IB-004

【既発表論文紹介】

Production Hardware Overprovisioning: Real-world Performance Optimization using an Extensible Power-aware Resource Management Framework

坂本 龍一* Tapasya Patki‡ カオ タン* 近藤 正章* 井上 弘士†
上田 将嗣† Daniel Ellsworth§ Barry Rountree ‡ Martin Schulz¶

出典：IEEE International Parallel & Distributed Processing Symposium (IPDPS2017)

HPC システムにおいて，最大消費電力が制約を上回ることを前提として大量のハードウェアを設置し，実行時に消費電力が制約を超えないように制御しつつジョブを実行するオーバープロビジョニング環境が注目されている．オーバープロビジョニング環境では電力制約内でイントールするノード台数が性能や電力特性，コストの面で重要なパラメタである．そこで本稿では，オーバープロビジョニング構成を想定した大規模な HPC システム上で，電力制約を考慮したリソースマネージャを用いて電力と性能の評価を行い，追加ノード台数とワークロードの特性に応じた最適なオーバープロビジョニング構成についての検討を行った IPDPS2017 の論文発表の紹介を行う．

将来の HPC システムでは，消費電力がシステム設計や実行性能を制約する最大の要因となると考えられている．しかし，現在の大規模計算機センターの電力設備状況や物理的な制約からするに，将来的にこれ以上の電力供給能力を持つ計算機センターを設置することは難しい．そこで，我々はシステムのピーク消費電力が制約を超過することを積極的に許容し，ハードウェアが持つ電力ノブや計算に用いるハードウェア資源量を調整することで，限られた電力資源を計算・記憶・通信等の各要素に適応的に分配し，実効電力を制約以下に制御しつつ高い実行効率を得るハードウェアオーバープロビジョニング HPC システムの研究を進めている．このようなオーバープロビジョニングシステムでは，様々な HPC システムに対応可能かつ，拡張可能なリソースマネージャが重要である．そこで，電力制約適応型スケジューラの研究を進めている．

本電力制約適応型リソースマネージャは様々な電力制御アルゴリズムの実装・評価を簡易化するための機能を提供する．多くの研究で用いられている電力制御アルゴリズムは，主に (1)ジョブの消費電力の予測と実行時間の予測，(2)計算ノード資源と電力資源の管理からなっている[1][2]．図 1 は提案する電力制約適応型リソースマネージャの概要を示している．本システムは SLURM を基に電力制御のためのインタフェースを拡張する形で開発を進めている．SLURM はジョブスケジューリングアルゴリズムやノードスケジューリングアルゴリズムをプラグインとして実装できるようになっている．そのため，新しいジョブスケジューリングアルゴリズム等を開発する場合は，これらのプラグインの枠組みを用いることでアルゴリズムの主要な部分の実装に集中することができる．我々は，既存の SLURM と連携するために Power scheduler，Node power manager を新たに追加した．さらに Low-level power plugin interface を追加した．また，アルゴリズムの実装を行う研究者が容易に電力モニタリング，電力キャッピングを行えるように電力モニタリング機能と電力制御機能の追加も行った．Power scheduler は分散されたノードの電力をモニタリングし電力の分配を行うことで，システム全体の電力資源を管理する役割を持ち Power monitor，Power analyzer，Power allocator の 3 つから構

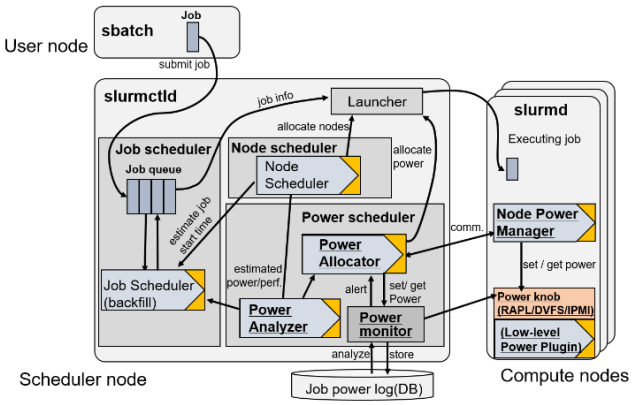

図 1 電力制約適応型 SLURM リソースマネージャの概要

* 東京大学
‡ Lawrence Livermore National Laboratory
† 九州大学
§ Colorado College
¶ Technical University of Munich

成される．Power monitor は周期的にノードの電力をモニタリングし，システム全体の消費電力を計算する役割を持つ．合わせてノードの電力のキャッピングの制御を行う．Power Analyzer はジョブの実行時間を予測する役割を持つ．利用可能なノード数や電力キャッピング等の制約情報からジョブの実行時間を予測する．ジョブスケジューリングプラグインは Power Analyzer プラグインが予測したジョブの実行時間情報を元にジョブの開始時刻を決定することで，システム全体のジョブスループットを最適化する． Power Allocator は Power Analyzer と協力し，静的な電力制御，動的な電力制御を行う機能を担う．Node Power Manager はノードサイドでの電力制御を行う役割を担う．Low-level power plugin はプロセッサに依存する低レベルな電源制御の機能を隠蔽する役割を持つ．Power allocator と Power analyzer はプラグイン形式となっており，研究者は実装評価を行いたいアルゴリズムをこれらのプラグインインタフェースを用いて実装することができる．

