

分散探索に基づく分散問題解決モデル*

北村 泰彦

小川 均†

1996年 2月 5日

1 はじめに

計算機技術とネットワーク技術の進歩によって、現在の計算機の利用形態は、従来の単体集中利用から、複数計算機の分散利用へと移行しつつある。しかしながら現在のところ、分散している計算機を効果的に利用するためには人間の介在が不可欠であるが、将来はそれぞれの計算機が自らの判断で、他の計算機と協調しながら、大規模な問題を効率よく解決するような技術が必要になるであろう。このような計算機の協調に関する研究は人工知能だけでなく、データベース、ソフトウェア工学、プログラミング言語、CSCW、分散処理などの分野においても注目されている [14]。

分散人工知能は主に、人工知能の立場からこのような計算機の協調について研究している [11, 3]。従来の人工知能の目標を一人の人間の知性を計算機上に模倣することであると見なすならば、分散人工知能はいわば人間の組織や社会を計算機ネットワーク上に模倣することであると見なすことができる。問題解決の視点からみたとき、従来の人工知能と分散人工知能のアプローチには違いが出てくる。分散人工知能が対象とする世界はその構成物や環境が動的に変化する開世界であり、そこでは解やその導出のための手続きの最適性という概念さえ定義することが不可能な場合もある。したがって、最適解を最適効率で導出することを要求することはほとんど意味をなさないことがある。そこでは、エージェントや環境に分散して存在している制約を充足させながら、できるだけ良い解をできるだけ効率よく導出する相対的な手法が求められる。また、エージェントの能力の視点から見ても、それぞれのエージェントは局所的で相対的な能力しか持たない。したがって、完全な知識を前提にした単一で強力なシステムよりも、環境の変化に柔軟に対応しながら、他のエージェントと協調的に問題解決を行うようなシステムが求められる。すなわち、分散人工知能は問題解決においてもエージェントの能力においても相対的な尺度しかない開世界を研究の対象にしており、そのような環境で動作するエージェントを作り上げることを研究の目標としているといえる。

このような分散人工知能の目指す目標を達成するためには従来の人工知能の手法を包含しつつも、さらにそれを拡張してゆく必要がある。本稿で紹介するいくつかの手法も、従来の閉世界を前提とするような手法を基にして、それらを拡張してゆくことによって分散人工知能の要求しているものを目指そうとしている。これまでの分散人工知能の研究は特定の応用領域を対象に、特定のアーキテクチャを前提として研究を進めているものが多かった。しかしながら、そのようなアプローチは時として問題をより一層複雑に見せてしまったり、得られた手法を他の応用領域に適用することが困難にしてしまうことがある。また、その手法の有効性や限界、手法間の関係が必ずしも明確にされていないという問題点もあった。したがって本稿では、従来の問題解決が探索により定式化されてきたように、分散人工知能の一分野である分散問題解決を分散探索により定式化し、これまでの手法を分類、比較してゆく。また、問題解決の過程で問題が変化する動的問題に対するアプローチも分散探索の視点から紹介する。すなわち、これまでの研究成果を問

*TR93-6R1:計測と制御第33巻第1号[特集] 知能システム初稿

†立命館大学理工学部

題に依存する様々な前提等を排除して、分散問題解決の基礎にあたる部分を紹介したいと願っている。したがって、本稿の内容は先に述べた分散人工知能の目標を達成するには大きなギャップが存在するが、それを満たすことが今後の研究課題であると考えている。また紙面の都合から分散人工知能の様々な研究課題の中のごく一部しかとりあげていないが、一般的な解説は石田と桑原によるもの [6, 10] がよくまとまっており、そちらを参照して頂きたい。

2 分散問題解決とその定式化

まず議論をはじめるとして、本稿で扱う分散問題解決の定義を与えておこう。分散問題解決とは“一つの大局的問題を疎結合、複数のエージェント（たとえば計算機）により解決する”ことである。特徴の一つは疎結合であるという点である。具体的には、LAN などの通信ネットワークにより計算機が結合されていることを意味し、通信コストが計算コストに比べて大きく、エージェント間でのメッセージ量がシステム全体の性能に大きな影響を与える。この点において、共有メモリ等の利用によってエージェント間の通信コストが極めて小さい並列システムによる問題解決とは異なっている。

