

# 焼きなまし法を用いた TDMA スケジューリング技術

杉原 真<sup>1,a)</sup> 岩永 明人<sup>2</sup>

**概要：**車載向けアプリケーションに代表される産業用アプリケーションを具現化するためには、デッドライン時刻までにノード間で通信メッセージを送信するネットワークシステムを安価に実現する必要がある。本稿では、通信バスの動作周波数を削減することが、ワイヤハーネスの低コスト化に貢献することを考慮し、ハードリアルタイム制約下において通信バスの動作周波数を最小化する TDMA スケジューリング技術を提案する。具体的には、フレームのペイロード長、及び、スロットマルチプレクシングを含む送信スケジュールを最適化し、FlexRay バスの動作周波数を最小化する。まず、対象とする設計問題を数理問題として定式化する。さらに、設計問題の解を求める焼きなまし法に基づく解法を与える。計算機実験により、提案手法は既存手法と比べ、62.3~63.2%の動作周波数を削減できることを確認した。

**キーワード：**TDMA 法, 製造コスト, 動作周波数, スロットマルチプレクシング

## Simulated Annealing-Based TDMA Scheduling Technique

**Abstract:** A cost-effective communication mechanism that is capable of sending out a communication message from node to node by their deadline is required for realizing industrial applications such as automotive ones. Reduction of the operating frequency of a communication bus generally contributes to lowering the cost of wire harness. This paper presents a design paradigm in which we optimize the length of a payload segment of a frame and slot multiplexing under hard real-time constraints so that we can minimize the operating frequency of the communication bus. We formulate the above design problem as a mathematical problem. We also apply simulated annealing (SA) to solving the design problem. Our experimental results show that our technique achieved 62.3–63.2% less operating frequency comparing than that of a conventional one.

**Keywords:** TDMA (time division multiple access) scheme, fabrication cost, operating frequency, slot multiplexing

### 1. はじめに

自動車産業においては、経済性、安全性、及び快適性を絶えず追求し、自動車製品を改良し続けている。自動車の進化はエレクトロニクスの進化に負うところが大きい。車載エレクトロニクスは次世代自動車の開発においてより重要な役割を果たすと考えられる。

車載エレクトロニクスは高価な電子デバイスと配線から構成されるために、商業的観点からは、システムを安価に組み上げる設計パラダイムが求められる。現状では、高級車には 100 個を越す ECU(electronic control unit) が搭載され、ECU は大規模なネットワークを介して互いに通信を行っている。自動車の実現には、高度でかつ大規模な分散システムが必要であり、低コストな分散システムを設計する技術必要不可欠である。

CAN (controller area network) は、相互接続された機器間

のデータ転送に用いられる通信ネットワークの標準であり、車載ネットワークに用いられている [2]。CAN においては、フレームの優先度に基づいて通信バスの使用順序が定まるために、上位の優先度を持つフレームが非決定的に送信される場合、下位の優先度を持つフレームの通信レイテンシは非決定的に定まる。通信レイテンシが非決定的に定まるのであれば、リアルタイム性を保証することは難しい。x-by-wire システムのようなハードリアルタイム性が要求されるシステムに CAN を適用する事は困難である [3]。FlexRay は TDMA (time division multiple access) 法に基づいた新しいネットワーク標準であり、ネットワーク帯域幅を使用する上で、通信レイテンシにおける決定性と柔軟性を兼ね備えている [4]。FlexRay の通信機構は Static TDMA 法で通信レイテンシにおける決定性を提供し、Flexible TDMA 法で通信帯域幅における柔軟性を与える。

FlexRay の通信機構の設計パラダイムについて研究する事は喫緊の課題である。我々はこれまでに、通信バスの動作周波数を削減すれば、ワイヤハーネスの調達コストを削減できる点に着目し、フレームのペイロードセグメントの大きさを最適化し、通信バスの動作周波数を最小化する設計方法論を提案した [10, 11]。さらに、転送されるデータの誤り率を考慮した TDMA スケジューリング技術を提案し

<sup>1</sup> 九州大学システム LSI 研究センター  
System LSI Research Center, Kyushu University, 744 Motoooka,  
Nishi-ku, Fukuoka, Fukuoka 819-0395, Japan

