

手続法記述言語KRIP/Lによる特許法の記述

新田 克己

長尾順太郎

水鳥 哲也

電子技術総合研究所 (株)日本ビジネスコンサルタント (株)明電舎

1. はじめに

法律のエキスパートシステムに期待される機能は静的なものとの動的なものに分けることができる。前者は「～のときはどうすればよいか」「～と～はどのような関係にあるか」「～に関する条文はどれか」等のように、条文にかかっている知識を整理し、要求に応じてそれを検索するものである。後者は実際の事例について法律を適用し、推論を行なうものである。この場合、法律を適用した最終結果だけでなく、どの条文を使ってどのように推論されたのかという根拠を示すことが重要である。

法律の知識ベースを構築するには、知識が法令文という整理された形で与えられているので、容易にできると考えられがちである。しかしながら、論理的に構成されているように見える法令文を論理式としてとらえ、現実の問題に適用すると、決して正しい推論を行なわない。それは、(i) 条文に使われている概念には意味の幅があり、単純な述語として表現できないこと、(ii) 条文はそれが使われるときの暗黙の文脈(環境)が仮定されており、それを補わなければ実際には使えないこと、等による。法令文を解釈するためには、法律家が常識として持っている一定の知識が必要であり、その知識を組み込んだ知識ベースを構成しなければならない。

法律の分野ではこのような問題点があるが、法律には色々な種類があり、それぞれ性質が異なるので、エキスパートシステムが作りやすいもの、作りにくいものが存在する。我々は特許法エキスパートシステムを構築するための言語KRIP/Lを開発した。特許法を対象としたのは、この法律には後述するように色々な法律的要素が含まれているからであり、また、手続法の性質は計算機による処理が比較的行ないやすいからである。ここではKRIP/Lで法律上で表われる現象をいかに記述するかについて説明する。

2. 特許法の特徴

特許法は、特許を取得するための国家に対する手続きとその権利内容に関して規定した法律である。その特徴は(i) 民法に対して特別法という関係にあり、特許法に規定していないことは民法の規定がデフォルトとして適用されること、(ii) 手続法と実体法という性質の異なる条文が混在していることである。

ここで、実体法とは社会生活における規範を定めた法律であり、手続法は実際に事件がおきたときに、その事件と実体法を照らし合わせて、適切な判断を行なうための法律である。例えば、刑法(実体法)は犯罪についての刑を規定しているがこれだけでは役に立たない。事件がおこると刑事訴訟法(手続法)の手続きに従って裁判が行なわれ、証拠の提出等によって事実を確認した後で刑法の規定により罪状が決まるのである。特許法では特許を与えるための要件を定めた部分と特許権の内容を定めた部分が実体法であり、特許庁の手続きや特許庁に対する手続き(特許出願、出願公開等)を定めた部分が手続法である。手続法は事実認定が比較的容易であるが、時間に関する記述が多いという特徴がある。

特許法は200以上の条文からなる。その中には手続きの数が50以上あり、それぞれにつ

いて手続きを行なえるための（主体的、客体的、手続き的）条件やその効果が規定されている。主要な手続きはフローチャートとして表現することができる。その一方、例外的な手続きは割り込み的に行なわれるので、それを整理された形で記述することは困難である。

3. 手続法記述言語KRIP/L

法律の知識表現には、すでに、述語論理による方法、フレームによる方法、意味ネットワークによる方法、ルールによる方法等が提案されている。しかし、手続法を記述するにはこれらの方法では十分ではないため、手続法記述言語KRIP/Lを開発した。その特徴は以下のとおりである。

i) 区間論理式とオブジェクト指向の2つを統合した知識表現言語である。オブジェクトは時間とともに変化する内部状態の履歴を保存することができるので、過去へさかのぼって内部状態を知ることができる。また、区間論理式の採用により、時間に関する条件とその他の条件の記述を分離し、わかりやすい記述ができる。

ii) Prolog の記述も可能である。

3.1 手続きのモデル

(1) 手続きの扱い

特許庁に対する手続きが適法であるためには(i)手続き人の要件、(ii)様式の要件、(iii)時期的要件、(iv)手続きに固有のその他の要件、等を満たさなければならない。特許庁に対して行なわれた手続きについては方式審査（手続き時から1月近く遅れることが多い）がされ、要件を満たさない点があると補正命令が出される。指定期間内に補正手続きを行なうと初めから完全な手続きがなされたとみなされるが、補正をしないと無効処分がなされ、手続きは初めからなかったものとみなされる。

