Vol.2015-CH-107 No.2
2015/8/9

漢字の包摂粒度の符号化に関する諸問題について

守岡 知彦[1]

概要：漢字を含む電子テキストをアーカイブする場合、符号化文字の指示対象を明確にすることが重要である。漢字において符号位置の包摂範囲を確定するのは包摂規準の役割であるが、さまざまな漢字文献を対象とした場合、汎用符号化文字集合が想定する包摂規準だけでは不十分な場合があり、複数の包摂規準を混在させた文字処理が必要な場合がある。ここでは、こうした場合の課題や適切な情報の記述や処理の可能性について議論したい。

1. はじめに

文字符号によって符号化された文字（符号化文字）はさまざまなテキストを電子化するための基盤となるものであり、符号化文字の指示対象である抽象文字は電子テキストを解釈する上での基礎となるものといえる。もしここで、抽象文字が不明確であったり一般的な文字用例とかけ離れたものであれば適切なテキストの符号化が行えなかったり符号化されたテキストの利用（解釈）に支障をきたすかもしれない。抽象文字は文字の概念を表現したものといえ、何をもって文字の特徴とするかは用字形 (script) 毎に異なる。ラテン文字のような表音文字の場合、抽象的な音価に対応するものとして文字概念が構成されるし、抽象的な意味に対応するものとして文字概念が構成される表意文字というものを想定することもできるだろう。

漢字の場合、伝統的に、形・音（抽象的な音価のカテゴリー）・義（抽象的な意味カテゴリー）の３要素の組合せによって文字概念が構成されると考えられてきたが、これは形によって音義の対応関係からなる形態素（語）を表現しているという意味で表語文字の一種といえ、その文字概念はもともと漢語（古代・古典中国語）の形態素に対応するものであったといえる。ただ、漢字が使われて来た年月の間に中国語自体も大きく変化し多音節語が増大したため１文字で書けない形態素が増加したし、また、中国語以外の言語を漢字で表現することも行われるようになり、漢字と形態素の関係性は常に変化し続けてきたといえ、漢字の文字概念はその規範意識とともに時代や地域によって変化するものといえ、固定的な文字概念を想定しづらい面がある。とはいえ、前近代の東アジアにおいては古典中国語（漢文）のリテラシーによってある程度共通の規範意識や文字概念が共有されてきたということもいえる。いずれにしてもこうした文字概念の可変化性は漢字の符号化に関する原理的な困難さ、すなわち、文字概念を固定してしまうとそこからはみ出るものが生じ得、それが符号化できなくなってしまうという問題をもたらしてしまうのである。

漢字が基本的に表語文字であると考えるならば、その文字概念は文字が指示する形態素（あるいはその集合）に立脚すべきであるといえ、音義や品詞情報、あるいは、テキスト中での文脈情報に基づいて符号化すべきものだといえるが、既存の文字符号化の枠組において品詞情報やテキスト中での文脈情報を文字の要素の一部として符号化することは基本的にできない。文字符号のための漢字の文字概念は文脈自由な客観的・機械的に判別可能な要素に立脚する必要がある。そうした観点で見た場合、音や義は字形によって直接表現されたものではなくその解釈によって得られる情報といえ、また、単一の文字だけでは判別できずテキスト中での文脈情報がないと解釈できない場合も少なくないし、それでも解釈できない場合もありうるので抽象文字の主たる弁別要素としては良くないといえる。このため、漢字の符号化文字は、基本的に、形に立脚せざるを得ないといえる。

漢字の形に基づいて漢字の抽象文字を構成する場合、字形（書かれたり印刷されたり表示された文字の具体的な形）そのものには無数の差異があり、その差異の全てを区別することには意味が無いため、なんらかの抽象形状を想定する必要がある。抽象形状を考える場合、字形デザイン上の要素を捨象し、文字弁別に関わる要素だけを残すような仕組みをうまく考えられれば良い訳であるが、前述のように、漢字の文字概念（そしてそれらの弁別）がそもそも形だけに依っている訳ではないことからことはそう簡単で

