

ネットワークとサーバの連携によるトータル電力使用量削減を 実現する接続信号シーケンスの提案

長名 保範^{*1}, 栗林 伸一^{*2}

Proposal of a signaling sequence for connection establishment to reduce the total energy consumption by both network and end systems

Yasunori OSANA^{*1} and Shin-ichi KURIBAYASHI^{*2},

ABSTRACT : Most of conventional measures to treat energy consumption handle servers and network devices separately. Therefore, we proposed the integrated method to reduce the total energy consumption by the cooperation of network and end systems, and clarified the information items to be exchanged for the cooperation of network and end systems. This paper proposes the precise signaling sequence for connection establishment to reduce the total energy consumption by both network and end systems.

Keywords : Reduction of power consumption, cooperation of network and end systems, signalling

(Received Sep. 11, 2009)

1. まえがき

移動（通勤や出張）の削減，物流の最適化，生産の最適化など，ICT技術によるエネルギー使用量の大幅削減が期待される（Green by ICT）⁽¹⁾⁻⁽³⁾。しかし，それに伴ってICT機器そのものの電力使用量が増加することが予想される（Green of ICT）。国の試算では，2025年には国全体の電力使用量に占めるICT機器使用電力量の割合は20%程度に達する見込みである⁽¹⁾。この中の約半分がネットワーク機器によるものと想定されており，ユビキタスネットワークやクラウドコンピューティングの普及によりこの比率はますます増加する可能性が高い。

ICT機器の省エネに対する議論は，従来はサーバやデータセンタに対する省エネ技術に対する検討が多い。サーバの電力量削減対策としては，機器への省エネ技術の導入，機器の‘仮想化’だけでなく，サーバ使用率が低い時にクロック数を落とす，サーバ台数を削減（一部サーバの電源OFFまたは休止（スリープ）状態とする），などが提案されている。

ネットワークに関連した電力量削減対策の検討も従来から進められている。サーバと同様に，使用しない機器を休止状態（sleeping）に設定，プロセッサクロック数の調整，回線速度の調整（rate-adaptation），実使用回線数の調整（link aggregation），経路集約，などがその主な対策である。

ところで，従来より電力使用量削減対策はサーバなどのエンドシステムとネットワークで別々に検討されることが多い。このため，例えば電力量削減対策としてサーバの処理速度を低下させると処理時間が長くなり，その結果ネットワーク帯域の保持時間も長くなり，ネットワーク側は逆に電力使用量が増加してしまう可能性がある。一方，ネットワーク側とエンドシステム側で連携した省エネ対策も考えられる。

これを踏まえ，筆者らはエンドシステム側とネットワーク側で連携することにより，それぞれ独立に実施する場合に比べてトータルの電力使用量を大幅に削減できる可能性を示した。そして，エンドシステム側とネットワーク側の具体的な連携方法と両者でやりとりすべき情報とそれを踏まえた信号シーケンスの概要を提案した⁽⁴⁾⁻⁽⁶⁾。また，近年の省エネルギー法の改正などを踏まえると，今後は利用者毎のネットワーク電力使用量の把握も必要となると想定される。そのため，筆者らは測定した交流

^{*1}：理工学部情報科学科助教 (osana@st.seikei.ac.jp)

^{*2}：理工学部情報科学科教授 (kuribayashi@st.seikei.ac.jp)

トラフィック量と定期的に調査するパケット転送経路情報をもとに、ネットワーク電力使用量を各利用者に簡易に按分する手法も提案した⁽⁴⁾⁻⁽⁶⁾。

本稿では、筆者らが提案したエンドシステム側とネットワーク側連携に必要な接続信号シーケンスの詳細を明らかにする。

2. エンドシステム側とネットワーク側連携に必要な接続信号シーケンスの概要

筆者らが文献(3)～(5)で提案した接続信号シーケンスを図1に示す。これは、単位時間当たりの想定電力使用量をサービス要求毎に推定し、それをコネクション確立時にエンドシステムとネットワーク間で交換するという考え方に基いたものである。また、ケース1とケース2はエンドシステム側からトータル電力使用量の削減策を提案する場合、ケース3とケース4はネットワーク側から提案する場合、に対応する。

