

ICNを適用したセンサーネットワークプラットフォーム実現方式の検討



水野修, 改田高大, 小池将史

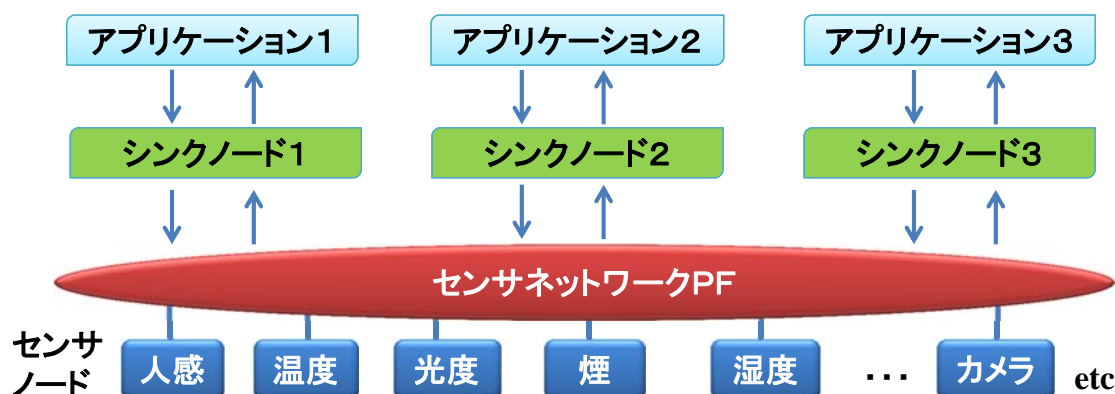
† 工学院大学大学院

Copyright © Kogakuin University (Japan). All Rights Reserved

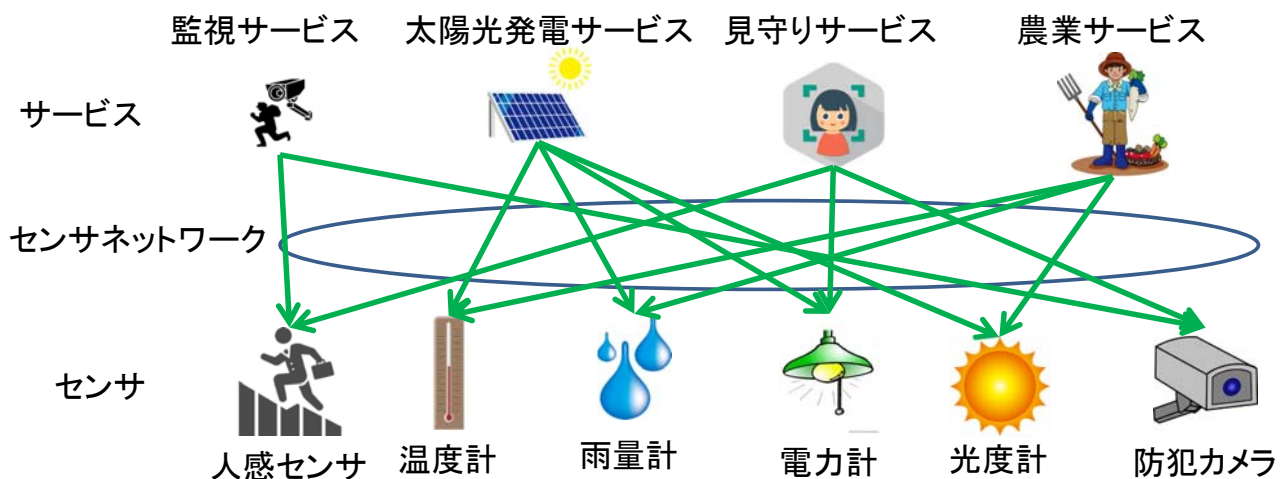
背景



- センサが自律的に通信により情報を送信し, サービスを実行するM2Mサービスが普及
- M2Mサービスはサービスごとにセンサネットワークを構築
- NetInfを適用することで複数のサービスを実現するセンサネットワークプラットフォームを構築することを提案

