Lời giải cho bài toán H G

Bước 1: Khởi tạo các biến

KnownFacts := H;

Sol := [];

Bước 2: Thực hiện suy luận tìm lời giải cho bài toán H
WHILE $G \not\subseteq \text{KnownFacts}$ DO

(1) *newf* là sự kiện được sinh tự động - RCN1

Nếu *newf* được sinh ra mới

KnownFacts := KnownFacts $\cup$ *newf*;

Sol := [op(Sol), ["Từ loại quy tắc RCN1", [{}, {kfacts}, {newf}]]];

(2) Tìm một luật dẫn rr để sử dụng - RCN2

Nếu tìm rr thì

KnownFacts := KnownFacts $\cup$ gr(rr);

Sol := [op(Sol), ["Từ loại quy tắc RCN2", [{rr}, {hr(rr)}, {gr(rr)}]]];

(3) Tìm một đối tượng o để sử dụng - RCN3

Nếu tìm được đối tượng o thì

ApDungMotDoiTuong(o);

(4) Tìm một tri thức hàm func để sử dụng - RCN4

Nếu tìm được func thì

ApDungMotTriThucHam(func);

(5) Tìm một hệ phương trình equals dựa trên kết hợp các sự kiện loại 5 -

Nếu tìm được equals then

newsfact := set_vars(equals) - KnownFacts;

kfacts := set_vars(equals) $\cap$ KnownFacts;

varnews := solve(equals, newsfacts);

KnownFacts := KnownFacts $\cup$ varnews;

Sol := [op(Sol), ["Tu RCN5", [{equals}, {kfacts}, {varnews}]]];

(6) Tìm một luật dẫn r có khả năng sinh ra đối tượng mới - RCN6

Nếu tìm được r then

KnownFacts := KnownFacts $\cup$ gr(r);

Sol := [op(Sol), ["Tu RCN6", [{r}, {hr(r)}, {gr(r)}]]];

END DO

Đánh giá thuật giải 3.2: Giả sử rằng ta gọi $T_1(H \rightarrow G)$ là ký hiệu độ phức tạp của thuật giải 3.2 với đầu vào bài toán $H \rightarrow G$ trên cơ sở tri thức K_1 . Thuật giải 3.2 được thiết kế dựa trên các phương pháp suy luận trên nền tảng mô hình COKB [13, 14], vì vậy độ phức tạp của thuật giải 3.2 đã được phân tích và đề cập trong báo cáo [14]. Khi đó ta có độ phức tạp của thuật giải 3.2 $T_1(H \rightarrow G) \approx O(\alpha)$, trong đó $\alpha = \max(|\text{CFS}|^4, |\text{CFS}| * |\text{Rules}|, |\text{CFS}| * (|\text{F2}| + \max(|\text{maxFS}(f2)|^{n+3}, |\text{maxFS}(f2)|^n * |\text{maxRr}(f2)|, |\text{maxFS}(f2)|^n * |\text{maxRf}(f2)|^2)), |\text{CFS}| * |\text{Ops}|, |\text{CFS}| * |\text{Funs}|, |\text{CFS}| * |\text{DR}|)$ (theo [14, tr.49]).

Từ ví dụ 3.1, ta có một lời giải cho của bài toán khi áp dụng thuật giải 3.2 được trình bày như dưới đây:

Bước 1:
 Từ ["HEPHUONGTRINH TUYENTINH", E],
 $E.Exprs[1] := x - 3y - z = 15,$
 $E.Exprs[2] := x + 4y + z = 16,$
 $E.Exprs[3] := 2x - 13y - 6z = -71,$
 $E.Exprs[4] := 3x + 5y + z = 17$
 Ta được: $E.\bar{A} = \left(\begin{array}{ccc|c} 1 & -3 & -1 & -15 \\ 2 & 4 & 1 & 16 \\ 3 & 5 & 1 & 17 \end{array} \right)$
 Áp dụng quy tắc tự động sinh sự kiện (RCN1)

