

DOI: <https://doi.org/10.59294/HIUJS.KHTT.2026.051>

PHƯƠNG PHÁP TÍCH HỢP MÔ HÌNH RELA-MODEL VÀ OPS-MODEL CÙNG CÁC PHƯƠNG PHÁP SUY LUẬN GIẢI VẤN ĐỀ TRÊN TRI THỨC TÍCH HỢP

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

TÓM TẮT

Mô hình tri thức quan hệ (Rela-Model) và mô hình tri thức Toán tử (Ops-Model) là hai mô hình khá hiệu quả cho việc biểu diễn các miền tri thức có các thành phần khái niệm, quan hệ, toán tử và luật. Sự hiệu quả này đã được minh chứng qua nhiều kết quả nghiên cứu sử dụng chúng trong thiết kế các hệ giải vấn đề thông minh. Tuy nhiên, hai mô hình Rela-Model và Ops-Model đang được nghiên cứu độc lập, việc nghiên cứu mô hình và xây dựng tri thức tích hợp từ tri thức quan hệ và tri thức toán tử là một trong những vấn đề nghiên cứu có ý nghĩa trong thực tiễn. Bài báo này sẽ trình bày một phương pháp tích hợp giữa mô hình tri thức quan hệ (Rela-Model) và mô hình tri thức toán tử (Ops-Model) cùng các vấn đề xây dựng tri thức tích hợp. Từ cơ sở đó, các bài toán trên tri thức tích hợp và các phương pháp suy luận giải vấn đề trên cơ sở tri thức tích hợp cũng sẽ được trình bày. Kết quả nghiên cứu cũng đã được vận dụng vào việc cài đặt, thử nghiệm trên miền tri thức hình học giải tích hai chiều. Kết quả của hệ thống cho ra lời giải tự nhiên, từng bước, có khả năng giải thích, phù hợp với môi trường dạy và học trong lĩnh vực giáo dục.

Từ khóa: bản thể học, mô hình tri thức tích hợp, mô hình tri thức quan hệ, mô hình tri thức toán tử, hệ giải vấn đề thông minh

AN INTEGRATED APPROACH OF RELA-MODEL AND OPS-MODEL WITH REASONING METHODS FOR PROBLEM SOLVING ON THE INTEGRATED KNOWLEDGE BASE

Do Van Nhon, Mai Trung Thanh

ABSTRACT

The Relational Knowledge Model (Rela-Model) and the Operator Knowledge Model (Ops-Model) are two effective models for representing knowledge domains that consist of concepts, relations, operators, and rules. Their effectiveness has been demonstrated through numerous research results in which they have been applied to the design of intelligent problem-solving systems. However, the Rela-Model and Ops-Model have been studied independently. Therefore, investigating a model and constructing integrated knowledge from relational knowledge and operator knowledge is a practically meaningful research problem. This paper presents an integration approach between the Relational Knowledge Model (Rela-Model) and the Operator Knowledge Model (Ops-Model), together with issues related to the construction of integrated knowledge. Based on this foundation, problems defined on integrated knowledge and reasoning methods for problem solving on the integrated knowledge base are also presented. The proposed research results have been applied to

* Tác giả liên hệ: Mai Trung Thành, Email: thanhmt@hiu.vn
(Ngày nhận bài: 05/4/2026; Ngày nhận bản sửa: 16/4/2026; Ngày duyệt đăng: 22/4/2026)

the implementation and experimentation in the domain of two-dimensional analytic geometry. Experimental results show that the system can generate natural, step-by-step, and explainable solutions, which are suitable for teaching and learning environments in education.

Keywords: *ontology, intelligent problem solvers, integrated knowledge base, rela-model, ops-model*

1. TỔNG QUAN NGHIÊN CỨU

Tích hợp tri thức có vai trò đặc biệt quan trọng trong thiết kế các hệ thống thông minh lớn và hiện đại [1 - 4]. Tích hợp tri thức là việc khai thác hay tái sử dụng các tri thức sẵn có để hình thành lên cơ sở tri thức mới [5], từ đó làm cơ sở cho việc phát triển các hệ thống có khả năng giải quyết các vấn đề thông minh cần sử dụng hay tích hợp từ nhiều miền tri thức khác nhau. Vấn đề về tích hợp tri thức đã được Thomas R. Gruber [1] đặt ra và được nhiều nhà nghiên cứu quan tâm [5 - 9], và cho đến nay chúng vẫn là một trong những thách thức lớn trong khoa học [2 - 4].

Nghiên cứu các phương pháp tích hợp tri thức không chỉ giúp cho việc phát triển các cơ sở tri thức mới một cách thuận tiện, dễ dàng [1] mà còn là cơ sở trong việc phát triển các hệ thống có khả năng suy luận giải vấn đề được đặt ra trên miền tri thức tích hợp. Một hệ thống có khả năng suy luận giải vấn đề trên miền tri thức tích hợp là một hệ thống không chỉ giải được các vấn đề trên miền tri thức đơn, mà còn có thể giải được các vấn đề được đặt ra trên tri thức tích hợp. Không những thế, quá trình suy luận giải vấn đề cũng phải được hoạt động một cách linh hoạt và hiệu quả.

Tri thức quan hệ và tri thức toán tử là những thành phần tri thức phổ biến trong thực tiễn, đặc biệt là trong lĩnh vực giáo dục đại học. Việc thiết kế một hệ thống thông minh vừa có khả năng giải vấn đề đặt ra trên tri thức quan hệ, vừa có khả năng giải vấn đề đặt ra trên tri thức toán tử, đặc biệt là có khả năng giải vấn đề cần sự tích hợp giữa hai miền tri thức này một cách linh hoạt và hiệu quả, là rất cần thiết và có ý nghĩa trong giáo dục.

Hiện nay có nhiều phương pháp biểu diễn tri thức, phương pháp tích hợp tri thức cho thiết kế các loại hệ thống có khả năng giải vấn đề đặt ra trên cơ sở tri thức. Ta có thể điếm qua một số công trình nghiên cứu về các phương pháp tích hợp tri thức hiện nay như sau.

Inès OsMan và cộng sự [10] đã trình bày các phương pháp tích hợp ontology một cách khá toàn diện (Holistic Ontology Integration Method). Bài báo 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). Tương tự công trình trong [11], nhóm tác giả đã có những nghiên cứu liên quan đến bài toán tích hợp ontologies. Kết quả đã đưa ra phương pháp đối sánh ontology về độ tương đồng ngữ nghĩa của các thực thể trong ánh xạ (mapping) các ontologies. Mô hình được đề xuất được tác giả đánh giá cao về các độ bao phủ (recall) và độ chính xác (precision). Tuy nhiên, các công trình này chỉ xem xét các dạng ontology được biểu diễn bằng Logic mô tả, và được biểu diễn dưới ngôn ngữ cho biểu diễn ngữ nghĩa web (OWL). Chúng không xét các vấn đề tích hợp cho thiết kế các hệ giải vấn đề thông minh.