[1] 京都大学人文科学研究所
Institute for Research in Humanities, Kyoto University

はない。例えば、「類」と「類」では「大/犬」の点の有無は捨象しても文字弁別に影響しないが、「大」と「犬」は別字なので常にこの両者を無視することはできない。よって、客観的・文脈自由的に決定可能な字形の差異を捨象すればする程、異なる音義・形態素の漢字が同じ抽象形状の漢字として衝突する度合が増えるといえ、また、逆に、微妙な字形差を区別すればする程、同じ音義・形態素を指示するはずの漢字が別字として区別されてしまい異体字の数をいたずらに増やしてしまう結果になるといえる。繰り返しになるが、「大/犬」のような形状の差異は、その形状そのものに意味があるというよりは、差異の存在がマークアップされていることに意味があるといえ、実際の文字概念の関係は音義や文脈情報も含めた漢字の解釈（あるいは、文字を解釈する人達が共有する規範意識）に依存しているといえる。いずれにせよ、既存の文字符号化の枠組を前提とする限り、どこかで恣意的な線を引くしかなく、どこで線を引いても例外が生じ得るといえる。

このために、JIS X 0213 の漢字の字体の包摂規準や UCS の統合漢字では同一視されてしまうような差異を区別したいケースが存在し得、これらよりも細かい包摂粒度の漢字符号化の仕組みがこれまでも幾つか提案・実装されてきたが、近年では、UCS の統合漢字で包摂された字体・字形差を区別するための枠組として IVS (Ideographic Variation Sequence) という仕組みが注目されている。これは漢字のグリフセットの個々のグリフに対して、統合漢字のコードポイントと Variation Selectors (VS) と呼ばれる一種の枝番の対からなるシーケンスを与え、その対応関係を IVD (Ideographic Variation Database) [1] に登録できるようにしたものである。ここで、IVS に登録されたグリフセットは「コレクション」と呼ばれる。現在、IVD には Adobe-Japan1 という日本語用漢字グリフセットのデファクトスタンダードと汎用電子および文字情報基盤という日本の行政用文字コードを収録しており、漢字のグリフ情報を交換するための標準となりつつある。

IVS は UCS 統合漢字で包摂された複数の字体を区別するための枠組を与えているが、そこで指示されるものの包摂範囲を明確に規定していない。むしろ、異なる包摂規準のグリフ集合を共存させるための仕組みが IVS といえ、統合・分離の規準は IVD (Ideographic Variation Database) に登録されるコレクションの側によって規定することを想定した仕組みといえる。しかしながら、Adobe-Japan1 には文字の同定・分離に関する明示された規定が存在しない。汎用電子の場合、一応の判断規準を設けているが、ソースコードセパレーション規定があり、分離されている例が文字の同定・分離に関する原則に則った結果なのかソースコードセパレーションによる例外なのかが明示されていないため、結局、総体としてどのような包摂規準を用いているのかが不明確である。このため、ソースに関する情報やフォントなどの実装、実際の使われ方などから、帰納的に類推するしかないが、その判断にはどうしても揺れが生じざるを得ない。

Adobe-Japan1 も汎用電子・文字情報基盤も少なくとも抽象文字よりは細かい包摂粒度を表現している訳であり、また、情報交換のための符号である以上、字形よりは抽象化された包摂粒度になっているはずであり、そうしたことから字形デザイン差を捨象した抽象字体粒度を表現したものであってしかるべきなのであるが、例えば、Adobe-Japan1 では「冴」と「冴」のような細かい差異も区別しており、汎用電子でも「伶」と「伶」のようなデザイン差と思われるものが区別されている。また、「八」と「八」や「交」と「交」のような両者で共通して区別されている細かい差異もある。このような字形デザイン差と思えるような微妙な差異を区別している箇所があることも包摂規準の帰納的な類推を行う上での困難さを生じさせているといえる。

人文情報学的データベースという観点で見た場合、テキストの解釈に影響し得るような差異は無視すべきでないといえるが、元々のテキストになかったであろう意味のない差異を符号化の結果生じさせるようなことは望ましくないと言える。ただ、前述のように、漢字の文字概念や弁別や同一性に関する判断は単一の字形を見ただけでは判断できないことが多々あり、テキストに対する深い理解を要するようなケースもあるが、そのようなテキスト解釈無しに漢字が符号化できないとしたらそのテキストの研究に電子テキストが使えないというジレンマに陥ってしまうので、現実的な暫定解も必要となる。多くの場合、この暫定解は、結局の所、元の字形の情報をなんらかの形で保持するということにならざるを得ない。とはいえ、どういう意図でどういう差異を記述し、どういう差異を捨象したのか、つまり、どういう包摂粒度を用いたのかという情報が適切に（機械可読な形として）記述されることが望ましいといえる。本稿ではこうした包摂粒度の情報の符号化に関する諸問題についてまとめ、幾つかの論点について議論したい。

