社会センサデータ生成・共有基盤における データ生成時間短縮のための社会センサ実行時刻決定手法

愛甲 善之助[1] 義久 智樹[2] 原 隆浩[1]

概要：近年，携帯型端末の普及により，SNS（Social Networking Service）が広く利用されている．筆者らの研究グループでは，SNS 等（外部データソース）に投稿された短文や写真の解析プログラムと，解析結果のデータ（社会センサデータ）を関連付けて共有するためのプラットフォーム，S^3 システムを研究開発してきた．S^3 システムにおいて，外部データソースにデータを要求してから社会センサデータの生成を完了するまでにかかる時間（周回内社会センサデータ生成時間）が短いほど，リアルタイム性の高い社会センサデータを生成できる．しかし，各社会センサの実行間隔が短すぎると，同時に複数の社会センサが実行されることになって社会センサデータの生成にかかる時間が長くなる．そこで本研究では，効率的に社会センサを実行するための実行時刻決定手法を提案する．提案手法では，各社会センサの社会センサ実行間隔に上限を設け，後の周回で実行しても上限内に実行できる社会センサの実行を中止する．評価の結果，社会センサ実行間隔と周回内社会センサデータ生成時間を短縮できることを確認した．

1. はじめに

近年，Twitter[*1]や Instagram[*2] といった SNS（Social Networking Service）から大量のデータが発生しており，それらを活用する動きが高まっている [1], [2]．これらのサービスにおけるユーザの投稿を解析することで，実社会を反映するデータを得られる．本研究では，SNS への投稿等（外部データソース）を解析して実社会のデータを取得することを，**社会センサ**によって社会をセンシングすると捉え，取得したデータを**社会センサデータ**と呼ぶ．例えば，Sakaki らは，Twitter のツイートに対して教師あり機械学習手法の SVM（Support Vector Machine）を用いて解析することで，地震に関するキーワードを含むツイートから地震の発生を検出する手法を提案している [8]．この解析結果は，地震の発生した地域と時刻を示す社会センサデータと捉えられる．他にも，社会センサデータとみなせるデータを生成する様々な研究が行われている [6], [7], [11]．これまでに筆者らの研究グループでは，社会センサデータの生成，共有を目的としたプラットフォームである，S^3 システム (Social Sensor Sharing System）を構築してきた [12]．

S^3 システムでは，データフロー制御機構が外部データソースからデータを受信し，各社会センサに送信する．データフロー制御機構が，外部データソースに繰り返してプル型でデータを受信するために，データ要求を送信する場合がある．本研究では，データフロー制御機構が外部データソースに対して，データを要求してから次に要求するまでを周回と呼ぶ．この場合，データフロー制御機構は，周回開始時刻になるたびに外部データソースにデータを要求して取得し，各社会センサの実行と同時にデータを送信する．実行された社会センサは，送信されたデータを受信して社会センサデータの生成を開始し，生成した社会センサデータを，S^3 システム内の社会センサデータベースに格納する（図 1）．社会センサを実行する間隔（社会センサ実行間隔）が短いほど，社会センサデータを生成する機会が増え，利便性の向上につながる．例えば，地震を検知する社会センサの社会センサ実行間隔が短いほど，頻繁に地震の検知を試みる．また，データフロー制御機構が外部データソースにデータを要求してから，社会センサデータの生成を完了するまでにかかる時間（周回内社会センサデータ生成時間）が短いほど，リアルタイム性の高い社会センサデータを生成できる．例えば，SNS におけるホットトピックを抽出して，その変遷を確認する社会センサにおいて，その変化が起こってから，社会センサデータとして出力されるまでの時間が短いほど，最近のホットトピックを抽出できる．これらのことから，社会センサ実行間隔と周回内社会センサデータ生成時間が短いことが望ましい．

S^3 システムでは，社会センサの実行を開始する時刻（社

1 大阪大学大学院情報科学研究科
2 大阪大学サイバーメディアセンター
*1 https://dev.twitter.com/
*2 https://www.instagram.com/

© 2019 Information Processing Society of Japan

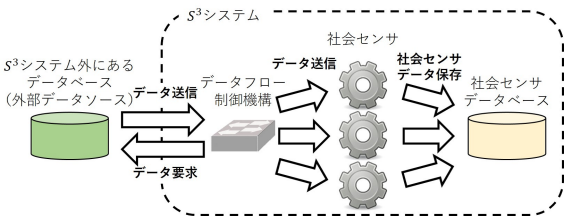

図 1 外部データソースを利用した社会センサデータ生成のイメージ

会センサ実行時刻）が早すぎると，同時に複数の社会センサが実行されることが多くなる．同時に複数の社会センサが実行されると，S^3 システムの処理負荷が大きくなり，社会センサデータの生成にかかる時間が長くなる．一方，社会センサ実行時刻が遅すぎると，社会センサ実行間隔が長くなる．S^3 システムにおいて，社会センサ実行時刻を適切に決定することで，社会センサ実行間隔や周回内社会センサデータ生成時間を短縮して，効率的に社会センサデータを生成できる．

