

VIỆN KHOA HỌC VÀ CÔNG NGHỆ VIỆT NAM

VIỆN TOÁN HỌC

||||| -

LÊ MẠNH HÀ

**CẤU TRÚC KHÔNG GIAN TRẠNG THÁI VÀ TÍNH
ĐẠT ĐƯỢC CỦA MỘT SỐ HỆ ĐỘNG LỰC RỜI RẠC**

LUẬN ÁN TIẾN SĨ TOÁN HỌC

HÀ NỘI - 2010

VIỆN KHOA HỌC VÀ CÔNG NGHỆ VIỆT NAM

VIỆN TOÁN HỌC

| | | | | | | | | | | | | |

LÊ MẠNH HÀ

**CẤU TRÚC KHÔNG GIAN TRẠNG THÁI VÀ TÍNH
ĐẠT ĐƯỢC CỦA MỘT SỐ HỆ ĐỘNG LỰC RỜI RẠC**

**Chuyên ngành: Đảm bảo toán học cho
máy tính và hệ thống tính toán
Mã số: 62 46 35 01**

LUẬN ÁN TIẾN SĨ TOÁN HỌC

TẬP THỂ HƯỚNG DẪN KHOA HỌC:
1. TS. Phan Thị Hà Dương
2. PGS. TS. Phan Trung Huy

HÀ NỘI - 2010

Lời cam đoan

Tôi xin cam đoan đây là công trình nghiên cứu của tôi. Các kết quả của luận án là mới và chưa từng được ai công bố trong bất kì công trình nào khác.

Tác giả

Lê Mạnh Hà

Lời cảm ơn

Tôi không thể diễn tả hết bằng lời lòng biết ơn sâu sắc của tôi đối với cô giáo TS. Phan Thị Hà Dương và cũng không lời nào có thể kể hết công lao của Cô đối với tôi. Hơn cả một người hướng dẫn khoa học, Cô rèn rũa tôi từng ngày trong suốt bốn năm tôi làm nghiên cứu sinh. Từ những ngày đầu tiên, kể từ khi tôi chưa được học nhiều về tổ hợp, về toán rời rạc, Cô đã dạy bảo, chỉ dẫn tôi một cách tỉ mỉ, nghiêm khắc và kiên trì. Và hơn cả, tôi luôn cảm nhận được tình thương quý, tin yêu của Cô dành cho tôi, tôi đã không ngừng phấn đấu và trưởng thành dưới sự dạy bảo và niềm tin yêu ấy. Đó là những tình cảm vô cùng quý giá đối với tôi, là nguồn động viên vô cùng to lớn và sẽ mãi thấp sáng niềm say mê nghiên cứu khoa học của tôi. Tôi sẽ còn phấn đấu nhiều hơn nữa để xứng đáng với công lao của Cô đã bỏ ra, xứng đáng với niềm tin của Cô đã dành cho tôi.

Tôi xin bày tỏ lòng biết ơn sâu sắc đến PGS. TS. Phan Trung Huy, thầy đã động viên giúp đỡ tôi từ những ngày đầu tiên khi tôi vừa mới bắt đầu thi nghiên cứu sinh. Trong suốt quá trình làm nghiên cứu sinh, tôi luôn nhận được những góp ý, động viên của Thầy về các kết quả mà tôi đạt được ở các buổi xêmina của Phòng. Thầy đã đọc và góp những ý kiến xác đáng đối với bản dự thảo của luận án này. Tôi xin tỏ lòng biết ơn sâu sắc đến Thầy.

Trong quá trình học tập và nghiên cứu tại Viện Toán, tôi luôn nhận được sự quan tâm sâu sắc của PGS. TS Phạm Trà Ân, Thầy Phạm Trà Ân không những chỉ bảo tôi về mặt kiến thức mà còn luôn quan tâm đến những khó khăn trong cuộc sống hàng ngày. Thầy đã đưa ra ý tưởng để giúp tôi tìm ra mối liên hệ giữa các hệ động lực rời rạc và các hệ tin học. Nhờ đó tôi đã có được một số kết quả của luận án ở chương 3. Tuy Thầy hiện nay đã nghỉ hưu nhưng Thầy đã dành thời gian để đọc và góp những ý kiến xác đáng đối với bản dự thảo của luận án này. Nhân dịp này

tôi xin chân thành cảm ơn Thầy.

Tôi xin cảm ơn các thầy và các anh chị em trong xêmina của phòng Cơ sở Toán học của tin học của Viện Toán học về những trao đổi, hỗ trợ và chia sẻ trong khoa học cũng như trong cuộc sống. Đặc biệt, tôi xin chân thành cảm ơn GS. TS. Ngô Đắc Tân và TS. Lê Công Thành đã góp những ý kiến xác đáng đối với các kết quả của luận án thông qua các buổi xêmina của phòng.

Tôi xin trân trọng cảm ơn Viện Toán học, các phòng chức năng, Trung tâm Đào tạo sau đại học của Viện Toán học đã tạo điều kiện tốt nhất giúp tôi học tập, nghiên cứu và tham gia một cách hiệu quả các buổi sinh hoạt khoa học của Viện để tôi có thể hoàn thành luận án này.

Tôi xin cảm ơn các bạn trong xêmina "Tính toán tổ hợp và các hệ động lực rời rạc" về những thảo luận và góp ý trong các buổi xêmina. Đặc biệt, tôi xin cảm ơn bạn Phạm Văn Trung và bạn Trần Thị Thu Hương đã cùng tôi học tập và trao đổi kiến thức dưới sự hướng dẫn của Cô giáo Phan Thị Hà Dương trong suốt hai năm qua. Bạn Trần Thị Thu Hương đã đọc kỹ bản thảo của luận án và chỉ ra các lỗi trong luận án. Nhân dịp này tôi trân trọng cảm ơn những ý kiến trao đổi của các bạn cũng như những tình cảm của các bạn đã dành cho tôi trong những lúc khó khăn trong cuộc sống.

Tôi xin cảm ơn khoa Toán trường Đại học Sư phạm - Đại học Huế đã trang bị cho tôi những kiến thức cơ bản về toán học. Tôi xin cảm ơn Ban giám hiệu trường Đại học Sư phạm - Đại học Huế đã cho tôi cơ hội được đi học tập và nghiên cứu. Tôi xin cảm ơn Ban chủ nhiệm khoa Giáo dục Tiểu học đã tạo điều kiện thu xếp công việc thuận lợi cho tôi trong suốt thời gian tôi làm nghiên cứu sinh.

Cuối cùng tôi muốn bày tỏ lòng biết ơn sâu sắc tới bố, mẹ, và em gái, những người đã cảm thông và chia sẻ mọi khó khăn cùng tôi suốt những năm tháng qua để tôi có thể hoàn thành luận án này.

Mục lục

Mở đầu	1
Chương 1. Kiến thức chuẩn bị	5
1.1 Tập thứ tự - Dàn	5
1.1.1 Tập thứ tự	5
1.1.2 Dàn	8
1.2 Một số kiến thức cơ bản về lý thuyết đồ thị	12
1.3 Hàm sinh	16
1.4 Hệ động lực rời rạc	18
Chương 2. Mô hình cột cát và phân hoạch của số tự nhiên	20
2.1 Phân hoạch số tự nhiên và hệ động lực rời rạc	21
2.1.1 Các định nghĩa và ký hiệu	21
2.1.2 Cấu trúc của $d\text{-}\mathcal{P}(n)$	23
2.1.3 Mối quan hệ giữa $d\text{-}\mathcal{P}(n+1)$ và $d\text{-}\mathcal{P}(n)$	26
2.1.4 Dàn vô hạn $d\text{-}\mathcal{P}(\infty)$	28
2.1.5 Cây vô hạn $T_{d\text{-}\mathcal{P}(\infty)}$	30
2.2 Phương pháp ECO và phân hoạch số tự nhiên	30
2.2.1 Phương pháp ECO	31

2.2.2	Phân hoạch d-chặt và phương pháp ECO	33
2.2.3	Cấu trúc đệ quy của cây vô hạn $T_{d-P(\infty)}$	34
2.3	Một số tính toán trên cây vô hạn	38
2.4	Kết luận chương 2	42
Chương 3. Các hệ động lực CFG và mạng Petri		44
3.1	CFG cổ điển	44
3.1.1	Các định nghĩa	44
3.1.2	Cấu trúc dàn của không gian trạng thái	46
3.1.3	Mô phỏng hệ SPM bằng CFG	48
3.1.4	CFG tô màu	48
3.2	Hệ động lực CCFG	52
3.3	Mạng Petri	55
3.4	Mối quan hệ giữa hệ động lực CFGs và mạng Petri	59
3.4.1	CFG và mạng Petri	59
3.4.2	CCFG và mạng Petri	61
3.4.3	CFG tô màu và mạng Petri	62
3.5	Kết luận chương 3	67
Chương 4. Tính đạt được của hệ CCFG trên đồ thị có hướng		68
4.1	Tính đạt được của một số mạng Petri	69
4.2	Cấu trúc thứ tự của CCFG trên DAG	71
4.3	Thuật toán xác định thứ tự của hệ CCFG trên DAG	75
4.3.1	Thuật toán sinh ra các lọc	77
4.3.2	Thuật toán so sánh hai trạng thái	82

4.4	Mạng vận tải	83
4.5	Tính đạt được của hệ CCFG trên đồ thị có hướng	86
4.6	Thuật toán	91
4.7	Kết luận chương 4	93
	Kết luận của luận án	94
	Các công trình liên quan đến luận án	96
	Tài liệu tham khảo	98

Danh sách các hình vẽ

1.1	Một số ví dụ về tập thứ tự.	6
1.2	Một số ví dụ về các dàn.	9
1.3	Ví dụ về đa đồ thị vô hướng (trái) và đa đồ thị có hướng (phải). . .	14
1.4	Ví dụ về đồ thị vô hướng (trái) và đồ thị có hướng (phải).	14
1.5	Ví dụ về đồ thị có hướng.	16
2.1	Luật rơi (V) và luật trượt (H) trong hệ Brylawsky.	22
2.2	Luật dọc (V) và luật ngang (H) trong trường hợp $d = 2$	24
2.3	Các phần tử đầu tiên của dàn vô hạn $2\text{-}\mathcal{P}(\infty)$	28
2.4	Cây các phân hoạch 2-chặt	31
2.5	Cấu trúc đệ quy của các cây con X_k	35
2.6	Biểu diễn cây T_{d-P} như một dây chuyền.	35
2.7	Biểu diễn cây T_P như một dây chuyền	35
2.8	Cây các phân hoạch chặt	37
2.9	Biểu diễn cây T_{SP} như một dây chuyền.	37
3.1	Quá trình chuyển trạng thái của một CFG với 9 chips.	45
3.2	Mã hoá một SPM bằng một CFG	48
3.3	CFG và không gian trạng thái tương ứng	49
3.4	Dàn ULD không là không gian trạng thái của một CFG nào	50

3.5	Không gian trạng thái của một CFG tô màu	51
3.6	Không gian trạng thái của một CCFG 2 chips	54
3.7	Ví dụ về mạng Petri.	57
3.8	Quá trình chuyển trạng thái sau một bước.	58
3.9	CFG và mạng Petri tương ứng	60
3.10	CCFG và mạng Petri tương ứng	62
3.11	CFG tô màu và mạng Petri tương ứng.	65
4.1	Không gian trạng thái của một CCFG với 2 chips.	73
4.2	Xét đỉnh 1 và đánh số lại các đỉnh.	80
4.3	Đỉnh 6 được thêm vào phản xích $\{1\}$ và được đánh số lại.	81
4.4	Đánh số lại các đỉnh liên quan đến đỉnh 2 và sinh lọc.	81
4.5	Thêm đỉnh 3, đỉnh 6, đánh số lại và sinh lọc tương ứng.	82
4.6	Thêm đỉnh 5, đánh số lại và sinh lọc.	82
4.7	Đầu vào và đầu ra của chương trình in ra các lọc	83
4.8	Một số kết quả của thuật toán so sánh hai trạng thái. Trái: đầu vào; phải: đầu ra.	84
4.9	Một trạng thái C trên đồ thị G	87
4.10	Mạng vận tải tương ứng với trạng thái c	87
4.11	Luồng cực đại f được xây dựng dựa trên luồng f_1 trong trường hợp $c_1(i) > 0, c_1(j) > 0$	90
4.12	Luồng cực đại f được xây dựng dựa trên luồng f_1 trong trường hợp $c_1(i) < 0, c_1(j) > 0$	91

Danh mục các ký hiệu

Ký hiệu	Giải thích	Trang
\leq, \geq	Quan hệ thứ tự bộ phận	5
\prec, \succ	Quan hệ phủ	6
$\phi : P \hookrightarrow Q$	Phép nhúng thứ tự	7
$\downarrow x$	ideal thứ tự sinh bởi phần tử x	7
$\downarrow Q$	ideal thứ tự sinh bởi tập con Q	7
$\uparrow x$	Lọc thứ tự sinh bởi phần tử x	7
$\uparrow Q$	Lọc thứ tự sinh bởi tập con Q	7
\vee	Cận trên bé nhất	8, 9
\wedge	Cận dưới lớn nhất	8, 9
\cong	Đẳng cấu dàn	10
$\mathcal{P}(X), 2^X$	Tập các tập con của tập X	12, 31
ULD	(Lớp các) dàn nửa phân phối trên	12
LLD	(Lớp các) dàn nửa phân phối dưới	12
$G[V']$	Đồ thị con của đồ thị G cảm sinh bởi tập đỉnh V'	14
SPM	Mô hình cột cát tuần tự (sequential Sand Piles Model)	21
$\mathcal{P}(n)$	Tập các phân hoạch của số tự nhiên n	22
$d\text{-}\mathcal{P}(n)$	Tập các phân hoạch d -chặt của số tự nhiên n	22
$d\text{-}\mathcal{P}$	Tập tất cả các phân hoạch d chặt	22
$L_B(n)$	Dàn Brylawski của số tự nhiên n	22
$\mathcal{SP}(n)$	Tập các phân hoạch chặt của số tự nhiên n	35
\leq_d	Quan hệ thứ tự trên $d\text{-}\mathcal{P}$	25
$a \downarrow^j$	Phân hoạch nhận từ a bằng cách thêm 1 vào thành phần thứ j	26
$\sqcup_{\ell \geq 1}$	Hợp rời của các tập với $\ell \geq 1$	26
$d\text{-}\mathcal{P}(\infty)$	Mở rộng vô hạn của $d\text{-}\mathcal{P}(n)$	27
\geq_∞	Quan hệ thứ tự trên $d\text{-}\mathcal{P}(\infty)$	29
$T_{d\text{-}\mathcal{P}(\infty)}, T_{d\text{-}\mathcal{P}}$	Cây các phân hoạch d -chặt của số tự nhiên	30
T_{SP}	Cây các phân hoạch chặt của số tự nhiên	35
$T_{\mathcal{P}}$	Cây các phân hoạch của số tự nhiên	36
$sp(k)$	Số các phân hoạch chặt của số tự nhiên k	40
CFG	Chip Firing Game	43
$CFG(G)$	Hệ động lực CFG trên đồ thị G	44
$CFG(G, \mathcal{O})$	Hệ động lực CFG trên đồ thị G , với trạng thái ban đầu \mathcal{O}	45
\geq_{CFG}	Quan hệ thứ tự trên $CFG(G, \mathcal{O})$	46
$\inf(a, b)$	Cận dưới bé nhất của a và b trong $CFG(G, \mathcal{O})$	47
$a \rightsquigarrow b$	b nhận được từ a trong $CFG(G, \mathcal{O})$	47
$L(CFG)$	Lớp dàn sinh bởi các CFG hội tụ	48
\mathcal{D}	Lớp các dàn phân phối	48
$ColCFG(G)$	CFG tô màu trên đồ thị G	50
CCFG	CFG tương tranh (Conflicting Chip Firing Game)	52
$CCFG(G, n)$	Hệ động lực CCFG trên đồ thị G , tổng số chip n	52
\geq_{CCFG}	Quan hệ thứ tự trên $CCFG(G, n)$	52
$\mathcal{F}(V)$	Tập tất cả các lọc thứ tự của V	72

Mở đầu

Năm 1987, Bak, Tang và Wiesenfeld [7, 8] đã đưa ra vấn đề đột biến tự tổ chức (Self Organization Criticality - SOC) trong vật lý: khi một hệ đang ở trạng thái ổn định (steady state, critical state) được nhiễu bằng một tác động nhỏ, thì hệ sẽ biến đổi đến một trạng thái ổn định mới. Tác động nhỏ này có thể gây nên những biến đổi lớn của hệ. Chẳng hạn như hiện tượng tuyết lở hay hiện tượng cát lở, chỉ cần sự chuyển động nhỏ mang tính địa phương của từng hạt (grain) có thể gây nên những biến đổi lớn toàn cục của cả núi tuyết hay các cột cát (sand piles). Đây là một trong những đặc trưng của hiện tượng SOC. Hiện tượng này thường xảy ra đối với các hệ vật lý trong tự nhiên và được các nhà Vật lý học trên mô hình hóa thành mô hình SPM (Sand Piles Model) của toán rời rạc. Từ đó có rất nhiều nghiên cứu về hiện tượng SOC và hệ SPM [20], [22], [24], [25], [26], [27], [28], [30], [44], [78]. Hệ SPM đã được nghiên cứu trong nhiều lĩnh vực khác nhau với nhiều cách tiếp cận khác nhau, điển hình là các công trình của Dhar (1990) [20, 21, 25] và Cori, Rossin (1998) [18] nghiên cứu hệ SPM bằng cách tiếp cận đại số và liên hệ với cây bao trùm của đồ thị; Goles và Kiwi [27] nghiên cứu các điểm dừng của hệ SPM. Đặc biệt, vào những năm 1990, Bjorner, Lovász và Shor [4, 5] đã nghiên cứu hệ động lực CFG - một mở rộng của hệ SPM - bằng cách tiếp cận của lý thuyết ngôn ngữ; N. Biggs (1993) [3] nghiên cứu tính hội tụ của một số hệ kinh tế để tìm ra các thời điểm có những biến động lớn. Morvan, Goles và Phan [30, 31, 32, 33, 34, 64] đã sử dụng cấu trúc dàn để chứng minh tính hội tụ; Phan, Latapy và Lê [52, 54, 57] đã sử dụng phương pháp cây hàm sinh nghiên cứu các mở rộng vô hạn của một số hệ cơ bản, tìm ra tính chất truy hồi của chúng và xây dựng một số thuật toán cũng

như chương trình mô phỏng hệ.

Mục đích của luận án này là nghiên cứu các hệ theo hướng tiếp cận cấu trúc của không gian trạng thái. Luận án sử dụng cấu trúc dàn để tìm hiểu tính hội tụ của các hệ mới, về các điểm đột biến của chúng và sử dụng kỹ thuật đếm bằng phương pháp ECO (Enumeration of Combinatorial Objects) để tính toán lực lượng của hệ. Việc chứng minh cấu trúc dàn của không gian trạng thái (configuration space) của hệ cho phép xác định tính hội tụ và trong một số trường hợp có thể chỉ ra được điểm dừng hay điểm đột biến của hệ. Ngoài ra, cấu trúc dàn cho phép xác định tính đạt được: những trạng thái đạt được từ hai trạng thái a và b cho trước có thể đạt được từ trạng thái $c = a \wedge b$, c là cận dưới lớn nhất của a và b . Phương pháp ECO là phương pháp mới [9], [10] rất hữu hiệu trong việc tính toán lực lượng của các hệ nhờ vào cấu trúc đệ quy của cây ECO, và có mối liên hệ chặt chẽ với hàm sinh. Tiếp theo, chúng tôi tìm hiểu mối quan hệ giữa các hệ CFG và mở rộng của nó với các hệ tin học nổi tiếng (mạng Petri). Mạng Petri đã được định nghĩa từ những năm 1962 [67], và đã được nghiên cứu trong nhiều công trình [13], [39], [42], [43], [45], [63], [65], [66], [68], [77]. Việc chứng minh mối liên hệ giữa các hệ CFG và mạng Petri cho phép sử dụng các phương pháp nghiên cứu cũng như các thuật toán của mạng Petri vào nghiên cứu các hệ CFG. Cuối cùng, chúng tôi sử dụng lý thuyết tập sắp thứ tự (order theory) để nghiên cứu cấu trúc thứ tự của không gian trạng thái của các hệ CFG mở rộng. Đặc biệt, chúng tôi còn tìm hiểu mối liên hệ giữa các hệ CFG mở rộng và lý thuyết luồng trong mạng để giải bài toán đạt được (reachability problem) của hệ CFG mở rộng. Bài toán đạt được là một bài toán quan trọng trong việc nghiên cứu các hệ. Một mặt nó cho biết các trạng thái nào có thể xảy ra, các trạng thái nào không bao giờ xảy ra. Mặt khác, nó cho ta biết mối quan hệ giữa các trạng thái, từ trạng thái nào được đến trạng thái nào. Trong trường hợp mạng Petri tổng quát, đây là bài toán mở. Chỉ có một số ít trường hợp giải được trong thời gian đa thức, còn nhiều trường hợp đã được chứng minh là NP đầy đủ. Trong luận án, chúng tôi đã xây dựng thuật toán giải bài toán đạt được của

hệ CCFG trong thời gian $O(|V|^3)$, trong đó $|V|$ là số đỉnh của đồ thị nền.

Luận án được chia làm 4 chương. Trong Chương 1, chúng tôi nhắc lại một số kiến thức cơ bản đã biết sẽ được sử dụng trong luận án như: lý thuyết tập sắp thứ tự, lý thuyết dàn, một số khái niệm liên quan đến lý thuyết đồ thị, phương pháp đếm bằng hàm sinh. Phần cuối chương này sẽ trình bày các khái niệm về hệ động rời rạc và một số bài toán liên quan.

Các kết quả mới của chúng tôi được trình bày trong các Chương 2, 3 và 4.

Nội dung của Chương 2 dựa trên kết quả của bài báo [56]. Trong chương này, chúng tôi nghiên cứu mối liên hệ giữa các mô hình cột cát mở rộng và phân hoạch của số tự nhiên. Hệ cột cát (Sand Piles Model - SPM) là một hệ động lực quan trọng được đề xuất bởi ba nhà Vật lý Bak, Tang và Wiesenfeld vào năm 1987 [7] để mô hình hóa hiện tượng đột biến tự tổ chức (Self-Organized Criticality - SOC). Hệ SPM này đã được chứng minh là một trường hợp đặc biệt của hệ Chip Firing Game (CFG) [30]. Theo các nghiên cứu [20], [21], [27], [28], [30], [31], [33], [34], ... mô hình cột cát có liên quan chặt chẽ với phân hoạch của số tự nhiên. Trong chương này, chúng tôi sẽ xét đến các mô hình cột cát với ngưỡng d cho luật vận động và mối liên hệ của chúng với các phân hoạch d -chặt của số tự nhiên. Phương pháp chính được sử dụng ở đây là phương pháp ECO (Enumeration of Combinatorial Objects), một phương pháp tính toán tổ hợp sử dụng cây sinh và được phát triển trong những năm gần đây [9], [10]. Phương pháp này cho phép chúng tôi chứng minh cấu trúc của không gian trạng thái và tính toán số các trạng thái của mô hình. Bên cạnh đó, nhờ có phương pháp này chúng tôi cũng nghiên cứu được cấu trúc đệ quy của tập các phân hoạch d -chặt và đưa ra chứng minh cho một số đẳng thức tổ hợp.

Chương 3 nghiên cứu về mối quan hệ giữa các hệ CFG và mạng Petri. Nội dung của chương này dựa trên kết quả của bài báo [58]. Trong phần đầu chương 3, chúng tôi nhắc lại các kết quả đã biết về hệ động lực CFG và các mở rộng của nó. Tiếp theo chúng tôi chứng minh song ánh giữa các hệ CFG và một số mạng Petri đặc

biệt.

Chương 4 của luận án được viết dựa trên kết quả của các bài báo [53, 55, 59]. Trong chương này chúng tôi nghiên cứu cấu trúc không gian trạng thái và bài toán đạt được của hệ động lực CCFG (**Conflicting Chip Firing Game - CFG tương tranh**) - một mở rộng của hệ động lực CFG. Phần đầu chương này chúng tôi nhắc lại bài toán đạt được của một số mạng Petri đặc biệt. Phần tiếp theo của Chương 4, chúng tôi nghiên cứu cấu trúc thứ tự của không gian trạng thái của hệ CCFG trên đồ thị có hướng không chu trình. Chúng tôi đưa ra khái niệm họ năng lượng của các trạng thái của hệ để đặc trưng cho thứ tự của không gian trạng thái và chúng tôi xây dựng thuật toán để xác định thứ tự này. Phần cuối chương này, chúng tôi nghiên cứu bài toán đạt được của hệ CCFG trên đồ thị có hướng tổng quát. Chúng tôi đưa ra khái niệm *mạng vận tải tương ứng* với trạng thái của hệ để đặc trưng cho tính đạt được của hệ CCFG. Chúng tôi sử dụng thuật toán Push-Relabel, một biến thể của thuật toán Ford-Fulkerson để giải bài toán đạt được của hệ CCFG trong thời gian $O(m^3)$ với m là số đỉnh của đồ thị nền của hệ CCFG.

Trong phần kết luận của luận án, chúng tôi tóm tắt lại các kết quả đã đạt được và nêu một số hướng nghiên cứu tiếp theo.

Chương 1

Kiến thức chuẩn bị

Trong chương này chúng tôi trình bày một số kiến thức cơ sở và một số kết quả đã biết về tập sắp thứ tự, dàn, hàm sinh, đồ thị và một số khái niệm và bài toán trong lý thuyết hệ động lực rời rạc nhằm giúp cho việc trình bày các kết quả trong các chương sau. Các kiến thức trong chương này được tham khảo trong các tài liệu [1, 11, 23, 73, 75, 76, 80].

1.1 Tập thứ tự - Dàn

1.1.1 Tập thứ tự

Định nghĩa 1.1.1. Cho P là một tập hợp. Một *thứ tự* (hay *thứ tự bộ phận*) trên P là một quan hệ hai ngôi \leq trên P thỏa mãn 3 tính chất sau với mọi $x, y, z \in P$,

- + Tính phản xạ: $x \leq x$,
- + Tính phản đối xứng: nếu $x \leq y$ và $y \leq x$ thì $x = y$,
- + Tính bắc cầu: nếu $x \leq y$ và $y \leq z$ thì $x \leq z$.

Một tập P được trang bị quan hệ thứ tự \leq được gọi là một *tập thứ tự* (*ordered set*) hay là *tập thứ tự bộ phận* (*partially ordered set*) và ký hiệu là (P, \leq) khi cần nhắc đến quan hệ thứ tự \leq .

Cho P là một tập thứ tự và Q là một tập con của P . Khi đó trên Q cảm sinh

một thứ tự từ P như sau: với mọi $x, y \in Q, x \leq y$ trong Q khi và chỉ khi $x \leq y$ trong P , và ta gọi (Q, \leq) là một tập thứ tự con của (P, \leq) .

Từ đây trở về sau ta ký hiệu P là một tập thứ tự.

Định nghĩa 1.1.2. Cho P là một tập thứ tự. Khi đó P được gọi là một *dây chuyền* (*chain*) nếu với mọi $x, y \in P$ ta có $x \leq y$ hoặc $y \leq x$, tức là hai phần tử bất kỳ trong P đều so sánh được với nhau. Tập thứ tự P được gọi là một *phản xích* (*antichain*) nếu với mọi $x, y \in P$ mà $x \leq y$ thì $x = y$, tức là hai phần tử bất kỳ khác nhau trong P không so sánh được với nhau.

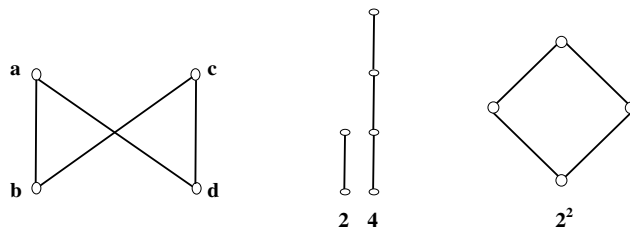
Định nghĩa 1.1.3. Cho P là một tập thứ tự, $x, y \in P$. Ta nói rằng *phần tử y phủ phần tử x* (y covers x) và ký hiệu là $x \prec y$ hay $y \succ x$ nếu $x < y$ và với mọi $z \in P$ mà $x \leq z < y$ thì $x = z$.

Biểu đồ Hasse (Hasse diagrams): Cho P là một tập thứ tự hữu hạn. Khi đó ta có thể biểu diễn các phần tử của P bởi các hình tròn nhỏ hay các điểm và các đoạn thẳng nối giữa các phần tử của P để chỉ quan hệ phủ. Biểu đồ Hasse biểu diễn tập thứ tự P được xây dựng như sau:

+ Với mỗi phần tử $x \in P$, cho tương ứng với một điểm $P(x)$ trong mặt phẳng \mathbb{R}^2 .

+ Với mỗi quan hệ phủ $x \prec y$, vẽ đoạn thẳng $\ell(x, y)$ nối điểm $P(x)$ với $P(y)$ sao cho $P(x)$ thấp hơn (theo nghĩa tọa độ thứ hai của $P(x)$ nhỏ hơn) $P(y)$.

Sau đây là một số ví dụ:



Hình 1.1: Một số ví dụ về tập thứ tự.

Mối quan hệ giữa các tập thứ tự được nhắc lại trong định nghĩa sau

Định nghĩa 1.1.4. Cho P và Q là các tập thứ tự. Khi đó, ánh xạ $\phi : P \rightarrow Q$ được gọi là

(i) *bảo toàn thứ tự (order-preserving)* nếu với mọi $x, y \in P$ mà $x \leq y$ thì $\phi(x) \leq \phi(y)$ trong Q ,

(ii) *một phép nhúng thứ tự (order embedding)* nếu với mọi $x, y \in P$, $x \leq y$ khi và chỉ khi $\phi(x) \leq \phi(y)$ trong Q ,

(iii) *một đẳng cấu thứ tự (order isomorphism)* nếu ϕ là một phép nhúng thứ tự và ϕ là một song ánh.

Nếu ϕ là một phép nhúng thứ tự thì ta ký hiệu $\phi : P \hookrightarrow Q$. Nếu tồn tại một đẳng cấu thứ tự giữa P và Q thì ta nói P đẳng cấu với Q và ký hiệu là $P \cong Q$.

Một trong những họ tập thứ tự quan trọng là *ideal thứ tự (order ideal)* và *lọc thứ tự (order filter)* được nhắc lại trong định nghĩa sau:

Định nghĩa 1.1.5. Cho P là một tập thứ tự và Q là một tập con của P . Khi đó:

(i) Q được gọi là một *ideal thứ tự (order ideal)* nếu với mọi $x \in Q, y \in P$ mà $y \leq x$ thì $y \in Q$,

(ii) Q được gọi là một *lọc thứ tự (order filter)* nếu với mọi $x \in Q, y \in P$ mà $y \geq x$ thì $y \in Q$.

Từ định nghĩa trên ta thấy rằng Q là ideal thứ tự khi và chỉ khi $P \setminus Q$ là lọc thứ tự. Sau này, để cho gọn, đôi khi ta nói ideal (tương ứng, lọc) thay cho ideal thứ tự (tương ứng, lọc thứ tự). Bây giờ, với $Q \subseteq P, x \in P$ ta có các ký hiệu sau:

$$\downarrow Q := \{y \in P \mid \exists x \in Q : y \leq x\}, \uparrow Q := \{y \in P \mid \exists x \in Q : y \geq x\},$$

$$\downarrow x := \{y \in P \mid y \leq x\}, \uparrow x := \{y \in P \mid y \geq x\}.$$

Từ các định nghĩa này, ta dễ dàng kiểm tra được $\downarrow Q$ là ideal bé nhất chứa Q và $\uparrow Q$ là lọc bé nhất chứa Q . $\downarrow Q$ (tương ứng, $\uparrow Q$) còn được gọi là ideal (tương ứng,

lọc) thứ tự sinh bởi Q . Họ tất cả các ideal (tương ứng, lọc) của P được ký hiệu là $\mathcal{O}(P)$ (tương ứng, $\mathcal{F}(P)$). $\mathcal{O}(P)$ và $\mathcal{F}(P)$ cũng là các tập thứ tự với quan hệ thứ tự bao hàm tập hợp.

Tiếp theo, chúng tôi sẽ nhắc lại định nghĩa một số phần tử đặc biệt của tập thứ tự như phần tử nhỏ nhất, lớn nhất, cực đại, cực tiểu.

Định nghĩa 1.1.6. Cho P là một tập thứ tự và $Q \subseteq P$. Khi đó

(i) phần tử $a \in Q$ được gọi là phần tử *cực đại* của Q nếu với mọi $x \in Q$ mà $a \leq x$ thì $x = a$,

(ii) phần tử $a \in Q$ được gọi là phần tử *lớn nhất* của Q nếu với mọi $x \in Q$ ta có $x \leq a$.

Các khái niệm phần tử cực tiểu và phần tử nhỏ nhất được định nghĩa đối ngẫu.

1.1.2 Dàn

Một số tính chất quan trọng của tập thứ tự P được thể hiện bởi sự tồn tại của cận trên và cận dưới của các tập con của P . Các tính chất này cho phép ta định nghĩa khái niệm dàn và dàn đầy đủ. Trước hết ta có các định nghĩa về cận trên và cận dưới.

Định nghĩa 1.1.7. Cho P là một tập thứ tự và $S \subseteq P$. Phần tử $x \in P$ được gọi là một *cận trên* của S nếu $s \leq x$ với mọi $s \in S$. Phần tử $x \in P$ được gọi là *cận trên nhỏ nhất* của S nếu

+ x là một cận trên của S , và

+ $x \leq y$ với mọi cận trên y của S .

Khái niệm *cận dưới* và *cận dưới lớn nhất* được định nghĩa đối ngẫu.

Cận trên nhỏ nhất (tương ứng, cận dưới lớn nhất) của tập S (nếu tồn tại) được ký hiệu là $\bigvee S$ (tương ứng, $\bigwedge S$). Đặc biệt, cận trên nhỏ nhất (tương ứng, cận dưới

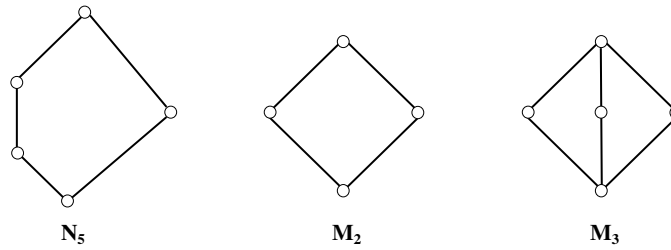
lớn nhất) của hai phần tử x và y được ký hiệu là $x \vee y$ (tương ứng, $x \wedge y$).