Trong công trình [5, 12] nhóm tác giả đã đề xuất phương pháp tích hợp các ontology mờ 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. Từ các khái niệm và các quan hệ giữa các khái niệm, nhóm tác giả đã đưa ra các thuật toán tích hợp dựa trên lý thuyết đồng thuận, cho ra kết quả là các khái niệm và các quan hệ của các phần tử thể hiện tốt nhất giữa các ontologies. Tuy nhiên cũng như các công trình [4, 10] không đề cập đến biểu việc biểu diễn các khái niệm có cấu trúc trừu tượng và không xét các vấn đề tích hợp cho thiết kế hệ giải vấn đề suy luận trên tri thức tích hợp.

Trong công trình [13] nhóm tác giả đã trình phương pháp phối hợp đơn giản 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, hệ phương trình tuyến tính và không gian vector. Một thành phần

được xây dựng với mục tiêu tạo kết nối giữa những miền tri thức có dạng COKB, cho phép nhận diện và chuyển đổi trạng thái qua lại giữa các tri thức ma trận, hệ phương trình tuyến tính và không gian vector. Một trong những điểm yếu của giải pháp này đó chính là tích hợp các cơ sở tri thức dạng COKB, mô hình COKB khá công kềnh khi vận dụng vào một số miền tri thức có số thành phần tri thức ít đa dạng như mô hình COKB đề xuất. Điều này làm giảm đi sự hiệu quả và khả năng linh hoạt được đòi hỏi của việc tích hợp. Tương tự, trong các công trình [14 - 18] các nhóm tác giả đã trình bày các phương pháp tích hợp bằng cách kết hợp cơ học giữa các thành phần trong mô hình biểu diễn tri thức. Nghĩa là chúng chưa có một cơ chế tích hợp đúng nghĩa từ các cơ sở tri thức đã có trước, đồng thời các phương pháp suy luận trên từng miền tri thức, tri thức tích hợp cũng đã bị bỏ ngỏ.

Trong công trình [19] nhóm tác giả đã đề cập đến phương pháp tích hợp giữa mô hình tri thức quan hệ (Rela-Model) và mô hình tri thức toán tử (Ops-Model). Các cơ sở lý thuyết, mô hình tổng hợp gồm các thành phần trong Rela-Model và trong Ops-Model được kết hợp cơ học thành một mô hình tri thức tích hợp. Cũng tương tự như [13], các vấn đề về việc xây dựng tri thức mới, các phương pháp suy luận trên cơ sở tri thức tích hợp cũng không được xem xét. Điều này làm mất đi tính linh hoạt, mềm dẻo cần có của một động cơ suy luận trên tri thức tích hợp.

Nhìn chung các phương pháp tích hợp hiện nay còn nhiều hạn chế, đặc biệt là các phương pháp hướng đến thiết kế các hệ thống thông minh giải vấn đề trên tri thức tích hợp một cách mềm dẻo và linh hoạt. Trong đó, việc tích hợp tri thức không chỉ đơn thuần là sự phối hợp một cách đơn như [13, 19] hay sự bổ sung cơ học như trong [14, 20] mà tích hợp tri thức còn phải đảm bảo được hiệu quả. Nghĩa là trong quá trình xây dựng tri thức tích hợp từ các tri thức có sẵn, cần có sự chọn lựa, quy tắc, kỹ thuật để hình thành một tri thức tích hợp (tri thức mới).

Trong bài báo này sẽ trình bày một mô hình tri thức tích hợp giữa tri thức quan hệ Rela-Model và tri thức toán tử Ops-Model. Trên cơ sở đó, các vấn đề như xây dựng tri thức tích hợp, các bài toán suy diễn trên tri thức tích hợp cũng sẽ được trình bày chi tiết trong bài báo này. Kết quả của bài báo cũng đã được vận dụng thử nghiệm trên miền tri thức hình học giải tích hai chiều (được tích hợp từ kiến thức hình học phẳng và kiến thức đại số). Lời giải của hệ thống cho ra một cách từng bước, tự nhiên, có thể giải thích được quá trình suy luận.

2. PHƯƠNG PHÁP TÍCH HỢP RELA-MODEL VÀ OPS-MODEL

2.1. Mối quan hệ tri thức

Một lớp các đối tượng chỉ xét quan mối quan hệ [21] có các đặc trưng gồm:

- Cấu trúc gồm một bộ ba thành phần ($r_attrs, r_facts, r_rules$). Trong đó r_attrs là tập thuộc tính của c ; r_facts là tập các tính chất nội tại vốn có của c ; r_rules là tập luật nội tại bên trong c . r_rules có 2 loại (r_rules_r, r_rules_f). r_rules_r là tập luật có dạng luật dẫn $h \rightarrow g$, h và g là hai tập sự kiện; r_rules_f là tập luật có dạng đẳng thức có giá trị thực, $u = v$ (sao cho $vars(u) \cap vars(v) = \emptyset$ và $\forall x \in vars(u) \cup vars(v), dom(x) \subseteq \mathbb{R}$).

- Mỗi lớp đối tượng sẽ được trang bị khả năng giải quyết các bài toán nội tại gồm: *Xác định bao đóng tập sự kiện; Kiểm tra sự giải được và tìm lời giải cho bài toán $GT \rightarrow KL$.*

Một mô hình tri thức quan hệ (Rela-Model) [21] là một mô hình gồm có 3 thành phần ($C_R, R_R, Rules_R$). Trong đó: C_R là tập các khái niệm, được phân cấp dựa trên cấu trúc của chúng; R_R là tập các quan hệ hai ngôi trên các khái niệm, $Rules_R$ là tập các luật dẫn trong miền tri thức. Các sự kiện cùng các thuật giải hợp nhất sự kiện cũng được xem xét theo mô hình (chi tiết theo [21]).

2.2. Mô hình tri thức toán tử

Một đối tượng xét các mối quan hệ tính toán [22] có các đặc trưng gồm:

- Cấu trúc gồm bộ ba thành phần ($o_attrs, o_facts, o_rules$). Trong đó: o_attrs là tập thuộc tính của

c ; o_facts là tập các tính chất nội tại vốn có của c ; o_rules là tập các luật dạng đẳng thức trên các đối tượng, với $r \in o_rules$ có dạng $u = v$ (với điều kiện $vars(u) \cap vars(v) = \emptyset$).

- Mỗi lớp đối tượng sẽ được trang bị khả năng giải quyết các bài toán nội tại gồm: *Xác định bao đóng tập sự kiện; Kiểm tra sự giải được và tìm lời giải cho bài toán $GT \rightarrow KL$.*

Một mô hình tri thức toán tử (Ops-Model) [22] là một mô hình gồm có 3 thành phần (C_{Op} , R_{Op} , $Rules_{Op}$). Trong đó, C_{Op} là tập các khái niệm, R_{Op} là tập các toán tử, $Rules_{Op}$ là tập các luật dẫn trong miền tri thức. Các sự kiện cùng các thuật giải hợp nhất sự kiện cũng được xem xét theo mô hình (chi tiết theo [22]).

