

# トリアージネットワークにおける密度変化に応じた データ送信手法の検討\*

齋藤 卓也<sup>†</sup> 小林 ひかる<sup>‡</sup> 田村 寛樹<sup>‡</sup> 戸口 裕人<sup>‡</sup> 重野 寛<sup>† §</sup>  
慶應義塾大学理工学部<sup>†</sup> 慶應義塾大学大学院理工学研究科<sup>‡</sup>  
独立行政法人科学技術振興機構, CREST<sup>§</sup>

## 1 はじめに

災害や事故が発生した際に傷病者にタグを付け、重傷度によって救命の順番を決定するトリアージ救命救急支援システムに注目が集まっている。中でもタグを電子化することでセンサネットワークを構築し、傷病者の病状を一括管理をする電子トリアージシステムの研究が活発に行われている [1]。傷病者はトリアージポストと呼ばれる、医療従事者が重傷度を判断する場所に搬入され、ポスト内で周期的に生体情報をシンクに送信する。

トリアージで扱うデータは傷病者の生体情報であるため、全データが重要であり高いデータ到着率が必要とされる。しかし事故や災害の規模が大きくなると、傷病者数の増加とともに、センサネットワークが構築されているポスト内のセンサ密度とトラフィック量が増加する。これによって帯域が圧迫されデータ到着率が低下する。そこで本稿ではセンサ密度に応じてデータ送信間隔を変更することで線型的に増加するトラフィック量を抑制し、データ到着率を維持する手法を提案する。

## 2 背景

### 2.1 トリアージネットワーク

トリアージ現場では災害・事故発生後、傷病者はトリアージポストに搬入される。そしてポスト内で医療従事者によって重傷度の高い順に赤 (緊急)、黄 (準緊急)、緑 (歩行可能)、黒 (死亡) のタグを付与される。その後、重傷度の高い順に医療機関へ搬入・治療が行われる。

現在、トリアージタグには紙製のものが利用されている。そのため、一度トリアージされた後の状態変化を、医療従事者が傷病者のもとへ行かないと把握できない。この問題点に対して、トリアージタグにセンサを取り付けることでポスト内にセンサネットワークを構築し、傷病者の生体情報を一括管理できるトリアージ支援ネットワークの研究が活発に行われている。以降トリアージタグを付けられた傷病者をノード、データ収集と管理を行うサーバをシンクと

する。ノードが定期的に生体情報のデータを送信することで、シンクはポスト内の傷病者の状態を把握でき、効率的な搬出と治療が可能となる。

### 2.2 既存方法と問題点

トリアージで扱うデータは傷病者の生体情報であるため、高いデータ到着率が必要とされる。既存手法である AODV ではシンクまでの経路構築・修復を効率的に行うことができる。しかし災害や事故の規模が大きくなると、傷病者の増加に伴いポスト内のセンサ密度とトラフィック量が増加する。AODV では重傷度に区別なく全ノード同一周期でデータを送信するため、傷病者数とともにトラフィック量は線形的に増加する。そのため帯域圧迫によるデータ到着率低下が起こりやすい。

## 3 提案

データ到着率維持のために、密度変化に応じてデータの送信周期を変更する手法 DTPDC(Data Transmission Protocol accommodating to Density Change) を提案する。DTPDC ではノードを重傷度ごとに区別するために、まずトリアージ新規搬入ノードは、重傷度に応じたレベル  $L$  を経路構築時にシンクへ送信する。 $L$  は送信周期を変更するノードを指定する際に利用する。 $L$  の高いノードは重傷度が高いため短い送信周期を維持し、また  $L$  の低いノードは密度に応じて送信周期を変更する。従って全体のデータ送信回数は減少し、トラフィック量を抑制できる。

図 1 に DTPDC の動作例を示す。まずノードはポスト搬入後、経路構築を行いレベル  $L$  をシンクに送信する。経路構築後、初期送信周期  $\alpha$ (sec/回) でシンクに向けてデータを送信する。シンクは一定時間  $dt$  毎に  $L$  の最大値  $L_{max}$ 、時刻  $t$  でのポスト内ノード数  $N_t$ 、ポスト内ノード数の最大値  $N_{max}$  を算出し、ブロードキャストする。トリアージ開始から傷病者の搬入が終了するまでポスト内のノード数は増加するため、 $N_{max}$  は搬入終了時刻でのポスト内ノード数  $N_{end}$  と等しい。しかし搬入終了時刻が不明なため、搬入が終了するまではポスト内の許容ノード数  $N_{limit}$  を利用する。よって  $N_{max}$  は式 (1) のように算出する。

$$N_{max} = \begin{cases} N_{limit} & (\text{搬入継続: } N_t > N_{t-dt}) \\ N_{end} & (\text{搬入終了: } N_t \leq N_{t-dt}) \end{cases} \quad (1)$$

