**KNOWLEDGE-BASED PROBLEM SOLVING**

**SUY DIỄN**

**REASONING**

* **Introduction**
* Các phương pháp suy diễn tự động nhằm vận dụng kiến thức đã biết trong quá trình lập luận để giải quyết vấn đề, trong đó quan trọng nhất là các chiến lược điều khiển giúp phát sinh những sự kiện mới từ các sự kiện đã có.
* Cơ sở suy luận thực tế của con người thường gồm các loại sau:
* Suy diễn dạng diễn dịch (Deductive Reasoning): là phương pháp suy lận dựa vào các quy luật của tri thức để rút ra kết luận từ một hay nhiều mệnh đề.
* Suy diễn dạng quy nạp (Inductive Reasoning): là phương pháp suy luận đưa ra kết luận có tính khái quát từ những tri thức riêng lẻ, ít khái quát hơn.
* Suy diễn loại suy (Analogical Reasoning): đây là dạng suy luận dựa trên niềm tin vào sự tương tự hay sự giống nhau. Cách suy luận này sẽ căn cứ trên một hay nhiều tình huống tương tự đã biết để dẫn ra kết luận cho trường hợp hay tình huống đang xem xét.
* Dựa trên các loại suy luận ấy, trong ngành trí tuệ nhân tạo, đã đề xuất các dạng suy luận để sử dụng cho các mô hình biểu diễn tri thức:
* Chiến lược suy diễn tiến: từ tập các sự kiện đã biết, sử dụng các luật suy diễn trong miền tri thức để suy ra (hay sinh ra) các sự kiện mới, quá trình này tiếp diễn đến khi đạt được mục tiêu của bài toán hoặc là đến khi bị bế tắc.
* Chiến lược suy diễn lùi: từ mục tiêu của bài toán, áp dụng các luật suy diễn của tri thức để truy ngược về phần giả thiết của bài toán.
* Lập luận dựa trên tình huống: phương pháp tìm lời giải bài toán trên cơ sở hiệu chỉnh lời giải của các bài toán tương tự đã biết.
* Suy diễn dựa trên các heuristic, tri thức bài toán mẫu, và mẫu bài toán: quá trình suy diễn này sử dụng các quy tắc heuristic. Bên cạnh đó, việc nghiên cứu sử dụng các bài toán mẫu và mẫu bài toán, giúp cho quá trình suy diễn mô phòng được tư duy của con người trong giải quyết vấn đề, giúp hệ thống nhanh chóng đạt đến các sự kiện mục tiêu.
* **Vấn đề suy diễn (trên CSTT):**
* Có cơ sở tri thức **K** *đã được mô hình hóa*.
* Bài toán suy diễn:  
  + Cho trước tập sự kiện **GT**.  
  + Xét tập sự kiện mục tiêu **KL**.  
  Thực hiện suy diễn (tìm kiếm) để suy ra được KL từ giả thiết GT (sử dụng tri thức K).  
  **Ký hiệu bài toán:** GT KL
* Giả định: trong K có các luật r có thể áp dụng để sinh ra các sự kiện mới từ một số các sự kiện đã biết.

Trong ứng dụng cụ thể, thì GT thường có cấu trúc gồm:

1. Tập hợp các đối tượng: O = { o1, …, on }
2. Tập các sự kiện trên các đối tượng: F = { f1, …, fk }

Có thể viết GT = (O, F) (mạng đối tượng).

Ngoài cấu trúc cơ bản dang “mạng đối tượng”, cón có thể có những yếu tố khác tùy theo ứng dụng thực tế.

Về KL, một số dạng yêu cầu/mục tiêu thường gặp:

1. Chứng minh một sự kiện
2. Tìm kiếm (v/d: tìm mối liên hệ giữa 2 đối tượng)
3. Tính toán

**Ví dụ:**

1. Vấn đề giải bài toán suy diễn tính toán trên một tam giác

* Mô hình tri thức về một tam giác
* Giả thiết và mục tiêu của bài toán
* Tiêu chuẩn áp dụng luật, điều kiện ràng buộc giới hạn phạm vi

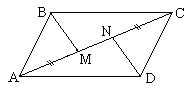
1. Vấn đề suy diễn trên một mạng tính toán, mạng các đối tượng tính toán
2. Vấn đề suy diễn trên CSTT có dạng ontology COKB (option)
3. Vấn đề suy diễn giải bài toán trên tri thức cụ thể như HHGT, HHKG, LG, Đ1C, ĐXC, HHVC, HHHC, v.v…

Ví dụ 1: HHP

* Đã có CSTT về HHP đã được xây dựng bao gồm các định lý, công thức, mệnh đề, … (rules)
* Cho một bài toán trong đó có một số đối tượng cùng với một vài sự kiện giả thiết, và yêu cầu (goal) chứng minh 2 tam giác bằng nhau.

