議論解析システムのアルゴリズム改良

東京大学大学院工学系研究科　　学生会員　○佐藤　岳文
東京大学大学院工学系研究科　　正会員　　堀田　昌英*

１．背景および目的

近年，公共政策の決定における市民関与・住民参加の必要性が叫ばれており，数々の政策に関して，住民などを交えた当事者による議論が行われている。しかし，多くのケースにおいて議論は意見の対立から長期化し，円滑な政策決定や施策に寄与しているとは言い難い。

討議型意思決定が困難な理由として，参加する当事者の価値観の多様さと，それに伴う議論のプロセスの複雑化が挙げられる。そこで問題の解決には，意見を分析し，議論の要点を示すことで意思決定を支援することが有効と考えられる。意見データ解析システム「CRANES(Coordinator for Rational Arguments through NEsted Substantiation)」は，議論の趨勢をコンピュータで分析し，各意見の集める注目度を定量的に測ることで議論の要点となる意見・議題を抽出する情報システムである[1)]。

本システムには，注目度指標の計算に非常に長い計算時間を要するという課題があり，著しく長時間に及ぶ政策討議では分析が現実的に不可能であった。そこで，大規模な議論への導入のために，計算時間短縮のためのアルゴリズムの改良を行った。

２．改良の概要

議論の要点となる意見を抽出するための指標となるのが，意見の注目度を示す NCI（Node Concentration Index）である[2)]。

一連の議論を小単位に分割してノード(node)とし，あるノードに対する質問や返答をその「子ノード」として接続していくと，議論単位からなるツリー構造を持つことになる。今，ある議論ツリーにおいてｘ個目のノード（注目ノード）が生成され，その後ｊ個のノードがツリーに追加され，そのうちｃ個が注目ノードの子孫ノードであるとする。この時，このノードのＮＣＩは，以下のように定義される。

$$\mathrm{NCI} = -\log\left[\sum_{c\le d\le j} P(x, j, d)\right]$$

ここで P(x,j,d)は，注目ノードの子孫数が d(0≦d≦j)である確率を，議論の傾向を基に推測したものである。旧システムで非常に時間がかかっていたのは，この P(x,j,d)の算出である。

改良で用いたのは，実際のツリーに似せた擬似ツリーをランダムに組み立てるシミュレーションをプログラム上で繰返し行い，全回数(以下，試行回数)のうち注目ノードが d 個の子孫を持っていた回数を，確率 P(x,j,d)の収束値と解釈する手法である。

シミュレーションでは，実際のツリーが構築されていく過程と同様に，主題となるノードを起点としてノードを一つずつ既存のものに接続していくが，あるノードを接続する際，それが既存ノードのうちどのノードに付くかを決定する上で乱数を用いる。シミュレーションで作成された擬似ツリーは，大きさは同じであってもツリーの構造は異なるので，当然それぞれのノードの子孫ノード数も様々な値を取る。したがって十分に多くの回数にわたって擬似ツリーを作成したとき，ｘ個目のノードが d 個の子孫ノードを持つ回数は，試行回数 × P(x,j,d)となる。よって，これを逆算することで P(x,j,d)を導き出すことができる。

改良後のアルゴリズムの概要を図１に示す。

キーワード　公共政策，意思決定，議論支援，情報社会基盤
*連絡先　　〒113-8656 東京都文京区本郷 7-3-1　東京大学大学院工学系研究科社会基盤学専攻　TEL03-5841-6088

