**LỜI NÓI ĐẦU**

Trong Trí tuệ nhân tạo, để phân tích, thiết kế và xây dựng các hệ thống ứng dụng thông minh như hệ chuyên gia, hệ trợ giúp quyết định, hệ giải toán dựa trên tri thức, ta phải quan tâm đến hai vấn đề cơ bản, quan trọng nhất, đó là thiết kế một cơ sở tri thức cho hệ thống và một động cơ suy diễn, tìm kiếm lập luận dựa trên tri thức. Trong thực tế, một dạng tri thức khá phổ biến của con người chính là tri thức quan hệ. Trong tri thức dạng này, sẽ gồm có các khái niệm và các quan hệ giữa các khái niệm, bên cạnh đó tri thức cũng có các luật suy diễn của miền. Ngoài ra, một dạng tri thức cũng khá phổ biến, đặc biệt là các miền tri thức tính toán, đó chính là tri thức về các toán tử. Tuy nhiên, các phương pháp nghiên cứu về biểu diễn tri thức hiện nay vẫn chưa nghiên cứu các miền tri thức này một cách thỏa đáng.

Trong chuyên đề này, chúng tôi sẽ trình bày một phương pháp biểu diễn cho hai dạng tri thức: tri thức quan hệ và tri thức toán tử dựa trên một tiếp cận toán học. Mô hình biểu diễn được xây dựng bằng các cấu trúc toán học như tập hợp, hoặc cấu trúc đại số khác. Thông qua đó, chúng tôi cũng sẽ mô hình hóa các bài toán trên miền tri thức và xây dựng các định lý cơ sở để thiết kế các thuật giải giải quyết các bài toán trên mô hình. Bên cạnh đó, các mô hình được xây dựng cũng được ứng dụng trong việc thiết kế các hệ cơ sở tri thức của các miền tri thức về hình học phẳng, đại số vector, dòng điện một chiều.

**MỤC LỤC**

[CHƯƠNG 1: TỔNG QUAN VỀ BIỂU DIỄN TRI THỨC QUAN HỆ VÀ TOÁN TỬ 3](#_Toc422696130)

[1.1 Biểu diễn tri thức 3](#_Toc422696131)

[1.2 Các phương pháp biểu diễn tri thức quan hệ 5](#_Toc422696132)

[1.3 Các phương pháp biểu diễn tri thức có chứa toán tử 7](#_Toc422696133)

[CHƯƠNG 2: XÂY DỰNG MÔ HÌNH TRI THỨC QUAN HỆ 10](#_Toc422696134)

[2.1 Mô hình tri thức quan hệ: 10](#_Toc422696135)

[2.1.1. Tập C – Tập các khái niệm: 10](#_Toc422696136)

[2.1.2 Tập H – tập các quan hệ phân cấp 14](#_Toc422696137)

[2.1.3 Tập R – tập các quan hệ: 14](#_Toc422696138)

[2.1.4 Tập Rules – tập các luật của miền tri thức: 15](#_Toc422696139)

[2.2 Thuật giải và Các định lý 16](#_Toc422696140)

[2.2.1 Mô hình bài toán và Lời giải 16](#_Toc422696141)

[2.2.2 Thuật giải suy diễn trên một đối tượng 17](#_Toc422696142)

[2.2.3 Thuật giải suy diễn của bài toán 18](#_Toc422696143)

[2.2.4 Các định lý cơ sở 20](#_Toc422696144)

[2.3 Ứng dụng xây dựng tri thức hình học phẳng: 21](#_Toc422696145)

[2.3.1 Thiết kế cơ sơ tri thức: 21](#_Toc422696146)

[2.3.2 Thiết kế động cơ suy diễn của hệ thống 23](#_Toc422696151)

[2.4 Kết luận 24](#_Toc422696152)

[CHƯƠNG 3: XÂY DỰNG MÔ HÌNH TRI THỨC TOÁN TỬ 26](#_Toc422696153)

[3.1 Mô hình tri thức toán tử: 26](#_Toc422696154)

[3.1.1. Tập C – Tập các khái niệm: 26](#_Toc422696155)

[3.1.2 Tập R – tập các quan hệ: 29](#_Toc422696156)

[3.1.3 Tập Ops – tập các toán tử: 29](#_Toc422696157)

[3.1.4 Tập Rules – tập các luật của miền tri thức: 30](#_Toc422696158)

[3.2 Mô hình bài toán và Thuật giải 32](#_Toc422696159)

[3.2.1 Mô hình bài toán và Lời giải 32](#_Toc422696160)

[3.2.2 Thuật giải suy diễn trên đối tượng 33](#_Toc422696161)

[3.2.3 Thuật giải suy diễn tìm lời giải của bài toán 34](#_Toc422696162)

[3.2.4 Đánh giá 36](#_Toc422696163)

[3.3 Ứng dụng 37](#_Toc422696164)

[3.3.1 Kiến trúc hệ giải bài toán thông minh 37](#_Toc422696165)

[3.3.2 Thiết kế cơ sở tri thức của miền tri thức Dòng điện một chiều: 38](#_Toc422696166)

[3.3.3 Thiết kế cơ sở tri thức của miền tri thức Đại số vector: 40](#_Toc422696167)

[3.4 Kết luận 43](#_Toc422696168)

[CHƯƠNG 4: KẾT LUẬN 44](#_Toc422696169)

[CÁC BÀI BÁO LIÊN QUAN ĐẾN CHUYÊN ĐỀ 45](#_Toc422696170)

[TÀI LIỆU THAM KHẢO 46](#_Toc422696171)

# CHƯƠNG 1: TỔNG QUAN VỀ BIỂU DIỄN TRI THỨC QUAN HỆ VÀ TOÁN TỬ

## 1.1 Biểu diễn tri thức

Trong khoa học Trí tuệ nhân tạo, biểu diễn tri thức và phương pháp suy diễn đóng một vai trò quan trọng, quyết định trong quá trình xây dựng và cài đặt hệ thống thông minh, đặc biệt là các hệ chuyên gia. Biểu diễn tri thức giúp cho hệ thống trở nên thông minh và dễ dàng tương tác hơn với người dùng. Để xây dựng các hệ thống trên, người ta phải thiết kế một cơ sở tri thức cho hệ thống và một động cơ suy diễn để giải quyết các vấn đề dựa trên tri thức. Chất lượng hoạt động của các hệ này phụ thuộc rất lớn vào tri thức đã có cũng như cách biểu diễn và tổ chức chúng. Việc xây dựng mô hình biểu diễn cho phép chúng ta có thể mô phỏng các tri thức, đồng thời có thể xác định được các kết quả có thể suy luận được từ tri thức. Mô hình biểu diễn tri thức là một bước quan trọng trong việc thiết kế các hệ cơ sở tri thức và các hệ chuyên gia. Ngày nay, có nhiều phương pháp mô hình hóa tri thức đã được đề xuất và ứng dụng. Các mô hình biểu diễn này phải đảm bảo các tiêu chuẩn: [5, 6]

* Tiêu chuẩn về sự đầy đủ của biểu diễn: mô hình phải biểu diễn đầy đủ các loại tri thức trong phạm vi nhất định của miền tri thức được khảo sát.
* Tiêu chuẩn về sự đầy đủ của suy luận:Mô hình phải biểu diễn được các bài toán tổng quát, đồng thời khả năng suy diễn để giải quyết các bài toán được đặt ra trên cơ sở tri thức hiện có. Trong quá trình suy luận, hệ thống phải biết kết hợp các thông tin hữu ích để có thể giải quyết bài toán một cách hợp lý. Hơn thế nữa, quá trình suy diễn này cũng phải được chứng minh tính đúng đắn một cách chặt chẽ.
* Tiêu chuẩn về sự tiếp nhận tri thức: mô hình có thể cập nhật tri thức mới.

Ngoài ra, đối với nhu cầu có thể ứng dụng các mô hình biểu diễn cho miền tri thức thực tế nhưng vẫn phải đảm bảo được tính chặt chẽ về phương pháp và suy luận, thì cần phải đảm bảo các tiêu chuẩn:

* Tính hình thức hóa: nghĩa là các đối tượng trong mô hình được nghiên cứu một cách hình thức, ở đây chúng phải được nghiên cứu thông qua các cấu trúc đại số hoặc các cấu trúc toán học khác. Việc hình thức hóa cũng chính là cơ sở để chứng minh cho tính đúng đắn của các thuật giải suy diễn một cách chặt chẽ.
* Tính ứng dụng: Các mô hình được xây dựng phải đảm bảo được có thể ứng dụng một cách trực tiếp vào các miền tri thức thực tế. Trên cơ sở đó minh chứng cho sự hiệu quả của mô hình.

Trong các sách [1, 2, 3] chúng ta đã biết những phương pháp để biểu diễn tri thức trong việc thiết kế hệ cơ sở tri thức, như mạng ngữ nghĩa, mạng nơron, đồ thị khái niệm.... Bên cạnh đó, một số các mô hình khác cũng được nghiên cứu và phát triển như: mạng tính toán (xem [7]), mô hình COKB (xem [8]), ontology (trong [4, 9]) … Các phương pháp này đều có những ưu điểm và khuyết điểm nhất định:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **STT** | **Phương pháp** | **Sự đầy đủ của biểu diễn** | **Khả năng cập nhật tri thức** | **Tính hình thức hóa** | **Tính ứng dụng** |
| 1 | Biểu diễn bằng logic |  | X | X | Không ứng dụng được cho các miền tri thức phức tạp |
| 2 | Biểu diễn bằng logic mô tả | X | X |  | Còn hạn chế trong việc biểu diễn các miền tri thức thực tế |
| 3 | Biểu diễn dạng mạng | Chưa thể hiện được trọn vẹn các kiến thức của một miền tri thức | X | Mộ số phương pháp đảm bảo tiêu chuẩn này | X |
| 4 | Biểu diễn dạng hệ luật dẫn |  | X |  | Không ứng dụng được cho các miền tri thức phức tạp |
| 5 | Biểu diễn tri thức dạng Frame và Script | X | X | Tính hình thức hóa chưa cao | X |
| 6 | Biểu diễn bằng ontology | X | X | Vẫn còn đang được nghiên cứu | X |
| 7 | Biểu diễn theo tiếp cận đại số | X | X | X | Các phương pháp hiện nay rất khó để áp dụng cho các miền tri thức thực. |

## 1.2 Các phương pháp biểu diễn tri thức quan hệ

Trong thực tế, một dạng tri thức khá phổ biến là tri thức có nền tảng là các khái niệm, các quan hệ giữa các khái niệm và các luật, chúng tôi gọi tắt các tri thức này là *tri thức có dạng quan hệ*. Đã có một số công trình nghiên cứu về tri thức có dạng quan hệ, tuy nhiên các phương pháp này đều chưa hiệu quả cho việc thiết kế các mô hình cũng như xây dựng các thuật giải cho các miền tri thức trong thực tế:

Cơ sở dữ liệu quan hệ được xây dựng trên lý thuyết Đại số quan hệ chỉ đáp ứng được nhu cầu tổ chức tổ chức các loại dữ liệu, và rất khó để có thể biểu diễn các loại tri thức, xem [10].

