Vol.2016-HPC-153 No.36
2016/3/3

電力を考慮した「京」の運用改善への取組み

井上文雄[†1]　宇野篤也[†1]　塚本俊之[†1]　末安史親[†2]　池田直樹[†3]　肥田元[†3]
庄司文由[†1]

概要：「京」では，ノードの利用効率の低下を防ぐために，36,865 ノード以上（最大 82,944 ノード）を使用する大規模ジョブを毎月第 2 火曜日から金曜日の 3 日間（大規模ジョブ実行期間）に集中して実行する運用を行っている．しかし，大規模ジョブ実行期間は，特定のジョブが全系を専有する場合が多いので，ジョブによっては電力消費が急激に増加することがある．消費電力が契約電力を超過すると，電力会社からペナルティが課されるため，大規模ジョブ実行期間に実行するジョブに対して，事前審査制度を導入したり，消費電力を 24 時間監視し最大電力を超過した際のジョブ緊急停止プロセスを整備するなど，運用面での対策を実施した．これらの対策を実施した 2014 年 6 月以降，大規模ジョブ実行期間での電力超過は発生していないが，より効率的かつ効果的な対策に改善するべく，事前審査時のジョブ消費電力推定の精度向上とジョブ緊急停止プロセスの簡略化に取組んでいる．本稿では，現状の改善状況と今後の取組みについて報告する．

キーワード：スーパーコンピュータ，「京」，電力超過対策，ジョブ緊急停止，事前審査制度

1. はじめに

スーパーコンピュータ「京」（以下，「京」）は，理化学研究所と富士通株式会社が共同開発した汎用性の高いスーパーコンピュータで，生命科学・医療，エネルギー，防災・減災，次世代ものづくり，物質と宇宙といった幅広い分野で活用されている．2012 年 9 月に共用を開始して以来，システムとして安定して運用を継続している[1]．しかし，共用開始から 1 年が経過した頃から，システム全体の消費電力が想定以上となるジョブが実行され，電力会社からの受電量が契約電力の上限を超える状況が発生する様になった．「京」は低消費電力の CPU を採用するなど，低消費電力を意識した設計となっているが，システムの規模が大きいため，ジョブ実行による消費電力の変動も非常に大きく，場合によっては，消費電力の変動が 4MW を超える場合もある．受電量が契約電力の上限を超えた場合，違約金の支払いが必要となる場合があり，さらに，この状況が複数回発生するようだと電力契約そのものの見直しにつながることになる．実際に 2013 年には，消費電力の超過が 3 回発生したため，違約金の支払いと契約の見直しにより運用コストの増加を招いた．そこで，これを回避するための緊急対策として，ジョブの緊急停止と事前審査制度を導入した[2]．これらの制度を導入した 2014 年 6 月以降，受電量が契約電力を超える状況は発生していない．しかし，これらの対策は緊急的なものであり，実際の運用を考えた場合，ジョブ緊急停止プロセスの自動化やジョブ消費電力推定の精度向上など改良の余地がある．

本稿では，これらの改良と今後に向けての取組みについて述べる．

†1 国立研究開発法人理化学研究所　計算科学研究機構
†2 富士通株式会社
†3 株式会社富士通ソーシアルサイエンスラボラトリ

2. 「京」の概要

「京」は，82,944 台の計算ノードと 1.27PiB のメモリ，11PB のローカルファイルシステム（以下，LFS）と 30PB のグローバルファイルシステム（以下，GFS），フロントエンドサーバなどの周辺機器から構成されている．図 1 に「京」のシステム構成概要を示す．

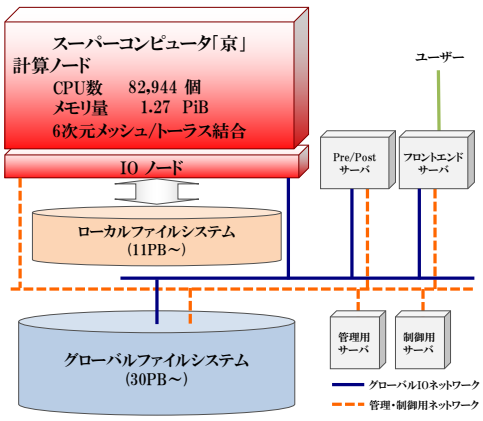

図 1「京」のシステム構成（概要）

「京」の運用に必要な電力は，電力会社から購入している商用電力と，自家発電により供給されている．自家発電として，5MW のガスタービン発電によるコジェネレーションシステム（以下，CGS）を 2 台備え，通常は 1 台ずつ交互に運転している．CGS の燃料となるガスは，ガス会社より購入している[3]．

