社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

IoT端末を考慮したシグナリング制御による モバイルコアノードの資源利用の効率化

安達　智哉† 　阿部　修也† 　長谷川　剛†† 　村田　正幸†

† 大阪大学大学院情報科学研究科　〒 565–0871 大阪府吹田市山田丘 1–5
†† 東北大学電気通信研究所　〒 980–0812 宮城県仙台市青葉区片平二丁目 1 番 1 号
E-mail: †{to-adachi,s-abe,murata}@ist.osaka-u.ac.jp, ††hasegawa@riec.tohoku.ac.jp

あらまし　モバイルネットワーク事業者は，自身が運用するモバイルコアネットワークのノード資源が枯渇しないように，収容端末台数や接続頻度に応じて資源割り当てを行う必要がある．一方，近年増加している IoT 端末は，接続される端末の台数を予測することは困難である．また，端末の通信開始時のシグナリング処理を軽減するために，RRC Connected Inactive と呼ばれる状態を導入し，端末情報をメモリに一時的に保存することが検討されている．これらのことから，今後，モバイルコアネットワークノードへの CPU やメモリ負荷が大きく変動することが予想され，効率的な資源割り当てが求められる．そこで本報告ではモバイルコアネットワークノードにおける CPU とメモリ間の負荷のオフロード手法を提案する．具体的には，ネットワークの負荷に合わせて端末の状態を制御することにより，モバイルコアネットワークノードの CPU およびメモリ負荷のバランスを調整し，収容可能な端末台数を最大化する．端末の状態制御は，端末がデータ送受信後にアイドル状態に遷移するまでの時間を制御することで実現する．提案手法により，モバイルコアネットワークノードの資源を増強することなく，収容可能な端末台数が最大で約 150%向上することを示す．

キーワード　モバイルコアネットワーク，M2M/IoT 通信，Long Term Evolution(LTE)，RRC Connected Inactive

1. ま え が き

モバイルネットワーク事業者は，自身が運用するモバイルコアネットワークのノード資源が枯渇しないように，収容端末台数や接続頻度に応じて資源割り当てを行う必要がある．主なノード資源として，CPU およびメモリが挙げられる。CPU は，アタッチやデタッチ等のシグナリング処理を実行するために必要とされる資源である．一方メモリは，ベアラなどのセッション情報を保持するために必要とされる資源である．これらのノード資源は，モバイルネットワークにおける通信を実現するために必須であり，どちらか一方でも枯渇することは許されない．

一方，近年は IoT 端末の急激な増加が注目されている．IoT 端末は，スマートフォンのようなユーザ端末とは異なり，家電や自動車，電気メータ，センサなど様々な場所，様々な用途で使用される可能性があり，端末の台数およびその分布を予測することは困難である．

また、IoT 端末は通信特性においてスマートフォン等の従来の端末とは異なり，データ送信に周期性や間欠性を持つという特徴がある。そのため，データの送信ごとにアイドル状態と接続状態を遷移することが予想される．その結果，端末のネットワーク接続やデータ送信に必要なシグナリングに関する通信や処理を行う，制御プレーンの輻輳が悪化すると考えられる．このような問題に対し，RRC Connected Inactive と呼ばれる IoT 端末の新たな状態を導入することによって，シグナリングの削減を目標とする研究が行われている [1,2]．RRC Connected Inactive とは，端末がネットワークから切り離された後も，モバイルコアネットワークでは，端末のセッション情報をメモリに保持し続けている状態である．端末のセッション情報を破棄するアイドル状態とは異なり，モバイルコアネットワークノードのメモリ資源に負荷が発生する一方で，その端末が再び接続状態へ遷移する際に発生するシグナリングの一部を削減することが可能となる。これにより，シグナリングを処理する際にモバイルコアネットワークノードに発生する CPU の負荷が削減され、制御プレーンの輻輳の抑制が期待できる．実際，文献 [1,2] においては，RRC Connected Inactive を導入することにより，シグナリングオーバヘッドの削減が可能であることを示している．

このように，接続台数の予測が難しい IoT 端末の普及や，モバイルコアネットワークノードに与える負荷を変動させるような新たな状態の導入により，モバイルコアネットワークノードへの CPU やメモリ負荷が大きく変動することが予想される．そのため、モバイルネットワーク事業者は，これまで以上に効率的に資源割り当てを行う必要がある．

(※ここから下の内容はまだ修正できておりません)