Bước 2:
 Từ: $E.\bar{A} = \left(\begin{array}{ccc|c} 1 & -3 & -1 & -15 \\ 2 & 4 & 1 & 16 \\ 3 & 5 & 1 & 17 \end{array} \right)$
 Ta được: $A_{temp} = \left(\begin{array}{ccc|c} 1 & -3 & -1 & -15 \\ 2 & -13 & -6 & -71 \\ 3 & 5 & 1 & 17 \end{array} \right)$
 ["TuongDuong", $E.\bar{A}$, A_{temp}]
 Áp dụng quy tắc quy tắc biến đổi sơ cấp trên dòng, $h2 - h1 \rightarrow h3$ (RCN4)

Bước 3:
 Từ: $A_{temp} = \left(\begin{array}{ccc|c} 1 & -3 & -1 & -15 \\ 2 & -13 & -6 & -71 \\ 3 & 5 & 1 & 17 \end{array} \right)$
 Ta được: $A_{temp} = \left(\begin{array}{ccc|c} 1 & -3 & -1 & -15 \\ 0 & 7 & 2 & 31 \\ 0 & -7 & -4 & -41 \end{array} \right)$
 Áp dụng quy tắc quy tắc biến đổi sơ cấp trên dòng, $h3 - 2 \cdot h1 \rightarrow h3$ (RCN4)

Bước 4:
 Từ: $A_{temp} = \left(\begin{array}{ccc|c} 1 & -3 & -1 & -15 \\ 0 & 7 & 2 & 31 \\ 0 & -7 & -4 & -41 \end{array} \right)$
 Ta được: $A_{temp} = \left(\begin{array}{ccc|c} 1 & -3 & -1 & -15 \\ 0 & 7 & 2 & 31 \\ 0 & 0 & 14 & 62 \end{array} \right)$
 Áp dụng quy tắc quy tắc biến đổi sơ cấp trên dòng, $h4 - 3 \cdot h1 \rightarrow h4$ (RCN4)

Bước 5:
 Từ: $A_{temp} = \left(\begin{array}{ccc|c} 1 & -3 & -1 & -15 \\ 0 & 7 & 2 & 31 \\ 0 & 0 & 14 & 62 \end{array} \right)$
 Ta được: $A_{temp} = \left(\begin{array}{ccc|c} 1 & -3 & -1 & -15 \\ 0 & 7 & 2 & 31 \\ 0 & 0 & -2 & -10 \end{array} \right)$
 Áp dụng quy tắc quy tắc biến đổi sơ cấp trên dòng, $h3 + h2 \rightarrow h3$ (RCN4)

Bước 6:
 Từ: $A_{temp} = \left(\begin{array}{ccc|c} 1 & -3 & -1 & -15 \\ 0 & 7 & 2 & 31 \\ 0 & 0 & -2 & -10 \end{array} \right)$
 Ta được:
 ["HEPHUONGTRINH TUYENTINH", E_{new}],
 $E_{new}.Exprs[1] := x - 3y - z = 15,$
 $E_{new}.Exprs[2] := 7y + 2z = 31,$
 $E_{new}.Exprs[3] := -2z = -10,$
 ["TuongDuong", $E_{new}.\bar{A}$, A_{temp}]
 Áp dụng luật sinh ra đối tượng mới (RCN6)

Bước 7:
 Từ: ["TuongDuong", $E.\bar{A}$, A_{temp}],
 ["TuongDuong", $E_{new}.\bar{A}$, A_{temp}]
 Ta được: ["TuongDuong", E , E_{new}],
 ["TuongDuong", $E.Exprs$, $E_{new}.Exprs$]
 Áp dụng quy tắc sinh sự kiện mới dựa trên sử dụng luật (RCN2)

Bước 8:
 Từ: $E_{new}.Exprs[1] := x - 3y - z = 15,$
 $E_{new}.Exprs[2] := 7y + 2z = 31,$
 $E_{new}.Exprs[3] := -2z = -10$
 Ta được: $E_{new}.Nghiem := \{ \{x = -1, y = 3, z = 5\} \}$
 Áp dụng loại quy tắc giải hệ phương trình (RCN5)