もう一つの特徴はシステムに与えられる問題は一つであり、それを複数のエージェントにより（協調的に）解決するという点である。この点において筆者らは分散人工知能に関する類似概念であるマルチエージェントシステムと区別している。マルチエージェントシステムでは各エージェントがあらかじめ個別の問題を持っており、共有資源により起因する競合を解消しながら問題を解決する。一般にマルチエージェントシステムのほうが大局的な前提の少ない、より開放型のシステムの実現を目指しているといえる。もちろん、分散問題解決とマルチエージェントシステムの研究は関連する部分が多く、分散問題解決はマルチエージェントシステムの基本的な問題解決手法を与えるものであると見なすことができる。

一般に人工知能で扱う問題解決はその過程がアルゴリズムとして明示的に記述できない非決定性が伴う。したがって、その過程には試行錯誤がともない、探索問題として定式化することができる [1]。分散問題解決では問題解決に複数のエージェントが関与し、単一のエージェントでは与えられた問題全体を解決することができないと仮定している。すなわち、各エージェントは問題空間を部分的にしか探索できないわけで、全体解を求めるためにはエージェント間の協調が必要になる。ここで、分散問題解決においては二つの非決定的な要因が存在することが明らかになる。そのひとつは先にも述べたように問題解決における非決定性である。もう一つは問題解決を行うエージェントをどのようにして見つけるかという問題割り当てにおける非決定性である。一般に分散問題解決はこのような二つの非決定性が入り組んでおり、その解決を複雑にしている。

分散問題解決を定式化するために問題を図 1 に示す AND/OR グラフにより表現しよう。問題は節点により表され、初期問題はグラフの根節点で示される。AND 分岐あるいは OR 分岐は親問題が子問題に分解されることを表す。ある問題が AND 分岐しているばあいは、その子問題全てを解決する必要がある。また、OR 分岐の場合はその子問題のいずれかが解決されればよい。葉節点にあたる問題は素問題であり、それ以上分解することができない。以下の議論では問題分割の視点から AND 分岐に注目し、(1) 問題が AND 分岐のみから表現される場合、(2) AND 分岐と OR 分岐とから表現される場合、(3) OR 分岐のみから表現される三つの場合の分類を行う。

次にエージェントの能力について述べる。エージェントはある問題が与えられたとき、その問題を (1) 全く解決できない、(2) 部分的に解決できる、(3) 問題を分割できる、(4) 完全に解決できるかのいずれかである。(2) の場合は子問題の一つだけ持つ OR 分岐節点、(3) の場合は AND 分岐あるいは OR 分岐節点、(4) の場合は葉節点でその問題は表現され、エージェント名がラベル付けされる。ここで分散問題解決の前提として、いずれの節点もいずれかのエージェントによりラベル付けされているが、一つのエージェントが全ての節点をラベル付けすることはないとする。したがって、単一のエージェントにより問題解決が行わ

れることはないが、複数のエージェントの協力により問題は解決されることになる。例えば、図1では部分問題は W,X,Y,Z の四つのエージェントによりラベル付けされており、部分問題 7,8 は二つのエージェント X,Y により解決可能である。

以上の定式化から、分散問題解決は適切なエージェントに問題が割り当てられ、そのエージェントが適切な局所的問題解決を行うときに成功する。ただし本稿では二つの議論を分離して行う。

3 問題割り当て法

問題割り当てはオペレーションズリサーチや分散処理におけるスケジューリング技法や負荷分散技法との関連が深いが、一般にスケジューリング問題は計算量の面から、最適解を求めることが困難な NP 完全問題に分類される [13]。それに加えて、分散問題解決では (1) AND グラフはあらかじめ明示的に与えられないかもしれない。すなわち問題解決の過程で明らかにされて行くので、あらかじめスケジュールすることが不可能である。(2) それぞれのエージェントに関する完全な大局的情報を用いることはできない。すなわち、大局的情報を得るためには通信コストがかかる上、エージェントは非同期に動作するので完全な情報を持つことは不可能である。(3) エージェントの関係は対等である。すなわち、あるエージェントが問題割り当てに関して、他のエージェントを完全に制御することができない。

