パブリッククラウドを用いた Hadoop の自動拡張機能の実現

1230313　柏原　星司　【 分散処理 OS 研究室 】

1　はじめに

近年, 扱うデータの巨大化に伴い, 分散処理ソフトウェアの需要が高まっている. その代表例として Apache Hadoop(以下 Hadoop と略す) が挙げられる. Hadoop の処理能力は, サーバが 1000 台程度あれば線形にスケールされる一方, 最低限 10 台以上のサーバが必要であるといわれている [1]. そのため初期投資が高い傾向にあるといえる. 本研究ではパブリッククラウドを用いた CPU 不足時の自動拡張機能によって初期投資を抑えた Hadoop システムの実現について検討する.

2　Hadoop の自動拡張機能の実現

2.1　拡張機能の構成

図 1 に拡張機能のシステム構成を示す. オンプレミス側にマスターサーバ, スレーブサーバ共に設置し, パブリッククラウド側にはスレーブサーバのみ設置し, オンプレミス環境で Hadoop 処理を実行する. オンプレミス環境とパブリッククラウド環境の仲介のため, ポートフォワーディングを用いる.

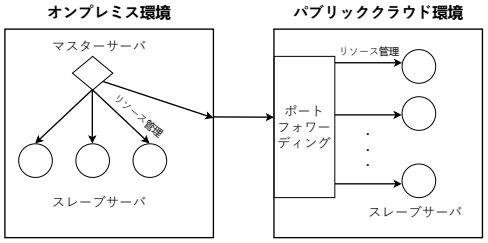

図 1　システム構成

2.2　拡張機能の処理

図 2 に拡張処理の流れを示す. 拡張処理はすべてのスレーブサーバの CPU 使用率を取得し, 平均 CPU 使用率が閾値を超過していることを検知すると, 対象のインスタンスを起動することによって実装をおこなう. Hadoop 処理が終了した場合, 拡張したスレーブサーバの停止をおこなう. インスタンスの起動および停止には Java の SDK を用いる.

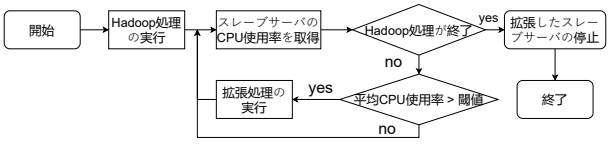

図 2　拡張処理の流れ

3　評価

本実験では本学のネットワークのセキュリティの関係上, オンプレミス環境とパブリッククラウド環境である AWS との接続が難しいと判断したため,EC2 を異なる VPC に配置することによる疑似環境での検証を行う. 実験の環境としてインスタンスタイプを m1.medium, OS を Ubuntu server 18.0.4 としたものを用意し, オンプレミス環境のスレーブサーバ数を 2, CPU 使用率の閾値を 90%に設定し検証をおこなう.

本機能の有用性を示すために以下の二項目を評価する.

1. インスタンス起動までの遅延時間
2. サンプルプログラムの実行時間

起動までの遅延については, 閾値を超過してからデータノード, ノードマネージャの両プロセスが起動するまでの時間を評価する. サンプルプログラムの実行時間については,Apache 公式が提供しているプログラムの中から PI を採用し, サンプルプログラム 10 回の平均実行時間を取得する. そして, 拡張機能の有無による平均実行時間の変化について評価を行う. 結果を以下に示す.

インスタンス起動までの遅延は平均 131.18 秒であった. また, サンプルプログラム実行時間は以下のような結果となった.

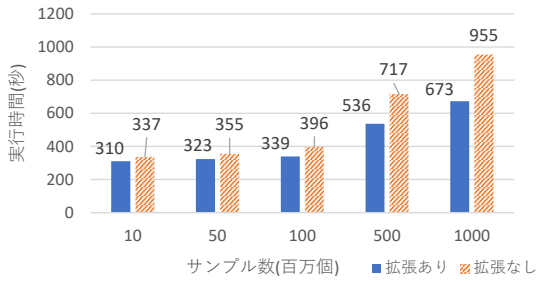

図 3　実行時間の変化

インスタンス起動までの遅延から, 実行時間が遅延時間より多い場合は効果がある. 本実験において, 図 3 より, いずれのサンプル数においても実行時間の減少を確認することができ, 特にサンプル数が $1000 * 10^6$ の時は 30%程度, 減少していることがわかる. この効果は, オンプレミスからパブリッククラウドへの拡張においても同様であると考えられる. そのため, 初期投資を抑えた Hadoop システムとして実用性が期待される.

4　まとめ

本研究では EC2 上で Hadoop の拡張機能を実現し, 有用性を確認した.

参考文献

[1] 清水宣行, 山口崇, 土田誠, 山口亜希子 : Hadoop を利用した超並列計算処理のチューニング指針, P ROVISION No.72 / Winter 2012, p.81-p.88(2012).