

Lập trình ràng buộc

- Giới thiệu -

Christophe Lecoutre

CRIL-CNRS UMR 8188
Universite d'Artois
Lens, Pháp

Tháng 1 năm 2021

Đề cương

1 CP kể từ khi bạn còn là một đứa trẻ

2 Ứng dụng CP

3 CP ngũ cǎnh hóa

4 CSP và COP

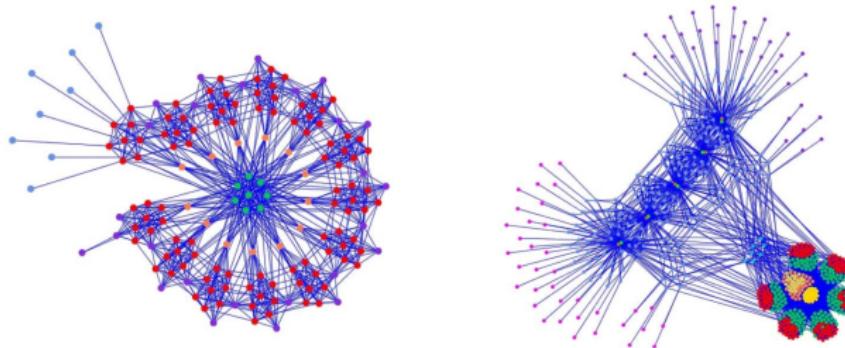
5 ba vấn đề đơn giản

Lập trình ràng buộc

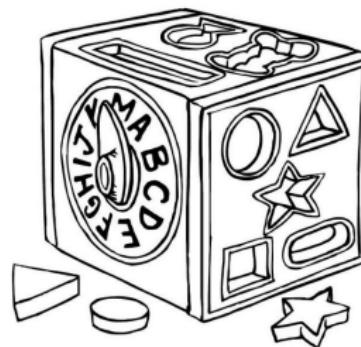
là về mô

hình hóa và giải quyết

Các vấn đề theo Ràng buộc



Chắc chắn, vấn đề đầu tiên bạn đã giải quyết!



Đây là một vấn đề phù hợp!

Hãy nhớ về câu đố đầu tiên của bạn!



Đãy nào gồm bốn số nguyên liên tiếp có tổng bằng 14?

Chúng ta cần bốn biến số nguyên:

- $x_1 \quad N, \quad x_2 \quad N, \quad x_3 \quad N, \quad x_4 \quad N$

Chúng tôi cần một số ràng buộc bình đẳng:

- $x_1 + 1 = x_2$
- $x_2 + 1 = x_3$
- $x_3 + 1 = x_4$
- $x_1 + x_2 + x_3 + x_4 = 14$

Khá thường xuyên, bạn phải đổi mặt với các Vấn đề về Tối ưu hóa!

Một phù thủy có thể chế tạo ra hai loại độc dược: • loại trước để truyền cảm hứng cho tình yêu • loại sau để khôi phục tuổi trẻ

Các lọ thuốc của luận án bao gồm những giọt nước chảy của cúc (Ut), đôi cánh của rồng (Ud) và bột của nhện (Mỹ), với tỷ lệ như sau: • 1 lọ thuốc cho tình yêu

cần 3 Ut , 1 Ud và 1 Us • 1

lọ thuốc dành cho tuổi trẻ cần 2 Ut , 3 Ud và 2 Us



Các vấn đề về tối ưu hóa khá thường xuyên!

Mô hình PyCSP3 đầu tiên của bạn

```
từ pycsp3 nhập *
```

```
# x là số lợ thuόc tinh yêu ma thuật x = Var (phạm vi  
(400))
```

```
# y là số lợ thuόc thanh nién ma thuật y = Var (phạm vi  
(400))
```

```
thỏa mãn
```

```
    (3 * x + 2 * y <= 800 ,  
     x + 3 * y <= 700 x + 2,  
     * y <= 400  
)
```

```
tối đa hóa (4
```

```
    * x + 5 * y  
)
```

Bạn sử dụng Lập trình Ràng buộc mỗi ngày!

✿ hoặc ông bà của bạn!

Sudoku

	2		5		1		9	
số 8			2		3			6
	3			6			7	
		1				6		
5	4						1	9
		2				7		
	9			3			số 8	
2			số 8		4			7
	1		9		7		6	

Bạn sử dụng Lập trình Ràng buộc mỗi ngày!

Sudoku

	2		5		1		9	
số 8			2		3			6
	3			6			7	
		1				6		
5	4						1	9
		2				7		
	9			3				số 8
2				4				7
	1		9		7		6	

Biến

Miền

Điền vào các ô trống với các giá trị trong 1..9

Hạn chế

Trong khi xem xét rằng:

- mỗi hàng phải chứa các số khác nhau
- mỗi cột phải chứa các con số
- mỗi khôi (hình vuông 3 × 3) phải chứa những con số khác nhau

Bạn sử dụng Lập trình Ràng buộc mỗi ngày!

Sudoku

	2		5		1		9	
số 8			2		3			6
	3			6			7	
		1				6		
5	4							1 9
		2				7		
	9			3				số 8
2					4			7
	1		9		7		6	

Giải pháp: gán một giá trị cho mỗi biến sao cho không có ràng buộc nào bị vi phạm

* Câu đố Sudoku rất đơn giản (ví dụ của a) Vấn đề Thỏa mãn Ràng buộc

Bạn sử dụng Lập trình Ràng buộc mỗi ngày!

Sudoku

123 456 789	2	123 456 789	5	123 456 789	1	123 456 789	9	123 456 789
số 8	123 456 789	123 456 789	2	123 456 789	3	123 456 789	123 456 789	6
123 456 789	3	123 456 789	6	123 456 789	123 456 789	7	123 456 789	
123 456 789	123 456 789	1	123 456 789	123 456 789	6	123 456 789	123 456 789	
5 4	123 456 789	123 456 789	123 456 789	123 456 789	123 456 789	1 9		
123 456 789	2	123 456 789	123 456 789	7	123 456 789	123 456 789	123 456 789	
123 456 789	9	123 456 789	3	123 456 789	123 456 789	123 456 789	123 456 789	
2	123 456 789	123 456 789	số 8	123 456 789	4	123 456 789	123 456 789	7
123 456 789	1	123 456 789	9	123 456 789	7	123 456 789	6	123 456 789

Giải câu đố bằng CP có nghĩa là:

- lý luận với các ràng buộc để lược bỏ các giá trị trong các miền biến đổi •
gán các biến để xây dựng các giải pháp

Câu chuyện thành công của CP: CP đã hạ cánh trên sao chổi

Vào ngày 13 tháng 6 năm 2015, phòng thí nghiệm robot Philae thức dậy trên sao chổi 67P / Churyumov-Gerasimenko để tiếp tục một loạt thí nghiệm.