Sau đây, chúng ta sẽ quan tâm đến các tập thứ tự P mà với mọi phần tử $x, y \in P$ đều tồn tại cận trên nhỏ nhất và cận dưới lớn nhất.

Định nghĩa 1.1.8. Cho L là một tập thứ tự khác rỗng. Khi đó

(i) Nếu $x \vee y$ và $x \wedge y$ tồn tại với mọi $x, y \in L$ thì L được gọi là một *dàn* (lattice).

(ii) Nếu $\bigvee S$ và $\bigwedge S$ tồn tại với mọi $S \subseteq L$ thì L được gọi là một *dàn đầy đủ* (complete lattice).



Hình 1.2: Một số ví dụ về các dàn.

Nếu L là một dàn thì các toán tử \vee và \wedge là các phép toán hai ngôi trên L . Khi đó ta có cấu trúc đại số $\langle L, \vee, \wedge \rangle$. Cấu trúc dàn con được định nghĩa như sau:

Định nghĩa 1.1.9. Cho L là một dàn, M là một tập con của L . Khi đó M được gọi là một *dàn con* của dàn L nếu với mọi $a, b \in M$, ta có $a \vee b \in M$ và $a \wedge b \in M$.

Như vậy, một tập con của L là dàn con của dàn L nếu nó đóng đối với các phép toán \vee, \wedge .

Dàn L được gọi là có phần tử đơn vị nếu tồn tại $1 \in L$ sao cho với mọi $a \in L, a = a \wedge 1$. Đối ngẫu lại, dàn L được gọi là có phần tử không nếu tồn tại $0 \in L$ sao cho với mọi $a \in L, a = a \vee 0$.

Một dàn hữu hạn luôn bị chặn bởi $0 = \bigwedge L$ và $1 = \bigvee L$.

Giống như mối quan hệ giữa các tập thứ tự, các dàn quan hệ với nhau thông qua các ánh xạ và được thể hiện qua định nghĩa sau đây:

Định nghĩa 1.1.10. Cho L và K là các dàn. Khi đó, ánh xạ $f : L \rightarrow K$ được gọi là một *đồng cấu* hay *đồng cấu dàn* nếu f bảo toàn các phép toán \vee và \wedge , tức là, với mọi $a, b, c \in L$:

$$f(a \vee b) = f(a) \vee f(b) \text{ và } f(a \wedge b) = f(a) \wedge f(b).$$

Nếu đồng cấu $f : L \rightarrow K$ là một song ánh thì ta có đẳng cấu dàn $L \cong K$.

Nếu đồng cấu $f : L \rightarrow K$ là một đơn ánh thì dàn con $f(L)$ của dàn K đẳng cấu với dàn L và ta nói f là một phép nhúng (dàn) L vào K .

Nếu L và K là các dàn bị chặn bởi 0 và 1 thì ta thường xét các ánh xạ $f : L \rightarrow K$ bảo toàn 0 và 1, tức là $f(0) = 0, f(1) = 1$. Các ánh xạ này được gọi là các $\{0, 1\}$ -đồng cấu.

Một số tính chất của đồng cấu dàn được thể hiện trong mệnh đề sau:

Mệnh đề 1.1.11. Cho L, K là các dàn và $f : L \rightarrow K$ là một ánh xạ. Khi đó:

(i) Các khẳng định sau là tương đương:

(a) f bảo toàn thứ tự;

(b) $(\forall a, b \in L) f(a \vee b) \geq f(a) \vee f(b)$;

(c) $(\forall a, b \in L) f(a \wedge b) \leq f(a) \wedge f(b)$;

Đặc biệt, nếu f là một đồng cấu dàn thì f bảo toàn thứ tự.

(ii) Các khẳng định sau là tương đương:

(a) f là đẳng cấu thứ tự;

(b) f là song ánh và là phép nhúng thứ tự;

(c) f là đẳng cấu dàn.

Một trong những lớp dàn có nhiều ứng dụng là dàn modula và dàn phân phối. Các lớp dàn này là không gian trạng thái của một số hệ động lực được xét đến trong các chương sau. Sau đây, chúng tôi nhắc lại tính chất modula và tính chất

phân phối của dàn cũng như các đặc trưng của chúng.

Định nghĩa 1.1.12. Cho L là một dàn. Khi đó L được gọi là

(i) *dàn phân phối* nếu L thỏa mãn luật phân phối:

$$(\forall a, b, c \in L) a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c).$$

(ii) *dàn modula* nếu L thỏa mãn luật modula:

$$(\forall a, b, c \in L) a \geq c \Rightarrow a \wedge (b \vee c) = (a \wedge b) \vee c.$$

Như vậy, nếu một dàn là phân phối thì modula. Ngược lại thì không đúng. Tồn tại những dàn modula nhưng không phân phối.

Ví dụ 1.1.13. Dàn M_3 (diamond) là dàn modula nhưng không phân phối. Dàn N_5 (pentagon) không phải là dàn modula và do đó cũng không là dàn phân phối.

Đặc trưng của dàn modula và dàn phân phối được trình bày trong Định lý sau đây. Định lý này có tên gọi là Định lý $M_3 - N_5$ vì các dàn M_3 và N_5 đặc trưng cho tính chất modula và tính chất phân phối.

Định lý 1.1.14. Cho L là một dàn. Khi đó:

(i) Dàn L không phải là dàn modula khi và chỉ khi L chứa dàn N_5 như là một dàn con.

(ii) Dàn L không phải là dàn phân phối khi và chỉ khi L chứa dàn M_3 hoặc dàn N_5 như là một dàn con.

Một lớp dàn đóng vai trò quan trọng trong việc nghiên cứu cấu trúc đại số của dàn và ứng dụng nhiều trong khoa học máy tính là lớp dàn Bun (Boolean lattice). Để định nghĩa dàn Bun trước hết ta định nghĩa phần tử bù (complement) của một phần tử.

Định nghĩa 1.1.15. Cho L là dàn có 0 và 1, a là một phần tử của L . Khi đó, phần tử $b \in L$ được gọi là được gọi là phần tử bù (complement) của a nếu $a \wedge b = 0$ và $a \vee b = 1$. Nếu a có duy nhất một phần tử bù thì ta ký hiệu phần tử bù của a là a' .

Định nghĩa 1.1.16. Dàn L được gọi là *dàn Bun (Boolean lattice)* nếu:

- (i) L là dàn phân phối,
- (ii) L có 0 và 1,
- (iii) mỗi phần tử $a \in L$ có duy nhất phần tử bù $a' \in L$.

Như vậy, dàn Bun là một dàn phân phối đặc biệt, trong đó có chứa các phần tử 0, 1 và toán tử lấy phần bù ' là một phần trong cấu trúc của dàn Bun.

Một ví dụ đơn giản nhất về dàn Bun là dàn $\mathcal{P}(X)$ các tập con của tập X . Trong dàn này, phần tử 0 chính là tập hợp rỗng, phần tử 1 chính là X , mỗi phần tử $A \in \mathcal{P}(X)$ có phần tử bù chính là tập hợp $X \setminus A$, phần bù của tập A trong tập X .

Một kết quả đã biết về biểu diễn dàn hữu hạn là: mọi dàn Bun hữu hạn đều đẳng cấu với dàn $\mathcal{P}(X)$, với một X nào đấy. Dàn $\mathcal{P}(X)$ còn có tên gọi là dàn siêu khối (hypercube).

Một dàn được gọi là nửa phân phối trên (upper locally distributive), ký hiệu là ULD [62], nếu mỗi đoạn giữa một phần tử và cận trên bé nhất của các phủ trên (upper cover) của nó là một siêu khối. Dàn nửa phân phối dưới (lower locally distributive - LLD) được định nghĩa đối ngẫu. Một trong những đặc trưng của tính chất phân phối là [74]: *dàn L là dàn phân phối khi và chỉ khi L vừa là dàn LLD vừa là dàn ULD.*

1.2 Một số kiến thức cơ bản về lý thuyết đồ thị

Trong phần này chúng tôi nhắc lại một số khái niệm liên quan đến lý thuyết đồ thị sẽ sử dụng trong các chương sau.

Định nghĩa 1.2.1. Một *đồ thị vô hướng* là một cặp có thứ tự $G = (V, E)$, ở đây V là một tập, E là tập với các phần tử là các đa tập 2 phần tử trên V .

Các phần tử của V gọi là các *đỉnh*, còn các phần tử của E được gọi là các *cạnh* của đồ thị vô hướng G . Nếu $e = \{a, b\}$ là một cạnh của G thì a và b được gọi là các đỉnh đầu mút của của cạnh e hay *các đỉnh liên thuộc với e* . Ta thường ký hiệu cạnh $\{a, b\}$ ngắn gọn là ab .

Định nghĩa 1.2.2. Một *đồ thị có hướng* là một cặp có thứ tự $G = (V, E)$, ở đây V là một tập, E là một tập con của tập tích Đề-các $V \times V$, tức là E là một quan hệ hai ngôi trên V .

Các phần tử của V gọi là các *đỉnh*, còn các phần tử của E được gọi là các *cung* của đồ thị có hướng G . Cụ thể hơn, nếu $(a, b) \in E$ thì (a, b) được gọi là cung của G với đỉnh đầu là a , đỉnh cuối là b và có hướng đi từ a đến b .

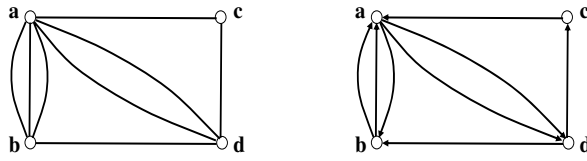
Đồ thị có hướng được định nghĩa như trên cũng thường được gọi là *đơn đồ thị có hướng*. Lý do là vì với hai đỉnh a và b bất kỳ tồn tại nhiều nhất một cung với đỉnh đầu là a và đỉnh cuối là b . Tương tự, đồ thị vô hướng được định nghĩa như trên cũng được gọi là *đơn đồ thị vô hướng*.

Trong trường hợp giữa các cặp đỉnh có thể có nhiều cung (đồ thị có hướng) hay nhiều cạnh (đồ thị vô hướng), thì ta có khái niệm *đa đồ thị* được định nghĩa như sau:

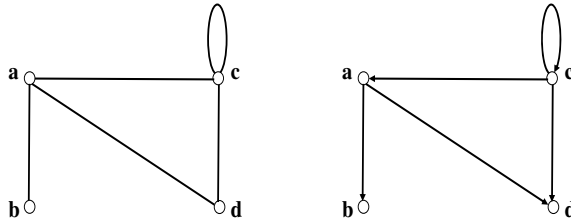
Định nghĩa 1.2.3. Một *đa đồ thị vô hướng* G là một cặp có thứ tự $G = (V, E)$, ở đây V là một tập còn E là một đa tập với các phần tử đều là đa tập 2 phần tử trên V . Tương tự, một *đa đồ thị có hướng* G là một cặp có thứ tự $G = (V, E)$, ở đây V là một tập còn E là một đa tập với các phần tử đều thuộc tích Đề-các $V \times V$.

Người ta thường biểu diễn đồ thị trên mặt phẳng như sau. Các đỉnh của G được biểu diễn bằng các vòng tròn nhỏ, các cạnh (hay cung) được biểu diễn bằng một đường cong nối các đỉnh với cạnh, mũi tên để chỉ hướng từ đỉnh đầu đến đỉnh cuối đối với đồ thị có hướng.

Có những đồ thị khác nhau nhưng sau khi đổi tên các đỉnh của các đồ thị đó



Hình 1.3: Ví dụ về đa đồ thị vô hướng (trái) và đa đồ thị có hướng (phải).



Hình 1.4: Ví dụ về đồ thị vô hướng (trái) và đồ thị có hướng (phải).

thì chúng lại trùng nhau. Những đồ thị như thế được gọi là *đẳng cấu* và trong lý thuyết đồ thị ta thường đồng nhất chúng. Cụ thể hơn, đồ thị có hướng (tương ứng, vô hướng) $G = (V, E)$ và $G' = (V', E')$ được gọi là *đẳng cấu* với nhau nếu tồn tại song ánh $\varphi : V \rightarrow V'$ sao cho $(a, b) \in E$ (tương ứng, $\{a, b\} \in E$) khi và chỉ khi $(\varphi(a), \varphi(b)) \in E'$ (tương ứng, $\{\varphi(a), \varphi(b)\} \in E'$). Song ánh φ như trên được gọi là đẳng cấu của G và G' . Hai đồ thị đẳng cấu với nhau G và G' được ký hiệu là $G \cong G'$.

Định nghĩa 1.2.4. Đồ thị $G' = (V', E')$ được gọi là *đồ thị con* của đồ thị $G = (V, E)$ nếu $V' \subseteq V$ và $E' \subseteq E$. Đồ thị con $G' = (V', E')$ của đồ thị $G = (V, E)$ được gọi là *đồ thị con bao trùm* của G nếu $V' = V$. Nếu E' chứa tất cả các cung hay các cạnh của G , mà cả hai đỉnh liên thuộc của nó đều thuộc V' thì $G' = (V', E')$ được gọi là *đồ thị con của $G = (V, E)$ cảm sinh bởi tập đỉnh V'* . Khi đó, G' cũng được ký hiệu là $G' = G[V']$.

Định nghĩa 1.2.5. Giả sử $G = (V, E)$ là một đồ thị có hướng. Một *đường đi có hướng* trong G là một dãy $v_0v_1v_2 \dots v_n$ sao cho $v_i \in V$ với mọi $i = 0, 1, \dots, n$ và $(v_{i-1}, v_i) \in E$ với mọi $i = 0, 1, \dots, n$.

Trong định nghĩa trên, đỉnh v_0 được gọi là *đỉnh đầu*, còn v_n được gọi là *đỉnh*

cuối trong đường đi có hướng $v_0v_1v_2 \dots v_n$. Một *chu trình có hướng* là một đường đi có hướng với đỉnh đầu và đỉnh cuối trùng nhau. Khái niệm *đường đi vô hướng* và *chu trình vô hướng* được định nghĩa tương tự.

Định nghĩa 1.2.6. Cho đồ thị vô hướng G . *Bậc* (degree) của đỉnh v , ký hiệu $d_G(v)$ hay $\deg(v)$ là số cạnh liên thuộc với v , trong đó khuyên được tính hai lần.

Định nghĩa 1.2.7. Cho đồ thị có hướng G . *Bậc đi ra* (out-degree) của đỉnh v , ký hiệu $\deg^+(v)$, là số cung xuất phát từ đỉnh v . *Bậc đi vào* (in-degree) của đỉnh v , ký hiệu $\deg^-(v)$, là số cung đi vào đỉnh v . *Bậc* của đỉnh v là tổng bậc đi ra và bậc đi vào của đỉnh đó: $\deg(v) = \deg^+(v) + \deg^-(v)$.

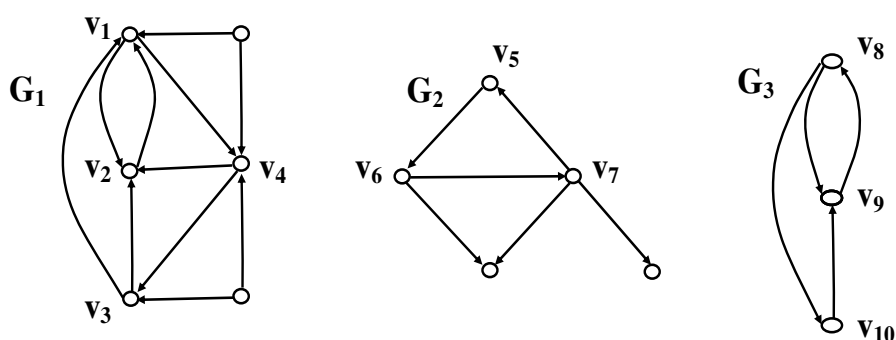
Định nghĩa 1.2.8. Đồ thị (có hướng, vô hướng) G được gọi là *liên thông* nếu giữa hai đỉnh bất kỳ đều tồn tại một đường đi vô hướng. Đồ thị có hướng G gọi là *liên thông mạnh* nếu giữa hai đỉnh bất kỳ đều tồn tại một đường đi có hướng.

Tập $S \subseteq V$ của một đồ thị (có hướng, vô hướng) $G = (V, E)$ được gọi là một *thành phần liên thông* của G , nếu $G[S]$ liên thông và với mọi $V'' \subseteq V$ và V'' thực sự chứa S , đồ thị $G[V'']$ là không liên thông.

Tương tự, tập $S \subseteq V$ của một đồ thị có hướng $G = (V, E)$ được gọi là một *thành phần liên thông mạnh* của G nếu $G[S]$ liên thông mạnh và với mọi $V'' \subseteq V$ và V'' thực sự chứa S , đồ thị $G[V'']$ là không liên thông mạnh.

Định nghĩa 1.2.9. Cho đồ thị có hướng $G = (V, E)$. Một *Thành phần đóng* (closed component) S của đồ thị G là một thành phần liên thông mạnh $S \subseteq V$ mà không có cung ra ngoài, tức là không tồn tại cung nào từ một đỉnh bất kỳ trong S đến một đỉnh bất kỳ trong $V \setminus S$.

Ví dụ 1.2.10. Đồ thị có hướng G gồm 3 thành phần liên thông (Hình 1.5) trong đó có ba thành phần liên thông mạnh là $\{v_1, v_2, v_3, v_4\}$, $\{v_5, v_6, v_7\}$, $\{v_8, v_9, v_{10}\}$ và hai thành phần đóng là $\{v_1, v_2, v_3, v_4\}$, $\{v_8, v_9, v_{10}\}$.



Hình 1.5: Ví dụ về đồ thị có hướng.

1.3 Hàm sinh

Trong phần này chúng tôi nhắc lại phương pháp đếm rất hữu hiệu là phương pháp đếm dùng hàm sinh. Khái niệm quan trọng làm cơ sở cho việc nghiên cứu và phát triển phương pháp này là chuỗi lũy thừa hình thức. Bằng các phép toán được định nghĩa trên các đối tượng này ta có thể biến chúng thành một cấu trúc đại số. Vì thế mà ta sử dụng được các công cụ cũng như các kết quả của các lĩnh vực toán học khác vào việc giải quyết các bài toán đếm trong tổ hợp. Giả sử cần đếm đối tượng tổ hợp phụ thuộc vào số tự nhiên n . Gọi a_n là số các đối tượng tổ hợp này. Ta cần phải biểu diễn a_n dưới dạng tường minh theo n . Hai loại hàm sinh thường dùng là *hàm sinh thường* (ordinary generating function) và *hàm sinh mũ* (exponential generating function). Sau đây, chúng tôi chỉ trình bày khái niệm hàm sinh thường, được sử dụng trong các chương sau.

Hàm sinh thường của dãy số $(a_n)_0^\infty$ là chuỗi lũy thừa hình thức $\sum_{n=0}^{\infty} a_n x^n$; trong đó a_n gọi là hệ số của x^n .

Sau đó, bằng các phương pháp đại số và giải tích, ta có thể tính được hàm sinh $A(x)$. Ta dùng khai triển Taylor của $A(x)$:

$$A(x) = b_0 + b_1 x + b_2 x^2 + \dots$$

Từ định nghĩa hàm sinh $A(x)$ và khai triển này, ta có $a_n = b_n$ và do đó ta có công

thức tường minh cho a_n .

Ký hiệu $\mathbb{C}[[x]]$ là tập tất cả các chuỗi lũy thừa hình thức, hệ số phức. Khi đó, theo quan điểm đại số giao hoán, $\mathbb{C}[[x]]$ là một vành địa phương chính quy đầy đủ một chiều. Để xác định sự tồn tại của các chuỗi lũy thừa hình thức, chúng tôi trình bày khái niệm *hội tụ* trong $\mathbb{C}[[x]]$.

Định nghĩa 1.3.1. Cho $F_1(x), F_2(x), \dots$ là một dãy các chuỗi lũy thừa hình thức và $F(x) = \sum_{n \geq 0} a_n x^n \in \mathbb{C}[[x]]$. Khi đó, ta nói dãy $F_i(x)$ *hội tụ* về $F(x)$ khi $i \rightarrow \infty$, (ký hiệu là $F_i(x) \rightarrow F(x)$), nếu với mọi $n \geq 0$, tồn tại $\delta(n)$ sao cho các hệ số của x^n trong $F_i(x)$ là a_n với mọi $i \geq \delta(n)$.

Một định nghĩa tương đương về sự hội tụ trong $\mathbb{C}[[x]]$ như sau: Trước hết, ta định nghĩa *bậc* của chuỗi lũy thừa hình thức $F(x) = \sum_{n \geq 0} a_n x^n$, ký hiệu $\deg F(x)$, là số tự nhiên n nhỏ nhất sao cho $a_n \neq 0$. Khi đó, $F_i(x)$ hội tụ khi và chỉ khi $\lim_{i \rightarrow \infty} \deg(F_{i+1}(x) - F_i(x)) = \infty$.

Ta định nghĩa tổng vô hạn $\sum_{j \geq 0} F_j(x)$ có giá trị $F(x)$ nếu $\sum_{j=0}^i F_j(x) \rightarrow F(x)$. Ta cũng định nghĩa tương tự cho trường hợp tích vô hạn $\prod_{j \geq 1} F_j(x)$. Trong trường hợp tích vô hạn, ta giả thiết mỗi phân tử $F_j(x)$ thỏa mãn $F_j(0) = 1$.

Từ các định nghĩa trên ta có hai mệnh đề sau:

Mệnh đề 1.3.2. Tổng vô hạn $\sum_{j \geq 0} F_j(x)$ hội tụ khi và chỉ khi $\lim_{j \rightarrow \infty} \deg F_j(x) = \infty$.

Mệnh đề 1.3.3. Tích vô hạn $\prod_{j \geq 1} (1 + F_j(x))$, với $F_j(0) = 1$ hội tụ khi và chỉ khi $\lim_{j \rightarrow \infty} \deg F_j(x) = \infty$.

Để kết thúc phần này chúng tôi trình bày một ví dụ về hàm sinh.

Ví dụ 1.3.4. Tìm công thức tường minh cho các số Fibonacci: $a_0 = a_1 = 1, a_n = a_{n-1} + a_{n-2}$ với $n \geq 2$.

Ta sẽ tìm hàm sinh cho dãy số Fibonacci $\{a_n\}_{n \geq 0}$. Ta có

$$\begin{aligned} F(x) &= \sum_{n \geq 0} a_n x^n = 1 + x + \sum_{n \geq 2} a_n x^n = 1 + x + \sum_{n \geq 2} (a_{n-2} + a_{n-1}) x^n \\ &= 1 + x + x \sum_{n \geq 2} a_{n-1} x^{n-1} + x^2 \sum_{n \geq 2} a_{n-2} x^{n-2} = 1 + x + x(F(x) - 1) + x^2 F(x). \end{aligned}$$

Giải phương trình này ta được $F(x) = \frac{x}{1-x-x^2}$. Khai triển Taylor của hàm $F(x)$ ta có

$$F(x) = \sum_{n \geq 0} \frac{1}{\sqrt{5}} \left(\left(\frac{1+\sqrt{5}}{2} \right)^n - \left(\frac{1-\sqrt{5}}{2} \right)^n \right) x^n$$

Sử dụng đồng nhất thức ta có công thức tường minh cho a_n :

$$a_n = \frac{1}{\sqrt{5}} \left(\left(\frac{1+\sqrt{5}}{2} \right)^n - \left(\frac{1-\sqrt{5}}{2} \right)^n \right).$$

1.4 Hệ động lực rời rạc

Trong phần này chúng tôi trình bày một số khái niệm và một số bài toán của hệ động lực rời rạc. Trước hết chúng tôi có định nghĩa tổng quát về hệ động lực rời rạc.

Định nghĩa 1.4.1. Hệ động lực rời rạc (discrete dynamical system) \mathcal{S} là một hệ gồm không gian trạng thái \mathcal{M} và tập các luật vận động \mathcal{R} . Các luật vận động cho phép ta đạt được các trạng thái mới từ các trạng thái trước đó. Một cách hình thức, hệ động lực rời rạc là một bộ ba $\mathcal{S} = (\mathcal{M}, \mathcal{R}, \mathbb{N})$, trong đó:

+ không gian trạng thái của hệ \mathcal{M} là một tập hợp,

+ \mathcal{R} là tập các hàm $\phi : \mathbb{N} \times \mathcal{M} \rightarrow 2^{\mathcal{M}}$ thỏa mãn $\phi(t_2, \phi(t_1, x)) = \phi(t_1 + t_2, x)$

với mọi $t_1, t_2 \in \mathbb{N}$ và với mọi $x \in \mathcal{M}$.

Đồ thị đạt được (reachability graph) của một hệ động lực rời rạc là một đồ thị có hướng xác định như sau: mỗi đỉnh của đồ thị biểu diễn một trạng thái (configuration) của hệ. Một cung có đỉnh đầu biểu diễn trạng thái a và đỉnh cuối biểu diễn trạng thái b nếu tồn tại $\phi \in \mathcal{R}$ sao cho $b \in \phi(1, a)$. Trong trường hợp này

ta nói b đạt được (hay nhận được) trực tiếp từ a , tức là trạng thái b nhận được từ trạng thái a bằng cách áp dụng luật vận động một lần. Ta cũng nói a là trạng thái kế trước (predecessor) trạng thái b hay a phủ (cover) b và b được gọi là một trạng thái kế sau (successor) của a , ký hiệu là $a \succ b$ hay $b \prec a$. Trạng thái b được gọi là đạt được (hay nhận được) (reachable) từ trạng thái a nếu b nhận được từ a bằng cách áp dụng một dãy các luật vận động.

Một trạng thái đạt được (từ trạng thái ban đầu) mà tại đó không thể áp dụng luật vận động gọi là điểm dừng (fixed point) của hệ. Hệ động lực được gọi là hội tụ nếu có đúng một điểm dừng.

Sau đây chúng ta sẽ phát biểu một số bài toán đạt được (reachability problem) của hệ động lực rời rạc:

+ **Bài toán t đạt được (t -reachability)**: Cho hệ động lực rời rạc S ; a, b là hai trạng thái của hệ, t là một số nguyên dương. Hãy xác định xem bắt đầu từ trạng thái a , hệ S có đạt được trạng thái b sau không quá t bước hay không.

+ **Bài toán đạt được (reachability)**: Cho hệ động lực rời rạc S ; a, b là hai trạng thái của hệ. Hãy xác định xem bắt đầu từ trạng thái a , hệ S có đạt được trạng thái b hay không?

+ **Bài toán xác định tính dừng (fixed point reachability)**: Cho hệ động lực rời rạc S và \mathcal{O} là một trạng thái của hệ. Hãy xác định xem bắt đầu từ trạng thái \mathcal{O} , hệ S có dừng hay không? Nếu hệ dừng thì hãy xác định điểm dừng.

Trong luận án này chúng tôi quan tâm đến bài toán xác định tính dừng khi nghiên cứu các phân hoạch của số tự nhiên trong chương 2, bài toán đạt được của hệ động lực CCFG trong chương 4, và cấu trúc không gian (hay đồ thị đạt được) của các hệ CFG và các mở rộng của nó trong chương 3 và 4.

Chương 2

Mô hình cột cát và phân hoạch của số tự nhiên

Trong chương này, chúng tôi nghiên cứu mối liên hệ giữa các mô hình cột cát mở rộng và phân hoạch của số tự nhiên. Hệ cột cát (Sand Piles Model - SPM) là một hệ động lực quan trọng được đề xuất bởi ba nhà Vật lý Bak, Tang và Wiesenfeld vào năm 1987 [7] để mô hình hóa hiện tượng đột biến tự tổ chức (Self-Organized Criticality - SOC). Hệ SPM này đã được chứng minh là một trường hợp đặc biệt của hệ Chip Firing Game (CFG) [30] (các kết quả về hệ CFG và các mở rộng của nó sẽ được trình bày trong các Chương 3 và 4 của luận án). Theo các nghiên cứu [20], [21], [27], [28], [30], [31], [33], [34], ... mô hình cột cát có liên quan chặt chẽ với phân hoạch của số tự nhiên. Trong chương này, chúng tôi sẽ xét đến các mô hình cột cát với ngưỡng d cho luật vận động và mối liên hệ của chúng với các phân hoạch d -chặt của số tự nhiên. Phương pháp chính được sử dụng ở đây là phương pháp ECO (Enumeration of Combinatorial Objects), một phương pháp tính toán tổ hợp sử dụng cây sinh và được phát triển trong những năm gần đây [9], [10]. Phương pháp này cho phép chúng tôi chứng minh cấu trúc của không gian trạng thái và tính toán số các trạng thái của mô hình. Bên cạnh đó, nhờ có phương pháp này chúng tôi cũng nghiên cứu được cấu trúc đệ quy của tập các phân hoạch d -chặt và đưa ra chứng minh cho một số đẳng thức tổ hợp.

2.1 Phân hoạch số tự nhiên và hệ động lực rời rạc

Mục đích chính của phần này là nghiên cứu mối quan hệ giữa hệ SPM mở rộng và tập $d\text{-}\mathcal{P}(n)$ các phân hoạch d -chặt ($d \geq 0$) của số tự nhiên n . Chúng tôi sẽ chứng minh rằng $d\text{-}\mathcal{P}(n)$ với quan hệ thứ tự thống trị (dominance order) chính là không gian trạng thái của một hệ động lực rời rạc. Một số kết quả trong mục này là mở rộng các kết quả của Latapy và Phan trong trường hợp phân hoạch thường ($d = 0$) [57] và các kết quả của Le, Phan trong trường hợp phân hoạch chặt ($d = 1$) [54].

2.1.1 Các định nghĩa và ký hiệu

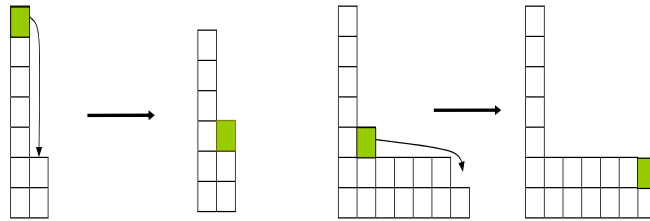
Trong phần này chúng tôi nhắc lại một số khái niệm về phân hoạch của số tự nhiên và hệ động lực rời rạc.

Định nghĩa 2.1.1. Phân hoạch a là một dãy các số nguyên dương không tăng (a_1, a_2, \dots, a_l) với $a_1 \geq a_2 \geq \dots \geq a_l > 0$, (quy ước $a_i = 0, \forall i \geq l + 1$). Các a_i gọi là các phần của phân hoạch a . Ta nói rằng a là một phân hoạch của n (hay a có trọng số n), ký hiệu là $a \vdash n$ hay $|a| = n$, nếu $\sum_{i=1}^l a_i = n$. Ta ký hiệu $\mathcal{P}(n) = \{a \vdash n\}$ là tập tất cả các phân hoạch của số nguyên dương n . Tập hợp $\mathcal{P}(n)$ được trang bị một thứ tự bộ phận, gọi là *thứ tự thống trị (dominance ordering)*, xác định như sau: với hai phân hoạch a và b của n , ta có $a \geq b$ khi và chỉ khi $\sum_{i>j} a_i \leq \sum_{i>j} b_i$ với mọi $j \geq 0$ [6]. Ta sử dụng định nghĩa này cho cả trường hợp a và b là phân hoạch của các số nguyên dương m và n với $m \geq n$ [57].

Phân hoạch của số tự nhiên liên hệ chặt chẽ với một hệ động lực nổi tiếng trong Vật lý, đó là mô hình cột cát tuần tự. Mô hình cột cát tuần tự (sequential Sand Piles Model (SPM)) được các nhà vật lý Bak, Tang và Wiesenfeld đưa ra vào năm 1987 để mô phỏng hiện tượng tự đột biến (SOC - Self organized criticality) [7]. Một trạng thái của hệ SPM là vô hạn các cột cát có thứ tự, trong đó chỉ có k cột chứa một số hạt cát (grains). Do đó mỗi trạng thái của hệ này được mô tả bằng

một bộ $s = (s_1, \dots, s_k)$, ở đây s_i là số hạt ở cột thứ i . Luật vận động (*luật rơi*) của hệ này như sau: một hạt có thể rơi từ cột i xuống cột $i + 1$ nếu $s_i - s_{i+1} \geq 2$. Luật này xác định một quan hệ phủ (covering relation) trên tập trạng thái đạt được. Bao đóng phản xạ và bắc cầu của quan hệ này là một thứ tự \leq_{SPM} và nó tuân theo thứ tự thống trị (dominance order) [33]. Trạng thái ban đầu là (n) , tức là n hạt ở cột đầu tiên, các cột khác không chứa hạt nào cả. Tập hợp tất cả các trạng thái đạt được từ trạng thái đầu (n) với thứ tự \leq_{SPM} là một dàn, ký hiệu là $SPM(n)$ [33]. Bằng cách mở rộng thêm luật vận động (*luật rơi* và *luật trượt*), thì mô hình SPM trở thành mô hình Brylawsky. Brylawsky (1973) đã chứng minh tập tất cả các phân hoạch của n với thứ tự thống trị là một dàn, ký hiệu là $L_B(n)$ [6], với cận trên bé nhất của hai phân hoạch p và q được xác định như sau:

$$\sup(p, q) = r \iff \forall j : \sum_{i=1}^j r_i = \min \left(\sum_{i=1}^j p_i, \sum_{i=1}^j q_i \right)$$



Hình 2.1: Luật rơi (V) và luật trượt (H) trong hệ Brylawsky.

Chúng tôi nghiên cứu các phân hoạch của số tự nhiên theo hướng mở rộng hai luật (luật rơi và luật trượt) bằng cách thêm ngưỡng cho các luật vận động. Chúng tôi nhận được các kết quả mở rộng của Brylawsky về cấu trúc dàn của tập $d\text{-}\mathcal{P}(n)$ các phân hoạch d -chặt của số tự nhiên n . Trước hết, chúng tôi trình bày một số định nghĩa về *phân hoạch chặt*, *phân hoạch d -chặt* của số tự nhiên và các luật vận động.

Định nghĩa 2.1.2. Phân hoạch $a = (a_1, a_2, \dots, a_l)$ gọi là *phân hoạch chặt* nếu các phần của a là các số nguyên dương khác nhau, tức là $a_1 > a_2 > \dots > a_l > 0$. Tập

hợp tất cả các phân hoạch chặt của số tự nhiên n được ký hiệu là $\mathcal{SP}(n)$.

Định nghĩa 2.1.3. Cho d là một số tự nhiên. Một phân hoạch d -chặt (d -strict partition) $a = (a_1, \dots, a_l)$ là một phân hoạch thỏa mãn $a_i - a_{i+1} \geq d$, với mọi $1 \leq i \leq l - 1$. Riêng với a_l có thể nhận giá trị nhỏ hơn d . Tập hợp tất cả các phân hoạch d -chặt của n ký hiệu là $d\text{-}\mathcal{P}(n)$.

