

## 複数の最適化エンジンを利用する拠点間データ・アプリの最適配置案の算出方式の検討 A study of Data and Application Placement Optimization using multiple Optimization Engines

野村 鎮平<sup>†</sup> 早坂 光雄<sup>†</sup>  
Shimpei Nomura Mitsuo Hayasaka

### 1. はじめに

世界で発生するデータ量は年々増大している[1]。そのため、各企業ではこの大量データから価値を抽出するため、データ分析や人工知能を使ったデータ利活用を推進している。データの発生場所は複数拠点に散っており、データの利活用では、地理的に離れた拠点で生成・格納されるデータを収集して分析する必要がある。そのため分散拠点間でデータを共有しデータ利活用を促進するハイブリッドクラウドの市場が拡大している[2]。

ハイブリッドクラウドで、分散した拠点間でのデータ分析を効率的に実行するには、分析対象データや分析用アプリを、コスト・性能・容量・可用性・再エネ率(再生可能エネルギー利用率)・ガバナンスといった KPI (Key Performance Indicator) を考慮して、移動・配置する必要がある。背反する複数の KPI 間で所望のデータ・アプリの配置を得るには、試行錯誤を繰り返す必要がある。

本研究では、KPI に関する配置を導出するエンジンを複数組み合わせ、エンジン間で配置の一致する案を結合することで複数案をユーザに提示する方式を提案する。これにより、ユーザに背反する KPI のバランスを見せる複数の配置案の算出が可能となり、試行錯誤回数を削減する。

### 2. ハイブリッドクラウドでの最適配置

ハイブリッドクラウドでは、分散拠点間でデータやアプリの管理・運用を行う。ハイブリッドクラウドの利用者は、コスト・性能・容量・可用性・再エネ率・ガバナンスを考慮してデータ・アプリの配置や割り当りソース量を決定する必要があり、非常に煩雑となる。

例えば、アプリをデプロイするシチュエーションによって、短期間で処理を完了するための高性能重視するケースや、アプリ実行による費用を抑えるための低コスト化を重視するケースなど、重視する KPI が異なるケースにそれぞれ対応した配置の決定が必要である。また、要件となる性能や可用性の水準を維持しつつ、定められた予算規模にコストを収めるなど、複数の KPI 間でのバランスを調整するための配置案の試行錯誤が必要である。こうした点に配慮した配置の決定は煩雑であり、ユーザの工数増大を招く。

本研究の対象とする、拠点間のデータ・アプリ最適配置を図 1 に示す。データ・アプリ最適配置では、コスト・性能・容量・可用性・再エネ率・ガバナンスといった KPI を考慮して、データとアプリの最適配置・リソース割り当てを含む配置案を提案・実行する。

最適配置では、ユーザはデータ・アプリの配置を決定す

<sup>†</sup> 株式会社日立製作所 Hitachi, Ltd.

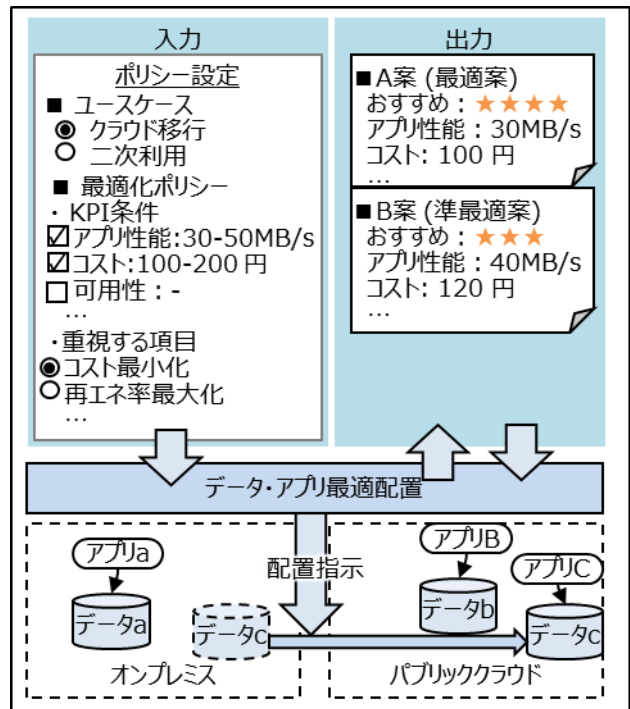


図 1 データ・アプリ最適配置

るための、最適化を適用するユースケースや KPI に関するポリシーの情報を入力する。最適配置は、入力された情報に基づき、データとアプリの複数の配置案をユーザに提案する。ユーザは、この提案された複数の配置案から好適な案を選択するか、他の案を得るために入力情報の変更を行う。ユーザが案を選択すると、最適配置はデータ・アプリの移動を行う。