共用開始時，「京」を運用している計算科学研究機構（以下，AICS）全体の消費電力の最大値として 17MW が想定されていた．表 1 に，17MW の内訳を示す．

表 1 AICS 全体の消費電力見込

内訳	消費電力見込
「京」本体（含むLFS）	10 MW
ジョブ実行による増分	～ 4 MW
その他施設（含むGFS）	～ 3 MW
合計	～17 MW

「京」本体（含む LFS）とその他施設（含む GFS）の想定値は，共用開始前の試験利用期間の実測値を基に決めた値であり，ジョブ実行による増分は LINPACK を実行した際の消費電力を参考にしている．

「京」の運用形態は，通常運用と大規模ジョブ実行運用の 2 つに分けられる．通常運用では，36,864 ノード以下の小中規模ジョブを最大 24 時間まで，大規模ジョブ実行運用では 36,865 ノード以上の大規模ジョブを最大 8 時間まで実行できる．原則，毎月第二火曜日から 3 日間を大規模ジョブ実行期間としている．

3. 「京」の電力変動

共用開始後の AICS 全体の消費電力は，通用運用時で平均 15MW，変動幅±1MW で当初の想定内に収まっていた．

しかし，2013 年度の大規模ジョブ実行時に，AICS 全体の消費電力が想定値の 17MW を超える状況が発生し，電力会社からの受電量が契約の上限を超えることになった．図 2 に電力超過発生時の電力需給状況のグラフを示す．

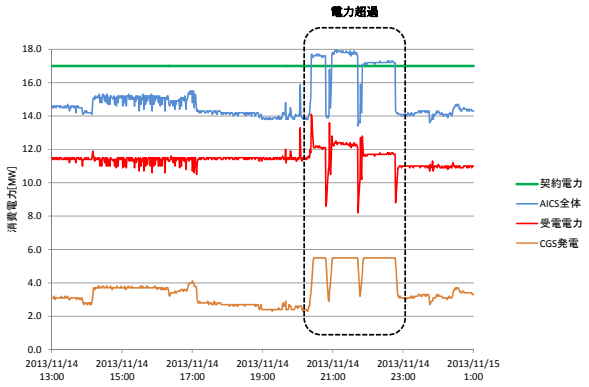

図 2 電力需給状況（電力超過時）

電力会社との契約では，毎時 30 分間（0～30 分と 30～60 分）における電力の平均使用量が契約電力を超えた場合に電力超過と判定され違約金を支払うことになっている．さらに，電力超過が頻繁に発生する場合には，契約内容の見直しを行うことになる．2013 年度には，この電力超過が 3 回発生したため，2014 年度の契約内容を見直すことになり，運用コストの増加の原因となった．このため，早急に電力超過への対策が必要となった．

4. 「京」の電力超過対策

前述のとおり電力超過対策は緊急性を要することから，まずは既存のシステムを最大限に利用した対応とし，運用状況を見ながら改善を行うこととした．また，当面は電力超過の発生する確率が高いと思われる，大規模ジョブ実行期間を対象とした．

まず，ユーザと運用に影響の少ない対応策として，供給電力量を増やす方法を検討した．現在の設備で供給電力を増やす方法として，以下の 2 つの方法が考えられる．

（1）電力会社からの購入電力を増やす

（2）自家発電量を増やす

（1）は，電気料金の増加となるため採用は難しい．（2）の方法では，停止状態の CGS が発電できるようになるまで 1，2 時間ほどの時間を要するため，消費電力が許容範囲を超えてから対応を始めたのでは間に合わない，常時 2 台運転する方法も検討したが，燃料費の増加となるため断念した．また，CGS の発電量を調整することである程度までの電力超過に対応することは可能だが，対応可能時間がオペレータの勤務時間内に限定されるため，夜間休日の対応ができないという課題があり採用は難しい．これらの理由から，供給電力を増やす方法を断念した．

