

VMの使用時間にもとづく作業時間の予測と 物理マシンの停止による省電力化

山野 倅平¹ 平尾 真斗² 串田 高幸¹

概要: 東京工科大学のコンピュータサイエンス学部の研究室である Cloud and Distributed Systems Laboratory で運用されている 7 台の物理マシンは、所属学生が自由に VM を作成し、いつでも作業できるよう 24 時間稼働している。物理マシンはサービスの運用を継続するため常に稼働しつづけ、その電力消費は絶えず発生する。課題は物理マシン内の仮想マシンが使用されていなくても物理マシンが稼働し続け、余剰な電力消費が発生していることである。提案では、曜日ごとに作業を行っているユーザ数を集計し、作業人数とその前後の作業人数から、1 分ごとにユーザが作業を行う可能性を示す Risk 値を設定し、物理マシンの未使用時間を予測する。これをもちいて PM の起動と停止を行う時間を決定する。作成した物理マシンの起動停止スケジュールに従い物理マシンの電源停止を行うことで電力消費を削減する。評価実験では、物理マシンが使用されていない時間に停止した時間、削減した電力量、停止した時間にユーザが作業した時間の 3 点で評価する。実際に提案を適用し物理マシンの起動と停止を行い、提案適用前と比較し 2024 年 11 月 11 日から 2024 年 11 月 15 日の平日の消費電力を評価する。削減電力量では、提案適用前と比較し提案適用後の方が一週間の総消費電力量を約 53.70%削減した。評価期間中の各 PM における PM 未使用時間中の停止時間割合では、平均で未使用時間中の約 51.08%の時間を停止した。停止時間 4086 分中にユーザが作業を行った時間は約 67 分で約 1.65%となった。

1. はじめに

背景

Physical Machine (以後:PM) とは、ハードウェアリソースを持つ実体として存在する計算装置である。PM は中央処理装置 (CPU)、メモリ、ストレージデバイス、ネットワークインターフェース、I/O デバイスから構成される。これらのコンポーネントは、ソフトウェアを実行し、計算タスクを処理するために相互に連携し稼働する。PM の運用には電力が必要である。電力消費はデータ処理、ネットワーク通信、冷却システムにより発生する [1,2]。特に、高性能のサーバは大量の電力を必要とする。これはサーバの性能を最大限に引き出すとともに、複数のタスクを同時に処理することを目的としている。また、サーバは 24 時間 365 日稼働しており、電力を消費し続ける [3]。これは、サーバのダウンタイムを防ぎ、常にサービスを提供し続けるためである [4]。PM の運用事例としては、データセンタ、ウェブホスティング、データベースサーバ、仮想化プラッ

トフォーム、ハイパフォーマンスコンピューティング、エンタープライズアプリケーションの運用、バックアップとリカバリの用途で利用されている [5-7]。国際エネルギー機関の報告では、2020 年時点で世界のデータセンタは約 200TWh の電力を消費している。これは、世界全体の電力消費の約 1%に相当する [8]。データセンタの数と規模は増加しており、それに伴い電力消費も年々増加している [9-11]。

Virtual Machine (以後:VM) とは、物理的なコンピュータシステムをソフトウェア的に模倣し、独立した複数の仮想環境を一つの物理ハードウェア上で実行可能にする技術である [12]。VM はハードウェアの抽象化を提供し、異なるオペレーティングシステムやアプリケーションが互いに干渉せずに動作できる。

また、仮想化技術の進展により、PM 上で VM を運用することが一般的になっている。VM は、ハードウェアのリソースを効率的に利用し、柔軟なシステム構成を可能にする。常時稼働する仮想マシンが存在する場合、PM を常に起動しておく必要があり、作業が行われていない時間にも電力を消費する [13]。

東京工科大学のコンピュータサイエンス学部の研究室である Cloud and Distributed Systems Laboratory (以後:

¹ 東京工科大学大学コンピュータサイエンス学部先進情報専攻
〒 192-0982 東京都八王子市片倉町 1404-1

² 東京工科大学大学院バイオ・情報メディア研究科コンピュータサイエンス専攻 クラウド・分散システム研究室
〒 192-0982 東京都八王子市片倉町 1404-1

CDSL) では、10 台の PM が運用されている。CDSL 内の PM 内では、米ブロードコム社の VMware 製品である ESXi が運用されている。2 台は基幹サーバとして、1 台は CDSL から外部への公開用サイトの運用目的で使用され、他の 7 台は CDSL の所属学生が自由に VM を作成し、作業を行う。それ以外の時間は、研究室外から PM 内に作成された VM に SSH でアクセスし作業を行う。研究室の学生がいつでも VM 上で作業できるよう、PM は 24 時間稼働している。

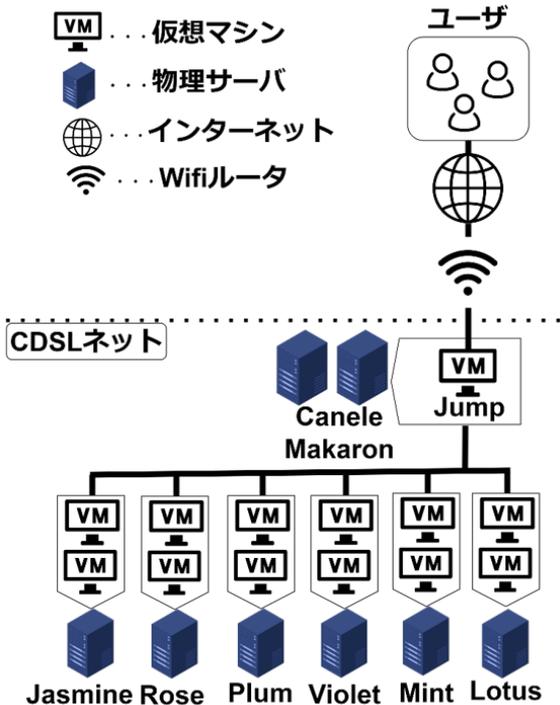


図 1: CDSL ネットワーク図

図 1 は CDSL のネットワーク図を示す。所属する学生が、研究室外から研究室内の PM 内部の VM にアクセスする際、インターネット上の大学の VPN を経由して研究室のネットワークに接続する。その際、Jump サーバを経由して目的の VM へアクセスし作業を行う。仮想マシンは研究室内で学生が作成する VM を指す。PM は合計で 10 台あり、Jasmine、Rose、Plum、Violet、Mint、Lotus の 7 台は学生が ESXi 上に自由に VM を作成し作業を行うことができる。Canele、Macaron は Jump サーバを運用している基幹サーバである。残りの 1 台は外部公開用サイトを運用する目的で使用される。