図1において使用する記号の定義を以下に説明する。なお、図1では今回提案する電力使用量削減策に関連するパラメータだけを表示している。表記されていない相手アドレスや要求された計算能力／帯域／サービス処理時間などのパラメータも当然コネクション確立要求メッセージに含まれ、エンドシステム、ネットワークノードで処理されるものとする。また、帯域は両方向同じ値が

割り当てられるものとする。

- Setup: コネクション確立要求メッセージ, Connect: コネクション確立応答メッセージ, Connect Ack: コネクション確立応答確認メッセージ
- D_A, D_B : トータル電力使用量の削減策を実施するとエンドシステム A, エンドシステム B それぞれで削減できる電力使用量
- U_A, U_B : トータル電力使用量の削減策を実施するとエンドシステム A, エンドシステム B それぞれで増加する電力使用量
- D_1, D_2 : トータル電力使用量の削減策を実施するとノード 1, 中継ノード, ノード 2 それぞれで削減できる電力使用量
- U_1, U_2 : トータル電力使用量の削減策を実施するとノード 1, 中継ノード, ノード 2 それぞれで増加する電力使用量
- flag: トータル電力使用量の削減策を実施する場合は '1' (ON), 実施しない場合は '0' (OFF), を設定。

3. 接続信号シーケンスの具体例

3.1 ケース1の具体例

- ① まず、エンドシステム A は図2に示す状態になっているものとする。つまり、通常状態では稼働している3つのサーバの1つをスリープ状態にしている（処理負荷が減ったため）。これにより、電力使用量は通常状