Trong [11], tác giả đã xây dựng một ontology cho một số loại quan hệ dựa trên cơ sở của logic vị từ và logic mô tả, tuy nhiên mô hình này vẫn còn rất hạn chế trong việc mô tả các miền tri thức dạng quan hệ phức tạp của con người

Trên cơ sở mạng ngữ nghĩa, trong [12], Helbig đã xây dựng Mạng mở rộng nhiều lớp (MultiNet) là một hệ thống biểu diễn tri thức dựa trên các mạng ngữ nghĩa. Trong hệ thống này, các biểu thức ngôn ngữ tự nhiên được biểu diễn bởi các cấu trúc khái niệm, trong đó các khái niệm cũng được biểu diễn bởi các nút và mối quan hệ giữa các niệm được biểu diễn bởi các cung. Tuy nhiên trong mô hình này, tác giả chỉ đề cập đến các quan hệ đơn giản giữa các khái niệm, đặc biệt là các khái niệm trong ngôn ngữ, các quan hệ phức tạp hơn trong miền tri thức thực vẫn chưa được tác giả đề cập đến.

Ở bài báo [13], dựa trên một số kết quả trong cơ sở dữ liệu quan hệ, bài báo đề xuất thuật toán xây dựng sơ đồ quan hệ từ bảng quyết định và thuật toán xây dựng bảng quyết định từ sơ đồ quan hệ, hai thuật toán này để giải quyết bài toán trích lọc tất cả các tri thức có thể từ bảng quyết định và bài toán xây dựng bảng quyết định thỏa mãn tập tri thức sẵn có dưới dạng phụ thuộc hàm. Bên cạnh đó, các tác giả trong [14] đã trình bày khái niệm về phân hoạch một hệ suy dẫn thành các hệ con, thuật toán phân hoạch hệ suy dẫn và định lý về biểu diễn cơ sở của hệ suy dẫn thông qua các cơ sở của hệ con. Tuy nhiên các kết quả này lại không thể áp dụng một cách trực tiếp trong các miền tri thức thực tế.

Trong chuyên đề này, chúng tôi sẽ đề xuất một mô hình mới để biểu diễn tri thức có nền tảng là các khái niệm, quan hệ và luật*.* Trong nghiên cứu này, mô hình chỉ đề cập đến các quan hệ hai ngôi và các luật suy diễn dưới dạng luật dẫn. Từ đó, mô hình cũng được áp dụng để đặc tả một phần tri thức của kiến thức hình học phẳng và xây dựng chương trình để giải quyết một số vấn đề trên miền kiến thức này.

## 1.3 Các phương pháp biểu diễn tri thức có chứa toán tử

Trong thực tế, một dạng tri thức khá phổ biến của con người chính là tri thức về các toán tử. Trong các miền tri thức tính toán như Đại số tuyến tính, Hình học giải tích, Vật lý, Hóa học, tri thức về toán tử giữa các khái niệm là một thành phần rất quan trọng, giúp chúng ta có thể biểu diễn chính xác hơn các kiến thức trong miền. Một số mô hình tri thức đã được ứng dụng và đề xuất, tuy nhiên chúng vẫn còn những khiếm khuyết nhất định:

Tác giả C.Yang và W.Cai cũng đã xây dựng cơ sở lý thuyết toán học trong việc biểu diễn tri thức dựa trên hệ luật mở rộng, các luật này được nghiên cứu trong việc giải quyết bài toán về kiểm tra sự mâu thuẫn trong các mô hình hình thức, trong [15]. Tuy vậy hệ thống hệ luật mở rộng này không hiệu quả trong hệ thống lớn, cũng như trong việc biểu diễn các tri thức có dạng mô tả, hoặc có cấu trúc.

Đối với các kết quả trong [16, 17], Y.Wang và các cộng sự cũng đã xây dựng các khái niệm của tri thức bằng cách sử dụng cấu trúc đại số để biểu diễn các thành phần của một khái niệm và những quan hệ trên nó. Tác giả cũng đã đề cập đến một số phép toán giữa các khái niệm trong miền tri thức, tuy nhiên trong mô hình, tác giả không đề cập đến các luật suy diễn của tri thức.

Công trình của tác giả trong [18] đã dùng cách tiếp cận đại số để giải quyết các vấn đề trong biểu diễn tri thức. Việc tiếp cận này sẽ cho ta một công cụ toán học để biểu diễn, lập luận và chứng minh các tính chất trong việc biểu diễn tri thức một cách chặt chẽ và chuẩn mực. Tuy nhiên các kết quả trước đây vẫn chưa thể áp dụng cho việc biểu diễn các tri thức thực tế phức tạp.

Jacques Calmet và cộng sự đã trình bày cách tiếp cận để chuyển đặc tả đại số của miền toán học về các tính toán thành các cơ sở tri thức, trong [19], từ đó áp dụng vào việc đặc tả ngôn ngữ hình thức FORMAL-S và hệ thống biểu diễn tri thức MANTRA. Tuy nhiên hệ thống này hiện nay đã ngừng phát triển.

E.Valencia và J.Giavitto [20] đã sử dụng cấu trúc đồ thị để biểu diễn tri thức cho việc giải quyết bài toán tương tự (analogy solving). Tuy nhiên, trong bài báo này lại chủ yếu chỉ đặc tả các quan hệ trên các thành phần đơn giản.

Tác giả trong [21] đã trình bày cấu trúc đại số của mô hình tri thức quan hệ, từ đó tác giả đã trình bày một phương pháp để thu nhận (acquisition) tri thức (một dạng của học máy). Tuy nhiên trong mô hình này, tác giả lại không chỉ rõ cấu trúc của một khái niệm trong tri thức cũng như không đề cập đến khái niệm luật của miền tri thức.

Trong bài báo [22], tác giả đã xây dựng một định nghĩa của cơ sở tri thức dựa trên cấu trúc đại số (D, , f), trong đó D là một miền dữ liệu (D là *đại số*), là tập các ký hiệu quan hệ (symbols of relations), và f là một interpretation (giải thích) của của các ký hiệu quan hệ, ví dụ, nếu là một quan hệ n-ngôi thì f() là tập con của tích Đề các Dn . Từ đó, tác giả giải quyết bài toán sự tương đương giữa hai cơ sơ tri thức thông qua việc chứng minh sự tương đương của hai nhóm tự đồng cấu tương ứng với hai cơ sở tri thức.

Mô hình tri thức các đối tượng tính toán (Computational Objects Knowledge Base – COKB) là một bộ gồm 6 thành phần *(C, H, R, Ops, Func, Rules)*, mô hình này có thể biểu diễn nhiều loại tri thức thực tế và đã được sử dụng để xây dựng các hệ thống thông minh (xem [23]). Mô hình COKB cùng với phương pháp suy diễn dựa trên tri thức bài toán mẫu [25] đã được ứng dụng trong việc xây dựng các hệ thống giải toán thông minh trong miền tri thức hình học phẳng, hình học giải tích. Tuy nhiên, trong mô hình này, khi áp dụng vào các miền tri thức thực tế thì rất phức tạp và ảnh hưởng đến tốc độ cũng như tính hiệu quả trong quá trình suy luận của hệ thống, hơn nữa trong mô hình này cũng chưa có sự nghiên cứu về thành phần toán tử, các nghiên cứu chưa đề cập đến các vấn đề cần phảỉ giải quyết trên toán tử như đặc tả toán tử, cũng như các bài toán liên quan đến toán tử.

Trong chuyên đề này, chúng tôi cũng sẽ trình bày một mô hình có nền tảng là các khái niệm, toán tử giữa các đối tượng trong tri thức và các luật*.* Trong nghiên cứu của bài báo, mô hình chỉ đề cập đến các toán tử hai ngôi và một số tính chất đặc trưng của toán tử: giao hoán, kết hợp, có phần tử trung hòa, đồng thời cũng nghiên cứu các lớp bài toán liên quan đến thành phần tri thức toán tử. Từ đó, vận dụng mô hình này để thiết kế cơ sở tri thức, bộ suy diễn cho hệ hỗ trợ giải tự động các bài tập trong kiến thức về Điện một chiều, và kiến thức về Đại số vector. Chương trình sẽ cho lời giải tự nhiên, tương tự như cách giải của con người.

# CHƯƠNG 2: XÂY DỰNG MÔ HÌNH TRI THỨC QUAN HỆ

Các kết quả trong chương này, đã được công bố trong [CĐ1, CĐ2].

Một số các ký hiệu được sử dụng trong chương này như sau:

* Σ\*: Tập hợp các biến
* Dom(x): miền giá trị của biến x, với x ∈ Σ\*
* ℜ: tập hợp các số thực
* Γ: tập hợp các sự kiện.
* var(u): Tập hợp các biến trong biểu thức u, var(u) ⊆ Σ\*
* card(S): số lượng phần tử của tập S

## 2.1 Mô hình tri thức quan hệ:

**Định nghĩa 2.1:** Mô hình biểu diễn tri thức quan hệ, gọi là *Rela-model*, là một bộ gồm 4 thành phần:

**K = (C, H, R, Rules)**

Trong đó:

* **C** là tập các khái niệm của miền tri thức.
* **H** là tập các quan hệ phân cấp giữa các khái niệm
* **R** là tập các quan hệ giữa các khái niệm trong tri thức, mỗi quan hệ này là một quan hệ hai ngôi giữa hai khái niệm trong tập C, và các quan hệ này không có dạng phân cấp.
* **Rules** là tập các luật suy diễn của tri thức, trong mô hình này ta chỉ xét các luật dưới dạng luật dẫn.

### 2.1.1. Tập C – Tập các khái niệm:

Mỗi khái niệm c ∈ C có một tập thể hiện, gọi là Ic, mỗi x ∈ Ic là một đối tượng của khái niệm c (đã xem, trang 62). Mỗi khái niệm là một lớp đối tượng được mô hình bởi cấu trúc sau:

(*Attrs, RulesObj*)

*Attrs* là tập các thuộc tính của đối tượng

*RulesObj* là tập các luật dẫn của đối tượng

Các khái niệm trong tập C được phân lớp như sau:

* Khái niệm cơ bản: gọi là C(0), lớp khái niệm này gồm tập số thực ℜ và các khái niệm khác có cấu trúc rỗng, nghĩa là: *Attrs = {}, RulesObj={ }*
* Khái niệm cấp 1: gọi là C(1), các thành phần của khái niệm thuộc lớp này thỏa điều kiện:



Và 

với FAttrs = {f Γ | var(f) ⊆ Attrs}, nghĩa là FAttrs là tập các sự kiện liên quan đến các thuộc tính trong tập Attrs

* Khái niệm cấp 2: gọi là C(2), các thành phần của khái niệm thuộc lớp này thỏa điều kiện:



Và 

với FAttrs = {f Γ | var(f) ⊆ Attrs}, nghĩa là FAttrs là tập các sự kiện liên quan đến các thuộc tính trong tập Attrs.