課題

課題は、PM 内の全ての VM で作業が行われていない時に、PM がアイドル状態として稼働し続けることにより余剰電力消費が発生することである。図 2 はアイドル状態における PM の電力消費を示す。CDSL では PM が常時

稼働しておりユーザが自由に VM で作業を行うことができる。PM 上で動作する VM が実際に使用されていない場合でも PM が稼働し、電力を消費し続けている。

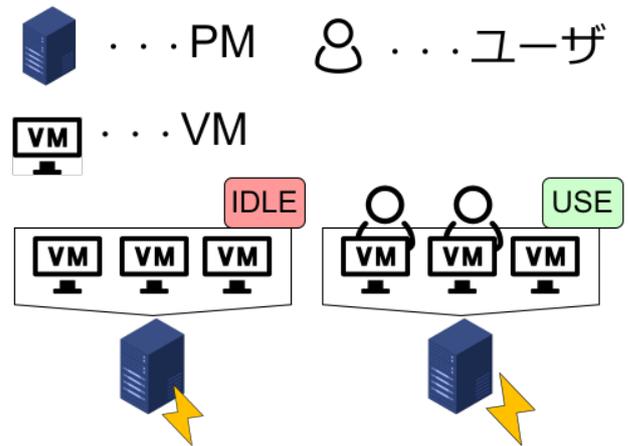


図 2: アイドル時間における電力消費

各章の概要

第 2 章では関連研究について説明する。第 3 章では、課題に対する提案方式について説明する。第 4 章では、実装したソフトウェアについて説明する。第 5 章では、評価実験について説明する。第 6 章では、議論を行う。第 7 章では、本論文におけるまとめである。

2. 関連研究

VM マイグレーションをもちいて、効率よく VM を PM 上に配置し、PM 運用時の電力消費削減を試みた先行研究がある。この研究では、アントコロニー最適化アルゴリズムをもちいて、電力消費とユーザエクスペリエンスの両立を行うアルゴリズム Power-aware and Performance-guaranteed VMP を提案している。この手法では動的な PM の電力消費の削減が課題となっている [14]。

PM がアイドル状態だった場合、PM をスリープ状態へ遷移させることで、PM の電力消費の削減を試みた先行研究がある。この研究では、PM をスリープ状態とアイドル状態の切り替えを行う。アイドル状態ではリクエストに対して高速でレスポンスすることが可能であり、スリープ状態ではリクエストに応じる前に状態遷移を行う必要があるが、より少ない電力で動作する。これらのトレードオフを活用し、ユーザ体験を損なわずに電力消費の削減を試みている。この手法は、PM をスリープ状態へ遷移させるため、PM の停止と比較して電力消費が多くなる [15]。

データセンタの典型的な運用では、サーバの利用率は 30 % 未満であることに着目し、サーバの電力消費削減を試みた先行研究がある。この研究では、システム全体が高性能なアクティブ状態と、瞬間的な負荷に応じてほぼ電

力消費のないアイドル状態の間を迅速な移行を利用してサーバの電力消費削減を試みている。提案ツールである PowerNap はアイドル状態時の電力と移行時間を最小化することに焦点を当てている。また、PowerNap にもとづき、エンタープライズブレードサーバにおけるアイドル電力の浪費を排除するために、新しいメカニズムと要件を作成している。この手法ではアイドル状態への移行のため、PM 自体の電力消費は発生している [16]。

消費電力推定モデルの課題に着目し、サーバの消費電力削減のためモデリング機能及びデータ選択機能を有する管理システムを提案することで試みた先行研究がある [17]。この研究では、PM の消費電力の推定では、推定精度とモデル構築とデータ収集のコストを削減を両立することが困難なため、モデリング機能及びデータ選択機能を有する管理システムを提案している。この論文では、データ収集時のサーバ消費電力を削減するため、監視機能の課題に対処し、「サーバパフォーマンス」、「データ量とサーバの消費電力の関係」を分析し、扱うデータの種類削減を行ったことで、選択したデータを用いた電力のモデルが構築可能であることを示した。加えて収集するデータ量を制限することで、サーバの消費電力を約 3W、監視のオーバーヘッドを約 30% 削減した。この手法では、PM の消費電力削減の手法に改善の余地がある。

3. 提案方式

本提案では、ユーザが作業を行っていない時間の予測スケジュールの作成による PM の停止を用いた電力消費を削減する手法を提案する。

PM 内の VM に誰もアクセスしていない状態を、VM の使用状況をもとに、1 分ごとにユーザが作業する可能性を示す Risk 値を設定し、Risk 値をもとに PM が使用されていない時間を予測する。各 PM 内に存在する全ての VM の過去 9 週間分のユーザのアクセスログを集積し、各曜日で 1 分ごとに VM 内で作業していたユーザの数をもとに、時間ごとのユーザが SSH でアクセスし作業を行う可能性を数値化する。この値を Risk 値として設定し、ユーザがアクセスしない時間を予測するスケジュールを作成する。このスケジュールをもとに PM の起動と停止を行う。

