

災害現場での複数台ロボット制御に向けた ROS の非同期分散処理と一対一通信拡張の設計と実装

小泉亘平^{†1} 菊池敬裕^{†1} 寒竹俊之^{†1} 菅谷みどり^{†1}

アブストラクト: 近年, 震災などの自然災害時には安全かつ迅速な人命救助においては, 通信制御を考慮した複数台の自律移動型ロボットの運用が求められている. こうした複数台ロボットの通信制御では, 通信の信頼性のみならず, 効率的なデータ共有, スケーラビリティの要求を満たす設計が重要である. 本研究では, これらを実現することを目的とし, Publisher/Subscriber モデルによりデータ共有と非同期通信を基礎としつつ, 災害現場で必要となる特定のノードに対する再配置通信を可能とする仕組みを実現するシステムの設計と実装を提案する. 実現のため, 予備的な調査をもとに ROS をベースとした Pub/Sub によるシステムの設計, 実装を行った. 本論文では設計, 実装, 評価の結果について述べる.

1. はじめに

震災などの自然災害時には安全かつ迅速な人命救助が求められている. ロボットを災害救助活動に活用する関心は, 震災以降高まり, 通信を通じた遠隔制御により大型, 小型ロボットを利用することで効果的な支援を行うことを目的とした研究は数多くなされている[1-4]. 特に通信を用いた制御は, ネットワーク制御システム(Networked Control Systems: NCS) とされ, センサ, コントローラ, アクチュエータをそれぞれ異なる空間に配置し, ネットワーク経由で通信を行うことでこれらの目的を達成するものであり, 省コスト性や遠隔操作性, オペレータの安全性といった利点が知られている[5].

これに対し近年, 複数台ロボットの制御を対象とした研究も提案されている [6,7,8]. 無線ネットワークによって結合された被災者発見システム[6] や, モデル予測制御[7,8]がある. これらは, 通信を制御理論に取り入れ, ロボット制御の対象が複数台になった場合でも, それが成立するモデルを提案している. しかし, 通信の信頼性や台数が増えた場合の計算負荷の分散は十分検討されていない. 負荷分散の観点ではアドホックネットワークを用いるなどの提案がなされている [9]. しかし, アドホックネットワークでは, データや情報が分散して管理されることから効率的なデータ共有や, 再計算に基づいたロボットの再配置などを行う場合には向いていない課題がある.

本研究の目的は, 災害現場で利用する複数台ロボットの支援を行う分散システムの設計, 実装を行うことである. 目的を達成するため, 災害現場において重要な, 効率的なデータ共有と収集, 再計算と再配置の指示を行うことを念頭とし, 通信の信頼性を備えた仕組みを提案する. 目的の達成にあたり, 基礎的な仕組みを, サ

ーバクライアント方式とし, 通信状態に依存した柔軟性の高い通信の実現に非同期でデータを共有する Publisher/Subscriber(以降, Pub/Sub)方式を用いるものとした. また, ロボットを通じたサーバによる情報収集, 再計算と再配置の指示はサーバが行うものとする. ことで, 効率的な探索を実現する. これは, 物理的に異なる位置に配置されているロボットが, 進捗に応じた非同期命令を受け取り, 効率的な再配置を行える仕組みである.

ROS(Robot Operating System, 以降 ROS) [7] は, 豊富なソフトウェアと, スケーラビリティを考慮したノード形式の実装, Pub/Sub モデルによる効率的な非同期分散処理を実現できることから, 本要求を満たす実装としてふさわしいと考え実装で用いるものとした. ただし, サーバ側で動的にクライアントからの情報を収集しつつ, 最適な再配置の指示を行う仕組みについては, 通常の Pub/Sub モデルを用いて一対一通信するための仕組みが存在しない. そこで本研究では Pub/Sub の共有データを, 通信ノード数に応じて柔軟に一対一通信向けに拡張するための拡張部分の設計と実装を提案する. Pub/Sub の一律通信のメリットを維持しつつ, 特定のノードのみと一対一通信する仕組みを拡張し, これらを柔軟に切り替えることで, 災害現場における分散システムの目的を達成する. 論文では, 本設計における予備調査とそれに基づく設計, 実装, 評価の結果について述べる.

本論文の構成は, 次の通りである. 2 節にて, 関連研究について述べる. 3 節にて, 提案, 4 節にて評価, 5 節にてまとめと今後の課題について述べる.