Trong trường hợp $d = 0$, $0\text{-}\mathcal{P}(n)$ chính là tập hợp $\mathcal{P}(n)$ tất cả các phân hoạch của n , và trong trường hợp $d = 1$, ta có $1\text{-}\mathcal{P}(n)$ chính là tập hợp $\mathcal{SP}(n)$ các phân hoạch chặt của n . Ký hiệu $d\text{-}\mathcal{P} = \bigcup_{n \geq 0} d\text{-}\mathcal{P}(n)$ là tập tất cả các phân hoạch d -chặt.

Sau đây, ta xét hệ động lực rời rạc có không gian trạng thái là $d\text{-}\mathcal{P}(n)$, trong đó các luật vận động của nó được xác định như sau:

Định nghĩa 2.1.4. Cho a là một phân hoạch d -chặt của n , ta áp dụng các luật vận động lên a để nhận được một phân hoạch d -chặt khác của n như sau: (Xem hình 2.2)

- Luật dọc (luật V):

$$(a_1, \dots, a_i, a_{i+1}, \dots, a_l) \rightarrow (a_1, \dots, a_i - 1, a_{i+1} + 1, \dots, a_l) \text{ nếu } a_i - a_{i+1} \geq d + 2$$

- Luật ngang (luật H) với độ dài $k \geq 2$:

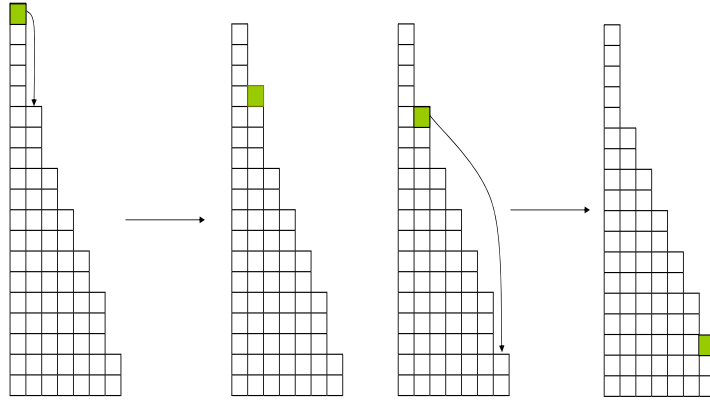
$$(a_1, \dots, p + k + 1, p + k - d, p + k - 2d, \dots, p + k - (k - 1)d, p + k - kd - 1, \dots, a_l) \rightarrow (a_1, \dots, p + l, p + k - d, p + k - 2d, \dots, p + k - (k - 1)d, p + k - kd, \dots, a_l)$$

và luật ngang với độ dài 1 : $(a_1, \dots, p + 2, p - d, \dots, a_l) \rightarrow (a_1, \dots, p + 1, p - d + 1, \dots, a_l)$.

Chú ý rằng luật ngang độ dài 1 cũng chính là một trường hợp của luật dọc.

2.1.2 Cấu trúc của $d\text{-}\mathcal{P}(n)$

Mục đích của phần này là nghiên cứu cấu trúc thứ tự của tập $d\text{-}\mathcal{P}(n)$. Ký hiệu $(n) = (n, 0, \dots, 0)$. Trước hết, chúng tôi chứng minh rằng mọi phân hoạch d -chặt



Hình 2.2: Luật dọc (V) và luật ngang (H) trong trường hợp $d = 2$.

của n đều đạt được từ trạng thái ban đầu (n) bằng cách áp dụng các luật vận động (dọc và ngang). Chúng tôi sử dụng kỹ thuật chứng minh trong [54].

Bổ đề 2.1.5. *Tập hợp $d\text{-}\mathcal{P}(n)$ chính là tập các phân hoạch d -chặt đạt được từ trạng thái đầu (n) bằng cách áp dụng hai luật vận động V và H .*

Chứng minh. Cho $a = (a_1, \dots, a_m)$ là một phân hoạch d -chặt của n và $a \neq (n)$. Ta chứng minh tồn tại một con đường từ (n) đến a . Bằng phương pháp truy hồi, ta chỉ cần chứng minh rằng tồn tại phân hoạch d -chặt a' của n và a' phủ a . Ta xét hai trường hợp:

+ Trường hợp 1: nếu tồn tại dãy a_i, a_{i+1}, \dots, a_j thỏa mãn đồng thời các điều kiện:

- $a_k - a_{k+1} \geq d$, với mọi $k = i, i+1, \dots, j-1$, nếu $i \neq j$; trong trường hợp $i = j$ thì a_i có thể nhỏ hơn d .
- $i = 1$, hoặc $a_{i-1} - a_i \geq d+1$,
- $j = m$ hoặc là $a_j - a_{j+1} \geq d+1$,

thì ta chọn $a' = (a_1, \dots, a_{i-1}, a_i + 1, a_{i+1}, \dots, a_{j-1}, a_j - 1, a_{j+1}, \dots, a_m)$. Khi đó, rõ ràng $a' \in d\text{-}\mathcal{P}(n)$ và ta có thể nhận được a từ a' bằng cách áp dụng luật H .

+ Trường hợp 2: nếu không tồn tại dãy số trong a như trường hợp 1 thì a thỏa mãn một trong các trường hợp sau:

- $a_1 - a_2 \geq d + 1$ và $m = 2$,
- hoặc là $a_2 - a_3 \geq d + 1$.

Khi đó ta chọn $a' = (a_1 + 1, a_2 - 1, a_3, \dots, a_m)$. Dễ dàng kiểm tra $a' \in d\text{-}\mathcal{P}(n)$ và ta có thể áp dụng luật V ở vị trí đầu tiên của a' để nhận được a . Mệnh đề được chứng minh. □

Chúng ta đã biết rằng tập hợp $\mathcal{P}(n)$ tất cả các phân hoạch của n với quan hệ thứ tự thống trị là một dàn [6], ký hiệu là $L_B(n)$, trong đó

$$\inf(a, b) = c \text{ khi và chỉ khi với mọi } j \geq 1, \text{ ta có } \sum_{i=1}^j c_i = \min\left(\sum_{i=1}^j a_i, \sum_{i=1}^j b_i\right).$$

Nhắc lại rằng $0\text{-}\mathcal{P}(n)$ chính là $\mathcal{P}(n)$ và cũng là $L_B(n)$ khi nhấn mạnh đến cấu trúc dàn. Bằng cách chứng minh tương tự như trong [54], ta có thể chứng minh rằng tập $d\text{-}\mathcal{P}(n)$ ($d \geq 1$) là một tập thứ tự con (subset) của $\mathcal{P}(n)$ và cũng là một dàn. Do đó ta có thể viết $b \leq a$ thay cho $b \leq_d a$ với a, b là hai phân hoạch d -chặt của n . Ta có kết quả sau:

Định lý 2.1.6. *Tập $d\text{-}\mathcal{P}(n)$ cùng với hai phép toán \vee và \wedge là một dàn. Hơn nữa, cận dưới lớn nhất của hai phân tử trong $d\text{-}\mathcal{P}(n)$ được xác định như trong $\mathcal{P}(n)$.*

Chú ý 2.1.7. *$d\text{-}\mathcal{P}(n)$ không phải là dàn con của $\mathcal{P}(n)$. Thật vậy, cận trên bé nhất của hai phân tử trong $d\text{-}\mathcal{P}(n)$ khác với cận trên bé nhất của hai phân tử trong $\mathcal{P}(n)$. Chẳng hạn, $(8, 4, 3, 1) \vee (7, 5, 4) = (8, 4, 4)$ không là một phân hoạch chặt. Tuy nhiên, ta vẫn có $a \vee_d b \geq a \vee b$ với mọi phân hoạch d -chặt a, b .*

Khi nghiên cứu các hệ động lực, một câu hỏi tự nhiên được đặt ra là: khi nào thì hệ hội tụ? tức là có duy nhất một điểm dừng. Nếu hệ hội tụ thì hãy xác định điểm dừng duy nhất đó. Trong trường hợp $d\text{-}\mathcal{P}(n)$, vì hệ này có cấu trúc dần nên có duy nhất điểm dừng. Chúng ta sẽ đưa ra công thức tường minh của điểm dừng duy nhất này. Thật vậy, gọi p là số tự nhiên duy nhất thỏa mãn

$$\frac{1}{2}p((p-1)d+2) \leq n < \frac{1}{2}(p+1)(pd+2).$$

Khi đó, ta có duy nhất biểu diễn $n - \frac{1}{2}p(2 + (p-1)d) = qp + r$, trong đó $r < p$. Gọi $\Pi(d)$ là phân hoạch d -chặt được xác định như sau:

$$\begin{aligned} \Pi(d) = & (1 + (p-1)d + q + 1, 1 + (p-2)d + q + 1, \dots, \\ & 1 + (p-r)d + q + 1, 1 + (p-r-1)d + q, \dots, 1 + d + q, 1 + q) \end{aligned}$$

Ta dễ dàng kiểm tra được $\Pi(d)$ là điểm dừng của hệ và ta có mệnh đề sau:

Mệnh đề 2.1.8. $\Pi(d)$ là điểm dừng của hệ $d\text{-}\mathcal{P}(n)$.

2.1.3 Mối quan hệ giữa $d\text{-}\mathcal{P}(n+1)$ và $d\text{-}\mathcal{P}(n)$

Trong phần này chúng tôi nghiên cứu mối quan hệ giữa $d\text{-}\mathcal{P}(n+1)$ và $d\text{-}\mathcal{P}(n)$. Trước hết, chúng tôi trình bày một số khái niệm cơ bản và các ký hiệu.

Cho $a = (a_1, a_2, \dots, a_m)$ là một phân hoạch d -chặt. Khi đó, phân hoạch nhận được từ a bằng cách thêm một vào thành phần thứ j được ký hiệu là $a^{\downarrow j}$. Để ý rằng $a^{\downarrow j}$ không nhất thiết là một phân hoạch d -chặt. Ta có một số khái niệm và ký hiệu sau:

+ Cho S là tập các phân hoạch d -chặt. Ký hiệu $S^{\downarrow j} := \{a^{\downarrow j} | a \in S\}$.

+ $Succ(a)$ là tập các phân hoạch d -chặt nhận được trực tiếp từ phân hoạch d -chặt a .

+ $d_i(a) = a_i - a_{i+1}$ với quy ước $a_{m+1} = 0$

+ Ta nói rằng a có một *bước nhảy (cliff)* tại vị trí i nếu $d_i(a) \geq d + 2$.

+ Nếu tồn tại $\ell \geq i$ sao cho $d_j(a) = d$ với mọi $i \leq j < \ell$ và $d_\ell(a) = d + 1$ hoặc $\ell = m$, thì ta nói a có một *cầu thang (slippery plateau)* tại vị trí i . Tương tự, ta nói rằng a có *cầu thang nhẩy (non-slippery plateau)* tại vị trí i nếu $d_j(a) = d$ với mọi $i \leq j < \ell$ và $d_\ell(a) \geq d + 2$. Số tự nhiên $\ell - i + 1$ được gọi là *độ dài* của cầu thang tại i . Đặc biệt, khi $\ell = i$, cầu thang có độ dài 1.

+ Tập các phân tử của $d\text{-}\mathcal{P}(n)$ bắt đầu bằng một bước nhẩy, cầu thang có độ dài ℓ hay có cầu thang nhẩy có độ dài ℓ được ký hiệu tương ứng là $C, d\text{-}P_\ell, nd\text{-}P_\ell$.

Chú ý rằng mỗi phân tử của $d\text{-}\mathcal{P}(n)$ đều thuộc vào một trong ba tập hợp này.

Để thấy rằng $a^{\downarrow 1}$ cũng là một phân hoạch d -chặt. Do đó, ta có thể định nghĩa phép nhúng $\pi : d\text{-}\mathcal{P}(n) \rightarrow d\text{-}\mathcal{P}(n)^{\downarrow 1} \subseteq d\text{-}\mathcal{P}(n+1)$ và bằng cách sử dụng công thức của phân tử infimum, ta dễ dàng chứng minh được mệnh đề sau:

Bổ đề 2.1.9. *Tập $d\text{-}\mathcal{P}(n)^{\downarrow 1}$ là một dàn con của $d\text{-}\mathcal{P}(n+1)$.*

Tiếp theo, chúng tôi sẽ đặc trưng các phân tử còn lại của $d\text{-}\mathcal{P}(n+1)$ mà không thuộc vào $d\text{-}\mathcal{P}(n)^{\downarrow 1}$.

Định lý 2.1.10. *Tập $d\text{-}\mathcal{P}(n+1)$ là hợp rời của các tập $d\text{-}\mathcal{P}(n)^{\downarrow i}$, tức là*

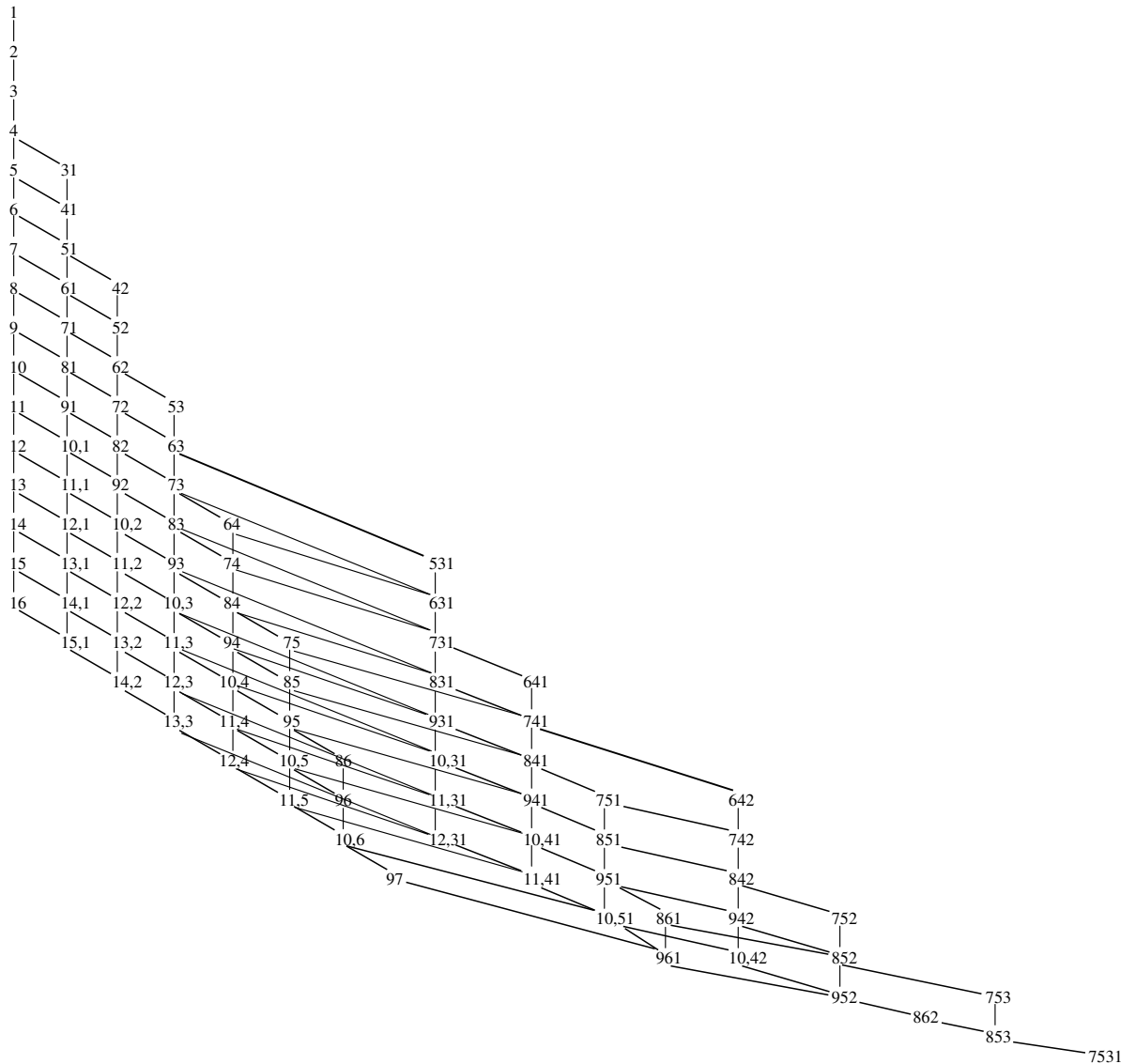
$$d\text{-}\mathcal{P}(n+1) = d\text{-}\mathcal{P}(n)^{\downarrow 1} \sqcup_{\ell \geq 1} d\text{-}P_\ell^{\downarrow \ell+1}(n).$$

Chứng minh. Để dàng chứng minh rằng mỗi phân tử của tập hợp $d\text{-}\mathcal{P}(n)^{\downarrow 1}$ và $d\text{-}P_\ell^{\downarrow \ell+1}$ là một phân tử của $d\text{-}\mathcal{P}(n+1)$. Hơn nữa, các tập hợp này là rời nhau. Như vậy, tập hợp ở vế phải là một tập con của tập hợp ở vế trái.

Tiếp theo ta sẽ chứng minh tập hợp ở vế trái là một tập con của tập hợp ở vế phải. Xét $b \in d\text{-}\mathcal{P}(n+1)$ tùy ý.

+ Nếu $b_1 - b_2 \geq d + 1$ thì $b \in d\text{-}\mathcal{P}(n)^{\downarrow 1}$.

+ Nếu b bắt đầu bằng một cầu thang hoặc cầu thang nhẩy có độ dài $\ell + 1, \ell \geq 1$, thì $b \in d\text{-}P_\ell^{\downarrow \ell+1}(n)$. □



Hình 2.3: Các phần tử đầu tiên của dãy vô hạn $2-\mathcal{P}(\infty)$.

2.1.4 Dãy vô hạn $d-\mathcal{P}(\infty)$

Trong các phần trước, chúng ta đã nghiên cứu cấu trúc của $d-\mathcal{P}(n)$, mở rộng từ $d-\mathcal{P}(n)$ lên $d-\mathcal{P}(n+1)$ và quan hệ cấu trúc giữa các tập này. Trong mục này, chúng ta tiếp tục nghiên cứu cấu trúc dãy của mở rộng vô hạn của $d-\mathcal{P}(n)$. Ta xem $d-\mathcal{P}(n)$ là hệ động lực rời rạc với không gian trạng thái là các phân hoạch d -chặt của n , với trạng thái đầu là (n) và hai luật vận động V và H. Mở rộng vô hạn của $d-\mathcal{P}(n)$ chính là $d-\mathcal{P}(\infty)$, là hệ động lực rời rạc với không gian trạng thái là tập tất cả các

phân hoạch d -chặt (sai khác đẳng cấu), trạng thái ban đầu là (∞) , với cột đầu tiên có vô hạn hạt và các cột còn lại không chứa hạt nào. Chúng ta ký hiệu $d\text{-}\mathcal{P}(\infty)$ là không gian trạng thái của hệ động lực này. Mỗi phần tử của $d\text{-}\mathcal{P}(\infty)$ sẽ có dạng $(\infty, a_2, a_3, \dots, a_k)$. Quan hệ thứ tự bộ phận giữa các phần tử trong $d\text{-}\mathcal{P}(\infty)$ được định nghĩa như sau:

$$a \geq_{\infty} b \text{ nếu } \sum_{i \geq j} a_i \leq \sum_{i \geq j} b_i \text{ với mọi } j \geq 2.$$

Như vậy, quan hệ thứ tự thống trị tương đương với quan hệ thứ tự cảm sinh bởi các luật vận động V và H của hệ động lực này. Hình vẽ 2.3 biểu diễn một số phân hoạch đầu tiên của $2\text{-}\mathcal{P}(\infty)$, trong đó thành phần đầu tiên ∞ không được trình bày ở đây. Chú ý là trên các hình vẽ của luận án này ta ký hiệu $a_1 a_2 \dots a_k$ thay cho (a_1, a_2, \dots, a_k) , để chỉ các phân hoạch của n .

Cho hai phần tử bất kỳ $a = (\infty, a_2, \dots, a_k), b = (\infty, b_2, \dots, b_{\ell})$ của tập $d\text{-}\mathcal{P}(\infty)$, ta xác định phần tử c như sau: $c_i = \max(\sum_{j \geq i} a_j, \sum_{j \geq i} b_j) - \sum_{j > i} c_j$ với mọi i sao cho $2 \leq i \leq \max(k, \ell)$ và $c_i = 0$ nếu $i > \max(k, \ell)$.

Khi đó, ta dễ dàng kiểm tra được $c \in d\text{-}\mathcal{P}(\infty)$, tức là $c_1 = \infty, c_i - c_{i+1} \geq d$ với mọi $i > 1$, và hơn thế $c = a \wedge b$. Từ đó ta có kết quả sau:

Định lý 2.1.11. *Tập $d\text{-}\mathcal{P}(\infty)$ cùng với hai phép toán \vee và \wedge là một dàn.*

Trong hình vẽ 2.3, ta có thể nhìn thấy các phần tử của $2\text{-}\mathcal{P}(\infty)$ đẳng cấu với $2\text{-}\mathcal{P}(n)$. Bây giờ ta sẽ xác định các đẳng cấu này. Ta định nghĩa hai phép nhúng chính tắc như sau:

$$\pi : \begin{array}{ccc} d\text{-}\mathcal{P}(n) & \longrightarrow & d\text{-}\mathcal{P}(\infty) \\ a = (a_1, a_2, \dots, a_k) & \mapsto & \pi(a) = (\infty, a_2, \dots, a_k) \end{array}$$

$$\chi : \begin{array}{ccc} d\text{-}\mathcal{P}(n) & \longrightarrow & d\text{-}\mathcal{P}(\infty) \\ a = (a_1, a_2, \dots, a_k) & \mapsto & \chi(a) = (\infty, a_1, a_2, \dots, a_k) \end{array}$$

Từ định nghĩa ta có ngay mệnh đề sau:

Mệnh đề 2.1.12. Các ánh xạ π và χ là các phép nhúng dần.

Sử dụng phép nhúng χ , ta có thể xem $d\text{-}\mathcal{P}(\infty)$ như là hợp rời của các $d\text{-}\mathcal{P}(n)$ với $n \in \mathbb{N}$:

$$d\text{-}\mathcal{P}(\infty) = \bigsqcup_{n \geq 0} d\text{-}\mathcal{P}(n).$$

2.1.5 Cây vô hạn $T_{d\text{-}\mathcal{P}(\infty)}$

Mục đích của phần này là nghiên cứu cấu trúc cây vô hạn $T_{d\text{-}\mathcal{P}(\infty)}$ sinh ra tất cả các phân hoạch d -chặt dựa vào cấu trúc của dàn vô hạn $d\text{-}\mathcal{P}(\infty)$. Cây $T_{d\text{-}\mathcal{P}(\infty)}$ chính là cây bao trùm của dàn $d\text{-}\mathcal{P}(\infty)$. Chúng tôi sẽ đưa ra một số tính chất của cây này. Từ Định lý 2.1.10, ta định nghĩa cây $T_{d\text{-}\mathcal{P}(\infty)}$ như sau:

- + $T_{d\text{-}\mathcal{P}(\infty)}$ là cây nhị phân vô hạn có gốc tại (0) ;
- + Mỗi nút $a \in T_{d\text{-}\mathcal{P}(\infty)}$ luôn có một con trái là $a^{\downarrow 1}$;
- + Nếu a bắt đầu bằng một cầu thang có độ dài $\ell + 1$, thì a có một con phải là $a^{\downarrow \ell + 1}$.

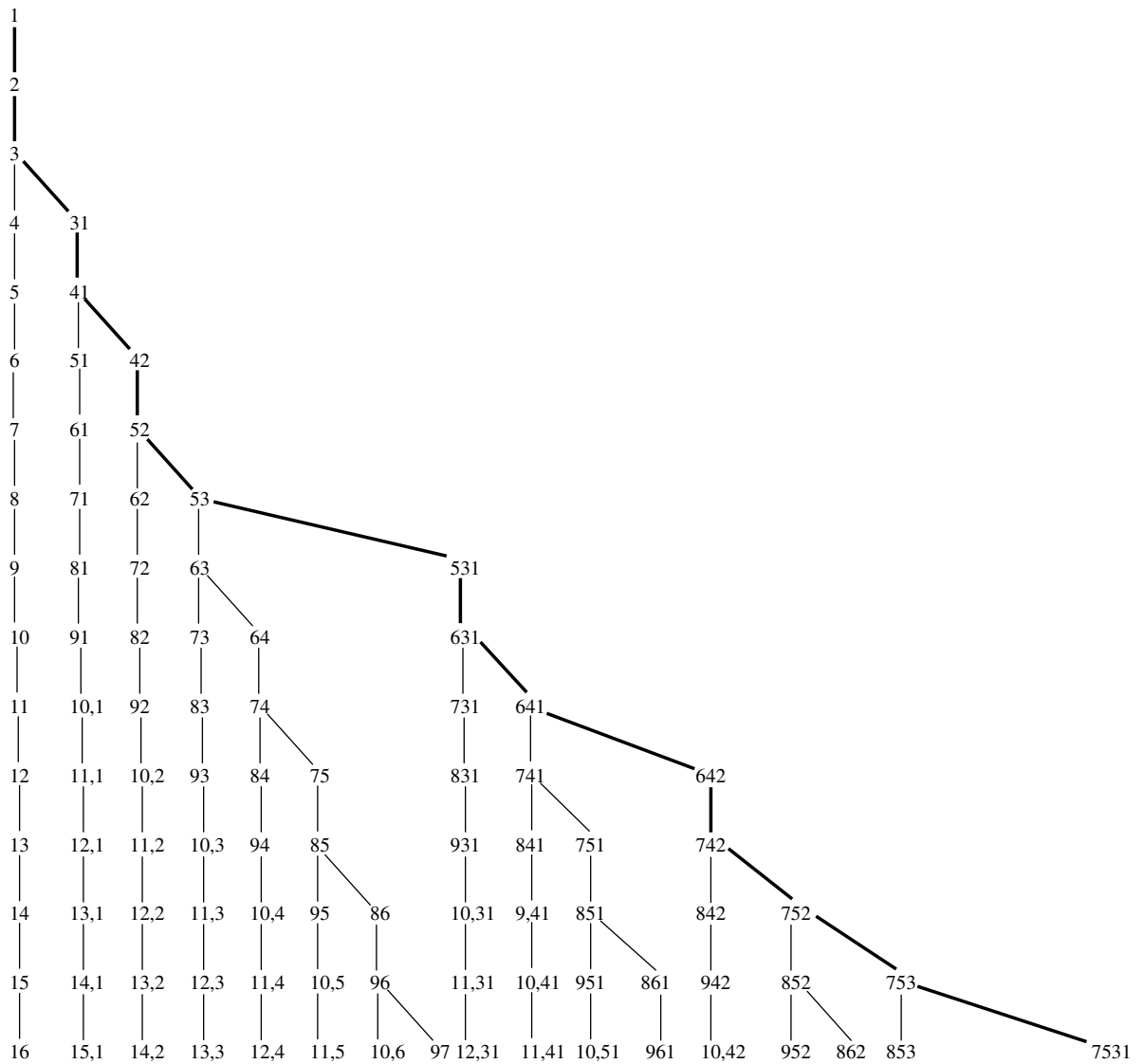
Hình vẽ 2.4 minh họa phần đầu của cây $T_{2\text{-}\mathcal{P}(\infty)}$.

Từ mệnh đề 2.1.12, ta có tính chất của cây $T_{d\text{-}\mathcal{P}(\infty)}$.

Mệnh đề 2.1.13. Tầng thứ n của cây $T_{d\text{-}\mathcal{P}(\infty)}$ là tập hợp tất cả các phân tử của $d\text{-}\mathcal{P}(n)$.

2.2 Phương pháp ECO và phân hoạch số tự nhiên

Phân hoạch của số tự nhiên, đặc biệt là bài toán đếm các dạng phân hoạch đã được nghiên cứu nhiều và có nhiều kết quả thú vị. Một số kết quả đã được chúng tôi trình bày trong mục 3.1 theo quan điểm của hệ động lực rời rạc. Trong mục này, chúng tôi nghiên cứu phân hoạch của số tự nhiên theo một hướng tiếp cận khác,



Hình 2.4: Cây các phân hoạch 2-chật

đó là sử dụng phương pháp ECO và chúng tôi cũng đạt được những kết quả tương tự. Theo cách tiếp cận này, chúng tôi định nghĩa toán tử ECO cho các phân hoạch d -chật và đưa ra cấu trúc của cây sinh vô hạn T_{d-P} , sinh ra đúng tất cả các phân hoạch d -chật.

2.2.1 Phương pháp ECO

Phương pháp ECO (Enumeration Combinatorial Objects) đã được giới thiệu đầu tiên bởi các nhà toán học Barucci, Del Lungo, Pergola, và Pinzani trong hai bài báo

chính [9, 10]. Phương pháp này chủ yếu dùng để giải bài toán đếm các cấu hình tổ hợp. Để giải bài toán đếm các cấu hình tổ hợp bằng phương pháp ECO, trước hết ta đưa ra định nghĩa toán tử ECO, sau đó xây dựng cây sinh tương ứng cho toán tử ECO này và rút ra hàm sinh tương ứng với cây sinh. Bài toán đếm trở nên rõ ràng hơn khi ta xây dựng được cấu trúc của cây sinh. Từ đó có thể đếm các cấu hình tổ hợp với các tính chất khác nhau. Sau đây, chúng tôi trình bày các định nghĩa cơ bản.

Định nghĩa 2.2.1. Cho \mathcal{O} là một lớp các đối tượng tổ hợp, $p : \mathcal{O} \rightarrow \mathbb{N}$ sao cho $O_n = \{x \in \mathcal{O} : p(x) = n\}$ thỏa mãn $|O_n|$ hữu hạn. Toán tử ECO trên lớp các đối tượng tổ hợp \mathcal{O} là một toán tử $\vartheta : O_n \rightarrow 2^{O_{n+1}}$ thỏa mãn hai điều kiện sau:

1. Với mỗi $Y \in O_{n+1}$, tồn tại $X \in O_n$, sao cho $Y \in \vartheta(X)$,
2. Với mọi $X_1, X_2 \in O_n$ mà $X_1 \neq X_2$, thì $\vartheta(X_1) \cap \vartheta(X_2) = \emptyset$.

Như vậy, với mỗi $Y \in O_{n+1}$ tồn tại duy nhất $X \in O_n$ sao cho $Y \in \vartheta(X)$. Điều này chứng tỏ họ các tập hợp $\{\vartheta(X) : X \in O_n\}$ là một phân hoạch của tập O_{n+1} . Do đó, ta có thể mô tả đệ quy các phần tử của \mathcal{O} . Từ đó có thể xây dựng hàm sinh cho các phần tử của \mathcal{O} với các tham số khác nhau.

Cho toán tử ECO ϑ . Khi đó, ta có thể xây dựng *cây sinh (generating tree)* cho toán tử này như sau:

- + Góc của cây đặt ở tầng 0, tương ứng với các đối tượng có kích thước nhỏ nhất (ở đây, các đối tượng thuộc tập hợp O_n gọi là có kích thước n).
- + Các đối tượng có cùng kích thước thì nằm trên cùng một tầng.
- + Con của mỗi đối tượng $O \in \mathcal{O}$ được xác định thông qua toán tử ECO ϑ , cụ thể là, tập các con của $O \in \mathcal{O}$ là $\vartheta(O)$.

Cho $\{|O_n|\}_n$ là dãy các đối tượng tổ hợp. Khi đó hàm $f_{\mathcal{O}}(x) = \sum_{n \geq m} |O_n| x^n$ được gọi là hàm sinh của dãy này. Đôi khi, hàm này cũng được gọi là *hàm sinh của cây sinh tương ứng*. Trong nhiều cấu trúc tổ hợp, các phần tử được sắp xếp một

cách đệ quy nên toán tử ECO còn có thêm tính chất kế tiếp và cây sinh cũng thỏa mãn luật kế tiếp (succession rule) và có cấu trúc đệ quy khá đẹp đẽ. Toán tử ECO và cây sinh tương ứng mà chúng tôi định nghĩa cho các đối tượng tổ hợp, cụ thể là phân hoạch của số tự nhiên sau đây không thỏa mãn luật kế tiếp.

2.2.2 Phân hoạch d -chặt và phương pháp ECO

Trước hết, ta định nghĩa toán tử ECO cho phân hoạch d -chặt của các số tự nhiên.

Định nghĩa 2.2.2. Toán tử $\vartheta : d\text{-}\mathcal{P}(n) \rightarrow 2^{d\text{-}\mathcal{P}(n+1)}$ xác định như sau:

+ Với mỗi $a = (a_1, a_2, \dots, a_l) \in d\text{-}\mathcal{P}(n)$, $a^{\downarrow 1} = (a_1 + 1, a_2, \dots, a_l)$ là một phần tử của $\vartheta(a)$, phần tử này gọi là *con trái* của a .

+ Nếu a bắt đầu bằng một cầu thang có dạng: $p, p-d, \dots, p-id, p-(i+1)d-1$, thì $\vartheta(a)$ chứa thêm phần tử $a^{\downarrow i+2}$ (gọi là *con phải* của a) bắt đầu bằng dãy $p, p-d, \dots, p-id, p-(i+1)d$.

Cây sinh tương ứng T_{2-P} của toán tử này trùng với cây bao trùm của dàn vô hạn $2\text{-}\mathcal{P}(\infty)$ (hình 2.4) được trình bày trong các phần trước theo quan điểm của hệ động lực rời rạc. Bây giờ ta sẽ chứng minh rằng toán tử ϑ trong định nghĩa trên là toán tử ECO cho phân hoạch d -chặt của các số tự nhiên.

Bổ đề 2.2.3. Toán tử ϑ là một toán tử ECO, tức là ϑ thỏa mãn hai điều kiện sau:

(i) Với mỗi $b \in d\text{-}\mathcal{P}(n+1)$, tồn tại $a \in d\text{-}\mathcal{P}(n)$ sao cho $b \in \vartheta(a)$,

(ii) Với mọi $a, a' \in d\text{-}\mathcal{P}(n)$ mà $a \neq a'$, thì $\vartheta(a) \cap \vartheta(a') = \emptyset$.

Chứng minh. (i) Lấy tùy ý $b = (b_1, b_2, \dots, b_l) \in d\text{-}\mathcal{P}(n+1)$. Khi đó

+ Nếu $b_1 - b_2 \geq d + 1$ thì chọn $a = (b_1 - 1, b_2, \dots, b_l)$. Ta có ngay $a \in d\text{-}\mathcal{P}(n)$ và $a^{\downarrow 1} = b$. Do đó $b \in \vartheta(a)$.