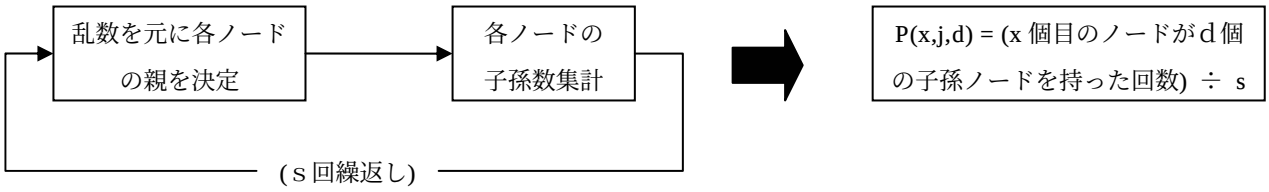

図１　改良後アルゴリズムのフローチャート

３．改良結果の検証

改良したアルゴリズムのシステムで，計算時間の短縮が達成されたかを検証する。

まず，総ノード数 1000 の議論ツリーに対し，試行回数を様々に変えて計算を行った。

また，総ノード数 1～500 のツリーについて，改良前及び改良後の試行回数 1000，3000，5000，１万回の５通りの方法で計算し，それぞれ計算に要した時間を計った。（Java 環境:Java(TM) 2 Runtime Environment, Standard Edition 1.4.2_06，OS:Windows XP Home Edition，プロセッサ:Intel(R) Pentium(R)4 CPU 2.40GHz）それぞれの結果を図２，図３に示す。

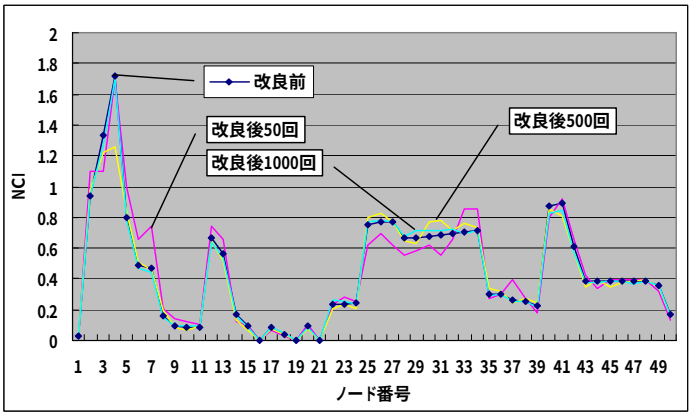

図２　総ノード数 1000 の計算結果比較

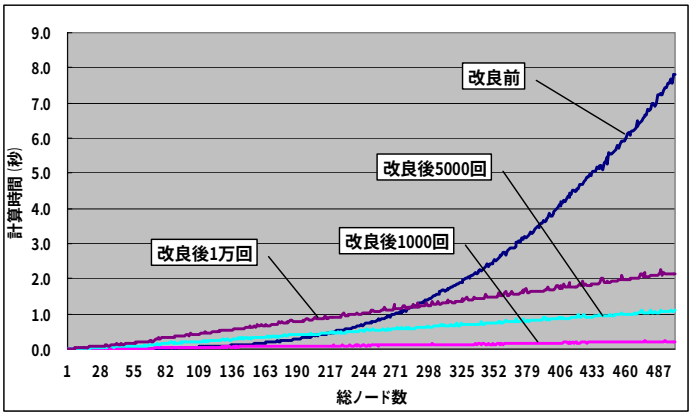

図３　総ノード数及び各計算方法と計算時間の関係

・計算精度

試行回数を増やせば精度は確かに上がるが，ノード数と同じかそれ以上の試行回数ならば改良前とほぼ同じ結果が得られる。また，各ノードの注目度を，グラフを見て判断する場合は，それよりもかなり小さい値（約 10 分の 2~4）の試行回数でも，さほど問題無く NCI が高いノードを見つけ出すことが可能と思われる。

・計算時間

改良後の方法での計算時間は総ノード数と試行回数に比例している。一方，改良前の方法での計算時間は，総ノード数の三乗にほぼ比例している。この結果から推測すると，ノード数 11000 のツリーの計算に要する時間は，改良前では約１日かかる。同じツリーを改良後の試行回数 1 万回で計算すると，約 1 分で済んだ。

４．結論

今回の改良で，計算に要する時間とコンピュータのメモリを大幅に削減することが出来た。今後，発言の数が数万に及ぶような大規模な議論に対しても CRANES を導入し，短い時間で結果を示すことが可能になる。また，少ないリソースしか持たないシステムにも実装できることから，CRANES の活用範囲は大きく拡がると思われる。

参考文献

1)堀田昌英，神野由紀：参画型パブリック・マネジメントの情報基盤 CRANES の開発 土木学会論文集VI 52 号
2)堀田昌英，榎戸輝揚，岩橋伸卓：多元的議論構造の可視化手法，社会技術論文集 vol.1,67-76