そこで本研究では，効率的に社会センサを実行するための，社会センサ実行時刻決定手法を提案する．提案手法では，社会センサ実行間隔および周回内社会センサデータ生成時間を短縮するために，各社会センサの社会センサ実行間隔に上限を設け，後の周回で実行しても上限内に実行できる社会センサの実行を中止する．これにより，他の社会センサの社会センサ実行間隔，および周回内社会センサデータ生成時間を短縮する．さらに，同時に複数の社会センサを実行しないように実行時刻を決定する．本研究では，これまでに研究開発してきた S^3 システムを拡張し，提案手法を実装した．拡張した S^3 システムを用いて，単純に一定間隔で社会センサを実行する手法と比較評価した結果，提案手法を用いることで，社会センサ実行間隔と周回内社会センサデータ生成時間を短縮できることを確認した．

以下では，第 2 章で関連研究を紹介し，第 3 章で筆者らの研究グループでこれまでに研究開発してきた S^3 システムについて説明する．第 4 章で提案手法について説明し，第 5 章では提案手法に関する評価と議論を行う．最後に，第 6 章で本研究のまとめと今後の課題について述べる．

2. 関連研究

2.1 データ取得スケジューリング

様々な Web アプリケーションの普及にともない，ネットワークを介してデータベース内のデータを問い合わせる Web アプリケーションが増加している．多くのアプリケーションへ素早く応答するため，データベースへの問合せ時刻やデータの取得時刻を管理して，データの取得順序を決定するスケジューリングの研究が行われている [3], [4]．EDEF-LVF では，問合せを行った Web アプリケーションが，指定した時刻までに該当するデータを送信するために，指定時刻が近い問合せから順に処理する．データ取得の順番を最適化させることで，指定時刻までにデータを送信できる確率を向上させている [4]．RT-IFTTT では，各 IoT センサと登録された処理（アプレット）の状態から，IoT センサへの問合せ時刻を動的に決定している [3]．RT-IFTTT では，アプレットが動作している計算機の処理負荷を考慮していない．そのため，アプレットのデータ生成頻度の増加して処理負荷が大きくなると，各アプレットがデータを生成するための時間が長くなり，リアルタイム性が損なわれるといった問題が発生する．

2.2 プロセススケジューリング

ユーザが登録したプログラムを適切に決定するシステムにおいて，プロセスの実行時刻を最適化させる研究が行われている [5], [9]．文献 [5] では，リアルタイムシステムにおける処理（タスク）の処理を効率的に行うことで，電力消費量を抑えることを目的としている．タスクは複数のサブタスクで構成されており，タスクが次に処理するサブタスクを確率的に予測する．タスクに割り当てる処理資源を抑えつつ，デッドライン（システムがプロセスの処理を完了させるまでの期限）までにタスクを完了させるように，タスクの実行時刻を決定している．また，文献 [9] では，タスク処理を完了させるために必要な時間と処理資源を予測して，複数のタスクがそれぞれのデッドラインまでに処理を完了できるように，タスクへの処理資源の割り当てと実行時刻を決定している．これらの研究では，デッドラインまでにタスク処理を完了することを目的としており，タスクの実行回数やリアルタイム性を考慮していなかった．

3. S^3 システムのデータフロー制御機構の設計

本章では，S^3 システムにおける社会センサデータのデータフロー制御機構の設計について述べる．

3.1 S^3 システムについて

S^3 システムは，社会センサデータの生成および共有を目的として設計，実装されたシステムである．まずはじめに，S^3 システムの概要と，改善すべき課題について述べる．

3.1.1 システムへの入力とデータの使用方法

S^3 システムは，ユーザが S^3 システムにアップロードしたファイルをもとに，S^3 システム内で動作する社会センサデータ生成プログラム（SSCP : Social Sensor Creation Program）を作成する．SSCP 作成のイメージを図 3 に示す．アップロードするファイルはそれぞれ，SNS のデータを解析するコードが記述されたファイル（SSFD : Social Sensor Function Description），出力される社会センサデータの型定義ファイル（SSTD : Social Sensor Type Definition），出力される社会センサデータを社会センサデータベースに格納する際の設定を記述したファイル（SSOC : Social Sensor Output Configuration）の 3 つのファイルで