+ Nếu $b_1 = b_2 + d$, ta viết b dưới dạng: $b = (b_1, b_1 - d, b_1 - 2d, \dots, b_1 - id, b_1 - (i+1)d, b_{i+3}, \dots, b_l)$, trong đó $b_1 - (i+2)d - 1 \geq b_{i+3} \geq 0$. Chọn

$a = (b_1, b_1 - d, b_1 - 2d, \dots, b_1 - id, b_1 - (i+1)d - 1, b_{i+3}, \dots, b_l)$. Ta có $a \in d\text{-}\mathcal{P}(n)$ và $b = a^{\downarrow i+2}$. Vậy $b \in \vartheta(a)$.

(ii) Giả sử $a, a' \in d\text{-}\mathcal{P}(n)$, và $a \neq a'$. Khi đó $a^{\downarrow 1} \neq a'^{\downarrow 1}$. Ta xét các trường hợp:

+ Nếu $a = (p, p - d, \dots, p - id, p - (i+1)d - 1, \dots)$ có con phải là $b = (p, p - d, \dots, p - id, p - (i+1)d, \dots)$ thì $b \neq a'^{\downarrow 1}$ vì nếu không thì a' không là phân hoạch d -chặt.

+ Tương tự, con phải (nếu có) của a' khác $a^{\downarrow 1}$.

+ Giả sử $a = (p, p - d, \dots, p - id, p - (i+1)d - 1, \dots)$ và $a' = (p', p' - d, \dots, p' - i'd, p' - (i'+1)d - 1, \dots)$ có các con phải tương ứng là $b = (p, p - d, \dots, p - id, p - (i+1)d, \dots)$ và $b' = (p', p' - d, \dots, p' - i'd, p' - (i'+1)d, \dots)$. Khi đó, nếu $b = b'$ thì $p = p', i = i'$ và do đó $a = a'$. Điều này mâu thuẫn với giả thiết là $a \neq a'$.

Vậy $\vartheta(a) \cap \vartheta(a') = \emptyset$. Mệnh đề được chứng minh. □

2.2.3 Cấu trúc đệ quy của cây vô hạn $T_{d\text{-}\mathcal{P}(\infty)}$

Trong phần này chúng tôi trình bày cấu trúc đệ quy của cây sinh $T_{d\text{-}\mathcal{P}(\infty)}$ và tính toán các hàm sinh tương ứng.

Để chứng minh cấu trúc đệ quy của cây $T_{d\text{-}\mathcal{P}}$ chúng tôi sử dụng một số dạng cây con của $T_{d\text{-}\mathcal{P}}$.

Định nghĩa 2.2.4. [57] Ta gọi cây con X_k của $T_{d\text{-}\mathcal{P}}$ thỏa mãn:

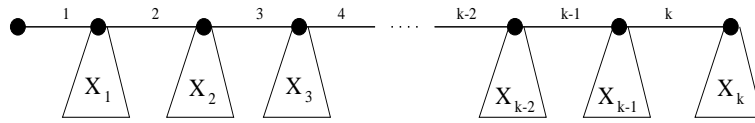
+ gốc của cây đặt tại phân tử $a = (m, m - d, m - 2d, \dots, m - (k-1)d, a_{k+1}, \dots)$, trong đó $a_{k+1} \leq m - kd - 1$,

+ Nếu a chỉ có một con thì X_k là toàn bộ cây con có gốc tại a ,

+ Nếu a có hai con thì X_k là cây gốc a và con trái của a ,

+ X_0 là một nút.

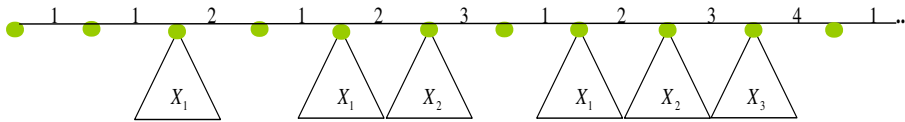
Sử dụng kỹ thuật chứng minh trong [57], ta có thể chứng minh rằng tất cả các cây con X_k là đẳng cấu với nhau (hình 2.5).



Hình 2.5: Cấu trúc đệ quy của các cây con X_k

Cấu trúc của các cây con X_k được thể hiện trong mệnh đề sau:

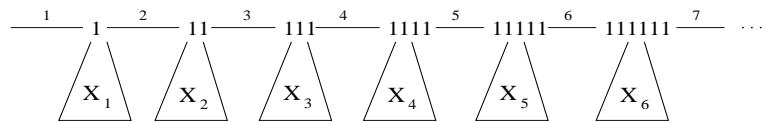
Mệnh đề 2.2.5. *Mỗi cây con X_k ($k \geq 1$) là một dây chuyền $k + 1$ nút, các cạnh của nó được dán nhãn $1, 2, \dots, k$ và nút thứ i được nối với nút tiếp theo bởi cạnh được dán nhãn i và nút thứ i là gốc của cây con $X_{i-1}, 1 \leq i \leq k + 1$.*



Hình 2.6: Biểu diễn cây T_{d-P} như một dây chuyền.

Từ cấu trúc đệ quy này cho phép ta biểu diễn cây T_{d-P} như một dây chuyền (hình 2.9). Trong trường hợp $d = 0$, ta nhận được cây T_P như trong [57] (hình 2.7) và trong trường hợp $d = 1$, ta nhận được biểu diễn của cây T_{SP} như trong [54] (hình 2.9):

Định lý 2.2.6. [57] *Cây T_P có thể được biểu diễn như một dây chuyền vô hạn như sau: nút thứ i của dây chuyền này là $b = \underbrace{(1, 1, \dots, 1)}_{i-1}$ được nối với nút tiếp theo bởi cạnh dán nhãn i , và b là gốc của cây con X_{i-1} .*



Hình 2.7: Biểu diễn cây T_P như một dây chuyền

Hơn nữa, ta có thể chứng minh tính chất mạnh hơn của mỗi cây con trong dây chuyền:

Hệ quả 2.2.7. *Cây con X_k của cây T_{d-P} có gốc tại $(1+(k-1)d, 1+(k-2)d, \dots, 1+2d, 1+d, 1)$ chứa đúng các phân hoạch có độ dài k .*

Chú ý rằng, trong trường hợp tổng quát cây T_{d-P} không có dạng $X_\infty := \lim_{k \rightarrow \infty} X_k$. Tuy nhiên, điều này đúng trong trường hợp $d = 0$.

Hệ quả 2.2.8. [57] *Cây T_P đẳng cấu với cây X_∞ .*

Theo bổ đề 2.2.3, ta có toán tử ECO cho các phân hoạch chặt được định nghĩa như sau:

Định nghĩa 2.2.9. $\vartheta_1 : \mathcal{SP}(n) \rightarrow 2^{\mathcal{SP}(n+1)}$ thỏa mãn: với mỗi $a = (a_1, a_2, \dots, a_l) \in \mathcal{SP}(n)$, $a^{\downarrow 1} = (a_1 + 1, a_2, \dots, a_l)$ là một phân tử của $\vartheta_1(a)$, gọi là (*con trái*) của a . Hơn nữa, nếu a có dạng $(m, m-1, \dots, m-i, m-i-2, a_{i+3}, \dots, a_l)$ ($a_{i+3} \leq m-i-3$) thì $\vartheta_1(a)$ chứa thêm phân tử (gọi là *con phải* của a) có dạng: $a^{\downarrow i+2} = (m, m-1, \dots, m-i, m-i-1, a_{i+3}, \dots, a_l)$.

Toán tử ECO này cho tương ứng cây dán nhãn các phân hoạch chặt T_{SP} được minh họa ở hình vẽ 2.8.

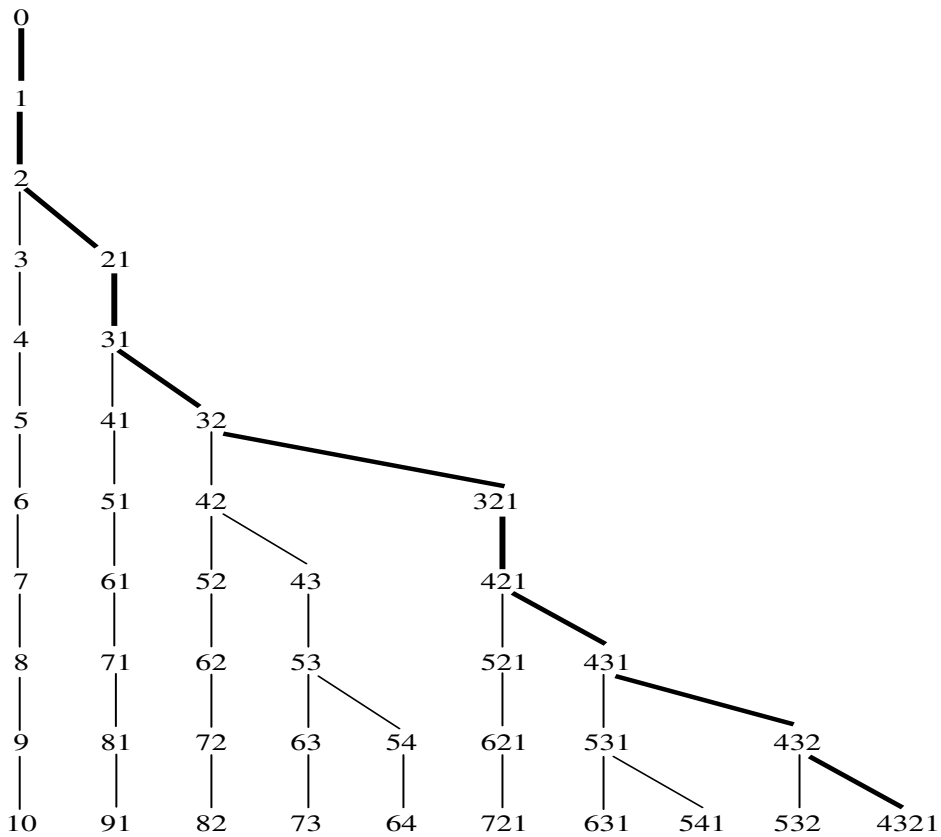
Với định nghĩa của cây X_k như trong mục 2.2.3, ta có thể mô tả cấu trúc đệ quy của cây T_{SP} như là một trường hợp đặc biệt:

Bổ đề 2.2.10. *Cho $a = (m, m-1, \dots, m-k, a_{k+2}, \dots)$ (với $a_{k+2} \leq m-k-2$).*

Khi đó:

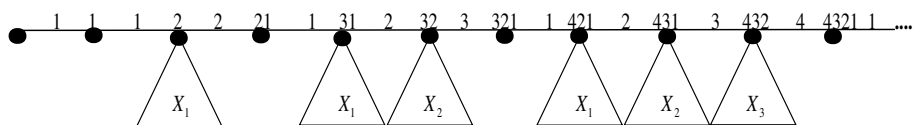
- + *Nếu a chỉ có một con trái thì cây con gốc tại a có dạng X_{k+1} ,*
- + *Nếu a có hai con thì cây gốc a chứa con trái của a có dạng X_{k+1} .*

Nhận thấy rằng, các phân tử trên đường phải cùng của cây T_{SP} có dạng $a = (m, m-1, m-2, \dots, m-k)$ nên thỏa mãn điều kiện trong Bổ đề 2.2.10. Mỗi nút $a = (m, m-1, m-2, \dots, m-k)$ trên đường phải cùng của cây T_{SP} treo một cây con X_{k+1} tương ứng. Cấu trúc đệ quy này cho phép ta biểu diễn cây T_{SP} như là một dây chuyền (hình 2.9) và ta có định lý sau:



Hình 2.8: Cây các phân hoạch chặt

Định lý 2.2.11. *Cây T_{SP} có thể được biểu diễn như một dãy chuyển vô hạn $(\ , 1, 2, 21, 31, 32, 321, \dots, (n - 1, n - 2, \dots, 1), (n, n - 2, \dots, 2, 1), \dots, (n, n - 1, \dots, 3, 2), (n, n - 1, \dots, 3, 2, 1), \dots$ với các cạnh tương ứng $1, 1, 2, 1, 2, 3, \dots, 1, 2, \dots, n, \dots$; mỗi nút trước cạnh $k \geq 1$ là gốc của cây con X_{k-1} .*



Hình 2.9: Biểu diễn cây T_{SP} như một dãy chuyển.

2.3 Một số tính toán trên cây vô hạn

Trong phần này chúng tôi sử dụng cấu trúc đệ quy của cây sinh T_P và kỹ thuật dán nhãn trên cây để tính toán hàm sinh của các dạng phân hoạch và chứng minh lại một số đẳng thức về phân hoạch. Với mỗi tính toán chúng tôi sẽ có cách dán nhãn thích hợp.

Trước hết, chúng tôi định nghĩa hàm sinh cho cây vô hạn.

Định nghĩa 2.3.1. a) Cho T là cây vô hạn, dán nhãn t trên các cạnh của nó theo một qui luật nào đó. Gọi $n_t(a)$ là số nhãn t trên đường đi từ gốc của cây T đến đỉnh a . Khi đó, hàm sinh (generating function) một biến của T ứng với cách dán nhãn đó là: $f_T(t) = \sum_{a \in T} t^{n_t(a)}$,

b) Cho T là cây vô hạn, dán nhãn t và s trên các cạnh của nó theo một qui luật nào đó. Gọi $n_t(a)$ (tương ứng, $n_s(a)$) là số nhãn t (tương ứng, s) trên đường đi từ gốc của cây T đến đỉnh a . Khi đó, hàm sinh (generating function) hai biến của T ứng với cách dán nhãn đó là: $f_T(t, s) = \sum_{a \in T} t^{n_t(a)} s^{n_s(a)}$.

Từ định nghĩa trên ta có ngay $f_T(t) = \sum_{n=0}^{\infty} p(n)t^n$, trong đó $p(n)$ là số đỉnh ở tầng thứ n của cây T .

Từ đây trở về sau, ta dán nhãn t trên tất cả các cạnh của cây T đang xét. Ta có kết quả đầu tiên trên cây con X_k và cây T_P .

Bổ đề 2.3.2. Hàm sinh cho mỗi cây con X_k là

$$f_{X_k}(t) = \frac{1}{(1-t)(1-t^2)\dots(1-t^k)}.$$

Hơn nữa, hàm sinh của cây T_P là

$$f_{T_P}(t) = \prod_{n=1}^{\infty} \frac{1}{(1-t^n)}, \text{ hoặc } f_{T_P}(t) = 1 + \sum_{n=1}^{\infty} t^n \prod_{k=1}^n \frac{1}{(1-t^k)}.$$

Chứng minh. Ta có $f_{X_1}(t) = \frac{1}{1-t}$. Khi đó, theo mệnh đề 2.2.5, ta có $f_{X_k}(t) = f_{X_{k-1}}(t) + t^k f_{X_k}(t)$. Chứng minh quy nạp ta tính được $f_{X_k}(t)$.

Hơn nữa, từ hệ quả 2.2.8, T_P có dạng X_∞ , nên ta có công thức đầu tiên của hàm sinh f_{T_P} . Tương tự, áp dụng định lý 2.2.6, ta nhận được công thức thứ hai của f_{T_P} . \square

Bây giờ, vì T_P là cây của các phân hoạch nên hàm sinh $P(t)$ của các phân hoạch của số tự nhiên bằng hàm sinh $f_{T_P}(t)$, tức là ta có

$$P(t) = \prod_{n=1}^{\infty} \frac{1}{(1-t^n)}.$$

Tiếp theo, bằng tính toán hàm sinh hai biến của cây T_P theo các cách khác nhau, ta nhận được đẳng thức Euler thứ nhất trong định lý sau:

Định lý 2.3.3. (Đẳng thức Euler 1)

$$1 + \sum_{k=1}^{\infty} \frac{s^k t^k}{(1-t) \dots (1-t^k)} = \prod_{i=1}^{\infty} \frac{1}{1-st^i}.$$

Chứng minh. Ta dán nhãn s trên cây T_P theo hai cách khác nhau và chứng minh rằng hai cách dán nhãn này cho cùng một hàm sinh.

Trước hết, ta dán nhãn s trên mỗi cạnh $a \rightarrow a^{\downarrow 1}$. Khi đó, với mỗi đỉnh λ trên cây T_P , nhãn s từ gốc cây đến đỉnh a đúng bằng thành phần lớn nhất $a(\lambda)$ của phân hoạch λ . Ký hiệu $f^1(t, s)$ là hàm sinh theo cách dán nhãn này. Theo cách dán nhãn này, cây T_P có dạng X_∞ nên ta có $f_{X_1}^1(t, s) = f_{X_1}^1(t, s) = \frac{1}{1-st}$ và do đó $f_{X_k}^1(t, s) = f_{X_{k-1}}^1(t, s) + st^k f_{X_k}^1(t, s)$. Từ đây suy ra

$$\sum_{l(\lambda) \leq k} t^{|\lambda|} s^{a(\lambda)} = f_{X_k}^1(t, s) = \prod_{i=1}^k \frac{1}{1-st^i}, \text{ và}$$

$$\sum_{\lambda \in P} t^{|\lambda|} s^{a(\lambda)} = f_{T_P}^1(t, s) = \prod_{i=1}^{\infty} \frac{1}{1-st^i}.$$

Tiếp theo, ta dán nhãn cây T_P theo cách thứ hai: dán nhãn s trên đường phải nhất của cây T_P . Gọi $f^2(t, s)$ là hàm sinh tương ứng với cách dán nhãn này. Khi đó với mỗi nút a trên cây T_P , số nhãn s từ gốc cây đến nút a bằng độ dài $l(\lambda)$ của

phân hoạch λ . Ta thấy rằng không có nhãn s trong các cây X_k nên hàm sinh $f_{X_k}^2$ chính là hàm f_{X_k} .

Theo cách dán nhãn này, cây T_P không có dạng X_∞ mà là tổng của các cây X_k nên ta có

$$\sum_{\lambda \in P} t^{|\lambda|} s^{l(\lambda)} = f_{T_P}^2(t, s) = 1 + \sum_{k=1}^{\infty} s^k t^k f_{X_k}(t) = 1 + \sum_{k=1}^{\infty} \frac{s^k t^k}{(1-t) \dots (1-t^k)}.$$

Cuối cùng, do số các phân hoạch của n có thành phần lớn nhất là k bằng số các phân hoạch của n có độ dài k , nên ta có hai hàm sinh $f_{T_P}^1(t, s)$ và $f_{T_P}^2(t, s)$ bằng nhau. Từ đó ta có đẳng thức Euler thứ nhất:

$$1 + \sum_{k=1}^{\infty} \frac{s^k t^k}{(1-t) \dots (1-t^k)} = \prod_{i=1}^{\infty} \frac{1}{1-st^i}.$$

□

Sau đây, ta tính hàm sinh cho các phân hoạch chặt và lý luận tương tự như trong Định lý 2.3.3, ta nhận được đẳng thức Euler thứ hai

Định lý 2.3.4. (Đẳng thức Euler 2)

$$1 + \sum_{k=1}^{\infty} \frac{s^k t^{k(k+1)/2}}{k \prod_{i=1}^k (1-t^i)} = \prod_{i=1}^{\infty} (1+st^i).$$

Chứng minh. Ta dán nhãn t trên mỗi cạnh của cây T_{SP} . Khi đó ta có công thức hàm sinh của cây T_{SP} :

$$f_{T_{SP}}(t) = \sum_{k=0}^{\infty} sp(k) t^k = 1 + \sum_{k=1}^{\infty} t^{k(k+1)/2} f_{X_k}(t) = 1 + \sum_{k=1}^{\infty} \frac{t^{k(k+1)/2}}{k \prod_{i=1}^k (1-t^i)},$$

ở đây $sp(k)$ là số các phân hoạch chặt của k .

Mặt khác, ta đã biết hàm sinh của các phân hoạch chặt là $SP(t) = \prod_{k=0}^{\infty} (1+t^k)$, nên ta có

$$1 + \sum_{k=1}^{\infty} \frac{t^{k(k+1)/2}}{k \prod_{i=1}^k (1-t^i)} = \prod_{k=0}^{\infty} (1+t^k).$$

Hoàn toàn tương tự, ta tính hàm sinh hai biến của cây T_{SP} . Trước hết, ta định nghĩa hàm sinh hai biến cho các phân hoạch chặt:

$$SP(t, s) = \sum_n \left(\sum_{\lambda \vdash n} s^{l(\lambda)} \right) t^n$$

ở đây, số mũ s biểu diễn độ dài của phân hoạch.

$$\text{Để thấy rằng } SP(t, s) = \prod_{i=1}^{\infty} (1 + st^i).$$

Hơn nữa, trên cây T_{SP} , nếu ta dán nhãn s dọc theo các cạnh $() \rightarrow 1$ và $(m, m-1, \dots, 3, 2) \rightarrow (m, m-1, \dots, 3, 2, 1)$, $m \geq 2$ trên đường phải nhất, với số mũ s biểu thị cho độ dài của các phân hoạch chặt thì hàm sinh hai biến của cây T_{SP} chính là $SP(t, s)$. Sử dụng cấu trúc đệ quy của cây T_{SP} (Định lý 2.2.11), ta có:

$$f_{T_{SP}}(t, s) = \sum_n \left(\sum_{\lambda \vdash n} s^{l(\lambda)} \right) t^n = 1 + \sum_{k=1}^{\infty} \frac{s^k t^{k(k+1)/2}}{\prod_{i=1}^k (1 - t^i)},$$

Từ đó ta có đẳng thức Euler 2.

□

Để kết thúc phần này, chúng tôi đưa ra công thức hàm sinh cho cây T_{d-P} . Ký hiệu $d-p(n) = |d-\mathcal{P}(n)|$, là số các phân hoạch d -chặt của n . Ta định nghĩa hàm sinh một biến và hàm sinh hai biến tương ứng của cây T_{d-P} như sau:

$$f_{T_{d-P}}(t) = \sum_{n=0}^{\infty} d-p(n) t^n \text{ và } f_{d-P}(t, s) = \sum_{n=0}^{\infty} \left(\sum_{\lambda \in d-\mathcal{P}(n)} s^{l(\lambda)} \right) t^n.$$

Sử dụng kỹ thuật dán nhãn như trên, ta tính được các hàm sinh của cây này như sau:

$$f_{T_{d-P}}(t) = \sum_{n=0}^{\infty} d-P(n) t^n = 1 + \sum_{n \geq 1} \frac{t^{\frac{n(nd+2-d)}{2}}}{(1-t)(1-t^2) \dots (1-t^n)}.$$

$$f_{T_{d-P}}(t, s) = \sum_{n=0}^{\infty} \left(\sum_{\lambda \in d-\mathcal{P}(n)} s^{l(\lambda)} \right) t^n = 1 + \sum_{n \geq 1} \frac{s^n t^{\frac{n(nd+2-d)}{2}}}{(1-t)(1-t^2) \dots (1-t^n)}.$$

Đặc biệt, trong trường hợp $d = 2$, ta nhận được công thức

$$f_{T_{2-P}}(t) = 1 + \sum_{n \geq 1} \frac{t^{n^2}}{(1-t)(1-t^2) \dots (1-t^n)},$$

là vế phải của đẳng thức Ramanujan:

$$1 + \sum_{n \geq 1} \frac{t^{n^2}}{(1-t)(1-t^2) \dots (1-t^n)} = \prod_{i=0}^{\infty} \frac{1}{(1-t^{5i+1})(1-t^{5i+4})}.$$

Để chứng minh đẳng thức này, ta xây dựng cây sinh cho vế phải (các phân hoạch với các phần có dạng $5d + 1$ hoặc $5d + 4$). Như vậy, với việc sử dụng công cụ duy nhất là cây ECO (cây sinh của toán tử ECO), chúng tôi đã chứng minh được một số song ánh giữa các dạng phân hoạch, và bằng kỹ thuật dán nhãn trên cây ECO đã chứng minh được một số đẳng thức về phân hoạch của số tự nhiên.

2.4 Kết luận chương 2

Trong chương này chúng tôi nghiên cứu tập các phân hoạch d -chặt của số tự nhiên và đã thu được các kết quả sau:

- Theo hướng tiếp cận hệ động lực rời rạc chúng tôi đã xây dựng hệ động lực rời rạc có không gian trạng thái chính là tập $d\text{-}\mathcal{P}(n)$ các phân hoạch d -chặt của một số tự nhiên n cho trước và nghiên cứu cấu trúc của tập này cũng như mở rộng của nó:

- + Chứng minh được $d\text{-}\mathcal{P}(n)$ có cấu trúc dàn và chỉ ra công thức tường minh của điểm dừng.

- + Đưa ra quan hệ cấu trúc giữa $d\text{-}\mathcal{P}(n)$ và $d\text{-}\mathcal{P}(n+1)$. Chứng minh cấu trúc dàn của $d\text{-}\mathcal{P}(\infty)$ - là mở rộng vô hạn của $d\text{-}\mathcal{P}(n)$. Đưa ra một số tính chất của cây bao trùm $T_{d\text{-}\mathcal{P}(\infty)}$ của dàn $d\text{-}\mathcal{P}(\infty)$.

- Theo hướng tiếp cận bằng phương pháp ECO, chúng tôi đã xây dựng toán tử ECO cho phân hoạch d -chặt của các số tự nhiên và xây dựng cây sinh tương ứng.

Cây sinh của toán tử ECO này chính là cây $T_{d-\mathcal{P}(\infty)}$ như đã trình bày theo cách tiếp cận của hệ động lực rời rạc.

+ Xây dựng được cấu trúc đệ quy của cây sinh $T_{d-\mathcal{P}(\infty)}$. Đặc biệt, khi $d = 1$ chúng tôi nhận được cấu trúc đệ quy của cây sinh của các phân hoạch chặt.

+ Chứng minh được đẳng thức Euler thứ nhất và đẳng thức Euler thứ hai bằng kỹ thuật dán nhãn trên các cây sinh.

Chương 3

Các hệ động lực CFG và mạng Petri

Trong chương này chúng tôi nhắc lại một số định nghĩa, tính chất, các hướng nghiên cứu về các hệ CFG và trình bày một số kết quả về mối quan hệ giữa các hệ động lực CFG và mạng Petri. Chúng tôi chứng minh các hệ CFG là các mạng Petri đặc biệt.

3.1 CFG cổ điển

Trong phần này, chúng tôi trình bày các kết quả về cấu trúc của CFG cổ điển. Các kết quả này được tham khảo trong các bài báo [4], [52].

3.1.1 Các định nghĩa

Định nghĩa 3.1.1. *Hệ động lực CFG (Chip Firing Game) [4] được định nghĩa trên một đa đồ thị (có hướng hoặc vô hướng) $G = (V, E)$. Mỗi trạng thái là một phân hoạch chip trên các đỉnh của đồ thị, luật vận động được định nghĩa như sau: mỗi đỉnh có thể cháy được nếu số chip tại đỉnh đó lớn hơn hoặc bằng bậc (đi) ra của nó và hoạt động cháy của đỉnh đó sẽ là chuyển một số chip đến các đỉnh lân cận dọc theo mỗi cạnh (hay cung) đi ra từ nó. Hệ này được ký hiệu là $CFG(G)$, G gọi là đồ thị nền của hệ.*

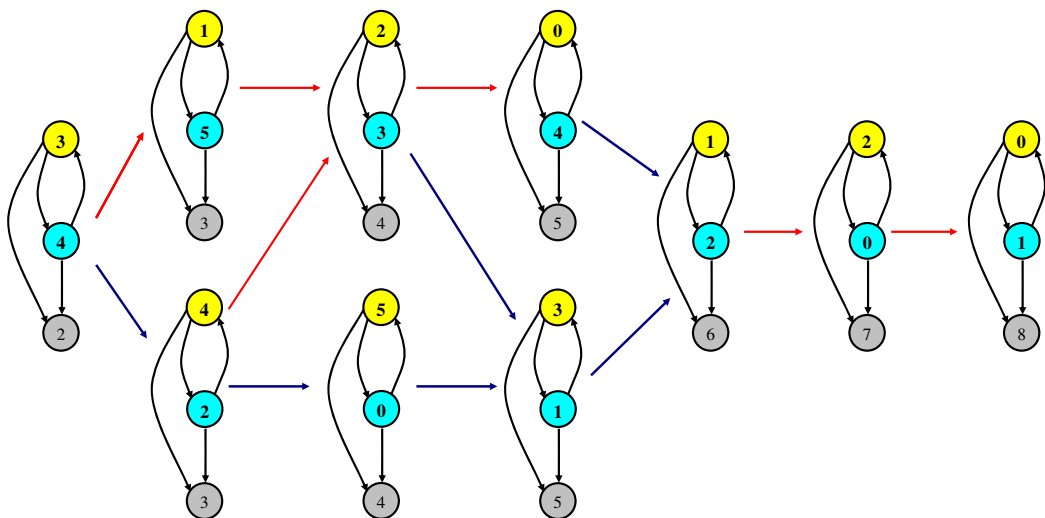
Nếu tại mỗi thời điểm chỉ áp dụng luật cháy cho một đỉnh (cháy được) thì ta có

hệ CFG tuần tự. Nếu tại mỗi thời điểm ta áp dụng luật chấy đồng thời cho tất cả các đỉnh (chấy được) thì ta có CFG song song. Khi xét một hệ CFG nếu không nói gì thêm thì ta hiểu CFG này là CFG tuần tự.

Một trạng thái $a = (a_1, \dots, a_m)$ của hệ là một hợp thành m phần của số tự nhiên n , ở đây n là tổng số chip của hệ (n sẽ không thay đổi trong quá trình biến đổi của hệ), $m = |V|$ là số đỉnh của đồ thị G , a_i là số chip tại đỉnh $i \in V$. Đỉnh i gọi là chấy được (firable) tại trạng thái a nếu $a_i \geq \deg^+(i)$ ($\deg^+(i)$ là tổng bậc đi ra của i).

Tùy theo mục đích nghiên cứu ta có một số ký hiệu của hệ động lực CFG như sau:

- $CFG(G, \mathcal{O})$ là hệ CFG với trạng thái ban đầu \mathcal{O} , trong trường hợp này ký hiệu $CFG(G, \mathcal{O})$ cũng để chỉ không gian trạng thái của hệ, tức là tập hợp tất cả các trạng thái nhận được từ trạng thái \mathcal{O} . Trong hệ $CFG(G, \mathcal{O})$ tổng số chip trên các đỉnh của đồ thị không thay đổi.
- $CFG(G, n)$ là hệ CFG với tổng số chip n không thay đổi trong suốt quá trình biến đổi của hệ, $CFG(G, n)$ là hợp của các $CFG(G, \mathcal{O})$.



Hình 3.1: Quá trình chuyển trạng thái của một CFG với 9 chips.

Việc nghiên cứu hành vi của hệ CFG là khá phức tạp, trong trường hợp tổng quát thì CFG có thể không dừng. Năm 1991, A. Bjoner, L.Lovász và P.W.Shor đã chứng minh tính dừng của CFG trong trường hợp vô hướng như sau:

Định lý 3.1.2. [4] Cho $CFG(G, n)$ trên đồ thị vô hướng $G = (V, E)$. Khi đó

a) Nếu $n > 2|V| - |E|$ thì $CFG(G, n)$ không dừng.

b) Nếu $|E| \leq n \leq 2|E| - |V|$ thì tồn tại trạng thái \mathcal{O}_1 sao cho $CFG(G, \mathcal{O}_1)$ không dừng và tồn tại trạng thái \mathcal{O}_2 sao cho $CFG(G, \mathcal{O}_2)$ dừng.

c) Nếu $n < |E|$ thì $CFG(G, n)$ dừng.

Cấu trúc dàn của CFG trong trường hợp có hướng đã được chứng minh trong [5], [52]. Tuy nhiên, đồ thị nền của lớp CFG này thỏa mãn tính chất không chứa thành phần đóng.

3.1.2 Cấu trúc dàn của không gian trạng thái

Trong phần này chúng tôi sẽ trình bày cấu trúc dàn của không gian trạng thái của CFG trên đồ thị nền không chứa thành phần đóng. Các kết quả của mục này được tham khảo trong [5], [52]. Trong toàn bộ mục này $G = (V, E)$ là đa đồ thị có hướng không chứa thành phần đóng. Trước hết ta đặc trưng cấu trúc thứ tự của không gian trạng thái.

Định nghĩa 3.1.3. (Quan hệ kế tiếp) Cho $CFG(G, \mathcal{O})$. Trên $CFG(G, \mathcal{O})$ ta xác định một quan hệ hai ngôi như sau: với $a, b \in CFG(G, \mathcal{O})$, $a \geq_{CFG} b$ khi và chỉ khi b nhận được từ a sau khi áp dụng luật cháy vào một số đỉnh. Đặc biệt, ta ký hiệu $a \rightarrow b$ để chỉ b nhận được từ a sau khi cháy một đỉnh.

Khi nói về quan hệ thứ tự trên CFG, chúng tôi ký hiệu \geq thay cho \geq_{CFG} .

Cho $p \in V^*$ là một dãy các đỉnh. Ký hiệu $|p|_i$ là số lần áp dụng luật cháy cho đỉnh i trong dãy p . Ta có bổ đề sau:

Bổ đề 3.1.4. Cho $s, t \in V^*$, a là một trạng thái bất kỳ. Khi đó nếu xuất phát từ a , áp dụng luật cháy cho các đỉnh trong s hoặc t cùng dẫn đến trạng thái b thì $|s|_i = |t|_i$.

Bổ đề này cho phép ta định nghĩa khái niệm *shot-vectơ* của hai trạng thái a và b với b là trạng thái nhận được từ a trong một CFG.

Định nghĩa 3.1.5. Cho $a, b \in CFG(G)$. Ký hiệu $k_i(a, b)$ là số lần cháy đỉnh i trong quá trình từ a đến b . *Shot-vectơ* $k(a, b)$ là bộ $(k_1(a, b), \dots, k_m(a, b))$, $|k(a, b)| = \sum_{i=1}^n k_i(a, b)$ là số lần áp dụng luật cháy để nhận được b từ a .

Nếu a, b cùng nhận được từ một trạng thái \mathcal{O} thì ta định nghĩa $k(\mathcal{O}, a) \leq k(\mathcal{O}, b)$ khi và chỉ khi $\forall i, k_i(\mathcal{O}, a) \leq k_i(\mathcal{O}, b)$. Hơn nữa, nếu $a \geq b$ thì ta có $k(\mathcal{O}, b) = k(\mathcal{O}, a) + k(a, b)$.