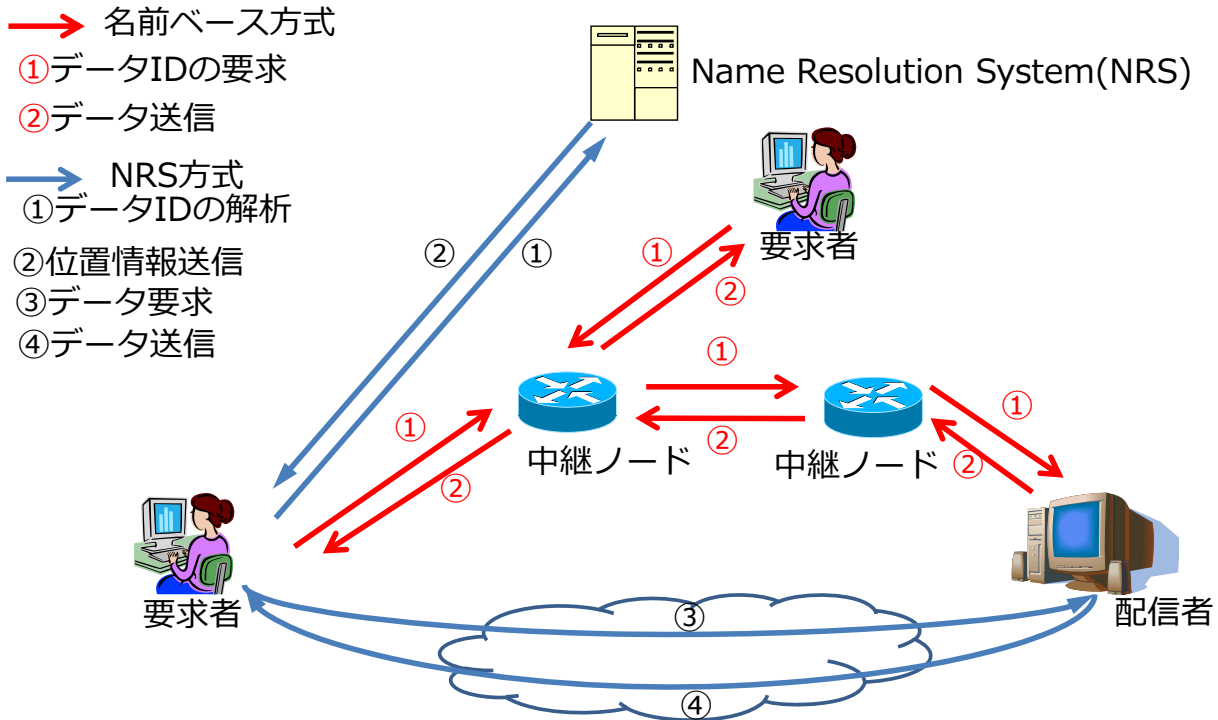


- サービスごとに必要とするセンサデータをセンサネットワークから収集
- センサネットワークを構成するノードは多種多様なセンサから構成



センサネットワークの要件

- 要件1：センサの負荷低減
多くのセンサノードは低性能であるため、負荷を小さくする必要あり
- 要件2：多対多通信への対応
サービスとセンサ間における多対多通信により大量のトラフィックに対する対応が必要
- 要件3：ネットワークの動的変化への対応
故障やバッテリー切れによるノードや経路の消失への対処が必要