Ví dụ 2.1: Trong kiến thức về hình học phẳng, một số khái niệm có thể được đặc tả với cấu trúc như sau:

C(0) = {ℜ, ĐIỂM, ĐƯỜNG}

Khái niệm ĐOẠN ∈ C(1) có cấu trúc:

Attrs = {\_A, \_B, len}, which:

\_A, \_B: ĐIỂM

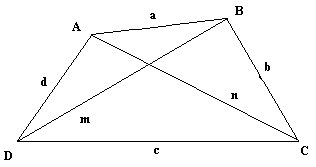
Dom(\_A) = IĐIỂM Dom(\_B) = IĐIỂM

len: độ dài của đoạn thẳng;

Dom(len) = ℜ

RulesObj = { }

Khái niệm TỨ\_GIÁC ∈ C(2) có cấu trúc:



*Attrs*= {\_A, \_B, \_C, \_D, a, b, c, d, m, n, S, p, ...}

\_A, \_B, \_C, \_D: Đỉnh của tứ giác

Dom(\_A) = IĐIỂM Dom(\_C) = IĐIỂM

Dom(\_B) = IĐIỂM Dom(\_D) = IĐIỂM

a, b, c, d: các cạnh của tứ giác

m, n: các đường chéo của tứ giác

Dom(a) = IĐOẠN Dom(b) = IĐOẠN

Dom(c) = IĐOẠN Dom(d) = IĐOẠN

Dom(m) = IĐOẠN Dom(n) = IĐOẠN

S, p: là các biến số thực biểu thị diện tích và chu vi của tứ giác

Dom(S) = ℜ Dom(p) = ℜ

*RulesObj*= { r1:{ m ⊥ n} → {S = ½\*m.len\*n.len}

r2: {b // d} → {DBC=BDA, BCA=CAD}

}

Bên cạnh cấu trúc, một đối tượng trong tri thức còn có các hành vi sau để giải quyết các vấn đề nội tại của nó:

1/ Xác định bao đóng của các sự kiện trong đối tượng.

2/ Cho biết lời giải của việc xác định thuộc tính của đối tượng từ các thuộc tính đã biết.

Trong các bài toán trên, thì việc xác định bao đóng của các sự kiện trong đối tượng chính là bài toán cơ sở để có thể giải quyết các bài toán sau. Vì vậy, trong báo cáo, chúng tôi chỉ đề cập đến hành vi này của đối tượng.

**Định nghĩa 2.2:** *Bao đóng của các sự kiện trong đối tượng*

Cho đối tượng Obj = (Attrs, RulesObj ) là một đối tượng của khái niệm trong C, và A = { f Γ | var(f) ⊆ Obj.Attrs}⊂ Γ, nghĩa là A là tập các sự kiện liên quan đến các thuộc tính của Obj.

Khi đó:

a/ Nếu r RulesObj: r có dạng: u(r) → v(r)

r *áp dụng được* trên A khi và chỉ khi 

b/ Đặt s = [r1, r2,…, rm] với rk ∈ RulesObj, s thỏa các điều kiện:

(1) r1 áp dụng được trên A. Đặt r1(A) = A v(r1)

(2) , rk áp dụng được trên rk-1(A).

Đặt rk(A) = rk-1(A) v(rk)

(3) ∀ r∈RulesObj\{r1, r2,…, rm}, r không áp dụng được trên rm(A)

Đặt DObj(A) = {s = [r1, r2,…, rm]| rk ∈ RulesObj, m ∈ N\*,

s thỏa các điều kiện conditions (1)(2)(3) }

c/ Đặt: 

, với dA DObj(A)

và card(dA) = min{card(s) | s DObj(A)}

**Định lý 2.1:** Cho đối tượng Obj **=** (Attrs, RulesObj ) là một đối tượng của một khái niệm trong C, và A là tập các sự kiện liên quan đến thuộc tính của Obj.

Giả sử tồn tại hai danh sách s = [r1, r2,…, rm] và t = [f1, f2,…, fn] với rk, fi ∈ RulesObj, s,t ∈ DObj(A).

Khi đó, ta có: rm(A) = fn(A)

Định lý 2.1 chỉ ra rằng định nghĩa Obj.Closure(A) trong định nghĩa 2.2c là một định nghĩa tốt.

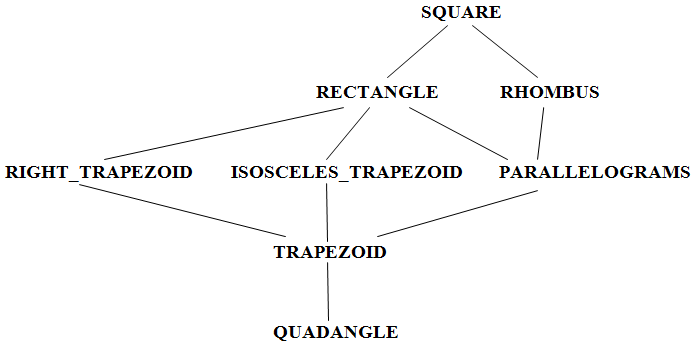
### 2.1.2 Tập H – tập các quan hệ phân cấp

H là tập các quan hệ phân cấp giữa các khái niệm, các quan hệ pah6n cấp này có thể được biểu diễn bằng biểu đồ Hasse.

[Ci, Cj]: Cj is a sub-concept of concept Ci



Ví dụ 2.2: Quan hệ phân cấp giữa các khái niệm tứ giác



### 2.1.3 Tập R – tập các quan hệ:

Thành phần này biểu diễn các quan hệ giữa các khái niệm trong miền tri thức. Mỗi quan hệ trong tập R là một quan hệ hai ngôi trên ci × cj , với ci, cj là các khái niệm trong tập C.



Trong trường hợp Φ ∈ R là quan hệ hai ngôi trên một khái niệm c∈C, các tính chất của Φ sẽ được khảo sát: tính phản xạ, đối xứng, phản xứng, bắc cầu.

Mỗi quan hệ được đặc tả như sau:

relation-def ::= RELATION <name>

ARGUMENT: argument-def+

PROPERTY: prob-type;

ENDRELATION

argument-def ::= name, <name>: type

prob-type ::= phản xạ | đối xứng | phản xứng | bắc cầu

Ví dụ 2.3: Trong miền tri thức Hình học phẳng, quan hệ VUÔNG GÓC giữa hai đường thẳng được biểu diễn như sau:

RELATION < VUÔNG GÓC> :

ARGUMENT: ĐƯỜNG, ĐƯỜNG

PROPERTY: đối xứng

ENDRELATION

### 2.1.4 Tập Rules – tập các luật của miền tri thức:

Mỗi luật r  Rules là một luật dẫn của miền tri thức có dạng sau:

u(r) = {f1, f2,…,fp}→ {q1,q2,…qk} = v(r)

Trong đó, fi và qi là các sự kiện của mô hình, với các loại sự kiện được phân loại như sau:

**Định nghĩa 2.3:** Phân loại các sự kiện của mô hình

|  |  |  |  |
| --- | --- | --- | --- |
| **Loại** | **Ý nghĩa** | **Đặc tả** | **Điều kiện** |
| 1 | Cho biết thông tin về loại của đối tượng | x : c | x \*, c C  Dom(x) Ic |
| 2 | Cho biết sự xác định của một đối tượng hoặc thuộc tính của một đối tượng. | x  x.a | x Ic (c C)  a x.Attrs |
| 3 | Sự xác định của một đối tượng thông qua một giá trị hằng | x = <const> | x Ic (c C) |
| 4 | Sự kiện về sự bằng nhau giữa một đối tượng với một đối tượng khác. | x = y | x Icx (cx C)  y Icy (cy C) |
| 5 | Cho biết quan hệ giữa các đối tượng | x Φ y | Φ R |

## 2.2 Thuật giải và Các định lý

### 2.2.1 Mô hình bài toán và Lời giải

*a/ Bài toán trong mô hình tri thức quan hệ*

**Định nghĩa 2.4:** Cho miền tri thức quan hệ K = (C, H, R, Rules), khi đó mô hình bài toán trên tri thức K là một bộ gồm 3 thành phần:

O = {O1, O2, . . ., On}, tập các đối tượng của bài toán

F = {f1, f2, . . ., fm}, tập các sự kiện của bài toán

G = { g1, g2, . . ., gk }, tập các mục tiêu. Mỗi mục tiêu gi có thể là:

* Xác định một đối tượng
* Xác định một quan hệ giữa các đối tượng

gi::= x | x = y | x Φ y (x Icx, x Icy, Φ R )

Bài toán được ký hiệu bởi **(O, F) → G**

*b/ Lời giải của bài toán*

**Định nghĩa 2.5:** Cho miền tri thức K = (C, H, R, Rules), đối tượng Obj **=** (Attrs, RulesObj ) là một đối tượng của khái niệm trong C, luật r Rules và A là tập các sự kiện, A ⊂ Γ. Ta có:

a/ Đặt A|Obj = {f A | var(f) ⊆ Obj.Attrs}

Obj(A) = Obj.Closure(A|Obj)

Obj(A) gọi là tập các sự kiện được suy diễn từ A bởi đối tượng Obj.

b/ Nếu r Rules: r có dạng u(r) → v(r)

r *áp dụng được trên* A khi và chỉ khi . Khi đó, đặt r(A) = A v(r)

r(A) gọi là tập các sự kiện được suy diễn từ A bởi luật r.

**Định nghĩa 2.6:** Cho miền tri thức K = (C, H, R, Rules), bài toán P = (O, F) → G trên K.

a/ Giả sử D = [d1,d2, …, dk] là danh sách các phần tử với dj Rules hay dj O. Đặt: F0 = F, F1 = d1(F0), F2 = d2(F1) , …. Fs = dk(Fk-1) và D(F) = Fk

Bài toán P được gọi là *giải được* khi và chỉ khi tồn tại danh sách D thỏa mãn G D(F).

b/ Khi đó,  :

+ Nếu dj Rules: sj = [dj, Fj-1, Fj\Fj-1]

+ Nếu dj O: Gọi A = Fj-1|dj

d = dj.Deduce(A) = [r1, r2,…, rm] ⊆ dj.RulesObj

pi = [ri, ri-1(A), ri(A)\ri-1(A)], ri d

sj = [pi | i=1..m];

sj được gọi là *bước giải* của bài toán P và  được gọi là *lời giải* của bài toán P.

### 2.2.2 Thuật giải suy diễn trên một đối tượng

**Thuật giải 2.1:** Trên miền tri thức Rela-model, cho đối tượng Obj = (Attrs, RulesObj) được đặc tả như trong mục 2.1.1 và A là một tập các sự kiện của Obj. Thuật giải sau sẽ xác định bao đóng của tập A bởi đối tượng Obj.