上述のような資源需要の予測が難しく，変動が激しいネットワークに対して，収容可能な端末台数の増加を目的とした既存研究には，スケールアウトの考え方を用いたものが多い [3–7]．これらの研究では主に稼働するサーバやインスタンスの数を資源の需要に応じて変動させることにより，ネットワークの変動に対応している [3]．しかし，この手法では，本来必要とされている資源量よりも多くの資源が供給される，オーバープロビジョニングが発生する問題がある．なぜなら，これらの研究では，サーバやインスタンス一台あたりの資源量は一定であることを前提にした研究が多く，細かい粒度で資源量を制御できないためである．また，必要とされる CPU とメモリの資源量の比があらかじめ分かっていることを前提とした研究が多く，必要とされる資源量の比が未知の場合は資源の効率的な利用ができないからである．例えば，CPU の資源不足を解消するためにケールアウトを行った場合，CPU と同時にメモリも増加する．しかし，メモリは元々ボトルネックにはなっていないため，新たに追加されたメモリはオーバープロビジョニングされたことになる．文献 [4] では，IoT 端末を収容している MME を単純にスケールアウトした場合，一部の資源が不必要にプロビジョニングされる可能性があることを述べている．

このような背景から，EPC ノードにおける CPU とメモリの資源需要の予測が難しような状況や変動が大きいような状況においても，どちらかの資源がボトルネックにならずに，効率的に資源を活用するアーキテクチャを考えることは重要である．実際，CPU とメモリの資源を効率よく活用する研究は，データセンタなどの分野では行われている．文献 [8] では，Server Disaggregation の考えをデータセンタに適用し，CPU やメモリなどの資源をモジュール化し，需要に合わせて自由に組み替えることを可能にすることにより，資源の効率的な利用が可能であることを示している．しかし，Server Disaggregation にはいくつかの課題がある．まず，CPU とメモリを分離するためには，大きなコストがかかる点が挙げられる．文献 [9]，[10] では，CPU とメモリを分離するためには，両者を結ぶための新しい高帯域ネットワークが必要になると述べている．また，両者の物理的な距離が増加することによって発生する遅延も考慮する必要があるため，新たなメモリアーキテクチャの構築が必要であると述べている．文献 [11] では，メモリを分離するためには，低遅延かつ高帯域のネットワーク接続が必要となるが，それを実装するためのコストは従来と比較して大幅に増加すると述べている．実際，文献 [12] では，Disaggregated Server に基づくインテルのラックスケールアーキテクチャを示しているが，このモデルでも CPU とメモリの分離はできていない．課題の 2 つ目として，短いタイムスケールでの制御が難しいという問題が挙げられる．これは，そもそも Server Disaggregation は短いタイムスケールでの制御を目的とした技術ではないためである．例えば，ストレージをモジュール化してサーバと分離する手法について述べられている文献 [13] では，時間スケールの細かいストレージ制御を行った場合，ストレージの再割り当て処理に伴うオーバーヘッドが大きくなると述べている．また，頻繁に構成を変化させることは，コスト面や消費電力の面でも不利である．

このように，Server Disaggregation を用いた制御では，制御に伴うオーバーヘッドの発生が避けては通れない課題となると予想される．特に，モバイルネットワークのように，突発的なトラヒックの増加が発生し，数分以下のオーダで資源量の制御を行う必要があるネットワークにおいては細かいタイムスケールでの効率的な制御が求められる．

そこで，本報告ではモバイルネットワークに特化した，柔軟かつ効率的な，EPC ノードにおける CPU とメモリ間の負荷のオフロード手法を考案する．具体的には，ネットワークの負荷に合わせて，UE の状態を制御することにより，メモリおよび CPU に与える負荷のバランスを調整させる．UE の状態の制御は，UE が最後にデータを送信したあと，Connected Inactive 状態からアイドル状態に遷移するまでの時間を設定することで実現する．この手法により，CPU が過負荷である場合は，UE が最後にデータを送信したあと，Connected Inactive 状態からアイドル状態に遷移するまでの時間を長く設定することにより，メモリの負荷を増加させる代わりに CPU の負荷を削減することが可能である．またその逆に，メモリが過負荷である場合は，この時間を短く設定することにより，CPU の負荷を増加させる代わりにメモリの負荷を削減できる．この時間の再設定処理は，数分単位のオーダーで可能でありかつ，それに伴い発生するオーバーヘッドは，Server Disaggregation と比較して僅かである．また，既存のシグナリングアーキテクチャに変更を加えることが可能であれば，秒単位の時間スケールでの制御も可能であると考えられる．