Mẫu cụ thể:

Given the parallelogram ABCD. Suppose M and N are two points of segment AC such that AM = CN. Prove that two triangles ABM and CDN are equal (see figure 2).



First, we have three objects O1 (parallelogram ABCD), O2 (triangle ABM) and O3 (triangle CDN). The beginning relations, by declaration and by program’s detection, are as follows:

* O2.b = O3.b (edge AM = edge CN),
* M ∈ AC, N ∈ AC,
* O2.c = O1.a (edge AB = edge AB),
* O3.c = O1.a (edge CD = edge CD).

The program executes deducing and recognizes the lack of facts for conclusion. By using heuristic rules 4 and 7 above, it produces new objects relative to O2, O3 and O1: O4 (triangle ABC) and O5 (triangle CDA), and it detects the following relations:

* O4 = O5 (equality of two triangles),
* O4.A = O2.A (equality of the first angles),
* O5.A = O3.A (equality of the first angles),
* O4.B = O1.B, and so on.

Then the program will deduce facts enough for conclusion:

* O2.b = O3.b,
* O2.c = O3.c,
* O2.A = O3.A.

Ví dụ 2: Biến đổi biểu thức logic

* Bài toán rút gọn BT Logic:

+ Cho biểu thức E phụ thuộc theo các biến logic.

+ Tìm bt F thỏa: (1) F ⇔ E, (2) F đơn giản hơn E (tối giản càng tốt)

* Bài toán chứng minh bt Logic:

+ Cho trước các biểu thức E1, E2, …, En (GT)

+ Chứng minh biểu thức F.

* **Suy diễn tiến:**
* **Ý tưởng:**Xuất phát từ giả thiết, thực hiện quá trình (lặp) tìm luật áp dụng (vận dụng) để từ những sự kiện đang có suy ra (sinh ra) sự kiện mới cho tới khi đạt được mục tiêu hoặc tới khi không còn luật có thể áp dụng (sinh ra sự kiện mới).
* **Thuật giải tổng quát:  
  Ký hiệu:  
  + facts: tập sự kiện được ghi nhận (đã biết)  
  + CTDL mới: deduction\_step gồm 3 phần  
   (1) luật r được áp dụng  
   (2) tập các sự kiện A được sử dụng (A  facts)  
   (3) tập sự kiện mới B được sinh ra (suy ra).**

**Ghi chú:**

* Một bước giải có thể do áp dụng cùng lúc nhiều luật.
* Một bước giải có thể do áp dụng những quy tắc nào đó mà không phải luật trong K.
* Phân loại lời giải: lời giải (nói chung), lời giải không có bước thừa, lời giải “tốt”, lời giải “tối ưu”, lời giải “tự nhiên hay thông minh”.
* Châu+trang (CNTN): “bước giải thu gọn” 🡪 “lời giải thu gọn”

**+ solution: danh sách các deduction\_step dẫn tới kết quả mục tiêu từ giả thiết.**

**Algorithm:**

**Bước 1:**

**facts = GT; solution = [];  
Bước 2:**

**while not (KL  facts) do**

**2.1: Tìm luật r trong K thỏa:  
 r áp dụng được trên facts để sinh ra sự kiện mới  
 (tìm được r đồng thời xác định được một deduction\_step)**

**2.2: if (tìm được r trong bước 2.1)**

**step = deduction\_step tương ứng của r;**

**bổ sung các sự kiện mới vào tập facts;**

**thêm step vào solution;**

**else**

**kết thúc suy siễn, kết luận: không tìm được lời giải**

**Bước 3: Tìm được lời giải: solution là ds các bước giải.**

* **Minh họa quá trình suy diễn tiến dạng chuỗi các bước suy diễn:**

Quá trình suy diễn tiến được thể hiện thông qua sơ đồ dưới đây.



Trong đó:

*r*i(i=1..k) là các luật lần lượt được áp dụng.