2. 関連研究

災害現場における複数台の自律移動ロボットを用いた災害救助方式として, 無線ネットワークによって結合

^{†1} 芝浦工業大学 Shibaura Institute of Technology

された被災者発見システムが提案されている[6]. これは無線ネットワークによって結合された複数台のロボットからなり, 各ロボットは被災者を検知するセンサを備えている. ネットワークの一端にはモニタ局を置き, これを通してオペレータが各ロボットにアクセスを行う. もしあるロボットが持つセンサが被災者を検知したなら, その情報が無線ネットワークを通してモニタ局に送られる. また, 被災者発見システムの基盤となる無線ネットワークとしてアドホックネットワーク(ad hoc network)[9,10]を用いている. アドホックネットワークは特に電波障害物の散在する災害現場におけるロボット間の接続性, 分散システムであることによる信頼性において被災者発見システムに適している.

先行研究[9]は, アドホックネットワークに自律的パス修復を適用している. 災害現場においてしばしば発生する通信リンクの切断を各ロボットの自律的な行動によって修復することにより災害現場におけるアドホックネットワークの信頼性をさらに高めるとされている. しかしアドホックネットワークを利用することにより, 末端での通信品質がアクセスポイント方式と比較して悪くなることや, メッセージが複数回落ちた場合にリンクが切断されたと判断し経路切り替え処理を開始しているため, 切り替え時に経路がない時間が存在し, パケットロスが生じる問題がある. また, 通信リンク回復のための時間がロボットの位置に依存してしまう問題がある. ロボット間の距離が遠いとその分通信リンク回復に時間がかかってしまう. アドホックネットワークのように, 常に通信のリンクが一定ではない場合, データ共有における効率性が低下する問題が発生する.

無線通信制約を考慮する複数台のロボットシステムのフォーメーション制御[8]では, 無線ネットワークにより通信を行う複数台のロボットシステムのフォーメーション形成を目的に, 各ロボットの運動とネットワーク構造の最適制御手法が提案されている. 本提案では, 電波などを用いた無線通信には一般に有効範囲が存在するが, その通信範囲内でネットワークの連結性を確保するためにロボットは移動を行いながらネットワーク構造を自律的に変化させていくためのモデル予測制御を用いて複数台のロボットのフォーメーション制御を行う方式を提案している. しかし, ロボット数 N に対して計算時間が指数関数オーダとなる課題がある. 災害現場が大きくなればなるほど探索ロボットの台数は増えるため先行研究では期待しているほどの探索効率を得ることができないと考えられる. 適当に処理をサーバに分担して, 分散処理を前提とした仕組みを検討することで, より効果的に利用できると考えられる.

3. 提案

3.1 課題

本研究の目的は災害現場で利用する複数台ロボット支援を行う分散システムの設計, 実装を行うことである. 災害現場においては, 効率的なデータ共有とデータ収集, 再計算と再配置の指示を行うことが重要であるが, 実現先行研究においては, データ共有の効率性, スケーラビリティに課題がある. また, これらを実現する仕組みについて, 基礎的な評価を踏まえた分散システムの設計, 実装が十分に行われていない課題がある.

本研究では, 基礎的な仕組みとして, サーバクライアント方式がふさわしいと考える. これは, データ共有の効率性上重要である. また, サーバからクライアントへの通信では, データロスなどを極力防ぐ信頼性を重要としつつ 1.ロボットの追加に対応するスケーラビリティを実現する分散型の構成を検討する. また, 2.クライアントが望ましいタイミングで非同期的にデータを共有できるようにする. サーバは分散配置されたクライアントとして配置されるロボットからの情報収集と, 再計算と再配置の指示を行うことを前提とし 3.サーバ側からの指示をロボットが個別に受け取る仕組みが必要である. これは, 物理的に異なる位置に配置されているロボットにとっては, 進捗が異なる作業に対し, 進捗に応じた個別の命令が送付されることが望ましいと考えるためである.

3.2 提案

本研究では, 3.1 に示した 3 つの要求を満たし効率的にデータを共有しつつ, スケーラビリティを実現する手法として, Pub/Sub モデルにより非同期通信を実現する ROS に着目した. ROS は課題の 1,2 の要件を満たす仕組みを提供すると考える.

一方, Pub/Sub モデルは, 3 の要件であるロボットの再配置指示などに必要となる, 特定のノードに対する通信を行う方法は考慮していない. そこで, 本研究では, Pub/Sub の一律通信のメリットを維持しつつ, 3. の要求を満たすために特定のノードのみと一対一通信する仕組みを拡張する仕組みを提案する.