最適配置にて、ユーザは状況に応じて、多様な最適化ポリシー(e.g., 性能重視, コスト重視, 可用性など)・ユースケース (e.g., クラウド移行, 二次利用など)を指定して、推奨拠点や割り当りソース量などの構成案を含む配置案を算出したい。またユーザは、背反する KPI (e.g. コストと性能, 電力と性能) のバランスを考慮しながら、最適案を選択する必要がある。

関連する従来技術[1]では、コスト最小化といった 1つの KPI にフォーカスして、1つの最適化案を算出・実行する研究がなされている。しかし従来技術は、コスト最小化の案のみを算出するもので、KPI やポリシーを選択できない。また、従来技術は 1つの案しか得られないため、ユーザは好適な案を得るまで入力値を変えて試行錯誤をする必要がある。

そこで本研究の最適配置では、ユーザのユースケース・最適化ポリシーの入力を受け付け、これにもとづき対応する最適化案を提示する。また、背反する KPI のバランスを見せるために複数の構成案を提示する。これにより、ユー

ザの構成案の比較検討を支援して、ユーザの試行錯誤の回数を低減する。

### 3. 課題

次に、データ・アプリ最適配置の実現に向けた2つの課題を説明する。

（課題1）多様な最適化ポリシー・ユースケースへの対応  
ユーザは状況に応じて、多様な最適化ポリシー・ユースケースを指定して、配置案を算出したい。しかし、1つの最適化エンジンへ全てのケースに対応する機能を作り込む必要があり、工数が大きい。例えば、1ユースケース/1最適化ポリシーのエンジンの研究開発に半年～1年といった期間を要する。

（課題2）複数KPIを考慮した最適化案の提示  
ユーザは、背反するKPIを考慮しながら、複数案を比較して配置案を選択したい。しかし、特定ユースケース/最適化ポリシーに特化した個々の最適化エンジンを用いると、複数のKPIのバランスをユーザに提示できない。

### 4. 提案方式

次に、課題解決のための2つの提案について図3に示す。

（提案1）複数の最適化エンジンの流用・自動選択  
提案1として、既にある複数の最適化エンジンを流用し、ユースケースやポリシーからエンジンを自動で選択する。これにより、1つの最適化エンジンに機能を作り込む工数は不要になる。また、最適化エンジンを追加時にも最適化エンジンの選択ロジックを使い回し、最適化ポリシー・ユースケースの拡張を容易化できる。

具体的には、予め各最適化エンジンの登録時にエンジンの開発者がサポートできる最適化ポリシー・ユースケースの情報を設定する。そして配置案の算出時には、設定した情報とユーザの選択する最適化ポリシー・ユースケースに基づき、適切な最適化エンジンを選択する。

（提案2）複数の最適化エンジンの算出案の結合  
提案2として、異なるKPIを考慮する複数の最適化エンジンで、個別に案を算出する。その中で配置先が同じものの情報を結合して、配置案とする。これにより背反するKPIの情報を提示できる。

具体的にはエンジンごとに算出した複数の配置案から、配置先の一致する案を見つける。そして一致した案のKPIの情報を結合する。加えて、配置先へデータ・アプリを移行するための時間とコストを、各拠点から収集した情報をもとに算出して、最終的な配置案とする。