\*Data Transmission Protocol accommodating to Density Change for Triage Network

<sup>†</sup>Takuya Saito, Hiroshi Shigeno

<sup>‡</sup>Faculty of Science and Technology, Keio University

<sup>‡</sup>Hikaru Kobayashi, Hiroki Tamura, Yuto Toguchi

<sup>‡</sup>Graduate School of Science and Technology, Keio University

<sup>§</sup>Hiroshi Shigeno, JST, CREST

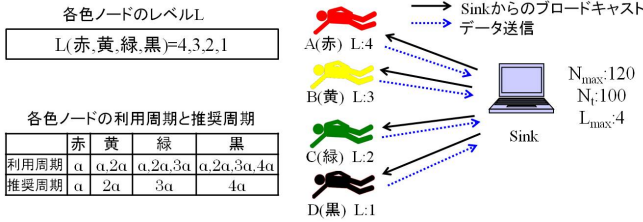


図 1: DTPDC の動作例

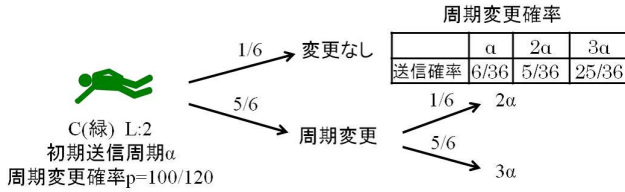


図 2: 緑ノードの周期変更アルゴリズム

シンクからのブロードキャストパケット受信後，ノードは重傷度に応じた自身のレベル  $L$  とポスト内ノードの最大レベル  $L_{max}$  を比較し， $L < L_{max}$  ならば周期変更の候補となる．送信周期変更の候補となったノードはデータ送信の度に確率  $P$  で  $\alpha$ (sec/回) から利用周期へ変更する．確率  $P$  は式 (2) を利用する．

$$P = \frac{N_t}{N_{max}} \quad (2)$$

このとき，高密度の際に推奨される周期を推奨周期と呼ぶ．利用周期，推奨周期は重要度に応じて異なる．また重傷度が高いノードは推奨周期が短く，重傷度が低いノードは推奨周期が長い．

図 2 に緑ノードを例に送信周期変更アルゴリズムを示す．緑ノードは送信周期を  $\alpha$  から， $P = 100/120 = 5/6$  の確率で変更し，その変更後の周期は確率  $P$  で推奨周期となる．つまり緑の推奨周期  $3\alpha$  となる確率は  $\frac{5}{6} \times \frac{5}{6} = \frac{25}{36}$  で表される． $N_t$  が大きい (ポスト内のノード数が多い) と推奨周期で送信する確率が高くなる．逆に  $N_t$  が小さいときは送信周期を変更する確率が低くなり，短い周期でデータ送信を行う確率が高くなる．

#### 4 シミュレーション評価

ネットワークシミュレータ QualNet5.0 に DTPDC を実装した．表 1 にシミュレーション条件を示す．本シミュレーションでは，災害発生後傷病者がポストに搬入，一定時間経過後重傷度の高いノードから搬出されていく環境を想定している．DTPDC の送信周期変更の挙動，トリアージポスト内での総トラフィック量，データ到着率の評価を行う．また比較対象として AODV を用いる．

##### 4.1 シミュレーション結果

図 3 で DTPDC の各色ノードの送信周期変更の挙動，図 4 に総トラフィック量とデータ到着率の関係を示す．図 3 より重傷度の低いノードが密度に応じて周期を変更していることが分かる．また図 4 より DTPDC がデータ送信周期を変化させることでトラフィック量を抑制し，データ到着率を維持できていることが分かる．

表 1: シミュレーション条件

シミュレータ	Qualnet5.0	配置	ランダム
試行範囲	50m × 50m	ノード数	120
ブロードキャスト間隔	300(sec)	ポスト許容ノード数	150
初期送信間隔	5(sec/回)	色比率 (赤:黄:緑:黒)	1:1:1:1
ノード参加時刻	0(sec)	レベル (赤, 黄, 緑, 黒)	4,3,2,1
搬入終了時刻	3600(sec)	無線通信距離	30(m)
搬出開始時刻	7200(sec)	データサイズ	64(bytes)
試行時間	10800(sec)	試行回数	3 回

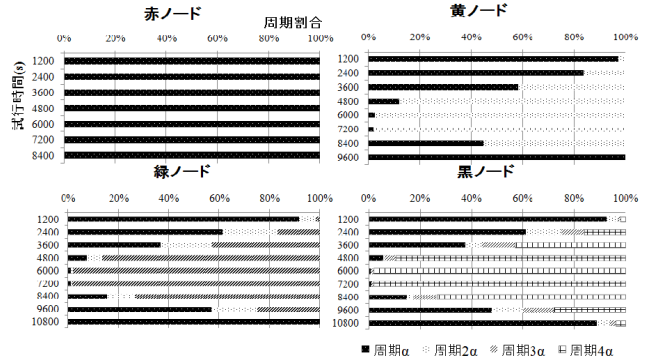


図 3: 各色ノードのデータ送信周期の変化

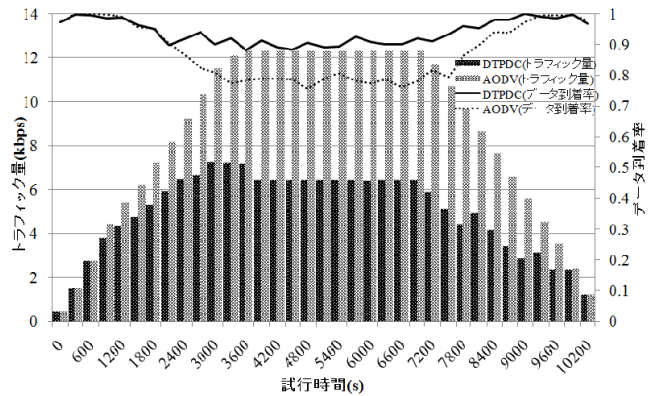


図 4: 総トラフィック量とデータ到着率

#### 5 おわりに

本稿では傷病者数の多いトリアージ支援ネットワークにおいて，密度変化に応じたデータ送信手法を検討した．コンピュータシミュレータを用いて提案の評価を行い，既存手法である AODV と比較して，データ送信周期を変更させることでデータ到着率を維持できることを確認した．

#### 参考文献

[1] Hikaru Kobayashi, Hiromitsu Tomozawa, Hiroshi Shigeno, Ken-ichi Okada, "Efficient Routing Using Node Pull Out Estimation", 4th International Conference on Complex, Intelligent and Software Intensive Systems(CISIS-2010), pp.193-200, February, 2010.