© 2019 Information Processing Society of Japan

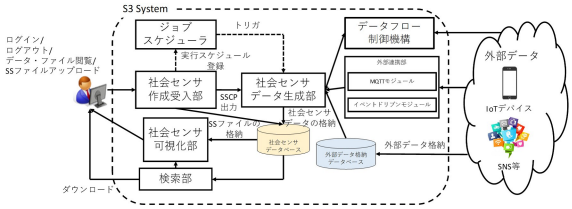

図 2　S^3 システムの構成図

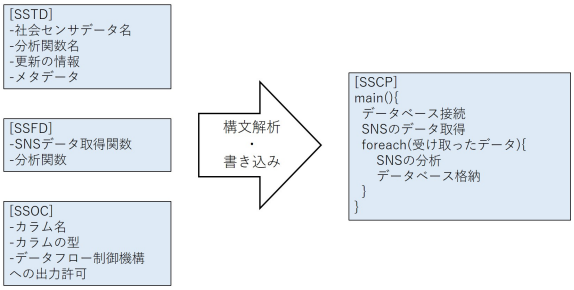

図 3　従来の SSCP の作成イメージ

ある．社会センサが実行されると，SSTD で指定された入力データを取得する関数が呼び出される．この関数は，ユーザが SSFD に記述するか，S^3 システム内であらかじめ定義されている．取得したデータを解析して，社会センサデータが生成される．生成された社会センサデータは，SSOC に定められた設定に基づいてプラットフォーム内の社会センサデータベースに格納され，ユーザ間で共有できる．共有するユーザやグループを限定する場合，SSOC に記述することで公開範囲を定められる．生成された社会センサデータは，Web インターフェースを介して，社会センサデータベースからダウンロードできる [10]．

3.2　データフロー制御機構

S^3 システムは，データフロー制御機構というモジュールを備えている．データフロー制御機構では，外部データソースからデータを受信して，社会センサに送信したり，ある社会センサが生成した社会センサデータを，別の社会センサへの入力として利用できる．以下に動作の手順を説明する．

S^3 システムは，アップロードされた社会センサがデータフロー制御機構を利用する場合，作成した社会センサを，データフロー制御機構にデータの送信先として登録する．登録の際，データフロー制御機構を利用する社会センサが受信するデータに関する情報（受信する社会センサデータを生成する社会センサの識別子や，外部データソースの名前）を，データコンテナを介して，データフロー制御機構に送信する．データコンテナは，S^3 システム内でデータを送受信するためのデータ構造である．登録の要求を受信したデータフロー制御機構は，受信するデータに関する情報に，送信先となる社会センサの識別子を対応付けて記憶し，外部データソースと社会センサの利用関係を管理する．

3.2.1　構成要素

データフロー制御機構は以下の要素で構成される．

社会センサデータルーティングテーブル

社会センサデータルーティングテーブルには，外部データソースを利用する社会センサが登録される．データフロー制御機構は，社会センサデータルーティングテーブル上の登録情報を参照し，外部データソースより取得されたデータの送信先の決定を行う．

ポーリングスケジューラ

ポーリングスケジューラは，各社会センサのデータ生成に関する情報を保持する．社会センサデータルーティングテーブルに登録された，社会センサの登録情報を参照し，外部データソースにデータを要求する時刻と，各社会センサを実行する順番を決定する．

データ取得器

データ取得器は，ポーリングスケジューラが管理する，各社会センサのデータ取得時刻を参照し，外部データソースへ問合せを行い，データを取得する．取得したデータは，直ちにデータフロー制御機構へ送信される．

社会センサデータコンテナ

社会センサデータコンテナは，データフロー制御機構と社会センサとの間で送受信される，外部データソースから取得したデータを格納するデータコンテナである．社会センサデータコンテナには，外部データソースから取得したデータの他，データフロー制御機構への要求，および要求に必要な情報が含まれる．データフロー制御機構は，これらの情報を参照して，社会センサの要求ごとに適した処理を行う．

3.2.2　プル型の外部データソースを利用する場合

プル型でデータを受信する場合，まずデータフロー制御機構は外部データソースに，データを受信できるようにするための登録を行う．その後，外部データソースに格納されたデータを取得するために，データフロー制御機構は外部データソースにデータの要求を行う．プル型でデータを受信する場合に，各社会センサがデータフロー制御機構を介して，外部データソースからデータを受信する処理の流れを以下に示す．

登録

各社会センサの登録時に，SSTD に外部データソースからプル型でデータを受信することが記述されていると，S^3 システムは，該当する社会センサのデータフロー制御機構への登録処理を行い，外部データソースから取得したデータの送信先として記憶する．また，利用される外部データソースから，データを受信する社会センサがなかった場合には，外部データソースにデータ受信の登録処理を行う．SSTD に指定された外

© 2019 Information Processing Society of Japan

部データソースが存在しない場合，登録の要求は破棄される．

外部データソースへのデータ要求

データフロー制御機構は，S^3 システムによって定められた周回開始時刻になると，外部データソースにデータを要求してデータを取得する．従来のデータフロー制御機構では，周回間隔を固定して周回開始時刻を決定していたが，本研究では，3.4 節で定める方法により周回開始時刻を決定する．

外部データソースからのデータ取得

データフロー制御機構が外部データソースに対し，データの要求を出したのち，外部データソースからデータが送信され，データフロー制御機構はそのデータを取得する．データフロー制御機構は，送信先社会センサ登録時に記憶した内容から，取得したデータを利用する社会センサが存在するか確認する．該当する社会センサが存在する場合，データフロー制御機構は外部データソースから取得したデータを，送信する社会センサと対応付けて一時的に保存する．