2.3. Phương pháp tích hợp mô hình Rela-Model và Ops-Model cùng vấn đề xây dựng cơ sở tri thức tích hợp

Định nghĩa 2.1: Mô hình tri thức tích hợp giữa Rela-Model và Ops-Model

Cho mô hình tri thức quan hệ Rela-Model gồm bộ ba (C_R , R_R , $Rules_R$) và mô hình tri thức toán tử Ops-Model gồm bộ ba (C_{Op} , Ops_{Op} , $Rules_{Op}$). Ta gọi một mô hình tri thức tích hợp giữa Rela-Model và Ops-Model là một mô hình gồm 7 thành phần sau đây:

$$(C_I, C_U, R_I, R_U, Ops_I, Ops_U, Rules_I)$$

Trong đó:

C_I là tập các khái niệm được tích hợp. $C_I = \{c \mid c \in C_R \wedge c \in C_{Op}\}$. Mỗi khái niệm $c \in C_I$ có cấu trúc gồm bộ ($attrs_I, facts_I, rules_I$), với $attrs_I$ là tập thuộc tính của khái niệm c , $attrs_I = r_attrs \cup o_attrs, facts_I = r_facts \cup o_facts, rules_I = r_rules \cup o_rules$. Các khái niệm trong tập C_I cũng được phân cấp dựa trên cấu trúc tập thuộc tính tương tự C_R, C_{Ops} (được trình bày trong [21] và [22]).

C_U là tập khái niệm khác C_I . C_U gồm hai thành phần (C_{UR}, C_{UO}).

- $C_{UR} = \{c \mid (c \notin C_I) \wedge (c \in C_R)\}$
- $C_{UO} = \{c \mid (c \notin C_I) \wedge (c \in C_{Op})\}$

R_I là tập quan hệ hai ngôi trên các khái niệm C_I . Trên R_I ta có hai loại quan hệ như sau.

- R_{IH} tập các quan hệ thứ tự trên C_I , $R_{IH} = \{(a, b) \mid (a, b) \in R_{RH} \wedge a, b \in C_I\}$
- R_{IR} tập các quan hệ khác trên C_I , $R_{IR} = \{(a, b) \mid (a, b) \in R_{RR} \wedge a, b \in C_I\}$

R_U là tập quan hệ hai ngôi trên C_U .

- $R_{UH} = \{(a, b) \mid (a, b) \in R_{RH} \wedge a, b \in C_{UR}\}$
- $R_{UR} = \{(a, b) \mid (a, b) \in R_{RR} \wedge a, b \in C_{UR}\}$

Ops_I là tập toán tử trên C_I . Có thể chia làm hai loại:

- Ops_{I1} tập các phép toán một ngôi trên C_I , $Ops_{I1} = \{op^1(a) \mid a \in C_I\}$
- Ops_{I2} tập các phép toán hai ngôi trên C_I , $Ops_{I2} = \{op^2(a, b) \mid a, b \in C_I\}$

Ops_U là tập các toán tử trên C_U . Có thể chia làm hai loại:

- Ops_{U1} tập các phép toán một ngôi trên C_U , $Ops_{U1} = \{op^1(a) \mid a \in C_{UO}\}$
- Ops_{U2} tập các phép toán hai ngôi trên C_U , $Ops_{U2} = \{op^2(a, b) \mid a, b \in C_{UO}\}$

$Rules_I$ là tập các tri thức luật trên miền tri thức tích hợp, $Rules_I = Rules_R \cup Rules_{Op}$.

Định nghĩa 2.2: Bài toán xây dựng tri thức tích hợp K_I

Cho cơ sở tri thức K_R (được mô hình hóa bởi mô hình tri thức quan hệ Rela-Model) và cơ sở tri thức K_{Ops} (được mô hình hóa bởi mô hình tri thức Ops-Model). Trên cơ sở K_R và K_{Ops} , hãy xác định cơ sở tri thức tích hợp K_I . K_I là một cơ sở tri thức tích hợp, có cấu trúc được mô hình theo định nghĩa 2.1. Tri thức tích hợp K_I có thể được xây dựng từ cơ sở tri thức K_R và cơ sở tri thức K_{Op} bởi thuật giải sau.

Thuật giải 2.1: Xây dựng cơ sở tri thức K_I từ K_R và K_{Op}

Input: K_R, K_{Op} ; // K_R là cơ sở tri thức quan hệ, K_{Op} là cơ sở tri thức toán tử
Output: K_I ; // K_I là cơ sở tri thức tích hợp.

Bước 1: Tạo các lớp c trong C_I . Duyệt tìm các khái niệm $c \in C_R$ và $c \in C_{Op}$. Nếu tồn tại c thì cập nhật cấu trúc ($attrs_I, facts_I, rules_I$) cho khái niệm c .
 + Kết nạp $r_attrs(c)$ và $o_attrs(c)$ vào $attrs_I(c)$;
 + Kết nạp $r_facts(c)$ và $o_facts(c)$ vào $facts_I(c)$;
 + Kết nạp $r_rules(c)$ và $o_rules(c)$ vào $rules_I(c)$;

Bước 2: Tạo các lớp c trong C_U
 + $C_{UR} = C_R - C_I$, với mọi $c \in C_{UR}$ thực hiện cập nhật tập thuộc tính, tập sự kiện, và tập luật theo c .
 + $C_{UO} = C_{Op} - C_I$, với mọi $c' \in C_{UO}$ thực hiện cập nhật tập thuộc tính, tập sự kiện, và tập luật theo c' .

Bước 3: Tạo quan hệ R_I
 + Duyệt tìm các khái niệm a và b , sao cho $(a, b) \in R_{RH}$. Nếu $a, b \in C_I$, thì chuyển (a, b) vào R_{IH} .
 + Duyệt tìm các khái niệm a và b , sao cho $(a, b) \in R_{RR}$. Nếu $a, b \in C_I$, thì chuyển (a, b) vào R_{IR} .

Bước 4: Tạo quan hệ R_U
 + Duyệt tìm các khái niệm a và b , sao cho $(a, b) \in R_{RH}$. Nếu $a, b \in C_{UR}$, thì chuyển (a, b) vào R_{UH} .
 + Duyệt tìm các khái niệm a và b , sao cho $(a, b) \in R_{RR}$. Nếu $a, b \in C_{UR}$, thì chuyển (a, b) vào R_{UR} .

Bước 5: Tạo toán tử Ops_I
 + Duyệt tìm các phép toán một ngôi op^1 , sao cho $op^1(a) \in Ops_I$. Nếu $a \in C_I$, thì chuyển op^1 vào Ops_{I1} .
 + Duyệt tìm các phép toán hai ngôi op^2 , sao cho $op^2(a, b) \in Ops_I$. Nếu $a, b \in C_I$, thì chuyển op^2 vào Ops_{I2} .

