

Attribute Graph Grammar and Tabular Forms

有田 友和

富山 聖宜

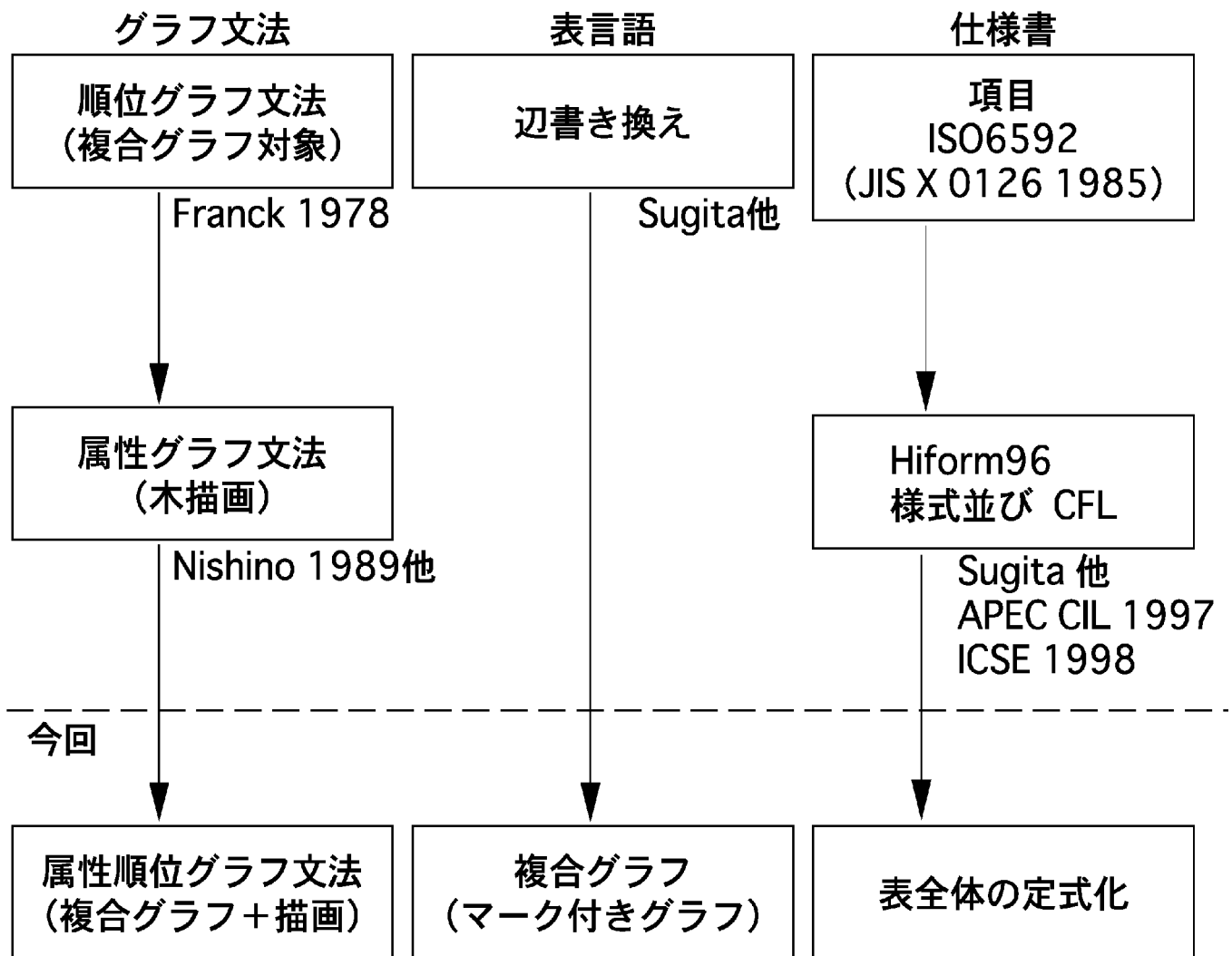
日本大学大学院総合基礎科学研究科

地球情報数理科学専攻

2000年1月31日

1. [はじめに]

背景



動機

- ・ 仕様書の機械的処理（描画，編集）が困難
- ・ 定式化が厳密ではない

目的

仕様書を属性順位グラフ文法を用いて定式化

- ・ 仕様書の定式化の方法を提案する
- ・ 項目の配置，大きさをある程度自由
- ・ 文法による機械的な描画

結果

- ・ 表をマーク付きグラフで表す.
- ・ 属性順位グラフ文法のクラスを定義する.
- ・ 仕様書を定式化する属性グラフ文法 HFAGG を構成する.
生成規則 280 個.
描画のための属性を各生成規則に加える.
- ・ HFAGG が属性順位グラフ文法であることを示す.