<sup>2</sup> 九州大学大学院統合新領域学府オートモーティブサイエンス専攻  
Department of Automotive Science, Graduate School of Integrated  
Frontier Sciences, Kyushu University, 744 Motoooka, Nishi-ku,  
Fukuoka, Fukuoka 819-0395, Japan

<sup>a)</sup> sugihara@src.kyushu-u.ac.jp

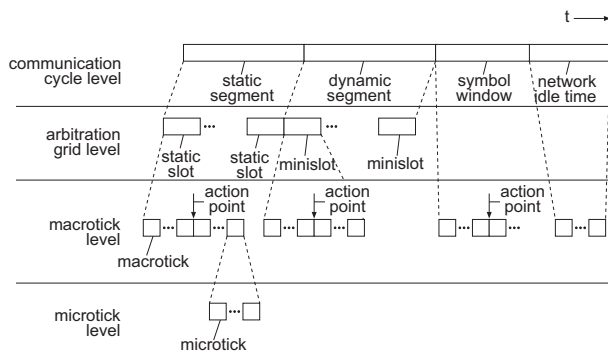


図1 通信サイクル内の時間階層.

た [5]. これらの研究では、スロットマルチプレクシング、すなわち、複数の通信サイクルに亘り、複数の通信信号を同一のスロットに割り当てる技法、を考慮しておらず、効果的に動作周波数を削減することができない。本稿では、スロットマルチプレクシングを考慮し、FlexRay バスの動作周波数を最小化し、FlexRay の通信機構を製造するために必要なコストを削減する設計方法論を提案する。具体的には、フレームのペイロード長、及び、スロットマルチプレクシングを含む送信スケジュールを最適化し、通信バスの動作周波数を最小化する設計技術を提案する。バスの動作周波数を削減することによって、より低速で安価な配線を使用でき、FlexRay ネットワークシステムを製造するコストを削減できる。

本稿の残りは次のように構成される。2 節では、FlexRay ネットワークプロトコルについて簡単に述べる。3 節では、本稿で取り扱う設計問題を定式化し、設計問題の解法として焼きなまし法を適用する。4 節では、実験結果により提案手法の有効性について議論する。5 節で結論を述べる。

## 2. FlexRay

本節では FlexRay 標準について簡単に調査する。FlexRay に関する詳細は仕様書を参照されたい [4].

### 2.1 通信サイクル

通信サイクルは、FlexRay システムのメディアアクセス法を構成するために周期的に繰り返される通信構造の一つのインスタンスである。通信サイクルは時間階層によって定義される。時間階層は四つの時間階層レベルから構成される。すなわち、マイクロティックレベル、マクロティックレベル、アービトレーショングリッドレベル、及び通信サイクルレベルの四つである。一つの通信サイクルに着目した時間階層を図 1 に示す。

通信サイクルレベルでは、通信サイクルが繰り返し実行される。一つの通信サイクルには、静的セグメント、動的セグメント、シンボルウィンドウ、及びネットワークアイドル時間が含まれる。本稿では特に静的セグメントに注目する。静的セグメント内では、送信を調停するために TDMA (time division multiple scheme) 法が用いられる。TDMA 法は共有される通信路のための通信路アクセス法である。TDMA 法では、チャンネルを時間的に分割し、タイムスロットが設けられる。各々のタイムスロットは、あらかじめ許可された通信メッセージのみが使用できる。動的セグメント内では、送信を調停するために FTDMA (flexible time division multiple access) 法が用いられる。シンボルウィンドウは

ネットワークにシンボルを送るために用いられる通信期間である。ネットワークアイドル時間は、通信が行われない期間であり、通信サイクルを完結するものである。

次の下位のレベルであるアービトレーショングリッドレベルはアービトレーショングリッドを含む。静的セグメントでは、アービトレーショングリッドは静的スロットと呼ばれる連続する時間間隔で構成される。動的セグメントでは、アービトレーショングリッドはミニスロットと呼ばれる連続する時間間隔で構成される。