*A*0 là tập các sự kiện ban đầu (haytập sự kiện giả thiết).



(*A*i là tập sự kiện có được sau khi áp dụng luật *r*i )

***Ví dụ 3.2:*** Với tri thức về tam giác, ta có các luật sau:





Khi đó, ví dụ 3.1 có thể được giải thông qua quá trình suy luận sau:



Với A0 = {C, a, b}

A1 = {C, a, b, S}

A2 = {C, a, b, S, ha}

* **Minh họa quá trình suy diễn tiến dạng “Cây suy diễn tiến”   
  (vét cạn)**

**??**

* **Những vấn đề kỹ thuật phải giải quyết khi vận dụng thuật giải suy diễn tiến ở trên:**

1. **Phân loại sự kiện, và định nghĩa sự hợp nhất của 2 sự kiện.**
2. **KL  facts hiểu theo nghĩa là qua sự hợp nhất các sự kiện**
3. **Tiêu chuẩn (hay điều kiện) mà luật “r áp dụng được trên facts để sinh ra sự kiện mới” ?**

* **Những nhận xét dẫn tới các vấn đề kỹ thuật khác:**

1. **solution thường là có các bước giải thừa (có sự kiện mới x được sinh ra là không cần thiết cho các bước suy diễn phía sau mà x cũng không phải là mục tiêu của bài toán), ta cần có thuật giải để loại các bước thừa:**

**input: GT🡪KL, solution**

**output: solution\_new**

**Algorithm:**

**??**

1. **Việc chọn luật r để áp dụng trên tập facts cần được xem xét nghiên cứu khi có nhiều luật có thể áp dụng.**
2. **Phép suy diễn vét cạn: xét tất cả các luật có thể áp dụng được ở mỗi bước suy diễn 🡪 quá trình suy diễn phải được ghi nhận lại dưới dạng cây suy diễn tiến.**

* **Suy diễn lùi: (xem sách chuyên khảo)**

**SUY DIỄN KẾT HỢP VIỆC SỬ DỤNG**

**HEURISTICS, MẪU BÀI TOÁN, BÀI TOÁN MẪU**

* **SUY DIỄN TIẾN/LÙI**

**Con người khi suy diễn tiến/lùi sẽ kết hợp với “những gì đó” để làm cho suy diễn có tính thông minh, hiệu quả:**

* **Kinh nghiệm**
* **Kỹ năng**
* **Khả năng trực giác, nhạy bén**
* **Tận dụng thông tin hữu ích bổ sung đối với vấn đề.**
* **Ghi nhớ những trường hợp khó hay đặc biệt, phổ biến để vận dụng khi cần thiết**
* **V.v…**
* **Khuyết điểm của thuật giải suy diễn tiến:**
* **Có những bước giải thừa trong lời giải được tìm thấy, số lượng thừa thường là nhiều.  
  🡪khắc phục: xây dựng thuật giải để loại bỏ bước thừa.**
* **Lời giải có được (sau khi loại bỏ bước thừa) chưa chắc là tối ưu (số bước giải là thấp nhất, chi phí tính toán thấp nhất, …)  
  🡪khắc phục:   
   C1:Kết hợp tiến và lùi sẽ tìm được lời giải có thể tốt hơn nhưng chưa chắc là tối ưu.  
   C2: lượng hóa hay số hóa (các tham số hay các trọng số) các tiêu chí tối ưu, và trong quá trình tìm lời giải ta phải tính toán trọng số của lời giải, có sự so sánh chọn lựa để có thể tìm được lời giải tối ưu.**

**Bài tập:**

1. **Trên mô hình tri thức hệ luật dẫn, xét bài toán suy diễn cơ bản; xây dựng thuật giải suy diễn tìm lời giải ngắn nhất theo số bước giải.**
2. **Trên mô hình tri thức hệ luật dẫn với trọng số về chi phí tính toán của mỗi luật:  
    (F, R, w)  
   Xây dựng thuật giải suy diễn tìm lời giải ngắn nhất theo trọng số.**

* **Khó thể hiện cách giải mang tính tự nhiên như con người.**
* **Độ phức tạp: cao (kém hiệu quả).**
* **Trường hợp đối các luật dạng phương trình, thì suy diễn tiến chưa thể hiện được khả năng kết hợp nhiều phương trình thành hệ phương trình để giải. Nói tổng quát hơn, thuật giải chưa thể hiện được tính thông minh, nhạy bén, linh hoạt trong tư suy suy luận của con người.**