したがって、このような環境においては、動的負荷分散を行わなければならないが、分散問題解決の手法としては人間社会の交渉過程をメタファとした以下に示す契約ネットプロトコル [15] が提案されている。

[契約ネットプロトコル]

Step1: 問題を持つエージェントがその存在を通知メッセージを用いて、他のエージェントに知らせる。

Step2: 受け取った通知メッセージの中から、解決の可能性のあるエージェントは、その解決コストに関する情報を記述した入札メッセージを送り返す。

Step3: 入札メッセージを受け取ったエージェントはその中で、最もよいものを選択し、判定メッセージにより問題を割り当てる。

契約ネットプロトコルでは、エージェントが対等の立場で局所判断と相互選択に基づき、問題が割り当てられてゆく(契約が結ばれてゆく)。先にも述べたように、動的環境における負荷分散やスケジューリング問題は計算量の点から何らかのヒューリスティックに頼らざるをえず、契約ネットプロトコルはそのようなヒューリスティックの一つを与えるものとして、いくつかの応用事例が報告されている [16, 12]。

4 分散問題解決手法

4.1 問題分割法 (AND グラフ探索)

問題が完全に分割されている場合、問題は完全な AND グラフによって表現されるとする。そこでは OR 分岐がないので問題解決における非決定性はなく、先に述べた問題割り当てを除いて、分散問題解決の過程はほとんど自明である。解を得るためには部分問題のすべてを解決する必要があり、いずれかのエージェントが部分問題の解決に失敗すればその時点で全体的な問題解決も失敗となる。

4.2 分散制約充足法 (AND/OR グラフ探索)

問題が AND/OR グラフで表現される場合には、問題解決の過程に非決定性があり、適切に OR 選択を行わなければ解が得られない場合が存在する。その原因としては局所的制約によるものと大局的制約によ

るものに分けられる。局所的制約は単一エージェントの判断によって選択可能なものであり、次節で述べることにする。大局的制約は制約がエージェント間にまたがって存在するもので、単一エージェント独自の判断で選択を行うことができず、他のエージェントとの調整が必要になる。例を図 2 にあげよう。3 つのエージェントが存在し、エージェントはそれぞれ変数への値の割り当てをおこなう。ただし大局的制約として、 $P1 \neq P3$ 、 $P2 \neq P3$ という条件が与えられており、それぞれのエージェントが個別に変数を割り当てをおこなっては充足解が得られない場合がある。

この問題を解決するためのアルゴリズムの一つとして非同期バクトラックアルゴリズム [17] が提案されている。

[非同期バクトラックアルゴリズム]

Step1: 各エージェントは自分の解を局所的に求める。

Step2: 大局的制約にかかわるエージェントは ID 番号の小さいものから大きいものへ、解を OK メッセージを用いて知らせる。

Step3: OK メッセージを受け取ったエージェントはその解を記録し、自分の解との無矛盾性を確認する。矛盾が存在すれば自分の解を他の可能な解に置き換え、その解を Step2 と同じように OK メッセージを用いて他のエージェントに知らせる。置き換えが不可能な場合は、そのエージェント以外の解集合を NOGOOD メッセージを用いて矛盾した解を持つ最も ID 番号の大きいエージェントに送る。

Step4: NOGOOD メッセージを受け取ったエージェントは無矛盾性の確認を Step3 と同様に行う。NOGOOD メッセージがそれ以上送れない状態になると解が存在せずに終了する。

以上のアルゴリズムではエージェントの全順序関係を示す ID 番号を用いることにより、エージェントの解情報が ID 番号の大きいものに集められてゆく。解に矛盾の存在する場合には NOGOOD メッセージが逆の方向に送られてゆくことにより、非同期バクトラックが行われ、アルゴリズムの完全性が保証される。