2つの提案を適用した最適配置の算出手順を図4に示す。手順は、事前準備とアプリ配置時の大きく2つのフェーズに分かれる。

まず1つめのフェーズの事前準備では、最適配置のプラットフォームに、組合せて利用する最適化エンジンの登録と、各拠点の情報の収集を行う。

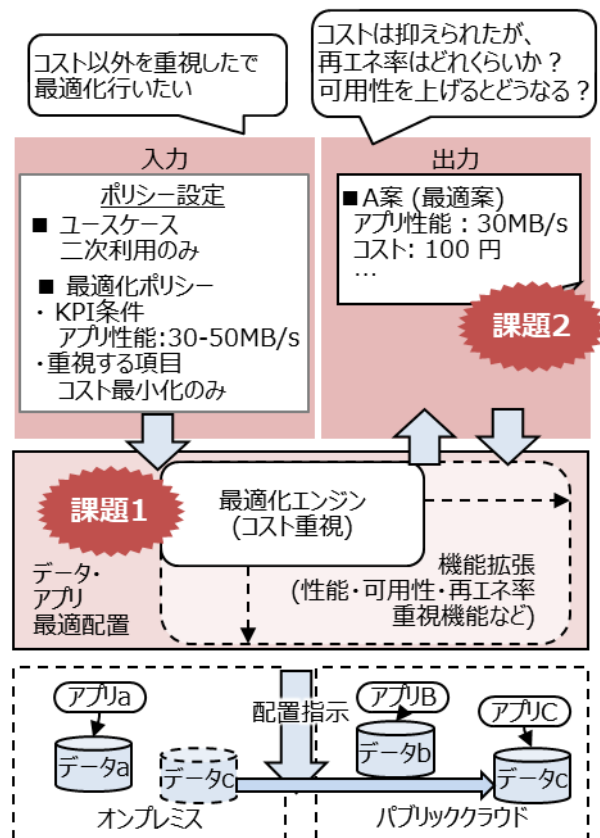


図2 課題

2つ目のフェーズのデータ・アプリ配置時には、ユーザから受領する最適化ポリシー・ユースケースにもとづき最適化エンジンを組合せて、背反するKPIの情報を含む配置案を提案する。