[1] NetInf Network of Information, <http://www.netinf.org/>

要件への対応

要件

対応

要件1：センサの負荷低減

NRSを用いた経路構築と、キャッシュ機能によりある程度満足できる

要件2：多対多通信への対応

キャッシュ機能により、多対多通信で発生する負荷へ対応

要件3：ネットワークの動的変化への対応

データIDにより、ネットワークの変化に関わらずセンサデータの取得が可能

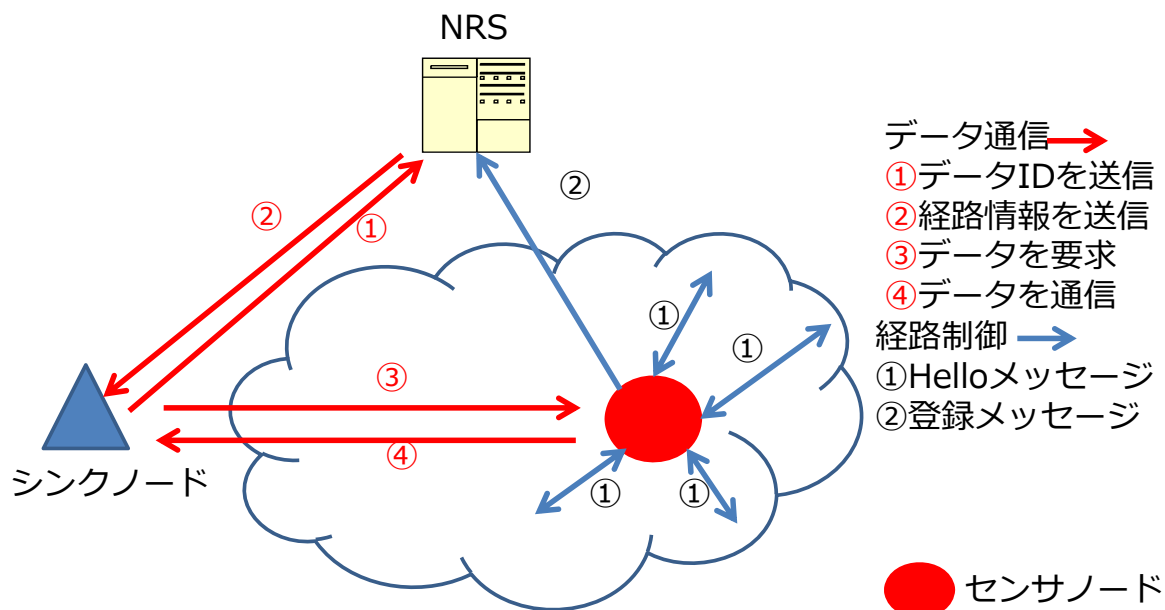
NetInfを適用したセンサネットワークプラットフォームを構築し，動作を検証

- 名前解決方式の決定
 - ・ NRS方式
 - ・ 名前ベース方式

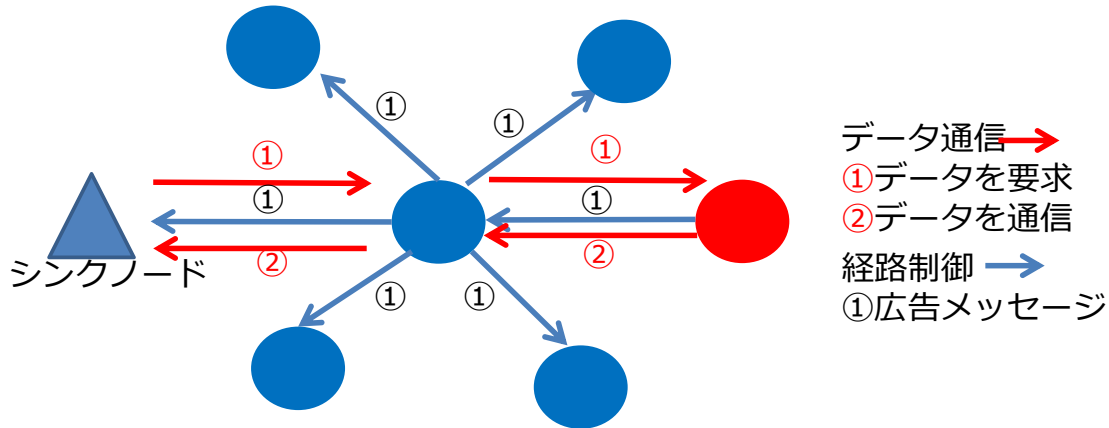
経路作成方式の決定

NRS方式

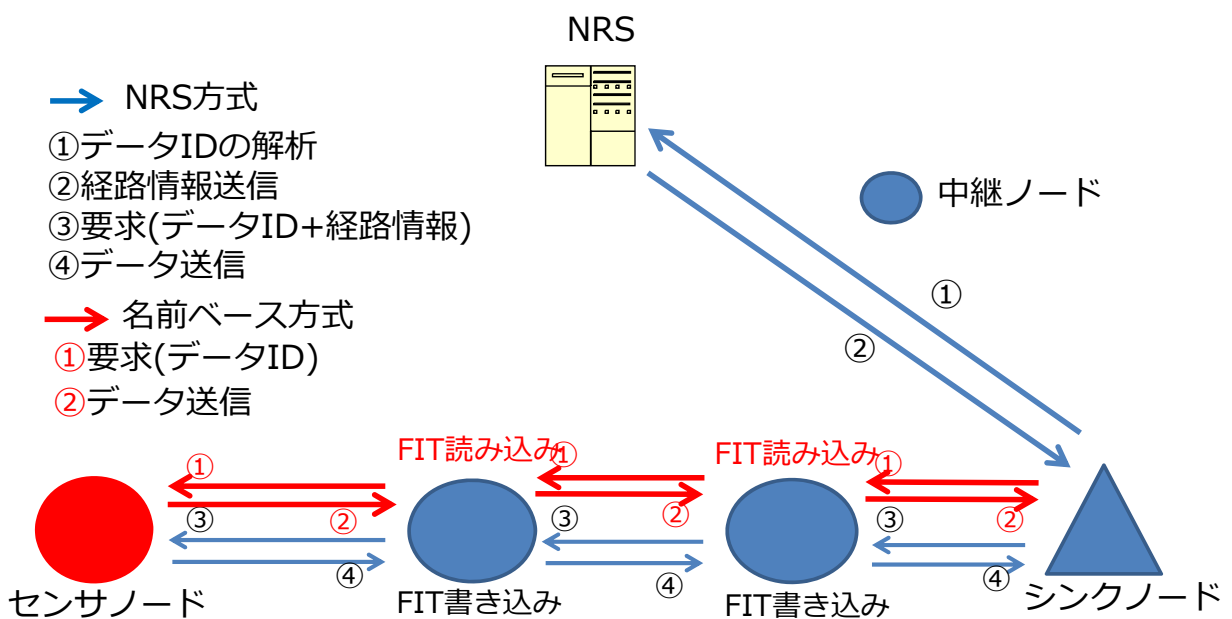
- 経路制御時に発生する信号はHelloメッセージと登録メッセージ
- NRSで経路情報を取得し，データの要求を送信



- データIDの広告メッセージをフラッディング