Bước 9:
 Từ: ["TuongDuong", $E.Exprs$, $E_{new}.Exprs$],
 $E_{new}.Nghiem := \{ \{x = -1, y = 3, z = 5\} \}$
 Ta được: $E.Nghiem := \{ \{x = -1, y = 3, z = 5\} \}$
 Áp dụng loại quy tắc sinh sự kiện tự động (RCN1)

Hình 4. Minh họa lời giải cho bài toán tại ví dụ 3.1
sẽ được tìm thấy bởi thuật giải 3.2

3.4.2. Phương pháp suy diễn giải quyết bài toán trên mô hình tri thức K2

Cơ sở tri thức K_2 được mở rộng từ K_1 , vì vậy quá trình suy luận giải bài toán trên tri thức K_2 ngoài sử dụng các loại quy tắc suy luận thì cần sử dụng phần cơ sở tri thức con trên miền K_1 .

Định nghĩa 3.4: Ta gọi một cơ sở tri thức K_{sub} được mô hình hóa theo COKB [13, 14]. Với K_{sub} gồm bộ $(C_{sub}, H_{sub}, R_{sub}, Ops_{sub}, Funcs_{sub}, Rules_{sub})$. Ta nói $K_{sub} \subseteq K_1$ hay K_2 (K_{sub} là phần cơ sở tri thức con của K_1 (hay K_2) khi và chỉ khi:

- $C_{sub} \subseteq C_1$ (hay C_2)
- H_{sub} là tập các quan hệ phân cấp trên tập khái niệm C_{sub}
- R_{sub} là tập các quan hệ khác trên tập khái niệm C_{sub}
- $Ops_{sub} = \{op_1, op_2, \dots, op_m\}, op_i (i = 0..m) : C_{sub} \times C_{sub} \rightarrow C_{sub}$ hoặc $C_{sub} \rightarrow C_{sub}$

$$- Funs_{sub} = \{f_1, f_2, \dots, f_n\}, \forall f_i (i = 0..n) : C_{sub}^u \rightarrow C_{sub}^v$$

$$- Rules_{sub} \subseteq Rules_1 \text{ (hay } Rules_2)$$

Định nghĩa 3.5: Trên miền tri thức K_1 cho một khái niệm $c \in C_1$ (hay C_2).

Ta gọi K_{csub} là phần cơ sở tri thức con chứa khái niệm c của K_1 (hay K_2) khi và chỉ khi:

- $C_{csub} = \{c' \mid c' \in C_{sub} : c' r_{sub} c \vee c' = c\}, r_{sub} \in R_{sub}$
- H_{csub} là tập các quan hệ phân cấp trên tập C_{csub}
- R_{csub} là tập các quan hệ khác trên tập khái niệm C_{csub}
- $Ops_{csub} = \{op_1, op_2, \dots, op_m\}, \forall op_i (i = 0..m) : C_{csub} \times C_{csub} \rightarrow C_{csub}$ hoặc $C_{csub} \rightarrow C_{csub}$
- $Fun_{csub} = \{f_1, f_2, \dots, f_n\}, \forall f_i (i = 0..n) : C_{csub}^u \rightarrow C_{csub}^v$
- $Rules_{csub} = \{r \mid r \in Rules_{sub} : \exists c \in h(r) \text{ or } g(r)\}$

Ta gọi tập khái niệm $C' \subseteq C_1$ (hay C_2). Ta gọi $K_{C'sub}$ là

phần cơ sở tri thức con chứa tập khái niệm C' của K_1 (hay K_2). Khi đó các thành phần trong $K_{C'sub}$ được xác định như sau:

- $C'_{sub} = U_{C'} C_{sub}$
- $H'_{sub} = U_{C'} H_{sub}$
- $R'_{sub} = U_{C'} R_{sub}$
- $Ops'_{sub} = U_{C'} Ops_{sub}$
- $Func'_{sub} = U_{C'} Func_{sub}$
- $Rules'_{sub} = U_{C'} Rules_{sub}$

Trên cơ sở tri thức K_2 bài toán $H \rightarrow G$ có thể được giải quyết được dựa trên chiến lược suy diễn tiến kết hợp với nguyên lý vét cạn các tri thức trên cơ sở tri thức K_1 sẽ được khai thác trong việc giải vấn đề trên K_2 . Chi tiết được trình bày ở thuật giải 3.3 dưới đây.