*Input:* Obj = (Attrs, RulesObj): đối tượng của miền tri thức Rela-model

A: Tập các sự kiện trên Obj

*Output:* Obj.Closure(A)

*Thuật giải:*

|  |
| --- |
| Result ← A  Deduce ← { }  **Do**  H ← Result  **Select** luật r trong RulesObj và r có thể áp dụng được trên Result  Result ← Result v(r)  Deduce ← Deduce { r }  **While** (H != Result)  Obj.Closure(A) = Result  Obj.Deduce(A) = Deduce |

### 2.2.3 Thuật giải suy diễn của bài toán

**Thuật giải 2.2:** Cho bài toán P = (O, F) → G như trong định nghĩa 2.4 của mô hình tri thức quan hệ Rela-model, ta có thuật giải sau tìm lời giải cho bài toán P:

*Input:* K = (C, R, Ops, Rules): Miền tri thức được biểu diễn bời Rela-model

P = (O, F) → G : Bài toán trên miền tri thức K

*Output:* Lời giải của bài toán P

*Thuật giải:* được xây dựng dựa trên chiến lược suy diễn tiến, trong đó các đối tượng tham gia vào quá trình suy diễn thông qua các hành vi của của chúng.

D empty list;

Solution\_found false;

**while** not(Solution\_found) **do**

**Use** đối tượng trong tập O và tập các sự kiện F để xác định tập bao đóng các sự kiện của mỗi đối tượng.

**if** (đối tượng Obj sinh ra sự kiện mới) **then**

**Update** tập sự kiện và đưa đối tượng Obj vào D;

**if** (goal G xác định) **then**

Solution\_found true;

**end if;**

continue;

**end if;**

**Search** một luật r trong tập Rules có thể áp dụng để suy diễn;

**if** (tìm thấy luật r) **then**

**Use** r để sinh ra sự kiện mới hoặc đối tượng mới;

**Update** tập sự kiện và thêm luật r vào danh sách D;

**if** (goal G xác định) **then**

Solution\_found true;

**end if;**

**end if;**

**end do;** { while }

**if** (Solution\_found) **then**

Từ danh sách D, xây dựng lời giải của bài toán P.

**else**

Không tìm thấy lời giải;

**end if;**

Các thuật giải cho mô hình tri thức toán tử thể hiện tư duy của con người trong quá trình suy diễn, tìm lời giải của bài toán. Thuật giải 2.1 chính là bước suy luận trên bản thân đối tượng, trong quá trình này, đối tượng sẽ tự động suy luận phát sinh các sự kiện thông qua hành vi nội tại của đối tượng. Bên cạnh đó, hệ thống cũng sẽ tìm lời giải thông qua việc phát sinh các sự kiện dựa trên các luật dẫn.

### 2.2.4 Các định lý cơ sở

**Định lý 2.2**:

1. Số các đối tượng được duyệt và các đối tượng mới được sinh ra trong thuật giải 2.1 và 2.2 là hữu hạn.
2. Số các quan hệ mới, các sự kiện mới được sinh ra trong thuật giải 2.1 và 2.2 là hữu hạn.

Định lý này cho chúng ta thấy rằng số lượng các đối tượng và sự kiện trong các thuật giải luôn hữu hạn, do đó các thuật giải này luôn dừng sau hữu hạn bước.

**Định lý 2.3:** Danh sách Deduce trong thuật giải 2.1 thỏa mãn các điều kiện (1)(2)(3) trong định nghĩa 2.2b về tập bao đóng sự kiện của đối tượng.

Chứng minh:

\* Dựa trên cách xây dựng danh sách Deduce, dễ thấy danh sách này thỏa điều kiện (1) và (2)

\* Đặt Deduce = [r1, r2,…, rm].

Giả sử ∃ r∈RulesObj\{r1, r2,…, rm} và r áp dụng được trên rm(A).



Khi đó, xét các biến trong vòng lặp của thuật giải 2.1:

H = rm(A)

Result = rm(A) ∪ v(r) ≠ H

Do đó theo điều kiện dừng của vòng lặp thì r ∈ Deduce (vô lý)

Vậy danh sách Deduce thỏa điều kiện (3)

**Bổ đề:** Cho mô hình tri thức quan hệ K = (C, H, R, Rules) và bộ (O, F) với O là tập các đối tượng và F là tập các sự kiện. Khi đó:

a/ Tồn tại một tập hợp lớn nhất L(O, F) thỏa mãn bài toán (O, F) → L(O, F) là giải được, nghĩa là:

Bài toán (O, F) → L(O,F) giải được

Và ∀ G, nếu bài toán (O, F) → G giải được thì G ⊆ L(O,F)

b/ Tồn tại một danh sách D = [d1, d2, ..., dk] sao cho D(F) = L(O, F)

**Định lý 2.4:** Cho miền tri thức toán tử K = (C, H, R, Rules) và bài toán P = (O, F) → G trên miền tri thức đó. Khi đó các mệnh đề sau là tương đương:

(i) Bài toán P là giải được.

(ii) G ⊆ L(O,F)

(iii) Tồn tại danh sách D như trong định nghĩa 2.6 sao cho G ⊆ D(F)

Chứng minh:

\* Sự tương đương của (i) và (iii) là do định nghĩa tính giải được của bài toán

\* (i) => (ii): Bài toán (O, F) → G giải được, nên theo bổ đề L(O,F) là tập hợp lớn nhất thỏa mãn bài toán (O, F) → L(O, F) , do đó G  L(O,F)

\* (ii) => (iii): Theo bổ đề, tồn tại một danh sách D = [d1, d2, ..., dk] sao cho D(F) = L(O, F) . Nhưng G  L(O,F), do đó G ⊆ D(F).

Định lý 2.4 chứng minh rằng thuật giải suy diễn tiến sẽ luôn cho ta kết quả của bài toán. Hơn nữa, các thuật giải 2.2 được thiết kế dựa trên chiến lược suy diễn tiến này, do đó định lý 2.4 cũng đã chứng minh cho tính hiệu quả giải thuật này.

## Ứng dụng xây dựng tri thức hình học phẳng:

Dựa trên tri thức hình học phẳng cấp THCS trong [24], miền tri thức này có thể được đặc tả dựa trên mô hình tri thức quan hệ, và chúng ta có thể thiết kế hệ hỗ trợ giải toán tự động trên miền tri thức.

### 2.3.1 Thiết kế cơ sơ tri thức:

Mô hình tri thức quan hệ để đặc tri thức hình học phẳng cấp THCS gồm bốn thành phần:

(C, H, R, Rules)

### *a) C–tập các khái niệm:*

Tập C gồm các khái niệm sau:

C(0) = {ℜ, ĐIỂM, ĐƯỜNG}

C(1) = {ĐOẠN, TIA}

C(2) = {GÓC, TAM\_GIÁC và các loại tam giác, TỨ\_GIÁC và các loại tứ giác, ĐƯỜNG\_TRÒN, …}

### *b) H–tập các quan hệ phân cấp:*

Giữa các khái niệm trong C, ta có các quan hệ phân cấp sau:

+ Các quan hệ phân cấp giữa các loại tam giác.

+ Các quan hệ phân cấp giữa các loại tứ giác

### *c) R–tập các quan hệ:*

Giữa các khái niệm trong C, ta có một số các quan hệ sau:

* Quan hệ trên khái niệm c: là quan hệ giữa các đối tượng thuộc cùng một khái niệm:

+ Quan hệ bằng nhau giữa các đoạn thẳng

+ Quan hệ bằng nhau giữa các tam giác

+ Quan hệ đồng dạng giữa các tam giác.

+ Quan hệ VUÔNG\_GÓC (⊥): giữa hai đường thẳng, giữa hai đoạn thẳng.

* Relations on ci × cj: là quan hệ hai ngôi giữa các đối tượng thuộc các khái niệm khác nhau:

+ Quan hệ TRUNGĐIỂM: giữa một điểm và một đoạn thẳng.

+ Quan hệ GIAO: giữa một đường thẳng và một đoạn thẳng. Trong trường hợp này, mệnh đề “e là giao điểm của a và b” được kí hiệu như sau:

e = a GIAO b

với e, a, b ∈ Σ\* và Dom(e) = IĐIỂM, Dom(a) ∈ {IĐƯỜNG, IĐOẠN}, Dom(b) ∈ {IĐƯỜNG, IĐOẠN}

### *d) Rules–tập các luật*

Hầu hết các định chất, mệnh đề, định lý trong kiến thức hình học phẳng có thể được biểu diễn thành các luật dẫn. Một số các luật như sau:

{a: ĐOẠN, b: ĐOẠN, c: ĐOẠN, a ⊥ c, b ⊥ c}  {a // b}

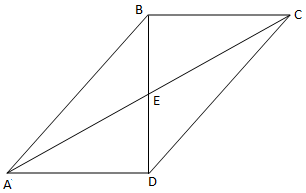
{O: ĐIỂM, AB: ĐOẠN, CD: ĐOẠN, O = AB GIAO CD}{AOC = BOD, AOD = BOC}.

{ABC: TAM\_GIÁC, M: ĐIỂM, N: ĐIỂM, M là TRUNGĐIỂM AB, N là TRUNGĐIỂM AC}{}

### 2.3.2 Thiết kế động cơ suy diễn của hệ thống

Bài toán trong tri thức hình học phẳng được mmô hình hóa như sau:

(O,F) → G

Ví dụ 2.4: Cho tứ giác ABCD, đường chéo BD vuông góc với BC và AD. Điểm E là giao điểm cuaả AC và BD. Biết rằng AE = EC.

Chứng minh: BE = ED

+ Mô hình bài toán:

O = {ABCD: TỨ\_GIÁC, E: ĐIỂM}

F = {BD ⊥ BC, BD ⊥ AD,

E = BD GIAO AC,

AE = EC

}

G = {BE = ED}

Động cơ suy diễn của hệ thống được xây dựng dựa trên các thuật giải 2.1 và 2.2. Suy diễn của hệ thống mô phỏng phương pháp suy nghĩ của con người trong quá trình tìm lời giải. Lời giải của chương trình rõ ràng, từng bước.

+ Lời giải của chương trình:

**Bước 1:** { BD ⊥ BC, BD ⊥ AD}

🡪 {BC // AD}

Bởi “Dấu hiệu nhận biết hai đường thẳng song song”

**Bước 2:** {ABCD: TỨ\_GIÁC, BC // AD}

🡪 {DBC = BDA}

bởi “Tính chất của tứ giác”

**Bước 3:** { E = BD GIAO AC}

🡪 {AED = BEC }

bởi “Tính chất góc đối đỉnh”

**Bước 4:** { AED: TAM\_GIÁC, BEC: TAM\_GIÁC,

EDA = EBC, AE = EC, AED = BEC}

🡪 {AED = BEC}

bởi “Dấu hiệu nhận biết tam giác bằng nhau”

**Bước 5:** { AED = CEB } 🡪 {DE = BE}

bởi “Tính chất tam giác bằng nhau”

## 2.4 Kết luận

Trong chương này, chúng tôi đã trình bày một phương pháp xây dựng mô hình tri thức quan hệ. Mô hình này được xây dựng dựa trên các cấu trúc toán học, đặc biệt là cấu trúc tập hợp và các cấu trúc đại số khác. Đồng thời dựa trên việc hình thức hóa của mô hình, chúng tôi cũng đã mô hình hóa các bài toán trên miền tri thức cũng như định nghĩa về lời giải của bài toán. Trên cơ sở đó, các thuật giải giải uqyết các vấn đề trên mô hình cũng được thiết kế, đồng thời tính dừng và hiệu quả của các thuật giải cũng được khẵng định thông qua các định lý cơ sở.