VM の使用状況を取得する時間が 1 分間隔の理由は、本提案で使用する last コマンドで時間を取得する際の最小単位が分であり、可能な限り細かく処理を行うことができるよう 1 分とした。ログの集積期間を 9 週間とした理由は、習慣形成にかかる期間の平均が 66 日であり、その平均に最も近い週数が 9 週間であるためとした [18]。曜日ごとに分割する理由として、各週の曜日ごとのスケジュールが存在するからである。この影響からユーザの VM 使用時間に変化があるため、データの分類は曜日ごとにした。

図 3 は提案の概要図である。以下の手順で PM の起動と

停止を行う。

- (1) 対象 PM の内部に存在する全ての VM から、ユーザが作業していた時間のログを last コマンドと who コマンドをもちいて取得する。その後、基幹 PM に設置された提案ツールを運用する VM へと送信する。
- (2) 集積した作業時間のログから、Risk 値を設定する。
- (3) 設定した Risk 値をもとに、使用されていない時間を予測し PM の停止スケジュールを作成する。その後、停止スケジュールをもちいて、起動と停止を行う。

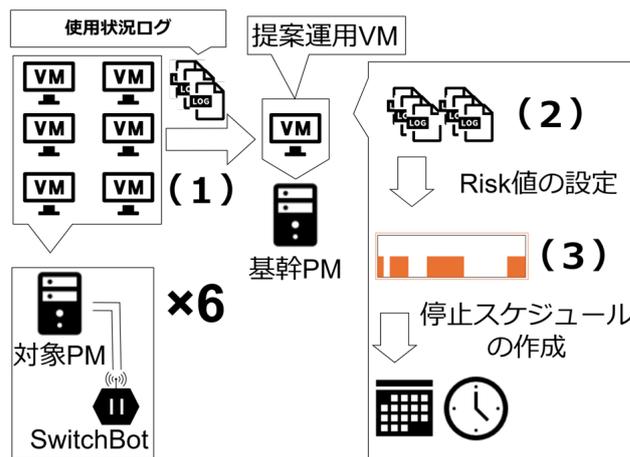


図 3: 提案の概要

VM 使用状況ログの集積

CDSL 内で稼働している 10 台の PM のうち、学生が自由に VM を作成し運用する用途で使用される、Jasmine, Rose, Plum, Violet, Mint, Lotus の 6 台を対象とする。対象 PM 内に存在する各 VM から、使用状況集積ツールをもちいて、10 分に一度 last コマンドと who -u を実行する。PM ごとに集積した結果を、提案ツールを運用する VM に送る。集積するログの内容は、last コマンドの出力結果と、who -u コマンドで取得したユーザごとのアイドル時間である。

PM ごとの VM 使用状況の整理

集積したログから、対象となる PM の VM 使用の状況を示すログを直近 9 週間分取得する。PM ごとの直近 9 週間の使用状況を示すログから 1 分ごとにログインしているユーザ数を合計する。その後、集計したログから、曜日、時、分が一致する時間帯ごとにユーザ数を合計する。

各時間ごとの Risk 値の設定

ユーザが SSH でアクセスし作業を行う可能性を示す Risk 値を設定する。Risk 値は 1 分ごとに全時間帯に設定される。Risk 値は 2 つの要素の合計で求める。1 つ目の要素は、同一曜日の各分ごとのユーザ数を集計し

た値である。2つ目の要素は、対象時間の前後のN分のユーザのアクセス数の変動から、対象時間に対するアクセスする可能性をRisk値の算出に利用する。これは、前後N分のVM利用ユーザ数の変動から、その時間帯に利用する可能性を表す。N分はユーザの作業時間のばらつきを求めるために過去のVMの使用状況をもとにした標準偏差を計算する。取得していたデータより、VMにアクセスしていたユーザは、同一曜日の同じ時間帯とその前後に再度作業を行う傾向があった。例えば、12:00にアクセスする場合、前後10分の11:50や12:10にもアクセスする可能性が高く、21:00に作業を終了した場合も同様に前後10分以内の20:50や21:10で終了する可能性が高い。

2つの要素を組み合わせて、各時間帯のRisk値を計算し、PMの停止時にユーザがPMを使用できなくなるリスクを算出する。そして、PMの停止条件としてRisk値が0であることを設定する。

Risk値を求める計算式

Risk値を求める手法を以下に示す。まず、現在の時刻を t とし、対象時間より前後それぞれN分間のデータを参照して変化量を計算する。その後、前後の時間の変化量を対象時間のユーザアクセス数の合計に加算することで、Risk値を決定する。

前後N分は、約6か月分の各PMの使用状況データを分析し、四分位数をもちいて異常値を除外し、作業時間の75%が1分からN分間に集中する時間範囲である、

以下にRiskを求める計算式を示す。対象時間より前のN分間の過去にあたるデータから、対象時間のユーザ数 $V(t)$ を基準とした際の、N分前までの平均変化量を求める。この変化量は対象時間に近い時間のユーザのアクセス数が対象時間に与える影響を表す。これを式に表したものを式(1)に示す。ここで、 $W_p(t)$ は、時刻 t における対象時間よりN分前までの平均変化量を示す。

$$W_p(t) = \frac{1}{N} \sum_{i=1}^N (V(t) - V(t-i)) \quad (1)$$

次に、対象時間よりN分後までの対象時間から未来にあたるデータに対して前述と同様の操作を行う。対象時間のユーザ数 $V(t)$ を基準とした際のN分後までの平均変化量を求める。これは、対象時間に近い時間のユーザのアクセス数が対象時間に与える影響を表す。これを式に表したものを式(2)に示す。ここで、 $W_f(t)$ は、時刻 t における対象時間よりN分後までの平均変化量を示す。