Thuật giải 3.3: Thuật giải tìm lời giải cho bài toán $H \rightarrow G$, trên mô hình tri thức K_2 .

Cho trước: Cơ sở tri thức K_1

Đầu vào: Giả thiết H , mục tiêu G

Đầu ra: Lời giải cho bài toán $H \rightarrow G$

Bước 1: Khởi tạo các biến

KnownFacts:= H ;

Sol:=[];

Bước 2: Thực hiện suy luận tìm lời giải cho bài toán H

WHILE $G \notin \text{KnownFacts}$ DO

(1) *newf* là sự kiện được sinh tự động - RCN1

Nếu *newf* được sinh ra mới

KnownFacts:=KnownFacts $\cup$ *newf*;

Sol:= $[op(Sol), ["\text{Từ loại quy tắc RCN1"}], \{ \}, \{kfacts\}, \{newf\}]]$;

(2) Tìm một luật dẫn rr để sử dụng - RCN2

Nếu tìm rr thì

KnownFacts:=KnownFacts $\cup$ $gr(rr)$;

Sol:= $[op(Sol), ["\text{Từ loại quy tắc RCN2"}], \{rr\}, \{hr(rr)\}, \{gr(rr)\}]]$;

(3) Tìm một đối tượng o để sử dụng

Nếu tìm được đối tượng o và o thuộc khái

niệm c' trong cơ sở tri thức K_1 - RCN7

+ Gọi K_{sub} là phần tri thức con chứa khái niệm c' ;

+ Thực hiện suy luận giải vấn đề $H \rightarrow G$, trên tri thức K_{sub} ;

+ Giả sử *newfacts* là tập sự kiện mới sinh ra được từ việc sử dụng K_{sub} ;

++ KnownFacts:=KnownFacts $\cup$ {*newfacts*};

++ Sol:= $[op(Sol), ["\text{Từ loại quy tắc RCN7"}], \{K_{sub}\}, \{o\}, \{newfacts\}]]$;

Nếu tìm được đối tượng o và o trong cơ sở tri thức K_2 - RCN3

+ *ApDungMotDoiTuong(o)*;

(4) Tìm một tri thức hàm func để sử dụng - RCN4

Nếu tìm được *func* thì

ApDungMotTriThucHam(func);

(5) Tìm một hệ phương trình equals dựa trên kết hợp các sự kiện loại 5 -

Nếu tìm được equals then

newsfact:= set_vars(equals) - KnownFacts;

kfacts:= set_vars(equals) $\cap$ KnownFacts;

varnews:=solve(equals, newsfacts);

KnownFacts:=KnownFacts $\cup$ varnews;

Sol:= $[op(Sol), ["\text{Từ loại quy tắc RCN5"}], \{equals\}, \{kfacts\}, \{varnews\}]]$;

(6) Tìm một luật dẫn r có khả năng sinh ra đối tượng mới - RCN6

Nếu tìm được r then

KnownFacts:=KnownFacts $\cup$ $gr(r)$;

Sol:= $[op(Sol), ["\text{Từ loại quy tắc RCN6"}], \{r\}, \{hr(r)\}, \{gr(r)\}]]$;

END DO

Đánh giá thuật giải 3.3: Giả sử rằng ta gọi $T_2(H \rightarrow G)$ là ký hiệu độ phức tạp của thuật giải 3.3 với đầu vào bài toán $H \rightarrow G$ trên cơ sở tri thức K_2 . Dễ dàng thấy rằng, thuật giải 3.3 cũng được thiết kế dựa trên quá trình suy diễn tổng quát giải bài toán trên mô hình COKB, vì vậy độ phức tạp của T_2 có hai trường hợp:

- Trường hợp khi không có sử dụng vét cạn: $T_2(H \rightarrow G) \approx O(\alpha^2)$

- Trường hợp khi sử dụng chiến lược vét cạn thông minh: $T_2(H \rightarrow G) \approx O(\alpha * \beta)$, $1 \leq \beta \leq \alpha$. Trong đó: $O(\beta)$

là độ phức tạp của $T_1(H \rightarrow G)$ trên tri thức K_{sub} .