Định lý 3.1.6. Cho $a, b \in CFG(G, \mathcal{O})$. Khi đó

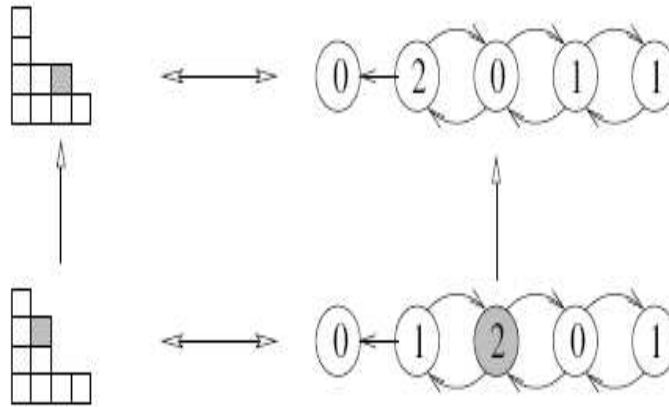
$$a \geq b \iff k(\mathcal{O}, a) \leq k(\mathcal{O}, b).$$

Định lý 3.1.7. Không gian trạng thái của $CFG(G, \mathcal{O})$ với quan hệ thứ tự kế tiếp là một dàn phân bậc. Hơn nữa, cận dưới đúng của hai phần tử a và b được xác định như sau: cho k là vectơ xác định bởi $k_i = \max(k_i(\mathcal{O}, a), k_i(\mathcal{O}, b))$, $\forall i \in V$ và c là trạng thái xác định bởi $k(\mathcal{O}, c) = k$. Khi đó $c = \inf(a, b)$.

Cấu trúc dàn của không gian trạng thái cho phép ta xác định được trạng thái gặp nhau của hai trạng thái bất kỳ: với $a, b \in CFG(G, \mathcal{O})$, tồn tại trạng thái c sao cho từ a và b đều đi được đến c , tức là tồn tại một con đường đi từ a đến c và một con đường đi từ b đến c . Hơn nữa, với mọi trạng thái d bất kỳ mà cả hai trạng thái a và b đều đi được đến d thì từ c cũng đi được đến d . Nếu $a \rightsquigarrow b$ trong $CFG(G, \mathcal{O})$ thì có thể có nhiều đường đi từ a đến b . Định lý 3.1.7 cho thấy mọi con đường đi từ a đến b đều có độ dài bằng nhau.

3.1.3 Mô phỏng hệ SPM bằng CFG

Trong thực tế có nhiều mô hình thực chất là trường hợp đặc biệt của CFG, trong số đó có thể kể đến các hệ như SPM và các mở rộng của nó, dàn Young, mô hình EFG (Edge Firing Games), các hệ thống đếm,... Kết quả mô phỏng các mô hình này bởi hệ CFG được trình bày trong [30]. Sau đây chúng tôi trình bày mô hình SPM được mô phỏng bởi hệ động lực CFG. Cho hệ $SPM(n)$. Xét đồ thị $G = (V, E)$, với $V = \{0, 1, \dots, n\}$ và $E = \{(i, i+1) | 1 \leq i \leq n-1\} \cup \{(i, i-1) | 1 \leq i \leq n\}$. Với mỗi trạng thái p của $SPM(n)$ xác định một trạng thái $\pi(p)$ của $CFG(G, n)$ như sau: đỉnh i chứa $p_i - p_{i+1}$ chips ($i \geq 1$), đỉnh 0 không chứa chip. Khi đó, với mỗi trạng thái $\pi(p)$ của CFG ta có $\pi(p) \rightarrow \pi(p')$ khi và chỉ khi $p \rightarrow p'$ trong $SPM(n)$. Do đó, với trạng thái đầu $\pi((n))$, không gian trạng thái $CFG(G, (n))$ đẳng cấu với $SPM(n)$. Hình 3.2 là một ví dụ.



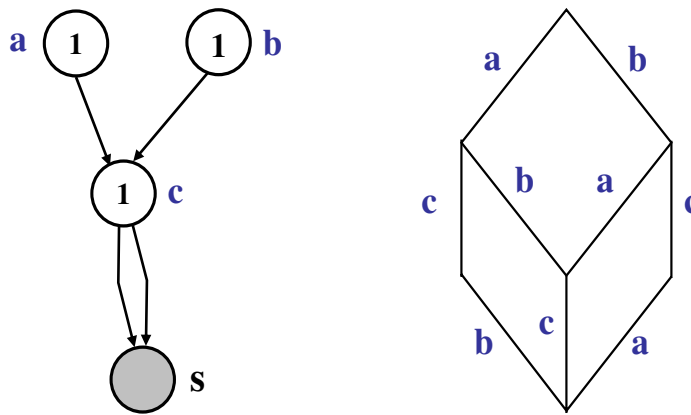
Hình 3.2: Mã hoá một SPM bằng một CFG

3.1.4 CFG tô màu

Như đã biết không gian trạng thái của một CFG hội tụ là một dàn phân phối. Ký hiệu $L(CFG)$ là tập tất cả các dàn sinh bởi các CFG hội tụ. Năm 2001, nhóm nghiên cứu của C. Magnien, H. D. Phan và L. Vuillon đã chứng minh rằng lớp các dàn phân phối thực sự chứa trong lớp dàn $L(CFG)$ và lớp dàn $L(CFG)$ thực sự

chứa trong lớp dàn ULD. Như vậy, với mỗi dàn phân phối L_D luôn tồn tại một CFG hội tụ có không gian trạng thái là L_D , và tồn tại những CFG hội tụ có không gian trạng thái là dàn không phân phối (hình 3.3). Tuy nhiên, tồn tại những dàn ULD không là không gian trạng thái của một CFG hội tụ nào (hình 3.4). Ký hiệu \mathcal{D} là tập tất cả các dàn phân phối. Khi đó ta có kết quả sau:

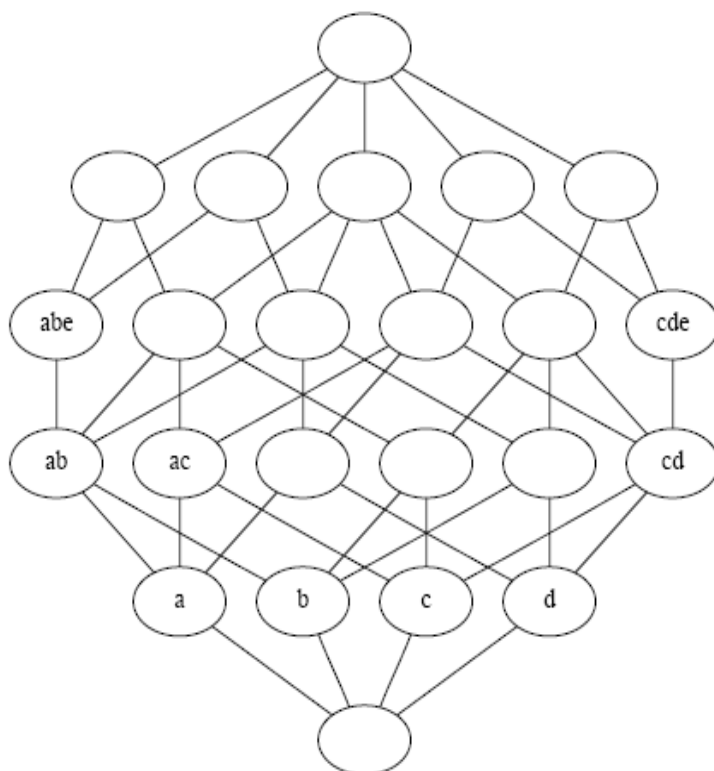
Định lý 3.1.8. [64, Định lý 4.1 và định lý 4.2] $\mathcal{D} \subsetneq L(\text{CFG}) \subsetneq \text{ULD}$.



Hình 3.3: CFG và không gian trạng thái tương ứng

Trong phần này chúng tôi giới thiệu một mở rộng của CFG sinh ra đúng lớp dàn ULD.

Định nghĩa 3.1.9. [64] Cho đồ thị $G = (V; E)$ và X là tập các màu. Ta gọi *đồ thị tô màu* (coloured graph) là bộ $(V; E; X; col)$ trong đó col là ánh xạ màu từ E vào X . Hạn chế của đồ thị này lên màu $c \in X$ là đồ thị $G_c = (V; col^{-1}(c))$ chỉ gồm các cạnh có màu c . Mô hình CFG tô màu được định nghĩa trên một đa đồ thị có hướng tô màu $G = (V; E; X; col)$. Mỗi trạng thái của CFG này được cho bởi một hàm $\sigma : V \rightarrow \mathbb{N}^X$, tại mỗi đỉnh chứa một số chip với các màu khác nhau. Với mỗi $v \in V, c \in X$, ta ký hiệu $\sigma_c(v)$ là số các chip có màu c tại đỉnh v . Tại mỗi thời điểm, mỗi đỉnh có một trạng thái là đóng hay mở. Luật vận động của CFG tô màu là việc mở các đỉnh. Điều kiện để có thể mở đỉnh v là:



Hình 3.4: Dàn ULD không là không gian trạng thái của một CFG nào

- v đang đóng,
- tồn tại một màu $c \in X$ sao cho v có thể cháy (theo nghĩa cổ điển) trên G_c (tức là, số chip có màu c tại đỉnh v lớn hơn hoặc bằng số cung có màu c đi ra từ đỉnh v).

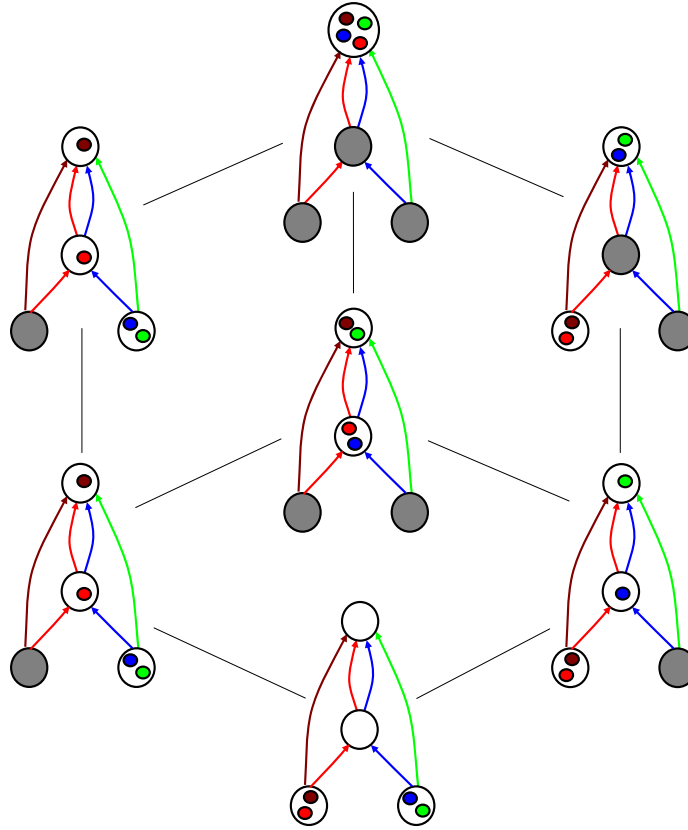
Việc mở đỉnh v gồm có:

- đánh dấu đỉnh v đã mở,
- với mỗi màu $c \in X$, xét CFG màu c hạn chế trên các đỉnh đã mở, cho CFG vận hành đến khi đạt đến trạng thái cuối cùng.

Mô hình CFG tô màu trên đồ thị G ký hiệu là $ColCFG(G)$.

Câu hỏi đặt ra là liệu CFG tô màu có dừng không? Để đảm bảo điều này thì các CFG màu c phải hội tụ. Điều đó có nghĩa là các đồ thị G_c phải không chứa thành

phần đóng [52]. Như vậy khi xét CFG tô màu trên đồ thị tô màu $G = (V; E; X; col)$, điều kiện không chứa thành phần đóng của các đồ thị $G = (V, E, \{c\})$ phải được thỏa mãn.



Hình 3.5: Không gian trạng thái của một CFG tô màu

Hành vi của CFG tô màu như sau: đầu tiên tất cả các đỉnh đều đóng, ta chọn một đỉnh có thể mở trên màu c_1 (thỏa mãn điều kiện mở được), chày đỉnh này trên đồ thị G_{c_1} đến khi đạt đến trạng thái dừng, ta được một trạng thái mới của CFG tô màu. Sau đó, lại tiếp tục mở đỉnh tiếp theo với màu c_2 , cho CFG vận hành như CFG cổ điển trên đồ thị $G = G_{c_2}$ và trên đỉnh vừa mở lần trước. Quá trình tiếp tục cho đến khi không còn đỉnh nào có thể mở được ta được trạng thái cuối cùng của CFG tô màu. Như vậy, CFG tô màu trên đồ thị tô màu $G = (V, E, X, col)$ có thể xem là hợp của các CFG cổ điển trên các đồ thị G_c . Không gian trạng thái của CFG tô màu được sắp thứ tự bộ phận theo quan hệ bao hàm: $\sigma \geq \sigma'$ khi và chỉ khi $s(\sigma) \subseteq s(\sigma')$, trong đó $s(\sigma)$ là *shot-set* của trạng thái σ , tức là tập tất cả các đỉnh

đã mở để nhận được trạng thái σ từ trạng thái ban đầu.

Ta thấy rằng CFG tô màu chính là mở rộng của CFG. Mỗi trạng thái σ của CFG tô màu được xác định duy nhất bởi shot-set $s(\sigma)$ của nó. Tập tất cả các shot-set tương ứng với không gian trạng thái của CFG tô màu. Tập hợp này được sắp thứ tự bởi quan hệ bao hàm tập hợp và hiển nhiên có phần tử nhỏ nhất là tập hợp rỗng. Hơn nữa, tập này đóng với phép toán hợp của hai tập hợp, do đó không gian trạng thái của CFG tô màu là một dàn. Các kết quả này được thể hiện trong định lý sau:

Định lý 3.1.10. [64, Định lý 5.2] *Không gian trạng thái của một CFG tô màu là một dàn ULD.*

Cuối cùng, kết quả trong [64] còn chứng minh được nhiều hơn thế. Bằng cách sử dụng khái niệm *phần tử bất khả quy*, C. Magnien, H. D. Phan và L. Vuillon trong [64] đã chứng minh rằng lớp CFG tô màu sinh ra đúng lớp dàn ULD. Ký hiệu $L(C)$ là dàn sinh bởi CFG tô màu C , tức là $L(C)$ là không gian trạng thái của CFG tô màu C . Khi đó, ta có định lý sau:

Định lý 3.1.11 (Định lý 5.6). [64] *Cho L là dàn ULD. Khi đó tồn tại CFG tô màu C sao cho $L = L(C)$.*

3.2 Hệ động lực CCFG

Trong thực tế có nhiều hệ CFG mở rộng với điều kiện cháy tại mỗi đỉnh được giảm nhẹ hơn. Điều kiện để một đỉnh v cháy được trong CFG cổ điển là khá chặt: nó phải đợi nhận được số chip nhiều cần thiết (không nhỏ hơn số cung đi ra từ nó) thì mới cháy được. Điều kiện này có thể được giảm nhẹ hơn phù hợp với một số mô hình thực tế như sau: tại trạng thái a , đỉnh v của CFG có thể cháy được nếu tại đỉnh này chỉ cần chứa ít nhất một chip. Hoạt động cháy cũng được thay đổi: chuyển một chip đến một đỉnh lân cận dọc theo một cung đi ra từ đỉnh v .

Như vậy, qua mỗi bước chuyển trạng thái của hệ CFG mở rộng này, chip được

chuyển từ đỉnh có chứa chip đến một đỉnh lân cận của nó dọc theo cung đi ra. Vì việc chuyển chip đến đỉnh này sẽ ảnh hưởng đến việc chuyển chip đến đỉnh khác nên có sự tương tranh giữa các cung đi ra từ một đỉnh. Vì vậy chúng tôi gọi hệ này là *hệ động lực CFG tương tranh (Conflicting Chip Firing Game - Chip Firing Game tương tranh)* [71] và ký hiệu là $CCFG(G, n)$ để chỉ hệ CFG tương tranh trên đồ thị có hướng G với tổng số chip n không thay đổi trong quá trình biến đổi của hệ.

Để định nghĩa hệ động lực CCFG, trước hết ta có định nghĩa hợp thành (composition) của một số tự nhiên n trên tập hợp S .

Định nghĩa 3.2.1. Cho n là một số nguyên dương, S là một tập hợp bất kỳ có lực lượng $|S| = k$. Một hợp thành của n trên S là một bộ sắp thứ tự (a_1, \dots, a_k) các số nguyên không âm sao cho $a_1 + \dots + a_k = n$.

Định nghĩa 3.2.2. *Hệ động lực CCFG (Conflicting Chip Firing Game - CFG tương tranh)* được định nghĩa trên đồ thị có hướng $G = (V, E)$, ký hiệu là $CCFG(G, n)$:

+ Mỗi trạng thái là một hợp thành của n trên V .

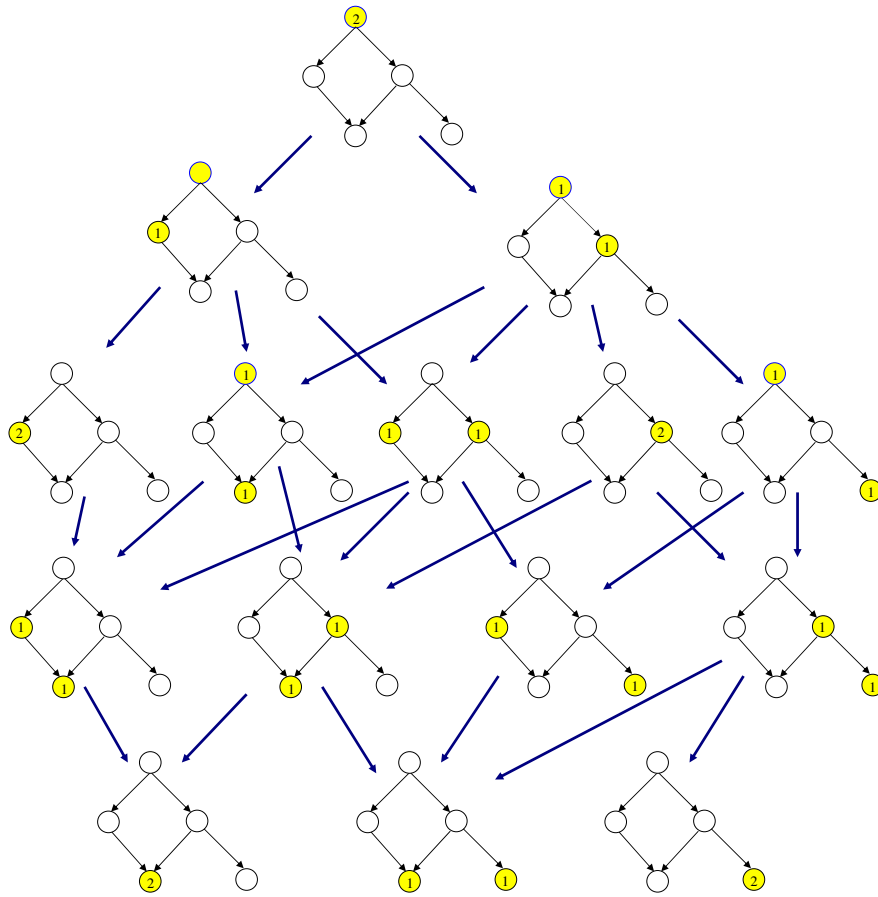
+ Luật vận động:

- Điều kiện cháy: Đỉnh $i \in V$ có thể cháy được tại trạng thái a nếu $a_i > 0$.
- Hoạt động cháy: khi đỉnh i cháy nó chuyển một chip đến một đỉnh lân cận của i .

Định nghĩa 3.2.3. (*Quan hệ kế tiếp*) Cho $CCFG(G, n)$. Trên $CCFG(G, n)$ ta xác định một quan hệ hai ngôi như sau: với $a, b \in CCFG(G, n)$, $a \geq_{CCFG} b$ khi và chỉ khi b nhận được từ a sau khi áp dụng luật cháy vào một số cung. Đặc biệt, ta ký hiệu $a \rightarrow b$ để chỉ b nhận được từ a sau khi cháy một cung.

Khi nói về quan hệ thứ tự trên CCFG, chúng tôi ký hiệu \geq thay cho \geq_{CCFG} .

Mô hình của chúng tôi được đề xuất năm 2006 [Phan, Pham]. Gần đây, chúng tôi tìm hiểu và được biết có những mô hình liên quan đang bắt đầu được nghiên



Hình 3.6: Không gian trạng thái của một CCFG 2 chips

cứu và đã có những kết quả nhất định.

Năm 2008, nhóm nghiên cứu của A.E.Holroyd và J.Propp [36] đã nghiên cứu mô hình Chip-firing trong mối quan hệ với mô hình rotor-router. Mô hình rotor-router được đề xuất đầu tiên vào năm 1996 bởi Priezzhev và cộng sự dưới tên gọi mô hình đường đi Òle (Eurléan walker model) [69] trong mối quan hệ với hiện tượng đột biến tự tổ chức (self-organized criticality). Có thể tóm lược về mô hình rotor-router như sau: mô hình này được định nghĩa trên đồ thị có hướng $G = (V, E)$, với mỗi đỉnh $v \in V$ cố định một thứ tự các cung đi ra từ v . Tại mỗi v , xác định một rotor $\rho(v)$ là một cung đi ra từ v . Đường đi của mỗi chip tại đỉnh v được xác định theo luật rotor-router như sau: chọn cung kế tiếp $\rho(v)$ trong thứ tự các cung đi ra từ v là $e = (v, w)$, sau đó cho chip di chuyển dọc cung e đến đỉnh w . Quá trình lặp lại

cho đến khi chip đi đến đỉnh hút là đỉnh không có cung đi ra thì dừng. Nếu đồ thị không có đỉnh hút thì chip lại tiếp tục đi và quá trình sẽ không dừng. Một cấu hình rotor là một hàm ρ xác định trên tập đỉnh không chứa đỉnh hút và nhận giá trị là $\rho(v) \in E$.

Như vậy, không gian trạng thái của mô hình rotor-router là tập hợp con của không gian trạng thái của hệ CCFG. Trong mô hình CCFG, tại mỗi thời điểm mỗi chip có nhiều khả năng chày và chúng ta xét hết tất cả các khả năng chày, còn trong mô hình rotor-router, tại mỗi thời điểm mỗi chip chỉ có một khả năng chày.

Trong [49] chúng tôi nghiên cứu mô hình rotor-router theo quan điểm của hệ động lực rời rạc. Chúng tôi chứng minh không gian trạng thái của mô hình rotor-router xuất phát từ một trạng thái ban đầu có cấu trúc dàn ULD và chỉ ra rằng lớp dàn sinh bởi các rotor-router không chứa lớp dàn phân phối.

Với việc nghiên cứu mô hình rotor-router trên đồ thị có hướng G với một đỉnh hút toàn cục (global sink), nhóm nghiên cứu A.E.Holroyd và J.Propp trong [36] đã chứng minh rằng nhóm các rotor-router đẳng cấu với nhóm sand pile trên đồ thị có hướng G . Hơn nữa, từ mối quan hệ giữa các trạng thái lặp (recurrent chip configuration) với các cấu hình rotor không chu trình suy ra được định lý ma trận cây (Matrix tree Theorem) trên đồ thị có hướng.

3.3 Mạng Petri

Mạng Petri (PN) là mô hình toán học biểu diễn các hệ thống hoạt động tương tranh, song song hoặc đồng bộ do nhà toán học C.A.Petri đề xuất vào năm 1962 [67]. Mạng Petri là một công cụ toán học hữu hiệu để mô phỏng và phân tích hiệu năng cho các hệ thống từ đơn giản đến phức tạp. Trải qua hơn 40 năm phát triển, từ một mạng Petri đơn giản ban đầu, các nhà nghiên cứu đã cho ra hàng loạt mạng Petri ở mức cao (high level) như Coloured Petri Net, Predicate Petri Net, Stochastic Petri Net, vv. Predicate-Transition Nets và Coloured Petri Nets, được giới thiệu đầu

tiên vào năm 1979 [68, 42, 43], và được nghiên cứu trong thập niên 80 bằng cách sử dụng một số khái niệm của đại số Petri nets đang được phát triển lúc bấy giờ. Người ta tin tưởng rằng các kỹ thuật này nắm bắt được tinh thần của những khái niệm đã được phát triển trước đó và hứa hẹn sẽ có nhiều ứng dụng. Chẳng hạn, nó có thể được sử dụng trực tiếp để xác định các hệ thống hoặc để định nghĩa ngữ nghĩa của các ngôn ngữ không hình thức khác. Nó cũng có thể phục vụ để tích hợp các kỹ thuật đang được sử dụng độc lập như các sơ đồ chuyển trạng thái hay sơ đồ luồng dữ liệu. Kỹ thuật này đặc biệt phù hợp với sự phát triển của các hệ phân tán song song trên cơ sở của tính đồng thời.

Trong phần này chúng tôi chỉ trình bày khái niệm mạng Petri đơn giản đặt trong mối quan hệ với các hệ động lực CFG.

Mạng Petri là một đồ thị hai phần đặc biệt gồm có ba loại đối tượng là các *vị trí* (*places*), các (*cái*) *chuyển* (*transition*) và các cung có hướng. Các cung có hướng nối các vị trí với các chuyển và ngược lại. Nghiên cứu hành vi động lực của mạng Petri là nghiên cứu sự biến chuyển của các trạng thái.

Định nghĩa toán học của mạng Petri như sau:

Định nghĩa 3.3.1. Một *mạng Petri* là một bộ tứ $N = (P, T, I, O)$, trong đó

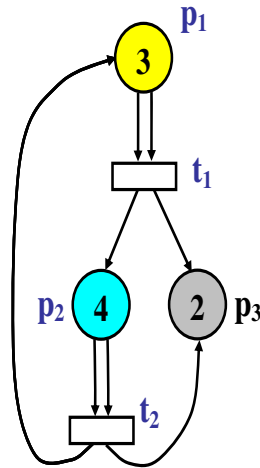
(1) $P = \{p_1, p_2, \dots, p_m\}$ là tập hữu hạn các *vị trí* (*places*);

(2) $T = \{t_1, t_2, \dots, t_n\}$ là tập hữu hạn các (*cái*) *chuyển* (*transition*), $P \cup T \neq \emptyset$, và $P \cap T = \emptyset$;

(3) $I : P \times T \rightarrow \mathbb{N}$ là *hàm vào*, xác định trọng số của cung đi từ vị trí đến chuyển;

(4) $O : T \times P \rightarrow \mathbb{N}$ là *hàm ra*, xác định trọng số của cung đi từ chuyển đến vị trí;

Mạng Petri $N = (P, T, I, O)$ không chứa trạng thái đầu được ký hiệu là N . Mạng Petri với trạng thái ban đầu $M_0 : P \rightarrow \mathbb{N}$ được ký hiệu là (N, M_0) .



Hình 3.7: Ví dụ về mạng Petri.

Một trạng thái (marking) của mạng Petri là một sự phân bố các *tokens* tại các vị trí. Số token và vị trí của các token có thể thay đổi trong quá trình biến đổi của mạng. Một trạng thái thường được ký hiệu là M , là một m -vector, với m là số các vị trí. Thành phần thứ p của M , ký hiệu là $M(p)$, là số token tại vị trí p . Mạng Petri được gọi là bảo toàn (*preserve*) nếu tổng số token không thay đổi trong suốt quá trình biến đổi của mạng. Một cặp (p, t) gọi là *tự lặp* (*self-loop*) hay *khuyên*, nếu p vừa là vị trí vào vừa là vị trí ra của chuyển t . Mạng Petri gọi là *thuần túy* (*pure*) nếu nó không có khuyên.

Một mạng Petri được mô tả bằng một đa đồ thị (có hướng) hai phần. Các vị trí thường được biểu diễn bởi các hình tròn, còn các chuyển thường được biểu diễn bằng các hình chữ nhật hay hình vuông. Cung đi từ vị trí p_j đến cái chuyển t_i xác định p_j là vị trí vào của của t_i , ký hiệu là $I(p_j, t_i) = 1$. Cung đi từ cái chuyển t_i đến vị trí p_j xác định p_j là vị trí ra của cái chuyển t_i , ký hiệu là $O(t_i, p_j) = 1$. Nếu $I(p_j, t_i) = k$ (hoặc $O(t_i, p_j) = k$), thì có k cung nối vị trí p_j với chuyển t_i (hoặc nối cái chuyển t_i với vị trí p_j). Số token tại mỗi vị trí được biểu diễn bằng các số nguyên hoặc các dấu chấm trong hình tròn.

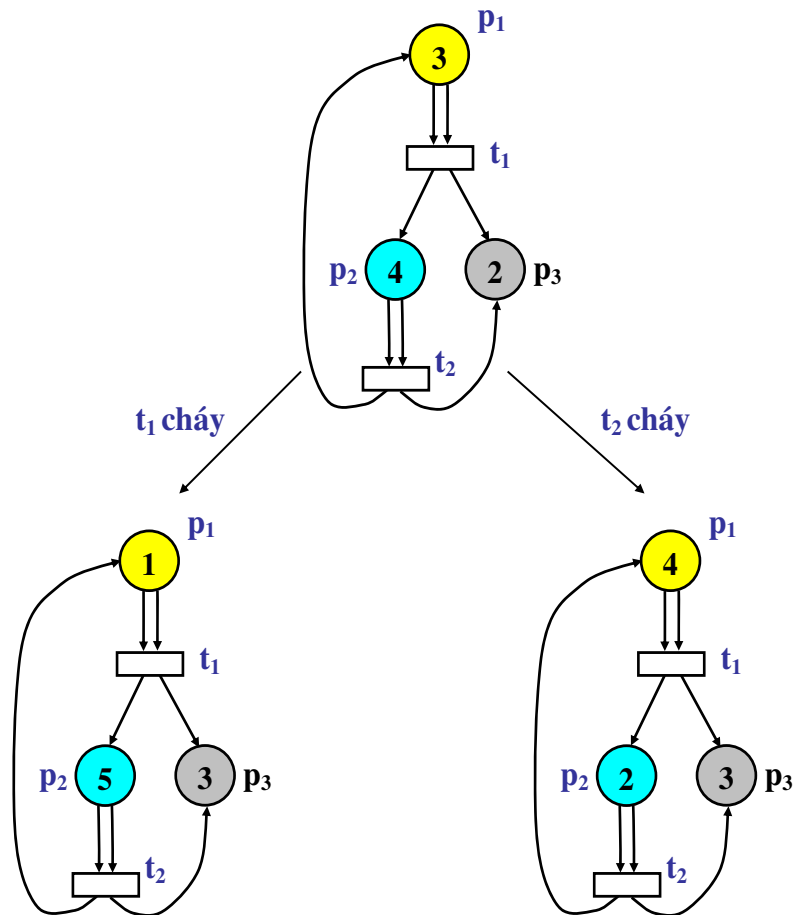
Một cái chuyển (transition) t gọi là *cháy được* (*firable*) hay *có khả năng kích*

hoạt (enabled) tại trạng thái M nếu mỗi vị trí vào p của t chứa ít nhất số token không nhỏ hơn trọng số của cung nối p với t , tức là, $M(p) \geq I(p, t)$ với mọi $p \in P$.

Khi t cháy số token ở vị trí vào p_1 sẽ mất đi đúng bằng trọng số cung nối p_1 với t và nó gửi đến các vị trí ra p_2 số token đúng bằng trọng số cung nối t với p_2 .

Nói cách khác khi chuyển t cháy tại trạng thái M sẽ dẫn đến trạng thái M' như sau:

$$M'(p) = M(p) - I(p, t) + O(t, p) \text{ với mọi } p \in P.$$



Hình 3.8: Quá trình chuyển trạng thái sau một bước.

Cho một mạng Petri bị chặn (N, M_0) , từ trạng thái M_0 ban đầu, ta có thể nhận được các trạng thái tiếp theo bằng cách cho cháy các chuyển (cháy được). Từ mỗi trạng thái mới này, lại tiếp tục nhận được các trạng thái mới khác. Để biểu diễn

quá trình này ta dùng khái niệm đồ thị (có hướng) trạng thái đạt được (*reachability graph*), hay gọi tắt là *đồ thị đạt được*. Mỗi đỉnh là một trạng thái đạt được từ trạng thái ban đầu, mỗi cung tương ứng với một cái chuyển cháy chuyển từ trạng thái này sang trạng thái khác. Ta ký hiệu đồ thị đạt được của mạng Petri (N, M_0) là (\mathcal{G}, M_0) .

3.4 Mỗi quan hệ giữa hệ động lực CFGs và mạng Petri

3.4.1 CFG và mạng Petri

Trong phần này chúng tôi chứng minh rằng hệ động lực CFG là một trường hợp đặc biệt của mạng Petri.

Cho hệ CFG trên đa đồ thị có hướng $G = (V, E)$. Ta sẽ xây dựng mạng Petri N sao cho với mọi trạng thái ban đầu \mathcal{O} với n chips của $CFG(G)$, tồn tại một trạng thái ban đầu M_0 sao cho đồ thị đạt được $R(N, M_0)$ của mạng Petri N đẳng cấu với đồ thị trạng thái của $CFG(G, n, \mathcal{O})$.

Định nghĩa 3.4.1. Ta định nghĩa ánh xạ ϕ từ tập các CFG vào tập các mạng Petri như sau. Với một (đa) đồ thị có hướng G , $\phi(CFG(G))$ là mạng Petri $(N) = (P, T, I, O)$, trong đó:

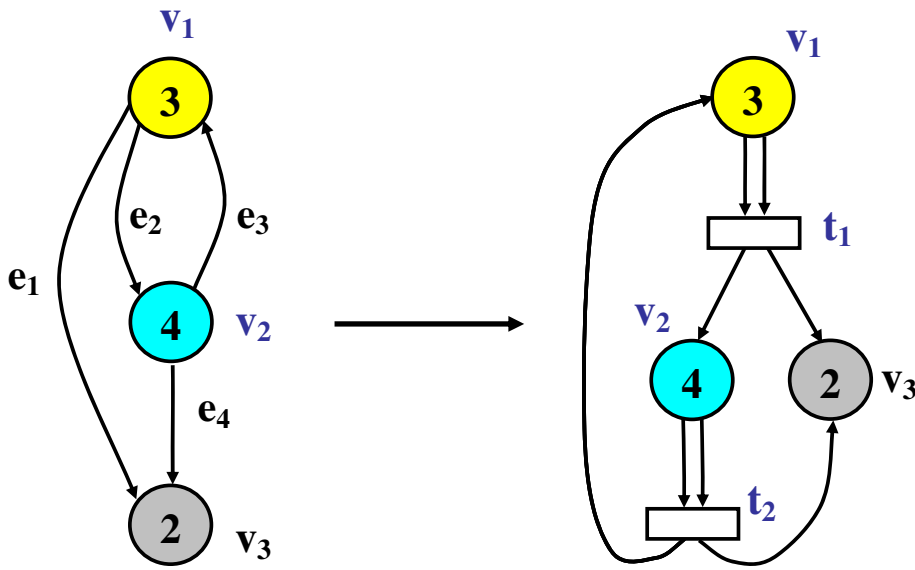
- Tập các vị trí là tập đỉnh của G : $P = V$

- Tập các chuyển: với mỗi $v \in V$ có $deg^+(v) > 0$, ta định nghĩa cái chuyển $t(v)$ thỏa mãn hai điều kiện:

(i) v là vị trí vào duy nhất của $t(v)$ và $I(v, t(v)) = deg^+(v)$,

(ii) các vị trí ra u của $t(v)$ là các đỉnh $u \in V$ sao cho $(v, u) \in E$, và $O(t(v), u) = w(v, u)$, ở đây $w(v, u)$ là trọng số cung (v, u) trong G .

