

ストリーム処理におけるダイナミック・パーティション手法の試作

A Prototype of Dynamic Partitioning Method for Stream Processing

東松 一真[†] 大濱 開[†] 笠松 大佑[†]

Kazuma TOMATSU[†] Haruki OHAMA[†] Daisuke KASAMATSU[†]

[†] 創価大学理工学部情報システム工学科

[†] Faculty of Science and Engineering, Soka University

1. はじめに

近年, ゲートレス改札やデジタルチケットといった非接触型サービスの研究開発[1]が進められている. 本稿では, リアルタイムに, これらのサービスで生成されるビッグデータを処理できるストリーム処理の偏りを改善するダイナミック・パーティション手法を提案する.

2. 課題

並列分散ストリーム処理[2][3]では, データが持つ何らかのルールに基づいてデータを振り分ける必要がある. その振り分ける先をパーティションとする. しかしながら, 各パーティションについて, 入力データ量に差がある時に, データ入力からデータ処理完了までのレイテンシーに偏りが発生する. したがって, 偏りを解消することが課題である.

3. ダイナミック・パーティション手法

システムは, Broker 1 台, Producer 4 台, Consumer 4 台と提案手法を実装した Controller 1 台で構成する(図 1).

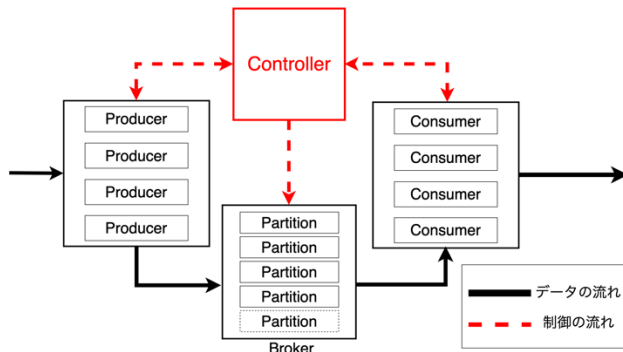


図1 システム構成図

パーティション制御について以下に述べる. パーティション毎のレイテンシーは, パーティション内の毎秒のレコードレイテンシーの平均で算出する. 偏りの検知には, レイテンシーが最も大きいパーティションの偏りを算出し, しきい値で検出する. 偏りを検知した際は, レイテンシーが最も小さいパーティションの, Consumer リソースを開放し, 同じデータグループのパーティションが持つ Consumer リソースを共有する. また, しきい値を超えたパーティションの処理リソースとして, 新しくパーティションを拡張し先程開放された Consumer リソースを割り当てる.

4. システム実験

4.1 実験条件

投入するデータグループの種別は, A, B, C の3種類とし, それぞれのデータサイズを 652 Byte, 675 Byte, 7,260 Byte

とした. データの平均投入量は, それぞれ 126 (msg/sec), 123 (msg/sec), 507 (msg/sec)とした. ストリーム処理のオープンソースソフトウェアである Apache Kafka を用いる. Kafka Broker は, 初期パーティション数 4 のトピックを 1 つ作成した. パーティション制御のしきい値は 30%とした.

4.2 実験結果

図 2 は, パーティション毎のレイテンシーを示したものである. 時間軸の 10 秒を境目に左右を比較すると, Partition1-3 のレイテンシーを基準とした時, Partition4 は Partition1-3 の4倍のデータ量があるにも関わらず, レイテンシーがおおよそ 4 倍から 2 倍の半分まで低下した. 時間軸の 10 秒付近では Partition1-2 のレイテンシーが一時的に高くなる原因は, 提案手法によるリソースの再計算と再割当てによるものである.

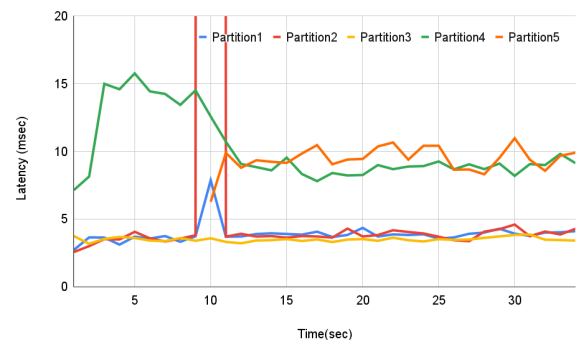


図2 パーティション毎のレイテンシー

5. 今後の課題

今後は, パーティション制御の再帰的な実行を可能にし, ストリーム処理の偏りを継続的に改善する手法を設計する.

参考文献

- [1] 根本潤, 遠山元道, ”ゲートレス鉄道サービスに向けた GPS 位置情報を用いた乗車区間判定方式の評価,” 情報処理学会論文誌データベース (TOD), Vol. 13, No. 1, pp. 33-44, 2020.
- [2] 杉浦健人, 石川佳治, ”並列ストリーム処理システムにおける DB を用いた内部状態の共有手法,” データ工学と情報マネジメントに関するフォーラム (DEIM), I8-1, 2020.
- [3] 一瀬絢衣, 竹房あつ子, 中田秀基, 小口正人, ”ビッグデータ処理基盤 Apache Spark のストリーミング機能を利用したセンサデータ解析フレームワークの検討,” マルチメディア, 分散協調とモバイルシンポジウム (DICOMO) 論文集, pp.1583-1587, 2017.