3.4.3. Phương pháp suy diễn giải quyết bài toán trên mô hình tri thức tích hợp K

Tương tự quá trình giải bài toán trên K là một quá trình suy diễn tiến để sinh ra sự kiện mới dựa trên các quy tắc, luật suy diễn trên cơ sở tri thức K. Đồng thời quá trình giải quyết vấn đề trên miền K_1 và K_2 cũng sẽ tham gia vào quá trình giải bài toán $H \rightarrow G$ trên miền tri thức K. Quá trình này sẽ được cụ thể hóa bằng thuật giải 3.4 dưới đây.

Thuật giải 3.4: Thuật giải tìm lời giải cho bài toán $H \rightarrow G$, trên mô hình tri thức K.

Cho trước: Cơ sở tri thức K gồm K_1, K_2

Đầu vào: Giả thiết H , mục tiêu G

Đầu ra: Lời giải cho bài toán $H \rightarrow G$

Bước 1: Khởi tạo các biến

KnownFacts:= H ;

Sol:=[];

Bước 2: Thực hiện suy luận tìm lời giải cho bài toán H

WHILE $G \notin \text{KnownFacts}$ DO

(1) *newf* là sự kiện được sinh tự động - RCN1

Nếu *newf* được sinh ra mới

KnownFacts:=KnownFacts $\cup$ newf;
Sol:=[op(Sol), ["Từ loại quy tắc RCN1", [{},
{kfacts}, {newf}]]];

(2) Tìm một luật dẫn rr để sử dụng - RCN2

Nếu tìm rr thì

KnownFacts:=KnownFacts $\cup$ gr(rr);
Sol:=[op(Sol), ["Từ loại quy tắc RCN2", [{rr},
{hr(rr)}, {gr(rr)}]]];

(3) Gọi K_{1Csub} là "phần tri thức con chứa tập khái niệm" trên miền tri thức K_1 .

Sử dụng miền tri thức K_{1Csub} để sinh ra sự kiện mới
- RCN7 (áp dụng thuật giải 3.2)

(4) Gọi K_{2Csub} là "phần tri thức con chứa tập khái niệm" trên miền tri thức K_2 .

Sử dụng miền tri thức K_{2Csub} để sinh ra sự kiện mới
- RCN8 (áp dụng thuật giải 3.3)

END DO

Đánh giá thuật giải 3.4: Giả sử rằng ta gọi $T_3(H \rightarrow G)$ là ký hiệu độ phức tạp của thuật giải 3.4 với đầu vào bài toán $H \rightarrow G$ trên cơ sở tri thức K (gồm: K_1, K_2). Dựa trên độ phức tạp của thuật giải 3.2 và thuật giải 3.3, độ phức tạp của T_3 có hai trường hợp:

- Trường hợp khi không có sử dụng vết cạn: $T_3(H \rightarrow G) \approx n * (\max(|Facts|, |Rules|, O(\alpha^2)))$

- Trường hợp khi sử dụng chiến lược vết cạn thông minh: $T_3(H \rightarrow G) \approx n * (\max(|Facts|, |Rules|, O(\beta_1 * \beta_2)))$. Trong đó: $O(\beta_1)$ và $O(\beta_2)$ là độ phức tạp của thuật toán 3.2, 3.3.

Với: $|Facts|$ là độ lớn của tập bước phát sinh sự kiện tự động; $|Rules|$ là độ lớn của tập luật trên cơ sở tri thức K.

Từ ví dụ 3.2, ta có một lời giải cho của bài toán p khi áp dụng thuật giải 3.4 được trình bày như dưới đây.

