数理解析研究所講究録
第333巻 1978年 89-96

On the Number of Chains in a Hasse Diagram

by

Hiroshi Narushima

Department of Mathematical Sciences, Faculty of Science, Tokai University,

Hiratsuka, Kanagawa, Japan

Abstract

A finite partially ordered set P is represented by the Hasse diagram H(P). Therefore an enumeration of chains in P can be carried out by enumerating chains in H(P). In the sequel P and H(P) are identified. Some simple algorithms for enumerating chains in a Hasse diagram are given. Then the algorithms are applied to obtain some recurrence formulas for enumerating non-isomorphic unlabeled rooted trees.

§1. 序

順序集合上の和積定理を応用するにあたって、つぎ点が問題となる。

(1) 与えられた順序集合 P と写像 $f: P \longrightarrow \wp(\Omega)$ に対して、f が P 上の弱射であるかどうか。

(2) 鎖 C に対して、$m(\bigcap_{x \in C} f(x))$ または $m(\bigcup_{x \in C} f(x))$ が算術的に計算可能かどうか。

(3) P に含まれる鎖に関する計算 $\sum_{C \in \mathcal{C}} (-1)^{\ell(C)} m(\quad)$ でより経済的な方法があるかどうか。

(4) (3)の計算量のひとつの目安となる"与えられた順序集合Pに含まれる鎖の個数"を有効に求めるアルゴリズムがあるかどうか、などである。

本講演は問題(4)を考察するものである。§2で、一般のハッセ図に含まれる鎖の数え上げまたは"加減数え上げ"、及び、特に木に含まれる鎖の数え上げと鎖の個数にもとづく木の分類漸化式について述べ、§3で、一般のハッセ図に含まれる極大鎖の数え上げまたは"加減数え上げ"と極大鎖の数え上げにもとづく木の一般分類漸化式について述べる。

§2. 鎖の数え上げ、加減数え上げ、鎖の個数にもとづく木の分類漸化式　始めに、一般の順序集合に含まれる鎖の数え上げをアルゴリズムAで、鎖の加減数え上げをアルゴリズムBで述べ、つぎに、特に、木に含まれる鎖の数え上げに有効であるアルゴリズムCを与え、更に、アルゴリズムCの値づけにもとづく木の分類漸化式について述べる。

Pをハッセ図とする。以下、Pの元xに対して、

$P_x = \{ y \in P \mid y < x \}$、 $P'_x = \{ y \in P \mid x \downarrow y$ (yはxの直下の元)$\}$

とする。

アルゴリズムA(鎖の数え上げ). Pをハッセ図とする。Pの極小元に値1を与える。Pの極小元以外の元xに対して、

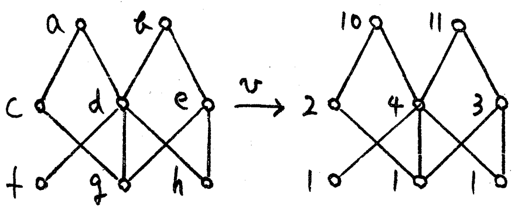
P'_xのすべての元が値をもっているならば、xに値 $\sum_{y\in P'_x} v(y)+1$ を与える。ここで、$v(y)$はyの値を示す。この値づけをPのすべての元が値をもつまでつづける。

定理1. Pを順序集合とする。

(1) Pの元xに対して、$v(x)$はxを最大元とする鎖の個数である。

(2) $\tilde{c}$でPに含まれる鎖の個数を表わすとき、$\tilde{c}=\sum_{x\in P} v(x)$ となる。

例1. アルゴリズムAによる値づけの例



$v(f)=v(g)=v(h)=1$, $v(c)=v(g)+1=2$
$v(d)=v(f)+v(g)+v(h)+1=4$
$v(e)=v(g)+v(h)+1=3$
$v(a)=v(c)+v(d)+v(f)+v(g)+v(h)+1=10$
$v(b)=v(d)+v(e)+v(f)+v(g)+v(h)+1=11$
$\tilde{c}=33$

アルゴリズムB(鎖の加減数え上げ). つぎの点を除いてアルゴリズムAと同じである。ハッセ図Pの極小元以外の元xに対して、P'_xのすべての元が値をもっているならば、xに値 $-(\sum_{y\in P_x} v_1(y))+1$ を与える。ここで、$v_1(y)$はyの値である。