この手法はエージェント間の制約が密接になってくると制約を満たす解を求めるために必要な通信量が大きくなり、通信コストの大きい場合には性能が低下する。充足解を求めるためのもう一つの手法は一つのエージェントに情報を集中させて、そのエージェントにより充足解を求めようとする手法である。ただし、求解を集中的に行うことは、それだけ並列性が低下することでもあり、本手法において分散が集中かは通信オーバーヘッドと並列性のトレードオフの関係にあるといえる。したがって、本手法はエージェントの組織化の問題とも関連がでてくる [4]。

4.3 分散グラフ探索法 (OR グラフ探索)

ここでは AND 分岐が存在しない、すなわちあらかじめ問題分割が行われていない場合に対する探索法について述べる。この場合の一つの解決手法は、分散制約充足法のように、各エージェントが部分解を交換し、それを統合することにより解を得るという方法である。しかしながら問題が分割されていない場合には、解をなしえない部分解の交換が多量に発生する可能性があり、系統的に解を導き出す手法が必要になる。そのような手法の一つとして、従来のグラフ探索の分散化を行った波及型探索アルゴリズム [7] がある。

[波及型探索アルゴリズム]

Step1: 初期問題をもつエージェントが解決を開始する。

Step2: 問題が OR 分割されると、部分問題のそれぞれを解決可能なエージェントに REQUEST メッセージを用いて割り当てる。

Step3: いずれかのエージェントが部分問題を解決できれば終了である。部分問題の解決に失敗したエージェントはそのことを FAIL メッセージを用いて、REQUEST メッセージを送ったエージェントに返す。

Step4: OR 分割したエージェントは送った REQUEST メッセージのそれぞれに対して FAIL メッセージを受け取った場合、分割した問題が初期問題であるならなら失敗で終了する。そうでないなら、その問題に関する REQUEST メッセージを送ったエージェントに FAIL メッセージを送り返す。

この手法では問題解決は OR 分岐にしたがって他のエージェントに広がって行き、OR 並列性が増加する。すなわち、いずれかのエージェントが部分問題の解決に行き詰まっても、他のエージェントが他の候補を並列的に解決しているの後戻りの必要がない。しかしながら本手法の問題点は探索が広がるにつれ、通信量も増加するという点である。ここでも、通信オーバーヘッドの観点からトレードオフの問題がある。すなわち、探索を広げればそれだけ (OR) 並列性が上がるが、通信量が増加するという点であり、通信オーバーヘッドの観点からの探索制御が重要になってくる [8]。

4.4 比較

以上の三つの手法を問題分割の視点から比較しよう。問題分割に関する知識があればそれだけ複数エージェントによる問題解決は容易になり、その並列性が上がる。逆に問題解決に関する知識が少なくなると、それだけ正しい解を見つけるためにはエージェント間で多くの通信が必要になる。

具体例を用いて比較しよう。図 3 は A から O の 15 の都市の交通情報に関するグラフを示している。これらの情報は図でも示されるように W から Z の 4 つのエージェントによって分散して管理されているとする。(エージェントの ID 番号は $W < Z < Y < Z$ の順であるとする。) ここで G-H 間, C-F 間は通行不可能になっている。問題は A から O までの移動経路を求めることである。それぞれのエージェントは部分的な経路しか導出できないので、この問題を解決するためにはエージェントの協調が必要になる。

この問題に対して問題分割に関する情報が完全な場合は、問題 $p(A,E), p(E,J), p(J,O)$ がエージェント W, X, Y に割り当てられ、それぞれ解 $a(A,C,E), a(E,H,J), a(J,N,O)$ を導出することで全体解が得られる。次に問題が不完全な形で分割されている場合の例を示そう。例えば、問題が部分問題 $p(A, P1), p(P2, P3), p(P4, O)$ に分割され、それがエージェント W, X, Y に割り当てられたとする。ここでの部分問題は出発地と目的地が完全に与えられていない不完全なものである。さらに制約として、W, X 間には $P1 = P2$ 、X, Y 間には $P3 = P4$ が与えられている。この問題解決において、エージェント W, X, Y が最初、それぞれの部分解、 $a(A,B,D), a(E,H,J), a(I,N,O)$ を導出したとしよう。この場合、これらの解は制約を充足しないので、4.2 のアルゴリズムにしたがって、エージェント W と Y がそれぞれ別の部分解 $a(A,C,E), a(J,N,O)$ に置き換えることにより、充足解が得られる。ここでは問題が不完全に分割されているために、部分解の整合を取るための通信が必要になる。