Một số kế hoạch do Philae thực hiện được mô hình hóa và giải quyết bằng công nghệ lập trình ràng buộc. Một số thuật toán chuyên dụng được phát triển bởi nhóm Nghiên cứu hoạt động & Ràng buộc của phòng thí nghiệm LAAS-CNRS.

Câu chuyện thành công của CP: Vấn đề về bộ ba Pythagore

Bài toán: Có thể tập $N = \{1, 2, \dots\}$ của số tự nhiên được chia thành 2 hai phần sao cho không phần nào chứa bộ ba (a, b, c) với $a^2 + b^2 = c^2$

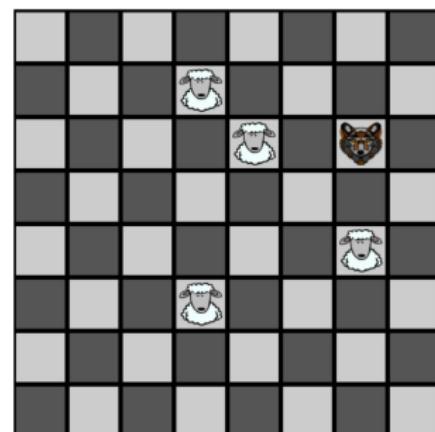
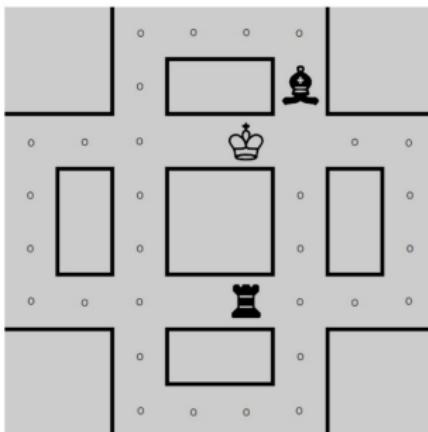
Một giải thưởng cho giải pháp đã được Ronald Graham đưa ra hơn hai thập kỷ trước.

Vấn đề này đã được giải quyết bằng phương pháp SAT kết hợp, sử dụng cả trình giải bài tập nhìn trước và CDCL.

Đây là bằng chứng Toán học có sự hỗ trợ của máy tính (**200TB**) lớn nhất từng được trình bày trong tài liệu.

Câu chuyện thành công của CP: Chơi trò chơi chung (GGP)

Chơi trò chơi chung (GGP) là thiết kế của các chương trình trí tuệ nhân tạo để có thể chơi thành công nhiều hơn một trò chơi.



Câu chuyện thành công của CP: Chơi trò chơi chung (GGP)



Woodstock, một người chơi trò chơi nói chung dựa trên ràng buộc được phát triển tại CRIL, dựa trên hai ý tưởng chính: • truyền tìm kiếm dựa trên ràng buộc (thành phần MAC) và • Lấy mẫu ngẫu nhiên dựa trên kẻ cướp (thành phần UCB).

WoodStock đã giành chiến thắng trong Giải chơi trò chơi tổng hợp quốc tế năm 2016 Cạnh tranh (IGGPC'16)

Câu chuyện thành công của CP: Âm nhạc trong CP

Kỹ thuật lập trình ràng buộc áp dụng cho sáng tác nhạc đương đại

Hai ứng dụng minh họa (với sự cho phép của Charlotte Truchet, LINA).

- ① Vấn đề về sự hòa hợp (Georges Bloch). Với một hợp âm cố định, chúng ta có các ràng buộc sau:
 - giảm thiểu EstradaDistance (hợp âm [i], FixedChord)
 - giảm thiểu GeorgesDistance (VF (hợp âm [i]), VF (hợp âm [i + 1]))

- ② Nhịp điệu không đồng bộ (Mauro Lanza). Mỗi mẫu được chơi lặp lại trên một giọng nói.
 - Biên số: các mẫu nhịp nhàng của thời lượng cố định d_1, \dots, d_n .
 - Ràng buộc: không có khởi động đồng thời giữa hai giọng nói, cho một thời lượng D.

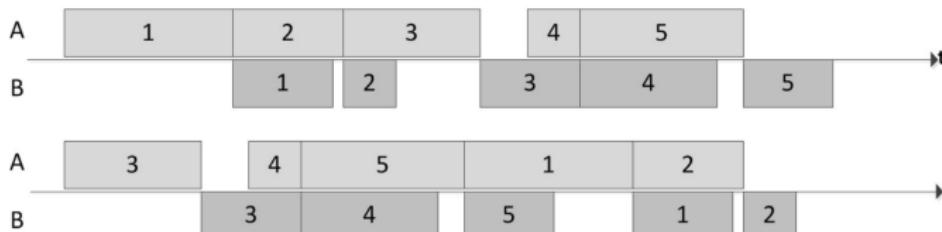
Chơi

Ứng dụng CP cỗ điển: Lập lịch cửa hàng lưu lượng

Cho n máy và m công việc, biết rằng:

- mỗi công việc yêu cầu thực hiện đúng n thao tác
- thao tác thứ i của công việc phải được thực hiện trên máy thứ i
- không máy nào có thể thực hiện đồng thời nhiều thao tác
- cho mỗi thao tác của từng công việc, thời gian thực hiện được chỉ định

Tìm lịch trình tốt nhất, tức là lịch trình có tổng thời gian thực hiện công việc ngắn nhất có thể.



Hình: Ví dụ về lập lịch tại cửa hàng (không chờ đợi) với 5 công việc trên 2 máy A và B. Một so sánh tổng thời gian được đưa ra cho hai chuỗi công việc khác nhau.

Ứng dụng CP cỗ điện: Xếp thứ tự trên ô tô

Đưa ra một dây chuyền lắp ráp để sản xuất ô tô với nhiều lựa chọn khác nhau, biết rằng:

- các trạm khác nhau cài đặt các tùy chọn khác nhau
- các trạm có thể xử lý tối đa một tỷ lệ nhất định số ô tô đi qua
- các ô tô yêu cầu một tùy chọn nhất định không được tập trung lại với nhau

Tìm một đơn đặt hàng sản xuất tôn trọng các hạn chế về năng lực của các trạm.