アービトレーショングリッドレベルはマクロティックレベルの上に位置し、マクロティックによって定義される。マクロティックはグローバル時間を表す最小単位である。図 1 に示されるマクロティックの境界はアクションポイントと呼ばれる。アクションポイントは送信器が FlexRay フレームの送信を開始する時刻である。

時間階層において最も低いレベルはマイクロティックである。一つのマイクロティックは時間における一間隔を示し、ノードに局所的な概念である。

### 2.2 メディアアクセス制御

FlexRay のプロトコルにおいては、メディアアクセス制御は繰り返しされる通信サイクルに基づいている。通信サイクル内においては、FlexRay は二つのメディアアクセス法を提供する。すなわち、STDMA (static time division multiple access) 法、及び FTDMA (flexible time division multiple access) 法である。前者の方法は静的セグメントで用いられ、後者の方法は動的セグメントで用いられる。パケットのように送信の単位である通信フレームは静的セグメント及び動的セグメントの両方で送られる。本稿では簡単に静的セグメントのための TDMA 法について簡単に調査する。

全てのフレームは各々のフレーム ID を持っている。フレーム ID はユニークに静的スロットに対応付けられており、フレーム ID によって送信スロットが決定される。フレーム ID をノードにユニークに割り当てることで調停は実現される。新しい静的スロットになると、スロットカウンタはその値を一つ増加する。フレームのフレーム ID がスロットカウンタと一致したときに、当該フレームは送られる。スロットカウンタの初期値は 1 である。静的セグメントの終わりにスロットカウンタは 1 に再初期化される。

全ての静的セグメントは同じ数のマクロティックから構成される。一つの静的セグメントのマクロティック数は大域的に定数である。図 2 に静的セグメントのタイミングの詳細を示す。

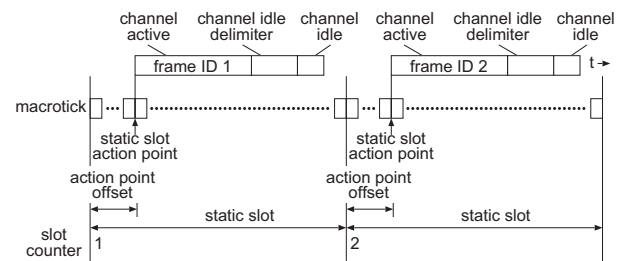


図2 静的セグメント内のタイミング.

### 2.3 フレームフォーマット

FlexRay フレームは送信のための入れものであり、三つ

のセグメントから構成される。すなわち、ヘッダセグメント、ペイロードセグメント、及びトレーラセグメントである。図3に FlexRay フレームのフォーマットを示す。

ノードはヘッダセグメント、ペイロードセグメント、及びトレーラセグメントの順でフレームを送信する。図3において、個々のセグメントにおいて、ノードは左から右の順序でデータを送信する。

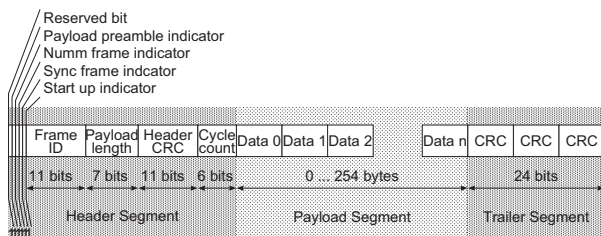


図3 フレームのフォーマット。

### 2.4 フレームの符号化

FlexRay フレームは5つのビット列で符号化される。すなわち、送信開始列 (TSS: transmission start sequence), フレーム開始列 (FSS: frame start sequence), バイト開始列 (BSS: byte start sequence), フレーム終了列 (FES: frame end sequence), 及び動的フレーム指示列 (DTS: dynamic trailing sequence) である。TSS 及び FSS はフレームの開始時に挿入される。フレームの全てのバイトにはその先頭に BSS が挿入される。FES はフレームの後に挿入される。DTS は動的セグメントにおいてフレームの FES の直後に挿入される。