2. 準備

2.1 仕様書 Hiform96[2,3]

— Hiform96 の特徴 —

- ・ JISX0126-1987 のガイドライン項目を全て含む
- ・ 記述形式は表形式
 - 関連項目を A4 サイズ 1 ページに配置 (様式)
 - 様式は 17 種類
 - 作成・管理等の目的に適している
- ・ 各様式の並べ方は句構造文法によって定義
- ・ 様式内の項目の配置は固定

2.1 仕様書 Hiform96 (つづき)

Hiform96 の様式の例

(A 1. プログラム概要書)

プロジェクトコード:	A
プログラム名:	プログラム概要書
ライブラリ登録コード:	版名:
著作者:	初版発行日:
文書責任者:	現行版発行日:
キーワード:	CR分類コード:
目的・範囲:	
背景情報:	
記述言語:	所要ソフトウェア:
操作:	所要ハードウェア:
関連文書:	
機能:	
例:	

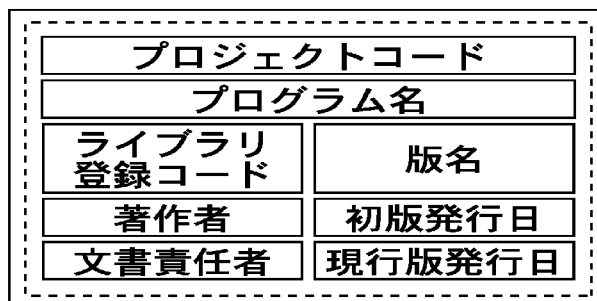
2.2 マーク付きグラフ

例. Hiform96 様式とマーク付きグラフ

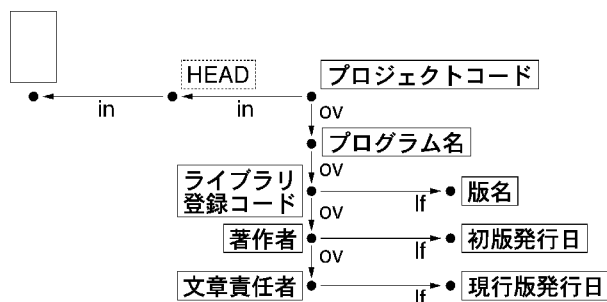
表 2.2 仕様書の一部

プロジェクトコード:	
プログラム名:	
ライブラリ登録コード:	版名:
著作者:	初版発行日:
文書責任者:	現行版発行日:

対応する複合グラフ



対応するマーク付きグラフ



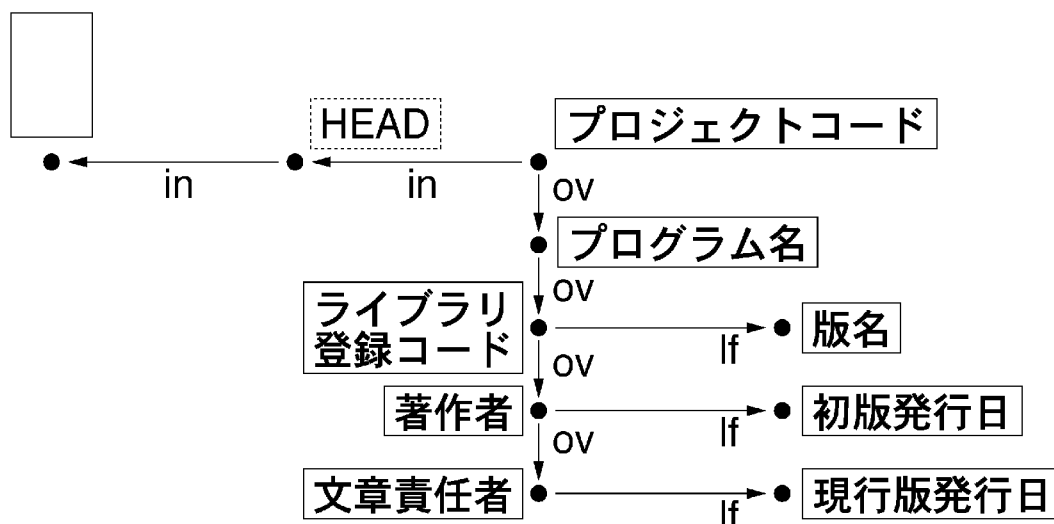
2.2 マーク付きグラフ (つづき)

定義1 マーク付きグラフ [1]

マーク付きグラフ (marked graph) は $G = (K, R, k, r)$ の4項組である。ただし、

- ・ K : ノードの有限集合 ($K \neq \phi$)
- ・ $R \subset K \times K$
- ・ $k : K \rightarrow V$ (ノードを マーク 付けする写像)
- ・ $r : R \rightarrow M$ (辺を ラベル 付けする写像)

□



2.3 順位グラフ文法 [1]

2.3.1 文脈自由グラフ文法

定義 2 文脈自由生成規則

文脈自由生成規則(context-free production)