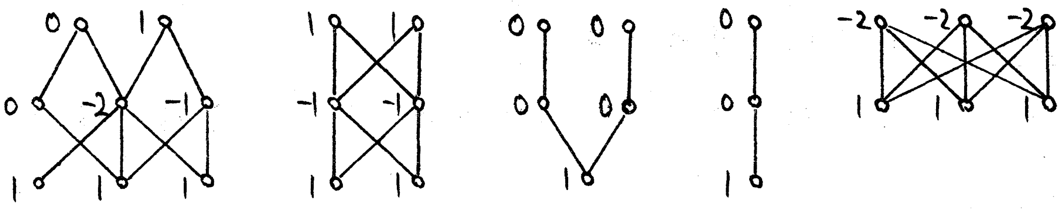
定理2. Pを順序集合とする。

(1) c_i^xでxを最大元とする長さiの鎖の数を表わすとき、Pの元xに対して、$v_1(x)=\sum_{0\le i}(-1)^i c_i^x$ が成り立つ。

(2) c_iでPに含まれる長さiの鎖の数を表わすとき、

$\sum_{0\le i}(-1)^i c_i = \sum_{x\in P} v_1(x)$ が成り立つ。

例2. 例2のハッセ図に対するアルゴリズムBによる値づけとその他の例



$\sum_{0\le i}(-1)^i c_i =$ 1、 2、[文献3] 1、 1、 −3

注意. 予稿1)のPropositionの(1)によって、最大元あるいは最小元をもつハッセ図に対しては、常に $\sum_{0\le i}(-1)^i c_i = 1$ である。

さて、例4の木に対して、アルゴリズムAを適用すれば、4.1のように値づけられるが、特に、木に含まれる鎖の数を求めるときには、つぎのアルゴリズムが有効である。

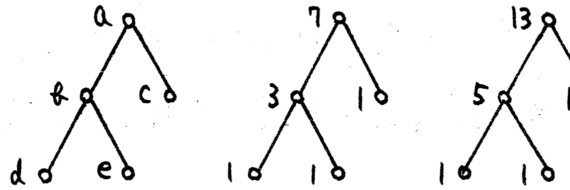
アルゴリズムC(木の鎖の数え上げ). つぎの点を除いてアルゴリズムAと同じである。木Tの極小元以外の元xに対して、P'_xのすべての元が値をもっているならば、xに値 $2\left(\sum_{y\in P'_x} v_2(y)\right)+1$ を与える。ここで、$v_2(y)$はyの値である。

定理3. Tを木(rooted tree)とする。

(1) Tの元xに対して、$v_2(x)$はx以下の(Tの)部分木に含まれる鎖の数である。

(2) Tに含まれる鎖の数を$\tilde{c}$で表わし、Tの最大元を1_Tで表わすとき、$\tilde{c} = v_2(1_T)$が成り立つ。

例3. 木に対するアルゴリズムAとCによる値づけの比較



$v_2(c) = v_2(d) = v_2(e) = 1$
$v_2(b) = 2(v_2(d) + v_2(e)) + 1 = 5$
$v_2(a) = 2(v_2(b) + v_2(c)) + 1 = 13$
$\tilde{C} = v_2(1_T) = v_2(a) = 13$

4.1
注意. アルゴリズムAとCの値づけのちがい：

$$\text{(A)}\quad v(x) = \sum_{y \in P_x} v(y) + 1 \qquad \text{(C)}\quad v_2(x) = 2\Big(\sum_{y \in P'_x} v_2(y)\Big) + 1$$

つぎに、アルゴリズムCの値づけにもとづく、同型でないunlabelな木の分類漸化式を述べる。

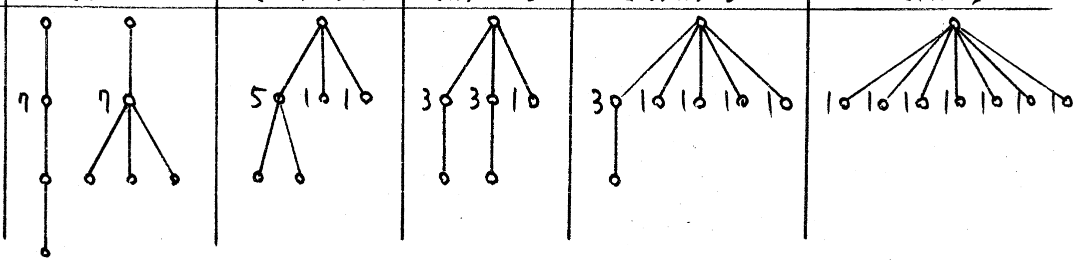
定理4. 正の整数nに対して、$v_2(1_T) = 2n+1$となる木Tの数を$f(2n+1)$で表わし、数nの奇数和因子への分割全体を$p^{(1)}(n)$で表わすとき、つぎの漸化式が成立する。