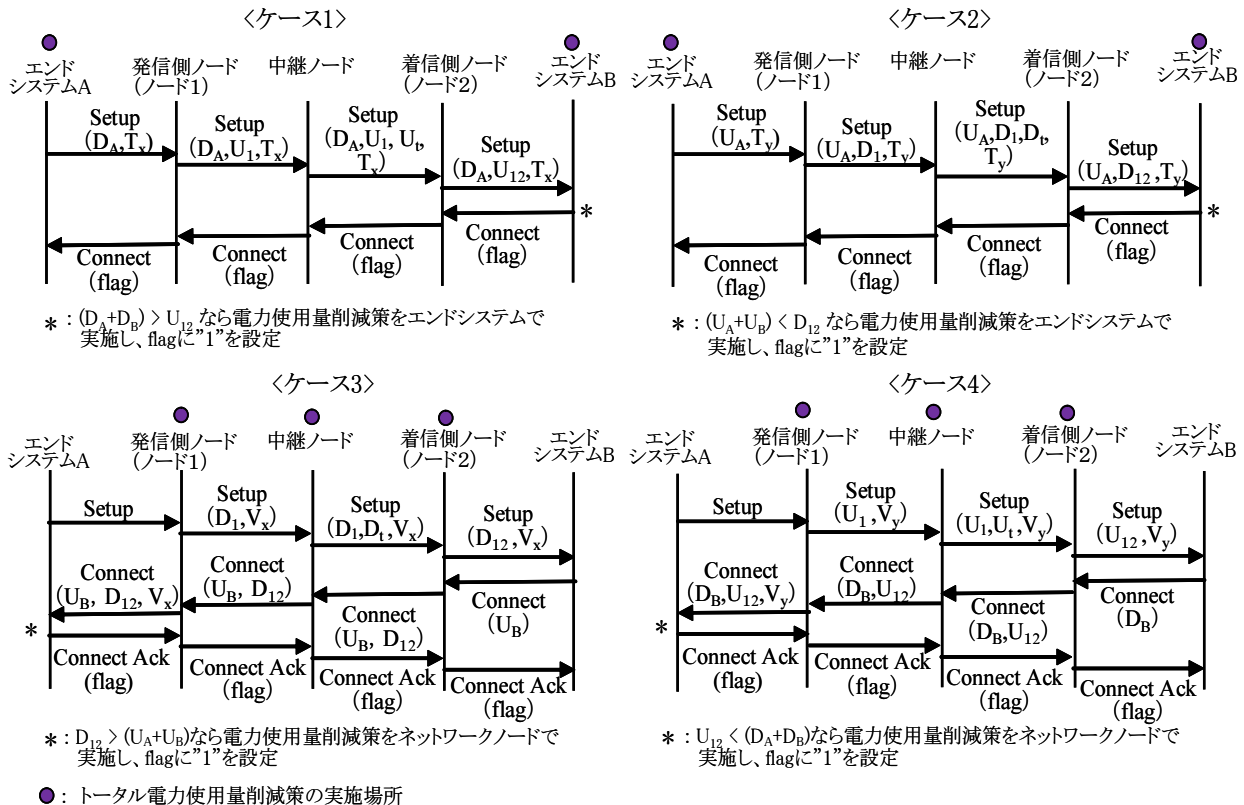


図1. エンドシステム側とネットワーク側の連携を前提とした接続信号シーケンス例

態よりも D_A だけ少なくなっている。また、最大割り当て可能な計算能力は W_x とする。

- ② 新たな要求が発生し、計算能力を W_x 以上必要とする場合、スリープ状態になっているサーバを稼動中に戻す必要がある。それによって、電力使用量が D_A だけ増加することになる。

ここで、新たな要求に割り当てる計算能力を W_x 以下にできれば、スリープ状態になっているサーバを稼動中に戻す必要がないため電力使用量は増加しない（他の2つのサーバはもともと稼動中であり、それを使用するだけ）。

- ③ そのため、もともと要求された計算能力が W_x 以上の場合には、エンドシステム A は新たな要求に割り当てる計算能力を W_x 以下にする電力使用量削減策を提案する（もともと要求された計算能力は W_x 以上とする）。つまり、 D_A だけ削減できると言える。

計算能力を小さくすることにより、サービス処理時間が増加する。その増加分を T_x とする。ネットワークノードは帯域を T_x 分だけ長く保持することで増加する電力使用量（もちろん増加しない場合もある）を計算することができる。

- ④ エンドシステム A は、 D_A と T_x をコネクション確立要求メッセージ (Setup) に載せて発信側ノード（ノード1）に送る。
- ⑤ ノード1は、帯域を T_x 分だけ長く保持することで増加する電力使用量 U_1 を算出し（ $U_1=0$ の場合もある）、 D_A 、 T_x と一緒に Setup メッセージに含め中継ノードに送る。
- ⑥ 同様に、中継ノードは T_x をもとに中継ノードで増加する電力使用量 U_i を算出し（ $U_i=0$ の場合もある）、 D_A 、 U_1 、 T_x と一緒に Setup メッセージに含め着信側ノード（ノード2）に送る。
- ⑦ ノード2は、 T_x をもとに着信ノードで増加する電力使用量 U_2 を算出し（ $U_2=0$ の場合もある）、 D_A 、 U_{12} （ $=U_1 + U_i + U_2$ ）、 T_x を Setup メッセージに含め着信側のエンドシステム B に送る。
- ⑧ エンドシステム B は、 T_x をもとに自身で削減できる電力使用量 D_B （エンドシステム A と同じように処理能力を抑えることで実現）を算出する。 $D_B=0$ の場合もある。そして、

$$(D_A + D_B) > U_{12} \quad (1)$$

ならばエンドシステム A が提案したトータル電力使用量の削減策を実施し、 $flag=1$ を設定したコネクション確立応答メッセージ (Connect) をノード2に返信する。逆に、式(1)が成立しない場合は、エンドシステム