- 広告メッセージで構築された経路により, データの要求を送信
- NRSは必要としない



NetInfを適用したセンサネットワーク



FIT(Fording Interface Table) : データIDのルーティングテーブル

- 各名前解決方式をグリッド状のセンサネットワークで経路制御時に発生する信号の量を評価

➤NRS方式

- NRSへの登録メッセージ
 - センサノードのhelloメッセージ
- } $O(n^{3/2})$

➤名前ベース方式

- 広告メッセージ
- $O(n^2)$

(n :ノード数)

実験

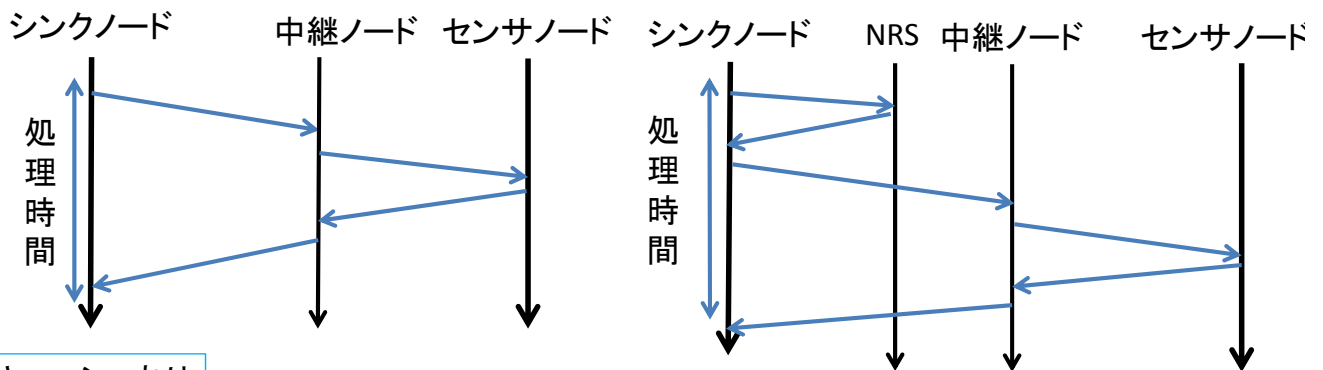
- NetInfを適用したセンサネットワークをXbeeを用いることで実装^[3]
- センサネットワークの動作を検証
- 評価指標
遅延時間：要求してからデータを受信するまでの時間
- シナリオ
 - シナリオ1:多対多通信への対応を検証
 - シナリオ2:ネットワークの動的変化を検証
- 比較対象
 - センサノードの故障前と後
 - キャッシュ有無とNRSベース方式, 名前ベース方式