次に，供給側ではなく消費側をコントロールする方法を検討した．検討の結果，ジョブの実行を制御するためユーザへの影響は大きいが，対象は大規模ジョブ実行期間に実行されるジョブに限定することができ，システム側での対応も可能であることから実施することとした．具体的には，消費電力が許容範囲を超えた場合に強制的に消費電力を下げるためのジョブ緊急停止と許容範囲を超えないようにジョブ実行時の消費電力をコントロールする事前審査制度である．

4.1 ジョブ緊急停止

ジョブ緊急停止とは，電力超過の兆候や電力超過が発生した場合に，実行中のジョブを強制的に停止することで，消費電力を下げる仕組みである．図 3 にジョブ緊急停止フローを示す．

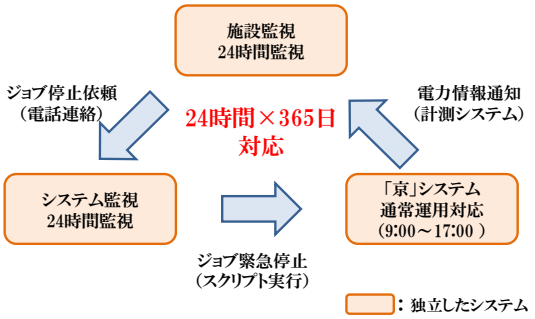

図 3 ジョブ緊急停止フロー

「京」の消費電力は，施設監視により常時監視されている．電力超過の兆候や電力超過が発生した時点で，施設監視担当者からシステム監視担当者へ，ジョブの停止を電話で依頼する．ジョブ停止依頼を受けたシステム監視担当者は，ジョブ停止スクリプトを実行することで，実行中のジョブを停止する．この一連の操作を，消費電力が許容範囲内に下がるまで繰り返す．この時，ノード数と消費電力は比例するとの仮定に基づき，割当ノード数の多いジョブから順に停止する．停止されたジョブは，電力超過が発生しないようにスケジューリングを調整し再実行する．

4.2 事前審査制度

事前審査制度とは，大規模ジョブ実行期間中に投入されるジョブの消費電力を推定し，電力超過を起こさない規模での実行を許可する制度である．図 4 に事前審査制度のフローを示す．

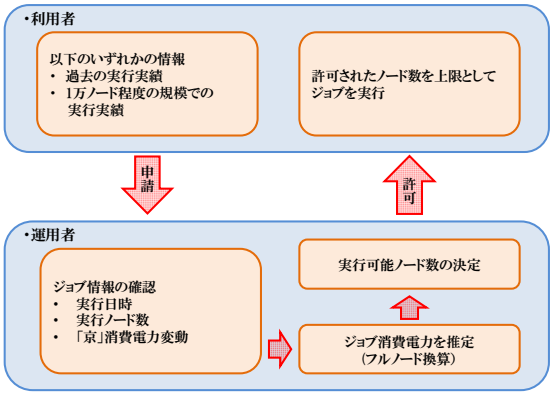

図 4 事前審査制度のフロー

「京」では，ジョブ単位で消費電力を測定する環境がないため，「京」全体の消費電力変動をジョブの消費電力とみなし，以下の式を用いて実行許可ノードを求める．

$$P_{node} = \frac{P_{job}}{N_{job}} \qquad N = \frac{P_{max}}{P_{node}}$$

ここで，P_{node} は 1 ノードあたりの消費電力，P_{job} はジョブの消費電力，N_{job} は計算ノード数，N は実行許可ノード数，P_{max} は許容電力である．

また，実行許可ノード数でも，他のジョブと同時実行された場合，電力超過を起こす可能性があるため，同時実行数の制御も実施する．

5. 電力対超過策の改善

前述の電力超過対策を実施して以降，大規模ジョブ実行期間中の電力超過は発生していない．しかし，実際の運用においていくつかの課題が見えてきた．

5.1 電力超過対策の問題点

5.1.1 ジョブ緊急停止

電力超過発生時には，時間的制約から即座に対応する必要がある．しかし，現在の緊急対策でのジョブ緊急停止は，ジョブ停止までの一連の操作を手動で行っており，ジョブの停止まで時間を要する場合がある．そのため，プロセスの簡略化と自動化による対応時間の短縮化が必要である．