は $p = (A, H, p^e, p^s)$ の 4 項組である. ただし,

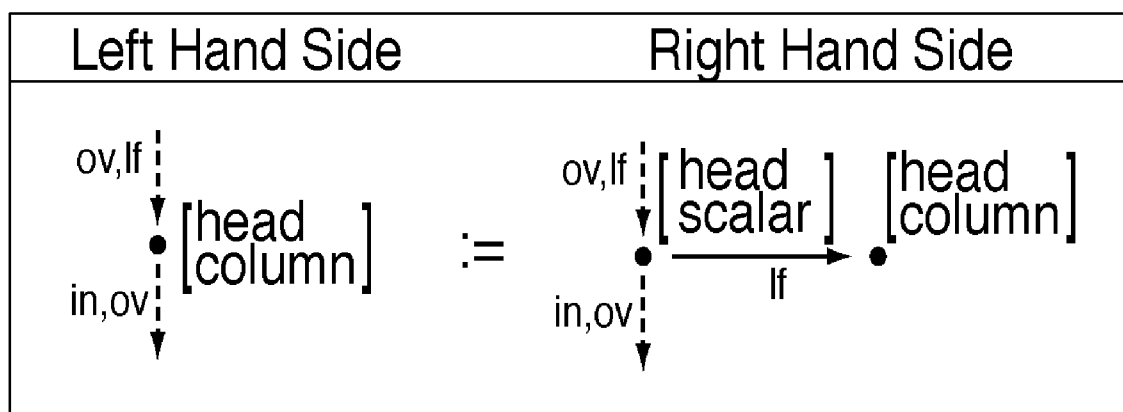
A : 一つの頂点のグラフ (p の左側)

$H = (K_h, R_h, k_h, r_h)$: 空でないグラフ (p の右側)

$p^e, p^s : M \rightarrow K_h$ (辺の全てのラベルの集合 M から K_h への部分関数)

□

生成規則の例



2.3.1 文脈自由グラフ文法（つづき）

定義3 文脈自由グラフ文法

文脈自由グラフ文法

(a context-free graph grammar)

は $GG = (V, T, M, P, S)$ の5項組である。ただし、

V : 頂点をマーク付けする記号の有限集合

T : 終端記号 ($\subset V$)

M : 辺のラベルの有限集合

P : 生成規則 $p = (A, H, p^e, p^s)$ の有限集合

S : 開始記号 ($\in V - T$)

□

2.3.2 順位関係

記法4

$m \in M$ に対し,

$$\dot{=}^m \stackrel{def}{=} \left\{ (A, B) \left| \begin{array}{l} A, B \in V \text{ かつ} \\ \text{右辺に辺 } (x, y) \text{ を持つ規則が} \\ \text{存在し } x, y \text{ はそれぞれ } A, B \\ \text{とマーク付けされ,} \\ (x, y) \text{ はラベル } m \text{ を持つ} \end{array} \right. \right\}$$

□

記法5

$m \in M$ に対し,

$$\rightarrow^m \stackrel{def}{=} \left\{ (A, B) \left| \begin{array}{l} A, B \in V \text{ かつ} \\ \text{生成規則 } p = (A, H, p^e, p^s) \\ \text{が存在かつ } B \text{ は } H \text{ の頂点} \\ p^e(m) \text{ のマークである} \end{array} \right. \right\}$$

□

2.3.2 順位関係 (つづき)

記法6

$m \in M$ に対し, $\leftarrow \cdot_m \stackrel{\text{def}}{=} \dot{\leftarrow}_m \cdot \overset{+}{\rightarrow}_m$ ただし $+$ は推移的閉包を表す

□

記法7

$m \in M$ に対し,

$\leftarrow_m \stackrel{\text{def}}{=} \left\{ (B, A) \right.$	$\left. \begin{array}{l} A, B \in V \text{ かつ 生成規則} \\ p = (A, H, p^e, p^s) \\ \text{が存在かつ } B \text{ は } H \text{ の 頂点} \\ p^s(m) \text{ の マークである} \end{array} \right\}$
--	---

□

記法8

$m \in M$ に対し,

$$\cdot \rightarrow_m \stackrel{\text{def}}{=} \overset{+}{\leftarrow}_m \cdot \dot{\rightarrow}_m$$

□

2.3.2 順位関係 (つづき)

記法 9

$m \in M$ に対し,
 $\langle \cdot \rangle_m \stackrel{def}{=} \overleftarrow{\cdot}_m \cdot \dot{\cdot}_m \cdot \overrightarrow{\cdot}_m$

□

定義 10 順位文法が矛盾しない

順位文法が 矛盾しない (conflictless) とはすべての $m \in M$ に対して関係 $\langle \cdot \rangle_m, \dot{\cdot}_m, \cdot \rangle_m, \langle \cdot \rangle_m$ が互いに素であることである

□

定義 11 順位関係

順位関係 (precedence relation) は以下のように定義される

$$\langle \cdot \rangle \stackrel{def}{=} \bigcup_{m \in M} \langle \cdot \rangle_m \quad \cdot \rangle \stackrel{def}{=} \bigcup_{m \in M} \cdot \rangle_m$$

$$\langle \cdot \rangle \stackrel{def}{=} \bigcup_{m \in M} \langle \cdot \rangle_m \quad \dot{\cdot} \stackrel{def}{=} \bigcup_{m \in M} \dot{\cdot}_m$$