Ứng dụng CP cỗ điện: Bồi dưỡng y tá

Với các ca bệnh viện sẽ được thực hiện bởi các y tá, biết rằng:

- y tá có nhiều trình độ khác nhau
- bệnh viện phải tôn trọng một số mô hình làm việc / nghỉ ngơi
- mỗi dịch vụ bệnh viện có một số nhu cầu

Tìm sự phân công tối ưu của các y tá trong ca làm việc.

Employee shift rostering

Populate each work shift with a nurse.

	Maternity nurses			Emergency nurses			Basic nurses		
	A Ann	B Beth	C Cory	D Dan	E Elin	G Greg	H Hue	I Ilse	
Largest staff first									
Maternity nurses	1 2 C A B	1 1 C A B	2 1 A C B	1 2 C A B	1 1 C A B	2 1 C A B	1 1	1 1 1 1 1	1 1
Emergency nurses	2 1 D G E	2 1 D G E	1 1 D E	2 1 D G E	2 1 D G E	1 1 D G	1 1	1 1 1 1 1	1 1
Any nurses	1 1 H I	1 1 1 1 1 H I G H I	1 1 1 1 1 H I G H I	1 1 H I	1 1 1 1 1 H I E H I	1 1 H I	1 1	1 1 1 1 1	1 1
Drools Planner									
Maternity nurses	1 2 C A B	1 1 C A B	2 1 A C B	1 2 C A B	1 1 C A B	2 1 C A B	1 1	1 1 1 1 1	1 1
Emergency nurses	2 1 D G E	2 1 D G E	1 1 D E	2 1 D G E	2 1 D G E	1 1 D G	1 1	1 1 1 1 1	1 1
Any nurses	1 1 H I	1 1 1 1 1 H I G H I	1 1 1 1 1 H I G H I	1 1 H I	1 1 1 1 1 H I E H I	1 1 H I	1 1	1 1 1 1 1	1 1

Ứng dụng CP cỗ điển: Định tuyến xe

Theo đơn đặt hàng của một số nhà trang phục, biết rằng:

- có sẵn một đội xe
- kho lưu trữ sản phẩm

Tìm tuyến đường tốt nhất của mỗi phương tiện giao hàng.



Ứng dụng CP cỗ điện: Tải tàu container

Cho các container được xếp lên một số tàu, biết rằng:

- một số cần cẩu có sẵn
- có các yêu cầu cụ thể về các thùng chứa cần được tải

Tìm lịch tải nhanh nhất



Không có CP nào được sử dụng ở đây!



Ứng dụng CP cỗ điển

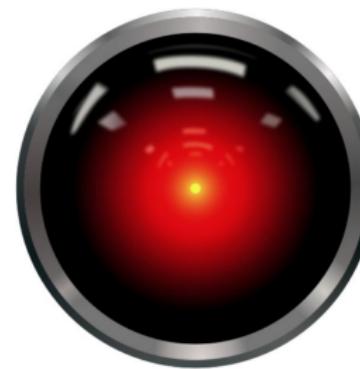
Nhiều miền ứng dụng khác:

- Cấu hình
- Tin học sinh học
- Lập kế hoạch
- Lên lịch sân bay
- Kiểm tra mô hình
- Khai thác dữ liệu
- . . .

Rẽ nào?

CP có nguồn gốc từ:

- Trí tuệ nhân tạo (AI)
- Nghiên cứu hoạt động (OR)



Mục tiêu nào?

Các vấn đề tuân theo các ràng buộc và / hoặc các chức năng mục tiêu.

Trên thực tế, tối ưu hóa thừa nhận nhiều khuôn khổ trong quy tắc:

- Lập trình toán học
- Lập trình tuyến tính
- Lập trình **tuyến tính số nguyên**
- Lập trình phi tuyến
- ...
- Meta-heuristics
 - Dựa trên quy đạo
 - Dựa trên dân số
- Lập trình ràng buộc

Phương châm nào?



"Nghệ thuật sống từ những ràng buộc và chết từ
tự do"

Lịch sử nào?

- Bắt đầu với tên gọi Lập trình lôgic ràng buộc (1987)
 - Giới thiệu các ràng buộc trong lập trình lôgic
 - Triển khai đầu tiên trong Prolog III, CLP, CHIP



Sau đó chuyển sang mô hình riêng (Lập trình ràng buộc) • Tạo ngôn ngữ / thư viện • Tạo bộ giải tự trị • Tách khỏi thế giới logic

Lập trình ràng buộc

Lập trình ràng buộc (CP) là một khuôn khổ chung để xuất các giải pháp thuật toán đơn giản, tổng quát và hiệu quả cho các vấn đề có ràng buộc.

CP hấp dẫn vì có sự tách biệt rõ ràng giữa:

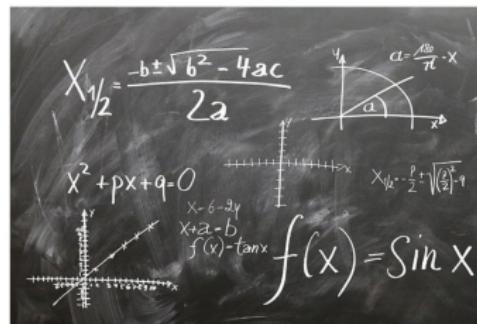
- một mặt, tính hình thức của nó giúp dễ dàng biểu diễn nhiều vấn đề •
- mặt khác, một nhóm lớn các thuật toán và kinh nghiệm học để tìm

các giải pháp

Mô hình hóa

Sau đó, có hai giai đoạn chính với CP:

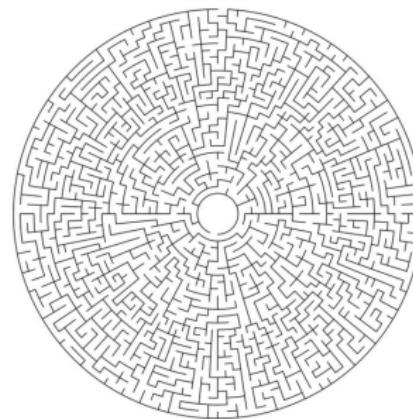
- Trong giai đoạn đầu tiên, được gọi là mô hình hóa, vấn đề phải được chính thức hóa được biểu diễn bằng các biến, các ràng buộc và có thể là các hàm mục tiêu.



Lý tưởng nhất, giai đoạn này hoàn toàn là khai báo, nhưng trong thực tế có thể yêu cầu một quy trình lập trình hạn chế (ví dụ: với ngôn ngữ logic hoặc đối tượng).