社会センサへの送信

データフロー制御機構は，S^3 システムによって定められた社会センサ実行時刻になると，社会センサを実行して外部データソースから受信したデータを送信する．従来のデータフロー制御機構では，外部データソースからデータを受信すると，すぐに各社会センサに送信していたが，本研究では，3.4 節および 3.5 節で定める方法により社会センサ実行時刻を決定する．外部データソースからデータを取得後，送信先の社会センサが動作していた場合，保存した外部データソースから取得したデータを，該当する社会センサに送信する．データを送信した後，データフロー制御機構は送信したデータを，データフロー制御機構から消去する．送信先の社会センサが動作していなかった場合，送信要求を発行した社会センサに対し，次項で説明する社会センサの登録削除処理を行う．

削除

プッシュ型のデータ受信の場合と同じく，各社会センサが作成されるときに設定された，社会センサの動作終了時刻を迎えたときや，入力データとして利用していた社会センサや外部データソースが動作を終了したとき，その他，社会センサが異常終了していたときに，データフロー制御機構は動作を終了した社会センサの登録を削除する．データフロー制御機構が登録した外部データソースを，削除する社会センサしか利用していなかった場合には，外部データソースからのデータ受信に関する登録も削除する．

3.3 設計方針

本研究では，効率的に社会センサを実行するために 2 つのアプローチを採用する．まず，同時に複数の社会センサを実行しないように実行時刻を決定する（単一実行）．また，各社会センサの社会センサ実行間隔に上限を設け，後の周回で実行しても上限内に実行できる社会センサの実行を中止する（限界社会センサ実行間隔）．これらのアプローチについて，次節から順に説明する．

3.4 単一実行

3.4.1 従来の実行時刻

データフロー制御機構が外部データソースからデータを受信して各社会センサに送信する際，従来手法では，ある社会センサへの送信が完了するとすぐに次の社会センサへデータを送信していた．この手法では，ある社会センサが実行されてからすぐに次の社会センサが実行されるため，同時に複数の社会センサが実行される可能性が非常に高い．このため，S^3 システムの処理負荷が大きくなり，社会センサデータの生成にかかる時間が長くなる．

3.4.2 単一実行の方法

同時に複数の社会センサを実行しないようにするために，提案手法では単一実行のアプローチを採用する．データフロー制御機構がデータを取得してから，各社会センサがデータを生成するまでに，S^3 システムには 3 つの手順がある．まず，データフロー制御機構が外部データソースに対してデータの要求を行う．次に，データフロー制御機構が外部データソースからデータを受信する．最後に，データフロー制御機構が社会センサにデータを送信する．データフロー制御機構から社会センサがデータを受信すると，社会センサデータの生成を開始し，生成が終了次第，社会センサデータの生成が完了する．上記のデータフロー制御機構がデータを取得して，社会センサがデータを生成するまでの動作一連の動作を周回と表現する．i 回目の周回を C_i とし，i 回目の周回開始時刻を cs_i で表す．C_i が開始するとすぐにデータフロー制御機構は，外部データソースに対するデータの取得を開始する．データフロー制御機構が外部データソースからデータの取得を完了する時刻を，周回内データ取得時刻と呼び，cd_i で示す．データフロー制御機構が外部データソースからデータを取得すると，N 個の社会センサの実行と同時にデータを送信する．社会センサ n が社会センサデータの生成を開始する時刻を，周回内社会センサ実行時刻と呼び，$css_{n,i}$ で示す．社会センサ n が社会センサデータの生成を終了する時刻を，周回内社会センサデータ生成完了時刻と呼び $csf_{n,i}$ で示す．i 回目の周期の中で社会センサデータの生成完了が，最も遅い社会センサの周回内社会センサデータ生成完了時刻を，周回終了時刻と呼び，cf_i で示す．社会センサ i 回目の周回が，開始される前の周回間隔 CD_i は下記となる．

© 2019 Information Processing Society of Japan

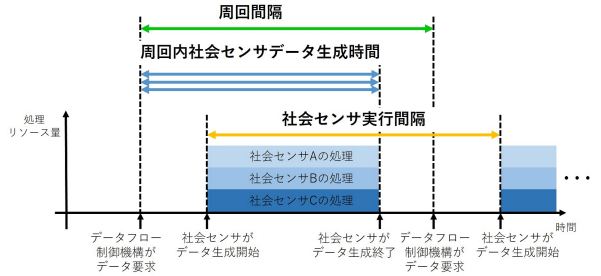

図 4　S^3 システムの処理リソース割り当てのイメージ

$$CD_i = cs_i - cs_{i-1} \tag{1}$$

また，社会センサ i の i 回目の周回内社会センサデータ生成時間 $CS_{n,i}$ は下記となる．

$$CS_i = css_{n,i} - csf_{n,i} \tag{2}$$

本研究では，これらの値を短縮することを目的としている．

複数の社会センサが動作しないようにするためには，まず，周回内で複数の社会センサが動作しないように下記を満たす必要がある．ただし，n が大きくなる順に社会センサを実行すると想定する．

$$css_{n,i} = csf_{n-1,i} \tag{3}$$

データフロー制御機構は，$css_{n,i}$ および $csf_{n-1,i}$ を決定できるため，データフロー制御機構の制御によりこれを実現できる．

また，周回間で複数の社会センサが実行されないようにするために，$i-1$ 回目の周回終了時刻よりも，i 回目の周回の最も早い周回内社会センサ実行時刻である cd_i が遅くなる必要がある．すなわち，次式を成立させる．

$$cf_{i-1} < cd_i \tag{4}$$

しかし，データフロー制御機構が，外部データソースからデータを取得するのにかかる時間 $cd_i - cs_i$ は，通信状況やデータフロー制御機構の負荷によって変化するため，計算で求めることが困難である．そこで提案手法では，この値を過去の周回の平均値で与える．