□

2.3.3 順位グラフ文法の定義

定義 12 順位グラフ文法

文脈自由グラフ文法 $GG = (V, T, M, P, S)$ が 順位グラフ文法 (a precedence graph grammar) $\stackrel{def}{\iff}$ 次の3つを満たすことである

- (i) 順位関係に矛盾がない
- (ii) 各還元が一意に定まる
- (iii) 文法において再帰的非終端記号が存在しない

□

— 定義 13 属性グラフ文法 [6,7,8] —

属性グラフ文法 (AttributeGraphGrammar) は、次の条件を満たす 3 項組 $AGG = (GG_u, A, F)$ である。

(1) $GG_u = (V, T, M, P, S)$ は、 AGG の 基底文脈自由グラフ文法。

また、 V 上のグラフ $G = (K, R, k, r)$ について、 $Lab(G) = \{k(n) | n \in K\}$ 。

(2) GG_u の各元 $X \in V$ に対して互いに素な 2 つの有限集合である、継承属性 の集合 $I(X)$ と 合成属性 の集合 $S(X)$ が付随している。

X の属性全体の集合を $A(X) = I(X) \cup S(X)$ で表す。

$A = \cup_{X \in V} A(X)$ を GG_u の 属性集合。

ただし、 $I(S) = \phi$ 。

また、 X の属性 a を $a(X)$ 、 a が取り得る値全体の集合を $V(a)$ 。

— 定義13 (つづき) —

(3) P の各プロダクション $p = (X_0, H, p^e, p^s)$ に対して, $S(X_0) \cup_{X \in \text{Lab}(H)} I(X)$ に含ませる全ての属性を定義するような 意味規則 の集合 F_p が付随している.

属性 $a(X_k) (0 \leq k \leq n)$ を定義する意味規則は次のような形をしている.

$$a(X_k) := F(a_1(X_{i_1}), \dots, a_m(X_{i_m})),$$

$$0 \leq i_j \leq |\text{Lab}(D)|,$$

$$X_{i_j} \in \text{Lab}(D), 0 \leq j \leq m.$$

ただし, f は以下の型の関数とする.

$$V(a_1(X_{i_1})) \times \dots \times V(a_m(X_{i_m})) \rightarrow V(a(X_k))$$

このとき, $a(X_k)$ は p において $a_j(X_{i_j}) (1 \leq j \leq m)$

に 依存する という.

集合 $F = \cup_{p \in P} F_p$ を AGG の 意味規則集合 という.

3.[結果]

要約

- 属性順位グラフ文法を定義した.
- 仕様書 Hiform96 を属性順位グラフ文法を用いて定義した
- HFAPGG のサイズは, 生成規則 280 個, 順位関係約 15000 個
- 各生成規則に仕様書描画のための属性を付属する.
- 格子状の表についても属性グラフ文法を用いて定式化を行う.

3.1 属性順位グラフ文法

定義 14 属性順位グラフ文法

属性グラフ文法 $AGG = (GG_u, A, F)$ が
属性順位グラフ文法.

\xLeftrightarrow{def}

GG_u が順位グラフ文法.

3.2 Hiform2000のための属性順位グラフ文法

3.2.1 HFAPGGの生成規則（一部）

Production	No. 1
The picture for this production	
Semantic rule	
$x(1)=0$ $y(1)=0$ $x(2)=x(1)$ $y(2)=y(1)$	$width(1) = width(2)$ $height(1)= height(2)$

Production	No. H1
The picture for this production	
Semantic rule	
$x(1)=0$ $y(1)=0$ $x(2)=x(1) + HMleft$ $y(2)=y(1) + HMtop$	$width(0) = width(1)$ $height(0)= height(1)$ $width(1) = width(2)$ $+HMleft+HMright$ $height(1)=height(2)$ $+HMtop+HMbottom$

Production	No. 2
The picture for this production	
Semantic rule	
$x(1)=x(0) + Mleft$ $y(1)=y(0) + Mtop$ $x(2)=x(1) + Mleft$ $y(2)=y(1) + Mcen$	$width(0) = \max(width(1),width(2))$ $height(0)= height(1)+height(2) +Mtop+Mcen+Mbottom$

Production	No. H2
The picture for this production	
Semantic rule	
$x(1)=x(0)$ $y(1)=y(0)$ $x(2)=x(1)$ $y(2)=y(1) + HM1$	$width(0) = \max(width(1),width(2))$ $height(0)= height(1)+height(2) +HSv$