実験シーケンス

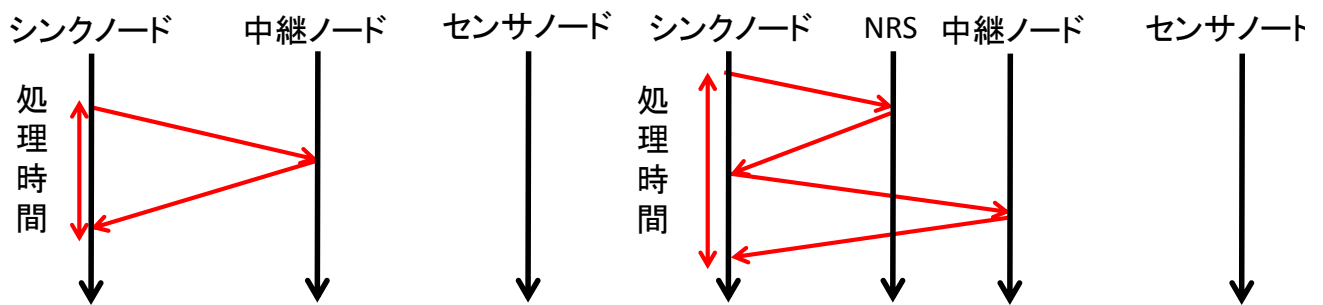
キャッシュなし

名前ベース方式

NRS方式

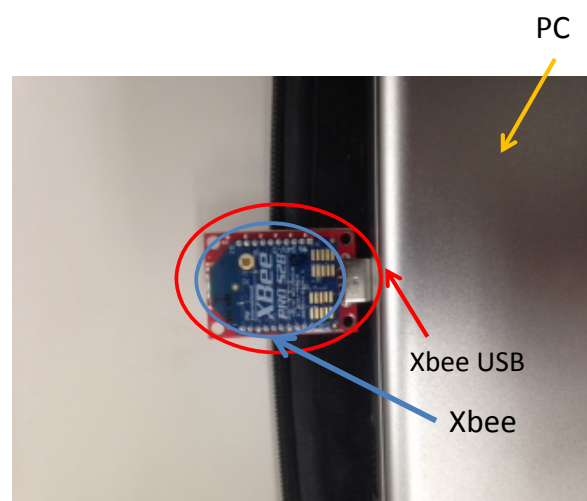


キャッシュあり



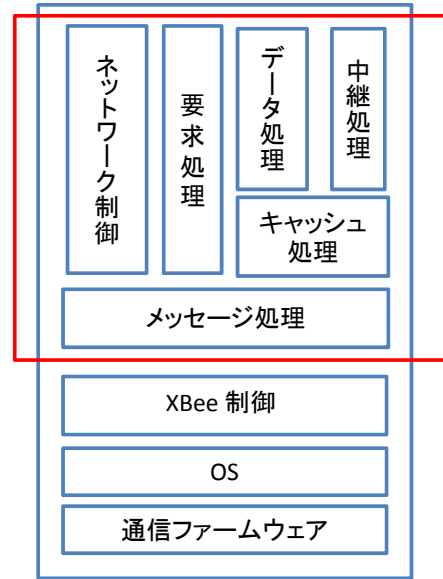
ハードウェアの構成

項目	仕様
OS	Windows 7 professional 32bit
無線送受信機	Xbee Pro S28
シリアル	パソコン(USBver2.0)
アンテナタイプ	チップアンテナ
通信ファームウェア	NRS: ZIGBEE COORDINATOR API NRS以外: ZIGBEE ROUTER API



- **メッセージ処理**
 - 受信したメッセージを各処理に振り分ける
 - 各処理の結果をメッセージの形に変換し、送信するための処理を実行
- **ネットワーク制御**
NRSにおける経路管理, センサノード位置情報の処理を実行
- **要求処理**
 - シンクノードの要求処理
 - センサノードの要求処理
- **キャッシュ処理**
要求受信時はキャッシュ動作を処理
- **データ処理**
センサデータの送信を処理
- **中継処理**
受信した要求, センサデータの中継処理