図4に、静的セグメントの符号化されたフレームを示す。TSS はネットワークを介して適切な接続準備を開始するために用いられる。送信ノードは、FlexRay のパラメータで規定される一定期間連続する LOW 信号から構成される TSS を生成する。FSS が TSS の後に続く。FSS は TSS の後の最初の BSS においてあり得る量子化誤差を補うために用いられる。FSS は1ビット時間の HIGH 信号によって構成される。ノードは送信されるフレームの TSS の直後のビットストリームに FSS を追加する。BSS は受信デバイスにビットストリームのタイミング情報を与えるために用いられる。BSS は1ビット時間の HIGH 信号と1ビット時間の LOW 信号から構成される。フレームデータの各バイトは、一つの BSS と8ビットのデータ列から構成される拡張バイト列として、通信路で送信される。FES はフレームの最後のバイト列の終了を示すために用いられる。FES は1ビット時間の LOW 信号と1ビット時間の HIGH 信号で構成される。ノードはフレームの最後の拡張バイト列の直後のビットストリームに FES を付加する。

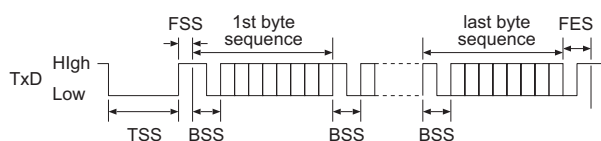


図4 フレームの符号化。

## 3. スロットマルチプレクシング最適化問題

通信バスの動作周波数を削減することによって、システム構築に用いるワイヤハーネスの調達コストを抑制することができる。本稿は、ハードリアルタイム制約の下で、スロットマルチプレクシング及びフレームのペイロード長を最適化し、バスの動作周波数を最小化する設計方法論を提案する。本節では、対象とする設計問題を定式化し、設計問題の解法として焼きなまし法を応用する。

### 3.1 定式化

本稿では、通信ノードがデッドラインまでに固定長のデータを送る要求を周期的に、あるいは、非周期的に発する振る舞いを通信信号と定義する。また、それぞれの送信要求において送信対象となる固定長データを通信メッセージと呼ぶ。実装の観点からは、通信メッセージは一個以上のフレームに格納され、送信元ノードから送信先ノードに送られることとなる。問題の簡単化のために、本稿では周期的な通信信号から構成されるネットワークシステムを仮定する。通信メッセージが非周期的に送られるネットワークシステムは、一般性を失わずに周期的なシステムに変換することができる。通信メッセージの最小間隔を通信信号の周期とすることで、非周期的信号を周期的信号とみなすことができる。通信信号は四つ組  $(N, C, D, S)$  で表される。ここで、 $N$  はネットワークノードを、 $C$  は周期を、 $D$  は相対デッドライン時刻を、 $S$  は通信メッセージのビット数を示す。ネットワークノード  $N$  は  $C$  単位時間毎に  $S$  ビットの通信メッセージを送信する要求を発生し、それぞれの通信メッセージを、送信要求が発せられてから  $D$  単位時間以内に送り終わる必要があると仮定する。

$N_{sig}$  個の通信信号  $\mathcal{S} = \{s_1, s_2, \dots, s_{N_{sig}}\}$  が、 $q$  個の静的スロットから構成される静的セグメントによって送信されるネットワークシステムを考えよう。本稿では、静的スロットの数  $q$  を変数として取り扱う。通信信号  $s_i$  は四つ組  $(N_i, C_i, D_i, S_i)$  で定義される。一つの通信信号に対して、一つの通信サイクルにおいては高々一つの静的スロットが割り当てられると仮定する。ネットワークシステムの動作周波数は  $w$  であり、ボーレートは  $w$  bps であるとする。より低い動作周波数を実現することによってより安価なワイヤハーネスを選択できるとの前提に立ち、目的関数を通信バスの動作周波数  $w$  の最小化とする。