**🡪khắc phục:**

* **Thu gọn lời giải: gộp nhiều bước giải thành 1 bước giải một cách hợp lý để thể hiện tính tự nhiên. Chẳng hạn, đối với các bước giải mà đối với con người là khá hiển nhiên hay tầm thương, không cần ghi ra.**
* **Sử dụng các heuristics, đặc biệt là các kinh nghiệm, trực giác, cảm tính, các thông tin hữu ích bổ sung thêm cho bài toán 🡪 thuật giải heuristic, metaheuristic.**
* **Sử dụng các mẫu bài toán.**
* **Sử dụng các bài toán mẫu.**

1. **Khái niệm**

* **Heuristics**: thông tin hữu ích bổ sung thêm, yếu tố kinh nghiệm, yếu tố cảm tính, trực giác, **các đặc điểm liên quan đến bài toán trong miền tri thức cụ thể**, v.v…
* **Mẫu bài toán** (problem): vấn đề hay bài toán thường thấy trong việc giải quyết vấn đề ứng dụng đặt ra, có tính cơ bản (độ phức tạp nhỏ, đơn giản), lời giải ngắn gọn.

**Ví dụ:**

1. Hệ giải các bài toán HHGT 3 chiều. Bài toán sau đây có thể xem là “mẫu bài toán”:

Với mặt phẳng (P), nếu biết 1 điểm M thuộc mặt phẳng (P), biết một đường thẳng (d) vuông góc với (P) thì ta có thể xác định được phương trình mặt phẳng (P).

1. Cân bằng phương trình phản ứng hóa học: …

* **Bài toán mẫu:** là bài toán thường là khó và có độ phức tạp cao mà chúng ta nên ghi nhớ, vì nếu chỉ dựa trên tri thức có trong CSTT và phương pháp suy diễn thông thường thì việc tìm ra lời giải là khó khăn và tốn nhiều chí phí. Việc nhớ dạng bài toán và lời giải hay cách giải của bài toán sẽ giúp ta nhanh chóng tìm được lời giải của bai toán.

**Ví dụ:**

1. Một bài tập HHGT 3D (loại khó) trong các đề thi tuyển sinh đại học.
2. Một bài tập về hóa học (loại khó) trong đề thi thi tuyển sinh đại học.

* Method-base:

1. **Suy diễn tiến/lùi với việc sử dụng heuristics**

* **Các bước thực hiện thiết kế:**
* **Xác định, tìm các heuristics**
* **Xây dựng thuật giải suy diễn dùng heuristics**
* **Phân tích đánh giá hiệu quả: lý thuyết, hay là thực nghiệm.**
* **Xác định, tìm các heuristics**
* **Tìm luật áp dụng: Xác lập độ ưu tiên chọn luật, thu hẹp tập luật. Các căn cứ: tần suất sử dụng luật, dạng bài toán (liên quan đến giả thiết và mục tiêu bài toán), kinh nghiệm chuyên gia.   
  🡪 Xây dựng các quy tắc heuristic cho việc chọn luật áp dụng.**
* **Tập sự kiện: phân loại sự kiện, tổ chức phân lớp tập sự kiện, xây dựng các quy tắc ưu tiên theo luật đang xét, giới hạn tạp sự kiện, v.v…   
  🡪 các quy tắc heuristic hỗ trợ thêm cho việc chọn luật, giảm thiểu tính toán trong quá trình so khớp sự kiện.**
* **Về mặt kỹ thuật: tối ưu hóa kỹ thuật so khớp sự kiện, và so khớp tập sự kiện.**
* **Thuật giải:**

**Algorithm:**

**Bước 1:**

**facts = GT; solution = [];  
Bước 2:**

**while not (KL  facts) do**

**2.1: Dựa trên các quy tắc heuristic cho việc chọn luật áp dụng, tìm luật r trong K thỏa:  
r áp dụng được trên facts để sinh ra sự kiện mới  
(tìm được r đồng thời xác định được một deduction\_step)**

