NSGA-Ⅲアルゴリズムを用いたWeb三層モデルの最適化に関する研究

齋藤 篤志（北海道大学大学院情報科学研究科）

# 1．概要

ネットワーク、サーバー、ストレージ、アプリケーション、サービスといったコンピュータリソースを仮想化し、ユーザーが利用したいときに利用したい分だけ利用することができるクラウドコンピューティングは、その利便性の高さから、個人から企業まで広く利用されるようになった。クラウドサービスを選択する幅が広がった一方で、ユーザーは自身の要求要件に最適なクラウドサービスを見つけ出す必要性に迫られるようになった。この問題の解決策として、ユーザーの要求に対して最適なプランを提示するクラウドブローカーサービスが求められている。本論文ではクラウドブローカーサービスに適した数理モデルを提案し、NSGA-3アルゴリズムを用いて実用的なクラウドブローカーサービスの実現を行う。

**2．関連研究**

クラウドサービス選択に関する研究ではsundarswaranらの研究がある。クラウドブローカーサービスの基本構造を示した上で、無数にあるクラウドサービスを符号化して管理する手法を述べている。この論文の主眼は効率的なインデックスと探索を行なうこと、この論文以前のようにスペックの一部でしか最適化されていない状況から複数のスペックの最適化をすることであった。確かにCSP　　indexを用いてクラウドサービスをバイナリ符号化し、CSS-queryalgorithmでハミング距離を用いて探索をすることは高速であろう。しかし、ハミング距離が近いからといって、それが最適解である保証がないことが問題である。

gragらの研究ではCSMICが提案するService Mesurement Index (SMI)というクラウドサービス評価基準を用い、それぞれが階層的に示された7つの性質に基づいて評価を行なう。それぞれのサービスをSMIで評価した上で、階層分析法(AHP)を用いてインスタンスの選択を行なっている。AHPは階層構造を探索するために優れているため、SMIとともに用いる必然性がある。しかし、AHPによる選択は荷重和法の拡張と言えるもので、結局のところ、パレート最適解の効率的な探索ができていない。

次に、アーキテクチャの最適化に関する研究である川勝らの研究では、世界的な計算機資源に焦点を当て、三層アーキテクチャの最適化を目指しており、最適化アルゴリズムとしてNSGA-Ⅱアルゴリズムを用いている。この論文の主眼は、データセンター内の計算機資源を最適化することであり、当然ながらクラウドブローカーサービスにおいて主眼となるユーザー目線での選択基準で目的関数は設定されていない。また、NSGA-Ⅱアルゴリズムを用いており、実験にかかった時間には触れていないが、目的関数の数が増加した場合に混雑度の計算で相当な時間がかかることが予想される。本論文ではNSGA-Ⅲアルゴリズムを用いて、目的関数の数の増加した場合にも実用可能な時間で処理できる点で優位性がある。

pawlukらの研究ではクラウドブローカーサービスのアーキテクチャや構築方法について、ほとんどの紙幅が割かれているが、クラウドブローカーサービスを実装し、最適なアーキテクチャを。ただし、実験では荷重和法が用いられており、効率的な最適解の探索ができているとは言いがたい。また、目的関数も一つから二つしか用いられておらず、ユーザーの本当に求めるサービスが提案できているかが疑問である。しかし、アーキテクチャを最適化し提示している点で上記二つの研究に比べて実用的であるといえる。

**3．数理モデル**

クラウドブローカーシステムが解くべき問題を制約条件つき多目的最適化問題として定義する。実用的なクラウドブローカーシステムにおいては、ユーザーからの要求や、クラウドを提供する企業によって複数の制約がつくことが普通であり、最適化をする際には考慮せざるを得ない。例えばユーザーからの要求では、クラウドを設置する地域に関して、自国外に置くことに抵抗があることもあるだろう。このような場合、仮に最適解が国外に設置することであっても、ユーザーに国外に設置するプランを提示すべきではない。また、クラウドを選択する場合、クラウドを選択する基準がコストや仮想マシンのパフォーマンス、耐障害性など、様々であるため、それは多目的最適化問題になる。よってクラウドブローカーシステムの数理モデルは制約条件つき多目的最適化問題である。数式で表すならば、変数空間の次元をn、目的関数の個数をmとし、制約充足付き多目的最適化問題の解を 、 、可能領域を とすると、以下のように記述される。

また可能領域は一般に個の制約条件を満たす領域として以下のように定義できる。

**4．提案手法**

**4.1　NSGA-Ⅲアルゴリズム**

NSGA-Ⅲアルゴリズムは、NSGA-Ⅱアルゴリズムをさらに改善した遺伝的アルゴリズムの一種である。NSGA-Ⅱアルゴリズムから高速非優越ソートとエリート主義を引き継ぎ、NSGA-Ⅱアルゴリズムで計算のボトルネックになっていた混雑度の計算は、参照点を用いた選択に変更されている。