しかし，提案手法には限界もある．それは，対応可能な資源需要に制限があることである．なぜなら，提案手法では，限られた CPU およびメモリの資源を効率的に利用することは可能であるが，双方の資源が過負荷になるような状況には対応できないからである．特に長期的かつ大規模な資源需要の変動に対しては，Server Disaggregation を用いた制御の方が適している．また，提案手法を用いなかった場合には常時 Connected Inactive 状態であった UE が，提案手法を用いることによりアイドル状態へと遷移する可能性があるため，データを送受信にかかる遅延時間が増加するなど，QoS の低下が発生する可能性がある．しかし，電気メータや気温計のようなリアルタイム性を必要としない IoT 端末であれば，これらの QoS の低下は無視できると考えられる．

そのため，提案手法と Server Disaggregation やスケールアウト/スケールインを組み合わせることにより，より効果のある資源管理が可能であると考えられる．例えば，長期的かつ大規模な制御は Server Disaggregation を用いて行う一方で，Server Disaggregation で対応できないような短いタイムスケールでの制御は提案手法を用いて行う．もしくは，資源需要の大きな変動に対してはスケールアウト/スケールインを用いて対応した上で，細かな調整に対しては提案手法を用いて対応する．上述ように，Server Disaggregation やスケールアウト/スケールインと提案手法組み合わせ，双方のデメリットを補うことで，モバイルネットワークの資源利用効率を向上させること

ができる．

References

[1] S. Hailu, M. Saily, and O. Tirkkonen, “RRC State Handling for 5G,” *IEEE Communications Magazine*, vol. 57, no. 1, pp. 106–113, Jan. 2019.

[2] I. L. Da Silva, G. Mildh, M. S ä ily, and S. Hailu, “A Novel State Model for 5G Radio Access Networks,” in *Proceedings of 2016 IEEE International Conference on Communications Workshops (ICC)*, May 2016, pp. 632–637.

[3] M. Shimizu, H. Nakazato, and H. Seshake, “Scale-Out Architecture for Service Order Processing Systems,” in *2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*, May 2013, pp. 880–883.

[4] P. C. Amogh, G. Veeramachaneni, A. K. Rangisetti, B. R. Tamma, and A. A. Franklin, “A Cloud Native Solution for Dynamic Auto Scaling of MME in LTE,” in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Oct. 2017, pp. 1–7.

[5] I. Alawe, Y. Hadjadj-Aoul, A. Ksentini, P. Bertin, and D. Darche, “On the Scalability of 5G Core Network: The AMF Case,” in *Proceedings of 2018 15th IEEE Annual Consumer Communications Networking Conference (CCNC)*, Jan. 2018, pp. 1–6.

[6] Y. Ren, T. Phung-Duc, J. Chen, and Z. Yu, “Dynamic Auto Scaling Algorithm (DASA) for 5G Mobile Networks,” in *2016 IEEE Global Communications Conference (GLOBECOM)*, Dec. 2016, pp. 1–6.

[7] C. H. T. Arteaga, F. Rissoi, and O. M. C. Rendon, “An Adaptive Scaling Mechanism for Managing Performance Variations in Network Functions Virtualization: A Case Study in an NFV-Based EPC,” in *2017 13th International Conference on Network and Service Management (CNSM)*, Nov. 2017, pp. 1–7.

[8] M. Mahloo, J. M. Soares, and A. Roozbeh, “Techno-Economic Framework for Cloud Infrastructure: A Cost Study of Resource Disaggregation,” in *Proceedings of 2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*, Sep. 2017, pp. 733–742.

[9] “Intel’ s Disaggregated Server Rack,” Moor Insights Strategy, Technical Report (TR), Aug. 2013.

[10] C. Devaki and L. Rainer, “Enhanced Back-off Timer Solution for GTP-C Overload Control,” Feb. 2016. [Online]. Available: http://www.freepatentsonline.com/y2016/0057652.html

[11] B. Abali, R. J. Eickemeyer, H. Franke, C. Li, and M. Taubenblatt, “Disaggregated and Optically Interconnected Memory: When will it be cost effective?” *CoRR*, vol. abs/1503.01416, 2015. [Online]. Available: http://arxiv.org/abs/1503.01416

[12] “Disaggregated Servers Drive Data Center Efficiency and Innovation,” Intel Corporation, Technical Report (TR), Jun. 2017.

[13] S. Legtchenko, H. Williams, K. Razavi, A. Donnelly, R. Black, A. Douglas, N. Cheriere, D. Fryer, K. Mast, A. D. Brown, A. Klimovic, A. Slowey, and A. Rowstron, “Understanding Rack-Scale Disaggregated Storage,” in *Proceedings of 9th USENIX Workshop on Hot Topics in Storage and File Systems (HotStorage 17)*. Santa Clara, CA: USENIX Association, 2017. [Online]. Available: https://www.usenix.org/conference/hotstorage17/program/presentation/legtchenko