Bước 1: Từ: S.L[1] = [-1, 5, 2], S.L[2] = [0, 3, -7], S.L[3] = [0, 0, 8] Ta được: S.L.length = 3 Áp dụng quy tắc tự động sinh sự kiện (RCN1) Bước 2: Từ: S.L.length = 3, V.dim = 3 Ta được: S.L.length = V.dim Áp dụng quy tắc tự động sinh sự kiện (RCN1) Bước 3: Từ: S.L[1] = [-1, 5, 2], S.L[2] = [0, 3, -7], S.L[3] = [0, 0, 8] Ta được: ["MATRAN", A_{new}], $A_{new} = \begin{pmatrix} -1 & 5 & 2 \\ 0 & 3 & 7 \\ 0 & 0 & 8 \end{pmatrix}$ Áp dụng luật sinh ra đối tượng mới (RCN6)	Bước 4: Từ: $A_{new} = \begin{pmatrix} -1 & 5 & 2 \\ 0 & 3 & 7 \\ 0 & 0 & 8 \end{pmatrix}$ Ta được: $\det(A) = -24$ Áp dụng tri thức hàm tính định thức trong cơ sở tri thức K_1 (RCN7) Bước 5: Từ: $\det(A_{new}) = -24$, ["HOVECTOR", S], ["KGV", V] Ta được: ["DLTT", S, V] Áp dụng tri thức luật trong cơ sở tri thức K_2 (RCN8) Bước 6: Từ: ["DLTT", S, V], S.L.length = V.dim Ta được: ["COSO", S, V] Áp dụng tri thức luật trong cơ sở tri thức K_2 (RCN8)
--	---

Hình 5. Minh họa lời giải cho bài toán tại ví dụ 3.2
sẽ được tìm thấy bởi thuật giải 3.4

4. KẾT QUẢ THỬ NGHIỆM VÀ BÀN LUẬN

4.1. Kết quả thử nghiệm

Hệ thống cũng đã thử nghiệm giải một số bài tập trong giáo trình Toán cao cấp tập 1 trong sách [15],

và một số bài tập trong giáo trình [16]. Với lời giải cho ra tự nhiên như cách dạy và học của con người trong môi trường giáo dục. Bảng dưới đây là một số kết quả thử nghiệm của hệ thống.

Bảng 1. Bảng kết quả thử nghiệm của kết quả nghiên cứu

STT	LỚP BÀI TOÁN	YÊU CẦU	TỔNG SỐ
1	Tính toán	Tính định thức ma trận	26
		Tính hạng ma trận	3
		Giải phương trình ma trận	6
		Tìm ma trận nghịch đảo	11
		Tìm nghiệm hệ phương trình tuyến tính/ giải hệ phương trình	48
		Tìm tọa độ vector theo cơ sở	4
		Tổng và giao không gian con	5

STT	LỚP BÀI TOÁN	YÊU CẦU	TỔNG SỐ
2	Chứng minh	Chứng minh họ vector là độc lập tuyến tính/phụ thuộc tuyến tính	25
		Chứng minh họ vector là cơ sở của một không gian vector	30

4.2. So sánh đánh giá và bàn luận

Kết quả nghiên cứu và thử nghiệm đã giải được tự động một số lớp bài toán tính toán, chứng minh. Tuy nhiên, đề tài chưa xem xét đến các lớp bài toán có đề cập đến tham số. Vì vậy chưa thể giải được một số bài tập có yêu cầu biện luận theo tham số.

So với các kết quả nghiên cứu trước đây, đề tài đã đưa ra được một dạng tích hợp tri thức theo mối quan hệ về sự mở rộng - thu hẹp cùng các kỹ thuật và phương pháp suy luận liên quan. Khi suy luận giải vấn đề tổng quát giải bài toán, thuật giải 3.4 đã thể hiện được sự hiệu quả khi đã có sự giới hạn không gian suy luận trên các miền tri thức con (áp dụng nguyên lý vét cạn). Điều này đã giúp giảm được độ phức tạp trong việc suy luận tìm kiếm lời giải cho bài toán, đặc biệt là kỹ thuật này phù hợp và tương tự cách con người thường sử dụng trong việc suy luận giải vấn đề.