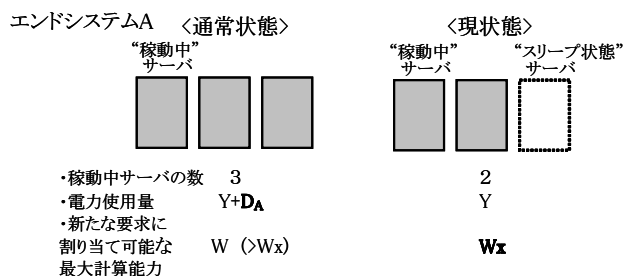


図2. 図1ケース1の具体的（エンドシステムAの状態）

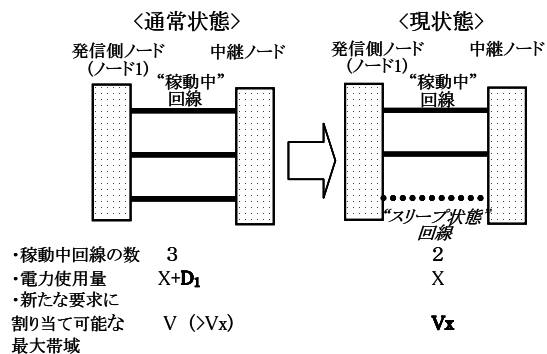


図3. 図1ケース3の具体例（発信側ノード1ー中継ノード間）

A が提案したトータル電力使用量の削減策を実施せず、 $flag=0$ を設定した Connect メッセージをノード2に送る。

- ⑨ ノード2、中間ノード、ノード1は Connect メッセージをエンドシステム A に向け中継する。
- ⑩ コネクション確立応答を受信したノードエンドシステム A は $flag=1$ なら提案した電力使用量削減策を実施し、 $flag=0$ なら削減策を実施しない。

なお、図2では稼動中サーバ数の増減を前提想定したが、サーバのクロック数増減を前提とした例も考えられる。

3.2 ケース2の具体例

ケース2は提案する電力使用量削減策がケース1と逆、つまりエンドシステム側の電力使用量を増加させる（これによってサービス処理時間が T_y だけ短縮）、以外はケース1とほぼ同じ手順となるため、詳細は省略する。

3.3 ケース3の具体例

- ① まず、発信側ノード（ノード1）と中継ノード間は図3に示す状態になっているものとする。つまり、通常状態では稼動している3本の回線1本をスリープ状態にしている（トラフィック量が減ったため）。これにより、電力使用量は通常状態よりも D_1 だけ少なくなっている。また、割り当て可能な最大帯域は V_x とする。

- ② 新たな要求が発生し、帯域を V_x 以上必要とする場

合、スリープ状態になっている回線を稼動中に戻す必要がある。それによって、電力使用量が D_1 だけ増加することになる。

ここで、新たな要求に割り当てる帯域を V_x 以下にできれば、スリープ状態になっている回線を稼動中に戻す必要がないため電力使用量は増加しない（他の 2 つの回線はもともと稼動中であり、それを使用するだけ）。つまり、 D_1 だけ削減できると言える。