3.5　限界社会センサ実行間隔の考慮

3.5.1　限界社会センサ実行間隔

データフロー制御機構は，外部データソースから取得したデータを利用する社会センサに，周回ごとにデータを送信する．一般に，社会センサ実行間隔が早いほど，社会センサデータの利便性が高まると考えられるが，社会センサ実行間隔が早くても，利便性が大きく高まらない社会センサがある．例えば，トレンドを検知する社会センサの場合，1 分ごとにトレンド（社会センサデータ）を生成するといったことが考えられる．このように，社会センサの作成者が設定する，最長の社会センサ実行間隔を限界社会センサ実行間隔と呼ぶ．周回間隔に比べて，限界社会センサ実行間隔が長い場合，周回間隔ごとに社会センサを実行しなくても利便性が大きく損なわれない．

表 1　社会センサデータコンテナに含まれる情報

フィールド	内容
動作要求名	データフロー制御部に要求する動作名を記述（register, devote, reuse, delete のいずれか）
送信社会センサ ID	このコンテナを送信した社会センサの ID を記述
要求先社会センサ ID	このコンテナを送信した社会センサの要求先社会センサの ID を記述
データ名	データ本体の名前を記述
データ本体	データ本体を記述
データ譲渡可否情報	生成したデータのほかの社会センサへの送信を許可するか否かを記述
外部データソース名	データを取得する外部データソースの名前を記述
限界社会センサ実行間隔	社会センサの最大の実行間隔を記述

3.5.2　限界社会センサ実行間隔を考慮した実行時刻

提案手法では，後の周回で実行しても限界社会センサ実行間隔を越えない場合，後の周回で該当する社会センサを実行する．ある周回で社会センサを実行しないことで，他の社会センサ生成時間を短縮できる．提案手法では，各社会センサの実行時間を過去の平均値を用いて予測し，社会センサの実行を中止した場合の，次の周回における周回内社会センサ実行時間を算出する．この周回内社会センサ実行時刻と，直近の周回内社会センサ実行時刻との差が，限界社会センサ実行間隔より短くなる場合に，社会センサの実行を中止し，次の周回まで遅らせる．

3.6　データフロー制御機構の拡張方針

表 1 で示した動作要求名に対して，データフロー制御部は動作を行う．動作要求名が “register” の場合，再利用する社会センサの登録を行う．“devote” の場合，社会センサが生成した社会センサデータの受信，“reuse” の場合，データフロー制御部が保持している社会センサデータの送信を行う．動作要求名が “delete” の場合，該当する社会センサの登録内容の削除をそれぞれ行う．

一例として，外部データソースからデータを受信する場合の処理手順を図 5 に示す．データ取得器が，ポーリングスケジューラを参照し，外部データソースからデータ取得を行う．ポーリングスケジューラによって決定されたデータ取得時刻に，それぞれの外部データソースに対応した，データ取得器を動作させる．データ取得器は，外部データソースに問い合わせ，必要なデータを取得する．その後，動作要求名を “devote”，受信社会センサ ID を該当するの社会センサ ID の組，データ名を “json”，データ本体を取得された JSON データに設定して，社会センサデータコンテナをデータフロー制御機構に送信する．

データフロー制御機構が，外部データソースから取得した情報を持つ，社会センサデータコンテナを受信すると，データを一時的に保持するための，受信処理を開始する．

© 2019 Information Processing Society of Japan

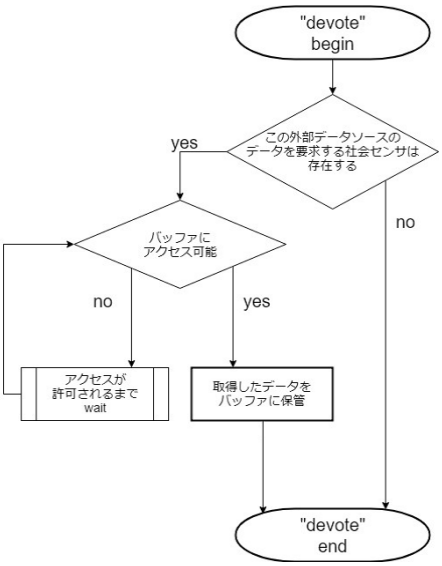

図 5　受信要求のフローチャート

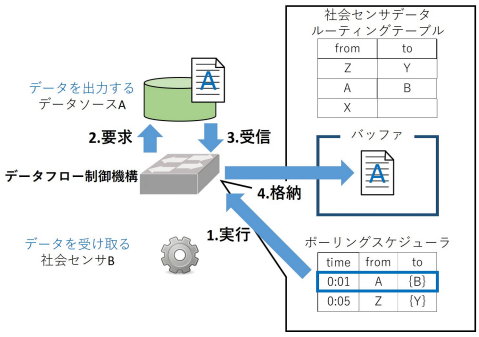

図 6　受信の要求に応じたデータフロー制御機構の動作イメージ

まず，社会センサデータルーティングテーブルを参照し，受信要求を発行した社会センサに対応する所定のバッファ領域に，一時的に保持する．このとき，データ保持領域では排他制御が行われるため，この操作が終了するまで保持するデータの読み書きは行われない．