また，停止ジョブの選択方法として，ノード数の大きいジョブから停止する方法を採用しているが，この方法だと電力超過の原因ではないジョブを停止する可能性があるなどの問題点がある．

5.1.2 事前審査制度

現在の事前審査制度では，毎回大規模ジョブを実行する際は審査を受ける必要がある．これはユーザにとって大きな負担であり，手続きの簡略化が必要である．また，審査ジョブから推定したジョブの消費電力と大規模ジョブ実行時の実測値が大きく異なる場合あり，審査精度の向上が必要である．

5.2 電力超過対策の改善策

電力超過対策の問題点に対し，以下のような改善策を検討した．

5.2.1 ジョブ緊急停止

ジョブ緊急停止の即時性における問題点に対し次のような対策を実施した．ジョブの停止は施設監視のオペレータからの連絡でシステム監視担当者が実施しているが，この部分を施設監視のオペレータが直接実施できるようにシステムを構築した．具体的には，システム管理に使用している Web システム上に，ジョブ緊急停止機能を追加し，「京」のシステムに不慣れな施設監視のオペレータでも容易に操作できるインターフェースを作成した．これにより，電力超過発生時に素早い対応をとることが可能となった．

また，停止ジョブの選択では，ノード数ではなくジョブ単位の消費電力を基準に選択する方法とした．「京」ではジョブ単位で消費電力を測定することは出来ない．そのため，システムに取り付けられている温度センサの情報を用いてジョブ単位の消費電力を推定する手法を開発した[4]．一般に，計算機で消費される電力は熱となって排出される．「京」には，ハードウェアの運用状況を監視するために，計算ラック毎に温度センサが搭載され，10 分間隔で CPU 温度，システムボード排気温度が収集されている．本手法では，この温度情報を用いてジョブ単位の消費電力の推定を行う．現在，この推定値を用いた停止ジョブ選択機能の実装を行っている最中である．

5.2.2 事前審査制度

事前審査制度では，大規模ジョブ実行期間中に投入するジョブは毎回事前審査を必須としていたが，即時性のあるジョブ緊急停止環境が整備されたことから，ユーザの利便性を考慮し，過去に実行実績のないジョブを除き，事前審

査なしで大規模ジョブの投入を許可することにした. また, これまでの実行実績から電力超過の可能性が高いジョブや, 事前審査による許可ノード数以上での実行を希望するユーザについては, 一時的に CGS を 2 台運転するなどの対策を実施し，許可ノード数を再調整した上で，CGS オペレータが対応可能な大規模ジョブ実行期間の前半にまとめて実行することにした.

事前審査では，推定値と実際に大規模ジョブとして実行された際の消費電力が大きく異なる場合が多くあった．事前審査ではシステム全体の電力変動からジョブ単位の消費電力を推定しているため誤差が大きくなると仮定し，ジョブ緊急停止で用いた温度情報からの推定手法を審査ジョブの消費電力推定に適用した場合について調査した.

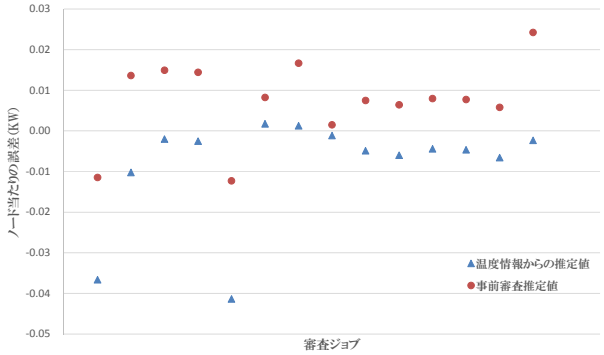

図 5 審査ジョブ消費電力誤差

図 5 は事前審査時の推定電力と温度情報から推定した消費電力を計算ノード単位で比較したものである．ジョブ単位では消費電力の測定ができないため，審査ジョブに対応する大規模ジョブの測定電力からノード単位の消費電力を逆算し，それに対する誤差をプロットしている．大規模ジョブの中には，同じ事前審査ジョブでも消費電力が異なる場合があることが分かっている．それらについては大規模ジョブ実行時のパラメータ等がジョブ毎に大きく異なり，事前審査とは条件が異なっていることが考えられるため除外した．このグラフから，温度情報から消費電力を推定することで精度がよくなっていることがわかる．しかし，一部ジョブについては推定値が大きく異なることが分かっており，現在，これらについてさらに詳細な調査を行っている.