Giải quyết

- 2 Trong giai đoạn thứ hai, được gọi là giải quyết, vấn đề do người dùng mô hình hóa phải được giải quyết bằng một công cụ phần mềm để tự động có được một giải pháp, tất cả các giải pháp, một giải pháp tối ưu, . . .



Sự hài lòng về ràng buộc

Vấn đề thỏa mãn ràng buộc (CSP) nằm ở cốt lõi của lập trình ràng buộc. Một ví dụ của vấn đề này được biểu diễn bằng một mạng ràng buộc (CN).

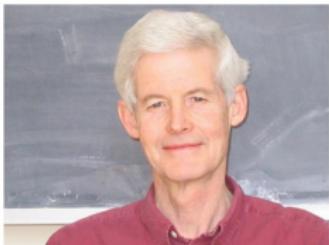


Lưu ý rằng SAT có liên quan chặt chẽ đến CSP:

- các biến là Boolean
- ràng buộc là các mệnh đề (loại bỏ các biến và phủ định của chúng)

Không thẻ bẻ gãy

Không có thuật toán đa thức nào được biết đến cho cả hai bài toán quyết định CSP và SAT.



Stephen Cook đã chỉ ra rằng SAT là một NP-hoàn thành vấn đề

Nhận xét.

Trong thực tế, mọi người cố gắng tìm ra các thuật toán hiệu quả cho một loạt các vấn đề bằng cách khai thác các cấu trúc của chúng. Đây là một thách thức thúc đẩy.

Thư mục

- Cơ sở của sự thỏa mãn hạn chế (Tsang, 1993)
- Nguyên tắc lập trình ràng buộc (Apt, 2003)
- Xử lý ràng buộc (Dechter, 2003) • Tìm kiếm cục bộ dựa trên ràng buộc (van Hentenryck & Michel, 2005) • Số tay lập trình ràng buộc (Rossi và cộng sự, 2006) • Truyền dẫn ràng buộc (Bessiere, 2006) • Mạng ràng buộc (Lecoutre, 2009) • Ràng buộc toàn cầu: một cuộc khảo sát (Regin, 2011) • Các phương pháp tích hợp để tối ưu hóa (Hooker, 2012)



Biến đổi

Định nghĩa

Một biến (có tên) x là một thực thể chưa biết phải được cho một giá trị từ một tập được gọi là miền (hiện tại) của x và được ký hiệu là $\text{dom}(x)$.

Nhận xét.

Miền của một biến x thường phát triển theo thời gian nhưng nó luôn được bao gồm trong một tập được gọi là miền ban đầu của x và được ký hiệu là $\text{dom}^0(x)$.

Ví dụ.

Các biến x và y có các miền hiện tại:

$$\text{dom}(x) = \{2, 3, 4\} \text{ và } \text{dom}(y) = \{1, 3, 4\}.$$

Các miền ban đầu của chúng là:

$$\text{dom}^0(x) = \text{dom}^0(y) = \{1, 2, 3, 4, 5\}.$$

Sản phẩm Descartes

Định nghĩa (Tuple)

Bộ giá trị τ là một chuỗi các giá trị, nằm giữa dấu ngoặc đơn và dấu phẩy được sử dụng làm dấu phân cách. Một bộ chứa giá trị r được gọi là r -tuple. Giá trị thứ i của r -tuple, với $1 \leq i \leq r$, được ký hiệu là $\tau [i]$.

Định nghĩa

Cho S_1, S_2, \dots, S_r là một dãy gồm r bộ. Sản phẩm Descartes

$$\prod_{i=1}^r S_i = \{s_1 \times s_2 \times \dots \times s_r$$

là bộ

$$\{(a_1, a_2, \dots, a_r) \mid a_1 \in S_1, a_2 \in S_2, \dots, a_r \in S_r\}.$$

Mỗi phần tử của $\prod_i S_i = \prod_{i=1}^r S_i$ tương ứng với một r -tuple.

Ví dụ

Ví dụ. Gọi x, y và z là ba biến sao cho $\text{dom}(x) = \text{dom}(y) = \{a, b\}$ và $\text{dom}(z) = \{a, c\}$. Sau đó chúng tôi có:

$$\begin{aligned} & (a, a, a), \\ & (a, a, c), \\ & (a, b, a), \\ & (a, b, c), \\ \text{dom}(x) \times \text{dom}(y) \times \text{dom}(z) = & ((b, a, a), (b, a, c), (b, b, a), (b, b, c)) . \end{aligned}$$

Quan hệ

Một quan hệ chỉ đơn giản là một tập con của tích Descartes.

Dịnh nghĩa (Quan hệ)

Một quan hệ R xác định trên một dãy gồm r tập S_1, S_2, \dots, S_r là tập con của tích Descartes $\prod_{i=1}^r S_i = \prod_{i=1}^r S_i$, tức là, ta có: $R = \prod_{i=1}^r S_i$.

Ví dụ

Một quan hệ R được xác định trên dom (x), dom (y), dom (z) là ví dụ:

$$R = \{ (a, a, a), (a, a, c), (a, a, c), (a, b, a), (b, a, a), (a, b, c), (b, a, c), (b, b, b), (b, b, c) \}$$

Hạn chế

Định nghĩa

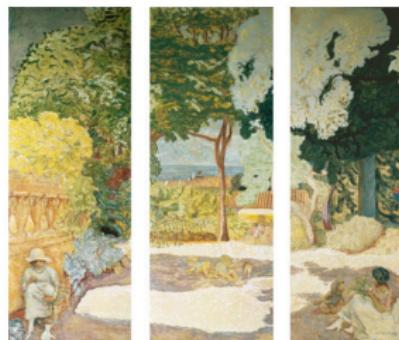
Một ràng buộc (với tên) c được xác định trên một tập hợp các biến (được sắp xếp hoàn toàn), được gọi là phạm vi của c và được ký hiệu là $\text{scp}(c)$, bởi một quan hệ $\text{rel}(c)$ mô tả tập các bộ giá trị được c cho phép đối với các biến trong phạm vi của nó.

Độ hiểm của một ràng buộc c là số biến liên quan đến c , tức là, $|\text{scp}(c)|$. Chúng ta nói rằng c là:

- binary iff arity của nó là 1,
- ternary iff arity của nó là 2,
- . . .
- non-binary iff độ hiểm của nó lớn hơn 2.

Đại diện cho các ràng buộc

Về mặt hình thức, một ràng buộc được định nghĩa là một quan hệ toán học. Trong thực tế, có ba cách khác nhau để biểu diễn một ràng buộc: • trong số nguyên, bằng cách sử dụng công thức Boolean (vị từ), • trong phần mở rộng, bằng cách liệt kê các bộ giá trị, • ngầm định bằng cách tham chiếu đến cái gọi là ràng buộc toàn cục.