**2.2: if (tìm được r trong bước 2.1)**

**step = deduction\_step tương ứng của r;**

**bổ sung các sự kiện mới vào tập facts;**

**thêm step vào solution;**

**else**

**kết thúc suy siễn, kết luận: không tìm được lời giải**

**Bước 3: Tìm được lời giải: solution là ds các bước giải.**

* **Ví dụ áp dụng:**

1. **Suy diễn tiến/lùi với việc sử dụng các mẫu bài toán**
2. **Suy diễn tiến/lùi với việc sử dụng các bài toán mẫu**
3. **Suy diễn tiến/lùi với việc sử dụng kết hợp H, SP, PP**

**SUY DIỄN GIẢI VẤN ĐỀ (BÀI TOÁN) TRÊN COKB**

1. **Cơ sở tri thức các đối tượng tính toán (COKB)**

* Mô hình
* Ngôn ngữ đặc tả
* Tổ chức lưu trữ
* Sự kiện và so khớp sự kiện

1. **Mô hình bài toán tổng quát và phân loại**

* Mô hình chung: (O,F)G
* Bài toán có tham số: (O,F)+TG  
  T: tập tham số
* Phân loại dựa trên: mục tiêu, giả thiết

1. **Phương pháp suy diễn tiến “thông minh” giải bài toán**

* Suy diễn tiến, và loại bỏ bước thừa
* Suy diễn tiến dựa trên các heuristics
* Sử dụng mẫu bài toán trong suy diễn
* Sử dụng bài toán mẫu trong suy diễn
* Thu gọn lời giải thể hiện lời giải “tự nhiên và thông minh”

**Tham khảo từ các tài liệu:**

1. Nhon V. Do, Ontology COKB for designing knowledge-based systems.New Trends in Software Methodologies, Tools and Techniques, H. Fujita et al. (Eds.). IOS Press, 09/2014
2. Nhon V. Do,*Ontology COKB for Knowledge Representation and Reasoning in Designing Knowledge-based Systems*.Communications in Computer and Information Science 513 *–* Intelligent Software Methodologies, Tools and Techniques – SoMeT 2014 Revised Selected Papers, Kamido Fujita & Ali Selamat (Eds.), pp. 101-118.Springer International Publishing Switzerland 2015.
3. Van Nhon Do, Diem Nguyen, Intelligent Problem Solving about Functional component of COKB model and Application. New Trends in Computational Collective Intelligence, David Camacho et al. (Eds.), pp. 27-37. Springer International Publishing Switzerland, 2015.
4. Nhon V. Do, Hien D. Nguyen and Thanh T. Mai, *Reasoning Method on Knowledge about Functions and Operators*. International Journal of Advanced Computer Science and Applications (IJACSA), 6(6), pp. 156 – 168 (2015).
5. Nhon V. Do, Thanh T. Mai, *Intelligent Problem Solving based on COKB Model*. Proceeding of 2015 IEEE International Conferene on Knowledge and Systems Engineering (KSE 2015), pp.216-221, Ho Chi Minh City, Vietnam, October 2015.

**MỘT SỐ ỨNG DỤNG**

* Giới thiệu một số đề tài tham khảo:

1. Hệ hỗ trợ học toán. ([P.2012.1](file:///C:\Users\DVNHON\AppData\Roaming\Projects\2012.1\BaoCao-NghiemThu))
2. Hệ quản lý tài liệu học tập CNTT. ([P.2010.2](file:///C:\Users\DVNHON\AppData\Roaming\Microsoft\BieuDienTriThuc%20-%20SuyLuan\Thuong_Bao%20cao%20nghiem%20thu%20SDB%20(18-04-2013).pdf), [paper](file:///C:\Users\DVNHON\AppData\Roaming\Microsoft\BieuDienTriThuc%20-%20SuyLuan\ACIIDS2013,Part1,%20LNAI%2078020476.pdf))
3. Hệ thu thập và tìm kiếm tin trên các báo điện tử. ([paper](file:///C:\Users\DVNHON\AppData\Roaming\Microsoft\BieuDienTriThuc%20-%20SuyLuan\KSE-News%20Aggregating%20System%20Supporting%20Semantic%20Processing%20Based%20on%20Ontology.pdf))
4. Hệ chẩn đoán bệnh. ([paper](file:///C:\Users\DVNHON\AppData\Roaming\Microsoft\BieuDienTriThuc%20-%20SuyLuan\IJCSI-10-4-2-308-316.pdf))
5. [7 luận văn](file:///I:\Backup\HuongDanLuanVan\FINISHED) vừa bảo vệ 02/2015.