このように特許法では手続きのミスに対して色々のところでそれを訂正したり、意見書や答弁書を提出したりすることができるようになってきている。また、特許庁の判断に対して審判請求や訴えの提起等の不服申し立てができるようになってきている。このような不服申し立て等で手続きの効力が（「みなす」の効果として）過去へ繰り上がったたり（これを遡及効果という）、繰り下がったりする。すなわち、実際におきた事実と法律的な効果のくいちがいが手続法の最大の問題である。

(2) 手続きのモデル

前述のように実世界では手続きがなされると①特許庁長官によってその正当性が審査（方式審査）され、正当であれば②効力が手続き時にさかのぼって発生する。手続きの効力によって新しく権利が発生することもあるし、他の対象物体の内部状態を変更することもある（図1(a)）。

これをオブジェクトのモデルで考える。オブジェクトは人や者や権利や手続きに対応する。他のオブジェクトに副作用を及ぼすオブジェクト（ここでは手続きに対応するオブジェクト）を特にイベントと呼ぶ。イベントはそれが起こった時刻で時間のチェーンにつながる。各イベントにはその手続きがなされた年月日や手続き人、手続き内容等の内部データの他に、その正当性を裏付けるための手段（どの条文をどの順序で調べるか）やどのような効力を他に及ぼすかがメソッドとして記述されている。

適用したい事件はイベントやオブジェクトの集合として表わし、事実データベースに格納されているものとする。システムはタイマーを持っているものとし、古いイベントから順にメッセージを送る。すると、イベント内のメソッドが起動されて①自分自身の正当性

を判断し（方式審査）、その結果をシステムに送る。システムは、そのイベントが正当ならば、新たにメッセージを送り、そのイベントの効果として②新たにオブジェクトを生成したり他のオブジェクトに副作用をおよぼしたりするのである（図1(b)）。

このように、オブジェクトの内部状態はイベントによって変更され、その効果は次第に蓄積される。内部状態の変更時刻もパラメータとして指定できるので、効果の発生時を任意に設定できる（遡及効果については4章）。

手続きの正当性を証明するための条文は論理式の形で与えられる。条文の中で効力を規定した部分はイベント中にかかれるので、論理式で表現するのは主に手続きの前提条件の部分である。この論理式はオブジェクトの内部状態を参照することができる。その際、状態の変化に注意する必要がある。特許法においては出願から特許登録まで数年かかることから、その間に状況が変化する。ある時点で成立していたことがある時点では成立しないことになる。知識ベースでは事実の判断をするときに判断基準（時間、主体、場所等）を定義しておくことが必要となる。条文にこれらの基準が規定していないことがあり、しばしば、学説の対立がある。