Các ràng buộc của Intensio (nal)

Định nghĩa (Ràng buộc Tương tác)

Một ràng buộc c là nguyên, hoặc được định nghĩa trong nguyên, nghĩa là nó được mô tả bằng công thức Boolean (vị từ) đại diện cho một hàm được xác định từ $\Pi x \text{ } \text{sc}p(c) \text{ dom}0(x)$ đến B.

Ví dụ.

Một ràng buộc cường độ nhị phân:

$$v \leq w + 2$$

Ràng buộc số nguyên bậc ba: x^6

$$= y \quad x^6 = z \quad y^6 = z$$

Nhận xét.

Các ràng buộc về số nguyên thường được sử dụng để biểu diễn các ràng buộc số học.

Các ràng buộc của phần mở rộng (al)

Định nghĩa (Ràng buộc Mở rộng)

Một ràng buộc c là mở rộng (hoặc được định nghĩa trong phần mở rộng) vì nó được mô tả rõ ràng,

- tích cực bằng cách liệt kê các bộ giá trị được c cho phép, • hoặc phủ định bằng cách liệt kê các bộ giá trị mà c không cho phép.

Ví dụ.

Nếu $\text{dom}(x) \times \text{dom}(y) \times \text{dom}(z) = \{0, 1, 2\}$ thì ràng buộc bậc ba từ trang chiếu trước có thể được xác định tích cực là: $hx, y, zi^3 = \{(0, 1, 2), (0, 2, 1), (1, 0, 2), (1, 2, 0), (2, 1, 0), (2, 0, 1)\}$

Nhận xét.

Các Ràng buộc Mở rộng còn được gọi là Ràng buộc Bảng.

Ràng buộc toàn cầu

Định nghĩa (Ràng buộc Toàn cầu)

Ràng buộc toàn cục là một mẫu ràng buộc nắm bắt ngữ nghĩa quan hệ chính xác và có thể được áp dụng trên một số biến tùy ý.

Ví dụ, ngữ nghĩa của allDierence là tất cả các biến phải nhận các giá trị khác nhau.

Ví dụ.

Ràng buộc bậc ba trước đây của chúng ta có thể được xác định bằng:

allDierence (x, y, z)

Nhận xét.

Nhiều ràng buộc toàn cầu sẽ được giới thiệu sau trong khóa học này.

Phiên bản CSP

Sự định nghĩa

Một ví dụ P của Vấn đề thỏa mãn ràng buộc (CSP), còn được gọi là
Mạng Ràng buộc (CN), bao gồm:

- một tập hợp hữu hạn các biến, được ký hiệu là vars
- một tập hữu hạn các ràng buộc, được ký hiệu là $\text{ctrs} (P)$, sao cho
 $c \in \text{ctrs} (P)$, $\text{scp}(c) \subseteq \text{vars} (P)$.



Nhận xét.

Một cặp (x, a) với $x \in \text{vars} (P)$ và $a \in \text{dom} (x)$ đôi khi được gọi là một nghĩa đen
của P (hoặc thậm chí, bằng cách lạm dụng ngôn ngữ, một giá trị của P).

Cuồng ché.

- ➊ Đề xuất định nghĩa cho khái niệm giải pháp (của một CSP cụ thể ví dụ).
- ➋ Tìm nghiệm nhỏ nhất và nghiệm lớn nhất (theo thứ tự từ vựng) của cá thể CSP P sau:

- $\text{vars}(P) = \{w, x, y, z\}$ với •
 $\text{dom}(w) = \{1, 2, 3\} \cdot \text{dom}(x) = \{1, 2, 3\} \cdot \text{dom}(y) = \{1, 2, 3, 4\} \cdot \text{dom}(z) = \{1, 2, 3, 4\} \cdot \text{ctrs}(P)$
 $= \{$
 $w = x,$
 $x \leq y + 1,$
 $y > z, \text{ hx},$
 $\text{zi } \{(1, 2), (2, 1), (2, 4), (3, 3)\}\}$

- ➌ trận hòa:

- đồ thị ràng buộc của P

CSP so với SAT

Vấn đề CSP liên quan đến việc xác định xem một mạng ràng buộc nhất định có giải pháp hay không. CSP “tổng quát hóa” SAT.



Ví dụ.

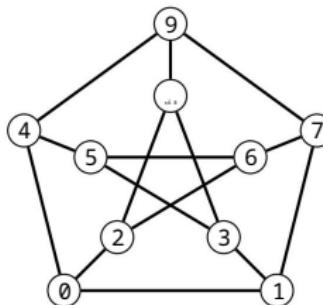
Đối với công thức CNF sau: (x

$$y \quad \neg z) \quad (\neg w \quad \neg x) \quad (w \quad \neg y \quad z)$$

chúng ta thu được một mạng ràng buộc P sao cho:

- $\text{vars}(P) = \{w, x, y, z\}$ st $v \in \text{vars}(P), \text{dom}(v) = B$ •
- $\text{ctrs}(P) = \{x \in y \in \neg z, \neg w \in \neg x, w \in \neg y \in z\}$

Một minh họa khác: Vấn đề tô màu đồ thị



Chúng ta phải gán:

- tại mỗi nút của đồ thị

Biến

- một màu được lấy trong $\{R, G, B\}$

Miền

- sao cho không có nút nào liền kề có cùng màu

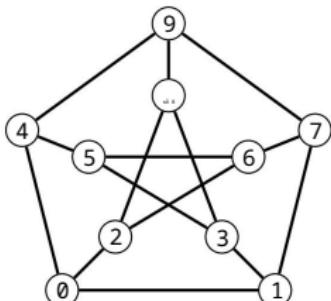
Hạn chế

Vấn đề tô màu đồ thị

Dữ liệu (đầu vào)

- biểu đồ $G = (V, E)$

tập hợp các màu {R, G, B}



Phiên bản CSP P:

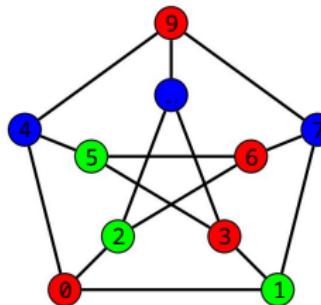
- $\text{vars } (P) = \{x_0, x_1, \dots, x_9\}$
với $i \in 0..9$, $\text{dom } (x_i) = \{\text{R}, \text{G}, \text{B}\}$
- $\text{ctrs } (P) = \{x_i = x_j : 0 \leq i < j \leq 9 \quad (i, j) \in E\}$

Nhận xét.