Hơn nữa, trên cơ sở mô hình tri thức quan hệ được xây dựng, chúng tôi cũng đã ứng dụng đặc tả một phần kiến thức hình học phẳng cấp THCS và thiết kế chương trình hỗ trợ giải tự động các bài tập trong miền tri thức này. Chương trình cho lời giải từng bước, tự nhiên, gần gũi và tương tự như cách giải của con người.

# CHƯƠNG 3: XÂY DỰNG MÔ HÌNH TRI THỨC TOÁN TỬ

Các kết quả trong chương này được công bố trong [CĐ3, CĐ4, CĐ5]

## 3.1 Mô hình tri thức toán tử:

Trong các ứng dụng thực tế, tri thức về các toán tử giữa các đối tượng trong tri thức là thành phần rất quan trọng, đặc biệt là các miền tri thức tính toán. Tuy nhiên, trong các mô hình tri thức hiện nay vẫn chưa đề cập đến các vấn đề liên quan đến toán tử như: đặc tả thành phần toán tử, quy tắc xác định toán tử. Trong mục này, chúng tôi sẽ trình bày một mô hình biểu diễn tri thức toán tử, đồng thời cũng khảo sát và giải quyết một số vấn đề liên quan đến thành phần toán tử.

**Định nghĩa 3.1:** Mô hình biểu diễn tri thức toán tử, gọi là *Ops-model*, là một bộ gồm 4 thành phần:

**K = (C, R, Ops, Rules)**

Trong đó:

* **C** là tập các khái niệm của miền tri thức.
* **R** là tập các quan hệ giữa các khái niệm trong tri thức, mỗi quan hệ này là một quan hệ hai ngôi giữa hai khái niệm trong tập C.
* **Ops** là tập các toán tử. Trong bài báo này chúng tôi chỉ xét toán tử hai ngôi trên các khái niệm trong tập C, cùng với việc khảo sát các tính chất của toán tử: đối xứng, kết hợp, phần tử trung hòa.
* **Rules** là tập các luật, các luật trong mô hình này được phân thành hai loại: luật dưới dạng luật dẫn và luật dưới dạng phương trình.

### 3.1.1. Tập C – Tập các khái niệm:

Mỗi khái niệm trong C là một lớp các đối tượng. Các đối tượng này có một cấu trúc xác định, đồng thời chúng cũng được trang bị một số các hành vi để giải quyết các vấn đề nội tại bên trong chúng.

*a/ Cấu trúc đối tượng*

Mỗi khái niệm trong C là một lớp các đối tượng được mô hình bởi bộ :

(*Attrs, EqObj, RulesObj*)

*Attrs* = {x1, x2, … ,xn} là tập các thuộc tính của đối tượng, trong đó xi là các biến đơn giản, nghĩa là các biến có giá trị là kiểu số thực hoặc là các kiểu không có cấu trúc.

*EqObj* = {e1, e2, … ,em } là tập các phương trình được gọi là quan hệ tính toán, các phương trình này biểu diễn quan hệ giữa các thuộc tính trong khái niệm.

*RulesObj* = {r1, r2, … ,rp}là tập các luật dẫn của đối tượng, mỗi luật rk này có dạng: u → v, với u ⊆ Attrs, v ⊆ Attrs và u ∩ v = Ø

Ví dụ 3.1: Trong tri thức về Dòng điện một chiều (Direct Circuit - DC), khái niệm MẠCH ĐIỆN bao gồm các thành phần sau:

*Attrs* = {R,U,I,P}: Tập các thuộc tính của MẠCH ĐIỆN

R,U,I,P: lần lượt là các biến có giá trị thực biểu diễn giá trị điện trở, hiệu điện thế, cường độ dòng điện, công suất của mạch điện.



*RulesObj*= { }

*b/ Hành vi của đối tượng*

Bên cạnh cấu trúc, một đối tượng trong tri thức còn có các hành vi sau để giải quyết các vấn đề nội tại của nó:

1/ Xác định bao đóng của các sự kiện trong đối tượng.

2/ Cho biết lời giải của việc xác định thuộc tính của đối tượng từ các thuộc tính đã biết.

3/ Tính toán trên đối tượng.

Trong các bài toán trên, thì việc xác định bao đóng của các sự kiện trong đối tượng chính là bài toán cơ sở để có thể giải quyết các bài toán sau. Vì vậy, trong báo cáo, chúng tôi chỉ đề cập đến hành vi này của đối tượng.

**Định nghĩa 3.2:** *Bao đóng của các sự kiện trong đối tượng*

Cho đối tượng Obj = (Attr, EqObj, RuleObj ) là một đối tượng của khái niệm trong C, và A là tập các sự kiện của Obj, các sự kiện trong A được phân loại như trong định nghĩa 2.4

a/ Nếu e ⊂ EqObj: khi đó e là hệ phương trình giữa k biến có kiểu giá trị là số thực {x1,x2,…,xk} ⊆ Attr

e *áp dụng được trên A* khi và chỉ khi từ các sự kiện loại 4 và 5 trong A, ta có:

+ e có thể giải để xác định giá trị các biến {x1,x2,…,xk}.

Đặt e(A) = A {x1,x2,…,xk }

+ HOẶC từ e có thể sinh ra các quan hệ mới dưới dạng phương trình giữa các biến {x1,x2,…,xk}

Đặt e(A) = A subs(A, e) với subs(A, e) là các phương trình trong e, trong đó có một số biến được thay bằng các giá trị từ các sự kiện loại 3 trong A.

b/ Nếu r ∈ RuleObj: r là luật dẫn có dạng: u(r) → v(r)

r *áp dụng được trên A* khi và chỉ khi u(r) ⊆ A. Đặt r(A) = A v(r)

c/ Đặt:



Obj.Closure(A) được gọi là bao đóng của tập sự kiện A trong đối tượng Obj. Thuật giải xác định Obj.Closure(A) sẽ được trình bày trong thuật giải 3.1 ở mục 3.2

### 3.1.2 Tập R – tập các quan hệ:

Thành phần này biểu diễn các quan hệ giữa các khái niệm trong miền tri thức. Mỗi quan hệ trong tập R là một quan hệ hai ngôi trên Ci × Cj , với Ci, Cj là các khái niệm trong tập C. Trong trường hợp Ci = Cj, các tính chất của quan hệ trên Ci sẽ được khảo sát: tính chất phản xạ, đối xứng, phản xứng, bắc cầu. Mỗi quan hệ được đặc tả như sau:

relation-def ::= RELATION <name>

ARGUMENT: argument-def+

PROPERTY: prob-type;

ENDRELATION

argument-def ::= name, <name>: type

prob-type ::= phản xạ | đối xứng | phản xứng | bắc cầu

Ví dụ 3.2: Trong miền tri thức Đại số vector, quan hệ VUÔNG GÓC giữa hai vector được biểu diễn như sau:

RELATION < VUÔNG GÓC> :

ARGUMENT: VECTOR, VECTOR

PROPERTY: đối xứng

ENDRELATION

### 3.1.3 Tập Ops – tập các toán tử:

Thành phần này biểu diễn các toán tử giữa các đối tượng trong miền tri thức, các toán tử cung cấp cho tri thức một phương pháp tính toán hiệu quả trong quá trình xử lý tri thức. Một toán tử trong mô hình Ops-model là một ánh xạ: Ci × Ci → Ci , với Ci là một khái niệm trong tập C. Bên cạnh đó, các tính chất toán tử được khảo sát trong bài báo này là: tính đối xứng, kết hợp, có phần tử trung hòa. Toán tử được đặc tả như sau:

operator-def ::= OPERATOR <name>

ARGUMENT: argument-def+

RETURN: return-def;

PROPERTY: prob-type;

ENDOPERATOR

argument-def ::= name, <name>: type

return-def ::= name : type

prob-type ::= giao hoán | kết hợp | có phần tử trung hòa

**Định nghĩa 3.3:** Một biểu thức giữa các đối tượng được định nghĩa như sau:

expr ::= object | expr <operator> expr

*Quy tắc xác định toán tử:*Tập luật Rules của mô hình tri thức toán tử được phân thành hai loại: luật xác định toán tử và luật suy diễn của miền tri thức. Do đó, kết quả của một toán tử trong tập Ops chính là một đối tượng của một khái niệm trong tập C, và giá trị các thuộc tính của đối tượng này được xác định dựa vào luật xác định toán tử trong thành phần Rules của mô hình tri thức.

Ví dụ 3.3: Trong tri thức về Dòng điện một chiều, toán tử “//” biểu diễn việc mắc song song giữa hai mạch điện được biểu diễn như sau:

OPERATOR < // > :

ARGUMENT: MẠCH ĐIỆN, MẠCH ĐIỆN

RETURN: MẠCH ĐIỆN

PROPERTY: giao hoán, kết hợp

ENDOPERATOR

(MẠCH ĐIỆN là khái niệm được định nghĩa trong ví dụ 3.1)

Giá trị toán tử này được xác định dựa vào luật R2 trong mục 3.4.2

### 3.1.4 Tập Rules – tập các luật của miền tri thức:

Trong báo cáo này, mỗi luật suy diễn r  Rules là một trong hai dạng sau:

**Rules = Rulededuce Ruleequation**

* Trường hợp 1: r  Rulededuce.Trong trường hợp này, r là một luật dẫn của miền tri thức có dạng sau:

u(r) = {f1, f2,…,fp}→ {q1,q2,…qk} = v(r)

Trong đó, fi và qi là các sự kiện của mô hình, với các loại sự kiện được phân loại như sau:

**Định nghĩa 3.4:** Phân loại các sự kiện của mô hình

|  |  |  |  |
| --- | --- | --- | --- |
| **Loại sự kiện** | **Ý nghĩa** | **Đặc tả** | **Ví dụ** |
| 1 | Cho biết thông tin về loại của đối tượng | <obj> : <type> | D : ĐIỆN TRỞ  AB : MẠCH ĐIỆN  u: vector |
| 2 | Cho biết sự xác định của một đối tượng hoặc thuộc tính của một đối tượng. | <obj>  <obj>.<attribute> |  |
| 3 | Sự xác định của một thuộc tính hay một đối tượng thông qua một giá trị hay một biểu thức hằng | <obj> = <const expr>  <obj>.<attribute> = <const expr> | D.U = 12  AB. I = 0.2 |
| 4 | Sự kiện về sự bằng nhau giữa một đối tượng hay một thuộc tính với một đối tượng hay một thuộc tính khác. | <obj> = <obj>  <obj>.<attribute> = <obj>.<attribute> | AB. I = D.I |
| 5 | Sự kiện về sự giữa các biểu thức của đối tượng. | <expression> = <expression> | AB = (D1//D2)+D3 |
| 6 | Sự kiện về quan hệ giữa các đối tượng | <obj> <relation> <obj> | u VUÔNG GÓC v |

* Trường hợp 2: r  Ruleequation.Trong trường hợp này, r là luật dưới dạng phương trình biểu diễn quan hệ tính toán giữa các đối tượng hoặc giữa các thuộc tính của đối tượng, r có dạng:

g(o1, o2,…, ok) = h(x1, x2,…, xp)

với oi, xi là các đối tượng và g, h là các biểu thức giữa các đối tượng.