さらに問題分割に関する知識が全く与えられない場合について考えてみよう。これを分散制約充足法で解決しようとする、四つのエージェントがそれぞれ部分解を導出し、それを統合して解を見つけようとする。この場合、W から Z のエージェントにはそれぞれ $p(A, P1), p(P2, P3), p(P4, O), p(P5, P6)$ が与えられたのと等価になる。最悪の場合、それぞれのエージェントから、2, 1, 2, 4 個の部分解が導出されることになるが、そのうち充足解を構成するものは 3 つしかない。ここで特に注目すべきことは、C-F が通行不可能なことから、本来問題解決に関与する必要のないエージェント Z も関与しているという点である。

次に波及型探索の場合を説明しよう。この場合探索は初期地点 A を持つエージェント W から探索が開始される。エージェントは二つの部分解 $a(A,B,D)$ と $a(A,C,E)$ を導出し、新たな部分問題 $p(D,O), p(E,O)$

をエージェント X に割り当てることになる。エージェント X では最初の問題解決は失敗で終了するが、2番目のものからは部分解 $a(E,H,J)$ が導出され、新たな部分問題 $p(J,O)$ がエージェント Y に割り当てられる。そしてエージェント Y は解 $a(J,N,O)$ を導出して、問題解決は成功する。このようにこの手法は並列性は高くないが、エージェント自らの判断で問題の分割、割り当てが行われる。また問題分割知識のない分散制約充足法では問題解決に加わったエージェント Z がこの例では加わることがない。最後にこの例のまとめとして、各エージェントで導出された部分解の数を表 1 に示す。

さて、これまで問題分割に関する知識は与えられたものとして議論してきたが、未分割の問題に対して、問題分割に関する知識をどのように学習してゆけばよいのかは分散問題解決でも未解決の重要な課題である。さらに、4.2でも述べたように問題分割は問題解決の階層化の問題、すなわちエージェント組織の問題とも関連しており、この問題に対してもさらに研究を進める必要がある。

5 動的問題へのアプローチ

これまでに述べてきた分散問題解決の手法は、与えられた大局問題の構造が問題解決の過程で変化しないという静的問題を暗黙の前提としてきたが、分散環境においては問題の構造自体が変化する動的問題を扱わなければならない場合が生じてくる。例えば、通信ネットワークにおける経路選択問題は一種の分散探索問題ととらえることができるが [9]、ここではネットワークの混み具合によってリンクのコストが変化し、適切なメッセージ転送経路（解経路）は動的に変化する。このような問題に対しては分散動的計画法 [2] による手法を利用することができる。すなわちネットワークを分散して管理するエージェントが互いにネットワークの混雑情報を交換しあい、協調的に経路選択を行ってゆく。ただし、ここでの問題点は、混雑の微妙な変化に対して、逐次その変化を知らせることは通信オーバーヘッドを招き、逆に問題解決の性能が低下するという点である。したがってある程度、解の精度を犠牲にしても、エージェント間での情報交換を切り捨ててしまうことが必要な場合もあり、ここでも問題解決と通信オーバーヘッドの間にはトレードオフの関係があるといえる。

また、石田らは状態空間グラフにおいて目標状態が移動する移動目標探索 [5] を提案しているが、これも問題解決の過程で問題の構造が変化する動的問題の一つといえる。

6 まとめ

本稿では分散探索の視点からこれまでに提案された分散問題解決手法の紹介と問題分割の視点からの比較を示した。本稿で述べた手法の位置づけを示すと図 4 のようになる。以上の手法はいずれにおいても通信の制約によるトレードオフの問題に直面し、状況に応じた有効なヒューリスティックを用いながら、総合的に問題解決を行う必要がある。また、分散問題解決におけるエージェント組織の問題、問題分割などの学習の問題など残されている課題は多いといえる。