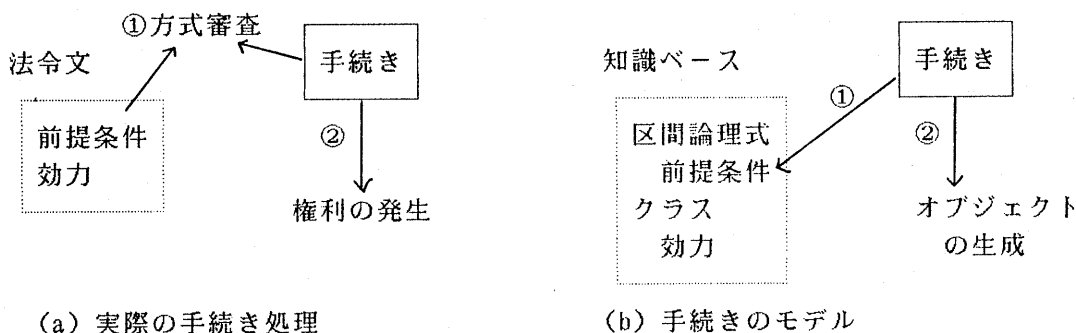


図1 手続きのモデル

3.2 クラスとオブジェクト

クラスとは一群のオブジェクトに共通の性質を記述したものであり、条文中に表われる名詞（手続き、権利、人間等）を定義するのに用いられるものである。そのシンタクスは以下のようなになる（条文の識別子は省略可能）。

```
class <クラス名>
  super : <クラス名>                :: <根拠条文の識別子>
  スロット : <値> OR {<タイプ>}    :: <根拠条文の識別子>
  . . .
  メッセージ --> <Prologゴール列>  :: <根拠条文の識別子>
  . . .
end_class.
```

super は上位クラスを表わし、そのスロットやメソッドが継承される。スロットは内部状態を表わす変数であり、右辺には実際の値かその値が満たすべきタイプが記述される。メッセージ（セレクタ+引数）は受けつけるメッセージを表わし、それが送られると右辺

のゴール列が実行される。条文の識別子は

```
section ( 特施規 3 , 1 , 1 )
```

のように<条文番号, 項番号, 論理式の識別番号>の組で与える。

例えば、出願審査請求という手続きは

```
class 出願審査請求
  super : 特許庁長官への手続き
  手続き人 : {手続き能力者}      :: section(..).
  手続き時 : {年月日}
  審査対象 : {出願実体}
  constraint(X) --> type_match(..), ...
  legal(X) --> legal(self,section(特48,1,_)), ...
  effect(X) --> event_val(..), o_modify(..), ...      :: section(..).
end_class.
```

のように記述される。これは、出願審査請求という手続きは特許庁長官への手続きの1つであり、手続き人は手続き能力者というタイプのオブジェクトでなければならないこと等を表わしている。constraintはこの手続きが正当であるために各スロットの値が満たすべき条件を記述し、legalはこの手続きが行なえるための前提条件はどの条文をどの順序で調べるべきかを記述し(条文を陽に指定するので、関連のない論理式は参照されない)、effectはこの手続きの効力を記述している。オブジェクトを発生させたり、副作用によりオブジェクトのスロットの値を変更させたり、ほかのオブジェクトにメッセージを送ったりするための組み込み述語が用意されている。なお、変更前のスロットの値も保存されているので、時刻を指定するとその時点での値を知ることも可能である。

3.3 区間論理式

条文は条文の識別子と条文の内容を表わす論理式の対で表現される。論理式はPrologの節、または、以下の区間論理式で表現される(BODYはPrologのゴール列)。

- (1) [A : B] => P :- <BODY>.
- (2) [A : B] => ~P :- <BODY>.
- (3) T @ P :- <BODY>.
- (4) [A : B] --> may Q :- <BODY>.
- (5) [A : B] --> must Q :- <BODY>.
- (6) [A : B] --> mustnot Q :- <BODY>.

その意味は(1)(2)は「イベントAが発生してからイベントBが発生するまでの間にBODYが成立するならばPが成り立つ(成り立たない)」であり、(3)は「BODYが成り立つときは常にPが成り立つ」を意味する。(4)(5)(6)は「イベントAが発生してからイベントBが発生するまでの間にBODYが成立するならば手続きをしてもよい(しなくてはならない、してはならない)」である。(1)(2)(3)はPrologの節を拡張して、有効期間を付加したものであり、主に実体法の記述に用いられる。(4)(5)(6)は法律特有の様相オペレータを導入したものであり、手続法の記述に用いられる。

イベントやオブジェクトが存在しているか否かを調べるために組み込み述語event, objectが用意されている。eventは

```
event(ID, Class, Condition, Time)
```

の形で用いられ、「ある時刻 (Time) に条件 (Condition) を満たし、クラスが Class に属するイベントを事実データベースから探し、そのイベントの識別番号を ID にセットする」働きをする (ID があらかじめわかっているときには、このイベントが条件を満たすか否かを調べる)。条件は