2. 漢字をめぐる 2 つの視点

常用漢字表をはじめとする現在の日本における日本語用漢字処理の実務においては、字形の上に『字体』という抽象形状の粒度を設けており、また、同様な音義を指す異体字関係にある複数の『字体』をまとめたものとして『字種』という概念を想定している。ここで『字体』というのは同じの骨格の字形をまとめた抽象形状粒度であり、字形デザイン差を捨象したものが字体に相当するといえる。ただ、どういう差異が字形デザイン差でありどういう差異が字体差になるか判断に迷う例も多々あり、実務上、曖昧にせざるを得ない場合も少なくない。いずれにしても、『字種』の

ような漢字の音義・形態素やテキスト解釈上の弁別要素という観点から見た粒度概念とは別に、筆法やフォントデザイン、印刷上の慣習等から見た字形デザイン差の概念に基づく粒度概念があった方が漢字の文字概念を考える上で便利だといえ、そうした抽象形状に基づく視点と言語・テキストからの視点との交わる部分に漢字の文字概念が成立する以上、その両者の対応関係を意識することは実務上も必要なことであるといえる。

一方、JIS X 0208/0213 で符号化された漢字や UCS の統合漢字といった汎用文字符号における抽象文字はある範囲の似た形状の字体をまとめたものとされており、抽象形状に基づく視点と言語・テキストからの視点の両者を勘案したようなものといえる。また、字体という抽象形状に基づく粒度を設定した場合、抽象文字-字体-字形の３粒度からなる階層を想定することができる。つまり、抽象文字と字形の間に、字形デザイン差を捨象した抽象的なグリフというものを想定している訳である。というか、むしろ、字形デザイン差を捨象した抽象形状としてのグリフというものを想定しないと抽象形状に立脚した文字概念としての漢字の抽象文字をうまく記述できないというべきかもしれない。つまり、字形デザイン差を捨象した字体というものを基礎に、『字体の包摂規準』というものを定めて同一視可能な異体部品を列挙するという方法でコードポイントの包摂範囲を明確化している訳である。実際には、ある字形差が字形デザイン差か字体差か判別しづらいケースも多々あり、JIS X 0208/0213 の漢字の字体の包摂規準や UCS の統合漢字の符号化作業で用いられている IRG Working Document Series (IWDS) [2] 1: List of UCV (Unifiable Component Variations) of Ideograhs には字形デザイン差といえるものも含まれている。

要するに、字形デザイン差を捨象した抽象形状としての『字体』という概念や『字体の包摂規準』で統合された抽象文字というものは、一見、漢字の抽象形状の粒度概念のように見えるが、実際には、字形デザイン差を機械的な基準で捨象しただけではうまく線引きできないような代物といえる。とはいえ、実務上（あるいは、形式上)、形の情報からなるべく客観的（無知識的、文脈自由的）に判断可能な規準を作る必要があり、ここに原理的な困難さが生じてしまう訳である。

3. 包摂規準とは

一般に文字符号の符号位置はある範囲の字形の集合を表現しているが、その包摂範囲は包摂規準と呼ばれるルールの集合によって表現される。

こうした包摂の概念は ISO/IEC 10646-1:1993 で理念的に示されていたが、*1 JIS X 0208:1997 では実際に網羅的な包摂規準の集合を示し、各符号位置のセマンティクスを明確化した。

包摂規準は、通常、同一視される字体（字形）を列挙する形で表現される。また、その包摂規準の例外となる適用除外文字が列挙される。この２つが包摂規準を定義する要素といえるが、*2 この他に適用文字例も示されている。

同一視される字体（字形）は、実際には、字体（字形）だけでなく、漢字構造のパターンで示されるものもある。

4. 多粒度漢字構造情報

多くの漢字は偏と旁などの部品の組み合わせによって構成されている。こうした漢字の部品の組合せ構造に関する情報のことを「漢字構造情報」と呼ぶことにする。[3] 漢字構造情報の機械可読な表現法として幾つかの形式が提案され利用されてきたが、[4] Ideographic Description Sequence (IDS) 形式が ISO/IEC 10646 [5] の一部として標準化されている。

漢字構造情報は部品の組合せ方を示すオペレーターと部品からなる構文木で表現できる。IDS はオペレーターとして IDC (Ideographic Description Characters), 部品として UCS の統合漢字および部品用文字を用いたものであるが、部品としてそれ以外のものを用いることも原理的には可能である。