一つの通信メッセージは一個以上のフレームの形で送られる。図3に示すように、一つのフレームはヘッダセグメント、ペイロードセグメント、及びトレーラセグメントから構成される。本稿では、ヘッダセグメントの大きさ及びトレーラセグメントのビット量をそれぞれ  $B_{hd}$  及び  $B_{tl}$  とする。ペイロードセグメントの大きさは、設計者が定められるものであるが、任意の静的セグメントにおいて同一である必要がある。一つのペイロードセグメントは  $p$  ( $0 \leq p \leq 127$ ) 個の16ビットワードから構成されるとする。なお、FlexRay の標準においては、ペイロードセグメントの大きさは偶数バイトであるとされ、その範囲は0以上254以下であることが規定されている。フレームのビット長  $f$  は以下のように示される。

$$f = B_{hd} + B_{tl} + 16p. \tag{1}$$

符号化されたフレームの大きさ  $f_{enc}$  は以下のように表される。



$$f_{\text{enc}} = f + \left( B_{\text{TSS}} + B_{\text{FSS}} + \frac{f}{8} \cdot B_{\text{BSS}} + B_{\text{FES}} + B_{\text{DLM}} + B_{\text{IDL}} \right)$$

$$= \left( 1 + \frac{B_{\text{BSS}}}{8} \right) f + B_{\text{TSS}} + B_{\text{FSS}} + B_{\text{FES}} + B_{\text{DLM}} + B_{\text{IDL}}$$

$$= (16 + 2B_{\text{BSS}})p + O, \quad (2)$$

$$O = \left( 1 + \frac{B_{\text{BSS}}}{8} \right) (B_{\text{hd}} + B_{\text{tl}})$$

$$+ B_{\text{TSS}} + B_{\text{FSS}} + B_{\text{FES}} + B_{\text{DLM}} + B_{\text{IDL}}. \quad (3)$$

ここで、 $B_{\text{TSS}}$ ,  $B_{\text{FSS}}$ ,  $B_{\text{BSS}}$ ,  $B_{\text{FES}}$ ,  $B_{\text{DLM}}$ , and  $B_{\text{IDL}}$  はそれぞれ TSS, FSS, BSS, FES, チャネルアイドルデリミタ, 及びチャネルアイドル時間を表す。

説明の簡単化のために、ハードデッドライン制約を持つ通信信号だけが与えられるとし、一つの通信サイクルは一つの静的セグメントだけで構成されるとする。本節で定義される問題は、動的セグメント、シンボルウィンドウ、及びネットワークアイドルタイムが固定長である限り、静的セグメント、動的セグメント、シンボルウィンドウ、及びネットワークアイドルタイムを持つシステムに拡張できる。仮定から、1つの通信サイクルの時間  $t_{\text{cc}}$  は  $q$  個の静的セグメントの時間と等しく、以下のように示される。

$$t_{\text{cc}} = \frac{q \cdot f_{\text{enc}}}{w}$$

$$= \{(16 + 2B_{\text{BSS}})p + O\} \frac{q}{w}. \quad (4)$$

ここで、 $q$  は一つの通信サイクルを構成する静的スロット数を表す整数変数であり、 $w$  は FlexRay バスの動作周波数を表す実数変数である。

FlexRay の標準では、スロットマルチプレクシング、すなわち、複数の通信サイクルに亘り、複数の通信信号を同一の静的スロットに割り当てている技法、を許容している。具体的には、64 個の通信サイクルに亘り、最大 64 個までの通信信号を同一の静的スロットに割り当てることを許容している。64 個の通信サイクルは繰り返し出現することとなる。スロットマルチプレクシングを用いることで、実行帯域幅の使用効率が向上するが、設計空間が大きくなるために、費やすべき設計エフォートは増加する。問題の簡単化のために、整数変数  $a_i$  ( $0 \leq a_i \leq 6$ ) を導入し、一つの通信信号は 64 個の通信サイクルで、 $2^{a_i}$  個の静的スロットを消費し、 $2^{6-a_i}$  個の通信サイクル毎に一つの静的スロットを消費すると仮定する。また、整数変数  $b_i$  ( $0 \leq b_i \leq 6$ ) を導入し、一つの通信メッセージを送るために  $2^{b_i}$  個のフレーム（静的スロット）を用いると仮定する。