Mỗi biến đại diện cho một nút và mỗi ràng buộc là nhị phân.

Vấn đề tô màu đồ thị

Một giải pháp



Nhận xét.

Các biến có miền tương ứng trùng ở đây. Tuy nhiên, hầu hết các bộ giải đều xử lý các biến số nguyên.

Exercice: viết mô hình PyCSP3.

Còn về Tối ưu hóa?

Định nghĩa

Một hàm mục tiêu o đôi với CN P kết hợp một giá trị số, được gọi là giá trị mục tiêu, với mỗi lần khởi tạo hoàn chỉnh các biến của P.

Chúng tôi có o:

$$\Pi x \text{ vars } (P) \text{ dom } (x) \quad R$$

Một hàm mục tiêu cho phép chúng ta:

- để phân biệt giữa các giải pháp, • để xác

định những giải pháp tốt nhất, dưới phô của một trong hai phương pháp giảm thiểu hoặc tối đa hóa.

Nhận xét.

Trong khóa học này, chúng tôi sẽ chủ yếu tập trung vào tối ưu hóa đơn mục tiêu.

Phiên bản COP

Sự định nghĩa

Ví dụ P của Bài toán Tối ưu hóa Ràng buộc (COP) bao gồm:

- tập hợp hữu hạn các biến, được ký hiệu là vars (P),
- tập hữu hạn của các ràng buộc, được ký hiệu là ctrs (P), sao cho

c ctrs (P), scp (c) vars (P),

- một hàm mục tiêu được ký hiệu bằng obj (P), được tối thiểu hóa hoặc tối đa.

Nhận xét.

Một cá thể COP có thể được coi là một cá thể CSP cơ bản cùng với một hàm mục tiêu.

Giải pháp cho Phiên bản CSP / COP

Định nghĩa

Giải pháp của một cá thể CSP P là việc gán một giá trị cho mỗi biến của P sao cho tất cả các ràng buộc của P đều được thỏa mãn. Tập nghiệm của P được kí hiệu là $sols(P)$.

Sự định nghĩa

- Giải pháp của phiên bản COP là giải pháp của CSP cơ bản.
- Để tối thiểu hóa (tương ứng, tối đa hóa), giải pháp tối ưu là giải pháp có giá trị mục tiêu nhỏ hơn hoặc bằng (tương ứng, lớn hơn hoặc bằng) giá trị mục tiêu của bất kỳ giải pháp nào khác.

Tìm một giải pháp tối ưu có thể rất "phức tạp"

Lập mô hình có Ràng buộc (Toàn cầu)

Chúng tôi minh họa các ràng buộc rất phổ biến:

- tất cả
- tổng
- phần mở rộng (hoặc bảng)

Ràng buộc tất cả

Cú pháp.

`allDifference (X) với X phạm vi của ràng buộc.`

Ngữ nghĩa học.

Một khởi tạo X thỏa mãn ràng buộc iff tất cả các giá trị được gán là khác nhau.

Quan trọng.

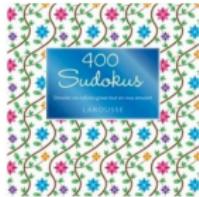
Ràng buộc này rất hữu ích cho nhiều vấn đề.

Ví dụ.

Cho ràng buộc tất cả Khác nhau (w, x, y, z):

- $\{w = 1, x = 2, y = 2, z = 3\}$ không thỏa mãn ràng buộc
- $\{w = 1, x = 2, y = 4, z = 3\}$ thỏa mãn ràng buộc

Câu đố Sudoku



	2		5		1		9	
số 8			2		3		6	
	3			6		7		
		1				6		
5	4						1	9
		2				7		
	9			3			số 8	
2			số 8		4			7
	1		9		7		6	

Lập mô hình Sudoku

Ví dụ.

Chúng ta có thể xác định P sao cho:

```

• vars (P) = {

    x11, x12 , . . . , x19,
    x21, x22 , . . . , x29,
    . . .

    x91, x92 , . . . , x99

} với dom (xij) = {1 ,. . . , 9}, i, j 1..9

• ctrs (P) =

    {allDierence (x11, x12, . . . , X19 ),
     allDierence (x21, x22 , . . . , X29 ),
     . . .

}

```

Nhận xét.

Đối với mỗi đầu mối, hãy thêm một ràng buộc một ngôi

Giải pháp cho câu đố Sudoku

	2		5	1	9			
			2	3			6	
	3		6		7			
		1			6			
5	4					1	9	
		2			7			
	9			3				
2				4				7
	1		9	7	6			

Câu đố

4	2	6	5	7	1	3	9	số 8
	5	7	2	9	3	1	4	6
1	3	9	4	6	8	2	7	5
9	7	1	3	8	5	6	2	4
5	4	3	7	2	6	8		1
6	8	2	1	4	9		7	5
7	9	4	6		3	2	5	số 8
2	6	5			1	4	9	3
3	1		9	5	7	4	6	2

Giải pháp

Tổng ràng buộc

Cú pháp.

Tổng ràng buộc được viết là:

$\Pr i = 1 \text{ ci } xi <\text{đầu}> L$

ở đâu:

- $ci \quad z, \quad i$
- $1..x <\text{đầu}> \quad \{<, \leq, \geq, >, =, \neq, =,$
 $, / \quad \} \cdot L$ một số nguyên (z), một khoảng hoặc một biến

Ngữ nghĩa học.

Ngay tức khắc

Ví dụ.

Ràng buộc $cwxyz : w + 2x + 4y + 5z \geq 42$

Hình vuông ma thuật



Melencolia (1514) của Họa sĩ người Đức Albrecht Dürer

2	9	4		
7	5	3		
6	1	8		

1	13	8	12	4		
16	5	9				
14	2	11	7			
15	3	10	6			

2	15	22	3	23	16		
13	20	12	4	19	17		
1	21	7					
10	11	14	24	6	18		
9	8	5	25				

Mô hình hóa hình vuông ma thuật

Ví dụ.

Đối với bình phương ma thuật bậc 3, chúng ta có thể xác định P sao cho:

- vars (P) =


```
{ x11, x12,
        x13 , x21,
        x22 , x23,
        x31, x32, x33 }
```

 với dom (xij) = {1, ..., 9}, i, j = 1..9
- ctrs (P) = {


```
P3 j = 1 x1j = 15
P3 j = 1 x2j = 15
...
}
```

Nhận xét.