## 3.2 Mô hình bài toán và Thuật giải

### 3.2.1 Mô hình bài toán và Lời giải

*a/ Bài toán trong mô hình tri thức toán tử*

**Định nghĩa 3.5:** Cho miền tri thức toán tử K = (C, R, Ops, Rules), khi đó mô hình bài toán trên tri thức K là một bộ gồm 3 thành phần:

O = {O1, O2, . . ., On}, F = {f1, f2, . . ., fm},

G = { g1, g2, . . ., gk }.

Trong đó, tập O là tập các đối tượng thuộc các khái niệm được đặc tả trong C, tập F là tập các sự kiện giữa các đối tượng trong O, và G là tập các mục tiêu. Trong miền tri thức về toán tử, các mục tiêu của bài toán bao gồm: Rút gọn biểu thức giữa các đối tượng, Xác định một thuộc tính của đối tượng, Xác định giá trị của biểu thức giữa các đối tượng.

Bài toán được ký hiệu bởi **(O, F) → G**

Trong bài báo này, các bài toán được phân thành hai dạng sau:

+ Dạng 1: Lớp các bài toán được suy diễn dựa trên việc sử dụng các hành vi của đối tượng và các luật dẫn trong tập luật Rules của tri thức.

+ Dạng 2: Lớp các bài toán được suy diễn dựa trên việc sử dụng các hành vi của đối tượng và các luật dạng phương trình trong tập luật Rules của tri thức.

*b/ Lời giải của bài toán*

**Định nghĩa 3.6:** Cho miền tri thức K = (C, R, Ops, Rules ), đối tượng Obj **=** (Attr, EqObj, RuleObj) là một đối tượng thuộc một khái niệm trong C, luật r ∈ Rules và A là tập các sự kiện. Ta có:

a/ Đặt A|Obj = {f | f ∈ A và f là sự kiện liên quan đến các thuộc tính của đối tượng Obj}

Obj(A) = Obj.Closure(A|Obj)

Obj(A) được gọi là tập các sự kiện được suy diễn từ A bởi đối tượng Obj

b/ Nếu r Rulededuce: r là một luật dẫn có dạng u(r) → v(r)

r *áp dụng được trên A* khi và chỉ khi u(r) ⊆ A. Đặt r(A) = A v(r)

c/ Nếu rRuleequation: r là phương trình giữa k đối tượng {x1,x2,…,xk}, khi đó r có dạng g(x1,x2,…,xi) = h(xi+1,xi+2,…,xk)

r *áp dụng được trên A* khi và chỉ khi:

+ card(A {x1,x2,…,xk }) = k-1. Đặt r(A) = A {x1,x2,…,xk }

+ HOẶC tất cả các đối tượng trong left(r) (hoặc right(r)) đều đã được xác định. Đặt r(A) = A r

r(A) được gọi là tập các sự kiện có thể được suy diễn từ A bằng luật r.

Trong đó: card(X) hàm trả về số phần tử của tập X

left(expr) hàm trả về vế trái của đẳng thức expr

right(expr) hàm trả về vế phải của đẳng thức expr

**Định nghĩa 3.7:** Cho bài toán S = (O, F) → G trên mô hình Ops-model. Giả sử D = [d1,d2,…, dm] là danh sách các phần tử với dj∈ Rules hay dj∈ O. Gọi L là tập các sự kiện của giải thiết bài toán. Đặt: A0 = L, A1 = d1(A0), A2 = d2(A1) , …. Am = dm(Am-1) và D(L) = Am

Bài toán S được gọi là *giải được* khi và chỉ khi tồn tại một danh sách D sao cho G D(L). Khi đó, D được gọi là *lời giải* của bài toán S.

### 3.2.2 Thuật giải suy diễn trên đối tượng

**Thuật giải 3.1:** Trên miền tri thức Ops-model, cho đối tượng Obj = (Attr, EqObj, RuleObj) được đặc tả như trong mục 2.1.1 và A là một tập các sự kiện của Obj. Thuật giải sau sẽ xác định bao đóng của tập A bởi đối tượng Obj.

*Input:* Obj = (Attr, EqObj, RuleObj): đối tượng của miền tri thức Ops-model

A: Tập các sự kiện trên Obj

*Output:* Obj.Closure(A)

*Thuật giải:*

Result ← empty

NewEq ← EqObj

Solution\_found false;

**Do**

H ← Result

**Select** phương trình f trong EqObj và giải phương trình f:

**Update** đẳng thức mới f vào NewEq

**Update** sự kiện mới f vào Result

**Select** hai phương trình {f, g} trong EqObj và giải hệ e = {f, g}

**Update** kết quả giải hệ phương trình e vào Result

**Select** luật dẫn r trong RuleObj và áp dụng để sinh ra sự kiện mới

**Update** v(r) vào Result

**While** (H != Result)

Result = Result NewEq

Obj.Closure(A) = A Result

### 3.2.3 Thuật giải suy diễn tìm lời giải của bài toán

**Thuật giải 3.2:** Cho bài toán S = (O, F) → G như trong định nghĩa 3.1 của mô hình tri thức toán tử Ops-model, ta có thuật giải sau để tìm lời giải cho bài toán S:

*Input:* K = (C, R, Ops, Rules): Miền tri thức được biểu diễn bời Ops-model

S = (O, F) → G : Bài toán trên miền tri thức K

*Output:* Lời giải của bài toán S

*Thuật giải:* được xây dựng dựa trên chiến lược suy diễn tiến, trong đó các đối tượng tham gia vào quá trình suy diễn thông qua các hành vi của của chúng.

Solution empty list;

Solution\_found false;

**while** not(Solution\_found) **do**

**Use** đối tượng trong tập O và tập các sự kiện F để xác định bao đóng của mỗi đối tượng.

**if** (đối tượng Obj sinh ra sự kiện mới) **then**

**Update** tập sự kiện và đưa đối tượng Obj vào Solution;

**if** (goal G xác định) **then**

Solution\_found true;

**end if;**

continue;

**end if;**

**Search** một luật r trong tập Rules có thể áp dụng để suy diễn;

**if** (tìm thấy luật r) **then**

**Use** r để sinh ra sự kiện mới hoặc đối tượng mới;

**Update** tập sự kiện và đưa luật r vào Solution;

**if** (goal G xác định) **then**

Solution\_found true;

**end if;**

**end if;**

**end do;** { while }

**if** (Solution\_found) **then**

Solution là một lời giải của bài toán;

**else**

Không tìm thấy lời giải;

**end if;**

Trong thuật giải trên, tại bước “tìm luật r trong tập Rules của miền tri thức có thể áp dụng”, khi S là bài toán dạng 1 thì thuật giải sẽ ưu tiên tìm các luật dẫn trong tập Rulesdeduce trước, ngược lại, nếu S là bài toán dạng 2 thì thuật giải sẽ ưu tiên tìm các luật dạng phương trình trong tập Rulesequation trước.

Các thuật giải cho mô hình tri thức toán tử thể hiện tư duy của con người trong quá trình suy diễn, tìm lời giải của bài toán. Thuật giải 3.1 chính là bước suy luận trên bản thân đối tượng, trong quá trình này, đối tượng sẽ tự động suy luận phát sinh các sự kiện thông qua hành vi nội tại của đối tượng. Bên cạnh đó, hệ thống cũng sẽ tìm lời giải thông qua việc phát sinh các sự kiện dựa trên các luật dẫn và việc giải các phương trình thông qua các luật phương trình của tri thức trong các thuật giải 3.2.

### 3.2.4 Đánh giá

**Định lý 3.1**:

1. Số các đối tượng được duyệt và các đối tượng mới được sinh ra trong thuật giải 3.1 và 3.2 là hữu hạn.
2. Số các quan hệ mới cũng như các phương trình mới được sinh ra trong thuật giải 3.1 và 3.2 là hữu hạn.

Định lý này cho chúng ta thấy rằng số lượng các đối tượng và sự kiện trong các thuật giải luôn hữu hạn, do đó các thuật giải này luôn dừng sau hữu hạn bước.

**Bổ đề 3.1:** Cho miền tri thức toán tử K = (C, R, Ops, Rules) và giả thiết (O, F) của bài toán trên mô hình. Gọi L là một tập các sự kiện trong giả thiết.

a/ Tồn tại một tập hợp lớn nhất  thỏa mãn bài toán (O, F) →  là giải được, nghĩa là:

Bài toán (O, F) →  giải được

Và ∀ G, nếu bài toán (O, F) → G giải được thì G ⊆ 

b/ Khi đó, tồn tại một danh sách D = [d1, d2, ..., dk] sao cho D(L) = 

**Định lý 3.2:** Cho miền tri thức toán tử K = (C, R, Ops, Rules) và bài toán S = (O, F) → G trên miền tri thức đó. Gọi L là tập các sự kiện của giả thiết bài toán. Khi đó các mệnh đề sau là tương đương:

(i) Bài toán S là giải được.

(ii) G ⊆ 

(iii) Tồn tại danh sách D như trong định nghĩa 3.3 sao cho G ⊆ D(L)

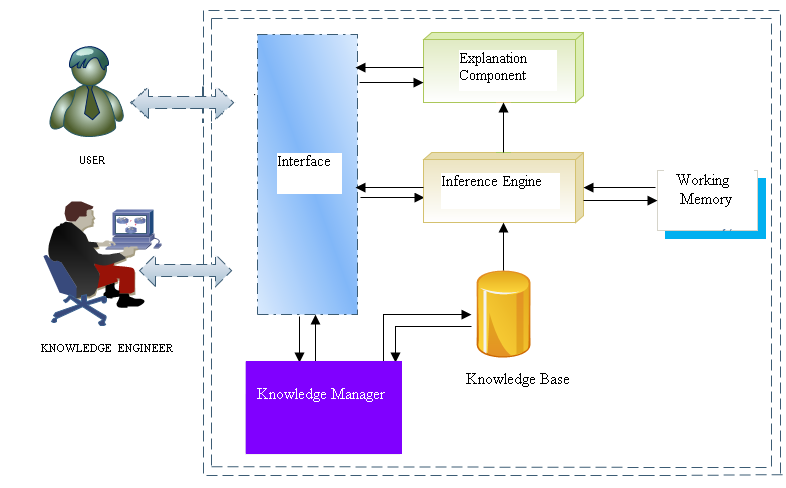
Định lý 3.2 chứng minh rằng thuật giải suy diễn tiến sẽ luôn cho ta kết quả của bài toán. Hơn nữa, các thuật giải 3.1 và 3.2 được thiết kế dựa trên chiến lược suy diễn tiến này, do đó định lý 3.2 cũng đã chứng minh cho tính hiệu quả của các giải thuật này.