$$W_f(t) = \frac{1}{N} \sum_{i=1}^N (V(t+i) - V(t)) \quad (2)$$

過去と未来の変化量の平均値を算出し、対象時間に対して

その平均値を重みづけする。この重みは、対象時間から過去と未来のPM使用状況が対象時間の利用可能性に与える影響を考慮したものである。式(3)にこれらの値を計算する式を示す。 $W(t)$ は、時刻 t における最終的な重みを示す。

$$W(t) = \frac{W_p(t) + W_f(t)}{2} \quad (3)$$

最後に、この重み付けを対象時間のユーザ数と合算することで、Risk値の設定を行う。具体的には、対象時間のデータ $V(t)$ に式3で求めた重み $W(t)$ を加えたものがRisk値となる。これを式に表したものを式(4)に示す。

$$R(t) = V(t) + W(t) \quad (4)$$

重み $W(t)$ が0となる場合は、リスク値に対象時間の作業人数 $V(t)$ をそのまま使用する。Risk値はPMを停止した時間に、アクセスがある可能性を指し、この値が小さければPMを停止するリスクは小さい。

1日のデータを求める場合、Risk値を求める対象時間を0:00, 0:01, 0:02...23:58, 23:59の形式で変更していき、この処理を1日の分数である、1440分全てに対して行う。

Risk値を用いた停止スケジュールの作成

1日の時間を10分ごとに一つのブロックとする。これは、PMの電源オンオフは短時間で切り替えることにより、起動停止が完了していない状態での再停止、再起動を行わないための処理である。PMを起動してから停止するまでの推奨時間は、一般的にハードウェアメーカーのガイドラインに従って設定されており、Dell PowerEdgeサーバーの場合、頻繁な電源サイクルを避けるため、サーバーをオンにしてから少なくとも10分以上稼働させることが推奨されている*1。0:00 0:09, 0:10 0:19, ... 23:50 23:59の形式で一日1440分を144個の10分ごとのブロックとする。この際、10分ごとに設定されていたRisk値を合計する。0:00から0:09にそれぞれRisk値が1ずつ設定されていた場合、このブロックのRisk値は10となる。そして、Risk値が0のブロックはPMを停止できると判断し、PMの停止スケジュールに追加する。

停止スケジュールを用いたPMの起動停止

作成した停止スケジュールをもちいて、PMの起動と停止を行う。PMを起動する場合、PMが起動してからVM上で作業ができるようになるまでの時間を考慮し、スケジュールされた起動時間の5分前にPMの起動を行う。5分とした理由は、CDSL内のPMを停止状態から起動し、VM上で作業が可能になるまでの時間を計測した結果にも

*1 <https://infocenter.dell.com/en-us/p/idrac9-virtual-power-cycle-remotely-power-cycle-dell-emc-poweredge-servers/>

とづいている。計測の結果、どの PM も約 3 分で VM 上で作業が可能になったため、VM が増加した場合に対応するため 5 分と設定した。停止する場合、内部の VM を正規の手段でシャットダウンした後、PM をシャットダウンする。このスケジュールにより、PM が誰も使用していない期間を予測し、その期間に電源を停止することで、電力消費を削減できる。

ユースケース・シナリオ

CDSL における PM の運用環境を想定する。CDSL では 10 台の PM が運用されている。そのうち 7 台は CDSL の所属学生が VM を ESXi 上に作成し自由に作業を行うために運用されている。PM 上の VM は学生が作業するために常時起動している。

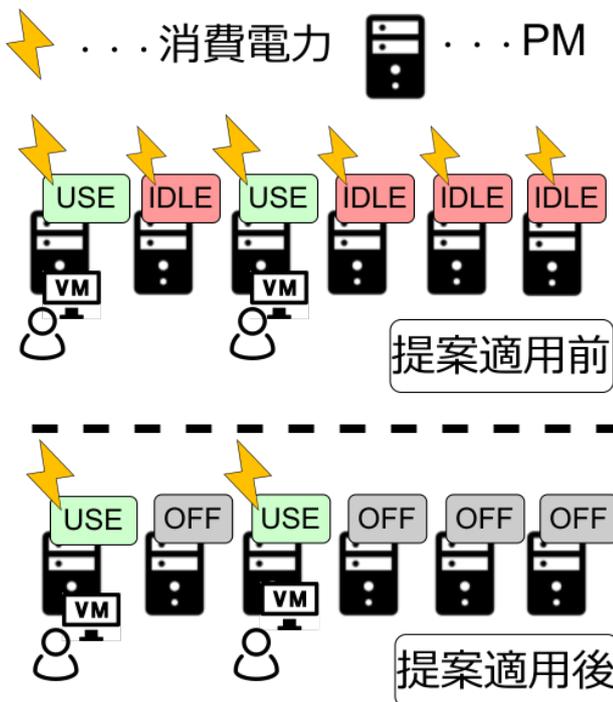


図 4: ユースケースシナリオ

図 4 は、ユースケースシナリオを示す。PM 上で運用されている VM は常時起動しており、いつ所属学生が作業をするか分からない。そのため、PM ごとにアイドル状態となっている時間が存在する。アイドル状態の PM では所属学生が作業しておらず、その時間帯における PM の電力消費は余剰となる。提案適用後、所属学生が作業していない時間を予測し、停止スケジュールを作成する。作成した停止スケジュールをもちいて誰も作業を行っていないアイドル状態時に PM の電源を停止し、余剰な電力消費を削減することができる。

4. 実装

実装の全体構成を図 5 に示す。提案手法をもとに、プロ

グラミング言語の Python とシェルスクリプトをもちいて本稿のソフトウェアを作成した。以下は Python で使用したライブラリである。

- datetime
- os
- shutil
- collections
- pandas
- pytz
- concurrent.futures
- time
- paramiko
- logging
- requests
- slack_bolt
- slack_sdk
- functools
- SwitchBotAPI