オーバープロビジョニング環境では追加ノードを加え余剰電力を有効に活用することで電力効率を向上させることが可能であるが，追加ノードが多すぎると待機電力が増加し，電力効率が悪化する恐れがある．そこで，実際のスパコンセンターに電力制約適応型 SLURM リソースマネージャをインストールし電力利用率の計測を行った．評価では 5 つのジョブミックスを用いた．85W, 70W, 55W の電力を消費する 3 つのジョブを準備しワークロード中に含まれるジョブの含有率の比を変化させる．85W:70W:55W = 1:0:0 (high-only)，3:2:1(many-high)，1:1:1 (middle)，1:2:3 (many-low)，0:0:1 (low-only)としている．ノード数を 400 ノード，540 ノード，680 ノード，820 ノード，960 ノードと変えた場合の電力利用率の結果を図 2 に示す．電力資源利用率は 400 ノードの TDP 電力を 1 として正規化を行っている． この結果よりノード数を 680 台まで増やした場合は電力利用率が向上することが分かる．一方でそれ以上ノード数を増やすと逆に電力利用率が低下することが分かる．追加ノードが少ない場合はノード利用率が高い．これは電力を十分に使うことができるためである．一方で追加ノードが多い場合，追加されたアイドルノードが消費する電力が増えることにより，ジョブが利用できる電力が減少している．

この時のジョブのスループットを図 3 に示す．ジョブスループットについてはノード数が増えるにつれ向上することが図 3 からわかる．これはオーバープロビジョニングにおいて追加した台数分，有効にジョブを実行できたためである．一方で，960 ノードの場合，スループットは大きく低下している．さらに，低下具合はジョブミックスによって大きく異なることが確認できる．CPU インテンシブなジョブが多いような high-only のようなケースではジョブが多くの電力を消費するため，追加ノードが少ない場合であっても電力資源が枯渇しやすい傾向にある．よって追加ノードは少ないほうが好ましい．一方で low-only のように個々のジョブの消費電力が少ないメモリインテンシブなジョブが多い場合，電力資源が枯渇しにくいため，多くのノードを追加しても性能が大きく向上する傾向があり，追加ノードは多い方が良いことが分かる．そのため，今後は追加ノード数を動的に管理する資源管理方式について検討を進める．

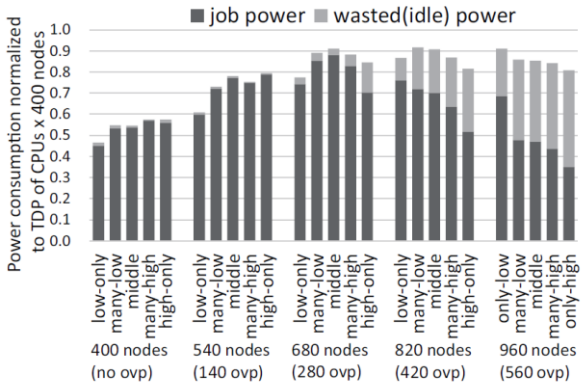

図 2 電力資源の平均利用率

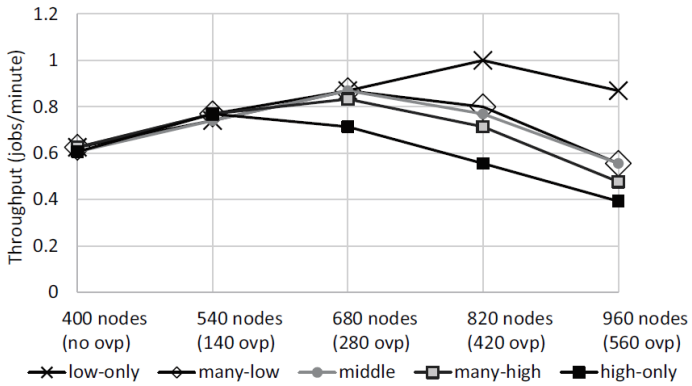

図 3 ジョブの平均スループット(jobs/minute)

参考文献

[1] Patki, T., Lowenthal, D., Rountree, B., Schulz, M. and de Supinski, B.:Exploring Hardware Overprovisioning in Power-constrained, High Performance Computing, HPDC'15

[2] Sarood, O., Langer, A., Gupta, A. and Kale, L.: Maximizing Throughput of Overprovisioned HPC Data Centers Under a Strict Power Budget, SC'14