4.1.1　高速非優越ソート

高速非優越ソートでは各個体に対して、優越している個体と優越されている個体の数を同時に数える。優越されている個体がゼロの個体をランク1、ランク1の個体を取り除いて優越されている個体がゼロの個体をランク2とする。これを全個体がなくなるまで繰り返すことによって、高速なランク分けを実現している。

4.1.2　エリート主義

NSGA-Ⅱアルゴリズムでは個体の選択の際に、エリート主義を導入している。エリート主義とは常に優良個体を保存する親母集団と探索用の子母集団を使い、探索で発見した優れた解の消失を防ぐ手法である。

4.1.3　参照点による選択

参照点とはパレートフロント上に規則的に配置され、最も近い個体が選択される”点”のことである。参照点の数𝐻はおおよそ可能解の数と同等とし、目的関数の数を𝑀、任意の数を𝑝としてH = M+p−1Cpで表される。参照点の数がH = M+p−1Cpで表されるのは、パスカルの三角形に従って配置するためである。これにより、目的関数の数が増加したとしても、パレートフロント上で広範囲かつ均等に最適解群が得られ、かつ計算時間は短くて済む。

## 4.2　目的関数

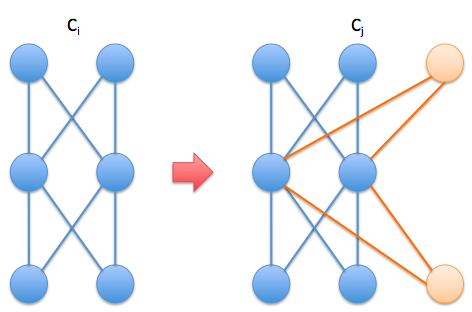
本論文ではNSGA-Ⅲアルゴリズムでの最適化に用いる４つの目的関数を提案する。なお、遺伝子が示す構成をciとする。

4.2.1　構成コスト

構成コストの目的関数fconfigration\_costでは、ある構成ciからある構成cjに、構成を変更した場合にかかるコストを求める。もし、初めてシステムを組み立てる場合は、ciは何もない構成であり、cjは新しく考えられた構成と考える。

Web サーバーの構成の変更にかかるコストをCweb 、アプリケーションサーバーの構成の変更にかかるコストをCapp 、データベースサーバーの構成の変更にかかるコストをCdb 、ロードバランサーの構成の変更にかかるコストをClbとすると、

で表すことが出来る。



（図１：構成変更の例。オレンジが変更部分。）

4.2.2　オペレーションコスト

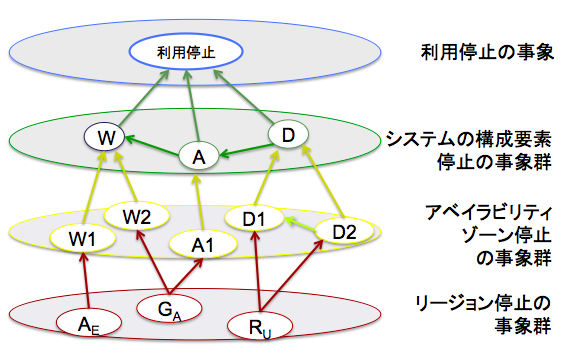
オペレーションコストの目的関数foperation\_cost で求めるのは、遺伝子が示すシステムを構成の場合、 一ヶ月あたりにかかる料金である。 Webサーバーのインスタンス料金の合計 をPweb 、アプリケーションサーバーのインスタンス料金の合計をPapp 、データベースサーバーのインスタンス料金の合計 をPdb 、ロードバランサーの料金の合計 をPlb 、VMの監視料金の合計をPwatchとすると以下のように記述される。

4.2.3　SLA

SLAの目的関数fSLAでは遺伝子が示すシステムの稼働率（一年間でサービスが利用可能な割合）を評価値とする。今回はベイジアンネットワークを用いて、利用停止に至る確率事象を計算し、一年間のうち何時間停止するかを計算する。

一般にベイジアンネットワークをとする。ただし、は確率構造、は条件確率付きパラメータである。n個の離散変数集合の場合、同時分布は一般に、

である。ベイジアンネットワークではを所与としているので、親ノード変数集合として、以下のように表せる。



（図２：ベイジアンネットワークによるシステムがダウンする事象の分析）

4.2.4　パフォーマンス

パフォーマンスの目的関数ではユーザーが求めるインスタンスのパフォーマンスに、遺伝子が合致する度合いを評価値とする。ユーザーリクエストは1秒あたりに処理されるリクエスト数をスループットとして、各層ごとに要求されることを想定した。それに対して各層のインスタンスのスループットは、コア数の合計を以下のように表したとき

Web サーバーのコア数の合計をCRweb 、アプリケーションサーバーのコア数の合計CRapp 、データベースサーバーのコア数の合計をCRdb 、VMのクロック数をCl、リクエストにかかるクロック数をPrとすると、以下のように表せる。

そして各層で求めたスループットの値とユーザーリクエストで要求された値を用いて、以下の目的関数で評価する。