- Tương ứng trạng thái: với mỗi trạng thái a trong $CFG(G)$ xác định trạng thái $M = \phi(a)$ trong N sao cho $M(v) = a(v)$, với mọi $v \in V$.



Hình 3.9: CFG và mạng Petri tương ứng

Bây giờ, ta xét đồ thị trạng thái của $CFG(G)$ và mạng Petri N tương ứng.

Định lý 3.4.2. Cho G là (đa) đồ thị có hướng, cho \mathcal{O} trạng thái đầu. Gọi N là mạng Petri nhận được từ $CFG(G)$ bởi ánh xạ ϕ , M_0 trạng thái của N tương ứng với \mathcal{O} . Khi đó mạng Petri (N, M_0) và hệ động lực $CFG(G, n, \mathcal{O})$ có cùng đồ thị đạt được.

Chứng minh. Trước hết ta chứng minh việc cháy đỉnh v trong CFG tương ứng với việc kích hoạt cái chuyển $t(v)$ trong mạng Petri. Thật vậy, gọi a là một trạng thái của $CFG(G, n, \mathcal{O})$ và v là một đỉnh bất kỳ của V . Đỉnh v cháy được tại trạng thái a khi và chỉ khi số chip tại v không nhỏ hơn bậc ra của v . Điều này tương đương với $I(v, t(v)) \leq M(v)$ và do đó cái chuyển $t(v)$ là kích hoạt được. Khi đỉnh v cháy, số chip tại v giảm đi $deg^+(v)$ chips và số chip tại các đỉnh u lân cận của v tăng lên $w(v, u)$ chips. Một cách tương đương, khi kích hoạt cái chuyển $t(v)$ trong mạng Petri N , số token tại v giảm đi $I(v, t(v) = deg^+(v)$ tokens. Mặt khác, với mỗi lân cận u của v , vị trí ra của $t(v)$ tăng lên $w(v, u)$ tokens.

Hơn nữa, cấu hình ban đầu M_0 của mạng Petri cũng chính là trạng thái ban

đầu \mathcal{O} của $CFG(G, n)$. Do đó, CFG và mạng Petri tương ứng có cùng đồ thị đạt được. \square

Bởi vì tổng số chip trong hệ động lực CFG không thay đổi trong suốt quá trình biến đổi của hệ nên từ định lý trên ta có ngay hệ quả sau

Hệ quả 3.4.3. *Hệ động lực CFG là mạng Petri bảo toàn.*

3.4.2 CCFG và mạng Petri

Trong phần này chúng tôi trình bày mối quan hệ giữa CCFG và mạng Petri. Cho $CCFG(G)$ trên (đa) đồ thị có hướng $G = (V, E)$. Ta định nghĩa ánh xạ ψ từ tập các CCFG vào tập các mạng Petri tương tự như trường hợp của CFG.

Định nghĩa 3.4.4. Cho (đa) đồ thị có hướng G , $\psi(CFG(G))$ là mạng Petri $(N) = (P, T, I, O)$, trong đó:

- Tập hợp các vị trí là tập đỉnh của đồ thị $P = V$
- Tập hợp các chuyển: Với mỗi cung $e = (u, v) \in E$, ta định nghĩa cái chuyển (transition) $t(e)$ thỏa mãn hai điều kiện sau:

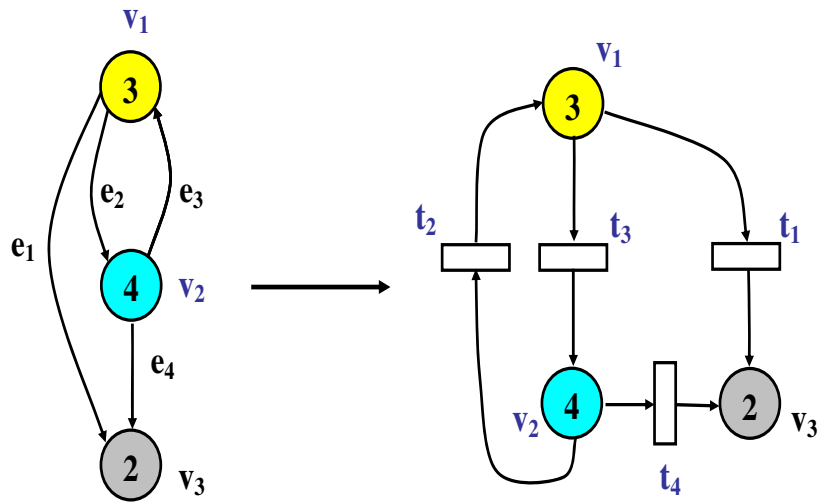
- (i) u là vị trí vào duy nhất của t và $I(u, t(e)) = 1$,
- (ii) v là vị trí ra duy nhất của t và $O(t(e), v) = 1$.

- Tương ứng trạng thái: với mỗi trạng thái a trong $CFG(G)$ xác định trạng thái $M = \phi(a)$ trong N sao cho $M(v) = a(v)$, với mọi $v \in V$.

Trong mạng Petri này ta có $|T| = |E|, |P| = |V|$.

Chứng minh tương tự như Định lý 3.4.2, ta có kết quả sau:

Định lý 3.4.5. *Cho G là (đa) đồ thị có hướng, cho \mathcal{O} là trạng thái ban đầu của $CCFG(G)$. Gọi N là mạng Petri nhận được từ $CCFG(G)$ bởi ánh xạ ψ , M_0 là cấu hình của N ứng với trạng thái \mathcal{O} . Khi đó, mạng Petri (N, M_0) và hệ $CCFG(G, n, \mathcal{O})$ có cùng đồ thị đạt được.*



Hình 3.10: CCFG và mạng Petri tương ứng

3.4.3 CFG tô màu và mạng Petri

Vấn đề phức tạp nhất là xây dựng phép nhúng từ lớp các CFG tô màu vào lớp các mạng Petri bởi vì sự khác biệt về cấu trúc giữa CFG hay CCFG và CFG tô màu. Hành vi chuyển trạng thái của CFG tô màu phức tạp hơn, gồm có việc mở các đỉnh và cháy như trong CFG cổ điển. Một khó khăn nữa là trong CFG tô màu, số chip tại mỗi đỉnh có các màu khác nhau, và chip màu nào thì chỉ được di chuyển theo cung đi ra có cùng màu. Việc xây dựng mạng Petri tương ứng với CFG tô màu không còn đơn giản như CFG cổ điển hay CCFG nữa. Chúng tôi, sử dụng ý tưởng liên hệ giữa cấu trúc của CFG tô màu so với CFG cổ điển để xây dựng mạng Petri tương ứng. CFG tô màu là hợp của các CFG nên mạng Petri tương ứng có cấu trúc khá phức tạp.

Cho $ColCFG(G)$ là CFG tô màu như trong định nghĩa 3.1.9 trên đồ thị tô màu $G = (V, E, X, col)$. Gọi m là số các màu. Với $u, v \in V$, $c \in X$, gọi $d(v, c)$ là số cung có màu c đi ra từ v , $d((v, u), c)$ là số cung màu c đi từ v đến u . Gọi N là số tự nhiên đủ lớn (lớn hơn tổng số chip). Mạng Petri được xây dựng như sau:

Định nghĩa 3.4.6. Ta định nghĩa ánh xạ χ từ tập các CFG tô màu vào tập các mạng

Petri như sau: Cho $ColCFG(G)$ CFG tô màu trên đồ thị tô màu $G = (V, E, X, col)$. $\chi(ColCFG(G))$ là mạng Petri $(N(G)) = (P, T, I, O)$, trong đó các thành phần được xây dựng như sau:

- Tập các vị trí P : với mỗi $v \in V$, có $2m + 1$ vị trí tương ứng của P bao gồm m vị trí $p(v, c)$ (mỗi vị trí cho một màu c), m vị trí $q(v, c)$ (mỗi vị trí cho một màu c) và một vị trí $r(v)$.
- Tập các chuyển T : với mỗi $v \in V$, có $2m + 1$ cái chuyển của T bao gồm m cái chuyển $t(v, c)$ (mỗi cái chuyển cho một màu c), m cái chuyển $f(v, c)$ (mỗi cái chuyển cho một màu c) và một cái chuyển (transiton) $\theta(v)$.
- Hàm vào (input function) I : với mỗi $v \in V, c \in C$, $I(p(v, c), t(v, c)) = \mathbf{N} + d(v, c)$, $I(r(v), \theta(v)) = \mathbf{N} + 1$, và $I(q(v, c), f(v, c)) = \mathbf{N} + d(v, c)$. Tất cả các giá trị còn lại bằng 0.
- Hàm ra (output function) O : với mỗi $v \in V, c \in C$, $O(t(v, c), r(c)) = 1$, $O(\theta(v), q(v, c)) = \mathbf{N}$, $O(f(v, c), q(v, c)) = \mathbf{N}$, và với mỗi lân cận u của v , $O(f(v, c), p(u, c)) = d((v, u), c)$ và $O(f(v, c), q(u, c)) = d((v, u), c)$. Các giá trị còn lại bằng 0.
- Vai trò của \mathbf{N} : các toán tử θ có thể thực hiện khi và chỉ khi các toán tử f không thực hiện được.

Để hiểu rõ cấu trúc phức tạp của mạng Petri này, ta có một số lý giải về ý tưởng xây dựng cấu trúc này như sau: để nhận được mạng Petri này, với mỗi đỉnh v của CFG, ta xây dựng mạng Petri tương ứng với m đỉnh để tách chip và cung có màu khác nhau. Các toán tử của CFG tô màu $ColCFG$ được mã hóa như sau: việc mở đỉnh v được mã hóa thành cái chuyển $\theta(v)$ và việc cháy đỉnh v với màu c được mã hóa thành cái chuyển $f(v, c)$. Điều kiện "tồn tại màu c sao cho v cháy được" được mã hóa thành cái chuyển $t(v, c)$. Vị trí $p(v, c)$ chứa số chip màu c tại đỉnh v trước khi mở đỉnh v , vị trí $q(v, c)$ chứa số chip màu c tại v sau khi mở đỉnh v và vị trí

$r(v)$ chứa số chip đảm bảo cho điều kiện mở được hay không mở được đỉnh v .

Để chứng minh rằng CFG tô màu và mạng Petri có cùng đồ thị đạt được, tiếp theo ta chỉ ra sự tương ứng giữa các trạng thái của CFG tô màu $ColCFG(G)$ với các cấu hình của mạng Petri $N(G)$.

Định nghĩa 3.4.7. Cho $ColCFG(G)$ là CFG tô màu, δ là một trạng thái của nó (ở đây $\delta_c(v)$ là số chip có màu c tại đỉnh v), gọi $N(G)$ là mạng Petri tương ứng CFG tô màu $ColCFG(G)$. Ta định nghĩa cấu hình M tương ứng với δ trong mạng Petri như sau:

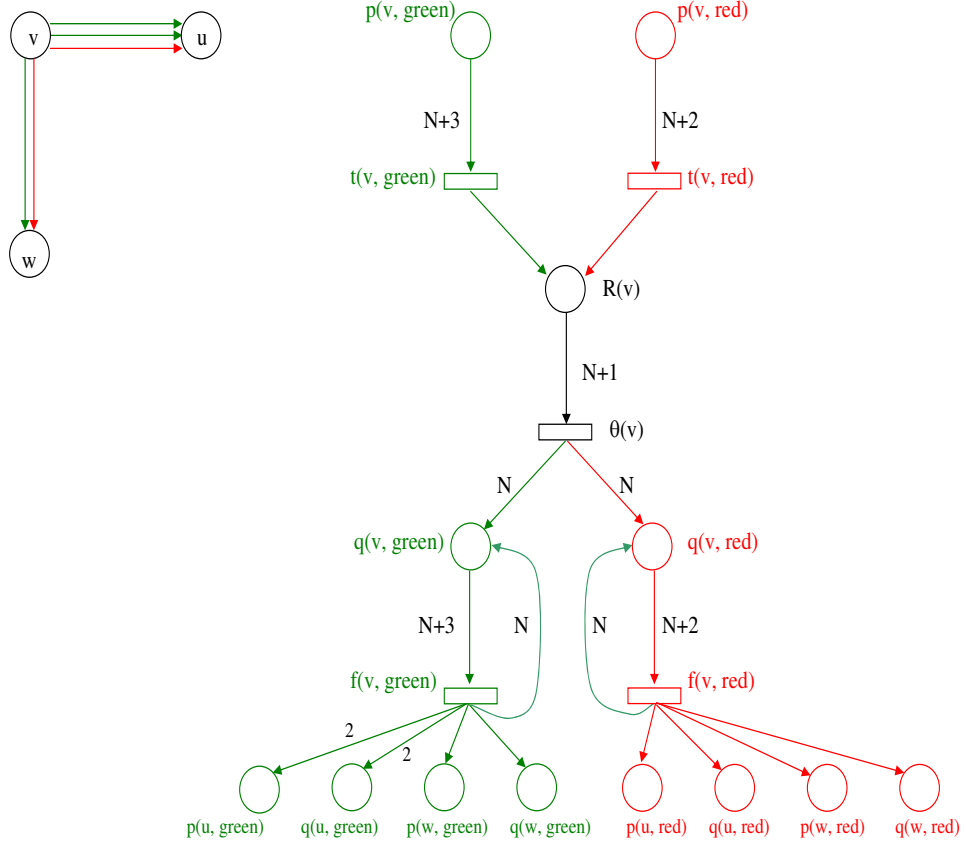
- Số token ở $p(v, c)$ là $\mathbf{N} + \delta_c(v)$.
- Số token ở $r(v)$ là \mathbf{N} .
- Số token ở $q(v, c)$ là $\delta_c(v)$.
- Mỗi cấu hình M được đồng nhất với tập giá trị $\{q(v, c), c \in X\}$.

Kết quả chính của chương này được phát biểu như sau:

Định lý 3.4.8. Cho CFG tô màu $ColCFG(G)$, δ là trạng thái ban đầu. Gọi $N(G)$ là mạng Petri nhận được từ $ColCFG(G)$ qua ánh xạ χ , M_0 là cấu hình của mạng Petri $N(G)$ tương ứng với δ . Khi đó mạng Petri $(N(G), M_0)$ và mô hình $ColCFG(G, \delta)$ có cùng đồ thị đạt được.

Chứng minh. Trước hết, ta chứng minh rằng mỗi toán tử của $ColCFG(G)$ tương ứng với một chuyển của $N(G)$. Ta sẽ chứng minh rằng việc mở đỉnh v được mã hóa thành cái chuyển $\theta(v)$ và việc cháy đỉnh v với màu c được mã hóa thành cái chuyển $f(v, c)$.

- Đầu tiên, tất cả các đỉnh của V đóng. Tương ứng trong N , tất cả các cái chuyển $\theta(v)$ không thể kích hoạt được vì $I(r(v), \theta(v)) = \mathbf{N} + 1$ và chỉ có \mathbf{N} tokens ở vị trí $r(v)$.



Hình 3.11: CFG tô màu và mạng Petri tương ứng.

Hơn nữa, trước khi kích hoạt cái chuyển $\theta(v)$, các cái chuyển $f(v, c)$ cũng không thể kích hoạt được vì $I(q(v, c), f(v, c)) = \mathbf{N} + d(v, c)$ và chỉ có $\delta_c(v)$ tokens ở vị trí $q(v, c)$ nhỏ hơn giá trị \mathbf{N} .

- Điều kiện mở đỉnh v trong CFG tô màu là: "tồn tại màu c sao cho v cháy được". Tương ứng trong mạng Petri N , điều kiện để kích hoạt được cái chuyển $\theta(v)$ là $I(r(v)) \geq I(r(v), \theta(v)) = \mathbf{N} + 1$. Như vậy, lúc đầu ở vị trí $r(v)$ có \mathbf{N} tokens, ta có thể kích hoạt $\theta(v)$ sau khi vị trí $r(v)$ nhận được ít nhất một token. Có m cái chuyển của vị trí ra $r(v)$ là $t(v, c)$, do đó $r(v)$ nhận được ít nhất một token nếu có ít nhất một cái chuyển $t(v, c)$ được kích hoạt. Điều kiện kích hoạt cái chuyển $t(v, c)$ là số token tại $p(v, c)$ không nhỏ hơn $I(p(v, c), t(v, c))$, tức là $\mathbf{N} + \delta_c(v) \geq \mathbf{N} + d(v, c)$, hay một cách tương đương $\delta_c(v) \geq d(v, c)$.

Đây cũng chính là điều kiện mở đỉnh v với màu c . Do vậy điều kiện mở đỉnh v tương đương với điều kiện kích hoạt cái chuyển $\theta(v)$.

- Hành vi mở đỉnh v trong CFG tô màu gồm có hai bước
 - thứ nhất là đánh dấu v đã được mở
 - thứ hai là với màu $c \in X$, xét CFG với các cung màu c và với các đỉnh đã được mở, cho CFG vận hành đến khi đạt đến trạng thái cuối cùng.

Tương ứng với hành vi đó trong mạng Petri N , ta xét đỉnh v và màu c_0 sao cho $\delta_{c_0}(v) \geq d(v, c_0)$. Transition $t(v, c_0)$ là kích hoạt được. Bây giờ xét màu c , nếu $\delta_c(v) \geq d(v, c)$ thì $t(v, c)$ cũng kích hoạt được và kéo theo cái chuyển $\theta(v)$ kích hoạt được. Với màu c , $q(v, c) = \delta_c(v) + \mathbf{N}$, và $I(q(v, c), f(v, c)) = \mathbf{N} + d(v, c)$. Rõ ràng $q(v, c_0) \geq I(q(v, c_0), f(v, c_0))$ do đó $f(v, c_0)$ kích hoạt được. Với màu c khác nếu thỏa mãn $\delta_c(v) \geq d(v, c)$ thì $q(v, c) \geq I(q(v, c), f(v, c))$, và $f(v, c)$ cũng được kích hoạt.

Các tokens từ vị trí $q(v, c)$ gửi vào các vị trí $p(u, c)$ và $q(u, c)$ với u là lân cận của v . Nếu u vẫn còn đóng thì cái chuyển $f(u, c)$ không kích hoạt được, tức là u không cháy được, tuy nhiên ta có thể thay đổi điều kiện để u cháy được bởi vì số token tại vị trí $p(u, c)$ được tăng lên như trong CFG cổ điển. Ngược lại, u đã được mở và $q(u, c)$ có thể cháy như trong CFG cổ điển.

- Tiếp theo ta xét tính chất: trong CFG tô màu $ColCFG(G)$, mỗi đỉnh v chỉ được mở nhiều nhất một lần.

Trong mạng Petri N tương ứng, nếu v đã được mở CFG tô màu, thì cái chuyển $\theta(v)$ trong mạng N đã được kích hoạt và $r(v)$ mất đi \mathbf{N} tokens. Trong khi đó số token mà vị trí $r(v)$ nhận được từ các chuyển $t(v, c)$ trong suốt quá trình biến đổi của mạng không thể vượt quá \mathbf{N} (mỗi bước biến đổi trạng thái $r(v)$ chỉ nhận được một token từ $t(v, c)$). Do đó, với điều kiện $I(r(v)) \geq I(r(v), \theta(v)) = \mathbf{N} + 1$, $\theta(v)$ không thể được kích hoạt thêm lần nữa. Như vậy, điều kiện mỗi đỉnh v chỉ được mở nhiều nhất một lần trong CFG tô

màu tương đương với điều kiện $\theta(v)$ chỉ kích hoạt nhiều nhất một lần trong mạng Petri tương ứng.

- Trong CFG tô màu $ColCFG(G)$, khi mở một đỉnh, ta cho CFG vận hành đến khi đạt được trạng thái cuối cùng. Điều kiện này tương đương với điều kiện trong mạng Petri: các chuyển θ kích hoạt được khi và chỉ các chuyển f không thể kích hoạt.

Cuối cùng, ta chứng minh rằng mỗi trạng thái trong CFG tô màu $ColCFG(G)$ tương ứng với một cấu hình trong mạng Petri $N(G)$. Thật vậy, theo định nghĩa 3.4.7, mỗi cấu hình của $N(G)$ được đồng nhất với tập hợp $\{q(v, c)\}$. Đầu tiên, có $\delta_c(v)$ tokens ở vị trí $q(v, c)$, do đó tập hợp $\{q(v, c)\}$ chính là số chip tại mỗi đỉnh v của V .

Sau khi biến đổi trạng thái, chẳng hạn đỉnh v cháy với màu c trong CFG thì v mất đi $d(v, c)$ chips. Trong mạng Petri $N(G)$, cái chuyển $f(v, c)$ được kích hoạt, và $I(q(v, c), f(v, c)) = d(v, c)$, nên $q(v, c)$ mất đi $d(v, c)$ tokens. Với mỗi lân cận u của v trong đồ thị G , khi v cháy sẽ thêm $d((v, u), c)$ chips màu c vào đỉnh u . Tương ứng trong mạng Petri N , ta có $O(f(v, c), q(u, c)) = d((v, u), c)$, tức là sau khi kích hoạt cái chuyển $f(v, c)$, vị trí $q(u, c)$ được thêm $d((v, u), c)$ tokens màu c .

Do đó mỗi trạng thái của CFG tô màu $ColCFG(G)$ tương đương với mỗi cấu hình của mạng Petri $N(G)$.

Như vậy mỗi toán tử của CFG tô màu $ColCFG(G)$ tương ứng với một cái chuyển của mạng Petri $N(G)$, và mỗi trạng thái của $ColCFG(G)$ tương ứng với mỗi cấu hình của $N(G)$, nên đồ thị đạt được của hai mô hình này đẳng cấu với nhau. \square

3.5 Kết luận chương 3

Trong chương này chúng tôi đã thu được kết quả sau:

Chứng minh các hệ động lực CFG là các mạng Petri đặc biệt.

Chương 4

Tính đạt được của hệ CCFG trên đồ thị có hướng

Bài toán đạt được (reachability) là một bài toán cơ bản khi nghiên cứu các hệ động lực. Có nhiều dạng bài toán đạt được của hệ động lực. Trong chương này chúng tôi nghiên cứu bài toán điển hình nhất được phát biểu như sau: cho trước hai trạng thái a và b của hệ động lực CCFG, hãy xác định xem b có nhận được từ a sau một số lần áp dụng luật cháy? Trước hết, chúng tôi xét hệ CCFG trên đồ thị có hướng không chu trình. Tập đỉnh V của đồ thị có hướng không chu trình như vậy được trang bị một thứ tự bộ phận. Dựa vào cấu trúc này chúng tôi xét các lọc của tập đỉnh V và đưa ra khái niệm họ năng lượng của các trạng thái của CCFG. Từ đó, chúng tôi đặc trưng tính đạt được của CCFG bằng cách sử dụng họ năng lượng này. Vì tập đỉnh V có cấu trúc thứ tự nên không gian trạng thái của CCFG cũng là một tập sắp thứ tự bộ phận. Chúng tôi cũng đưa ra thuật toán (gọi là thuật toán A) xác định thứ tự của các trạng thái của CCFG dựa vào họ năng lượng của các trạng thái (trong phần 4.2). Tuy nhiên, vì phải xét hết tất cả các lọc nên trong trường hợp xấu nhất, thuật toán A chạy trong thời gian hàm mũ của $|V|$. Thuật toán này có hiệu quả khi phải so sánh nhiều cặp trạng thái cùng một lúc của cùng một hệ CCFG và nhất là trong trường hợp đồ thị có kích cỡ nhỏ. Trong trường hợp tổng quát hơn, khi xét hệ CCFG trên đồ thị có hướng (có thể có chu trình), để đặc trưng tính đạt được của CCFG chúng tôi đưa ra khái niệm mạng vận tải tương ứng với các trạng thái

của CCFG (trong phần 4.4). Dựa vào khái niệm này, chúng tôi đặc trưng tính đạt được của CCFG bằng cách tính toán giá trị của luồng qua mạng vận tải tương ứng (trong phần 4.5). Từ đây, sử dụng các thuật toán xác định luồng cực đại, chúng tôi đưa ra thuật toán (B) với thời gian đa thức của $|V|$ để xác định tính đạt được của CCFG. Khi xét các hệ CCFG trên đồ thị nền lớn thì thuật toán B là một cách khắc phục nhược điểm của thuật toán A. Khi xét các hệ CCFG trên đồ thị nền nhỏ và cần phải so sánh nhiều cặp trạng thái cùng lúc thì ta sử dụng thuật toán A. Trong thực tế, thông thường các bài toán được xét trên đồ thị nền cố định và phải so sánh nhiều cặp trạng thái, ví dụ nếu ta muốn xây dựng không gian trạng thái với cấu trúc thứ tự, nên thuật toán A thường được sử dụng hơn vì chỉ cần chạy thuật toán xác định các lọc một lần và sau đó so sánh được nhiều cặp trạng thái. Trong khi đó, muốn sử dụng thuật toán B, thì mỗi lần so sánh một cặp trạng thái ta phải xác định mạng vận tải tương ứng với trạng thái lệch của cặp trạng thái đó. Đây là một trong những nhược điểm của thuật toán B và cũng là lý do để sử dụng thuật toán A thuận tiện hơn. Trước hết, trong phần 4.1, để thấy được độ phức tạp của bài toán đạt được của các hệ động lực CFG, chúng tôi nhắc lại độ phức tạp của bài toán đạt được của một số mạng Petri.

4.1 Tính đạt được của một số mạng Petri

Trong chương trước chúng tôi đã chứng minh rằng các hệ động lực CFG là các mạng Petri đặc biệt. Hệ CCFG là một máy hữu hạn trạng thái (*finite state machine*) tức là mạng Petri mà mỗi transition có đúng một vị trí vào và đúng một vị trí ra. Bài toán đạt được của mạng Petri là cho trước mạng Petri $(N; M_0)$ và một cấu hình M của N , hãy quyết định xem M có đạt được từ M_0 hay không? Bài toán này được phát biểu đầu tiên bởi Keller [46] và nó trở nên quan trọng vì có nhiều bài toán khác tương đương đệ quy với bài toán này. Độ phức tạp của bài toán đạt được của mạng Petri vẫn là bài toán mở trong nhiều năm trở lại đây. Tuy nhiên, nó cũng

đã được giải quyết cho một số mạng Petri đặc biệt [15, 38, 45, 13]. Trong phần này chúng tôi nhắc lại một số mạng Petri đặc biệt và độ phức tạp của bài toán đạt được tương ứng. Trước hết, chúng tôi định nghĩa một số mạng Petri đặc biệt:

Định nghĩa 4.1.1. Mạng Petri (N, M_0) được gọi là

- *không tương tranh* (conflict-free) nếu mỗi vị trí có nhiều nhất một cái chuyển ra (output transition),
- *không hút* (sinkless) nếu mọi chu trình của N không làm mất hết token trong quá trình biến đổi,
- *chuẩn tắc* (normal) nếu các cái chuyển vào (input transitions) cũng là các cái chuyển ra (output transitions) của các vị trí trong các chu trình của N ,
- *mạng BPP* (Basic Parallel Process) nếu mọi cái chuyển đều có đúng một vị trí vào,
- *mark graph* nếu mọi vị trí của nó có đúng một cái chuyển vào (input transition) và đúng một cái chuyển ra (output transition),
- *k-bị chặn* nếu với mọi trạng thái đạt được M và mọi vị trí p ta có $M(p) \leq k$,
- *bị chặn* nếu tập trạng thái đạt được là hữu hạn.

Bài toán đạt được của mạng Petri là quyết định được và độ phức tạp thời gian của nó đã được giải quyết trong một số trường hợp đặc biệt được nêu trong mệnh đề sau:

Mệnh đề 4.1.2. Bài toán đạt được của mạng Petri (N, M_0) là

- NP-dầy đủ nếu N không có chu trình [77],
- NP-dầy đủ nếu N là không hút và chuẩn tắc [39],
- NP-dầy đủ nếu N là không tương tranh,

- NP-đầy đủ nếu N là mạng BPP [41],
- thuộc P nếu N là không tương tranh và bị chặn [37],
- thuộc P nếu N là mark graph.

Các kết quả đã biết về bài toán đạt được của mạng Petri là giải được trong thời gian đa thức nếu nó không tương tranh và bị chặn hoặc là mark graph. Đối với mạng Petri, bài toán đạt được của máy hữu hạn trạng thái đã được giải quyết trong [65] bằng phương pháp phương trình ma trận, sử dụng ma trận tác động (incidence matrix) và phương trình trạng thái. Với một máy hữu hạn trạng thái liên thông, trạng thái M là đạt được từ trạng thái M_0 khi và chỉ khi phương trình $M = M_0 + Ax$ có nghiệm nguyên không âm x , ở đây A là ma trận tác động. Bài toán này có thể được kiểm chứng trong thời gian đa thức, tức là thuộc NP. Một mô hình mở rộng của máy hữu hạn trạng thái là máy hữu hạn trạng thái với bố các cung (state machines with batch processing arcs) [63], bài toán đạt được của máy này có độ phức tạp là NP-khó. Như vậy, có thể khẳng định rằng cho đến nay chưa có thuật toán cụ thể nào để giải bài toán đạt được của máy hữu hạn trạng thái hay của CCFG ngoài việc sử dụng phương trình ma trận. Trong chương này, trước hết chúng tôi xét CCFG trên đồ thị có hướng không chu trình. Chúng tôi sẽ đưa ra đặc trưng tính đạt được của CCFG mà trong trường hợp này chính là cấu trúc thứ tự của CCFG và thuật toán xác định thứ tự này. Sau đó, trên đồ thị có hướng tổng quát chúng tôi sử dụng thuật toán Push-Relabel, một biến thể của thuật toán Fork-Fukerson để giải bài toán đạt được của CCFG trong thời gian đa thức.

4.2 Cấu trúc thứ tự của CCFG trên DAG

Để đặc trưng cấu trúc thứ tự của không gian trạng thái của CCFG, trong phần này chúng tôi đưa ra định nghĩa họ năng lượng của các trạng thái của CCFG. Để ý rằng nếu $G = (V, E)$ là DAG thì tập đỉnh V là một tập sắp thứ tự như sau: với mọi

$u, v \in V$, $u \leq v \Leftrightarrow u = v$ hoặc tồn tại một đường đi từ v đến u . Ký hiệu $\mathcal{F}(V)$ là tập tất cả các lọc thứ tự của V .

Định nghĩa 4.2.1. Cho $G = (V, E)$ là một đồ thị có hướng không chu trình (DAG), $a = (a_1, a_2, \dots, a_{|V|})$ là một hợp thành của n trên V . Năng lượng $e(A, a)$ của trạng thái a trên tập $A \subseteq V$ được định nghĩa là $e(A, a) = \sum_{i \in A} a_i$. Dãy số $(e(A, a)_{A \in \mathcal{F}(V)})$ được gọi là họ năng lượng của trạng thái a và đại lượng $\mathcal{E}(a) = \sum_{A \in \mathcal{F}(V)} e(A, a)$ được gọi là năng lượng tổng của trạng thái a .

Từ định nghĩa trên ta có ngay mệnh đề sau

Bổ đề 4.2.2. Các trạng thái của CCFG được xác định duy nhất bởi họ năng lượng của nó. Nghĩa là, nếu a và b là hai trạng thái của $CCFG(G, n)$ có cùng họ năng lượng thì $a = b$.

Chứng minh. Lấy tùy ý $v \in V$. Ta có $Pred(v)$ và $Pred(v) \setminus \{v\}$ là các lọc của V nên theo giả thiết ta có

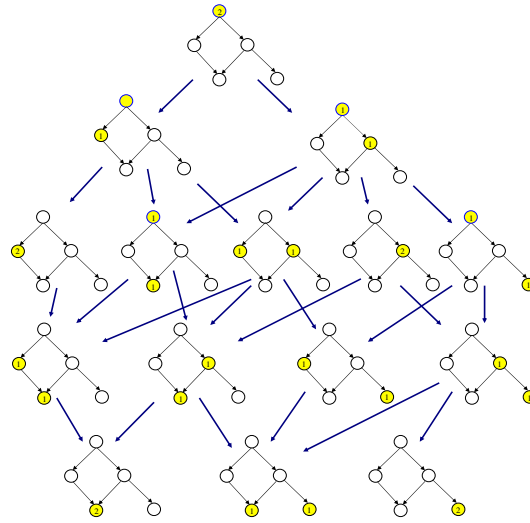
$$e(Pred(v), a) = e(Pred(v), b) \text{ và } e(Pred(v) \setminus \{v\}, a) = e(Pred(v) \setminus \{v\}, b)$$

Từ đó suy ra $a(v) = b(v)$. □

Sau đây, ta chứng minh rằng không gian trạng thái của CCFG có cấu trúc thứ tự bộ phận.

Bổ đề 4.2.3. $(CCFG(G, n), \leq)$ là một tập được sắp thứ tự bộ phận.

Chứng minh. Ta sẽ chứng minh rằng $b \leq a$ kéo theo $e(A, b) \leq e(A, a)$ với mọi $A \in \mathcal{F}(V)$, tức là quá trình cháy của hệ làm cho năng lượng của các trạng thái giảm thực sự: nếu $b < a$ thì $\mathcal{E}(b) < \mathcal{E}(a)$. Ta chỉ cần chứng minh cho trường hợp $a \rightarrow b$. Giả sử rằng b nhận được từ a bằng cách chuyển một chip từ u đến v . Khi đó $a(u) - 1 = b(u)$, $a(v) + 1 = b(v)$ và $a(w) = b(w)$ với mọi $w \neq u, v$, ở đây $a(u)$ là số chip tại đỉnh u của trạng thái a . Lấy tùy ý $A \in \mathcal{F}(V)$.



Hình 4.1: Không gian trạng thái của một CCFG với 2 chips.

+ Nếu $v \in A$ thì $u \in A$ do $u \geq v$ trong (V, \leq) và A là một lọc. Do đó $e(A, a) = e(A, b)$.

+ Nếu $v \notin A$ thì

$$e(A, a) = \begin{cases} e(A, b) & \text{nếu } u \notin A \\ e(A, b) + 1 & \text{nếu } u \in A \end{cases}$$

Do đó $e(A, a) \geq e(A, b)$ với mọi $A \in \mathcal{F}(V)$. Nói riêng $e(\text{Pred}(u), a) > e(\text{Pred}(u), b)$.