具体的な手順を以下に記載する。

（フェーズ1）事前準備

1-1. 最適化エンジンの開発者が、各エンジンの登録時に、サポートできる最適化ポリシー・ユースケースの情報を設定

1-2. 最適配置PFが、定期的に配置先の各拠点から、データ・アプリの配置情報やITインフラリソースの構成/稼働/課金情報を収集・蓄積する

（フェーズ2）アプリ配置時

2-1. ユーザが、最適化ポリシー・ユースケースを指定して、最適配置の算出をリクエスト

2-2. 最適化エンジン切替制御部が、ユーザが選択した最適化ポリシー・ユースケースに基づき、最適化エンジンを1つ以上選択（解決1）

2-3. 最適化エンジン切り替え制御部は、1-2で収集した情報を最適化エンジンへ入力する

2-4. 2-2で選択された最適化エンジンごとに、複数の配置案を算出

2-5. 最適化エンジン切替制御部が、エンジンごとの複数の配置案から配置先の一致する案を結合する（解決2）

2-6. 最適化エンジン切替制御部が、UIを介して、算出した案をユーザに提示する

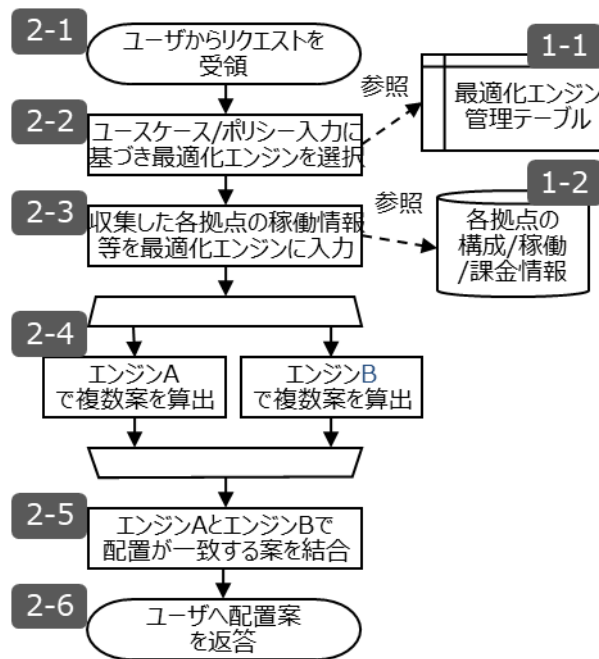
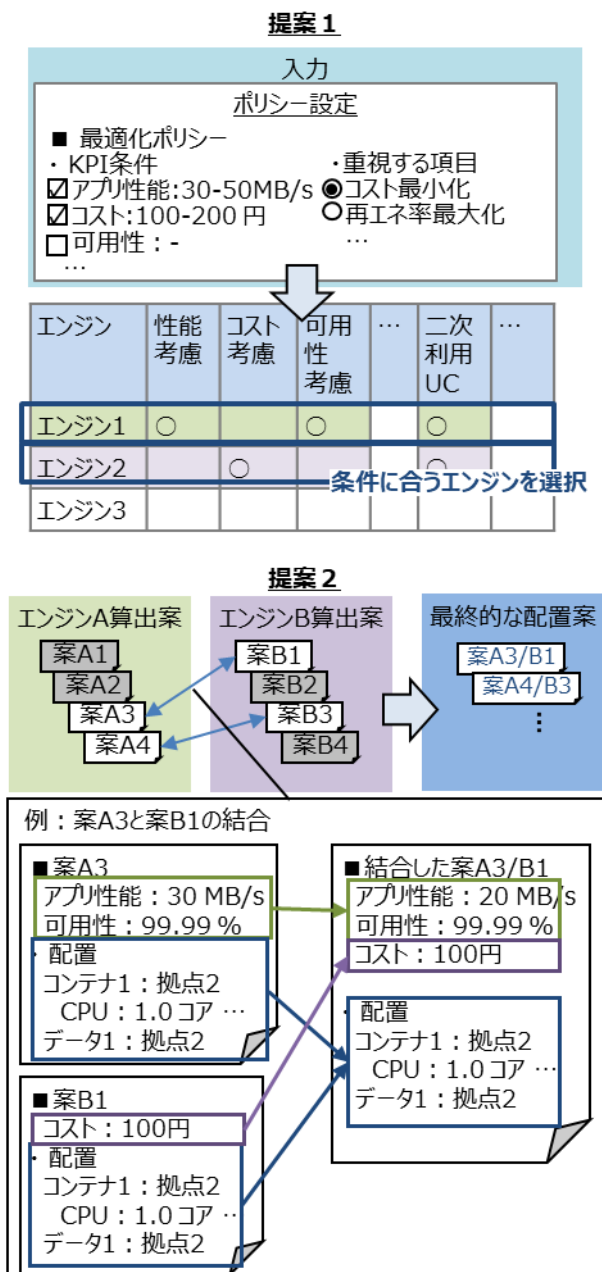


図 4 提案方式での処理フロー

慮するためには、ユーザが別途見積もりを行う必要があり、案算出の試行錯誤とは工数が発生する

従来方式での2つ目の問題点は、配置案を1案ずつ算出する点である。ユーザが複数の配置案を比較し最適な案を選択するためには、必要な数だけの配置案算出のための入力作業を繰り返す必要がある。

一方で提案方式では、複数の最適化エンジンを組み合わせることで、複数の最適化ポリシー・ユースケースをサポートし、一度に複数の配置案をユーザへと提示する。

従来方式の1つ目の問題に対して、提案方式ではエンジンに組み合わせにより考慮できる KPI を拡張することで対処する。ユーザは複数の KPI の指定が可能となり、最適な配置案を得やすくなるため、試行錯誤回数を削減できる。

次に従来方式の2つ目の問題に対して、提案手法ではユーザの一度の配置案算出の入力に対して、最適化案と妥協推奨案を含む複数案を提示する。これにより、ユーザが比較する複数の配置案を得るのに必要な試行錯誤回数を削減する。

以上の2点の改善による、提案方式による試行錯誤の回数の削減効果について表 1 に示す。仮定として、入力可能な KPI 項目について、従来方式ではコストのみ、提案方式ではコストに加え性能と再エネ率をサポートすることを想定する。また、一度の入力につき算出可能な配置案について、従来方式では1案、提案方式では5案を仮定する。

1つ目の特徴である、入力可能な KPI の拡張による試行錯誤回数の削減効果について、定量的な効果の算出は難しいため、1.0倍以上の効率化とする。2つ目の特徴である

複数案の提示による試行錯誤回数の削減効果は、一度の入力で5案を算出するため5.0倍の効率化とする。よって、2つの特徴により5.0倍以上の効率化が見込まれる。

## 5. 評価

### 5.1 試行錯誤回数の削減効果

まず最適配置への提案方式の適用による、試行錯誤回数の削減効果について、評価する。

従来方式として、提案手法適用前の特定の最適化ポリシー・ユースケースに特化した個々の最適化エンジンにより、配置案を1案ずつ算出するケースについて述べる。

従来方式での1つ目の問題は、考慮できる KPI が単一である点である。このため、ユーザは入力において単一の KPI しか指定できず、この入力値の調整により好適な案を得ることとなり、試行錯誤回数の増大を招く。また、ユーザは最適化エンジンのサポートしない KPI について、考慮できない。この最適化エンジンでサポートしない KPI を考