$$f(2n+1) = \sum_{x \in p^{(1)}(n)} \prod_{i=1}^{k} f(2i-1) H \lambda_{2i-1}$$

ここで、$x = \{(1)^{\lambda_1}(2)^{\lambda_2} \cdots (2i-1)^{\lambda_{2i-1}} \cdots (2k-1)^{\lambda_{2k-1}}\}$、
$\sum_{i=1}^{k} \lambda_{2i-1}(2i-1) = n$、であり、Hは重複組合せを表わす。

例4. 鎖の数が15となる木：$v_2(1_T) = 2n+1 = 15$、$n = 7$

$p^{(1)}(7)$	$\{(7)\}$	$\{(5)(1)^2\}$	$\{(3)^2(1)\}$	$\{(3)(1)^4\}$	$\{(1)^7\}$
$f(15)$					



例5. $f(2n+1)$の計算例

$2n+1$	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29
$f(2n+1)$	1	1	1	2	2	3	4	6	7	10	13	17	22	29	38

§3．極大鎖の数え上げ、加減数え上げ、木の分類漸化式

ハッセ図Pに含まれる鎖 $x_0 < x_1 < \cdots < x_i < x_{i+1} < \cdots < x_k$ を、$\forall i\ (0 \le i \le k-1)$ に対して x_{i+1} が x_i の直上の元 $(x_i \uparrow x_{i+1})$ であるとき、極大鎖という。

アルゴリズムD（極大鎖の数え上げ）．つぎの点を除いてアルゴリズムAと同じである。ハッセ図Pの極小元以外の元 x に対して、P'_x のすべての元が値をもっているならば、x に値 $\sum_{y \in P'_x} v'(y) + 1$ を与える。$v'(y)$ は y の値である。

定理5．　Pを順序集合とする。

(1)　Pの元 x に対して、$v'(x)$ は x を最大元とする極大鎖の数である。

(2)　Pに含まれる極大鎖の数を $\tilde{d}$ で表わすと、$\tilde{d} = \sum_{x \in P} v'(x)$ となる。

アルゴリズムE（極大鎖の加減数え上げ by H. Era (恵羅)）．つぎの点を除いてアルゴリズムAと同じである。ハッセ図Pの極小元以外の元 x に対して、P'_x のすべての元が値をもっているならば、x に値 $-(\sum_{y \in P'_x} v_1'(y)) + 1$ を与える。$v_1'(y)$ は y の値である。

定理6．　Pを順序集合とする。

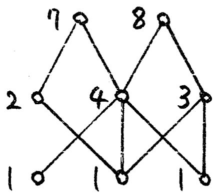
(1)　d_i^x で x を最大元とする長さ i の極大鎖の数を表わすとき、$v_1'(x) = \sum_{0 \le i} (-1)^i d_i^x$ が成り立つ。