データフロー制御機構が，外部データソースからデータを受信し，バッファにデータを保存する際の動作イメージを図 6 に示す．図 6 では，社会センサ A が社会センサデータを生成するとデータフロー制御機構に送信する．データフロー制御機構は社会センサデータルーティングテーブルを確認し，再利用する社会センサがあれば，所定のバッファ領域に格納する．

4.　データフロー制御機構の実装

第 3 章で述べた設計に基づき，S^3 システム上にデータフロー制御機構を実装した．

4.1　実装環境

社会センサルーティングテーブルおよび社会センサデータバッファの構築，社会センサからの要求の処理ロジックなど，データフロー制御部は Java1.8.0_66 で実装した．

表 2　クラス内変数と対応する
社会センサデータコンテナのフィールド名

変数名	フィールド
OperationName	動作要求名
ReuseSSID	送信社会センサ ID
RequiredSSID	要求先社会センサ ID
DataName	データ名
Data	データ本体
PermitFlow	データ譲渡可否情報

社会センサの新規登録，社会センサの確認，社会センサデータの取得などの機能は，先行研究のものをそのまま利用した．ユーザがシステムに入力するファイルも先行研究と同様に，SSFD は Java クラスファイルもしくは JAR ファイル，SSTD および SSOC は XML ファイルとしている．

4.2　データフロー制御部の実装

本節では，データフロー制御部の各構成要素の実装方法について説明する．

4.2.1　社会センサデータルーティングテーブルと社会センサデータバッファの実装

社会センサデータの要求先と送信先，送信するデータの種類および社会センサデータを，Java の連想配列を用いて格納した．社会センサ ID をキー値とし，その社会センサが生成したデータを再利用する社会センサを連想配列でもち，さらに再利用するデータ名とデータ本体を保持する．

4.2.2　社会センサデータコンテナの実装

社会センサとデータフロー制御部の間で送受信させる社会センサデータコンテナは，Java のクラスとして定義した．表 2 に示すように，各フィールドに対応する変数を設定し，社会センサデータコンテナに対して読み書きするためのメソッドを実装した．データフロー制御部は社会センサデータコンテナを用いて，社会センサデータの送受信を行う．

5.　評価

本章では，提案手法の評価結果について示す．

5.1　実験環境

社会センサ実行間隔や周回内社会センサデータ生成時間を評価するために，実機による計測を行った．S^3 システムをインストールする計算機として，CPU：Core i7@3.5 GHz，コア数：8 個（ハイパースレッディング）、メモリー：32GB の計算機を用いた．ただし，実験で使用するコアはそのうちの 1 つのみに限定した．Twitter を外部データソースとして利用し，ツイートを形態素解析およびクラスタリングする社会センサを用いた．

評価値として，平均社会センサ実行間隔と平均周回内社

© 2019 Information Processing Society of Japan

会センサデータ生成時間，限界社会センサ実行間隔超過率を用いた．限界社会センサ実行間隔超過率は，全社会センサの実行数に対する，各社会センサに設定された，限界社会センサ実行間隔を超えた社会センサ実行の割合である．

5.2 限界社会センサ実行間隔を一律に変化させた場合

外部データソースの数を 10 個，各外部データソースを利用する社会センサの数が 10 個で，S^3 システムに 100 個の社会センサを登録させた状態で，すべての社会センサの限界社会センサ実行間隔を変更させて平均社会センサ実行間隔，平均社会センサデータ生成時間，限界社会センサ実行間隔超過率を計測した．従来手法として，周回間隔を固定間隔（LC = 0〜120 [s]）で与える手法と比較した．ただし，固定間隔は社会センサの限界社会センサ実行間隔を基準にして与え，半分の時間でデータ取得を行う固定間隔（LC/2 [s]），同じ時間でデータ取得を行う固定間隔（LC [s]），倍の時間でデータ取得を行う固定間隔（2LC [s]）を用いた．固定間隔が 0 [s] の場合は，データフロー制御機構が外部データソースにデータを要求するとすぐに再びデータを要求する．結果を，図 8 に示す．いずれの手法においても，平均社会センサデータ生成時間が収束しない場合がある．固定間隔（2LC [s]）の場合，提案手法よりも短い限界社会センサ実行間隔において，平均社会センサデータ生成時間が収束しているが，平均社会センサ実行間隔は提案手法よりも，最大で 100 [s] 近く長くなっている．また，固定間隔（2LC [s]）の場合，限界社会センサ実行間隔超過率は常に 1 になっている．提案手法では，固定間隔（2LC [s]）以外のどの手法よりも早く，平均社会センサデータ生成時間を収束している．さらに，図 8(a)，(b) の提案手法のグラフが，ある限界社会センサ実行間隔から，平均社会センサ実行間隔と平均社会センサデータ生成時間が大きく変わっていないことから，ほぼ一定の間隔で社会センサデータを生成できることがわかった．