CHISE 文字オントロジー [6][7] では、抽象文字より細かい包摂粒度として、統合字体粒度、抽象字体粒度、抽象字形粒度、例示字形粒度を設けている。また、抽象文字より荒い包摂粒度として超抽象文字粒度を設けている。そして、これらの包摂関係を記述している。こうした抽象文字以外の包摂粒度も部品として用いることで、抽象字体粒度の漢字構造や抽象字形粒度の漢字構造といった包摂範囲付きの漢字構造を表現することができる。これを「多粒度漢字構造情報」と呼ぶことにする（図 1)。[8]

本稿では、包摂粒度付き文字情報を、超抽象文字は「〈*字*〉」、抽象文字は「〈字〉」、統合字体は「{字}」、抽象字体は「字」、抽象字形は「《字》」、例示字形は「『字』」のように表記することにする。

5. 包摂規準の書き換え規則化

適用除外を無視すれば、包摂規準は漢字構造情報の構文木の部分木に対する書き換え規則と看做すことができる。そして、漢字構造情報を項と看做すと、包摂規準を用いて漢字構造情報を簡約化する項書き換え系を考えることができる。

包摂規準を項書き換え系の書き換え規則に変換する方法

*1 2000 年版では附属書 S にまとめられた。

*2 この２つの要素を用いて、ある符号位置に対応する例示字体（字形）に対し、それが適用除外文字でなければ、その例示字体（字形）中の部品を同一視される他の字体（字形）に置き換えたものもその符号位置に対応するものとする訳である。