シグナル  $i$  の通信メッセージの大きさ  $S_i$  は、割り当てられた静的セグメントで送ることができるフレームのペイロードセグメントの総計を超えてはならない。よって、以下のような制約が導入される。

$$2^{b_i+4}p \geq S_i, \forall i. \quad (5)$$

通信信号の転送要求を満たす転送能力を実現する必要がある。すなわち、一つの通信メッセージを送信するために要する時間は、通信信号の周期以下である必要がある。よって、以下の制約が導入される。

$$2^{-a_i+b_i+6}t_{\text{cc}} \leq C_i, \forall i. \quad (6)$$

上記の制約は、式 (4) を用いて、以下のように変形できる。

$$\{(16 + 2B_{\text{BSS}})p + O\} \frac{2^{-a_i+b_i+6}q}{w} \leq C_i, \forall i. \quad (7)$$

通信信号が周期的に 64 サイクルのうち  $2^{a_i}$  個の通信サイクルを使用する形態は、位相が  $2^{6-a_i}$  通り存在するために、 $2^{6-a_i}$  通り存在する。通信信号の位相を示す整数変数  $c_i$  ( $1 \leq c_i \leq 64$ ) を導入する。

通信信号の位相が決まれば、通信信号に割り当てられる通信サイクルが全て定まる。通信信号が使用する通信サイクルを示すために、以下のような二値変数  $d_{i,j}$  を導入する。

$$d_{i,j} = \begin{cases} 1 & j = c_i + 2^{-a_i+6} \cdot m \ (0 \leq m \leq 2^{a_i} - 1) \text{ の場合.} \\ 0 & \text{それ以外.} \end{cases} \quad (8)$$

一つの通信サイクルに含まれる静的スロットの数  $q$  は、一つの通信サイクルに割り当てられる通信信号の最大数によって決定される。一つの通信サイクルあたりの静的スロットの数  $q$  は以下のように表される。

$$q = \max_j \sum_i d_{i,j}. \quad (9)$$

一つの通信メッセージは一つ以上のフレームの形で送信される。図 5 に、一つの通信メッセージが一つ以上のフレームの形で送られる様子を示す。一番目のフレームが静的スロットによって送られる前に待ち時間が生じる。通信信号が通信サイクル中の静的スロットの何れかに割り当てられると仮定する。以下に示すように、最大待ち時間は、割り当てられた静的スロットの間隔と等しいとする。

$$2^{-a_i+6}t_{\text{cc}}. \quad (10)$$

待ち時間の後に、フレームを送る時間が続く。一つの通信メッセージを送る時間が、以下に示す、割り当てられた通信サイクルの総計に等しいとする。

$$2^{-a_i+b_i+6}t_{\text{cc}}.$$

通信信号  $i$  の通信メッセージを送信するための最悪レイテンシ  $t_i$  は最悪待ち時間と一つの通信メッセージを送る時間の和であり、以下のように表される。

$$t_i = (2^{-a_i+6} + 2^{-a_i+b_i+6})t_{\text{cc}}. \quad (11)$$

ハードデッドライン制約を満たすために、通信信号  $i$  の通信メッセージを、相対デッドライン  $D_i$  を超えないように、送る必要がある。式 (11) より、以下の制約が導かれる。

$$(2^{-a_i+6} + 2^{-a_i+b_i+6})t_{\text{cc}} \leq D_i, \forall i.$$

上記の制約は、式 (4) を用いて、以下のように変形できる。

$$\{(16 + 2B_{\text{BSS}})p + O\} (2^{-a_i+6} + 2^{-a_i+b_i+6}) \frac{q}{w} \leq D_i. \quad (12)$$

数理モデルは以下のように与えられる。

<p><b>Minimize</b> the cost function <math>w</math></p> <p><b>subject to</b></p> <p>(1) <math>2^{b_i+4}p \geq S_i, \forall i.</math></p> <p>(2) <math>\{(16 + 2B_{\text{BSS}})p + O\} \frac{2^{-a_i+b_i+6}q}{w} \leq C_i, \forall i.</math></p> <p>(3) <math>q = \max_j \sum_i d_{i,j}.</math></p> <p>(4) <math>\{(16 + 2B_{\text{BSS}})p + O\} (2^{-a_i+6} + 2^{-a_i+b_i+6}) \frac{q}{w} \leq D_i, \forall i.</math></p>
---