[国籍: 日本, 住所: ~日本, 年齢: (ID2 #年齢)]

のように、対象としているイベントのスロットの値が特定の値と等しい (等しくない) ことを指定したり、対象としているイベントのスロットの値が特定のオブジェクト (ここでは ID2 で表わしている) の特定のスロットの値と等しいことを指定したり、Prolog のゴール列を記載したりすることができる。

引数として時刻を与えるのは、スロットの値が時刻とともに変化するためである。この引数が未定義変数ならば最新の値で判定される。

記述例を以下にあげる。これは特許法 33 条 1 項の一部であり、「発明をして特許を受ける権利を持つものは特許出願の前にそれを移転することができる」ことを内容とするものである。

section(特33,1,1).

```
[event(ID,発明行為,[ ],_) : event(ID2, 出願,[請求の範囲:(ID#内容)],_)]  
--> may 出願前の移転([主体:(ID3#主体),対象:ID3])  
:- object(ID3,特許を受ける権利,[ 客体:(ID# 内容) ],self#手続き時) .
```

4. 特許法の問題点と KRIP/L による記述

(1) 有効期間

法令文にかかれている規定が常に適用できるとは限らない。例えば、特許法 29 条に「特定の要件を満たす発明をした者は特許を受ける権利を持つ」旨の規定があるが、同 33 条に「特許を受ける権利は移転することができる」と規定している。29 条の内容は、(i) 発明時には特許を受ける権利が発生すること、(ii) その権利の主体は発明者であること、を表わす。(ii) は発明時には成立するが、移転と同時に成立しなくなるものと解釈しなければならない。

KRIP/L ではこの関係を、2 つの方法で記述することができる。1 つは、区間論理式を用いて、(ii) の内容は発明時から移転時まで成立するという記述である。

```
[event( _, 発明, ... ) : event( _, 移転, ... ) ] => 特許を受ける権利 (発明者)
```

もう 1 つは、特許を受ける権利をオブジェクトで表わし、発明時にはオブジェクトの生成を行ない、移転時にはオブジェクトの主体のスロットを変更する方法である。

(2) 実体法の起動

実体法は手続法によって起動されることがある。例えば、特許法 29 条に「特定の要件を満たす発明をした者は特許を受ける権利を持つ」旨の規定があるが、32 条には「特定の発明に関しては 29 条の規定にかかわらず特許を受けることはできない」と規定してある。さらに 49 条には「特許の審査においてその発明が 29 条に該当しないとき、32 条に該当するとき、..には拒絶査定をしなければならない」と規定されている。29 条、32 条は実体的規定、49 条は手続的規定である。29 条や 32 条は 49 条から陽に参照されて初めて使用することができるのであり、仮に 32 条が 49 条から参照されていないときには、これを理由に拒絶査定をしてはならないことになる。また、このことから 29 条と 32 条は独立に適用することができ、29 条と 32 条を論理式として結合する必要はないことになる。

KRIP/Lではこの関係を

[...] --> must 拒絶査定

:- apply(特29, ..), not(apply(特32, ..)), ...

のように条文を陽に指定することによって表現する。

(3) 例外規定

法令文には多くの例外規定や例外の例外規定がある。例えば、特許法29条には「特許を受けるためには、刊行物に掲載される前に出願する（新規性を有する）ことを要する」旨の規定があり、30条には「特定の手続きをしたときは新規性を喪失しないとみなす」規定がある。

この場合、以下のように様相記号Mで表現したり、

特許を受けることができる :- M 出願前に刊行物に掲載

not 出願前に刊行物に掲載 :- 特定の手続き

以下のように本規定と例外規定を組み合わせた論理式で表現したりすることができる。

特許を受けることができる :- 出願前に刊行物に掲載 \wedge not 特定の手続き

しかし、前者の方法は、各規定の例外規定の条文はあらかじめわかっているので効率的ではない。後者はすべての例外規定を組み合わせると、論理式が複雑になりすぎる。KRIP/Lでは例外規定を2つの方法で記述できる。1つはメソッドで条文を参照するとき、例外規定(30条)を先に調べ、本規定(29条)を後から調べる方法である。もう1つは後述の遡及効果の機能で刊行物記載の事実の効力を無視する方法である。