Có một tổng ràng buộc cho mỗi đường chéo chính.

Phần mở rộng ràng buộc (hoặc bảng)

Cú pháp.

Ràng buộc này có thể được viết $X \in T$ với:

- $X = hx_1, x_2, \dots, x_r$, phạm vi của ràng buộc $\in T$,
- một bộ r -tuples Đối với một ràng buộc bảng phủ định, hãy
- thay thế bởi /.

Nhận xét.

Ràng buộc hữu ích / bắt buộc khi không biết ngữ nghĩa.

Tạo lưới ô chữ

MOTS CROISÉS PAR JEAN-FRANÇOIS DEMAY

GRILLE N° 48

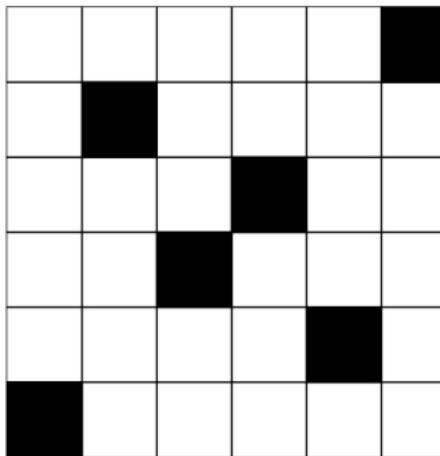
horizontal:
 1. H. H. V. V. V. VII. X.
 2. [REDACTED]
 3. [REDACTED]
 4. [REDACTED]
 5. [REDACTED]
 6. [REDACTED]
 7. [REDACTED]
 8. [REDACTED]

vertical:
 1. [REDACTED]
 2. [REDACTED]
 3. [REDACTED]
 4. [REDACTED]
 5. [REDACTED]
 6. [REDACTED]
 7. [REDACTED]
 8. [REDACTED]

diagonale:
 1. [REDACTED]
 2. [REDACTED]
 3. [REDACTED]
 4. [REDACTED]
 5. [REDACTED]
 6. [REDACTED]
 7. [REDACTED]
 8. [REDACTED]

Solutions de la grille n° 47:

1. Hiver long	1. Beau-sœur	1. Bébé-sœur	1. Tasse à thé				
2. [REDACTED]	2. [REDACTED]	2. [REDACTED]	2. [REDACTED]	2. [REDACTED]	2. [REDACTED]	2. [REDACTED]	2. [REDACTED]
3. [REDACTED]	3. [REDACTED]	3. [REDACTED]	3. [REDACTED]	3. [REDACTED]	3. [REDACTED]	3. [REDACTED]	3. [REDACTED]
4. [REDACTED]	4. [REDACTED]	4. [REDACTED]	4. [REDACTED]	4. [REDACTED]	4. [REDACTED]	4. [REDACTED]	4. [REDACTED]
5. [REDACTED]	5. [REDACTED]	5. [REDACTED]	5. [REDACTED]	5. [REDACTED]	5. [REDACTED]	5. [REDACTED]	5. [REDACTED]
6. [REDACTED]	6. [REDACTED]	6. [REDACTED]	6. [REDACTED]	6. [REDACTED]	6. [REDACTED]	6. [REDACTED]	6. [REDACTED]
7. [REDACTED]	7. [REDACTED]	7. [REDACTED]	7. [REDACTED]	7. [REDACTED]	7. [REDACTED]	7. [REDACTED]	7. [REDACTED]
8. [REDACTED]	8. [REDACTED]	8. [REDACTED]	8. [REDACTED]	8. [REDACTED]	8. [REDACTED]	8. [REDACTED]	8. [REDACTED]



Tạo mô hình ô chữ

Đối với một lưới cụ thể, chúng ta có thể xác định P sao cho:

- vars (P) chứa một biến x với dom (x) = {a, . . . , z} đối với mỗi ô trắng,
- ctrs (P) chứa phần mở rộng ràng buộc cho mỗi trình tự (tối đa) của các tế bào trắng.

Ví dụ.

Đối với hàng:

	x1	x2	x3	x4	x5	
--	----	----	----	----	----	--

chúng ta có thể tạo ràng buộc:

hx1, x2, x3, x4, x5i {abbes, Abdos, . . . , zozos}

Nhận xét.

Một từ chẳng hạn như "abbes" là một phím tắt cho bộ (a, b, b, e, s).

Giải pháp cho Trường hợp ô chữ



Kết quả tốt thu được với bộ giải AbsCon (Ace): •

suy nghiệm chung • lọc chung (STR)

Với Sébastien



- khởi động lại

Mots croisés - Rectangles de côté 7 - Mozilla Firefox

[File](#) [Edit](#) [View](#) [History](#) [Bookmarks](#) [Tools](#) [Help](#)

[Mots croisés - Rectangles de côté 7](#)

RECTANGLES 7

1 record à égaliser
6 records à établir

Le Grand Défi des mots croisés

Les Rectangles classiques de côté 7

[Index](#)
[Archives](#)
[Règle du jeu](#)
[Carrés classiques](#)
[Rectangles classiques](#)

[de côté 3](#)
[de côté 4](#)
[de côté 5](#)
[de côté 6](#)
[de côté 7](#)
[de côté 8](#)
[de côté 9](#)
[de côté 10](#)
[de côté 11](#)
[de côté 12](#)
[de côté 13](#)

[Records du monde](#)

R	E	C	L	A	M	E	R	64
E	C	O	U	L	A	N	E	64
C	O	N	C	E	R	T	S	64
L	U	C	A	R	N	E	S	64
A	L	E	R	T	E	R	A	64
M	A	R	N	E	R	A	I	64
E	S	T	E	R	A	S	E	64
49	49	49	49	49	49	49	49	64

7 x 8 - Arthur BALL - 24 septembre 2008

D	R	A	W	B	A	C	K	64
R	E	M	A	R	C	H	A	64
E	P	I	T	O	I	E	S	64
P	O	N	T	I	E	R	S	64
O	N	T	E	R	A	M	R	64
N	A	M	E	R	A	I	T	64
U	R	A	N	A	I	S	E	64
49	49	49	49	49	49	49	49	64

7 x 8 - Christophe LECOUTRE et Sébastien TABARY - 18 avril 2008

[Done](#)