表1 試行錯誤回数の削減効果

#	項目	試行錯誤削減効果
1	KPIの拡張	> 1.0
2	複数案の提示	5.0
	合計	> 5.0

## 5.2 サポートする KPI・ユースケースの拡張

最適配置への提案方式適用によるサポートする KPI・ユースケース拡張への効果を評価した。

組み合わせる最適化エンジンとして、クラウド移行・バックアップ向けに性能・可用性観点で最適化するエンジン(エンジン A)、クラウド移行・データ二次利用向けにコスト観点で最適化するエンジン(エンジン B)、データ二次利用向けに再生可能エネルギーの利用率の観点での最適化するエンジン(エンジン C)を想定する。

表2は提案方式を適用した最適配置機能とそれぞれのエンジンのみを使った最適配置の比較を行ったものである。

まず提案方式による効果について、設定可能な KPI は、エンジン A でコスト観点の最適化、エンジン B で性能・可用性観点、エンジン C で再エネ率観点の最適化を、それぞれにサポート可能である。これに対し、提案手法でエンジン A~C により組み合わせることで、最適配置にてコスト・性能・可用性・再エネ率観点での最適化をサポート可能とする。

また、サポートするユースケースについては、エンジン A, B, C それぞれでは、最大2つでもユースケースのみをサポートするのに対し、提案手法では3つのユースケースをサポートできる。また提案方式により、未サポートである DR (Disaster Recovery) のユースケースについてもエンジンの追加によりサポートの拡張が可能である。

加えて提案方式により、ユーザへの最適化案・妥協推奨案を含む複数案の提示が可能となるため、コスト・性能・可用性・再エネ率の KPI 間のバランスをユーザに提示できる。

以上のように、本手法の適用によって、最適配置でサポートする KPI・ユースケースを拡大できる。また未サポートである KPI・ユースケースについても、最適化エンジンの追加により今後の拡張可能である。加えて、複数の KPI 間を考慮する最適化案・妥協推奨案を提示が可能となる。

## 6. 関連研究

参考文献[3]では、オンプレミスのリソース枯渇時にパブリッククラウドにアプリを逃してユースケースに対し、コストが最小化になるような、移行先のパブリッククラウドと移動対象アプリを、算出して提案し実行する研究がある。コストという1つの KPI にフォーカスしており、性能といった他の KPI をしない。また単一の案を算出するもので、複数案の算出といった点は考慮されない。

表2 ユースケース/最適化ポリシーの拡張

#	項目	提案方式	エンジン A	エンジン B	エンジン C
1	KPI	コスト	○	○	×
2		性能	○	×	○
3		可用性	○	×	○
4		再エネ率	×	×	×
5	ユースケース	クラウド移行	○	○	○
6		バックアップ	○	○	×
7		DR	×	×	×
8		データ二次利用	○	×	○
19	複数案の算出	○	×	×	×

## 7. おわりに

本研究では、ユーザの試行錯誤回数の削減に向けて、相反する複数の KPI の間で最適案・妥協推奨案を含む複数案の導出技術の確立を目的として検討を行った。複数 KPI をサポートする機能を1つの最適化エンジンに作り込むと、工数が大きくなり問題となる。そこで本報告では、既存の複数の最適化エンジンを流用し、最適化ポリシーやユースケースからエンジンを自動で選択する方式を提案した。加えて、異なる KPI を考慮する複数の最適化エンジンで、個別に案を算出し、その中で配置先が同じ案の情報を結合する方式を提案した。本方式により、ユーザに相反する KPI のバランスを見せる複数の配置案の算出が可能となり、試行錯誤回数を削減できる。机上評価した結果、本方式により、試行錯誤回数の1/5への削減の達成見込みを得た。

### 参考文献

- [1] Reinsel, D., Gantz, J., Rydning, J., "Data age 2025: The evolution of data to life-Critical", IDC (2017)
- [2] Ritu J., Stewart B., Robert W., "IDC's Worldwide Data Services for Hybrid Cloud Forecast, 2018-2022", IDC (2018)
- [3] Guo, Tian, et al. "Cost aware cloud bursting for enterprise applications." ACM Transactions on Internet Technology (TOIT) 13.3 (2014): 10