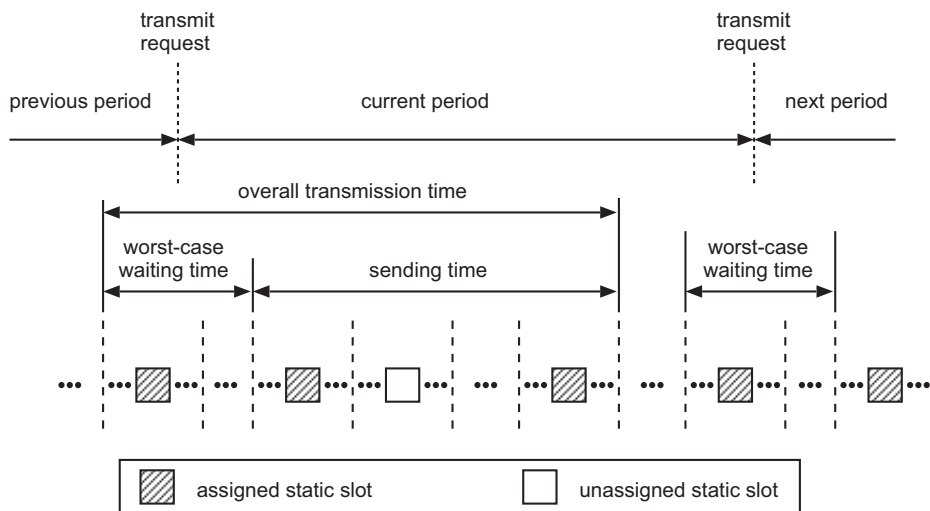


図5 通信メッセージの伝送時間.

### Variables

- $w$  is a real variable.
- $p$  is an integer variable.
- $q$  is an integer variable.
- $a_i$  is an integer variable.
- $b_i$  is an integer variable.
- $c_i$  is an integer variable.
- $d_{i,j}$  is a binary variable.

### 3.2 焼きなまし法を用いた TDMA スケジューリング

焼きなまし (SA: simulated annealing) 法は非決定性アルゴリズムの一つである [9]. 評価値が改善される解を採択することに加えて, ある条件のもとで評価値が悪化する解も採択する点が SA 法の特徴として挙げられる. 探索処理の初期段階では, 評価値が悪化する解を採択する確率が大きく, 探索処理が進むにつれ, 評価値が悪化する度合いが小さい解のみが採択されやすくなる. 最終的には, 評価値を改善する解のみが採択される.

SA 法の擬似コード [9] を図 6 に示す. 手続き SA においては, 手続き Metropolis の呼び出しが繰り返される. 手続き Metropolis が呼び出される度に, 温度を表す変数  $T$  に冷却率  $\alpha$  が乗じられ,  $T$  は減少する. 一方, 手続き Metropolis が呼び出される度に,  $M$  に  $\beta$  が乗じられ,  $M$  は増加する.  $M$  は手続き Metropolis における解の探索回数を決定するものである. 手続き Metropolis においては, 与えられた解の近傍を戻す手続き Neighbor によって得られた一時解を得る. 新たに得られた解と現在解の評価値の差分に基づいて, 処理が異なる. 評価値が削減された場合,  $NewS$  で  $BestS$  を更新する. さもなければ, 確率的に, 評価値の増加を許し,  $NewS$  で  $CurS$  を更新する.

本研究においては, SA 法を 3.1 節で示した問題の解法として用いた. 解空間は, 変数  $a_i, b_i, c_i, w, p$  によって構成される. 近傍を求める手続き Neighbor においては, 一定の確率に基づいて, 解を構成する変数を一つ選択し, 増減させる. 整数変数について, 1 だけ増減する.  $w$  に関しては, 正規分布 ( $\mu = 0, \sigma = 50$ ) に基づいて増減値を決定した.