Bước 6: Tạo toán tử Ops_U
 + Duyệt tìm các phép toán một ngôi op^1 , sao cho $op^1(a) \in Ops_I$. Nếu $a \in C_{UO}$, thì đưa op^1 vào Ops_{U1} .
 + Duyệt tìm các phép toán hai ngôi op^2 , sao cho $op^2(a, b) \in Ops_I$. Nếu $a, b \in C_{UO}$, thì đưa op^2 vào Ops_{U2} .

Bước 7: Tạo luật, kết nạp $Rules_R$ và $Rules_{Op}$ vào $Rules_I$

Hình 1. Thuật giải xây dựng tri thức tích hợp

Ý tưởng chính của thuật giải trong xây dựng tri thức tích hợp K_I là sự hợp nhất một cách tuần tự các thành phần trong cơ sở tri thức quan hệ K_R và tri thức toán tử K_{Ops} . Trong quá trình hợp nhất các vấn đề liên quan đến sự không đồng nhất cấu trúc cũng sẽ được xem xét và xử lý trong quá trình cài đặt.

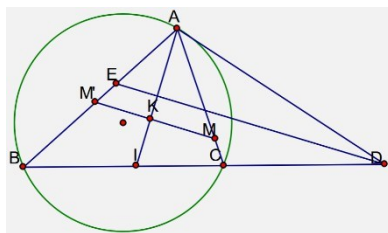
Tri thức về hình học giải tích hai chiều là một trong những miền tri thức được xây dựng từ việc tích hợp các tri thức từ hình học phẳng và các kiến thức về đại số cơ bản.

Ví dụ: khái niệm đoạn thẳng AB trong miền tri thức hình học phẳng xét về phương diện cấu trúc chúng ta chỉ quan tâm đến các thuộc là hai điểm đầu đoạn thẳng và giá trị mang ý nghĩa độ dài của đoạn thẳng. Cấu trúc của đoạn thẳng AB có thể được biểu diễn bởi lớp đối tượng theo mô hình tri thức quan hệ [21] như sau: $r_attrs(AB) = \{A, B, a\}$, trong đó A, B là hai điểm thuộc khái niệm Điểm, a là một giá trị thực. Trong kiến thức đại số cơ bản, khái niệm phương trình bậc nhất (PTBN) hai ẩn có dạng $a*x + b*y + c = 0$, có thể được biểu diễn bởi lớp đối tượng theo mô hình tri thức toán tử [22] như sau: $o_attrs(PTBN) = \{a, b, c, x, y\}$, trong đó bộ (a, b, c) là các hệ số nhận giá trị thực, bộ (x, y) là các biến có miền giá trị thực. Trong hình học giải tích hai chiều, đoạn thẳng AB sẽ được hình thành bằng việc bổ sung thêm e (phương trình đoạn thẳng của đoạn AB) và hai vector \vec{n}, \vec{u} (vector pháp tuyến, vector chỉ phương). Ta có cấu trúc tập thuộc tính của AB sau khi được xây dựng theo thuật giải 2.1 như sau: $attrs_I(AB) = \{A, B, a, e, \vec{n}, \vec{u}\}$, với A, B là khái niệm thuộc lớp điểm; a là giá trị độ dài của đoạn AB , $a \in \mathbb{R}$; e là phương trình đường thẳng thuộc lớp PTBN; \vec{n} và \vec{u} thuộc lớp Vector.

3. BÀI TOÁN TRÊN MÔ HÌNH TRI THỨC TÍCH HỢP VÀ PHƯƠNG PHÁP SUY LUẬN GIẢI VẤN ĐỀ TRÊN TRI THỨC TÍCH HỢP

Định nghĩa 3.1: Cho miền tri thức K_R được mô hình hóa theo Rel-Model và tri thức K_{Op} được mô hình hóa theo Ops-Model. Ta gọi K_I cơ sở tri thức tích hợp được xây dựng từ cơ sở tri thức quan hệ K_R và cơ sở tri thức toán tử K_{Op} (theo thuật giải 2.1). Ta có bài toán trên mô hình K_I có dạng $GT \Rightarrow KL$. Trong đó GT là một bộ (O, F) , với O là tập các đối tượng; F là tập các sự kiện; KL là các tập mục tiêu của bài toán. Mỗi mục tiêu thuộc một trong hai loại yêu cầu sau đây: Tìm lời giải cho bài toán tính giá trị của một đối tượng/thuộc tính của đối tượng hoặc một biểu thức; Tìm lời giải cho bài toán chứng minh sự xác định của một sự kiện.

Ví dụ 3.1: Trong mặt phẳng với hệ tọa độ Oxy, cho một tam giác ABC có A(1;4), tiếp tuyến tại A của đường tròn ngoại tiếp tam giác ABC cắt BC tại D, đường phân giác trong của ADB có phương trình $x - y + 2 = 0$, điểm M(-4;1) thuộc cạnh AC. Viết phương trình đường thẳng AB.



Hình 2. Hình minh họa cho bài tập tại ví dụ 3.1

Ta có thể mô hình hóa bài toán tại ví dụ 3.1 trên như sau:

$O = \{[A, \text{"DIEM"}], [B, \text{"DIEM"}], [C, \text{"DIEM"}], [D, \text{"DIEM"}], [E, \text{"DIEM"}], [I, \text{"DIEM"}], [K, \text{"DIEM"}], [O, \text{"DIEM"}], [TAMGIAC[A, B, C], \text{"TAMGIAC"}], [TAMGIAC[A, D, B], \text{"TAMGIAC"}], [DUONGTRON[O, R], \text{"DUONGTRON"}]\};$
 $F = \{[\text{"PHANGIAC"}, PHANGIAC[D, E], GOC[A, D, B]], [\text{"PHANGIAC"}, PHANGIAC[A, I], GOC[B, A, C]], [\text{"THUOC"}, I, DOAN[B, C]], [\text{"THUOC"}, M, DOAN[A, C]], [\text{"THUOC"}, N, DOAN[A, B]], [\text{"DOIXUNG"}, K, DOAN[M, N], DOAN[A, I]]\};$
 $G = \{[\text{"VIETPHUONGTRINH"}, [DUONGTHANG[A, B].e]\}.$

Quá trình suy luận trên cơ sở tri thức tích hợp K_I là quá trình mở rộng dần tập sự kiện ban đầu GT thông qua việc áp dụng các quy tắc suy luận để sinh ra các sự kiện mới. Quá trình mở rộng này sẽ dừng lại khi hoặc đạt được các sự kiện mục tiêu trong KL hoặc là không thể sinh ra thêm sự kiện mới. Ta có quy tắc suy luận để sinh sự kiện mới trên K_I được trình bày dưới đây.