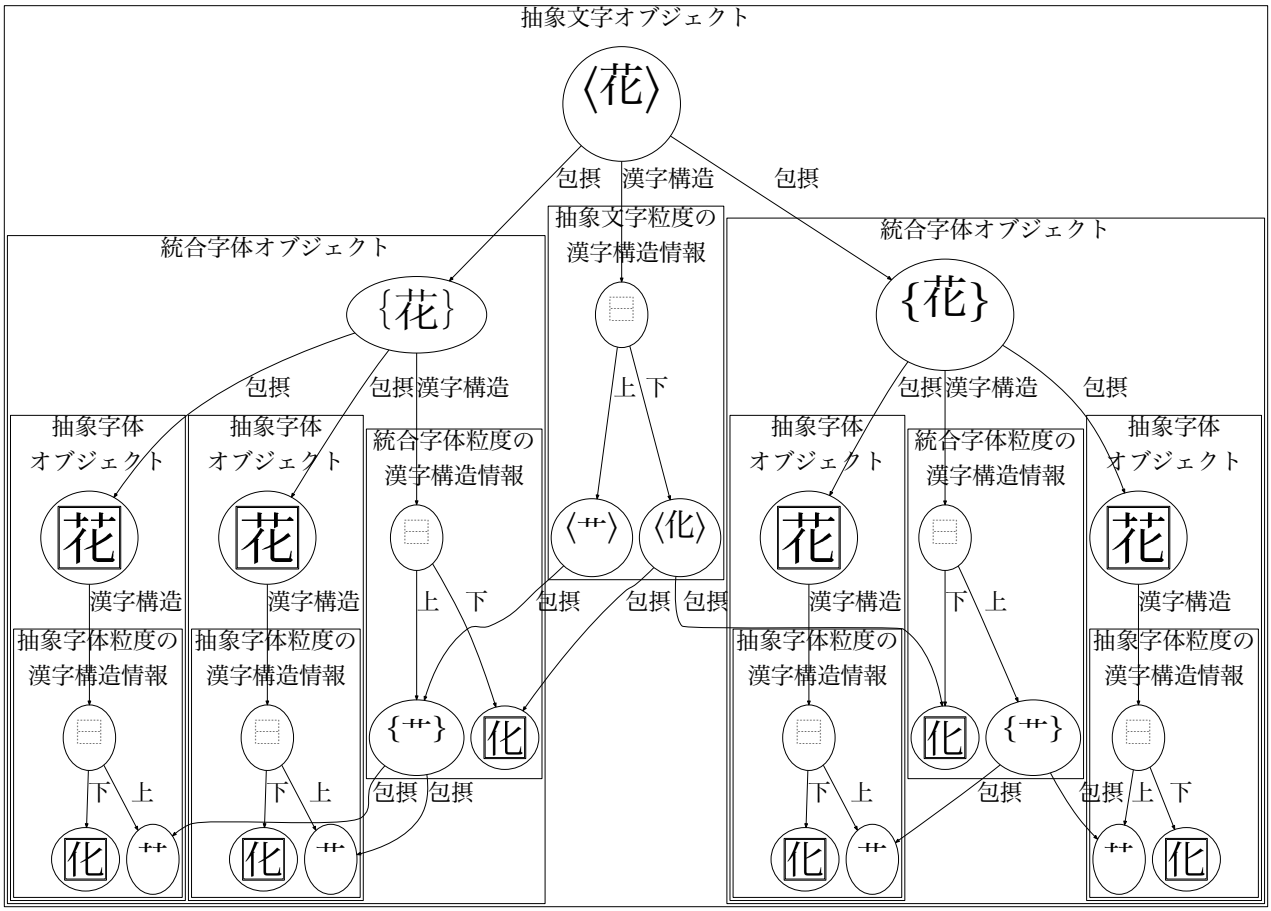

図 1 多粒度漢字構造情報の概念図（花）

としては、包摂規準の中で同一視されるものとして列挙されている各パターンの内、その 1 つを代表パターン（部品）とし、それ以外を異体パターン（部品）として、異体パターンを代表パターンに書き換える規則と看做す方法が考えられる。こうして異体パターンを含む漢字構造情報を代表パターンからなる漢字構造情報に正規化する訳である。[*3]

もうひとつの方法としては、包摂規準に対応する抽象的なパターンを表現する項を設け、包摂規準の中で同一視されるものとして列挙されている各パターンからこの抽象パターンへの書き換え規則とする方法である。

例えば、

艹 → 〈艹〉 (1)
艹 → 〈艹〉 (2)
匕 → 〈*匕*〉 (3)
匕 → 〈*匕*〉 (4)

という書き換え規則がある時、「花」は

花 = ⿱艹化
→ ⿱〈艹〉化 ; 書き換え規則 (1) を適用
= ⿱〈艹〉⿰亻匕 ; 「化」を分解
→ ⿱〈艹〉⿰亻〈*匕*〉 ; 書き換え規則 (3) を適用

という風に書き換えることができる。同様に、「花」は

花 = ⿱艹化
→ ⿱〈艹〉化 ; 書き換え規則 (1) を適用
= ⿱〈艹〉⿰亻匕 ; 「化」を分解
→ ⿱〈艹〉⿰亻〈*匕*〉 ; 書き換え規則 (4) を適用

という風に書き換えることができる。

前者の方法は部品に包摂粒度の概念がないため、そのままでは多粒度漢字構造情報に適用できない。そこで、後者の方法を採ることにする。[*4]

いずれにしても、こうした項書き換え系で漢字構造情報に書き換え規則を適用して書き換えるという行為を、適用可能な書き換え規則がなくなるまで適用すると、もうこれ以上書き換えることができない標準形を得ることができる。

ある包摂規準の集合において、複数の（多粒度）漢字構造情報の標準形が一致した場合、それらはこの包摂規準において同一と看做すことができる。よって、この項書き換

[*3] [9] では CHISE 漢字構造情報データベース [4] を用いた IDS の正規化アルゴリズムとそれによる漢字の同一性のチェック手法を提案している。

[*4] 単一の包摂粒度しか必要ないとしても、単一の包摂粒度間での書き換えを行った場合、ある漢字構造情報中の部品が既に書き換えられたものかそうでないかが判別し辛く、JIS X 0208:1997 等が課している同一部品の複数回書き換えを禁止する制約を実装するためには何らかの工夫が必要であると考えられる。

え系は文字の同値性を調べるための系と看做すことができる。

例えば、前述の例の場合、「花」と「花」に書き換え規則 (1)〜(4) を適用した結果、同じ標準形

⿱〈艹〉⿰亻〈*匕*〉

を得ることができた。このことから、この２つの字形はこれらの書き換え規則によって表現された包摂規準の下では同一視できることが判る。

ところで、この例では書き換え規則を左から適用していったが、書き換え規則の適用順序は任意であり、

花 = ⿱艹化
= ⿱艹⿰亻匕 ; 「化」を分解
→ ⿱艹⿰亻〈*匕*〉 ; 書き換え規則 (3) を適用
→ ⿱〈艹〉⿰亻〈*匕*〉; 書き換え規則 (1) を適用

という風に書き換えても良い。この例のように、どういう順序で書き換えても同じ結果になることを『合流性がある』という。

しかしながら、合流性は一般には成立しない。例えば、

匕 → 〈*匕*〉 (4)
匕 → 〈*七*〉 (5)

という書き換え規則があった場合、「花」の右側の部品「匕」には (4) と (5) の２つの書き換え規則が適用可能であり、(4) を適用した場合と (5) を適用した場合で標準形が異なってしまう。