Mô hình tri thức COKB là một trong những mô hình mạnh mẽ cho thiết kế các cơ sở tri thức có thành phần đa dạng và cấu trúc khái niệm trừu tượng. Bên cạnh đó, mối quan hệ về mở rộng tri thức, thu hẹp tri thức suy luận giải vấn đề là một trong những tư duy tự nhiên của con người trong thực tiễn. Vì vậy, nghiên cứu việc tích hợp các cơ sở tri thức dạng COKB dựa trên mối quan hệ sự mở rộng - thu hẹp là vô cùng tiềm năng trong việc vận dụng vào các miền tri thức khác. Đặc biệt là các miền tri thức mà mô hình COKB đã chứng minh sự khả thi như hình học phẳng, hình học không gian, hình học giải tích 2 chiều, 3 chiều, đại số tuyến tính, một số miền tri thức trong vật lý, hóa học, v.v.

5. KẾT LUẬN

Dựa trên cơ sở tri thức COKB, bài báo đã đưa ra

được một phương pháp tích hợp tri thức theo sự mở rộng - thu hẹp tri thức. Cụ thể là xem xét các trường hợp thu hẹp - mở rộng cấu trúc của khái niệm. Mô hình được xây dựng cho phép biểu diễn được miền tri thức đại số tuyến tính và giải quyết các vấn đề được đặt ra trên cơ sở tri thức có mối quan hệ mở rộng - thu hẹp. Đặc biệt là các chiến lược suy diễn dựa trên các nguyên lý thông minh của con người trong quá trình suy luận giải vấn đề thực tế. Kết quả nghiên cứu của bài báo cũng đã cài đặt thử nghiệm thành công hệ hỗ trợ giải bài tập tự động trên miền tri thức Đại số tuyến tính. Hệ thống cho ra lời giải tự nhiên, phù hợp với ngữ cảnh dạy và học trong lĩnh vực giáo dục.

Bài báo chỉ xem xét việc tích hợp tri thức dựa trên việc thu hẹp - mở rộng thành phần tri thức khái niệm, chưa xem xét đến các thành phần tri thức khác trên mô hình COKB. Bên cạnh đó, việc dựa vào mô hình tri thức COKB cũng còn khá cồng kềnh. Chưa quan tâm đến việc đưa ra các mô hình tinh gọn có tính linh hoạt để từ đó có thể module hóa các tri thức theo dạng từng thành phần. Công việc tiếp theo của bài báo đó chính là nghiên cứu và phát triển một số dạng mô hình tri thức tích hợp theo dạng từng phần. Nghĩa là cần phải đưa ra được các dạng ontology lõi (core) cùng với các giải pháp tích hợp mang tính tổng quát trên các ontology lõi. Từ đó làm cơ sở cho việc phát triển các công cụ hỗ trợ cho việc phát triển các hệ quản trị cơ sở tri thức trên máy tính.

LỜI CẢM ƠN

Nghiên cứu này được Trường Đại học Quốc tế Hồng Bàng cấp kinh phí thực hiện dưới mã số đề tài GVTC18.03.

TÀI LIỆU THAM KHẢO

[1] Pinto HS, Perez AG, Martins JP, "Some issues on ontology integration," in *16th International Joint Conference on Artificial Intelligence (IJCAI'99) Workshop: KRR5: Ontologies and Problem-Solving Methods: Lesson Learned and Future Trends*, Stockholm, 1999.

[2] N. T. Nguyen, "Advanced Information and Knowledge Processing," *Springer*, 2008.

[3] Đ. V. Nhơn, *Các hệ cơ sở tri thức*, Đại học Quốc Gia TP. HCM, 2021.

[4] Inès Osman, "Ontology Integration:

Approaches and Challenging Issues," *Information Fusion*, vol. 71, pp 38-63, 2021.

[5] Euzenat J, Portisch J, Hladik M, Paulheim H., "Background knowledge in ontology matching: A survey," *Semantic Web*, vol. 15, no. 6, pp. 2639 - 2693, 2024.