3.2.2 HFAPGGGの順位関係表 (一部)

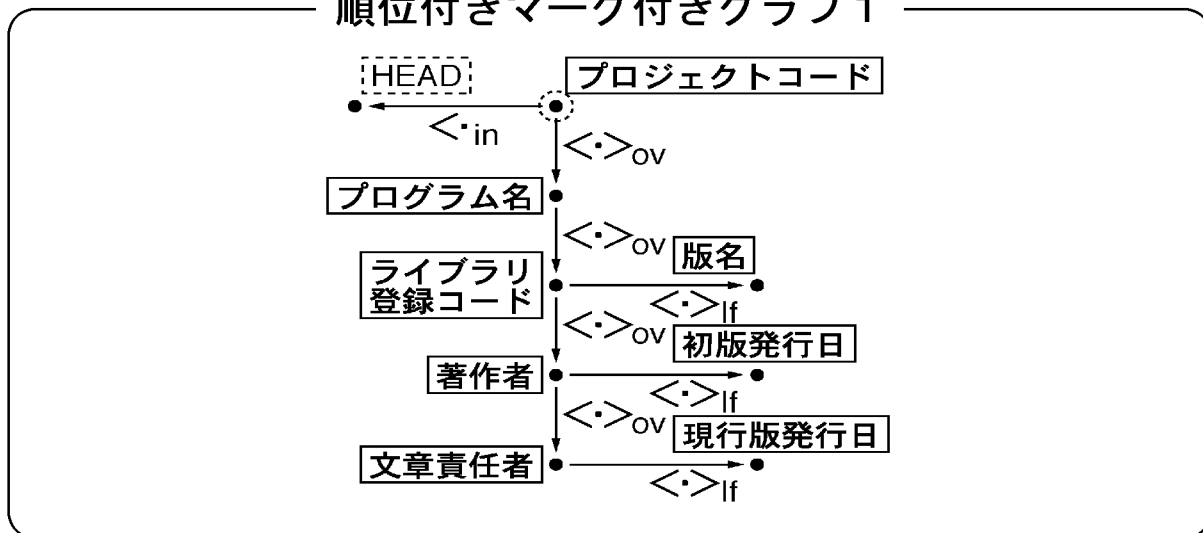
	Right			[head scalar]			[head column]			[head row]			[head root]			['HEAD']			[head]			
	in	ov	If	in	ov	If	in	ov	If	in	ov	If	in	ov	If	in	ov	If	in	ov	If	
共通部分終端記号		<.>	<.>				<.>	>		<.>	>		>	>								
[head scalar]		<.>	<.		<.		<.>	=		<.>	>		>	>								
[head column]		<.>	<.>		<.>		<.>	>		<.>	>		>	>								
[head row]		<.					<.			<.			<.									
[head root]													=									
[head]																						
['HEAD']																						

3.2.3 HFAPGGのサイズ

様式	生成規則数	順位関係表
共通部分	16	14×42,29×87
A1	18	17×51
A2	13	12×36
A3	13	12×36
A4	14	13×39
A5	10	9×27
A6	8	7×21
B共通	10	7×21,8×24
B1	8	7×21
B2	24	23×69
B3	25	24×72
C共通	10	7×21,12×36
C1	8	7×21
C2	21	20×60
C3	23	22×66
C4	10	9×27
C5	8	7×21
C6	8	7×21
D1	8	7×21
D2	25	24×72
合計	280	15330セル