(4) 違法な手続き

違法な手続きであってもその効果が当然に無効であるとは限らない。例えば、前提条件が満たされていない手続きがあり、方式審査でそれが発見されると、特許庁長官から補正命令が出され、それに従ったときには手続き時にさかのぼって効力が発生する。従わないときには無効処分がなされ、手続きは初めからないとみなされる。ところが過誤により手続きの効力が発生してしまうと、後で違法性が発見されても、その効力を取り消す手続きが新たに必要である。取り消しの手続きによって、効力が初めからなかったものとみなされるものがあるし、取り消し手続きのときから消滅するものもある、また、時効により取り消すことができないものもある。さらに、取り消しによって、ある効力だけが取り消され、他の効力は存続することもある。

論理式では、前提が満たされないのに効力が発生してしまう問題を記述するのは容易ではない。KRIP/Lでは時間の経過にそったシミュレーションを行なうので、この状況を直感的に表現することができる。この問題は後述の遡及効果の項にも関連する。

(5) 変動する概念（疑似オブジェクト）

実務においてはオブジェクトの概念があいまいなものがある。例えば、特許庁に手続きを行なう者は手続き能力者でなければならない。通常の成人は手続き能力者であるが、破産宣告を受けると手続き能力者ではなくなることになる。従って、仮に手続き能力者のクラスを自然人の下位クラスに定義すると、時間とともにクラスが変動することになる。

KRIP/Lでは、自然人のオブジェクトを定義しておき、この内部状態がある条件（日本国籍を持ち、成人であって、破産宣告等を受けていないことが特許法7条に規定されている）を満たすときに限り、手続き能力者のオブジェクトであると擬制するのである。すなわち、手続き能力者を

section(特7, 1, 1) .

T@手続き能力者(ID) :-

object(ID, 自然人, [国籍: 日本, . . .], _, T) .

と記述し、あるオブジェクトのスロットの値がこの右辺の条件を満たすときには、このオブジェクトは「手続き能力者」のクラスに属するものと擬制する。これについては次の下位概念の項でもふれる。

(6) 下位概念

条文に表われる用語には階層構造がある。例えば「手続きをした者はその補正ができる」という条文に対し、出願や出願審査請求は手続きの一種であることを知らなければ、出願の補正ができるかどうかの判断ができないことになる。このため、システム述語の機能を拡張してある。例えば、あるオブジェクト(ID)のクラス名がClass であり、時刻T に条件(Cond)を満たすか否かを判定するシステム述語は

object(ID, Class, Cond, T)

であるが、Class の対応においては(i) 直接Class に合致するもの、(ii) Class の下位概念のクラス名に合致するもの、(iii) 論理式で定義されるもの、の順に対応させる。(ii) の機能により、下位概念の探索を行なうことができ、(iii) の機能により、疑似オブジェクトが実現できる。

(7) 遡及効果

手続きを取り下げたり、無効処分があったりすると、その手続きは初めからなかったものとみなされる。また、特許要件を満たさない特許出願に対して、誤って特許権が発生してしまったときには、第三者は無効審判を請求し、その権利をつぶすことができる。無効審判が確定すると、特許権は初めからなかったものとみなされる。その審判に不服ある特許権者が再審を請求するとその審判がまたくつがえることがある。

このように、事実としては存在しないのに、効果としては存在するような「みなす」規定が単調論理では扱いにくい点である。

KRIP/Lでは「取り下げ」や「無効処分」等があるとその時点の状況を凍結して、時間を元に戻し、シミュレーションの再実行を行なう機能がある。例えば図2(a)でA～Dは手続きである(DはBに対する無効処分とする)。シミュレーションが時刻T1からT3まで進行し、審査状況がX～Zと変化しているものとする。T4でDが発生すると、その時点の状況を凍結して、状況をT2の直前Xに戻し、Bがなかったものとしてシミュレーションが再開される(図2(b))。T2からT4まではシミュレーションをやり直すことになり、例えば、TAの時点の状況をT4以前に判断したとき(Y)と、T4以後に判断したとき(Y')は異なることになるが、凍結された状況は保存されているのでどちらの状態も知ることができる(斜線は凍結される部分を表わす)。