[6] Đ. V. Nhơn, "Các hệ thống trí tuệ nhân tạo ứng dụng trong giáo dục," *Tạp Chí Khoa học Trường Đại học Quốc tế Hồng Bàng, Số chuyên đề (2023)*, pp. 11-22, 2023.

[7] Inès Osman, Salvatore Flavio Pileggi, Sadok Ben Yahia, Gayo Diallo, "An Alignment-Based Implementation of a Holistic Ontology Integration Method," in *MethodsX*, Elsevier, 2021.

[8] H. H. Hạnh, H. N. Phát, P. C. Vinh, "Ontology, tiếp cận mới về đối sánh," in *Kỷ yếu Hội nghị Quốc gia lần thứ VIII về Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR)*, 2019.

[9] T. H. Bằng, *Phương pháp tích hợp Ontology*, Đại học Quốc Gia TP. HCM, 2021.

[10] Đ. V. Nhơn, N. Đ. Hiền, P.T. Vương, "Xây dựng hệ hỗ trợ giải toán đại số tuyến tính trên cơ sở tri thức gồm các miền tri thức phối hợp," *Tạp chí Khoa học Trường Đại học Cần Thơ, Số p.h CD Công nghệ TT*, pp. 10-18, 2017.

[11] Hien D. N., L. Nguyen, Nha P. T., Van-Thanh N.L., Sang V., "Design an Intelligent Problem Solver in Mathematics based on Integrated-Knowledge Model," in *International Conference on Knowledge and Systems Engineering (KSE)*, Bangkok, Thailand, 2021, pp. 1-6, DOI: 10.1109/KSE53942.2021.9648716.

[12] Nhon V. D., Hien D. N., Thanh T.M., "A Method of Ontology Integration for Designing Intelligent Problem Solvers," *Journal of Applied Sciences*, vol. 9, no. 18, p. 3793, 2019, DOI: [10.3390/app9183793](https://doi.org/10.3390/app9183793).

[13] N. Do, Thanh T. M. "Perfect COKB Model and Reasoning Methods for the design of Intelligent Problem Solvers," in *International Conference on Intelligent Software Methodologies, Tools, and Techniques*, Japan, 2017, pp. 35-50, DOI: 10.3233/978-1-61499-800-6-35.

[14] M. T. Thành, "Phương pháp suy luận giải quyết vấn đề trên mô hình COKB," Luận văn thạc sĩ, Trường Đại học Công nghệ Thông tin - Đại học Quốc gia TP. Hồ Chí Minh, TP. Hồ Chí Minh, 2017, link luận văn: <http://ir.vnulib.edu.vn/handle/VNUHCM/9582>.

[15] N. Đ. Trí, T. V. Dũng, T. X. Hiền, N. X. Thảo, *Toán cao cấp*, Nhà xuất bản Giáo dục Việt Nam, 2016.

[16] C. T. Tình, H. M. Linh, L. H. Tuấn, L. H. M. Vân, *Đại số tuyến tính*, Đại học Quốc Gia TP. Hồ Chí Minh, 2019.

An integrated knowledge model based on the expansion - reduction approach and a reasoning method for problem solving

Mai Trung Thanh, Do Van Nhon

ABSTRACT

Research on methods of knowledge representation and reasoning in computing plays a particularly important role in the design of intelligent systems and knowledge-based problem-solving systems, especially those that require the use of multiple knowledge domains. Currently, studies on integrated knowledge representation methods remain limited and need further development, particularly in addressing integration challenges for designing real-world intelligent applications. This paper presents an integrated knowledge representation model based on the relationship between reduction and expansion approaches. The problems and the reasoning methods for solving them, based on the integrated knowledge base, are considered and presented in this paper. In particular, the paper explores intelligent exhaustive search principles for inference based on integrated knowledge. An experimental

implementation has been conducted in the domain of Linear Algebra. The system's solutions resemble the way humans commonly teach and learn in educational contexts.

Keywords: *Ontology COKB, integrated knowledge methods, reasoning methods, integrated knowledge base, intelligent problem solvers*

Received: 25/6/2025

Revised: 28/8/2025

Accepted for publication: 11/9/2025