3.3 HFAPGGによる構文解析

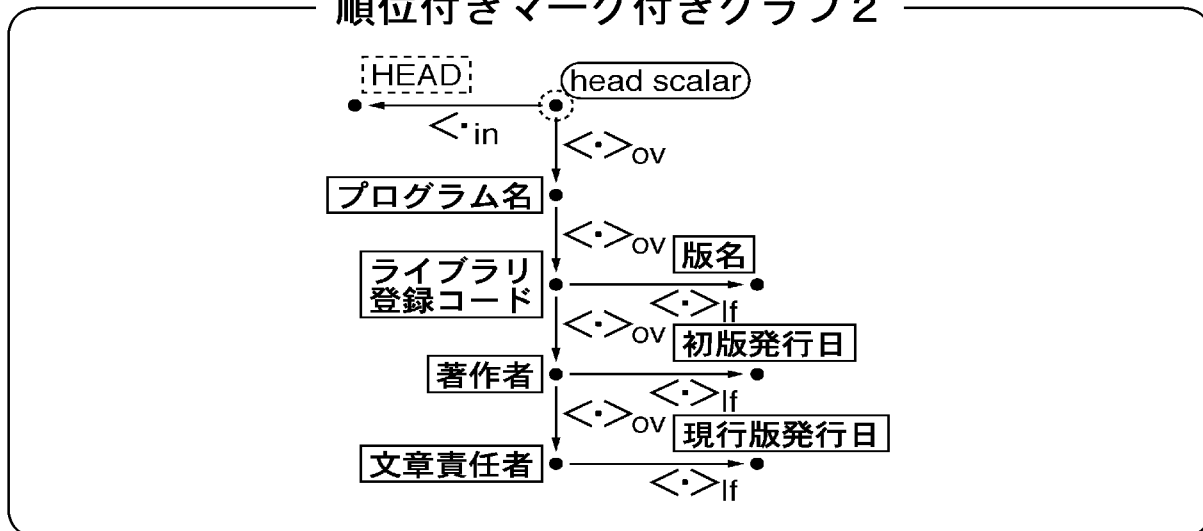
順位付きマーク付きグラフ 1



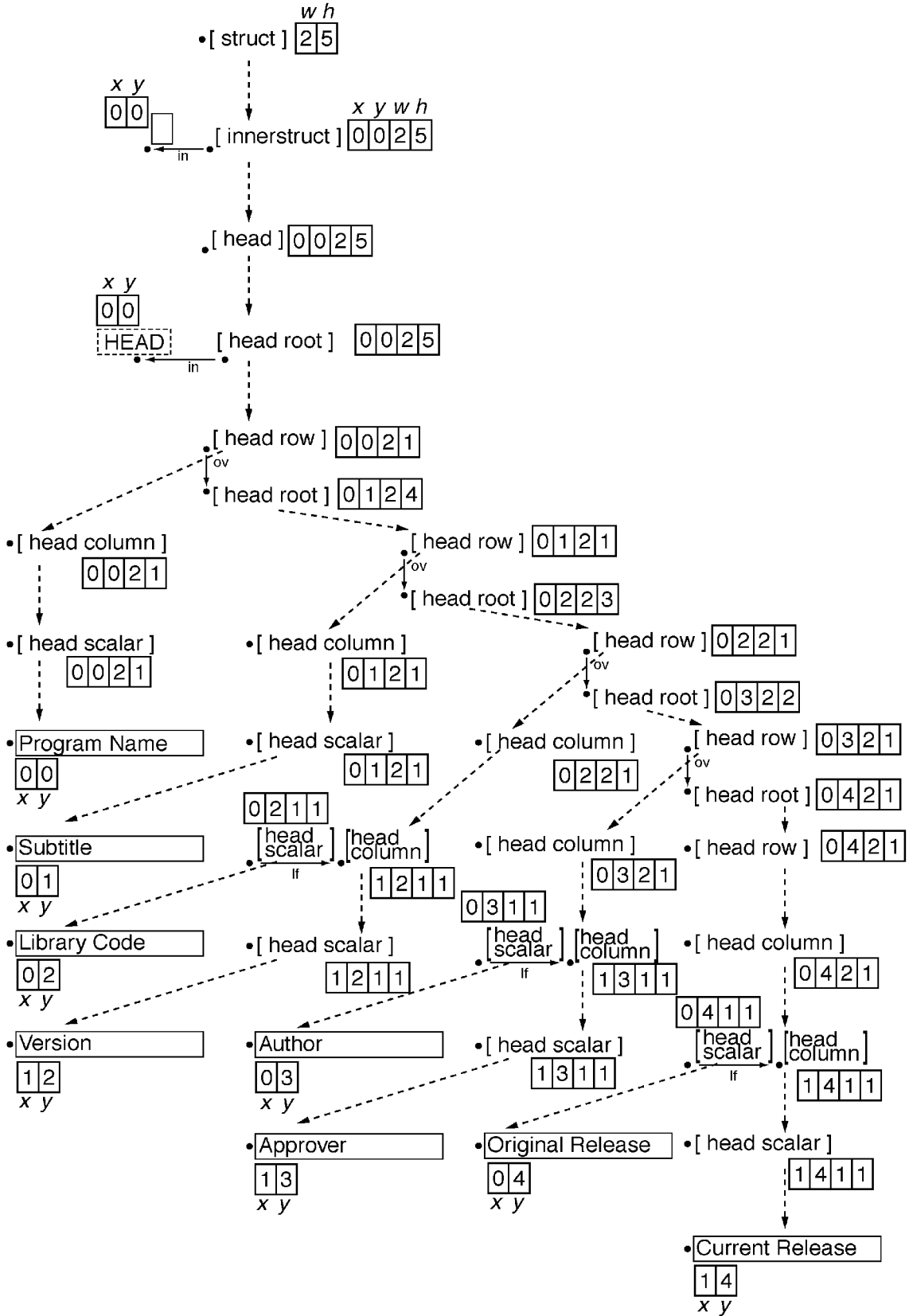
生成規則 番号	生成規則	生成規則
H7	$\begin{array}{c} \text{ov,lf} \\ \downarrow \\ \text{[head scalar]} \\ \downarrow \\ \text{in,ov,lf} \end{array}$	$:= \begin{array}{c} \text{ov,lf} \\ \downarrow \\ \text{プロジェクトコード} \\ \downarrow \\ \text{in,ov,lf} \end{array}$