- ③ そのため、もともと要求された帯域が V_x 以上の場合には、発信側ノード（ノード 1）は新たな要求に割り当てる帯域を V_x 以下にする電力使用量削減策を提案する。帯域を小さくすることにより、エンドシステム間のデータ転送時間が長くなり、その結果エンドシステムの電力使用量が増加する可能性がある。
- ④ 発信側のエンドシステム A からコネクション確立要求メッセージ (Setup) を受信したノード 1 は、帯域の削減による電力使用量削減策を提案する。帯域の削減によるノード 1 の電力使用量削減分 D_1 を算出する。そして、帯域の削減値 V_x と一緒に Setup メッセージに含め中継ノードに送る。
- ⑤ 同様に、中継ノードは V_x をもとに中継ノードの電力使用量削減分 D_1 を算出し ($D_1=0$ の場合もある)、 D_1 、 V_x と一緒に Setup メッセージに含め着信側ノード（ノード 2）に送る。
- ⑥ ノード 2 は、 V_x をもとにノード 2 の電力使用量削減分 D_2 を算出し ($D_2=0$ の場合もある)、 $D_{12} (=D_1+D_2)$ を Setup メッセージに含め着信側のエンドシステム B に送る。なお、ノード 2 は D_{12} を一時的に保持する。
- ⑦ エンドシステム B は、 V_x をもとに電力使用量増加分 U_B を算出する（帯域を小さくすることにより、エンドシステム間のデータ転送時間が長くなるため）。 $U_B=0$ の場合もある。そして、Connect メッセージに含めてノード 2 に返信する。
- ⑧ ノード 2 は保持していた D_{12} と一緒に U_B を Connect メッセージに含めて中継ノード経由でノード 1 に送る。
- ⑨ ノード 1 は D_{12} 、 U_B 、 V_x を含む Connect メッセージをエンドシステム A に送る。
- ⑩ エンドシステム A は、エンドシステム B と同様に V_x をもとに電力使用量増加分 U_A を算出する。そして、
$$D_{12} > (U_A+U_B) \quad (2)$$
ならば、エンドシステム A は $\text{flag}=1$ を設定した Connect Ack メッセージをノード 1 に送る。
- ⑪ Connect Ack メッセージを受信したノード 1、中継ノード、ノード 2 は、 $\text{flag}=1$ なら提案した電力使用量の

削減策を実施し、 $\text{flag}=0$ なら削減策を実施しない。

なお、図 3 では稼動中回線数の増減を前提想定したが、回線速度増減 ($1\text{Gb/s} \rightarrow 100\text{Mb/s}$ など) を前提とした例も考えられる。

3. 4 ケース 4 の具体例

ケース 4 は提案する電力使用量の削減策がケース 3 と逆、つまり帯域を増加させる（これを V_y と表示）ことによりノードの電力使用量を増加させる（これによってエンドシステム間のデータ転送処理時間が短くなる）、以外はケース 3 とほぼ同じ手順となるため、詳細は省略する。

4. むすび

本稿では、既に提案したエンドシステム側とネットワーク側で連携することでトータルの電力使用量を削減できる方式を実現するための接続信号シーケンスの具体例を明らかにした。

今回は 1 つ 1 つの要求毎に判断することを前提としているが、複数のユーザや複数の要求により資源は共用されるのが一般的である。今後はそれを考慮した接続信号シーケンスを検討する必要がある。また、今回は 1:1 接続形態を前提としたが、上り方向トラフィック量と下り方向トラフィック量のアンバランス（片方向通信も含む）な形態、マルチポイント接続形態、複数ネットワーク経由形態、などを前提とした検討が今後必要である。

【参考文献】

- (1) 経済産業省，“グリーン IT イニシアティブ”，平成 19 年 12 月 6 日
- (2) グリーン IT 推進協議会，“ポスト京都に挑む日本のグリーン IT”，グリーン IT フォーラム（2009.3）
- (3) ITU Symposium on “ICTs and Climate Change” Summary Report, London, June 17&18,2008
http://www.itu.int/dms_pub/itu-t/oth/06/0F/T060F0060090001PDFE.pdf
- (4) 畠山，長名，栗林，“エンドシステムとネットワークの連携による電力使用量削減策の提案”，情報処理学会 研究報告 2006-DPS-137（2008.11）
- (5) 栗林，長名，“エンドシステムとネットワークの連携によるトータル電力使用量削減法の実現方式”，電子情報通信学会 情報ネットワーク研究会 IN2009-27（2009.7）
- (6) K. Hatakeyama, Y. Osana and S. Kuribayashi, “Reducing

total energy consumption with collaboration between network and end systems,"NBiS2009, 7B-4, Aug. 2009.