Định nghĩa 3.2: Quy tắc suy luận

Ta có các quy tắc suy luận trên mô hình tri thức Rela-Model, Ops-Model và tri thức tích hợp được phân thành các loại sau:

- Các quy tắc suy luận trên Rela-Model
 - (RRM1) Loại quy tắc “Tự động sinh ra các sự kiện mặc nhiên”.
 - (RRM2) Loại quy tắc “Sinh ra sự kiện mới dựa trên việc áp dụng một luật dẫn”.
 - (RRM3) Loại quy tắc “Sinh ra sự kiện mới dựa trên khả năng giải quyết vấn đề của một đối tượng”.
- Các quy tắc suy luận trên Ops-Model
 - (ROM1) Loại quy tắc “Tự động sinh ra các sự kiện mặc nhiên hay các quy tắc đại số cơ bản”.
 - (ROM2) Loại quy tắc “Sinh ra sự kiện mới dựa trên việc áp dụng một luật dạng đẳng thức”.
 - (ROM3) Loại quy tắc “Sinh ra sự kiện mới dựa trên kết hợp nhiều luật dạng đẳng thức để giải hệ phương trình”.
 - (ROM4) Loại quy tắc “Sinh ra sự kiện mới dựa trên khả năng giải quyết vấn đề của một đối tượng”.
- Các quy tắc suy luận trên mô hình tri thức tích hợp
 - (RIK1) Loại quy tắc “Tự động sinh ra các sự kiện mặc nhiên hay cá quy tắc đại số cơ bản”.
 - (RIK2) Loại quy tắc “Sinh ra sự kiện mới dựa trên việc áp dụng một luật dẫn”.
 - (RIK3) Loại quy tắc “Sinh ra sự kiện mới dựa trên việc áp dụng một luật dạng đẳng thức”.
 - (RIK4) Loại quy tắc “Sinh ra sự kiện mới dựa trên kết hợp nhiều luật dạng đẳng thức để giải hệ phương trình”.
 - (RIK5) Loại quy tắc “Sinh ra sự kiện mới dựa trên khả năng giải quyết vấn đề của một đối tượng”.

Định nghĩa 3.3: Bước giải và lời giải

Cho miền cơ sở tri thức tích hợp K_I và bài toán $GT \Rightarrow KL$, ta có:

- Một *bước giải* là một bộ ba (rf, kf, nf). rf là một quy tắc suy luận được dùng để sinh ra sự kiện mới; kf là tập các sự kiện cần để áp dụng được rf ; nf là tập sự kiện mới được sinh ra từ luật rf .
- Một *danh sách các bước giải* có dạng $S = [s_1, s_2, s_3, \dots, s_m]$, với mỗi s_i là một *bước giải*. Ta ký hiệu $GT_0 = GT$, $GT_1 = s_1(GT_0)$, $GT_2 = s_2(GT_1)$, $GT_3 = s_3(GT_2)$, ..., và $S(GT) = GT_m$. Ta nói bài toán $GT \Rightarrow KL$ là giải được khi $KL \subseteq S(GT)$ và S là một *lời giải* của bài toán.
- S được gọi là *lời giải* không có bước thừa khi và chỉ khi không tồn tại $S' \subseteq S$ sao cho $KL \subseteq S'(GT)$. 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 loại bỏ bước thừa trong lời giải

```

Input: KL;
Output: NewS một lời giải mới không có bước giải thừa.
TempVars ← KL;
NewS ← [];
for i from length(S) down to 1 do
  if S[i](Fn) ⊆ TempVars then
    TempVars ← TempVars \ S[i](Fn);
    TempVars ← TempVars ∪ S[i](Prem);
    NewS ← [S[i], op(NewS)]; // thêm vào đầu của danh sách NewS;
  end if;
end do;
return NewS;

```

Hình 3. Thuật giải loại bỏ bước giải thừa trong S

Trên cơ sở các quy tắc suy luận, bước giải và lời giải trên cơ sở tri thức K_1 . Lời giải cho bài toán $GT \Rightarrow KL$ có thể được tìm thấy bởi thuật giải 3.2 dưới đây.

Thuật giải 3.2: Thuật giải tìm lời giải cho bài toán $GT \Rightarrow KL$.

```

Input:  $K_1, GT, KL$ ; //  $K_1$  là cơ sở tri thức tích hợp
Output: Sols; // Sols là lời giải cho bài toán  $GT \Rightarrow KL$ 
Bước 1: Khởi tạo các biến: KnownFacts := H và Sols ← [];
Bước 2: while  $KL \not\subseteq \text{KnownFacts}$  do
  2.1 Áp dụng loại quy tắc RIK1 để tự động sinh ra sự kiện mới từ các sự kiện sẵn có. Nếu sinh ra được sự kiện mới thì kết nạp sự kiện mới vào KnownFacts, lưu lại bước thực hiện vào biến Sols.
  2.2 Áp dụng loại quy tắc RIK2 để tìm luật dẫn r trong  $rules_I$  có thể áp dụng được. Nếu tìm được r thực hiện áp dụng luật r để sinh sự kiện mới. Cập nhật sự kiện mới vào KnownFacts, lưu lại bước thực hiện vào biến Sols.
  2.3 Áp dụng loại quy tắc RIK3 để tìm luật dạng đẳng thức trong  $rules_I$  có thể áp dụng được. Nếu tìm được r thực hiện áp dụng đẳng thức r để sinh sự kiện mới. Bổ sung sự kiện mới vừa tìm được vào KnownFacts, lưu lại bước thực hiện vào biến Sols.
  2.4 Áp dụng loại quy tắc RIK4 để kết hợp các đẳng thức trong  $rules_I$  thành các hệ phương trình. Nếu tìm được hệ phương trình có thể kết hợp được thì thực hiện giải hệ phương trình để sinh sự kiện mới. Trường hợp có thể sinh ra sự kiện mới thì thực hiện bổ sung sự kiện mới vào KnownFacts, lưu lại bước thực hiện vào biến Sols.
  2.5 Áp dụng loại quy tắc RIK5 để tìm đối tượng o sao cho o (thỏa điều kiện  $attrs_I(o) \neq \emptyset$ ). Nếu tìm được đối tượng o thì thực hiện công việc tìm bao đóng tập sự kiện trên đối tượng o. Bổ sung sự kiện mới vừa tìm được vào KnownFacts, lưu lại bước thực hiện vào biến Sols.
  2.6 Nếu không có sự kiện nào được sinh ra ở bước 2.1, 2.2, 2.3, 2.4, 2.5 thì sang bước 3.
Bước 3: Nếu  $KL \not\subseteq \text{KnownFacts}$  thì bài toán không giải được; Ngược lại thì thực hiện áp dụng thuật giải 3.1 để rút gọn bước giải thừa trong Sols. Xuất kết quả trong Sols.

```

Hình 4. Thuật giải tìm lời giải cho bài toán trên mô hình tri thức tích hợp

Ta có thể tìm được lời giải cho bài tập tại ví dụ 3.1 bởi thuật giải 3.2. Dưới đây là lời giải được tìm thấy bởi thuật giải 3.2 cho bài tập tại ví dụ 3.1.