Các thuật giải đều được suy diễn dựa trên một số hữu hạn các luật trong tri thức, do đó quá trình tìm kiếm sẽ dừng sau hữu hạn bước suy luận. Với việc so khớp các sự kiện sinh ra với mục tiêu của bài toán nên thuật giải sẽ cho một lời giải đúng của bài toán.

## 3.3 Ứng dụng

### 3.3.1 Kiến trúc hệ giải bài toán thông minh

Kiến trúc của một hệ thống giải bài tập thông minh được trình bày trong [24, 25]. Một hệ thống giải bài tập thông minh là một hệ cơ sở tri thức, hỗ trợ việc truy vấn, tìm kiếm và giải các bài tập tự động trên một miền tri thức xác định. Hệ thống này có cấu trúc của một hệ chuyên gia, gồm các thành phần: cơ sở tri thức, động cơ suy diễn, bộ giải thích, bộ nhớ làm việc, quản lý tri thức và giao diện.



*Hình 1:* Kiến trúc hệ giải toán thông minh.

Dựa trên kiến trúc này của hệ giải bài toán thông minh, trong đó cơ sở tri thức của hệ thống được biểu diễn bằng mô hình tri thức toán tử, chúng tôi đã tổ chức cơ sở tri thức của hai miền tri thức: Dòng điện một chiều trong kiến thức vật lý cấp Trung học cơ sở và tri thức về Đại số vector. Trên các cơ sở tri thức này, cùng với các thuật giải 3.1 và 3.2, chúng tôi đã thiết kế động cơ suy diễn cho các hệ thống giải bài tập thông minh trên các miền tri thức đó.

### 3.3.2 Thiết kế cơ sở tri thức của miền tri thức Dòng điện một chiều:

Dựa trên cơ sở tri thức điện một chiều trong Sách giáo khoa vật lý cấp trung học cơ sở [27], miền tri thức này có thể được biểu diễn bởi mô hình tri thức toán tử như sau:

**(C, Ops, Rules)**

Trong miền tri thức này, thành phần biểu diễn quan hệ R đã được giản lược.

*C–Tập các khái niệm*

Tập hợp C gồm các khái niệm sau: “ĐIỆN TRỞ”, “ĐÈN”, “BIẾN TRỞ”, “MẠCH ĐIỆN” (gọi tắt là MẠCH), “TỤ ĐIỆN” (gọi tắt là TỤ), “NGUỒN”.

*Ops–Tập các toán tử giữa các khái niệm:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Toán tử | Ý nghĩa | Kiểu của toán hạng | Giá trị | Tính chất |
| + | Các mạch điện mắc nối tiếp với nhau | MẠCHݞ× MẠCH | MẠCH | giao hoán  kết hợp |
| // | Các mạch điện mắc song song với nhau | MẠCH × MẠCH | MẠCH | giao hoán  kết hợp |
| o | Các tụ điện mắc nối tiếp với nhau | TỤ × TỤ | TỤ | giao hoán  kết hợp |
| Ξ | Các tụ điện mắc song song với nhau | TỤ × TỤ | TỤ | giao hoán  kết hợp |

Trong miền tri thức này, các đối tượng thuộc khái niệm điện trở cũng có thể xem là một mạch điện có 1 điện trở, vì vậy chúng ta có thể áp dụng phép toán + và // cho các đối tượng điện trở và tính chất của hai phép toán này vẫn không thay đổi.

*Rules–Tập các luật của miền tri thức*

Luật của miền tri thức được phân thành 02 loại: luật suy diễn và phương trình

+ Luật suy diễn:

R1: *Luật về mạch điện nối tiếp*

{D1, D2, AB: Mạch điện, AB = D1 + D2}

→ {AB.V = D1.V + D2.V, AB.I = D1.I = D2.I, AB.R = D1.R + D2.R}

R2: *Luật về mạch điện song song*

{D1, D2, AB: Mạch điện, AB = D1 // D2}

→ {AB.V = D1.V = D2.U, AB.I = D1.I + D2.I,  }

R3: *Luật về tụ điện nối tiếp*

{Cap1, Cap2, Cap: Tụ điện, Cap = Cap1 o Cap2}

→ {Cap.Q = Cap1.Q = Cap2.Q, Cap.V = Cap1.V + Cap2.U,

}

R4: *Luật về tụ điện song song*

{Cap1, Cap2, Cap: Tụ điện, Cap = Cap1 Ξ Cap2}

→ {Cap.Q = Cap1.Q + Cap2.Q, Cap.V = Cap1.V = Cap2.U, Cap.C = Cap1.C + Cap2.C}

+ Luật phương trình:

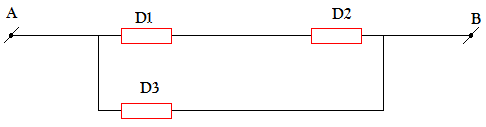
R5: D1 + D2 // D3 = (D1 + D2) // D3

R6: D1 // D2 + D3 = (D1 // D2) + D3

R7: C1 o C2 o C3 = (C1 o C2) o C3

R8: C1 Ξ C2 Ξ C3 = (C1 Ξ C2) Ξ C3

Bên cạnh cơ sở tri thức trên, mô hình bài toán của tri thức Điện một chiều được định nghĩa trong định nghĩa 3.1, và bộ suy diễn của hệ thống được xây dựng dựa trên các thuật giải 3.1 và 3.2. Động cơ suy diễn này mô phỏng cách suy nghĩ, giải quyết bài toán của con người và cho lời giải các bài toán một cách tự nhiên, từng bước.

Ví dụ:

*Bài toán S1:* Cho ba điện trở được mắc như hình vẽ. Điện trở D1 = 30 Ω, điện trở D3 = 60 Ω. Cường độ dòng điện qua mạch là 0.3A, qua điện trở D3 là 0.2A .

Tính cường độ dòng điện qua điện trở D1?

+ Đặc tả bài toán:

O:= { D1, D2, D3: ĐIỆN TRỞ ; AB: MẠCH}

F:= { AB = D1 + D2 // D3

D1.R = 30, D3.R = 60, D3.I = 0.2, AB.I = 0.3}

G:={D1.I}

+ Lời giải của chương trình:

1. D1 + D2 // D3 = (D1 + D2) // D3 bởi áp dụng luật R7
2. Gọi [EF, MẠCH], và EF = D1 + D2

AB = D1 + D2 // D3

D1 + D2 // D3 = (D1 + D2) // D3

🠪 AB = EF // D3

1. AB = EF // D3 → AB.I = EF.I + D3.I áp dụng luật “mạch điện song song” R2
2. EF = D1 + D2 → EF.I = D1.I áp dụng luật “mạch điện nối tiếp” R1
3. Từ { D3.I = 0.2, AB.I = 0.3, AB.I = EF.I + D3.I} → EF.I = 0.1

áp dụng luật “Giải phương trình”

1. Từ { EF.I = 0.1, EF.I = D1.I} → D1.I = 0.1

### 3.3.3 Thiết kế cơ sở tri thức của miền tri thức Đại số vector:

Dựa trên cơ sở tri thức Đại số vector [28], một phần kiến thức của miền tri thức này có thể được biểu diễn bởi mô hình tri thức toán tử như sau:

**(C, R, Ops, Rules)**

*C–Tập các khái niệm*

Tập hợp C gồm các khái niệm sau: “ĐIỂM”, “VECTOR”, “ĐOẠN THẲNG”, “TAM GIÁC”, “TỨ GIÁC”.

*R–Tập các quan hệ giữa các khái niệm:*

Tập hợp R gồm các quan hệ giữa các khái niệm trong miền tri thức Đại số vector, R gồm một số loại quan hệ sau:

+ Quan hệ TRUNG ĐIỂM: giữa một điểm và một đoạn thẳng.

+ Quan hệ TRỌNG TÂM: giữa một điểm và một tam giác.

+ Quan hệ THUỘC: điểm thuộc một đoạn thẳng

+ Quan hệ GIAO: giữa một đoạn thẳng và một đọan thẳng

+ Quan hệ VUÔNG GÓC: giữa hai đoạn thẳng, hai vector hoặc giữa một đoạn thẳng và một vector

*Ops–Tập các toán tử giữa các khái niệm:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Toán tử | Ý nghĩa | Kiểu của toán hạng | Giá trị | Tính chất |
| + | Cộng vector | Vector × Vector | Vector | giao hoán  kết hợp  có phần tử trung hòa |
| . | Tích vô hướng giữa hai vector | Vector × Vector | Số thực | giao hoán |
| o | Tích hữu hướng giữa hai vector | Vector × Vector | Vector |  |

*Rules–Tập các luật của miền tri thức*

Luật của miền tri thức được phân thành hai loại: luật suy diễn và luật phương trình

+ Luật suy diễn:

R1: {AB: Đoạn , M: Điểm, M TRUNG ĐIỂM AB} → { }

R2: {u, v: vector, u VUÔNG GÓC v} → {u.v = 0}

R3: {a,b,c: vector, c = a o b} → {c VUÔNG GÓC a, c VUÔNG GÓC b}

R4: {ABC: tam giác, G: điểm, G TRỌNG TÂM ABC} → {}

R5: {ABC: tam giác, M: điểm, G: điểm, M TRUNG ĐIỂM BC, G TRỌNG TÂM ABC}

→ {}

+ Luật phương trình:

R6: A,B: Điểm, 

R7: A,B,C: Điểm, 

R8: u: vector, u2 = u.u = |u|2

R9: u, v: vector, u . v = |u|.|v|.cos(u,v)

R10: u, v: vector, u o v = |u|.|v|.sin(u,v)

R11: u, v: vector, u o v = - v o u

Mô hình bài toán của tri thức Điện một chiều được định nghĩa trong định nghĩa 3.1, và bộ suy diễn của hệ thống được xây dựng dựa trên các thuật giải 3.1 - 3.2. Trong hệ thống này, chương trình sẽ giải quyết các bài toán chứng minh một đẳng thức vector thông qua việc biến đổi một biểu thức vector.

Ví dụ:

*Bài toán S2:* Cho bốn điểm A, B, C, D. Gọi J là trung điểm của AB.

Chứng minh rằng: 

+ Đặc tả bài toán:

O:= { A, B, C, D, J: Điểm}

F:= { J TRUNG ĐIỂM AB}

G:={ }

+ Lời giải của chương trình:



1. 
2.  áp dụng luật R6
3. 
4.  áp dụng luật R7
5. J TRUNG ĐIỂM AB →  áp dụng luật R1
6. Kết luận: Ta có điều phải chứng minh

## 3.4 Kết luận

Tri thức toán tử là một miền tri thức có nhiều ứng dụng, đặc biệt là các miền tri thức tính toán. Trong chương này, chúng tôi đã trình bày việc xây dựng mô hình tri thức toán tử. Mô hình này được hình thức hóa dựa trên các cấu trúc toán học. Đồng thời dựa trên việc hình thức hóa của mô hình, chúng tôi cũng đã mô hình hóa các bài toán trên miền tri thức cũng như định nghĩa về lời giải của bài toán. Trên cơ sở đó, các thuật giải giải uqyết các vấn đề trên mô hình cũng được thiết kế, đồng thời tính dừng và hiệu quả của các thuật giải cũng được khẳng định thông qua các định lý cơ sở.