5.3 限界社会センサ実行間隔の異なる社会センサを登録した場合

前節の評価では，S^3 システムに登録されているすべての社会センサの，限界社会センサ実行間隔が同じ場合の評価を行ったが，本節では異なる場合の一例として，各外部データソースごとに限界社会センサ実行間隔が異なる，2 種類の社会センサ各 50 個を用いて評価実験を行った．1 種類は 15 [s] の社会センサであり，もう 1 種類は 15, 30, 60 [s] で変化させた場合の，平均社会センサ実行間隔，平均社会センサデータ生成時間，限界社会センサ実行間隔超過率を計測した．限界社会センサ実行間隔を考慮しながら社会センサの実行を中止する場合を間引きあり，しない場合を間引きなしと表現する．結果を，図 9 に示す．横軸が変化させた方の社会センサの限界社会センサ実行間隔である．

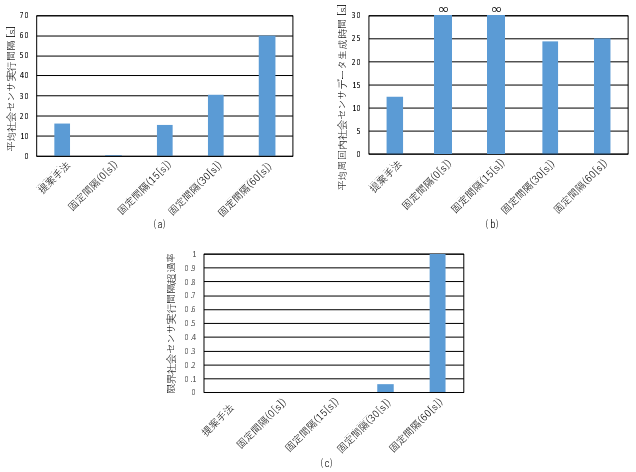

図 7 外部データソース数と登録社会センサ数を固定した場合の評価結果

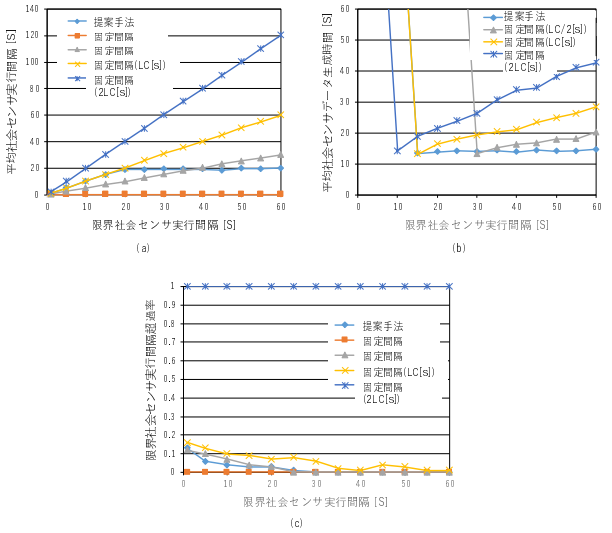

図 8 限界社会センサ実行間隔を一律に変化させた場合の評価結果

括弧内が（15 [s]）のグラフは，限界社会センサ実行間隔が 15 [s] の社会センサの評価値，（可変）のグラフは，限界社会センサ実行間隔を変化させた社会センサの評価値である．間引きなし（15 [s]）と間引きなし（可変）では，平均社会センサ実行間隔と平均社会センサデータ生成時間に，大きな違いがないことがわかる．しかし，間引きなし（15s）の限界社会センサ実行間隔超過率は間引きなし（可変）に比べ，高くなっている．これは，限界社会センサ実行間隔の異なる社会センサを同じ周回に行うことで，S^3 システムの負荷が大きくなったためである．間引きあり（15 [s]）の社会センサでは，平均社会センサ実行間隔と平均社会センサデータ生成時間が短く，間引きあり（可変）の社会センサでは，平均社会センサ実行間隔と平均社会センサデータ生成時間が長くなる．また，間引きあり（15 [s]）の限界社会センサ実行間隔超過率は減少傾向にある．これは，間引