作成範囲



送信メッセージの構成

ノードIDはMACアドレスの2オクテットの10進数

データID+"/"+メッセージの種類+"/"+PITの更新+"/"+TTL+"/"+
+要求元ノードID+"/"+経路情報+":"+データ本体



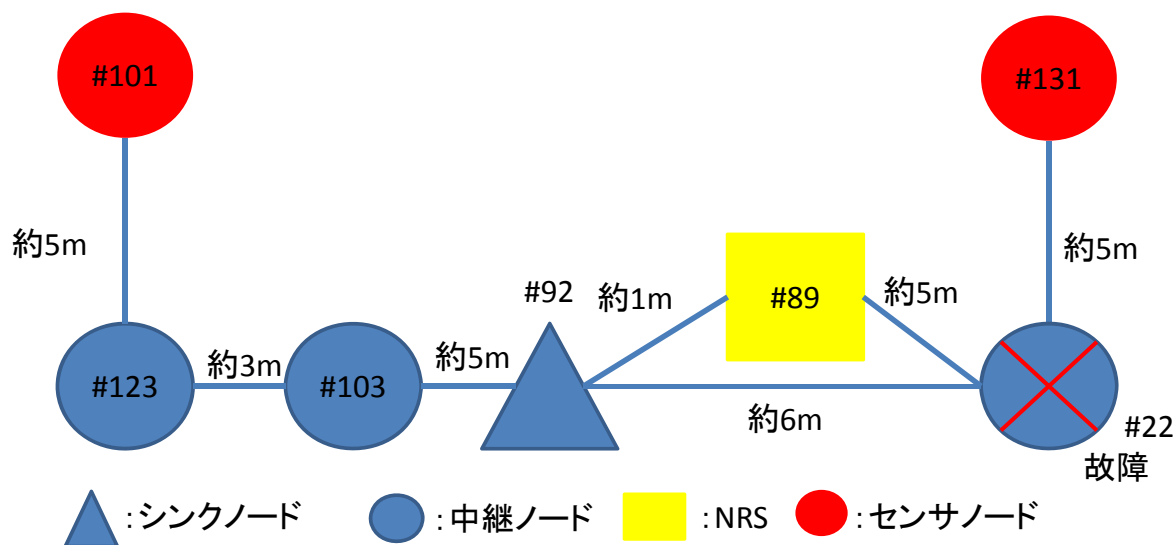
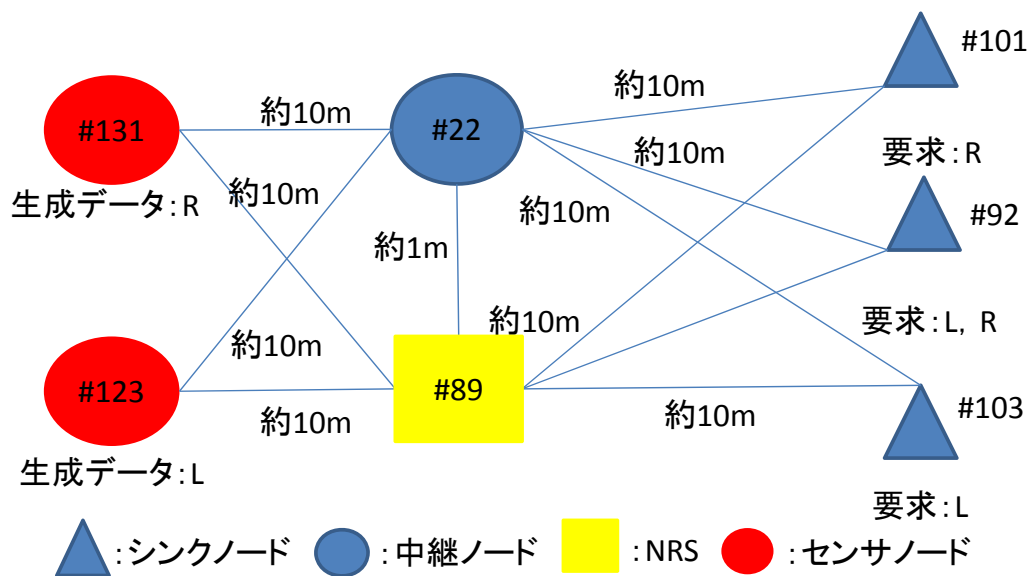
データID=センサID+"_"+タイムスタンプ
センサIDはセンサの種類によって固定
メッセージの種類