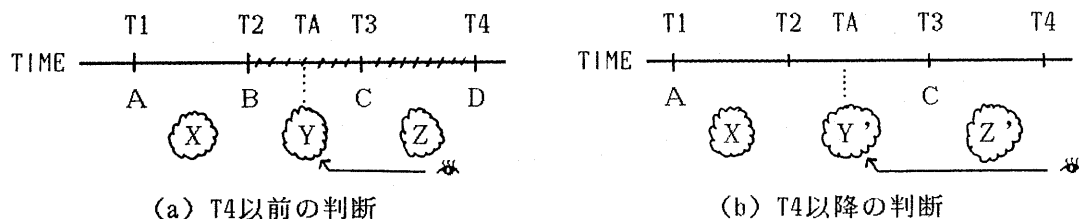


図2 凍結機能

(8) 期限問題

特許法の多くの手続きについては、それを行ないうる期限が決まっている。また特許権の存続期間も法定されている。この期限は、一般には

```
[ event(ID, 出願,..): limit(ID# 出願日, 7年, _) ] -->
```

```
may 出願審査請求 :- . . .
```

のように区間論理式で記述される(limitは期間についての加算を行なうシステム述語)。しかし、(i) 手続きの期限は最終日が休日ならば期限が1日延長されること(権利の期限は延長されない)、(ii) 手続きによっては請求や職権で期限が延長されることがあること、(iii) 手続きによっては期限内に手続きをしないと特別な効力が発生するものがあること、等の問題がある。

KRIP/Lでは、(i) についてはシステム述語の拡張で対処している。(iii) の例としては48条の3「出願してから7年内に出願審査請求をしないときは取り下げとみなす」旨の規定がある。これは出願の効果として、その7年先に仮のイベントを用意し、出願審査請求がなかったときにはその仮イベントの効果として、出願の取り下げを行なうことによって解決する。(ii) については何の手当もしていないが(iii) と同じ対処は可能である。

5. おわりに

特許法の記述について述べたが、実際には各条文ごとに細かい問題点があり、ここで述べたのは一例にすぎない。また、民法の規定や解釈上でのメタ知識を補充する必要もある。エキスパートシステムの成否は法律の分析による部分が大きく、条文の表面的な解釈だけでは良い知識ベースはできない。KRIP/Lの機能により、論理式レベルでは記述がむずかしい問題のかなりの部分は記述できるものと考えている。しかし、その一方、KRIP/Lでは細かい点まで記述しなければならぬ。法律概念を整理し、より高レベルの言語仕様を採用することも考えられる。

現在、パーソナル・コンピュータ上で基本条文だけを備えた特許法エキスパートシステムを試作しているところである。メモリの不足で条文の一部しか使用できないため、近いうちにPrologマシンPSIへ移植する予定である。

謝辞 研究の機会を与えられました棟上昭男ソフトウェア部長、佐藤孝紀情報システム研究室長に感謝いたします。討論をしていただいた電総研SIG.ANYグループの方々、情報システム研究室の方々に感謝いたします。

参考文献

- [1] 新田、他：「工業所有権法エキスパートシステムの事例問題解決機能」
ロジックプログラミングコンファレンス'85(1985).
- [2] 新田、他：「KRIPユーザーズ・マニュアル」電総研内部資料 ETL-RM-85-36J.
- [3] 吉野：「法適用過程における推論へのコンピュータの応用」
法とコンピュータNo.3(1985).
- [4] 池田：「人工知能言語による法律の解釈と適用」日経コンピュータ(1984).
- [5] 吉藤：「特許法概論」有斐閣.
- [6] 「工業所有権法逐条解説」発明協会.