本提案で作成したソフトウェアは、以下の 3 つの要素で構成されている。1 つ目は、PM の使用状況を把握するため、各 PM 内の全ての VM に設置されたログ集積用コンポーネントである。2 つ目は、Risk 値の設定や電源の起動と停止の機能を持つコンポーネントであり、基幹サーバ内の VM に配置する。3 つ目は、物理的に電源を停止するため、物理マシンに接続された SwitchBot である。

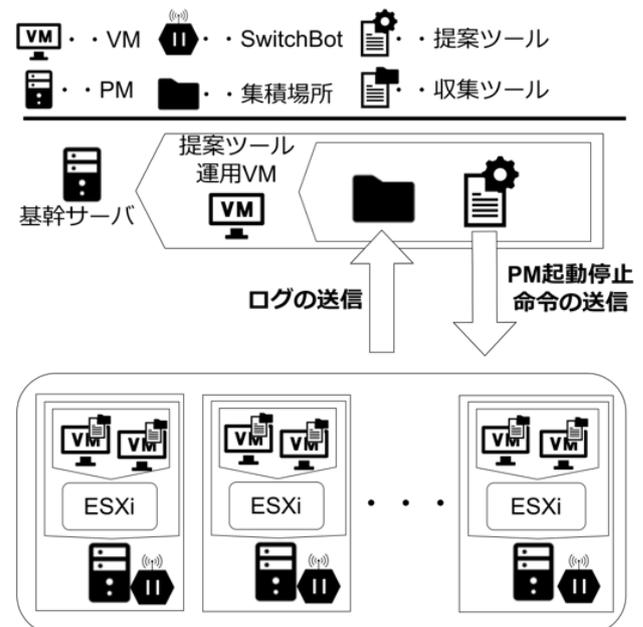


図 5: 実装全体図

ログの集積

提案の対象となる PM に存在する全ての VM にログ集積ツールを導入する。ログ集積ツールとして、提案ツール

を運用する VM ヘログを送信するシェルスクリプト SMDRP_log_collect.sh を作成した。各 VM 内で 10 分おきに last コマンドと who -u コマンドの出力結果を提案ツールを運用する VM へ送信する。この際、last コマンドでは、各 VM ごとのユーザのアクセス時間を取得し、who -u コマンドでは 10 分おきのユーザ単位のアイドルタイムを計測する。SMDRP_log_collect.sh は systemd の timer 機能を使用して 10 分おきに実行する。

PM 停止スケジュールの作成

集積されたログには、PM に存在する全 VM の時間ごとにアクセスしたユーザの人数が PM ごとに記録されている。本稿では、CDSL で運用される 6 台の PM を対象とした。このログを同一曜日、同一時間帯でユーザのアクセスを集計し、合計したデータに整形する。合計したデータをもとに提案アルゴリズムをもちいて Risk 値を算出し、Risk 値にもとづいた PM 停止スケジュールを作成する。

PM の起動及び停止

PM 停止スケジュールの作成の項で作成された、停止スケジュールを参照し PM の電源を起動と停止 Switchbot プラグミニとその API である SwitchbotAPI をもちいて行う。SwitchBot プラグミニは、スマートホーム自動化デバイス的一种であり、遠隔起動や自動起動に対応していないデバイスの遠隔起動や自動起動を可能とする。SwitchBot プラグミニは遠隔から WiFi を経由して電源の ON/OFF を行う。消費電力の計測を行う機能を保有し、専用 API を使用して電源の ON と OFF を管理できる。本実装では PM 全てのコンセントと電源の間に SwitchBot プラグミニを接続し、管理用 VM から遠隔で電源の ON/OFF を管理する。

停止する手法として、提案ツールから、ESXi に SSH し、VM の一覧と起動状態を取得し稼働している全ての VM にシャットダウン命令を行う。この処理が終了した後に ESXi をメンテナンスモードへ変更し、ESXi のシャットダウンコマンドを実行する。シャットダウンコマンドの実行が完了した場合、停止する対象の PM に接続された SwitchBotAPI を使用して、SwitchBot で電源を停止する。

起動する手法として、PM の電源を ON にした場合 ESXi は自動起動設定にする。そして、SwitchBotAPI を使用して電源をつける。これにより疑似的に PM のスイッチを ON にした状態を再現することで、PM が立ち上がると自動的に ESXi が立ち上がる。その後、ESXi に対して 1 分間隔でメンテナンスモードの解除命令を送信し続ける、この処理は ESXi が完全に起動しメンテナンスモードの解除命令が実行されるまで繰り返される。メンテナンスモードが解除された場合、ESXi 内の自動起動設定が行われている VM に対して起動命令を送信し VM の再起動を行う。

5. 評価実験

評価実験では、PM の未使用時間における PM の停止率、削減電力量、PM の停止時間におけるユーザの作業時間の 3 点について評価を行う。また本提案において、作業時間の前後のばらつきを求めめるため、実際に作業時間の標準偏差を計算した結果、40 分となった。そのため本評価実験でユーザの作業時間のばらつきを示す値である N を 40 分とした。図 6 は、異常値を除いたユーザの作業時間のデータセットにおけるユーザの作業時間の分布を示しており、作業時間の約 75%がこの 40 分間に集中していることを確認したため、前後の時間を 40 分間とした。