(2)　d_iでPに含まれる長さiの極大鎖の数を表わすとき、

$\sum_{0\leq i}(-1)^i d_i = \sum_{x\in P} v_1'(x)$ が成り立つ。

<u>注意</u>、アルゴリズムA，BとアルゴリズムD、Eのちがいは、P_xの元についての和がP_x'の元についての和となっていること。

<u>例6</u>、　例1のハッセ図に対するアルゴリズムD及びEによる値づけ



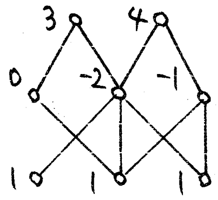
f, g, h, c, d, e に対してはAと同じ。

$v'(a) = v'(c) + v'(d) + 1 = 7$

$v'(b) = v'(c) + v'(e) + 1 = 8$

$\tilde{\alpha} = 27$

<u>Dによる値づけ</u>



f, g, h, c, d, e に対してはBと同じ

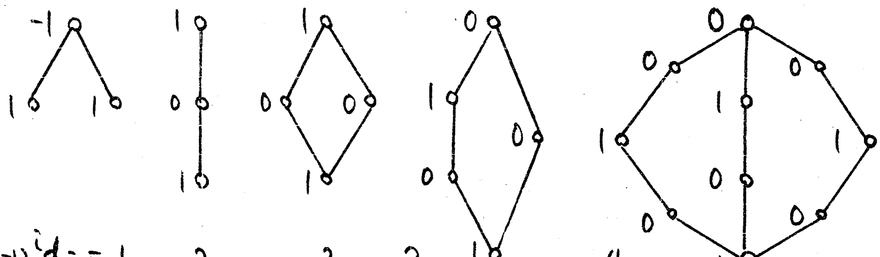
$v_1'(a) = -(v_1'(c) + v_1'(d)) + 1 = 3$

$v_1'(b) = -(v_1'(c) + v_1'(e)) + 1 = 4$

$\sum_{0\leq i}(-1)^i d_i = 7$

<u>Eによる値づけ</u>

<u>注意</u>、極大鎖に関して文献(予稿1)[4]のPropositionの(1)は成立しない。最大元をもつハッセ図に対して、$\sum_{0\leq i}(-1)^i d_i$ = 一定とは限らない。



$\sum_{0\leq i}(-1)^i d_i$ = <u>1</u>、 <u>2</u>、 <u>2</u>、 <u>2</u>、 <u>4</u>.

さて、アルゴリズムDによる木の分類を考えるに当って、つぎの補題が重要である。

<u>補題</u>、　木Tの元xに対して、$v'(x)$はxを最大元とする(Tの)部分木の元の個数に等しい。

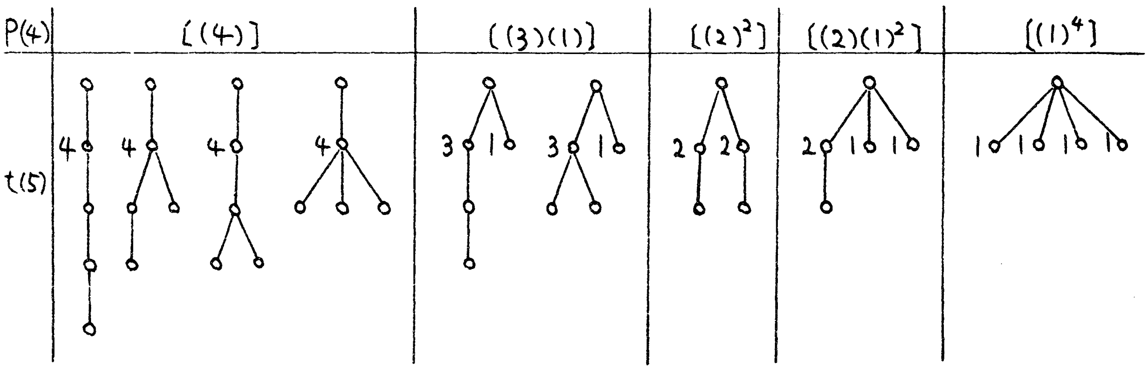
定理7.　正の整数nに対して、$v(\mathbb{D}_T)=n$となる木、すなわち、点の個数がnである木の数を$t(n)$で表わし、数nの分割全体を$P(n)$で表わすとき、つぎの漸化式が成立する。

$$t(n+1)=\sum_{x\in P(n)}\prod_{k=1}^{n}{}_{t(k)}H_{\lambda_k}$$

ここで、$x=[(1)^{\lambda_1}(2)^{\lambda_2}\cdots(k)^{\lambda_k}\cdots(n)^{\lambda_n}]$、$\sum_{k=1}^{n}k\lambda_k=n$であり、Hは重複組合せを表わす。

注意.　$t(n)$の(上の)漸化式はアルゴリズムDによらず自然に考えることもできる。

例7.　点の数が5である木



文献

1. 成島　弘、組合せ理論の基礎、数解研講究録 179、1-18、1973.
2. 成島　弘、和積定理とReduced Mapsについて、同259、44-61、1976.
3. 成島　弘、Hasse図に含まれる鎖の個数について、LA sym. 論文集、1977. to appear.
4. H. Narushima, Principle of Inclusion-Exclusion on Partially Ordered Sets, to appear.
5. F. Harary and E. M. Palmer, Graphical Enumeration, Academic Press, 1973.