7 x 9 - Michel SALMONS - 30 sep 2006

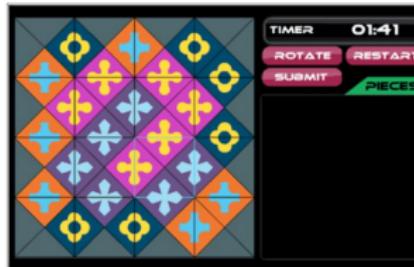
D	E	R	A	D	A	M	E	S	99
E	P	A	L	E	M	E	N	T	99
M	I	D	I	N	E	T	T	E	99
E	N	A	M	O	U	R	E	R	99
L	A	M	E	N	T	E	R	A	99
A	G	E	N	C	E	R	A	I	99
T	E	S	T	E	R	A	I	T	99
49	49	49	49	49	49	49	49	49	99

7 x 10 - Michel SALMONS - 28 oct 2006

R	A	F	F	U	T	A	M	E	S	100
E	R	R	O	N	E	M	E	N	T	100
C	L	A	R	I	N	E	T	E	100	
R	E	N	A	M	O	U	R	E	100	
A	U	G	M	E	N	T	E	R	100	
N	M	E	E	N	N	E	R	A	100	
S	I	E	S	T	E	R	A	I	100	
49	0	49	49	49	49	49	49	49	100	

Vấn đề cuối cùng trước khi ra đi: Eternity II

Câu đố đồ chơi: 16 miếng



Phiên bản thương mại: 256 mảnh - 2 triệu đô la để giành chiến thắng



Eternity II

Với Christian

Thử với bộ giải AbsCon (Ace): • hai kiểu máy

khác nhau

- heuristic chung chung
- khởi động lại



Kết quả tốt nhất: -29 ($480 - 29 = 451$ liên kết đúng) thu được trong vài phút (nhưng sau
2/3 tuần để phản ánh / mô hình hóa / phát triển)

Epilog to Eternity II

Eternity II - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Eternity II

ETERNITY II™

The final date for the correct solution of the Eternity II puzzle passes without a winner, and the \$2m Prize for a correct solution to the Eternity II puzzle goes unclaimed.

The original £1m Eternity puzzle sparked a gaming frenzy in back in 1999 and this sequel raised the bar on puzzle prize funds even further – the prize \$2 million!!

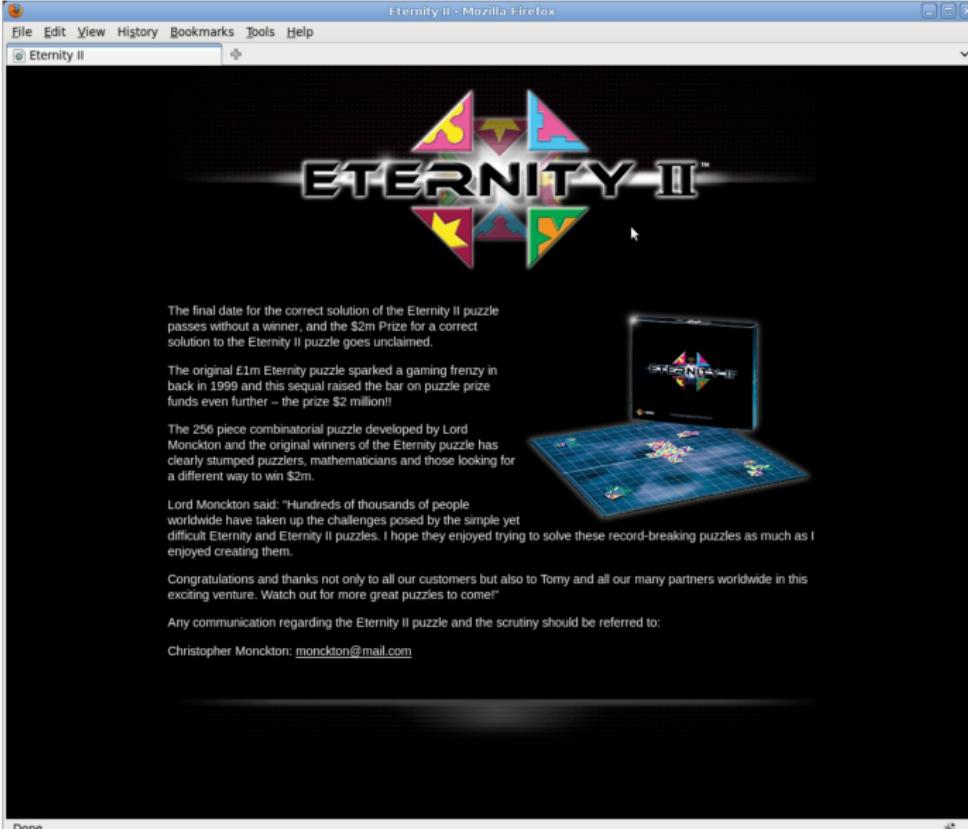
The 256 piece combinatorial puzzle developed by Lord Monckton and the original winners of the Eternity puzzle has clearly stumped puzzlers, mathematicians and those looking for a different way to win \$2m.

Lord Monckton said: "Hundreds of thousands of people worldwide have taken up the challenges posed by the simple yet difficult Eternity and Eternity II puzzles. I hope they enjoyed trying to solve these record-breaking puzzles as much as I enjoyed creating them."

Congratulations and thanks not only to all our customers but also to Tomy and all our many partners worldwide in this exciting venture. Watch out for more great puzzles to come!"

Any communication regarding the Eternity II puzzle and the scrutiny should be referred to:

Christopher Monckton: monckton@mail.com



Done

Apt, KR 2003.

Nguyên tắc lập trình ràng buộc.

Nhà xuất bản Đại học Cambridge.

Bessiere, C. 2006.

Tuyên truyền ràng buộc.

Chap. 3, trang 29-83 của: Sổ tay Lập trình Ràng buộc.

Elsevier.

Dechter, R. 2003.

Xử lý ràng buộc.

Morgan Kaufmann.

Hooker, JN 2012.

Các phương pháp tích hợp để tối ưu hóa.

Springer.

Lecoutre, C. 2009.

Các mạng ràng buộc: kỹ thuật và thuật toán.

ISTE / Wiley.

Régin, J.-C. 2011.

Ràng buộc toàn cầu: một cuộc khảo sát.

Chap. 2, trang 63-134 của: Tối ưu hóa Kết hợp.

Springer.

Rossi, F., van Beek, P., & Walsh, T. (chủ biên). Năm 2006.

Sổ tay Lập trình Ràng buộc.

Elsevier.

Tsang, E. 1993.

Cơ sở của sự thỏa mãn hạn chế.

Báo chí Học thuật.

van Hentenryck, P., & Michel, L. 2005.

Tìm kiếm cục bộ dựa trên ràng buộc.

Báo chí MIT.