3.3 探索領域の分割・探索システム設計と実装

本システムを実現するためのサーバ, クライアント, Topic の配置を表した全体のシステム構成設計図を図 1 に示す.

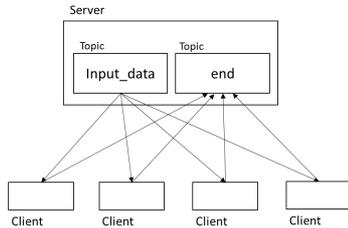


図1 サーバ, クライアント, Topic の配置を表した全体のシステム構成設計図

ここで, 情報共有のための Topic はすべてサーバに配置するものとする. 探索領域をクライアントに Publish する際は入力データ(input_data)を介して通信を行い, クライアントが探索を終了した場合には end を介してサーバと通信を行う. Topic を共通にせず, 機能ごとに分けることにより必要でないものまで Subscribe することを防ぐ.

3.4 設計と実装の課題と改善

探索領域の分割・探索方法システムの設計, 実装において 2 点の課題について述べる. 1 点目は探索領域を Publish する Topic を共通すると全てのクライアントがメッセージを Subscribe する. つまり, ブロードキャスト方式のようにすべてのクライアントにメッセージを送信した後に, クライアント内でそのメッセージが必要かどうかを判断し, 取捨選択する必要があることから, 不効率である. 2 点目はクライアントがサーバから指示された探索領域を Subscribe した後, 処理を実行すると, その時点で Subscribe 状態を維持せず, サーバとの通信をシャットダウンする. このため, サーバから探索領域の変更などの突然の通信があった場合, Subscribe することができない問題が発生する. 以上の課題の改善のためにクライアントごとに Topic を分散させるものとした. 再設計後の全体のシステム構成再設計図を図 2 に示す.

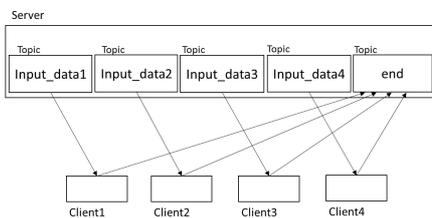


図2 全体のシステム構成再設計図

クライアントは自分に登録されている Topic が監視されたらそのまま Subscribe し, 必要なメッセージかどうか付度が不要となる. 事前の実装では Publisher と Subscriber は別々に分かれており, subprocess.call によ

り連続的に動作させた. 改善では Subscriber の中に Publisher を組み込むことで, クライアントは常に Subscribe 状態を維持し, 突然のサーバからの通信に対しても Subscribe 可能とする, という構成とした. これらの課題改善の設計により, 探索開始から探索終了を行えるものとした.

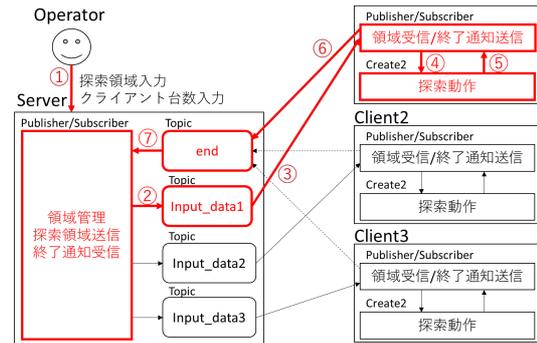


図3 Topic の分散化を行った設計図

4 評価

4.1 目的

ROS の Pub/Sub モデルを用いた複数台におけるサーバとクライアントの対一通信を可能としたミドルウェアの台数増加における性能評価を行う.

4.2 方法

クライアントに対して 100 回探索行動を行うよう, サーバから探索領域の命令を Publish する. この時台数増加するに伴い, サーバがクライアントに Publish してから Subscribe するまでの時間とクライアントが Publish してから Subscribe するまでの時間を計測し, その性能評価を実施した. 計測区間を図 13 に示す. 図 13 における計測区間 1 はサーバがクライアントに探索領域を Publish してからクライアントが Subscribe するまでの区間. 計測区間 2 はクライアントがサーバに探索終了の通知を Publish してからサーバが Subscribe するまでの区間とする.

4.3 結果

クライアント台数ごとの平均値(Average)・標準偏差(Standard Deviation)を表 1 に示す.