謝辞

本解説に関して議論して頂いた MARK (Multi-Agent Research community in Kansai) 会員の石田亨、小野典彦、桑原和宏、榎木哲夫、西部喜康、松原繁夫、山田誠二、横尾真の各氏に感謝します。

参考文献

- [1] R.B. Banerji: 人工知能, 共立出版 (1983)

- [2] D.P. Bertsekas: Distributed dynamic programming, *IEEE Trans. Autom. Control*, **AC-27-3**, 610/616 (1982)
- [3] L. Gasser: Conceptual modeling in distributed artificial intelligence, *人工知能学会誌*, **5-4**, 401/410 (1990)
- [4] 平山, 山田, 豊田: 山登り法を用いた分散制約充足における組織化, *人工知能学会全国大会論文集*, 269/272 (1993)
- [5] T. Ishida and R.E. Korf: Moving target search, *Proc. IJCAI*, 204/210 (1991)
- [6] 石田, 桑原: 分散人工知能 (1): 協調問題解決, *人工知能学会誌*, **7-6**, 945/954 (1992)
- [7] Y. Kitamura and T. Okumoto: Diffusing inference: An inference method for distributed problem solving, *Cooperating Knowledge Based Systems 1990*, 79/94, Springer-Verlag (1991)
- [8] 北村, 寺西, 辰巳, 奥本: 分散探索における通信制御, *情処学人工知能研資*, 89-5 (1993)
- [9] 北村, 鄭, 辰巳, 奥本: 動的問題のための協調探索手法, *マルチエージェントと協調計算 II*, 近代科学社 (1993)
- [10] 桑原, 石田: 分散人工知能 (2): 交渉と均衡化, *人工知能学会誌*, **8-1**, 17/25 (1993)
- [11] V.R. Lesser: An overview of DAI: viewing distributed AI as distributed search, *人工知能学会誌*, **5-4**, 392/400 (1990)
- [12] T.W. Malone, R.E. Fikes, K.R. Grant, and M.T. Howard: Enterprise: A market-like task scheduler for distributed computing environments, *The Ecology of Computation*, 177/205, Elsevier Science Publishers B.V. (1988)
- [13] 野下, 高岡, 町田: 基本的算法, 岩波書店 (1985)
- [14] M. Papazoglou and T. Sellis: Preface, *Int. J. Intell. Cooperative Inf. Syst.*, **1-1**, 1/4 (1992)
- [15] R.G. Smith: The contract net protocol: High-level communication and control in a distributed problem solver, *IEEE Trans. Comput.*, **C-29-12**, 1104/1113 (1980)
- [16] J.A. Stankovic, K. Ramamritham, and S. Cheng: Evaluation of a flexible task scheduling algorithm for distributed hard real-time systems, *IEEE Trans. Comput.*, **C-34-12**, 1130/1143 (1985)
- [17] 横尾, ダーフィ, 石田, 桑原: 分散充足制約による分散協調問題解決の定式化とその解法, *信学論 D-I*, **J75-D-I-8**, 704/713 (1992)