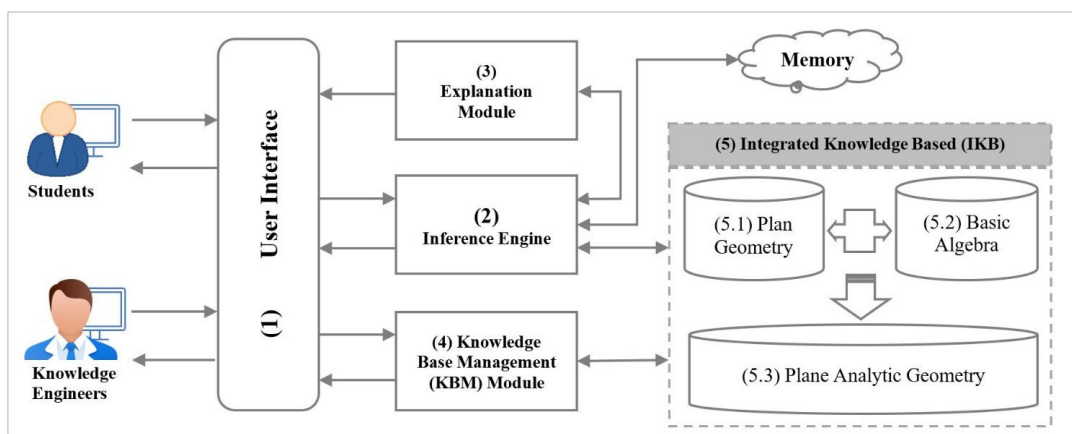
<p>Bước 1 ["PHANGIAC", DOAN[A, I], GOC[B, A, C]] ⇒ GOC[A, I, D].a = GOC[A, B, C].a + GOC[B, A, I].a, GOC[I, A, D].a = GOC[C, A, D].a + GOC[C, A, I].a <i>Ap dụng loại quy tắc RIK2</i></p> <p>Bước 2 GOC[A, I, D].a = GOC[A, B, C].a + GOC[B, A, I].a, GOC[I, A, D].a = GOC[C, A, D].a + GOC[C, A, I].a, GOC[B, A, I].a = GOC[C, A, I].a, GOC[A, B, C].a = GOC[C, A, D].a ⇒ GOC[A, I, D].a = GOC[I, A, D].a <i>Ap dụng loại quy tắc RIK1</i></p> <p>Bước 3 GOC[A, I, D].a = GOC[I, A, D].a ⇒ [TAMGIAC[D, A, I], "TAMGIACCAN"] <i>Ap dụng loại quy tắc RIK5</i></p> <p>Bước 4 [TAMGIAC[D, A, I], "TAMGIACCAN"], ["PHANGIAC", DOAN[D, E], GOC[A, D, B], ["THUOC", I, DOAN[B, D]]] ⇒ ["VUONGGOC", DOAN[D, E], DOAN[A, I]] <i>Ap dụng loại quy tắc RIK5</i></p> <p>Bước 5 DOAN[D, E].e = [x - y + 2 = 0] ⇒ DOAN[A, E].n.x = 1, DOAN[A, E].n.y = -1, DOAN[A, E].u.x = 1, DOAN[A, E].u.y = 1 <i>Ap dụng loại quy tắc RIK2</i></p> <p>Bước 6 ["VUONGGOC", DOAN[D, E], DOAN[A, I]] ⇒ DOAN[A, I].n = DOAN[D, E].u <i>Ap dụng loại quy tắc RIK2</i></p> <p>Bước 7 DOAN[A, I].n = DOAN[D, E].u, DOAN[A, E].u.x = 1, DOAN[A, E].u.y = 1 ⇒ DOAN[A, I].n.x = 1, DOAN[A, I].n.y = 1 <i>by using the rule type RIK1</i></p> <p>Bước 8 DOAN[A, I].n.x = 1, DOAN[A, I].u.y = 1, A.x = 1, A.y = 4 ⇒ DOAN[A, I].e = [x + y - 5 = 0] <i>Ap dụng loại quy tắc RIK1</i></p> <p>Bước 9 DOAN[A, I].e = [x + y - 5 = 0] ⇒ DOAN[A, I].n.x = 1, DOAN[A, I].n.y = 1, DOAN[A, I].u.x = 1, DOAN[A, I].u.y = -1 <i>Ap dụng loại quy tắc RIK1</i></p>	<p>Bước 10 ["DOIXUNG", K, DOAN[M, N], DOAN[A, I]] ⇒ ["VUONGGOC", DOAN[M, N], DOAN[A, I]], ["THUOC", K, DOAN[A, I]], ["THUOC", K, DOAN[M, N]], ["TRUNGDIEM", K, DOAN[M, N]] <i>Ap dụng loại quy tắc RIK2</i></p> <p>Bước 11 {"VUONGGOC", DOAN[M, N], DOAN[A, I]} ⇒ DOAN[M, N].n = DOAN[A, I].u, DOAN[M, N].u = DOAN[A, I].n <i>by using the rule type RIK2</i></p> <p>Bước 12 DOAN[M, N].n = DOAN[A, I].u, DOAN[A, I].u.x = 1, DOAN[A, I].u.y = -1 ⇒ DOAN[M, N].n.x = 1, DOAN[M, N].n.y = -1 <i>Ap dụng loại quy tắc RIK1</i></p> <p>Bước 13 DOAN[M, N].n.x = 1, DOAN[M, N].n.y = -1, M.x = -4, M.y = 1 ⇒ DOAN[M, N].e = [x - y + 5 = 0] <i>Ap dụng loại quy tắc RIK1</i></p> <p>Bước 14 ["THUOC", K, DOAN[A, I]], ["THUOC", K, DOAN[M, N]], DOAN[A, I].e = [x + y - 5 = 0], DOAN[M, N].e = [x - y + 5 = 0] ⇒ K.x = 0, K.y = 5 <i>Ap dụng loại quy tắc RIK4</i></p> <p>Bước 15 ["TRUNGDIEM", K, DOAN[M, N]], K.x = 0, K.y = 5, M.x = -4, M.y = 1 ⇒ N.x = 4; N.y = 9 <i>Ap dụng loại quy tắc RIK1</i></p> <p>Bước 16 A.x = 1, A.y = 4, N.x = 4; N.y = 9, ["THUOC", N, DOAN[A, B]] ⇒ DOAN[A, B].n.x = 5, DOAN[A, B].n.y = -3 <i>Ap dụng loại quy tắc RIK1</i></p> <p>Bước 17 DOAN[A, B].n.x = 5, DOAN[A, B].n.y = -3, A.x = 1, A.y = 4 ⇒ DOAN[A, B].e = [5x - 3y + 7 = 0] <i>Ap dụng loại quy tắc RIK5</i></p>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Hình 5. Lời giải minh họa được tìm thấy bởi thuật giải 3.2

4. KẾT QUẢ THỬ NGHIỆM HỆ THỐNG

4.1. Cấu trúc hệ thống

Trong phần này bài báo sẽ trình bày việc áp dụng thử nghiệm kết quả nghiên cứu vào thiết kế hệ giải vấn đề thông minh theo kiến trúc được trình bày tại [23], theo hình dưới đây.