Từ đó suy ra $b \leq a$ kéo theo $\mathcal{E}(b) \leq \mathcal{E}(a)$. Hơn nữa, nếu $b < a$ thì $\mathcal{E}(b) < \mathcal{E}(a)$. Vậy $(CCFG(G, n), \leq)$ là một tập được sắp thứ tự bộ phận. \square

Như vậy, không gian trạng thái của CCFG là một tập được sắp thứ tự bộ phận. Vấn đề đặt ra là làm thế nào để đặc trưng được quan hệ thứ tự này? Làm thế nào để xác định trạng thái a có dẫn đến được trạng thái b hay không? Bài toán đặt được của một hệ động lực luôn là bài toán khó. Trong mô hình CFG cổ điển, cấu trúc thứ tự được đặc trưng bởi khái niệm shot-vector. Tuy nhiên, đây là một đặc trưng "gián tiếp", chỉ so sánh được hai trạng thái a và b khi chúng cùng xuất phát từ một trạng thái ban đầu. Bài toán so sánh hai trạng thái bất kỳ trong hệ động lực CFG vẫn còn là bài toán mở. Khái niệm shot-vector có được là do cấu trúc phân bậc

của không gian trạng thái của CFG, theo nghĩa giữa hai trạng thái $a \geq b$ có thể có nhiều đường đi nhưng độ dài các đường đi ấy luôn bằng nhau. Không gian trạng thái CCFG không còn tính chất phân bậc này nữa, vì thế chúng ta không thể dùng khái niệm tương tự như shot-vector trong CFG cổ điển. Chúng tôi đặc trưng thứ tự của CCFG dựa vào họ năng lượng. Đặc trưng này phức tạp hơn rất nhiều vì phải kiểm tra năng lượng trên tất cả các lọc của tập đỉnh V , nhưng nó có ưu điểm là kiểm tra được thứ tự của hai trạng thái bất kỳ, không nhất thiết phải xuất phát từ một trạng thái như trong CFG. Có thể nói họ năng lượng phụ thuộc trực tiếp vào bản thân của các trạng thái, không gián tiếp như shot-vector.

Sau đây, chúng ta phát biểu định lý về đặc trưng thứ tự của CCFG:

Định lý 4.2.4. Cho a và b là hai trạng thái của $CCFG(G, n)$. Khi đó $a \geq b$ trong $CCFG(G, n)$ khi và chỉ khi $e(A, a) \geq e(A, b)$, với mọi $A \in \mathcal{F}(V)$.

Chứng minh. Điều kiện cần có thể rút ra trực tiếp từ chứng minh bổ đề 4.2.3. Ta chứng minh điều kiện đủ bằng cách chỉ ra dãy cháy dẫn từ trạng thái a đến b . Chúng ta chứng minh bằng quy nạp theo lực lượng của tập V .

+ Dễ dàng kiểm tra trong trường hợp $|V| = 1$.

+ Giả sử $|V| > 1$. Ta xét hai trường hợp:

- Trường hợp 1: tồn tại lọc $A \in \mathcal{F}(V)$, $A \neq \emptyset$, $A \neq V$ sao cho $e(A, a) = e(A, b)$. Khi đó trên đồ thị cảm sinh $G_1 = G[A]$ ta có $A' \in \mathcal{F}(A)$, suy ra $A' \in \mathcal{F}(V)$. Theo giả thiết $e(A', a) \geq e(A', b)$ với mọi $A' \in \mathcal{F}(A)$, nên theo giả thiết quy nạp ta có $a \rightsquigarrow b$ trên đồ thị con $G[A]$. Bây giờ, ta xét đồ thị con cảm sinh $G_2 = G[V \setminus A]$. Rõ ràng, nếu $B \in \mathcal{F}(V \setminus A)$ thì $A \cup B \in \mathcal{F}(V)$. Gọi B là một lọc bất kỳ của $F(V \setminus A)$. Ta có $e(A \cup B, a) \geq e(A \cup B, b)$ nên $e(B, a) \geq e(B, b)$. Do đó, theo giả thiết quy nạp ta có b nhận được từ a trên đồ thị $G[V \setminus A]$.
- Trường hợp 2: $e(A, a) \geq e(A, b) + 1$ với mọi $A \in \mathcal{F}(V)$, $A \neq \emptyset$, $A \neq V$. Khi

đó tồn tại một đỉnh hút v sao cho $a(v) < b(v)$. Gọi u là một lân cận của v ($(u, v) \in E$). Ta định nghĩa trạng thái c như sau: $c(u) = b(u) + 1, c(v) = b(v) - 1$ và $c(w) = b(w), \forall w \neq u, v$. Khi đó $c \rightarrow b$. Từ đây suy ra, $\mathcal{E}(c) > \mathcal{E}(b)$ và do đó $\mathcal{E}(a) - \mathcal{E}(c) < \mathcal{E}(a) - \mathcal{E}(b)$.

Ta chỉ việc chứng minh $e(A, a) \geq e(A, c), \forall A \in \mathcal{F}(V)$. Khi đó, nếu hai trạng thái a và c thỏa mãn trường hợp 1 thì $a \geq c$ còn nếu a và c thỏa mãn trường hợp 2 thì lại tồn tại trạng thái c' sao cho $c' \rightarrow c$ và $e(A, a) \geq e(A, c'), \forall A \in \mathcal{F}(V)$. Quá trình lặp lại hữu hạn lần vì không gian trạng thái là hữu hạn. Do đó $a \geq b$. Để chứng minh $e(A, a) \geq e(A, c), \forall A \in \mathcal{F}(V)$, ta gọi $A \in \mathcal{F}(V)$ là một lọc tùy ý và xét hai trường hợp:

+ Nếu $v \in A$ thì $u \in A$, do $u \geq v$ in (V, \leq) và $A \in \mathcal{F}(V)$. Do đó, $e(A, a) \geq e(A, b) = e(A, c)$.

+ Nếu $v \notin A$ thì

$$e(A, c) = \begin{cases} e(A, b), & \text{nếu } u \notin A \\ e(A, b) + 1, & \text{nếu ngược lại} \end{cases}$$

Do đó, $e(A, c) \leq e(A, b) + 1 \leq e(A, a)$ (do $e(A, a) > e(A, b)$). \square

Định lý được chứng minh xong. \square

4.3 Thuật toán xác định thứ tự của hệ CCFG trên DAG

Mục đích của phần này là đưa ra thuật toán xác định cấu trúc thứ tự của hệ CCFG, hay một cách tương đương là xác định tính đạt được của hệ CCFG trên đồ thị có hướng không chu trình. Cho trước hai trạng thái a và b của $CCFG(G, n)$, thuật toán sẽ xác định a có dẫn đến được b hay không?

Bài toán đạt được của một hệ động lực là một vấn đề phức tạp bởi vì không gian trạng thái là rất lớn. Trong mô hình CCFG, đồ thị nền G luôn được cho trước có kích thước đủ nhỏ. Chẳng hạn như số các máy tính trong mạng LAN, số nút trong một mạng nội bộ của mô hình rotor router hay số các máy trong một hệ phân tán.

Tuy nhiên, tổng số chip n hay số công việc là rất lớn, nó có thể bằng số mũ cả kích thước của đồ thị nền G . Số phần tử của không gian trạng thái chính là số hợp thành của n trên tập đỉnh V của đồ thị nền. Do đó kích thước của không gian trạng thái thậm chí có thể bằng hàm mũ so với kích thước của đồ thị nền. Thuật toán của chúng tôi đưa ra sau đây có độ phức tạp không phụ thuộc vào tổng số chips n mà chỉ phụ thuộc vào kích thước của đồ thị nền G . Do đó rất hữu hiệu đối với các mạng có đồ thị nền cố định, đặc biệt là đồ thị nền có kích thước nhỏ.

Thuật toán gồm có hai thuật toán con:

- + Thuật toán sinh ra các lọc (Thuật toán I),
- + Thuật toán so sánh hai trạng thái (Thuật toán II).

Như vậy đối với các mạng có đồ thị nền cố định thì chỉ cần chạy Thuật toán I một lần để in ra các lọc của đồ thị. Sau đó, muốn so sánh hai trạng thái a và b thì ta chỉ cần chạy Thuật toán II. Từ đó, ta thấy rằng thuật toán này rất hiệu quả trong trường hợp cần so sánh một dãy các cặp trạng thái $(a_1$ và $b_1)$, ... , $(a_k$ và $b_k)$ của hệ trên một đồ thị nền cố định, vì ta chỉ cần chạy một lần Thuật toán I và k lần Thuật toán II.

Sau đây, chúng tôi trình bày ý tưởng chạy các thuật toán trên.

Thuật toán I:

Đầu vào: ma trận kề $E = (e_{ij})_{m \times m}$ của đồ thị $G = (V, E)$ với $|V| = m$

Đầu ra: tập chứa các lọc $\mathcal{F}(V)$.

Thuật toán II:

Đầu vào: hai trạng thái: $a = (a_1, \dots, a_m)$ và $b = (b_1, \dots, b_m)$.

Đầu ra: trả lời Có nếu $a \geq b$ và KHÔNG nếu ngược lại.

4.3.1 Thuật toán sinh ra các lọc

Để in ra các lọc của tập thứ tự V , chúng tôi sử dụng song ánh giữa tập các lọc $\mathcal{F}(V)$ và tập các phản xích $\mathcal{A}(V)$ của V . Thuật toán I được mô tả như sau:

- Hàm Floyd: chuyển ma trận kề $M[m][m]$ ($m = |V|$) của đồ thị G thành ma trận độ dài đường đi ngắn nhất $M[m][m]$ sao cho với $i, j \in V$, $M[i][j]$ là độ dài đường đi ngắn nhất từ i đến j hoặc là $2m$ nếu không tồn tại đường đi từ i đến j . Như vậy nếu $M[i][j] < 2m$ thì $j \leq i$ trong V .

- Hàm Filter(h): tính toán các phản xích của V , và với mỗi phản xích A , hàm Filter xác định lọc F tương ứng và lưu lại trong một tập.

+ Mỗi phản xích được ký hiệu là $A[]$, và với mỗi số nguyên h , $u = A[h]$ là một phần tử của A .

Ta định nghĩa mảng $index[i]$ với $i \in V$ như sau:

$$index[i] = \begin{cases} 0 & \text{nếu } i \text{ không so sánh được với các phần tử của } A \\ -u & \text{nếu } i \leq u \\ u & \text{nếu } i > u \end{cases}$$

+ Ta bắt đầu bằng một phản xích A gồm chỉ 1 phần tử và viết lọc tương ứng với phản xích này. Sau đó ta thêm vào A một phần tử khác đang được đánh số 0.

+ Mỗi lần thêm vào A một phần tử ta đánh số lại các phần tử trong A như sau:

· Đầu tiên các phần tử $i \in V$ được đánh số 0.

· Mỗi lần thêm một phần tử u vào A ($u = A[h]$), ta cập nhật chỉ số $i \in V$

như sau:

$$\text{nếu } index[i] = 0 \text{ thì } \begin{cases} index[i] = -u \text{ với } i \leq u, \\ index[i] = u \text{ với } i > u. \end{cases}$$

Sau đây, chúng tôi trình bày hàm Index đánh số các đỉnh của đồ thị, từ đó in ra các phản xích và các lọc tương ứng bằng hàm Filter.

1) **Hàm đánh số:** có biến kiểu nguyên h , đánh số các đỉnh của đồ thị. Hàm này đánh số các đỉnh đã đánh số 0 thành $-AC[h]$ nếu có đường đi từ đỉnh đó đến đỉnh $AC[h]$ hoặc chính là đỉnh $AC[h]$. Các đỉnh mà từ đỉnh $AC[h]$ có thể đi đến được thì đánh số là $AC[h]$.

```

procedure index (interger  $h$ )
 $k = PX[h]$ ;
 $index[k] = -k$ ;
for  $i := 1$  to  $m$  do
    if  $index[i] == 0$  then
        if  $M[k][i] < 2 \times m$  then  $index[i] = k$ ;
        if  $M[i][k] < 2 \times m$  then  $index[i] = -k$ ;
    end if
end for

```

2) **Hàm Filter:** có biến kiểu nguyên h . Hàm này sinh ra các lọc tạo bởi phản xích gồm h phần tử đầu tiên trong mảng $AC[m]$ đồng thời gọi đệ quy $Filter(h + 1)$.

```

procedure Filter (interger  $h$ )
 $index(h)$ ; // đánh số các đỉnh liên quan đến đỉnh  $AC[h]$ .
for  $i := 1$  to  $m$  do
    if ( $DS[i] < 0$ ) then print  $i$ ; // viết lọc sinh bởi phản xích
end for
 $l := 0$ ; // biến đếm các đỉnh không so sánh được với  $h$  đỉnh  $PX[1], \dots, PX[h]$ .
for  $i := 1$  to  $m$  do
    if  $DS[i] = 0$  then  $l = l + 1$ ;

```

```

end for

if  $l == 0$  then return; // tất cả các đỉnh đã được đánh số thì thoát ra.

else for  $i := PX[h] + 1$  to  $m$  // vòng lặp cho các đỉnh có chỉ số lớn hơn
 $PX[h]$ .

    if  $DS[i] == 0$  then // các đỉnh không so sánh được với  $h$  phần tử
         $PX[h + 1] = i$ ; // đầu tiên được thêm vào mảng  $PX[m]$ .

        Filter( $h + 1$ );

        for  $j := 1$  to  $m$  do
            if ( $DS[j] == -i$  OR  $DS[j] == i$ ) then  $DS[j] = 0$ ; //
        end for // xóa các đỉnh đã được đánh số liên quan đến
 $PX[h + 1]$ .

    end if

end for

end else

```

Tiếp theo, chúng ta phân tích độ phức tạp của thuật toán.

Định lý 4.3.1. *Thuật toán I sinh ra tất cả các lọc của V với độ phức tạp $O(m^3 + m|\mathcal{F}(V)|)$, trong đó $m = |V|$.*

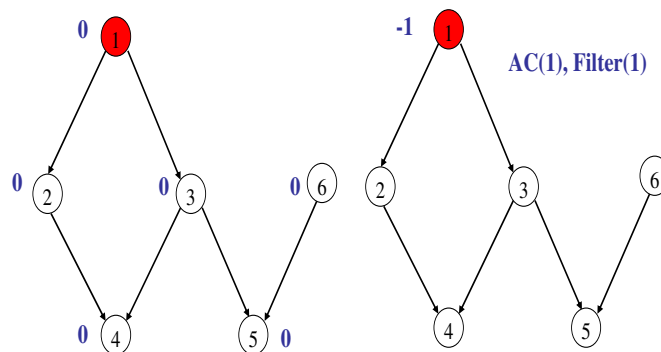
Chứng minh. Ta cần chứng minh rằng thuật toán sinh ra tất cả các lọc của V (hay một cách tương đương, tất cả các phản xích) và mỗi lọc được sinh ra đúng một lần. Thật vậy, ta hãy xét quá trình sinh một phản xích có h phần tử $A = \{p_1, \dots, p_h\}$. Ta bắt đầu bằng một phản xích chỉ có một phần tử $A := \{p_1\}$ ($AC[1] = p_1$). Sau đó, ta thêm một phần tử $\{p_2\}$ ở "vòng for" **for** $i := AC[1] + 1$ **to** m (phần tử này được đánh số lớn hơn phần tử p_1 và không so sánh được với phần tử p_1 trong (V, \leq)). Lúc này ta có $A := \{p_1, p_2\}$. Tương tự, trong "vòng for" **for** $i := AC[h - 1] + 1$

to m , ta thêm phần tử $AC[h] = p_h$ vào A và gọi hàm **Filter**(1). Hàm này sinh ra phản xích $A = \{p_1, \dots, p_h\}$. Hơn nữa, các đỉnh nằm trong một phản xích được đánh số tăng dần và mỗi khi thêm một đỉnh mới thì đỉnh này phải không so sánh được với các phần tử trước đó trong phản xích. Do vậy, mỗi lọc tương ứng với một phản xích được sinh ra đúng một lần.

Bây giờ, ta phân tích độ phức tạp của thuật toán. Thuật toán FloydWarshall chạy trong thời gian $O(m^3)$. Vòng for thứ hai trong hàm main có m bước và mỗi bước chạy trong thời gian $O(m)$. Mỗi khi gọi hàm Filter thì hàm đánh số (index) cũng được gọi trong hàm Filter. Hàm đánh số chạy trong thời gian $O(m)$. Vậy, độ phức tạp của thuật toán là $O(m^3 + m^2 + m \cdot |\mathcal{F}(V)|)$ hay là $O(m^3 + m \cdot |\mathcal{F}(V)|)$. Định lý được chứng minh. \square \square

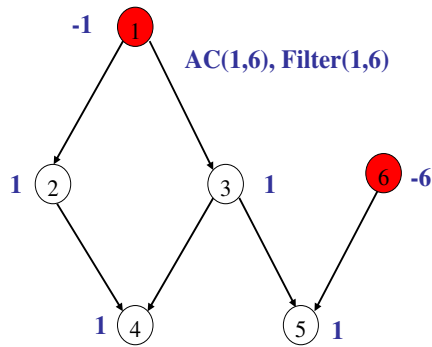
Ví dụ 4.3.2. Thuật toán sinh ra các lọc. Trong ví dụ sau đây, các đỉnh của đồ thị được biểu diễn bằng các hình tròn và được đánh số, các đỉnh được tô màu là các đỉnh được thêm vào để tạo nên phản xích mới.

(i) Trước hết xét một đỉnh bất kỳ, chẳng hạn ta xét đỉnh 1. Lúc đầu tất cả các đỉnh được đánh số 0. Sau đó, đánh số lại các đỉnh: Các đỉnh không nhỏ hơn đỉnh 1 trong thứ tự của tập đỉnh V được đánh số -1. Các đỉnh lớn hơn đỉnh 1 được đánh số 1, các đỉnh không so sánh được với đỉnh 1 vẫn giữ nguyên (được đánh số 0). Như vậy, ta có phản xích đầu tiên là $\{1\}$ và lọc đầu tiên là $\{1\}$.



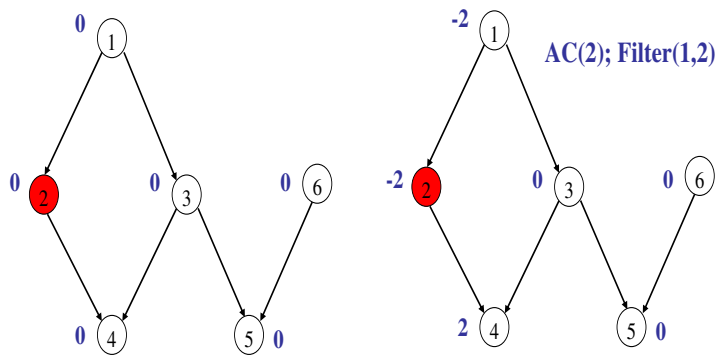
Hình 4.2: Xét đỉnh 1 và đánh số lại các đỉnh.

Tiếp theo, ta thêm đỉnh không so sánh được với đỉnh 1 là đỉnh 6, lúc này đỉnh 6 được đánh số lại là -6. Các đỉnh còn lại đã được đánh số khác 0 nên thoát ra và ta có phản xích mới gồm các đỉnh được đánh số nhỏ hơn 0 là $\{1, 6\}$ và lọc tương ứng $\{1, 6\}$. Hình vẽ sau minh họa điều đó.



Hình 4.3: Đỉnh 6 được thêm vào phản xích $\{1\}$ và được đánh số lại.

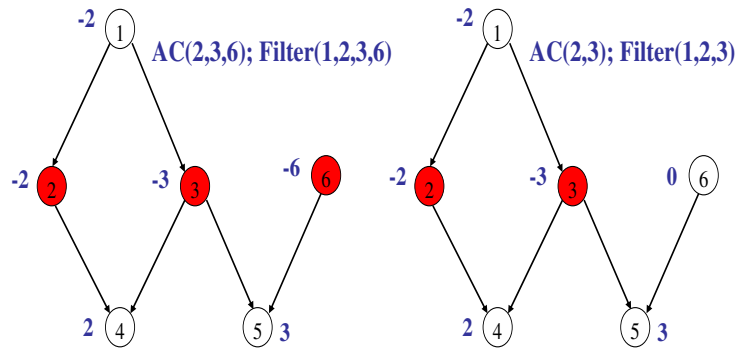
(ii) Bước tiếp theo ta xét đỉnh 2, và bây giờ các đỉnh lại được đánh số 0. Quá trình được lặp lại tương tự như trên và được minh họa bởi các hình vẽ sau. Chú ý rằng các đỉnh đã được đánh số rồi thì không đánh lại khi xét thêm đỉnh mới:



Hình 4.4: Đánh số lại các đỉnh liên quan đến đỉnh 2 và sinh lọc.

+ Thêm đỉnh 3 (đang được đánh số 0) vào phản xích $\{2\}$, đánh số lại các đỉnh liên quan đến đỉnh 3 và sinh lọc tương ứng

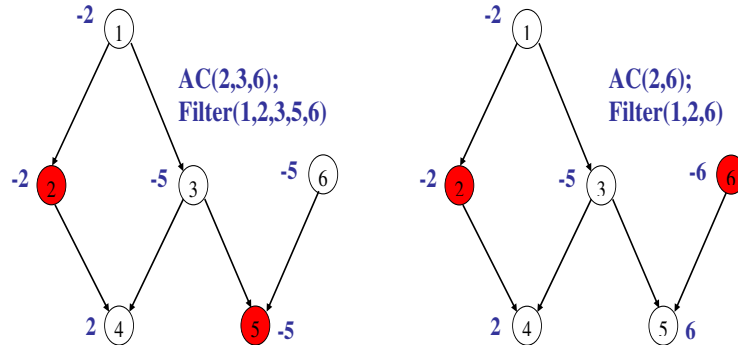
Thêm đỉnh 6 (đang được đánh số 0) vào phản xích $\{2, 3\}$, đánh số lại các đỉnh liên quan đến đỉnh 6 và sinh lọc tương ứng:



Hình 4.5: Thêm đỉnh 3, đỉnh 6, đánh số lại và sinh lọc tương ứng.

Như vậy khi xét phản xích $\{2, 3\}$, các đỉnh đã được đánh số khác 0 nên quay lại khi đang xét đỉnh 2.

+ Thêm đỉnh 5 (đang được đánh số 0) vào phản xích $\{2\}$, đánh số lại các đỉnh liên quan đến đỉnh 5 và sinh lọc. Tương tự, thêm đỉnh 6 vào phản xích $\{2\}$, đánh số lại và sinh lọc (Hình 4.6):



Hình 4.6: Thêm đỉnh 5, đánh số lại và sinh lọc.

(iii) Quá trình tiếp tục, ta xét các đỉnh còn lại với chỉ số tăng dần và làm tương tự như trên.

4.3.2 Thuật toán so sánh hai trạng thái

Dựa vào định lý 4.2.4 về đặc trưng thứ tự của hệ CCFG, chúng tôi xây dựng thuật toán so sánh hai trạng thái a và b .

Dữ liệu chuẩn bị:

1. Hai mảng một chiều $a[m]$ và $b[m]$ cho hai trạng thái a và b ;
2. Một file F chứa tất cả các lọc.
3. Một biến C để quyết định $a \geq b$?

Độ phức tạp của Thuật toán II phụ thuộc vào số phần tử của mỗi lọc và kết quả $a \geq b$ hay không. Trong trường hợp xấu nhất, Thuật toán II chạy trong thời gian $O(|V| \cdot |\mathcal{F}(V)|)$ vì phải tính năng lượng của a và b trên tất cả các lọc.

Chú ý: Thuật toán I và thuật toán II đã được lập trình bằng ngôn ngữ C^{++} và cho ra kết quả mong muốn (xem hình 4.7, 4.12).

```

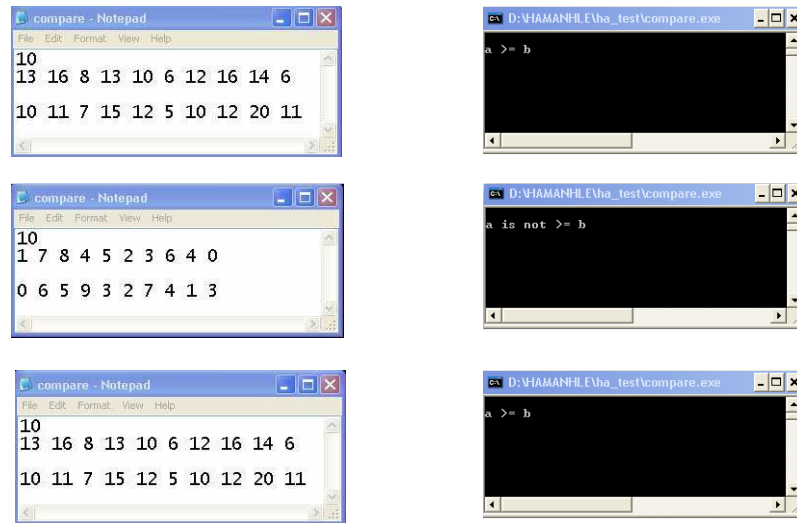
filter.inp - Notepad
File Edit Format View Help
10
0 1 1 1 1 0 0 0 0 0
0 0 1 0 1 1 0 0 0 0
0 0 0 1 0 0 1 0 0 0
0 0 0 1 0 0 1 0 0 0
0 0 0 0 0 0 0 1 0 0
0 0 0 1 0 0 0 0 1 0
0 0 0 0 0 0 0 1 0 1
0 0 0 0 0 0 0 1 0 1
0 0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 1 0 1
0 0 0 0 0 0 0 1 0 1
0 0 0 0 0 0 0 0 0 0

filter.out - Notepad
File Edit Format View Help
1 1
2 1 2
3 1 2 3
4 1 2 3 5
5 1 2 3 5 6
4 1 2 3 6 6
6 1 2 3 5 6 9
5 1 2 3 4 5 6
6 1 2 3 4 5 6 7
7 1 2 3 4 5 6 7 9
8 1 2 3 4 5 6 7 9
7 1 2 3 4 5 6 9
3 1 2 5 6
4 1 2 5 6
6 1 2 3 5 6 7
3 1 2 6
5 1 2 3 6 7
7 1 2 3 5 6 7 9
9 1 2 3 4 5 6 7 8 9
5 1 2 5 6 9
10 1 2 3 4 5 6 7 8 9 10
11
  
```

Hình 4.7: Đầu vào và đầu ra của chương trình in ra các lọc

4.4 Mạng vận tải

Tiếp theo chúng tôi sẽ đưa ra đặc trưng cho tính đạt được của hệ CCFG trên đồ thị có hướng tổng quát. Trước hết, chúng tôi trình bày khái niệm mạng vận tải tương ứng với cấu hình của đồ thị. Khái niệm này sẽ được dùng để đặc trưng tính đạt được



Hình 4.8: Một số kết quả của thuật toán so sánh hai trạng thái. Trái: đầu vào; phải: đầu ra.

của hệ CCFG trên đồ thị có hướng. Chúng tôi cũng đưa ra thuật toán thời gian đa thức để giải quyết bài toán đạt được của CCFG trên đồ thị có hướng.

Trong phần này chúng tôi sẽ đưa ra khái niệm mạng vận tải tương ứng với cấu hình của CCFG. Trước hết, xin nhắc lại một số định nghĩa cơ bản về mạng vận tải và luồng trên mạng.

Một **mạng vận tải** được định nghĩa là bộ $G = (V, E, w, s, t)$, trong đó (V, E) là một đồ thị có hướng liên thông không có khuyên, $w : E \rightarrow \mathbb{R}^+ \cup \{0\}$ là hàm trọng lượng trên tập cung E , s là một đỉnh xác định cho trước gọi là *đỉnh phát*, còn t là một đỉnh cho trước khác gọi là *đỉnh thu*. Giả sử $f : E \rightarrow \mathbb{R}^+ \cup \{0\}$ là hàm xác định trên tập cung E . Khi đó, với mỗi cung $(u, v) \in E$ ta sẽ viết giá trị $f((u, v))$ của hàm f trên cung (u, v) . Ta có định nghĩa *luồng (flow)* trong mạng vận tải như sau:

Định nghĩa 4.4.1. Cho mạng vận tải $G = (V, E, w, s, t)$. Luồng trong mạng vận tải G là một hàm $f : V \times V \rightarrow \mathbb{R}$ với $f(u, v) = 0, \forall (u, v) \notin E$ và thỏa mãn các tính chất sau:

- **Khả năng thông qua:** $f(u, v) \leq w(u, v)$, với mọi $u, v \in V$.

- **Tính chất đối xứng:** $f(u, v) = -f(v, u)$, với mọi $u, v \in V$.
- **Tính bảo toàn luồng:** $\sum_{v \in V} f(u, v) = 0$, với mọi $u \in V \setminus \{s, t\}$.

Với mỗi cung (u, v) , giá trị $w(u, v)$ gọi là *khả năng thông qua* của cung (u, v) , còn giá trị $f(u, v)$ của luồng f trên cung (u, v) gọi là *giá trị của luồng f qua cung (u, v)* . Điều kiện thứ nhất trong định nghĩa của luồng nói rằng giá trị của luồng f qua một cung $(u, v) \in E$ bất kỳ không vượt quá khả năng thông qua của cung đó. Điều kiện thứ hai trong định nghĩa của luồng nói lên tính đối xứng, luồng đi từ u đến v phải bằng đối của luồng đi từ v đến u . Điều kiện thứ ba thể hiện luật bảo toàn luồng của Kirchhoff: ở mỗi đỉnh trừ đỉnh phát và đỉnh thu, tổng giá trị của luồng qua các cung đi vào đỉnh đó bằng tổng giá trị của luồng qua các cung đi ra từ nó.

Giá trị của luồng f ký hiệu là $|f|$ và được định nghĩa bằng $|f| = \sum_{v \in V} f(s, v)$, tức là tổng luồng đi ra khỏi nguồn s .

Bài toán tìm luồng lớn nhất: *Giả sử $G = (V, E, w, s, t)$ là một mạng vận tải đã cho. Hãy tìm một luồng f trên mạng G sao cho giá trị $|f|$ là lớn nhất trong các giá trị của luồng trong G .*

Bài toán tìm luồng lớn nhất trong mạng vận tải thường gặp trong các ứng dụng thực tiễn, chẳng hạn như cân xác định cường độ lớn nhất cho phép trong một mạng điện, lượng dòng chảy lớn nhất trong một mạng sông ngòi, vv và vv. Bài toán này đã được giải quyết bằng một số thuật toán nổi tiếng như thuật toán Ford-Fulkerson, thuật toán Push-Relabel, vv và vv. Đặc biệt, thuật toán Push-Relabel chạy trong thời gian $O(|V|^3)$. Chúng tôi sẽ sử dụng thuật toán Push-Relabel để giải bài toán đạt được của CCFG trong mục 4.5.

Tiếp theo, chúng tôi sẽ trình bày khái niệm cấu hình của đồ thị, một khái niệm mở rộng của khái niệm trạng thái của CCFG.

Định nghĩa 4.4.2. Một *cấu hình* (configuration) của đồ thị $G = (V, E)$ là một hàm

số $c : V(G) \rightarrow \mathbb{R}$ và được ký hiệu là $c = \{c(v)\}_{v \in V}$, với $c(v)$ là giá trị tại đỉnh v .

Định nghĩa 4.4.3. Cho $a = \{a(v)\}_{v \in V}$ và $b = \{b(v)\}_{v \in V}$ là hai cấu hình của đồ thị G . Cấu hình $\{a(v) - b(v)\}_{v \in V}$, ký hiệu là $a - b$, được gọi là *cấu hình lệch* của hai cấu hình a và b .

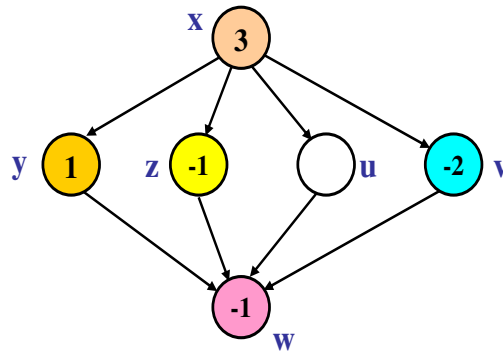
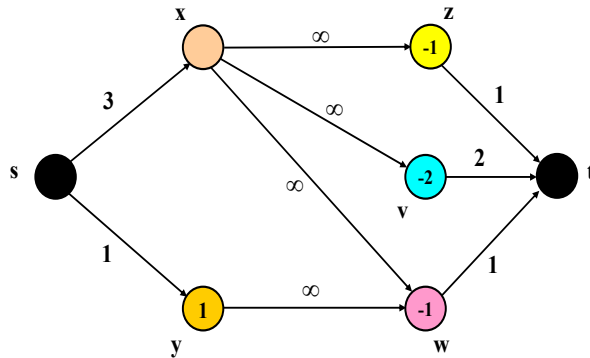
Sau đây, chúng tôi trình bày khái niệm *mạng vận tải tương ứng* (corresponding flow network) với cấu hình của đồ thị. Với mỗi cấu hình $c = \{c(v)\}_{v \in V}$ của đồ thị G , ta xác định duy nhất một mạng vận tải tương ứng (G_c, w, s, t) (xem hình 4.9 và hình 4.10). Ta sẽ dựa vào khái niệm này để giải bài toán đạt được của CCFG.

Định nghĩa 4.4.4. Cho $c = \{c(v)\}_{v \in V}$ là một cấu hình của đồ thị G . *Mạng vận tải tương ứng* với cấu hình c là mạng (G_c, w, s, t) được định nghĩa như sau:

- s là đỉnh nguồn và t đỉnh đích.
- $V(G_c) = \{s, t\} \cup P \cup N$, với $P = \{v \in V(G) | c(v) > 0\}$, $N = \{v \in V(G) | c(v) < 0\}$.
- $E(G_c) = \{(s, v) | v \in P\} \cup \{(v, u) | v \in P, u \in N, v \rightsquigarrow u \text{ trong } G\} \cup \{(u, t) | u \in N\}$
- $$\begin{cases} w(s, v) = c(v), \forall v \in P; \\ w(u, t) = -c(u), \forall u \in N; \\ w(v, u) = +\infty, \forall v \in P, \forall u \in N \text{ sao cho } (v, u) \in E(G_c). \end{cases}$$

4.5 Tính đạt được của hệ CCFG trên đồ thị có hướng

Một trong những kết quả chính của chương này là định lý đặc trưng về tính đạt được của hệ CCFG trên đồ thị có hướng. Sau đây, chúng tôi sẽ trình bày định lý đặc trưng về tính đạt được của hệ CCFG dựa vào mạng vận tải tương ứng với cấu hình của đồ thị.

Hình 4.9: Một trạng thái C trên đồ thị G .Hình 4.10: Mạng vận tải tương ứng với trạng thái c .