この例のように、ある項に２つの書き換え規則を適用可能な場合があり得るが、両者のそれぞれを適用して異なる結果を得る時、それらの異なる結果を『危険対』と呼ぶ。危険対が存在する場合、書き換え規則の適用順の違いによって、標準形が異なるものになり得る。

また、書き換え規則の集合と項の組合せ次第では、適用可能な書き換え規則が永遠になくならずループしてしまうことがある。[*5] 例えば、

己 → 巳 (a)
巳 → 己 (b)
巳 → 巳 (c)

という書き換え規則の下では、「巻」の書き換えは永遠に続く。これを『停止しない』という。また、逆に、いつか適用可能な書き換え規則がなくなることを『停止性が存在する』という。

結局、この項書き換え系を文字の同値性のチェックに用いるためには、停止性と合流性の２つの性質を満たしている必要がある。この両者を満たしていることを『完備である』という。[*6]

JIS X 0208/0213 の漢字の字体の包摂規準や UCS における事実上の包摂規準といえる IWDS 1 は、残念ながら、そのままでは完備な系にならない。完備な系にするためには、異なるグループの抽象部品が同じ形の部品に衝突しているケースにおいて、その粒度において同じ形と判定されるとしても別オブジェクトとして区別するといったなんらかの工夫が必要になると考えられる。そして、このような工夫を施して完備な系にしたものを『符号化された包摂規準（包摂ポリシー)』と呼ぶことにする。

6. 包摂規準の部品オブジェクト化

包摂規準の内、部品に対する書き換えとして記述可能なものは、同一視されるものとして列挙された全ての字体（字形）を包摂する抽象文字オブジェクトとして記述できる。CHISE 文字オントロジーでは抽象文字や字体・字形といった包摂粒度の異なるオブジェクト間の包摂関係を `->denotational` 素性と `->subsumptive` 素性を用いて表現しているが、[*7] こうした文字単位での包摂関係の記述と同様な方法で部品オブジェクト間の包摂関係を記述する訳である（図 2)。こうすることで、抽象文字粒度に相当する（字体の）包摂規準や、字体粒度に相当する（抽象字形の）包摂規準といったものを混在して記述した上で、それぞれの関係を包摂関係として書くことが可能である。

5 節で述べたように、包摂規準を書き換え規則と看做す場合、例えば図 2 の例の場合、

{艹} → 〈艹〉 (1)
艹 → {艹} (2)
艹 → {艹} (3)
丬 → 〈艹〉 (4)

という４つの書き換え規則が得られる。このうち、(1) と (4) は抽象文字粒度への書き換え規則であり、(2) と (3) は統合字体粒度への書き換え規則である。

もし、抽象文字粒度の包摂規準の集合を用いたいなら、この (1) 〜 (4) の全てを用いれば良い。また、もし、統合字体粒度の包摂規準の集合を用いたいなら、

艹 → {艹} (2)
艹 → {艹} (3)
丬 → {丬} (4')

という風に、例えば図 2 の木を統合字体粒度の階層までで止めたものを使えば良い。

このように、文字オブジェクト間の継承関係を用いることで、複数の包摂粒度の情報を 1 つの包摂関係の木（有向グラフ）にまとめることで、コンパクトに記述することができる。

包摂規準の中には文字単位のオブジェクト間の包摂関係と一致するものも少なくなく、こうした場合は既

*5 JIS X 0208:1997 や JIS X 0213 の包摂規準に関する規定では、ひとつの部品（「部分字体」）に対して複数の包摂規準を順次適用することを禁止することで、停止性を保証している。
*6 項書き換え系の完備化アルゴリズムとしては「クヌース・ベンディックス完備化アルゴリズム」が知られている。
*7 字体より荒い包摂粒度に対しては `->denotational` 素性を用い、字体（ないしは、統合字体）よりも細かい包摂粒度に対しては `->subsumptive` 素性を用いるようにしている。

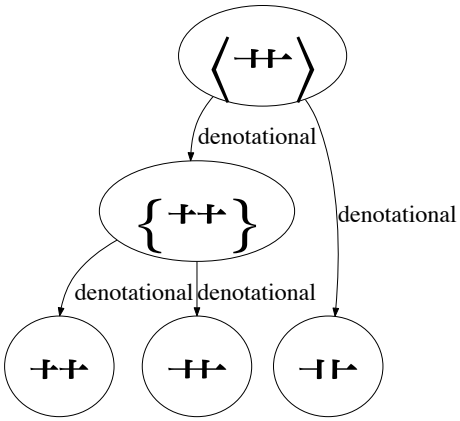

図 2　部品の包摂関係の概念図（草冠）

存の文字間の包摂関係の情報がそのまま利用できる。しかしながら、文字としては異体字関係にないが、部品としては同一視され得るようなものは問題である。こうした場合、階層的素性名方式を用いて、部品専用の包摂関係素性（例えば、`->denotational@component` 素性と`->subsumptive@component` 素性）を用いたり、文字単位のものとは別に部品用のオブジェクトを用意するといった方法が考えられる。