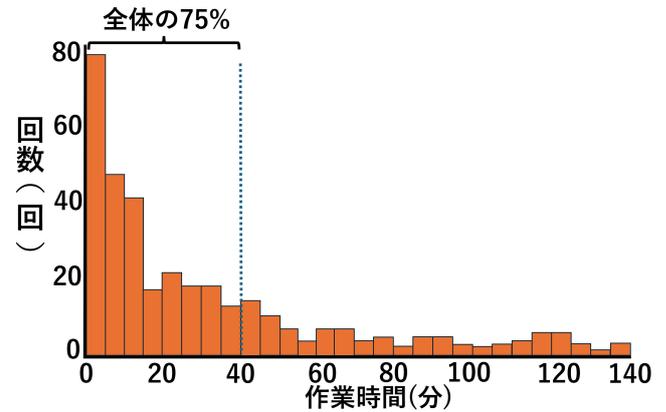


図 6: PM 利用ユーザの作業時間分布

PM 未使用時間における PM 停止割合では、PM が実際に未使用だった時間のうち、未使用時間中に実際に停止した時間の割合で評価を行う。削減電力量では、提案適用前の 1 日の平均電力消費量と、提案適用後の平均電力消費量を SwitchBot をもちいて計測し、Wh (ワット時) で削減割合を評価する。PM 停止時間におけるユーザの作業時間では、PM を停止している期間にユーザが作業のためにアクセスした回数とその作業時間と、停止時間の割合で評価を行う。

実験環境

実験は CDSL で運用されている 6 台の PM, Jasmine, Rose, Plum, Violet, Mint, Lotus の 6 台を対象とする。対象となる PM には、仮想化プラットフォームとして VMware ESXi が導入されている。また、対象 PM 内の全ての VM には、本提案で使用される VM 使用状況の集積ツールが導入されている。PM の起動と停止に関する提案ツールの主要なコンポーネントは、基幹サーバに設置された VM で運用されており、ここで全体の管理やデータ処理を一元的に行う構成となっている。

物理的な環境として、各 PM の電源装置に SwitchBot プラグミニが接続されており、各デバイスが PM の電源制御

と電力消費のモニタリングを行う。提案ツールでは、VM内でAPIを介することでPMの電源状態の制御を行う。

また、アルゴリズムで定めたPM停止時間中にユーザが作業をする場合にPMを起動するために、PMの電源起動用のSlackBotを作成した。このSlackBotを呼び出すことで、停止されていたPMが再稼働する。また、再稼働時の時間が記録される。

実験結果と分析

提案を適用しPMの起動と停止スケジュールをもとに、PMの起動停止を行った。図7は、対象PMの1台であるMintの2024年11月11日の起動停止スケジュールである。全てのPMは個別のスケジュールをもとに起動時間にPMを稼働させ、停止時間にはPMを停止させる。

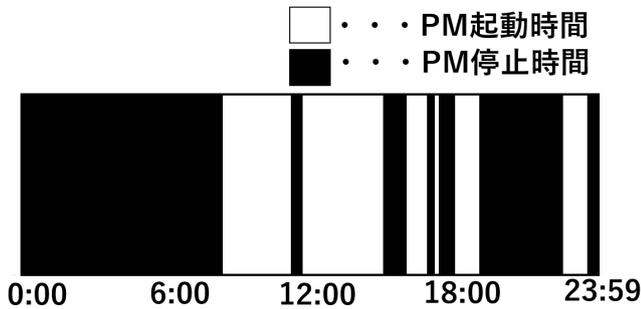


図7: 2024年11月11日 Mint 起動停止スケジュール

PMの未使用時間におけるPM停止割合について、評価を行った。実際に各PMが未使用だった総時間に対して、提案アルゴリズムを使用して作成したスケジュールを用いることで、停止できた時間の割合、停止できたかで評価を行う。各PMの提案を適用した2024年11月11日から2024年11月15日における平日5日間の未使用時間中の停止総時間が停止できた割合を求める。PMの未使用時間中におけるPMの停止割合についての評価実験の結果を図8に示す。MintのPMの未使用時間の平均は約1360分、PMの停止時間の平均は約900分となった。JasmineのPMの未使用時間の平均は約1356分、PMの停止時間の平均は約644分となった。LotusのPMの未使用時間の平均は約1340分、PMの停止時間の平均は約654分となった。RoseのPMの未使用時間の平均は約1112分、PMの停止時間の平均は約420分となった。VioletのPMの未使用時間の平均は約1178分、PMの停止時間の平均は約672分となった。PlumのPMの未使用時間の平均は約1436分、PMの停止時間の平均は約796分となった。評価期間中の各PMにおけるPMの未使用時間中の停止時間割合の平均として、約51.08%の時間を停止した。

停止時間の精度が十分に向上しなかった理由として、評価期間がレポート提出前に近く、レポート提出に関連した夜間作業や長時間作業を入力として使用されていたため

ある。

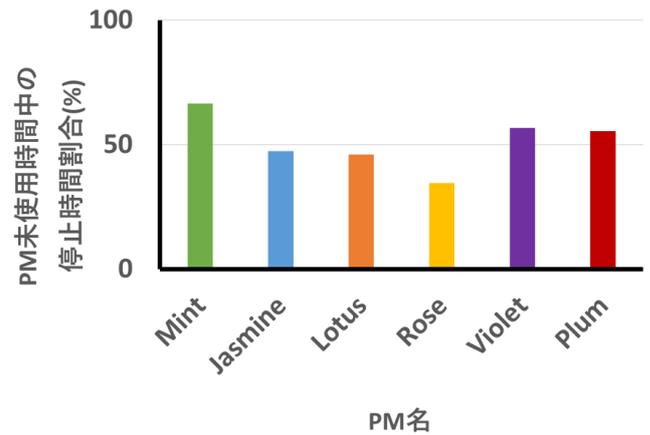


図8: 未使用時間中の停止時間割合

提案を適用した際の電力消費量の変化について評価を行った。各PMに提案を適用した2024年11月11日から2024年11月15日における平日5日間の総消費電力量と、提案適用前で評価実験の前週にあたる2024年11月4日から2024年11月8日の平日5日間の総消費電力量を比較する。