Định lý 4.5.1. Cho a và b là hai trạng thái của $CCFG(G, n)$. Khi đó b đạt được từ a khi và chỉ khi luồng trên mạng vận tải tương ứng với cấu hình lệch $c = a - b$ đạt giá trị cực đại là $\sum_{c(v) > 0} c(v)$.

Chứng minh. Gọi (G_c, w, s, t) là mạng vận tải tương ứng với cấu hình c .

a) Trước hết, ta chứng minh điều kiện cần. Gọi f là giá trị luồng cực đại trên mạng (G_c, w, s, t) . Theo giả thiết, giá trị của luồng cực đại bằng $\sum_{c(v) > 0} c(v)$ nên ta có

$$f(s, v) = c(v), \forall v \in P \text{ và } f(s, v) = \sum_{u: (v, u) \in E(G_c)} f(v, u), \forall v \in P$$

$$f(u, t) = -c(u), \forall u \in N \text{ và } f(u, t) = \sum_{v:(v,u) \in E(G_c)} f(v, u), \forall u \in N.$$

Khi đó xuất phát từ trạng thái a , tại mỗi đỉnh $v \in P$, ta chuyển $f(v, u)$ chips từ đỉnh v đến đỉnh $u \in N$ nếu $(v, u) \in E(G_c)$ (ta có thể chuyển được vì $v \rightsquigarrow u$ trong G). Sau việc chuyển này thì tại mỗi đỉnh $v \in P$ số chips còn lại là $a(v) - \sum_{(v,u) \in E(G_c)} f(v, u) = a(v) - c(v) = a(v) - (a(v) - b(v)) = b(v)$; trong khi đó số chips tại mỗi đỉnh $u \in N$ sẽ là $a(u) + \sum_{(v,u) \in E(G_c)} f(v, u) = a(u) + f(u, t) = a(u) - c(u) = a(u) - (a(u) - b(u)) = b(u)$. Như vậy, sau quá trình này ta có được trạng thái b .

b) Sau đây, ta chứng minh điều kiện đủ, tức là nếu b nhận được từ a thì mạng vận tải tương ứng với cấu hình lệch $c = a - b$ có giá trị luồng lớn nhất là $\sum_{c(v) > 0} c(v)$. Ta sẽ chứng minh bằng quy nạp theo độ dài của bước chuyển $a \rightsquigarrow b$. Dễ dàng kiểm tra được trong trường hợp $a \rightarrow b$.

Giả sử $a \rightsquigarrow b$ có độ dài $k + 1$ và khẳng định đúng cho các bước chuyển có độ dài k . Khi đó tồn tại trạng thái b_1 sao cho $a \rightsquigarrow b_1$ và $b_1 \rightarrow b$ bằng cách chuyển một chip dọc cung (i, j) và giả thiết luồng cực đại của mạng vận tải tương ứng với cấu hình $c_1 = a - b_1$ có giá trị $\sum_{c_1(v) > 0} c_1(v)$.

Giả sử f_1 là luồng cực đại của mạng vận tải (G_{c_1}, w_1, s, t) với $f_1(s, v) = c_1(v), \forall v \in P_1, f_1(u, t) = -c_1(u), \forall u \in N_1$, với $V(G_{c_1}) = \{s, t\} \cup P_1 \cup N_1$. Ta sẽ xây dựng luồng cực đại f của mạng vận tải (G_c, w, s, t) dựa trên luồng f_1 . Để ý rằng $V(G_c) = \{s, t\} \cup P \cup N$ và trong trường hợp này P hay N có thể chứa đỉnh v với $c(v) = 0$.

Ta xét các trường hợp sau:

- Nếu $c_1(i) = c_1(j) = 0$: sau bước chuyển $b_1 \rightarrow b$ ta có $P = P_1 \cup \{i\}, N = N_1 \cup \{j\}, c_1(i) = 1, c_1(j) = -1$. Ta xây dựng luồng f của mạng vận tải tương

ứng với trạng thái c như sau:

$$f(s, v) = f_1(s, v), \forall v \in P, f(u, t) = f_1(u, t), \forall u \in N$$

$$f(s, i) = 1 = w(s, i), f(j, t) = 1 = w(j, t), \text{ và } f(i, j) = 1$$

(để ý rằng $i \rightarrow j$ in $V(G)$)

Để dàng kiểm tra được f thỏa mãn các điều kiện của luồng và $f(s, v) = w(s, v), \forall v \in P$, do đó f là một luồng cực đại.

- Nếu $c_1(i) > 0, c_1(j) > 0$: tương tự ta có $P = P_1, N = N_1, c(i) = c_1(i) + 1, c(j) = c_1(j) - 1$ (nếu $c(j) = 0$ thì $w(s, j) = 0$).

Vì $c_1(j) > 0$ nên tồn tại $u \in N$ sao cho $(j, u) \in E(G_{c_1})$ và $f_1(j, u) > 0$. Hơn nữa, $(i, j) \in E(G), (j, u) \in E(G_{c_1})$ nên suy ra $(i, u) \in E(G_{c_1})$. Luồng cực đại f được xây dựng như sau:

$$f(s, i) = f_1(s, i) + 1 = w(s, i), f(s, j) = f_1(s, j) - 1 = w(s, j),$$

$$f(i, u) = f_1(i, u) + 1, f(j, u) = f_1(j, u) - 1$$

và $f = f_1$ trên các cung còn lại.

- Nếu $c_1(i) < 0, c_1(j) < 0$: ta có $P = P_1, N = N_1, c(i) = c_1(i) + 1, c(j) = c_1(j) - 1$. Vì $f_1(i, t) > 0$ nên tồn tại $v \in P$ sao cho $(v, i) \in E(G_e)$ và $f_1(v, i) > 0$. Do $(i, j) \in E(G)$ nên ta có $(v, j) \in E(G_e)$. Luồng cực đại f được xây dựng như sau:

$$f(i, t) = f_1(i, t) - 1 = w(i, t), f(j, t) = f_1(j, t) + 1 = w(j, t),$$

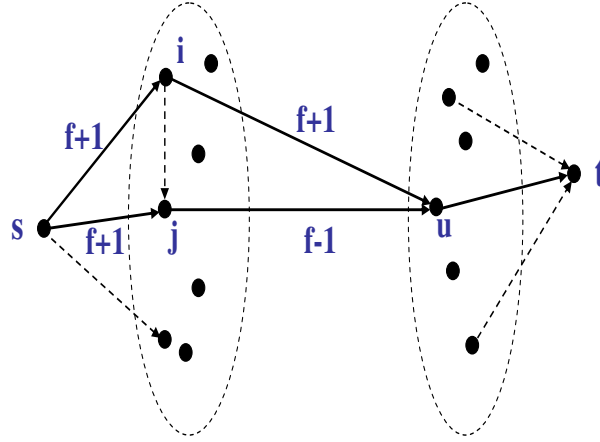
$$f(v, i) = f_1(v, i) - 1, f(v, j) = f_1(v, j) + 1$$

và $f = f_1$ trong các trường hợp còn lại.

- Nếu $c_1(i) > 0, c_1(j) < 0$: ta có $P = P_1, N = N_1, c(i) = c_1(i) + 1, c(j) = c_1(j) - 1$. Ta xây dựng luồng cực đại f như sau:

$$f(s, i) = f_1(s, i) + 1 = w(s, i), f(i, j) = f_1(i, j) + 1, f(j, t) = f_1(j, t) + 1 = w(j, t)$$

và $f = f_1$ trên các cung còn lại.



Hình 4.11: Luồng cực đại f được xây dựng dựa trên luồng f_1 trong trường hợp $c_1(i) > 0, c_1(j) > 0$.

- Nếu $c_1(i) < 0, c_1(j) > 0$: ta có $P = P_1, N = N_1, c(i) = c_1(i) + 1, c(j) = c_1(j) - 1$. Khi đó ta xét hai trường hợp:

+ Nếu tồn tại một đường đi có hướng từ j đến i thì $(j, i) \in E(G_c)$ và hàm f được xây dựng như sau:

$$f(s, j) = f_1(s, j) - 1 = c_1(j) - 1 = c(j) = w(s, j)$$

$$f(j, i) = f_1(j, i) - 1, f(i, t) = f_1(i, t) - 1 = c_1(i) - 1 = c(i) = w(i, t),$$

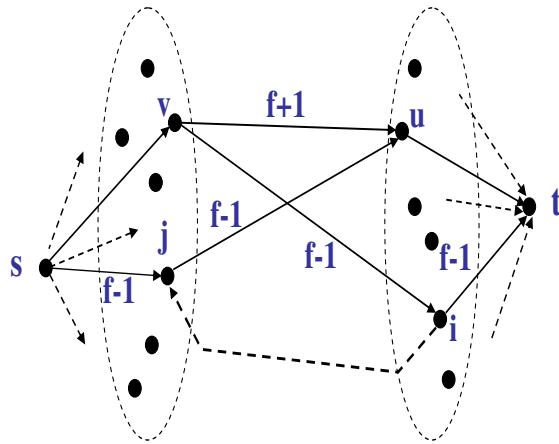
là luồng cực đại cần tìm.

+ Nếu không có đường đi từ j đến i tức là $(j, i) \notin E(G_c)$ thì tồn tại $v \in P, v \neq j, f_1(v, i) > 0$ và $u \in N_1, u \neq i, f_1(j, u) > 0$. Như vậy, tồn tại đường đi $v \rightsquigarrow i \rightsquigarrow j \rightsquigarrow u$ trong G và do đó $(v, u) \in E(G_c)$. Luồng cực đại f được xây dựng như sau:

$$f(s, j) = f_1(s, j) - 1 = w(s, j), f(j, u) = f_1(j, u) - 1$$

$$f(v, u) = f_1(v, u) + 1, f(v, i) = f_1(v, i) - 1, f(i, t) = f_1(i, t) - 1 = w(i, t)$$

và $f = f_1$ trên các cung còn lại.



Hình 4.12: Luồng cực đại f được xây dựng dựa trên luồng f_1 trong trường hợp $c_1(i) < 0, c_1(j) > 0$.

- Nếu $c_1(i) = 0, c_1(j) \neq 0$: Ta có thể thêm i vào P_1 với $w(s, i) = 0$ và lý luận tương tự như trong trường hợp $c_1(i) > 0$, ta sẽ nhận được luồng cực đại f .
- If $c_1(i) \neq 0, c(j) = 0$: Tương tự như trên, ta thêm j vào N với $w(j, t) = 0$ và xây dựng luồng cực đại f như trong trường hợp $c(j) < 0$.

Như vậy, trong mọi trường hợp ta có thể xây dựng luồng cực đại của mạng vận tải (G_c, w, s, t) có giá trị là $\sum_{c(v) > 0} c(v)$. Định lý được chứng minh xong. \square

4.6 Thuật toán

Như đã trình bày trong phần trước, bài toán đạt được của mạng Petri chỉ giải được trong thời gian đa thức nếu nó không tương tranh và bị chặn. Để ý rằng mark graph cũng không tương tranh. Trong trường hợp có yếu tố tương tranh ta phải xét hết tất cả các trường hợp xảy ra của các transition nên phải chạy trong thời gian mũ. Trong phần đầu của chương này chúng tôi cũng đã đưa ra thuật toán để giải bài toán đạt được của hệ CCFG trên đồ thị không có chu trình. Yếu tố tương tranh giữa các cung khi thực hiện hoạt động cháy dẫn đến việc phải xét tất cả các khả năng cháy được của các cung nên cần phải tính năng lượng của các trạng thái trên tất cả các lọc, tức là ta phải in ra tất cả các lọc của tập V . Do đó, thuật toán này chạy

trong thời gian $O(|V|^3 + |V| \cdot |\mathcal{F}(V)|)$, tức là thời gian mũ đối với $|V|$.

Trong mục này, chúng tôi sẽ đưa ra thuật toán xác định tính đạt được của hệ CCFG trên đồ thị có hướng dựa vào mạng vận tải. Theo định lý 4.5.1 thì chỉ cần tìm luồng cực đại trên mạng vận tải tương ứng với cấu hình lệch $a - b$ của đồ thị thì xác định được trạng thái a có dẫn đến được trạng thái b hay không? Thuật toán này chạy trong thời gian đa thức của $|V(G)|$.

Thuật toán

Đầu vào: đồ thị $G = (V, E)$ và hai trạng thái a, b .

Đầu ra: trả lời Yes nếu b đạt được từ a và trả lời No nếu ngược lại. Trong trường hợp câu trả lời là Yes, thì in ra dãy chầy dẫn từ a đến b .

Thuật toán gồm có ba bước:

- Dùng thuật toán Floyd để tạo nên mạng vận tải có độ phức tạp là $O(|V(G)|^3)$.
- Dùng thuật toán Push-Relabel tìm luồng cực đại với độ phức tạp là $O(|V(G)|^2 \sqrt{|E(G_c)|})$ [35, 79]. Nếu giá trị luồng lớn nhất là $\sum_{c(v)>0} c(v)$ thì câu trả lời là Yes. Câu trả lời là No nếu ngược lại.
- In ra dãy chầy dựa vào chứng minh của Định lý 4.5.1 trong thời gian $O(|V(G)|^2)$.

Như vậy, thuật toán này chạy trong thời gian $O(|V(G)|^3)$ và ta có mệnh đề sau.

Mệnh đề 4.6.1. *Thuật toán xác định tính đạt được của hệ CCFG trên đồ thị có hướng $G = (V, E)$ chạy trong thời gian $O(|V|^3)$.*

Việc xây dựng mạng vận tải tương ứng với cấu hình $a - b$ của đồ thị chỉ liên quan đến các đỉnh v có số chip khác nhau giữa hai trạng thái a và b . Do đó, thực tế độ phức tạp của thuật toán trên chỉ là k^3 với k là số đỉnh có số chip khác nhau giữa a và b . Thuật toán này rất hiệu quả khi dùng để so sánh hai trạng thái của các mạng phức tạp có đồ thị nền lớn.

4.7 Kết luận chương 4

Trong chương này chúng tôi nghiên cứu tính đạt được của hệ CCFG trên đồ thị có hướng và đã đạt được những kết quả sau:

- Xây dựng họ năng lượng của các trạng thái của hệ CCFG để đặc trưng cấu trúc thứ tự của hệ CCFG trên đồ thị có hướng không chu trình.
- Xây dựng thuật toán xác định thứ tự của hệ CCFG trên đồ thị có hướng không chu trình.
- Xây dựng mạng vận tải tương ứng với trạng thái của CCFG trên đồ thị có hướng để đặc trưng tính đạt được của hệ CCFG trên đồ thị có hướng.
- Xây dựng thuật toán trong thời gian $O(|V|^3)$ (V là tập đỉnh của đồ thị nền) xác định tính đạt được của hệ CCFG trên đồ thị có hướng

Kết luận của luận án

Như vậy, trong luận án này chúng tôi đã thu được những kết quả chính sau:

1. Chứng minh cấu trúc dàn của tập $d\text{-}\mathcal{P}(n)$ các phân hoạch d -chặt của một số tự nhiên n cho trước (d là số tự nhiên cho trước) và mở rộng vô hạn $d\text{-}\mathcal{P}(\infty)$ của nó.
2. Xây dựng toán tử ECO cho phân hoạch d -chặt của các số tự nhiên và xây dựng cây sinh tương ứng. Chứng minh cấu trúc đệ quy của cây sinh $T_{d\text{-}\mathcal{P}(n)}$ và cây sinh vô hạn $T_{d\text{-}\mathcal{P}(\infty)}$. Từ đó chứng minh được một số đẳng thức tổ hợp.
3. Chứng minh các hệ động lực CFG là các mạng Petri đặc biệt.
4. Đặc trưng cấu trúc thứ tự của hệ động lực CCFG trên đồ thị có hướng không chu trình.
5. Xây dựng thuật toán xác định thứ tự của hệ CCFG trên đồ thị có hướng không chu trình.
6. Đặc trưng tính đạt được của hệ CCFG trên đồ thị có hướng.
7. Xây dựng thuật toán trong thời gian $O(|V|^3)$ (V là tập đỉnh của đồ thị nền) xác định tính đạt được của CCFG trên đồ thị có hướng.

Một số hướng phát triển của luận án

1. Nghiên cứu thời gian đạt đến điểm dừng và thời gian hội tụ của các mô hình CFG.
2. Nghiên cứu mô hình CCFG trong mối quan hệ với mô hình rotor-router và bài toán du động ngẫu nhiên (random walk).
3. Nghiên cứu cấu trúc không gian trạng thái của hệ CCFG và mối quan hệ với các lớp dàn phân phối, dàn ULD.
4. Nghiên cứu mô hình CCFG và bài toán rotor-router tô màu.
5. Đặc trưng không gian trạng thái của mô hình CCFG.
6. Nghiên cứu mô hình CFG có hướng và đa thức Tutte.
7. Nghiên cứu các matroid trên đồ thị có hướng và mối quan hệ với mô hình CCFG.

Các công trình liên quan đến luận án

A. Các bài đăng ở tạp chí:

1. Le Manh Ha and Phan Thi Ha Duong (2009), "Integer partitions in discrete dynamical models and ECO method", *Vietnam J. Math.*, (2-3) **37**, pp. 273-293.

2. Le Manh Ha and Phan Thi Ha Duong (2010), "Order structure and energy of conflicting Chip Firing Game", *Acta Math. Vietnam.*, (2) **35**, pp. 289-301.

B. Các bài đăng trong kỷ yếu Hội nghị quốc tế, có phản biện, có số ISBN:

3. LE Manh Ha, PHAM Tra An, PHAN Thi Ha Duong (2009), "On the relation between Chip Firing Games and Petri Nets", *Proceeding of IEEE-RIVF International Conference on Computing and Communication Technologies*, ISBN: 978-1-4244-4567-7, pp. 328-335.

4. LE Manh Ha; NGUYEN Anh Tam; PHAN Thi Ha Duong.(2010), "Algorithmic aspects of the Reachability of Conflicting Chip Firing Game", *Advances in Intelligent Information and Database Systems* in series *Studies in Computational Intelligence*, Springer, ISSN: 1860-949X, Vol.283, pp. 359-370.

5. Le Manh Ha (2010), "The lattice structure of rotor-router model", *Proceeding of IEEE-RIVF International Conference on Computing and Communication Technologies*, ISBN: 978-1-4244-8072-2, pp. 236-241.

C. Bài đăng hoàn thiện:

6. LE Manh Ha, PHAM Van Trung, PHAN Thi Ha Duong (2010), "Reachability of Conflicting Chip Firing Game and Flow network"

Các kết quả trong luận án đã được báo cáo tại các hội nghị khoa học và xemina:

- Hội nghị quốc tế về tính toán kỹ thuật cao và ứng dụng 2007 (ACOMP 2007), TP HCM 3/2007.

- Đại hội toán học toàn quốc lần thứ 7, Quy Nhơn 8/2008.

- Hội thảo tối ưu và tính toán khoa học lần thứ 7, Bà vù 22-25/4/2009.

- Hội nghị quốc tế IEEE-RIVF 2009 về Công nghệ thông tin và Truyền thông, Đà

năm 5/2009.

- Hội thảo khoa học quốc gia lần thứ IV: Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (Fair 2009), Hà nội 12/2009.
- Hội nghị quốc tế ACIIDS lần II (ACIIDS 2010): Các hệ thống thông tin và Cơ sở dữ liệu thông minh, Huế 3/2010.
- Hội thảo tối ưu và tính toán khoa học lần thứ 8, Ba vì 20-23/4/2010.
- Hội nghị quốc tế IEEE-RIVF 2010 về Công nghệ thông tin và Truyền thông, Hà nội 11/2010.
- Xemina của nhóm "Tính toán tổ hợp và các hệ động lực rời rạc" - Viện Toán học.
- Xemina của phòng "Cơ sở Toán học của Tin học" - Viện Toán học.
- Hội nghị đánh giá kết quả làm việc của nghiên cứu sinh - Viện Toán học 10/2007, 10/2008, 10/2009.

Tài liệu tham khảo

Tiếng Anh

- [1] George Andrews, Kimmo Eriksson (2004), *Integer Partitions*, Cambridge University Press, Cambridge, pp. x+141. ISBN: 0-521-84118-6; 0-521-60090-1.
- [2] P. Bak. (1997), *How nature works-The science of Self-Organized Criticality*, Oxford university press, ISBN 0-387-94791-4.
- [3] Biggs, Norman (1993), *Algebraic graph theory*, 2nd edition, Cambridge University Press, Cambridge, pp. viii+205. ISBN: 0-521-45897-8.
- [4] A. Bjorner and L. Lovász and W. Shor. (1991), "Chip-Firing Games on Graphes", *E.J. Combinatorics*, **12**, pp. 283-291.
- [5] A. Bjorner and L. Lovász. (1992), "Chip firing games on directed graphes", *Journal of Algebraic Combinatorics*, **1**, pp. 305-328.
- [6] T. Brylawski. (1973), "The lattice of interger partitions", *Discrete Mathematics*, **6**, pp. 201-219.
- [7] Bak, P. and Tang, C. and Wiesenfeld, K.(1987), "Self-Organized Criticality: An Explanation of 1/f Noise", *Physics Rewiew Letters*, **59**, pp. 381-384.
- [8] Bak, P. and Tang, C. and Wiesenfeld, K.(1988), "Self-organized criticality", *Phys. Rev. , A*, **38**, pp. 364-374.
- [9] E.Barucci, A.Del Lungo, E.Pergola, and R.Pinzani. (1998), "A methodology for plane tree enumeration", *Discrete mathematics*, **180**, pp. 45-64.
- [10] E.Barucci, A.Del Lungo, E.Pergola, and R.Pinzani. (1999), "ECO: A methodology for the enumeration of combinatorial objects", *Journal of Difference equations and Applications*, **5**, pp. 435-490.
- [11] Chris Barrett, Harry B. Hunt III, Madhav V. Marathe, S.S. Ravi, Danien J. Rosenkrantz, Richard E. Stearns (2003), "Reachability problems for sequential

- dynamical systems with threshold functions", *Theoretical computer science*, **295**, pp. 41-64.
- [12] Claude Berge (1971), "Principles of combinatorics", *Volume 72 of Mathematics in science and engineering*, Academic Press.
- [13] A. Cheng; J. Esparza and J. Palsberg (1995), "Complexity Results for 1-safe Nets", *Theor. Comput. Sci.*, (1-2) **147**, pp. 117-136.
- [14] F. Commoner; A.W. Holt; S. Even; and A. Pnueli (1971), "Marked Directed Graphs", *Journal of Computer and System Science.*, **5**, pp. 511-523.
- [15] E. Cardoza, R.J. Lipton and A. R. Meyer (1976), "Exponential Space Complete Problems for Petri Nets and Commutative Semigroups", *8th Annual Symposium on Theory of Computing.*, pp. 50-54.
- [16] Clémence Magnien. (2003), "Classes of Lattices Induced by Chip Firing (and Sandpile) Dynamics", *European Journal of Combinatorics*, (6) **24**, pp. 665-683.
- [17] C. Moore, M. Nilsson (1999), "The computational complexity of sandpile", *J. Stat. Phys.*, (6) **24**, pp. 205-224.
- [18] Rossin D., Cori R. (2000), "On the Sandpile group of Dual graphs", *E. J. Comb.*, **21**, pp. 447-459.
- [19] Jorg Desel; Javier Esparza. (1994), *Free choice Petri nets*, Cambridge University Press, In the series Cambridge Tracts in Theoretical Computer Science.
- [20] D. Dhar (1990), "Self-organized critical state of sandpile automaton models", *Phys. rev. Lett.*, **14**, pp. 1613-1616.
- [21] D. Dhar (1996), "Extended operator algebra for abelian sandpile models", *Phys. A*, **224** (1-2), pp. 162-168.
- [22] D. Dhar and S.N. Majumbar (1990), "Abelian sandpile model on the bethe lattice", *Journal of Physics*, **A23**, pp. 4333-4350.
- [23] B.A. Davey and H.A. Priestley. (1990), *Introduction to Lattices and Order*, Cambridge University Press.

- [24] E. Duchi, R. Mantaci, H. D. PHAN and D. Rossin. (2006), "Bidimensional Sand Pile and Ice Pile Models", *Pure Mathematics and Applications*, (1-2) **17**, pp. 71-96.
- [25] D. Dhar, P. Ruelle, S. Sen and D. Verma (1995), "Algebraic aspects of sandpile models," in *Journal of Physics.*, vol. A28, no. 4, pp. 805-831.
- [26] Jacques Duran. (1997), *Sables, poudres et grains*, Eyrolles sciences.
- [27] E. Goles and M.A. Kiwi (1993), "Games on line graphes and sand piles," in *Theoret. Comput. Sci.* vol. 115, pp. 321{349.
- [28] E. Goles and M.A. Kiwi (1993), "Sand-piles dynamics in a one-dimensional bounded lattice," in *Cellular automata and cooperative systems* in series *NATO Adv. Sci. Inst. Ser. C Math. Phys. Sci.*, vol. 396, pp. 211{225.
- [29] H. J. Genrich and K. Lautenbach (1979), "The analysis of distributed systems by means of predicate/ transition nets," in *Lecture Notes in Computer Science.* vol. 70, pp. 125-146.
- [30] E. Goles and M. Latapy and C. Magnien and M. Morvan and H. D. Phan. (2004), "Sandpile Models and Lattices: a Comprehensive Survey", *Theoret. Comput. Sci.*,**322**, pp. 383-407.
- [31] E. Goles and M. Morvan and H.D. Phan (2002), "About the Dynamics of Some Systems Based on Integer Partitions and Compositions", *FPSAC'2000*,**6**, pp. 214-225.
- [32] E. Goles and M. Morvan and H.D. Phan (2002), "Lattice structure and convergence of a Game of Cards", *Ann. of Combinatorics*,**6**, pp. 327-335.
- [33] E. Goles and M. Morvan and H.D. Phan (2002), "Sandpiles and order structure of integer partitions", *Discrete Appl. Math.*,**117**, pp. 51-64.
- [34] E. Goles and M. Morvan and H.D. Phan (2002), "The structure of Linear Chip Firing Game and related models", *Theoret. Comput. Sci.*,**270**, pp. 827-841.
- [35] Goldberg, Andrew V. and Tarjan, Robert E. (1988), "A new approach to the maximum-flow problem", *J. Assoc. Comput. Mach.*, (4) **35**, pp. 921-940.
- [36] Alexander E. Holroyd, Lionel Levine, Karola Mészáros, Yuval Peres, James Propp and David B. Wilson (2008), "Chip-Firing and Rotor-Routing on Directed Graphs", *In and out of equilibrium 2.*, *Progr. Probab.*, (3) **60**, pp. 331-364.

- [37] R. Howell and L. Rosier (1988), "Completeness results for conflict-free vector replacement systems", *Journal of Computer and System Sciences.*, **37**, pp. 349-366.
- [38] R. Howell, L. Rosier, D. Huynh and H. Yen (1986), "Some complexity bounds for problems concerning finite and 2-dimensional vector addition systems with states", *Theoretical Computer Science*, **46**, pp. 107-140.
- [39] R. Howell, L. Rosier and H. Yen (1993), "Normal and Sinkless Petri Nets", *Journal of Computer and System Sciences.*, **46**, pp. 1-26.
- [40] S.-T. Huang. (1993), "Leader election in uniform rings", *ACM Trans. Programming Languages Systems.*, (3) **15**, pp. 563-573.
- [41] Dung T. Huynh (1983), "Commutative Grammars: The Complexity of Uniform Word Problems", *Information and Control.*, (1) **57**, pp. 21-39.
- [42] K. Jensen, "Coloured Petri nets and the invariant method", *Theoretical Comput. Sci.*, vol. 14, pp. 317-336.
- [43] K. Jensen, "How to find invariants for coloured Petri nets." in *Lecture Notes in Computer Science*, vol. 118, pp. 327-338.
- [44] H.J. Jensen. (1998), *Self-Organised Criticality*, Cambridge university press.
- [45] N.D. Jones; L.H. Landweber and Y.E. Lien (1977), "Complexity of Some Problems in Petri Nets", *Theoretical Computer Science.*, **4**, pp. 277-299.
- [46] R.M. Keller (1975), "A Fundamental theorem of Asynchronous Parallel Computation", *JParallel Processing.*, **LNCS 24**, pp. 102-112.
- [47] S. R. Kosaraju. (1982), "Decidability of reachability in vector addition systems", in *Proc. 74th Annual ACM Symp. Theory Computing, San Francisco.*, **AA**, pp. 267-281.
- [48] Latapy, Matthieu. (2001), "Partitions of an integer into powers", *Discrete Math. Theor. Comput. Sci. Proc.*, **AA**, pp. 215 - 228 (electronic).
- [49] Le Manh Ha (2010), "The lattice structure of rotor-router model", *Proceeding of IEEE-RIVF International Conference on Computing and Communication Technologies*, ISBN: 978-1-4244-8072-2, pp. 236-241.

- [50] R. J. Lipton. (1976), "The reachability problem requires exponential space", *New Haven, CT, Yale University, Department of Computer Science, Res. Rep.* 62.
- [51] M. Latapy, R. Mataci, M. Morvan, and H.D. Phan. (2001), "Structure of some sand piles model", *Theoret. Comput. Sci.*, **262**, pp. 525-556.
- [52] M. Latapy and H.D. Phan. (2001), "The Lattice structure of Chip Firing Games", *Physica D*, **115**, pp. 69-82.
- [53] LE Manh Ha; NGUYEN Anh Tam; PHAN Thi Ha Duong.(2010), "Algorithmic aspects of the Reachability of Conflicting Chip Firing Game", *Advances in Intelligent Information and Database Systems* in series *Studies in Computational Intelligence*, Springer, ISSN: 1860-949X, Vol.283, pp. 359-370.
- [54] Minh Ha LE and Thi Ha Duong PHAN. (2007), "Strict partitions and discrete dynamical systems", *Theoretical Computer Science.*, **389**, pp. 82-90.
- [55] Le Manh Ha and Phan Thi Ha Duong (2008), "Order structure and energy of conflicting Chip Firing Game", *Acta Math. Vietnam.*, (2) **35**, pp. 289-301.
- [56] Le Manh Ha and Phan Thi Ha Duong (2009), "Integer partitions in discrete dynamical models and ECO method", *Vietnam J. Math.*, (2-3) **37**, pp. 273-293.
- [57] Matthieu Latapy and Thi Ha Duong Phan (2009), "The lattice of integer partitions and its infinite extension ", *Discrete mathematics*, **309**, pp. 1357-1367.
- [58] LE Manh Ha, PHAM Tra An, PHAN Thi Ha Duong (2009), "On the relation between Chip Firing Games and Petri Nets", *Proceeding of IEEE-RIVF International Conference on Computing and Communication Technologies*, ISBN: 978-1-4244-4567-7, pp. 328-335.
- [59] LE Manh Ha, PHAM Van Trung, PHAN Thi Ha Duong (2010), "Reachability of Conflicting Chip Firing Game and Flow network", *submitted to DMTCS*.
- [60] E. W. Mayr.(1984), "An algorithm for the general Petri net reachability problem", *SIAM, J. Comput.*, (3) **13**, pp. 441-460.
- [61] Merino López; Criel.(1997), "Chip firing and the Tutte polynomial", *Annals of Combinatorics*, (3) **1**, pp. 253-259.

- [62] Bernard Monjardet (1990) "The consequences of dilworth's work on lattices with unique irreducible decompositions", *In K.P. Bogart, R.Freese and J.Kung editors. The Dilworth theorems Selected papers of Robert P.Dilworth*, Birkhauser, Boston, pages 192-201.
- [63] Nami Mizumo; Atsushi Ohta; Kohkichi Tsuji (2008), "Reachability problem of state machines with batch processing arcs", *Proc. The 23rd International Technical Conference on Circuits/Systems, Computers and Communication (ITC-CSCC 2008)*, pp. 293-296.
- [64] C. Magnien and H. D. Phan and L. Vuillon. (2001), "Characterization of Lattices Induced by (extended) Chip Firing Games", *Discrete Math. Theoret. Comput. Sci.*, **AA**, pp. 229-244.
- [65] T. Murata.(1989), "Petri nets: properties, analysis and applications", *Proceedings of the IEEE*, (4) **77**, pp. 541-580.
- [66] T. Murata.(1977), "State equation, controllability, and maximal matchings of Petri nets", *IEEE Trans. Automat. Contr.*, (3) **AC-2**, pp. 412-416.
- [67] Petri, C.A., Bonn (1962) "Kommunikation mit Automaten" : *Institut fur Instrumentelle Mathematik, Schriften des IIM Nr. 2*, Second Edition:, New York: Griffiss Air Force Base, Technical Report RADC-TR-65{377, Vol.1, 1966, Pages: Suppl. 1, English translation
- [68] H. J. Genrich and K. Lautenbach (1979), "The analysis of distributed systems by means of predicate/ transition nets," in *Lecture Notes in Computer Science*. vol. 70, pp. 125-146.
- [69] V.B.Priezzhev, D.Dhar, A. Dhar and S. Krishnamurthy (1996), "Eulerian walkers as a model of selforganised criticality", *Phys. Rev. Lett.*, **77**, pp. 5079-5082.
- [70] Thi Ha Duong PHAN (2008), "Two sided Sand Piles Model and unimodal sequences", *RAIRO-Theor. Inf. Appl.*, **42**, pp. 631-646.
- [71] Tra An PHAM; Thi Ha Duong PHAN and Thi Thu Huong TRAN (2007), "Conflicting Chip Firing Games on directed graphs and on tree", *VNU Journal of Science. Natural Sciences and Technology* , **24**, pp. 103-109.

- [72] PHAN Thi Ha Duong and TRAN Thi Thu Huong (2006), "Stable Sand Piles Model", *Discrete Math. Theor. Comput. Sci. Proc.*, **AG**, pp. 407-41.
- [73] Reinhard Diestel. (2005), *Graph Theory*, Electronic Edition.
- [74] Felsner, Stefan; Knauer, Kolja B.(2009), "ULD-lattices and Δ -bonds," *Combin. Probab. Comput.*, (5) **18**, 707-724.
- [75] Stanley, Richard P. (1998), *Enumerative combinatorics*, Cambridge University Press (Vol. 1).
- [76] Stanley, Richard P. (1999), *Enumerative combinatorics*, Cambridge University Press (Vol. 2).
- [77] I.A.Stewart. (1992), "On the reachability problem for some classes of Petri nets", *Research Report*, University of Newcastle upon Tyne.
- [78] C. Tang (1993), *Self-Organised Criticality*, OCPA Newsletter.
- [79] Tuncel, Levent (1994), "On the complexity of preflow-push algorithms for maximum-flow problems", *Algorithmica*, (4) **11**, pp. 353-359.

Tiếng Việt

- [80] Ngô Đắc Tân (2003), *Lý thuyết tổ hợp và đồ thị*, NXB Đại học quốc gia Hà nội.