7. 包摂粒度の符号化

包摂規準は対応する抽象部品オブジェクトによって表現することができる。このような部品オブジェクト間の包摂関係の階層がある時に、全ての部品オブジェクトの包摂関係の森をある包摂粒度で切ることで、その包摂粒度に対応した抽象部品の集合を得ることができる。

しかしながら、包摂規準の集合は必ずしも同じ包摂粒度で揃っているとはいえない。また、JIS X 0208:1997, JIS X 0213:2000, JIS X 0213:2004 や UCS 統合漢字を対象とした IRG Working Document Series (IWDS) 1, 2 [2] といった包摂規準の集合は概ね抽象文字の粒度を対象としたものであるが、包摂粒度が概ね同一な部分と細かさに差異がある部分（ある集合では包摂されるものがある集合では分離されていたりする）がある。このため、単純に抽象文字の包摂粒度で切るだけでは問題がある。

この問題は包摂規準の集合で表現される包摂ポリシーに名前を付けることと、異なる包摂ポリシーに属する包摂規準の間に項書き換え系を構成可能な関係を記述することで解決できると考えられる。

この内、包摂ポリシーの名前付け（あるいは、各包摂基準に対する番号付け）は容易であるが、異なる包摂ポリシーに属する包摂規準の間の関係を完備な項書き換え系を構成可能な形で記述するためには解釈と工夫が必要な部分があり、一般には機械的に実現可能ではないといえる。これは一つには既存の包摂ポリシー自体が符号化されていない（完備な項書き換え系を機械的に生成できない）ことによるといえる。よって、複数の包摂粒度間の関係を機械的にチェックするためにも、個別の包摂ポリシーの符号化が不可欠といえ、その上で、代表的な包摂ポリシー間の関係を部品オブジェクト間の包摂関係として記述する（あるいは、機械的に変換する）ための仕組みを用意することが重要であると考えられる。

8. 包摂範囲の記述

包摂ポリシー全体や異なる包摂ポリシー間の関係全体を完備化・符号化するのは必ずしも容易ではないが、個々の漢字に対し、字種-字体-字形のような異なる包摂粒度のものの間の包含関係を記述するのは比較的容易であるといえる。一方の包摂ポリシーでは区別され他方では区別されないような差異が字体の差異であるか字形デザイン差であるか判別しづらいような場合であっても、もし包含関係になっているのであればその関係を記述すること自体は簡単である。また、互い違いになっている場合でも、包含関係の木構造に分割し、その分割された木の集合として包摂規準を表現することが可能である。それぞれの包摂ポリシーで「字体」のような同一の名称が異なる包摂粒度を指している場合も、同様な方法でその関係を記述することができる。

個々の漢字に対し、異なる包摂粒度の文字オブジェクト間の包摂（包含）関係が記述されている時に、図 1 のように、その各粒度のオブジェクトに多粒度漢字構造情報を張り付けた場合、多粒度漢字構造情報の部品オブジェクト間の包摂関係が定義されていれば、部品オブジェクト間の包摂関係から項書き換え系を生成し、多粒度漢字構造情報に対する書き換え計算を行うことができる。ここで、多粒度漢字構造情報を目的とする包摂粒度に正規化した結果と文字間の包摂関係が一致していればその全体を項書き換え系

と看做したときに合流性が存在することになる。このような文字とその部品の包摂関係のグラフを書くのは、包摂規準の集合を完備化しながら過不足無く包摂ポリシーを記述することよりも容易であるといえる。

9. おわりに

漢字を含むテキストの符号化において、しばしば、UCS や JIS X 0208/0213 ではユニファイされてしまった差異を区別したいケースがあり、そのための仕組みが提案されてきた。現在では IVS がそのための標準として存在しており、曲がりなりにも UCS の統合漢字では 1 つのコードポイントに包摂された複数の字体・字形を区別することが可能となった。このことは、漢字を含む電子テキストが複数の包摂規準の混在したものになったことを意味しており、そのための漢字処理モデルが必要だといえる。本稿では、このような複数の包摂規準・包摂ポリシーが混在している場合の文字処理の可能性について議論した。