表 1 台数ごとの平均値(Ave)・標準偏差(Std)
(単位:ms)

	3 台		5 台		7 台	
	Pub	Sub	Pub	Sub	Pub	Sub
Ave	55.65	68.24	63.13	72.33	74.16	87.36
Std	5.76	8.21	5.52	8.87	5.0	8.61

結果より、本研究における提案手法はクライアント台数が増加するに伴い、サーバの Pub/Sub 時間ともに比例するように増加した。サーバからクライアントへの Publish 平均時間は 3 台から 5 台において 13.4% 増加し、5 台から 7 台では 17.5% 増加した。またサーバのクライアントからの Subscribe 平均時間は 3 台から 5 台において 6.0% 増加し、5 台から 7 台において 20.8% 増加した。

4.4 考察

サーバが 1 台目のクライアントからの終了通知を Subscribe し領域管理中の場合、別のクライアントからの終了通知により Topic が更新されても Subscribe() により監視していないため、メッセージを Subscribe することができない。したがってクライアント台数が増加するに伴い、同時にサーバへ終了通知を Publish される可能性が高まるため、平均値・標準偏差の値が増加している。ただし、これらはサーバの Pub/Sub 処理におけるマルチスレッド化を行うことで改善できると考える。現段階では処理が逐次的に経過時間に従って実施しているが、クライアントからの終了通知を Subscribe する Subscriber、クライアントごとの領域管理、次の探索領域をクライアントに Publish する Publisher の 3 つに分割し、並列化することで将来的に課題に対応できると考える。同時に複数のクライアントからサーバに対し探索終了通知が Publish されたとしても、割り込み可能な状態で Subscriber が Topic の監視を行っていれば、Topic が更新されるとすぐに Subscribe することができる。

5 まとめと今後の課題

本研究において、従来の ROS の Pub/Sub モデルの性能を損なうことなく、特定のクライアントのみにメッセージを送付する仕組みを実現した。従って、より実践的な災害現場を想定した複数台ロボットにおけるフォーメーション制御に用いることにより、より効率的な探索を行えると考える。

また、実機実験においてクライアント台数 7 台まで行った。したがってクライアント台数 8 台以上に増えた場合、Publish 平均時間・Subscribe 平均時間ともにどのように増加、もしくは収束するのか調査する必要がある。

複数台の自律型ロボットを扱うにあたり、効率的に動作させるためにはサーバからクライアントは一律通信を行い、クライアント同士は一对一通信を行うようなシステムが必要である。したがって、本研究の提案手法をさらに改善し ROS 上のライブラリとしてコンポーネント化することにより、ROS ユーザと共有することができ ROS の

発展につながる。また、先行研究で述べたような災害現場における具体的な複数台ロボットのフォーメーション制御に適用することで、より効率的な探索を実現することができると思う。

6 参考文献

- [1]“Quince 開発コンセプト” 千葉工業大学 未来ロボット技術研究センター
<http://www.furo.org/ja/robot/quince/index.html>(参照 2018-01-11)
- [2]“空気噴射により瓦礫を浮上して乗り越える策状ロボットを開発” JST トップ
<http://www.jst.go.jp/pr/announce/20170612/>(参照 2018-01-11)
- [3] 徳田 一. “レスキューロボットに見る災害対応の取り組み”. 和歌山大学防災研究教育センター紀要. 2015, 第 1 号
- [4] T. Kamegawa, T. Yamasaki, F. Matsuno, "Evaluation of snake-like rescue robot "kohga" for usability of remote control", IEEE International Workshop on Safety Security and Rescue Robotics SSRR, pp. 25-30, 2005.
- [5] 須原 亨, 法月 広夢, 内村 裕, ネットワークベース系におけるパケットロスと変動時間遅れを考慮した制御系の設計, 電気学会論文誌, 137 巻 (2017) 2 号, p. 87-94
- [6] Syh-Shiuh Yeh, Cheng-Chung Hsu, Tzu-Chiang Shih, Remote control realization of distributed rescue robots via the wireless network, SICE Annual Conference, 2008, 20-22 Aug. 2008.
- [7]“About ROS”. ROS.org. <http://www.ros.org/about-ros/>(参照 2018-01-11)
- [8] 前掲“Mobility and Sensing Demands in USAR”. IECON2000. 2000, Vol.1, pp.138-142
- [8] 杉山久佳, 辻岡哲夫, 村田正. “ネットワーク化された群ロボットによる被災者発見システム”. 情報処理学会論文誌. 2005, Vol. 46, No. 7, pp1777-1788
- [9] 松井進. “アドホックネットワークの実用化に向けた課題と実用化動向”. REAJ 誌. 2012, Vol.34, No.8, pp532-539