Hình 6. Kiến trúc hệ thống giải vấn đề trên tri thức tích hợp [23]

- User Interface là nơi cho phép người sử dụng có thể giao tiếp với hệ thống và cũng là nơi cho phép hệ thống hiển thị kết quả cho người dùng.
- Knowledge Engineers là kỹ sư tri thức có vai trò thực hiện tổ chức, quản lý, cập nhật tri thức.
- Inference Engine là động cơ suy diễn của hệ thống, cho phép suy diễn giải các vấn đề tự động được đặt ra trên cơ sở tri thức tích hợp, trong đó được trang bị thêm module cho phép nhận diện được các dạng bài tập và thực hiện xử lý lựa chọn miền tri thức phù hợp cho việc giải vấn đề.

- Explanation Module là thành phần cho phép xử lý các kết quả từ động cơ suy diễn, thực hiện hậu xử lý các kết quả từ hệ thống sao cho thân thiện với người dùng.
- KBM là thành phần cho phép nhà kỹ sư tri thức có thể thực hiện các tính năng như quản trị cơ sở tri thức.
- IKB là thành phần tri thức của hệ thống, trong đó tri thức tích hợp bao gồm (5.1) tri thức về hình học phẳng, (5.2) tri thức về các kiến thức đại số cơ bản, và tri thức hình học giải tích hợp từ tri thức hình học phẳng và tri thức đại số cơ bản. Các vấn đề tích hợp, cũng như xây dựng tri thức tích hợp đã được trình bày tại định nghĩa 2.1, 2.2 và thuật giải 2.1.

4.2. Cài đặt thử nghiệm

Để đảm bảo được tính hiệu quả và đầy đủ, về phía tài liệu được sử dụng trong việc thu thập tri thức và các bài tập dùng để thử nghiệm được lấy từ nguồn sách giáo khoa (SGK) môn Toán đang được sử dụng trong giảng dạy cho học sinh cấp trung học cơ sở, trung học phổ thông. Trong đó các kiến thức và bài tập của miền đại số cơ bản được lấy từ SGK Toán lớp 6-9 (phần đại số). Đối với kiến thức và bài tập về hình học phẳng được lấy từ SGK Toán lớp 6-9 (phần hình học). Hình học giải tích hai chiều được lấy từ SGK Toán lớp 10 (phần phương pháp tọa độ trong mặt phẳng). Bên cạnh đó, các kiến thức và các bài tập cũng được thu thập, chọn lọc từ các sách bài tập, bài giải liên quan theo từng tương ứng.

Các kết quả nghiên cứu đã được áp dụng vào cài đặt hệ giải bài tập trên miền tri thức hình học giải tích hai chiều (27 bài), tri thức hình học phẳng (22 bài) và đại số cơ bản (28 bài tập). Trong đó tri thức hình học giải tích hai chiều được xây dựng dựa trên việc tích hợp tri thức về hình học phẳng và đại số và đại số, chi tiết về các bài tập thử nghiệm được chi tiết theo bảng dưới đây.

Bảng 1. Bảng số liệu tổng hợp số lượng bài tập thử nghiệm

STT	MIỀN TRI THỨC	CÁC DẠNG BÀI TẬP	SỐ BÀI TẬP
1	Đại số cơ bản	Rút gọn biểu thức	7
		Giải phương trình bậc nhất	5
		Giải phương trình bậc hai	8
		Giải hệ phương trình	8
2	Hình học phẳng	Tính giá trị của đối tượng/thuộc tính	12
		Chứng minh sự xác định của một sự kiện	10
3	Hình học giải tích hai chiều	Xác định tọa độ của điểm	12
		Viết phương trình đường thẳng/đoạn thẳng	15
Tổng			77

Các bài tập được thử nghiệm đều cho ra lời giải từng bước, tự nhiên, và có thể giải thích được trong quá trình tìm lời giải. Về mặt phương pháp và mô hình, cơ chế suy luận dựa trên lập luận logic từ các quy tắc suy luận (luật) để sinh ra sự kiện mới, vì vậy động cơ suy diễn hoàn toàn đáp ứng được vấn đề giải thích được quá trình tìm lời giải. Về lời giải được tìm từ hệ thống các tiêu chí về tự nhiên được đánh giá dựa trên ý kiến của chuyên gia gồm các giáo viên dạy trực tiếp các kiến thức này, các bài giải sẵn có trong các sách bài tập, bài giải cũng được dùng để quan sát, so sánh với lời giải của hệ thống.

4.3. Đánh giá và thảo luận

Một trong những mục tiêu quan trọng của nghiên cứu tích hợp tri thức đó chính là có khả năng tái sử dụng lại các tri thức có sẵn. Trong bài báo này đã trình bày một mô hình tri thức tích hợp giữa tri thức quan hệ Rela-Model và Ops-Model. Bên cạnh đó, việc xét bài toán xây dựng tri thức từ các tri thức sẵn có cũng được xem xét và trình bày. So với các giải pháp trình bày trước đây [13, 19-20] chỉ tập trung vào thiết kế mô hình tri thức tích hợp. Chưa xét đến các vấn đề xây dựng tri thức mới từ các tri thức sẵn có. Vì vậy, trong quá trình thiết kế, cần phải tốn chi phí trong việc xây dựng cơ sở tri thức

tích hợp một cách thủ công.

Tuy nhiên, đối với việc suy diễn tiến được trình bày trong bài báo này chưa thể hiện đủ các khía cạnh tư duy của con người trong việc giải vấn đề trên tri thức. Ở những điều này, thì các công trình của các nhóm tác giả [13, 19 - 20, 23] đã trình bày một cách đầy đủ và hệ thống.

So với một số hệ thống dựa trên mô hình ngôn ngữ lớn LLMs thì giải pháp này cho phép thiết kế các hệ thống có khả năng suy luận dựa trên tri thức chắc chắn (tri thức rõ). Đặc biệt việc tìm kiếm lời giải dựa trên các suy luận có các luật, vì vậy hệ thống vừa đảm bảo được độ tin cậy và vừa có khả năng giải thích được quá trình suy luận. Điều này là một trong những điểm mà các giải pháp dựa trên LLMs cần phải cải thiện. Tuy nhiên, LLMs lại cung cấp giải pháp giao tiếp tự nhiên, thân thiện. Vì vậy, việc kết hợp giữa LLMs và tri thức trong thiết kế các hệ thống AI là một trong những xu hướng mang tính thời đại.