Hơn nữa, trên cơ sở mô hình tri thức toán tử được xây dựng, chúng tôi cũng đã ứng dụng đặc tả một phần kiến thức Đại số vector và kiến thức về dòng điện một chiều. Trên cơ sở tri thức đó, chúng tôi đã xây dựng các chương trình hỗ trợ giải bài tập tự động cho các miền tri thức tương ứng.

# CHƯƠNG 4: KẾT LUẬN

Trong báo cáo này, chúng tôi đã trình bày một tiếp cận toán học trong việc xây dựng các mô hình tri thức quan hệ và tri thức toán tử. Phương pháp này xây dựng các mô hình dựa trên các cấu trúc toán học, đặc biệt là cấu trúc tập hợp và cấu trúc đại số khác. Các mô hình và khái niệm trong mô hình được xây dựng một cách hình thức. Bên cạnh đó, các vấn đề cần phải giải quyết trên mô hình cũng được mô hình hóa và phát biểu một cách tổng quát, cùng với các khái niệm về tính giải được, bước giải, lời giải của bài toán cũng được định nghĩa một cách rõ ràng. Việc xây dựng các mô hình tri thức quan hệ và toán tử bằng tiếp cận toán học này đã cho chúng ta một nền tảng lý luận trong việc thiết kế các mô hình biểu tri thức. Trên cơ sở đó, các thuật giải giải quyết các vấn đề cũng được thiết kế, đồng thời các thuật giải này cũng được chứng minh về tính dừng và tính hiệu quả thông qua các định lý cơ sở.

Hơn nữa, các mô hình tri thức quan hệ và tri thức toán tử cũng được ứng dụng trong việc đặc tả các miền tri thức toán học và vật lý như: Hình học phẳng, Đại số vector, Dòng điện một chiều. Từ đó, các chương trình hỗ trợ giải bài tập tự động cho các miền tri thức này cũng đã thiết kế xây dựng. Các chương trình cho lời giải các bài toán do người dùng nhập vào từng bước, tự nhiên.

Trong thời gian tới, chúng tôi sẽ nghiên cứu xây dựng mô hình tri thức toán tử một cách hoàn chỉnh hơn. Đồng thời, từ các kết quả về việc xây dựng mô hình tri thức quan hệ và tri thức toán tử, chúng tôi sẽ tiếp tục nghiên cứu một mô hình trong đó tích hợp cả hai thành phần quan hệ và toán tử. Mô hình tích hợp này sẽ được xây dựng dựa trên các cấu trúc toán học cũng như khái niệm liên quan đến mô hình cũng sẽ được định nghĩa một cách hình thức, rõ ràng.

# CÁC BÀI BÁO LIÊN QUAN ĐẾN CHUYÊN ĐỀ

[CĐ1] Đỗ Văn Nhơn, Nguyễn Đình Hiển, *Mô hình biểu diễn tri thức dạng quan hệ và Ứng dụng xây dựng hệ hỗ trợ giải toán thông minh*, Kỷ yếu Hội thảo quốc gia lần thứ XV: Một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông- Hà Nội, 03-04/12/2012, trang 191-195, In Quý III/2013.

[CĐ2] Hien D. Nguyen, Vuong T. Pham, Nhon V. Do, Model of Knowledge about Relations and Appliction, Proceeding of 2015 IEEE International Conference on Computing & Communication Technologies – Research, Innovation, and Vision for Future (RIVF 2015), pp. 13-16, Can Tho, Vietnam, January 2015.

[CĐ3] Van Nhon Do, Hien D. Nguyen, *Reducing model of COKB about Operators Knlwedge and Solving problems about Operators*, D. Camacho et al. (eds.), *New Trends in Computational Collective Intelligence*, pp. 39-49, Studies in Computational Intelligence 572, Springer (2014)

[CĐ4] Nguyễn Đình Hiển, Đỗ Văn Nhơn, *Mô hình tri thức toán tử và Ứng dụng xây dựng hệ cơ sở tri thức Điện một chiều*, Kỷ yếu Hội nghị Quốc gia lần thứ VII về Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR 2014), Thái Nguyên, ngày 19 - 20/06/2014, trang 274-283, DOI 10.15625/FAIR VII.2014-0351, ISBN: 978-604-913-300-8

[CĐ5] Nguyễn Đình Hiển, Đỗ Văn Nhơn, *Mô hình tri thức toán tử và Ứng dụng xây dựng hệ hỗ trợ giải bài toán thông minh*, Tạp chí Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam, ISSN: 0866-708X, Tập 52, số 4D, trang 60-76 (2014)

# TÀI LIỆU THAM KHẢO

[1] Stuaart Russell & Peter Norvig, Artificial Intelligent – A modern approach (third edition)”, Prentice Hall (2010)

[2] Frank van Harmelem & Vladimir & Bruce, *Handbook of Knowledge Representation*, Elsevier (2008)

[3] F. Lehmann, *Semantic Networks in Artificial Intelligence*, Elsevier Science Ltd (2008).

[4] John F. Sowa (2000), *Knowledge Representation - Logical, Philosophical, and Computational Foundations*, Inc. (2000)

[5] R. Davis, H. Shrobe, and P. Szolovits,“*What is a Knowledge Representation?*”,AI Magazine, 14(1):17-33, 1993

[6] E. Rich and K. Knight, “*Artificial Intelligence*", chapter 4-6, McGraw companies IncHill (2006).

[7] Nhon Do, Hien Nguyen, “*A reasoning method on Computation Network and Its applications*”, In Proceeding of 2011 International MultiConference of Engineers and Computer Scientists, (IMECS 2011), ISBN: 978-988-18210-3-4, pp. 137-141, Hongkong , March 2011.

[8] Nhon Do, Hien Nguyen, *A Reasoning method on Knowledge Base of Computational Ojects and Designing a System for automatically solving plane geometry problems*, In Proceeding of World Congress on Engineering and Computer Science 2011*,* (WCECS 2011)*,* ISBN: 978-988-18210-9-6, pp. 294-299, San Francisco, USA, October 2011.

[9] Tom Gruber, “*Ontology*”, Entry in *Ecyclopedia of Database Systems*, Ling Liu and M. Tamer Ozsu (Eds), pp. 1963-1965, Springer-Verlag (2009).

[10] Alexander Fronk, Rudolf Berghammer, *A Relation-Algebraic Approach to Liveness of Place/Transitions Net,* LNCS 5827, pp.103-118, Springer (2009)

[11] Thomas Bitter, Maureen Donnelly, *Computational ontologies of parthood, component hood, and contaiment*, Proceeding of International Joint Conferences on Artificail Intelligent 2005 (IJCAI 2005), Edinburgh, Scotland, UK.

[12] Helbig, H. (2005) Knowledge representation with multilayered extended semantic net-works: the MultiNet paradigm. URL – <http://pi7.fernuni-hagen.de/forschung/multinet/multinet_en.html>.

[13] Nguyễn Long Giang, Vũ Đức Thi, “*Một số thuật toán trên bảng quyết định sử dụng lý thuyết cơ sở dữ liệu quan hệ*”, Kỷ yếu Hội nghị: Một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông lần thứ XIV (@2011), trang 303-317, Cần Thơ (2011)

[14] Trần Minh Cảnh, Bùi Duy Tuấn, Trương Thị Thu Hà, Nguyễn Xuân Huy, “*Phân hoạch hệ suy dẫn*”, Kỷ yếu Hội nghị: Một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông lần thứ XIV (@2011), trang 418-426, Cần Thơ (2011)

[15] Chunyan Yang, Wen Cai, *Knowledge Representations based on Extension Rules*, Proceedings of the 7th World Congress on Intelligent Control and Automation, Chongqing, China (2008)

[16] Yousheng Tian, Yingxu Wang, Kai Hu, *A Kmowledge Representation Tool for Autonomous Machine Learning Based on Concept Algebra,* LNCS 5540, pp. 143-160, Springer (2009)

[17] Wang, Y., & Chiew, V., *On the cognitive process of human problem solving*, Cognitive Systems Research (2008), doi:10.1016/j.cogsys.2008.08.003

[18] Jānis Cīrulis*, An Algebraic Approach to Knowledge Representation*, Mathematical Foundations of Computer Science 1999, Lecture Notes in Computer Science Volume 1672, 1999, pp 299-309

[19] Jacques Calmet, Indra A. Tjandra, *Building bridges between knowledge representation and algebraic specification*, Methodologies for Intelligent Systems , Lecture Notes in Computer Science Volume 869, 1994, pp 295-304

[20] Erika Valencia, Jean-Louis Giavitto, *Algebraic Topology for Knowledge Representation in Analogy solving*, Proceeding of European Conferenc Artificial Intelligence (ECAI 1998), pp. 88-92, Brighton, England, 1998

[21] Jakub M. Tomczak, Jerzy Swiatek, *Knowledge acqusition method for Realational Knowledge Representaion* (2009)

[22] T. Plotkin, M. Knyazhansky, *Symmetries of Knowledge Base*, Annals of Mathematics and Artificial Intelligence (2012)

[23] Nhon V. Do, “*Ontology COKB for designing knowledge-based systems*”, Proceeding of 13th International Conference on Intelligent Software Methodologies, Tools, and Techniques (SOMET 2014), pp. 354-373, *New Trends in Software Methodologies, Tools and Techniquies,*Hamido Fujita et al. (eds.), IOS press.(2014)

[24] Nhon Van Do, “*Intelligent Problem Solvers in Education: Design Method and Applications, Intelligent Systems”*, Prof. Vladimir M. Koleshko (Ed.), ISBN: 978-953-51-0054-6, InTech, (2012). URL - <http://www.intechopen.com/books/intelligent-systems/intelligent-problem-solvers-in-education-design-method-and-applications>

[25] Nhon V. Do, Hien D. Nguyen, Thanh T. Mai, *Designing an Intelligent Problems Solving System based on Knowledge about Sample Problems,* Proceeding of 5thAsian conference on Intelligent Information and Database Systems (ACIIDS 2013), Kuala Lumpur, Malaysia, March 2013, LNAI 7802, pp. 465-475, Springer.

[26] Bộ Giáo dục và Đào tạo, 2010, “Sách giáo khoa và Sách bài tập hình học cấp Trung học cơ sở”, NXB Giáo dục.

[27] Bộ Giáo dục và đào tạo, “*Sách giáo khoa và Sách bài tập Vật lý lớp 9”*, NXB Giáo dục, 2012.

[28] Bộ Giáo dục và đào tạo, “*Sách giáo khoa và Sách bài tập Toán lớp 10*”, NXB Giáo dục, 2012.