© 2019 Information Processing Society of Japan

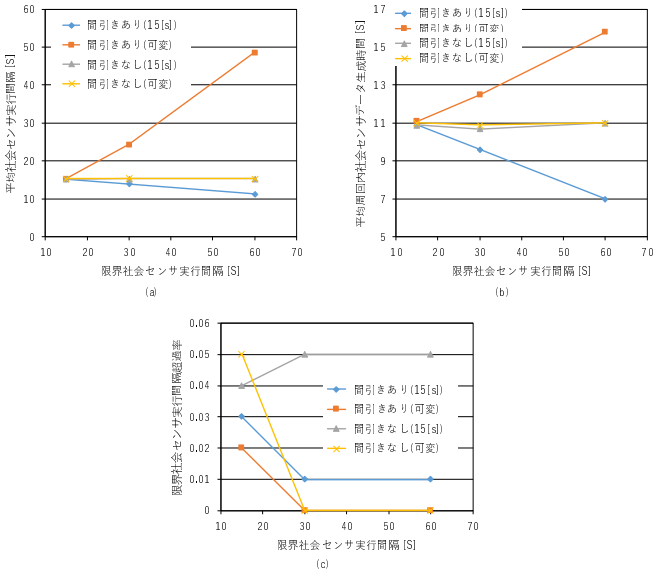

図 9 限界社会センサ実行間隔の異なる社会センサを登録した場合の評価結果

きあり（可変）の社会センサの実行回数を減らしたことによって，間引きあり（15 [s]）の社会センサの実行回数が増えたためである．

6. まとめ

本研究では，S^3 システムにおいて社会センサ間でデータ送受信を行うための，データフロー制御機構の設計および実装を行った．提案したデータフロー制御機構では，データフロー制御部が社会センサ間のフロー制御情報を管理しプロセス間通信によって社会センサ間の社会センサデータのデータフロー制御を行う．提案するフロー制御機構を用いることで，社会センサが生成される毎に別の社会センサへの生成されたデータの入力を自動的に行える．比較評価の結果，S^3 システムにおいてデータフロー制御機構を用いることで，社会センサデータの再利用に適したデータフロー制御機構を提供できることが分かった．

今後の課題として，複数のデータ要求先を指定する機能やデータの送信間隔を調整する機能の設計実装を考えている．

謝辞 本研究の一部は，公益財団法人 I-O DATA 財団および文部科学省科学研究費補助金（JP18K11316）の研究助成によるものである．ここに記して謝意を表す．

参考文献

[1] Chen, M., Mao, S. and Liu, Y.: Big data: A survey, *Mobile Networks and Applications*, Vol. 19, No. 2, pp. 171–209 (2014).

[2] Coronado, M. and Iglesias, C. A.: Task Automation Services: Automation for the Masses, *IEEE Internet Computing*, Vol. 20, No. 1, pp. 52–58 (2016).

[3] Heo, S., Song, S., Kim, J. and Kim, H.: RT-IFTTT: Real-Time IoT Framework with Trigger Condition-aware Flexible Polling Intervals, *Proc. of IEEE Real-Time Systems Symposium (RTSS)*, pp. 266–276 (2017).

[4] Kim, J.-E., Abdelzaher, T., Sha, L., Bar-Noy, A. and Hobbs, R.: Sporadic Decision-Centric Data Scheduling with Normally-Off Sensors, *Proc. of IEEE Real-Time Systems Symposium (RTSS)*, pp. 135–145 (2016).

[5] Nakada, T., Hatanaka, T., Ueki, H., Hayashikoshi, M., Shimizu, T. and Nakamura, H.: An Energy-Efficient Task Scheduling for Near-Realtime Systems with Execution Time Variation, *IEICE Transactions on Information and Systems*, Vol. 100, No. 10, pp. 2493–2504 (2017).

[6] Nguyen, D. T. and Jung, J. E.: Real-time event detection for online behavioral analysis of big social data, *Future Generation Computer Systems*, Vol. 66, pp. 137–145 (2017).

[7] Ouyang, Y., Guo, B., Zhang, J., Yu, Z. and Zhou, X.: SentiStory: multi-grained sentiment analysis and event summarization with crowdsourced social media data, *Personal and Ubiquitous Computing*, Vol. 21, No. 1, pp. 97–111 (2017).

[8] Sakaki, T., Okazaki, M. and Matsuo, Y.: Earthquake Shakes Twitter Users: Real-Time Event Detection by Social Sensors, *Proc. of International Conference on World Wide Web*, pp. 851–860 (2010).

[9] Zitterell, T. and Scholl, C.: A Probabilistic and Energy-Efficient Scheduling Approach for Online Application in Real-Time Systems, *Proc. of the 47th Design Automation Conference*, pp. 42–47 (2010).

[10] 愛甲善之助，谷山雄基，横山正浩，中嶋奎介，義久智樹，原隆浩ほか：社会センサデータ生成・共有基盤におけるデータフロー制御機構，マルチメディア，分散協調とモバイルシンポジウム 2017 論文集，Vol. 2017, pp. 1026–1033 (2017).

[11] 杉谷卓哉，白川真澄，原隆浩，西尾章治郎ほか：教師あり機械学習を用いたツイート投稿時のユーザ位置推定手法，情報処理学会研究報告データベースシステム (DBS)，Vol. 2013, No. 26, pp. 1–8 (2013).

[12] 中嶋奎介，谷山雄基，横山正浩，義久智樹，原隆浩，西尾章治郎ほか：社会センサデータ生成・共有基盤システムの設計と実装，DICOMO2016 シンポジウム論文集，Vol. 2016, pp. 1215–1222 (2016).

© 2019 Information Processing Society of Japan