5. KẾT LUẬN

Bài báo đã trình bày một mô hình tri thức tích hợp giữa Relational-Model và Ops-Model cùng các vấn đề như tạo dựng tri thức mới, các vấn đề suy diễn giải vấn đề trên miền tri thức tích hợp. Từ kết quả nghiên cứu trên, bài báo cũng đã vận dụng vào cài đặt thử nghiệm trong việc xây dựng tri thức hình học giải tích hai chiều từ kiến thức hình học phẳng, kiến thức về đại số. Kết quả cài đặt cho ra lời giải từng bước, tự nhiên và có thể giải thích được quá trình tìm lời giải. Nghiên cứu giúp việc sử dụng tri thức sẵn có để tạo dựng tri thức mới, vấn đề này có ý nghĩa quan trọng trong việc đóng góp phát triển các hệ thống thông minh giải vấn đề dựa trên tri thức, đặc biệt là tri thức quan hệ và tri thức toán tử.

Kết quả được trình bày vẫn còn một số vấn đề chưa được nghiên cứu sâu như phát triển các kỹ thuật suy luận trên tri thức tổng hợp từ tri thức hình học phẳng, đại số, tri thức tích hợp (giữa hình học phẳng và đại số) giúp việc suy luận trở nên linh hoạt và hiệu quả. Thuật giải tìm lời giải tập trung vào việc suy diễn tiến, chưa khai thác các nguyên lý heuristics như (vét cạn, tham lam, thư tự, hay các hàm heuristics), suy luận dùng mẫu bài toán, dùng bài toán mẫu. Bên cạnh đó, tri thức và bài tập thử nghiệm cũng còn hạn chế. Tính năng cập nhật tri thức cũng cần được nghiên cứu và xem xét.

Các nghiên cứu tiếp theo sẽ tập trung vào giải quyết các vấn đề sau như: hoàn thiện các cơ sở lý thuyết và các phương pháp suy diễn giúp hệ thống thể hiện được khả năng suy luận mềm mại, linh hoạt và hiệu quả trên đa miền tri thức. Tiếp tục phát triển các phương pháp tích hợp tri thức cho các dạng tri thức phổ biến khác như giữa tri thức hàm và tri thức quan hệ, giữa tri thức hàm và tri thức toán tử, giữa tri thức hàm, tri thức toán tử và tri thức quan hệ. Từ đó phát triển các cơ sở lý thuyết chặt chẽ, giúp cung cấp phương tiện cho các nhà phát triển có thể xây dựng các ứng dụng thông minh có khả năng giải vấn đề trên tri thức tích hợp

LỜI CẢM ƠN

Nghiên cứu này được Trường Đại học Quốc tế Hồng Bàng cấp kinh phí thực hiện dưới mã số đề tài GVTC19.01.

TÀI LIỆU THAM KHẢO

- [1] Thomas R. Gruber và cộng sự, "Enabling Technology for Knowledge Sharing," *AI Magazine*, vol. 12, no. 3, pp. 36-56, 1991.
- [2] Gayo Diallo và cộng sự, "Ontology integration: Approaches and challenging issues," *Inf. Fusion*, vol. 71, pp. 38-63, 2021.
- [3] A. De Baas và cộng sự, "Review and Alignment of Domain-Level Ontologies for Materials Science," *IEEE Access*, vol. 11, pp. 120372-120401, 2023.
- [4] R. Díaz và H. Xin, "Knowledge graphs in heterogeneous catalysis: Recent advances and future opportunities," *Chinese Journal of Chemical Engineering*, vol. 84, pp. 179-189, 2025.
- [5] N. T. Nguyen, *Advanced Methods for Inconsistent Knowledge Management*, London: Springer, 2008.

- [6] J. Pinto HS và cộng sự, "Some issues on ontology integration," in *Proceedings of the IJCAI-99 workshop on Ontologies and Problem-Solving Method*, Stockholm, 1999.
- [7] W. N. Borst, "Construction of engineering ontologies for knowledge sharing and reuse," Ph.D. dissertation, University of Twente, Enschede, The Netherlands, 1997.
- [8] Jeongsoo Lee và cộng sự, "An ontology architecture for integration of ontologies," trong *The Semantic Web - ASWC 2006*, Berlin, Heidelberg: Springer, pp. 518-532, 2006.
- [9] M. J. Sousa, "Knowledge Integration in Problem Solving Processes," in *Advances in Intelligent Systems and Computing*, Berlin, Springer, 2013, p. 97-109.
- [10] I. Osman, S. F. Pileggi, S. B. Yahia, và G. Diallo, "An alignment-based implementation of a holistic ontology integration method," in *MethodsX*, vol. 8, p. 101344, 2021.
- [11] H. H. Hạnh và cộng sự, "Một tiếp cận mới về đối sánh Ontology," trong *Kỷ yếu Hội nghị Quốc gia lần thứ VIII về Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR)*, 2019.
- [12] H. B. Truong và cộng sự, "An overview of fuzzy ontology integration methods based on consensus theory," trong *Advanced Computational Methods for Knowledge Engineering*, Springer, pp. 217-227, 2014.
- [13] Đ. V. Nhon, N. Đ. Hiên, và 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*, tập 53, số phần A, pp. 1-11, 2017.
- [14] N. V. Do, H. D. Nguyen, và T. T. Mai, "A method of ontology integration for designing intelligent problem solvers," *Appl. Sci.*, vol. 9, no. 5, p. 863, 2019.
- [15] N. P. Tran *et al.*, "A method for integrating of knowledge model and functional component and application in intelligent problem solver," trong *Advances and Trends in Artificial Intelligence. Theory and Applications*, Hradec Kralove, Czech Republic, 2024.
- [16] H. D. Nguyen và cộng sự, "Ontology-based integration of knowledge base for building an intelligent searching chatbot," *Sensors Mater.*, vol. 33, no. 9, pp. 3101-3121, 2021.
- [17] H. D. Nguyen *et al.*, "Design an ontology-based model for intelligent querying system in mathematics education," *J. Interdiscip. Math.*, vol. 26, no. 3, pp. 449-473, 2023.
- [18] N. V. Hung & *et al.*, "Integrating knowledge based on ontology and knowledge graph for chatbot in education query," *J. Ho Chi Minh City Univ. Educ.*, vol. 21, no. 5, 2024.
- [19] H. D. Nguyen, N. V. Do, V. T. Pham, và A. Selamat, "A method for knowledge representation to design intelligent problems solver in mathematics based on Rela-Ops model," *IEEE Access*, vol. 8, pp. 76991-77012, 2020.
- [20] N. P. Tran & *et al.*, "A method for integrating of knowledge model and functional component and application in intelligent problem solver," trong *Advances and Trends in Artificial Intelligence*, 2024. (*Lưu ý: Tài liệu này trùng lặp với [16]*)
- [21] A. Selamat và cộng sự, "Knowledge-based model of expert systems using Rela-Model," *Int. J. Softw. Eng. Knowl. Eng.*, vol. 28, no. 7, pp. 1047-1090, 2018.
- [22] N. V. Do và H. D. Nguyen, "Method for solving problems on a knowledge model of operators," trong *Proc. IEEE Conf. e-Learning, e-Management and e-Services (IC3e)*, 2016, pp. 132-137.
- [23] N. V. Do, "Intelligent problem solvers in education: Design method and applications," trong *Intelligent Systems*, InTech, 2012, pp. 315-338.