提案適用時の削減電力量を図9に示す。図9のX軸は評価実験の対象となるPM名を示し、Y軸は1週間の総消費電力量を示す。Mintでは提案適用前の週の総電力消費は12.06kWhであった。提案適用後には4.98kWhとなった。Lotusでは提案適用前の週の総電力消費は13.25kWhであった。提案適用後には6.03kWhとなった。Jasmineでは提案適用前の週の総電力消費は13.94kWhであった。提案適用後には7.72kWhとなった。Violetでは提案適用前の週の総電力消費は12.25kWhであった。提案適用後には5.38kWhとなった。Roseでは提案適用前の週の総電力消費は14.77kWhであった。提案適用後には7.4kWhとなった。Plumでは提案適用前の週の総電力消費は13.33kWhであった。提案適用後には5.51kWhとなった。提案を適用した週の総消費電力量は、提案適用前の週に比べ全体の平均で約53.70%の消費電力を削減した。

PMを停止時間した時間に対して、削減電力量が約2.62%多かった。これはPMの停止時にVMを全てシャットダウンし、次の起動時に再起動設定を行われたVMしか起動しないため、評価実験開始前と比較しPM内で起動していたVMの数が減り、各PMの1時間当たりの消費電力量が減少していたことが要因である。

PM停止時間におけるユーザの作業時間割合について、評価を行った。これは、スケジュールにおいてPMを停止する時間に対してユーザがアクセスをして作業を行ったスケジュールが外れた場合について評価を行った。各PMの提案を適用した2024年11月11日から2024年11月15日における平日5日間のPM停止時間におけるユーザの作業

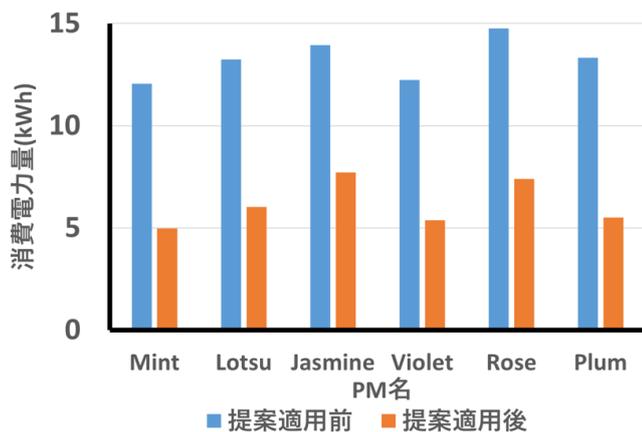


図 9: 削減電力量

時間を比較し評価を行った。評価期間中の総 PM 停止時間は約 4086 分であり、停止時間中に合計 6 回のアクセスがあり、その作業時間は合計約 67 分だった。停止時間中の約 1.65%の時間が作業されている。

6. 議論

本稿の提案において、PM を起動しているが、ユーザが作業していない時間が存在し、まだ電力削減の余地がある。入力に PM から取得できるマシンリソースのみしか使用していないため、Google の提供する健康管理アプリケーション Google Fit や、GPS を用いた位置情報を活用し、ユーザの生活習慣にもとづいた情報を取得し、入力として扱うことでユーザの作業時間をより詳細に特定し、作業が行われていない時間の特定精度が向上する。

本稿の提案では、VM 内に SSH 接続の有無のみを入力としている。例えば、SSH アクセスに作業を行っていない期間を特定できていない。VMware の公式ドキュメントにより、VMware Host Client のユーザーインターフェイスセッションタイムアウトのデフォルト時間は 15 分である。^{*2} 解決策として、who コマンドの-u オプションを使用しユーザが最後にキーボードで入力を行った時間を特定し、15 分以上入力がなかった場合、作業が中断したとみなすことで停止精度を向上できる。

本稿の提案では、ユーザを人数としてのみ認識しているため、各ユーザごとの特性を識別していない。解決策として、10 分ごとに集積ツールから集積するユーザごとのデータで識別を行い、各ユーザごとの特性や作業時間を特定することで、他の PM における同一ユーザの作業から作業傾向の取得が可能となる。各ユーザ毎の傾向を提案アルゴリズムの入力としてリスク値を算出することで、より正確にユーザそれぞれが PM 内で VM を使用する時間の傾向を把握できるようになり、更に提案アルゴリズムの正解率を

^{*2} <https://docs.vmware.com/jp/VMware-vSphere/7.0/com.vmware.vsphere.hostclient.doc/GUID-9D0473C3-232B-4280-8F1E-2A99829706F9.html>

向上できる。

本稿の提案では、PM が停止期間に作業が行われていた時間が全体のうち約 1.65%存在した。現在の提案では、時間を 10 分ごとブロック化するという処理を例えば、0:00 から 0:09 の形式で、固定の時間に行うため、ユーザの作業時間に柔軟に対応できないという課題が存在する。この課題を解決するため、提案手法を一部変更し、Risk 値を求めた後、時間を 10 分単位でブロック化しない。停止可能な時間と起動時間が 10 分以上連続しない場合には、起動や停止を行わない。これにより柔軟に起動停止スケジュールを作成でき、PM 停止中のユーザがアクセスする可能性を減少させられる。

7. おわりに

課題は物理マシン内の仮想マシンが使用されていなくても物理マシンが稼働し続け、余剰な電力消費が発生していることである。この課題に対し、過去の VM 使用時間から、ユーザが作業を行う可能性を示す Risk 値を設定する。Risk 値をもとにユーザが物理マシンを使用する時間を予測し、物理マシンの停止スケジュールを作成し、物理マシンを起動と停止を行う手法を提案した。評価実験では、削減した電力量、PM 未使用時間中の停止時間割合、停止時間中におけるユーザの作業時間割合の 3 点で評価する。削減した電力量では、実際に提案を適用し物理マシンの起動と停止を行い、提案適用前と比較し平日 1 週間の消費電力を評価した。評価実験では、提案適用前と比較し提案適用後の方が一週間の総消費電力量を約 53.70%削減した。評価期間中の各 PM における PM 未使用時間中の停止時間では、平均として、未使用時間中の約 51.08%の時間を停止した。停止時間内にユーザが作業を行った時間の割合は約 1.65%となった。本提案により、ユーザの作業時間を避けながらユーザが PM を使用していない期間の約 51.08%を停止した。これにより総消費電力量を約 53.70%削減した。