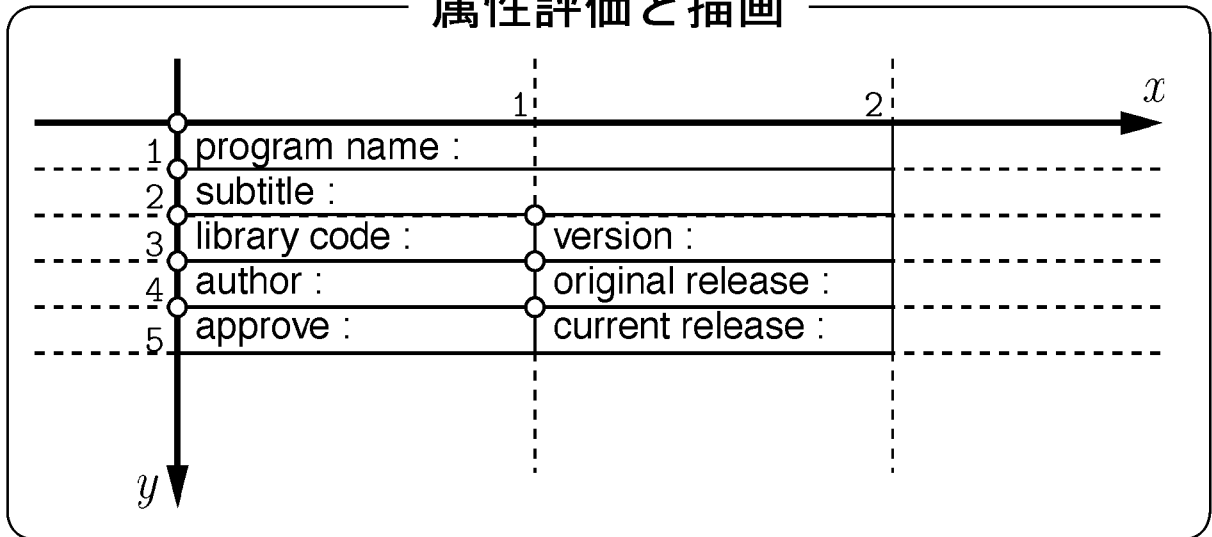
順位付きマーク付きグラフ 2



導出木と属性評価



属性評価と描画



3.4 HFAPGGの性質

3.4.1 HFPGGによって生成される表の例

プロジェクトコード:	A
プログラム名:	プログラム概要書
ライブラリ登録コード:	版名:
文書責任者:	現行版発行日:
著作者:	初版発行日:
CR分類コード:	キーワード:
目的・範囲:	
背景情報:	
記述言語:	所要ソフトウェア:
操作:	所要ハードウェア:
関連文書:	
例:	
機能:	

3.4 HFAPGGの性質（つづき）

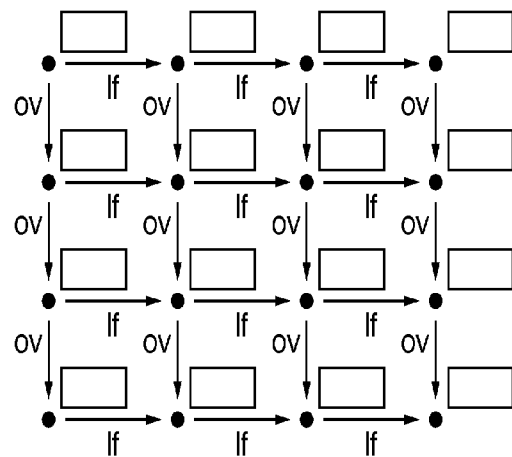
3.4.2 HFAPGGによって生成されない表の例

プロジェクトコード:	A
プログラム名:	プログラム概要書
著作者:	初版発行日:
文書責任者:	現行版発行日:
キーワード:	CR分類コード:
目的・範囲:	
背景情報:	
記述言語:	所要ソフトウェア:
操作:	所要ハードウェア:
関連文書:	
ライブラリ登録コード:	版名:
機能:	
例:	

格子状の表

- ・仕様書には変数表のように、行や列で関係付けられた表は必要になる。
- ・格子状の表の構造は、上下左右の項目が明確にわかる形である必要がある。

例 格子状の表とそのマーク付きグラフ



定義 NCE グラフ文法

The set of all (concrete) graphs over Σ and Δ is denoted $GR_{\Sigma, \Delta}$

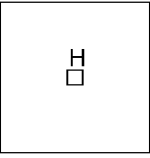
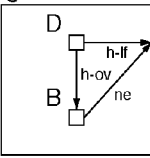
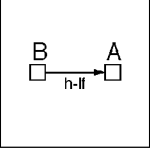
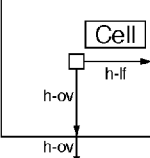
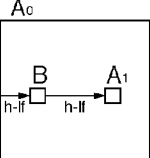
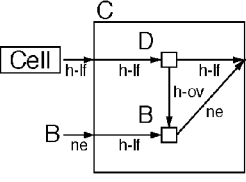
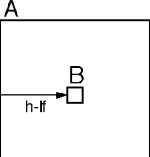
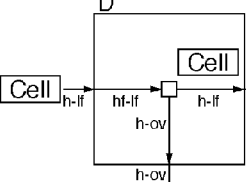
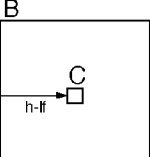
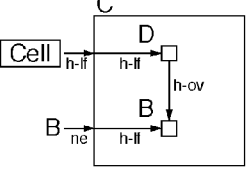
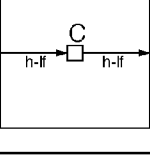
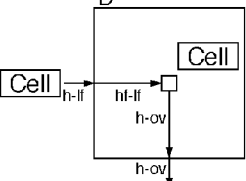
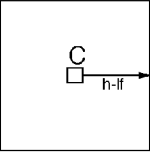
A *graph with (neighbourhood controlled) embedding* over Σ and Δ is a pair (H, C) with $H \in GR_{\Sigma, \Delta}$ and $C \subseteq \Sigma \times \Delta \times \Delta \times V_H \times \{in, out\}$. C is the *connection relation* of (H, C) , and each element $(\sigma, \beta, \gamma, x, d)$ of C (with $\sigma \in \Sigma$, $\beta, \gamma \in \Delta$, $x \in V_H$, and $d \in \{in, out\}$) is a *connection instruction* of (H, C)

