

第 6 章

関連研究

6.1 空間パーサ生成系の関連研究

空間パーサ生成系の関連研究としては、SPARGEN[2]、Penguins[3] [4]、恵比寿 [5] [6] [8]、VIC[9] などがある。

SPARGEN は、OOPLG (Object-Oriented Picture Layout Grammars) を用いて図形言語の文法を定義することで空間パーサを生成するシステムである。OOPLG では、図形の属性や制約を C++ を用いて定義している。Penguins では、図形の文法を CMG を用いて空間パーサを生成している。しかしながら、これらの空間パーサ生成系は、テキストを用いて図形言語の文法を定義するので、ユーザは文法を理解している必要があり、一般のユーザにとって使いやすいものとはいえないという問題があった。

我々が研究を行っている恵比寿では、ユーザが入力した図形を用いて大まかな図形の CMG の文法を自動的に生成するようにしている。そのあとユーザは、生成された文法を見ながら、制約の追加または削除などをおこない修正する。

恵比寿の図形エディタは、図形言語の文法を定義する定義ウィンドウと実際に図形の解釈を行う実行ウィンドウに分れていたが、「Rainbow」では一つのウィンドウに統合した。なぜなら、ユーザは文法を定義しながら図形を描いて文法の正しさを検査することの繰り返しによって文法を定義していくので、文法の定義モードと図形の実行モードを一緒にする方が便利である。そうすれば、一つの非終端シンボルとしたい図形の中に他の生成規則によって定義された非終端シンボルを認識することができる。また、複雑な図形を表示するのに画面の空間を有効に使える。

さらに、「Rainbow」では恵比寿が扱う制約の他にレイアウト制約を実現している。特に、軟かいレイアウト制約の導入により、図形の全体を解釈しながらインタラクティブにバランスよくレイアウトすることが可能になっている。

VICでは、視覚的な制約入力インターフェイスを恵比寿上にも実装し、テキスト編集を行わずにCMGの文法を生成している。

6.2 レイアウトシステムの関連研究

最近、Marriotらは空間パーサ系 Penguins[12]にレイアウト制約を追加することで図形のレイアウトを行うことを提案している。Penguinsが提供するレイアウト制約は、「Rainbow」の硬いレイアウト制約に相当するものであるが、Marriotらは、硬いレイアウトの具体的な応用例として二分木、数学の方程式、状態遷移図などを挙げている。

しかしながら、レイアウトでは、図形の全体を把握しやすくバランスよくレイアウトすることが大事であるが、Penguinsでは、図形の全体をバランスよく分りやすくレイアウトするための軟らかいレイアウト制約を提供していない。

その他のレイアウトシステムとしてTRIP[27]がある。TRIPとは、テキスト形式の抽象オブジェクトとその関係からレイアウトされた図形を生成するシステムである。抽象オブジェクトとその関係はユーザが定義するマッピング規則により、図形オブジェクトとその関係に変換され、レイアウトシステムより図形が生成される。マッピング規則は、ユーザがPrologで記述している。TRIPでは、制約に基づいて図形要素の位置が決められてから図形が生成されるが、そのあと図形要素間に制約が課せられないので、制約を保つつつ図形を動的に編集することができない。TRIPでもグラフィックレイアウトシステムが存在し、無向グラフの描画アルゴリズムを使って図形のレイアウトを行うが、図形要素の位置を決める図形関係と混じってレイアウトすることはできない。