参考文献

- [1] Liu, W., Wei, K., Yan, X. and Qin, C.: Spatiotemporal Load Regulation Modeling for Internet Data Centers with Applications to Wind Power Fluctuation Suppression, *2023 5th Asia Energy and Electrical Engineering Symposium (AEEES)*, pp. 1157–1163 (online), DOI: 10.1109/AEEES56888.2023.10114214 (2023).
- [2] Kumakura, K., Oguchi, M., Kamiyama, T. and Yamaguchi, S.: CPU Usage Trends in Android Applications, *2022 IEEE International Conference on Big Data (Big Data)*, pp. 6730–6732 (online), DOI: 10.1109/Big-Data55660.2022.10020953 (2022).
- [3] Buyya, R., Beloglazov, A. and Abawajy, J.: Energy-efficient management of data center resources for cloud computing: a vision, architectural elements, and open challenges, *arXiv preprint arXiv:1006.0308* (2010).
- [4] Bahari, H. I. and Shariff, S. S. M.: Review on data center issues and challenges: Towards the Green Data Center,

- 2016 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), pp. 129–134 (online), DOI: 10.1109/ICCSCE.2016.7893558 (2016).
- [5] Zhong, M., Li, D., Chen, T., Chen, H., Li, Y., Shang, X., Zhao, C. and Liu, C.: Idle and Max Power Forecasting for Linear Server Power Consumption Model Using Random Forests, *2021 IEEE 5th Conference on Energy Internet and Energy System Integration (EI2)*, pp. 4295–4299 (online), DOI: 10.1109/EI252483.2021.9713656 (2021).
- [6] Apparao, P., Iyer, R., Zhang, X., Newell, D. and Adelmeyer, T.: Characterization & analysis of a server consolidation benchmark, *Proceedings of the fourth ACM SIGPLAN/SIGOPS international conference on Virtual execution environments*, pp. 21–30 (2008).
- [7] Chen, W., Liu, B., Paik, I., Li, Z. and Zheng, Z.: QoS-Aware Data Placement for MapReduce Applications in Geo-Distributed Data Centers, *IEEE Transactions on Engineering Management*, Vol. 68, No. 1, pp. 120–136 (online), DOI: 10.1109/TEM.2020.2971717 (2021).
- [8] Hogade, N., Pasricha, S. and Siegel, H. J.: Energy and Network Aware Workload Management for Geographically Distributed Data Centers, *IEEE Transactions on Sustainable Computing*, Vol. 7, No. 2, pp. 400–413 (online), DOI: 10.1109/TSUSC.2021.3086087 (2022).
- [9] Möbius, C., Dargie, W. and Schill, A.: Power Consumption Estimation Models for Processors, Virtual Machines, and Servers, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 25, No. 6, pp. 1600–1614 (online), DOI: 10.1109/TPDS.2013.183 (2014).
- [10] Hirose, K.: DC powered data center with 200 kW PV panels, *2018 International Power Electronics Conference (IPEC-Niigata 2018 -ECCE Asia)*, pp. 822–825 (online), DOI: 10.23919/IPEC.2018.8507890 (2018).
- [11] Kumar, R., Khatri, S. K. and José Diván, M.: Effect of Cooling Systems on the Energy Efficiency of Data Centers: Machine Learning Optimisation, *2020 International Conference on Computational Performance Evaluation (ComPE)*, pp. 596–600 (online), DOI: 10.1109/ComPE49325.2020.9200088 (2020).
- [12] Rodríguez-Haro, F., Freitag, F., Navarro, L., Hernández-sánchez, E., Farías-Mendoza, N., Guerrero-Ibáñez, J. A. and González-Potes, A.: A summary of virtualization techniques, *Procedia Technology*, Vol. 3, pp. 267–272 (2012).
- [13] Anala, M. R. and Shobha, G.: Comparative study of application performance on virtual machine and physical machine, *2012 IEEE International Conference on Computational Intelligence and Computing Research*, pp. 1–6 (online), DOI: 10.1109/ICCIC.2012.6510258 (2012).
- [14] Zhao, H., Wang, J., Liu, F., Wang, Q., Zhang, W. and Zheng, Q.: Power-Aware and Performance-Guaranteed Virtual Machine Placement in the Cloud, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 29, No. 6, pp. 1385–1400 (online), DOI: 10.1109/TPDS.2018.2794369 (2018).
- [15] Mohan Raj, V. and Shriram, R.: A study on server Sleep state transition to reduce power consumption in a virtualized server cluster environment, *2012 Fourth International Conference on Communication Systems and Networks (COMSNETS 2012)*, pp. 1–6 (online), DOI: 10.1109/COMSNETS.2012.6151371 (2012).
- [16] Meisner, D., Gold, B. T. and Wensich, T. F.: Powernap: eliminating server idle power, *ACM SIGARCH Computer Architecture News*, Vol. 37, No. 1, pp. 205–216 (2009).
- [17] Fujita, K., Iwasa, E. and Kaneko, M.: Study on data center server management system for using server power estimation technology, *2022 27th Asia Pacific Conference on Communications (APCC)*, pp. 560–565 (online), DOI: 10.1109/APCC55198.2022.9943609 (2022).
- [18] Lally, P., Van Jaarsveld, C. H., Potts, H. W. and Wardle, J.: How are habits formed: Modelling habit formation in the real world, *European journal of social psychology*, Vol. 40, No. 6, pp. 998–1009 (2010).