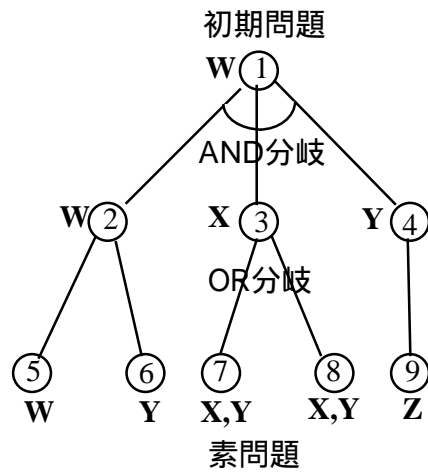


図 1: AND/OR グラフ

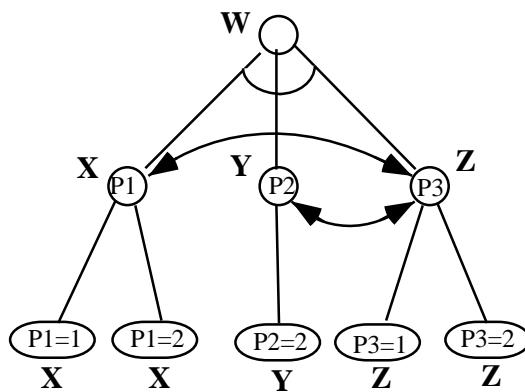


図 2: 大局的制約

表 1: 導出された部分解数による比較

分割手法	部分解数
完全問題分割	3
不完全問題分割	5
問題未分割 (分散制約充足)	9
問題未分割 (波及型探索)	4

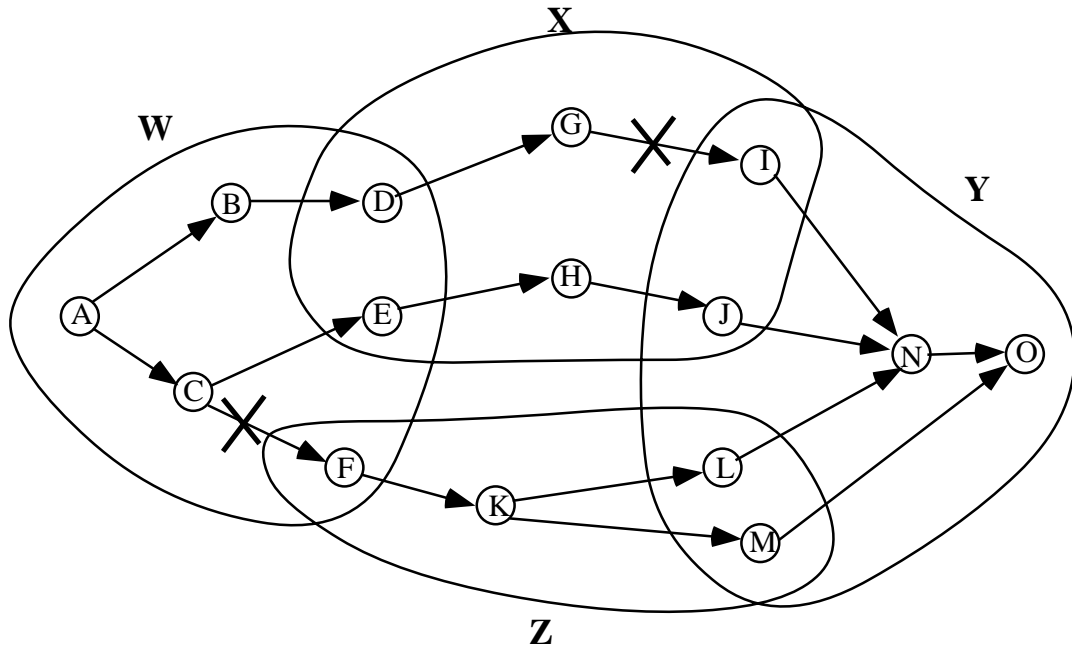


図 3: 例題

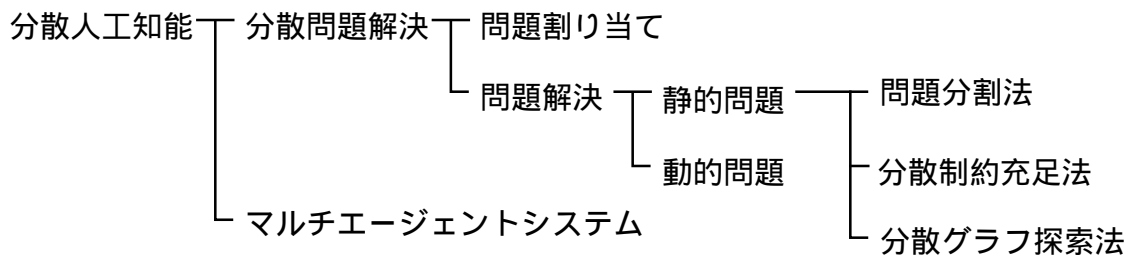


図 4: 各手法の位置づけ