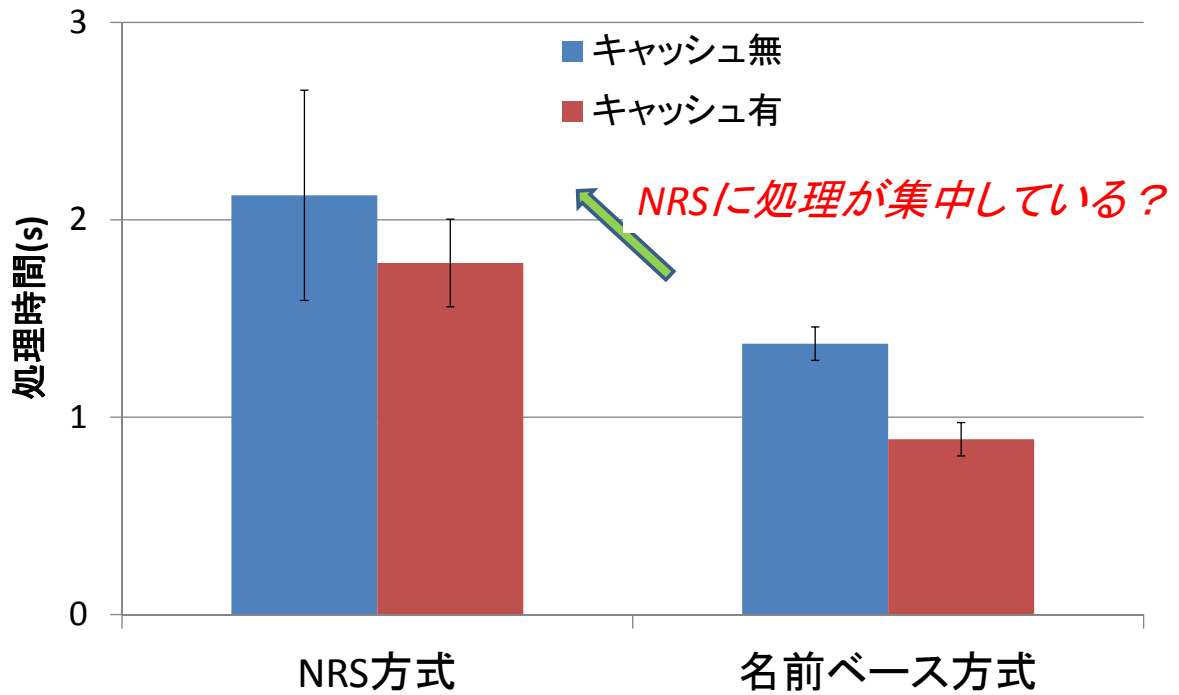
ネットワーク制御メッセージ

0:Hello 1:Hello_Answer 2:NRS_Route 3:Regist 4:Regist_Answer

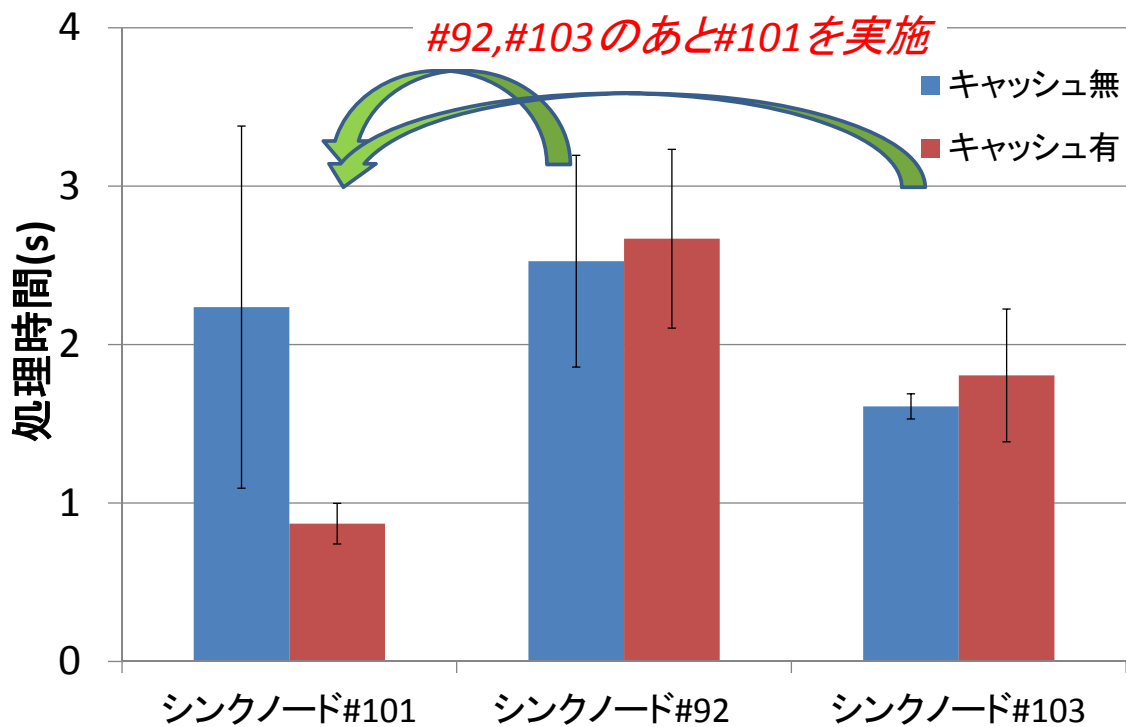
5:Requet_NRS 6:Request_Answer 7:Request_Location 8:Request 9:Send_Data