6. 今後に向けた取り組み

緊急対策を実施した後，実行されたジョブの情報と消費電力について収集と分析を行っている．その結果，過去のジョブ実行情報を利用して，ジョブ投入時にそのジョブの消費電力を予測することが可能と判った[5]．現在，この予測値を基に，未来のシステム全体の消費電力変動をリアルタイムに予測するシステムの開発を行っている．このシステムにより，数時間先の電力変動を予測することができ，電力が大きく変動することが予測される場合に，予め CGS を 2 台運転するなどの対応が可能となる．ただし，この場合でも CGS 対応時間が，CGS オペレータの勤務時間に限定される．そこで，電力変動予測をジョブスケジューリングに組み入れ，オペレータの対応が必要となる電力変動の時間帯をオペレータの勤務時間帯に持ってくることはできないか検討を行っている.

また，ジョブの実行情報の分析結果から，アプリケーションの性能情報と消費電力間に相関関係があることも判っている[6]．このアプリケーションの性能情報は，ソースコードから見積もることができ，ソースコードレベルで消費電力を削減できる可能性がでてきた．現在，さらに詳しい調査を実施している.

7. おわりに

本稿では，電力超過時の緊急対策として実施したジョブ緊急停止と事前審査制度の改善状況について述べた.

ジョブ緊急停止の改善点として，ジョブ停止機能の Web 化により，電力超過が発生した場合やその兆候が見られた場合，施設監視オペレータが即座にジョブの緊急停止できる環境を構築した.

事前審査制度の改善点としては，ジョブ緊急停止環境の改善を受け，審査内容を緩和した．さらに，審査時の実行許可ノード数の予測精度をあげる取組みとして，ラック温度情報から推定された消費電力を用いた手法の検討を開始した.

また，新たな取組みとして，過去のジョブ実行情報から，未来の電力変動を予測するシステムの開発と電力変動の予測を用いて消費電力をコントロールするスケジューリングについての検討を開始した．さらに，アプリケーション側からの省電力対策への取組みを開始した.

今後は，これらの取組みで得られた知見について，今後の環境構築への展開を行いたいと考えている.

参考文献

[1] 山本啓二，宇野篤也，塚本俊之，菅田勝文，庄司文由: スーパーコンピュータ「京」の運用状況，情報処理, Vol.55, No.8, pp.786-793 (2014).
[2] 井上文雄，宇野篤也，塚本俊之，松下聡，末安史親，池田直樹，肥田元，庄司文由: 電力消費量の上限を考慮した「京」の運用: 情報処理学会研究報告, Vol.2014-HPC-146, No.4 (2014).
[3] 黒川原佳，庄司文由: スーパーコンピュータ「京」システム概要，情報処理, Vol.53, No.8, pp.759-766 (2012).
[4] 宇野篤也，肥田元，井上文雄，池田直樹，塚本俊之，末安史親，松下聡，庄司文由: 消費電力を考慮した「京」の運用方法の検討，情報処理学会論文誌, ACS, Vol.8, No.4, pp.13-25 (2015).
[5] 山本啓二，末安史親，宇野篤也，塚本俊之，肥田元，池田直樹，庄司文由: 過去の実行実績を利用したジョブの消費電力予測，情報処理学会研究報告, Vol.2015-HPC-151, No.2, (2015).
[6] 黒田明義，北澤好人，塚本俊之，小山謙太郎，井上晃，南一

生: スーパーコンピュータ「京」を用いたアプリケーション性能特性と使用電力の相関解析, 情報処理学会論文誌, ACS, Vol.8, No.4, pp.1-12 (2015).