The set of all graph with embedding over Σ and Δ is denoted $GRE_{\Sigma, \Delta}$.

Definition

An *edNCE grammar* is tuple $G = (\Sigma, \Delta, \Gamma, \Omega, P, S)$ where Σ is the alphabet of node labels, $\Delta \subseteq \Sigma$ is the alphabet of edge labels, $\Omega \subseteq \Delta$ is the alphabet of final edge labels, P is the finite set of productions, and $S \in \Sigma - \Delta$ is the initial nonterminal. A production is of the form $X \rightarrow (D, C)$ with $X \in \Sigma - \Delta$ and $(D, C) \in GRE_{\Sigma, \Delta}$. \square

格子状の表の生成規則と意味規則

Productions and Semantic Rules for Matrix Forms (Horizontal Derivation 1)	
<p>S</p>  <p> $x(A) = x(S)$ $y(A) = y(S)$ $width(S) = width(H)$ $height(S) = height(H)$ </p>	<p>C</p>  <p> $x(D) = x(C)$ $y(D) = y(C)$ $x(B) = x(C)$ $y(B) = y(C) + height(D)$ $width(C) = \max(width(D), width(B))$ $height(C) = height(D) + height(B)$ </p>
<p>H</p>  <p> $x(B) = x(A)$ $y(B) = y(A)$ $x(A) = x(H) + width(B)$ $y(A) = y(H)$ $width(H) = width(B) + width(A)$ $height(H) = \max(height(B), height(A))$ </p>	<p>D</p>  <p> $x(D) = x(Cell)$ $y(D) = y(Cell)$ $width(Cell) = width(D)$ $height(Cell) = height(D)$ </p>
<p>A_c</p>  <p> $x(B) = x(A_c)$ $y(B) = y(A_c)$ $x(A_c) = x(A_c) + width(B)$ $y(A_c) = y(A_c)$ $width(A_c) = width(B) + width(A_c)$ $height(A_c) = \max(height(B), height(A_c))$ </p>	<p>C</p>  <p> $x(D) = x(C)$ $y(D) = y(C)$ $x(B) = x(C)$ $y(B) = y(C) + height(D)$ $width(C) = \max(width(D), width(B))$ $height(C) = height(D) + height(B)$ </p>
<p>A</p>  <p> $x(B) = x(A)$ $y(B) = y(A)$ $width(A) = width(B)$ $height(A) = height(B)$ </p>	<p>D</p>  <p> $x(D) = x(Cell)$ $y(D) = y(Cell)$ $width(Cell) = width(D)$ $height(Cell) = height(D)$ </p>
<p>B</p>  <p> $x(C) = x(B)$ $y(C) = y(B)$ $width(B) = width(C)$ $height(B) = height(C)$ </p>	<p>C</p>  <p> $x(D) = x(C)$ $y(D) = y(C)$ $x(B) = x(C)$ $y(B) = y(C) + height(D)$ $width(C) = \max(width(D), width(B))$ $height(C) = height(D) + height(B)$ </p>
<p>B</p>  <p> $x(C) = x(B)$ $y(C) = y(B)$ $width(B) = width(C)$ $height(B) = height(C)$ </p>	<p>D</p>  <p> $x(D) = x(Cell)$ $y(D) = y(Cell)$ $width(Cell) = width(D)$ $height(Cell) = height(D)$ </p>
<p>B</p>  <p> $x(C) = x(B)$ $y(C) = y(B)$ $width(B) = width(C)$ $height(B) = height(C)$ </p>	

4.[結論]

4.1 [まとめ]

—— まとめ ——

- 表をマーク付きグラフで表す.
- 属性順位グラフ文法のクラスを定義した.
- 仕様書を属性グラフ文法を用いて形式的に定義した.
(1)生成規則280個,(2)描画のための属性
- HFAGGが属性順位グラフ文法であることを示した.
- 格子状の表の生成規則を定義した.

4.2 [今後の課題]

—— 今後の課題 ——

- HFAGGの属性評価の計算量.
- 編集操作の定式化.
- 仕様書支援システム（編集, 描画）への応用