そもそも漢字の形の微小な差異を区別したい場合、その何を何のために区別しようとしているのかを考えなければならない。形の差異で表現したいものは形やその差異そのものというよりはテキスト解釈に影響し得るなんらかの『意味』上の差異であり、漢字の表語文字としての性格を鑑みれば、それは音義や形態素やその文脈上の振舞いの差異であろう。しかしながら、現状の文字符号化の仕組み（符号化文字モデル）のもとではそうした情報に立脚して抽象文字を構成することができず、主に形の情報に立脚しつつ、それが表現した何かに配慮しつつ漢字の抽象文字を構成するしかないといえる。包摂規準は、形式上、字形デザイン差を捨象した抽象形状を与えるものであるが、実際には形だけでは規定し得ないものといえる。よって、原理的に包摂規準には漢字の形状だけでは解決しないような歪や例外の類が内在し得るといえ、そうした問題のある箇所を検出することはその機械処理において必須の要件の一つだといえる。

複数の包摂規準や包摂ポリシーをサポートした漢字処理を行うためにはそうしたものを機械可読な形で表現し、機械処理できることが重要である。本稿では多粒度漢字構造情報と包摂規準の書き換え規則化に基づく計算可能なモデルを提案し、このモデルでサポート可能なケースと難しいケースについて議論した。この難しいケースは部品レベルにおける別字衝突の場合だといえ、字形の情報だけでは解釈が難しいケースだと考えられる。

こうしたケースも含めて、より抜本的な解決を計るためには、形以外の情報も含めた漢字弁別（あるいは、弁別に関わらない字形デザイン差）に関する知識を機械可読な形で蓄積し利用する必要があるといえる。このためには古典中国語の形態素やその用例の情報を統合した [10] 漢字のグリフコーパスやグリフオントロジーを整備していくことが重要であるといえる。

ただ、漢字の抽象形状に基づく視点だけでも扱える問題は少なくなく、異なる包摂粒度を共存させるための漢字処理を実現することは既存の符号化文字の仕組みの重要性を鑑みれば必要なことだといえる。このためには本稿で議論した包摂粒度の符号化の枠組だけでなく、実際に複数の粒度からなる漢字のグリフ間の関係や多粒度漢字構造情報の整備が重要である。そのため、CHISE project では、現在、このような観点に基づいた粒度情報と多粒度漢字構造情報の整理拡充作業を行っている所である。[11]

参考文献

[1] : Ideographic Variation Database, http://unicode.org/ivd/.
[2] : IRG Working Document Series, http://appsrv.cse.cuhk.edu.hk/~irg/irgwds.html.
[3] 守岡知彦：CHISE 漢字構造情報データベース，東洋学へのコンピューター利用第 17 回研究セミナー，pp. 93–103 (2006).
[4] 守岡知彦，クリスティアン・ウィッテルン：文字データベースに基づく文字オブジェクト技術の構築，情報処理振興事業協会平成 13 年度 成果報告集，情報処理振興事業協会 (2002). http://www.ipa.go.jp/NBP/13nendo/reports/explorat/charadb/charadb.pdf.
[5] International Organization for Standardization (ISO): *Information technology — Universal Coded Character Set (UCS)* (2014). ISO/IEC 10646:2014.
[6] Morioka, T.: CHISE: Character Processing based on Character Ontology, *Large-scale Knowledge Resources (LKR2008)*, LNAI, No. 4938, pp. 148–162 (2008).
[7] 守岡知彦：文字オントロジーに基づく文字処理について，情処研報， Vol. 2006, No. 112, pp. 25–32 (2006). 2006-CH-72.
[8] 守岡知彦：CHISE に基づくグリフ・オントロジーの試み，人文科学とコンピュータシンポジウム論文集—デジタル・ヒューマニティーズの可能性，情報処理学会シンポジウムシリーズ，Vol. 2009, No. 16, 情報処理学会，情報処理学会，pp. 9–14 (2009).
[9] 川幡太一：IDS による UCS 漢字の「同一性」の判定手法，東洋学へのコンピューター利用第 17 回研究セミナー，pp. 105–119 (2006).
[10] 守岡知彦：古典中国語形態素コーパスの Linked Data 化の試み，じんもんこん 2013 論文集，情報処理学会シンポジウムシリーズ，Vol. 2013, No. 4, 情報処理学会，情報処理学会，pp. 187–194 (2013).
[11] 守岡知彦：CHISE における漢字字体・字形粒度の整理規準について，東洋学へのコンピューター利用第 26 回研究セミナー，pp. 153–190 (2015).