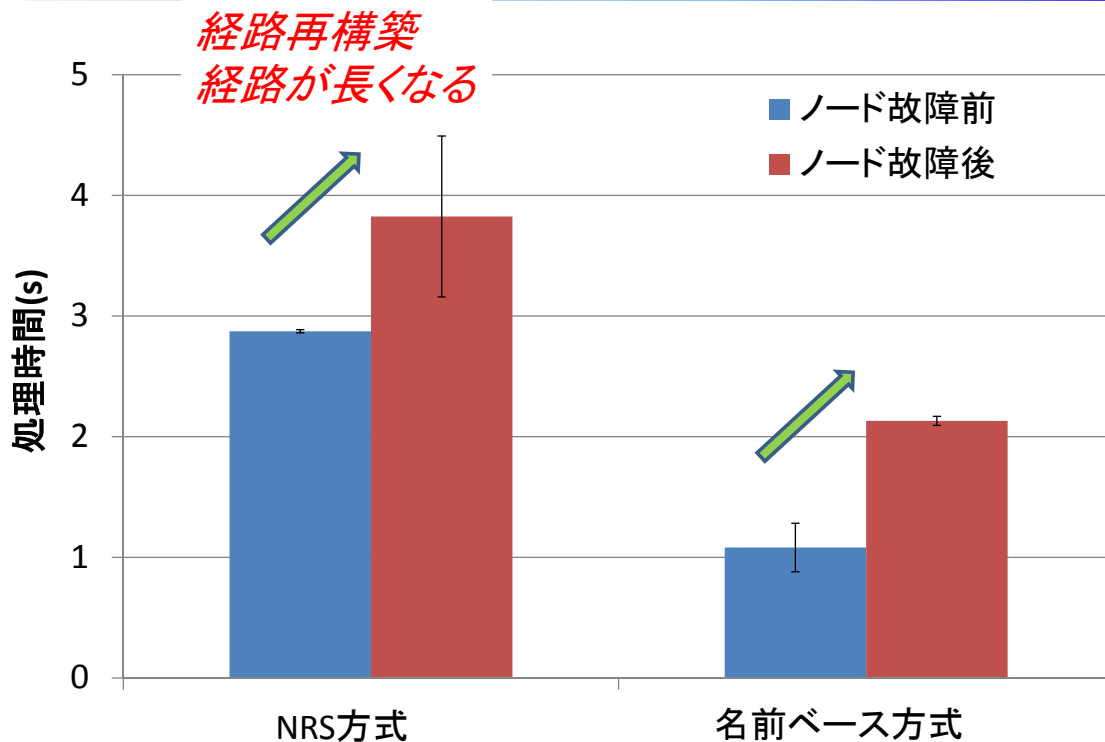
要求関連のメッセージ





実験結果(シナリオ1) シンクノードごと(NRS方式)





考察

- シナリオ1より、キャッシュ機能によりシンクノード同士のセンサデータの共有により通信時間の短縮を確認
- シンクノードによってはデータを生成するセンサノードでデータを取得したため、通信時間は短縮されない
- シナリオ2より、ネットワークの動的変化が発生したが、シンクノードの設定を変更せずに他のセンサからデータを取得
- NetInfによりネットワークの動的変化に対応できていることを確認

- キャッシュの効果は大きそう
 - 経路を維持しなくてよい. サーバレス可
 - 経路が短くできる分, 電力も節約できそう
- センサデータのサイズが小さい (データIDと同じくらい?)
 - NRSに位置データを置くのと, データそのものを置くのと変わりがない?
 - データIDを広告するのと, データそのものを広告するのと変わりがない?
- ICNはプル型⇒イベント駆動で生成されるデータの扱い?

センサーネットワークに適したICNがある(かも知れない)

まとめと今後の課題

- まとめ
 - センサネットワークにNetInfを適用したときの名前解決方式をモデルにより評価
 - 実験により提案したセンサネットワークの動作を検証
- 今後の課題
 - 実装の仕様: とくにNRS
 - センサーネットワークに適したICNは?