### Pseudo-code for Simulated Annealing

```

Procedure SA( $S_0, T_0, \alpha, \beta, M, MaxTime$ )
  //  $S_0$  は初期解
  //  $T_0$  は初期温度
  //  $\alpha$  は冷却率
  //  $\beta$  は定数
  //  $M$  は次のパラメータ更新までの時間
  //  $MaxTime$  は総処理時間
  begin
     $T = T_0$ ;
     $CurS = S_0$ ;
     $BestS = CurS$ ;
     $CurCost = Cost(CurS)$ ;
     $BestCost = Cost(BestS)$ ;
     $Time = 0$ ;
    repeat
      Metropolis( $CurS, CurCost, BestS, BestCost, T, M$ );
       $Time = Time + M$ ;
       $T = \alpha T$ ;
       $M = \beta M$ ;
    until ( $Time \geq MaxTime$ );
  end

Procedure Metropolis( $CurS, CurCost, BestS, BestCost, T, M$ )
  begin
    repeat
       $NewS = Neighbor(CurS)$ ;
       $NewCost = Cost(NewS)$ ;
       $\Delta Cost = NewCost - CurCost$ ;
      if ( $\Delta Cost < 0$ ) then
        if ( $NewCost < BestCost$ ) then
           $BestS = NewS$ ;
        endif
      else
        if ( $RANDOM < e^{-\Delta Cost/T}$ ) then
           $CurS = NewS$ ;
        endif
      endif
       $M = M - 1$ ;
    until ( $M = 0$ );
  end
  
```

図6 SA 法の擬似コード

## 4. 実験

### 4.1 準備

SAE ベンチマークは自動車向け分散制御システムの通信信号要件の一例である. SAE ベンチマークは 53 種類の通信信号から構成される. SAE ベンチマークの詳細は Kutlu らによって与えられている [6]. 本稿の実験においては, 大

きな通信信号数に対しても SA に基づいたソルバが品質の高い解を求めることができることを確認するために、SAE ベンチマークを用いて仮想的な通信信号集合を作成した。実験に用いた通信信号集合を表 1 に示す。表 1 中のベンチマーク  $b_n$  は  $n$  個分の SAE を含むことを意味する。

表 1 実験に用いた通信信号集合.

ベンチマーク名	通信信号数
$b_1$ (SAE ベンチマーク)	53
$b_2$	106
$b_3$	159
$b_4$	212
$b_5$	265
$b_6$	318
$b_7$	371

本稿で示す計算機実験には、表 2 に示す、Park らによって与えられたネットワークパラメータ集合 [7] を用いた。チャンネルアイドルレミタの直後にはチャンネルアイドルタイムがないと仮定した。

表 2 ネットワークパラメータ.

Factor	Length
Header w/o BSS	5 B/frame
Header w BSS	45 bits/frame
Trailer w/o BSS	3 B/frame
Trailer w BSS	27 bits/frame
TSS	9 bits/frame
FSS	1 bit/frame
FES	2 bits/frame
Idle delimiter	11 bits/frame
Action point offset	1 MT/frame, 10 bits/frame
BSS	2 bits/frame byte

3.1 節に示した問題の解を与える、SA 法に基づいたソルバを C++ 言語によって開発した。SA のパラメータとして、表 3 に示すものを用いた。200 通りのパラメータの組合せに対して、SA 法に基づくソルバを実行し、解を求めた。

表 3 SA のパラメータ.

パラメータ	
$S_0$	SM を行わない場合の最適解 [10, 11]
$T_0$	100
$\alpha$	0.9, 0.99, 0.999, 0.9999, 0.99999
$\beta$	1.00001, 1.0001, 1.001, 1.01
$M$ の初期値	10, 20, 30, 40, 50, 60, 70, 80, 90, 100

## 4.2 実験結果

提案手法の有効性を示すために、SA に基づくソルバで得られる TDMA スケジュールを、既存研究の手法 [10, 11] を用いて得られるものと比較した。提案手法はスロットマルチプレクシングを採用するものであり、一方、既存手法はスロットマルチプレクシングを採用しないものである。図 7 に実験結果を示す。得られたバスの動作周波数は、通信信号集合の規模に比例して増加することが窺える。既存の手法と比べて、提案手法は 62.3~63.2%の動作周波数を削減する結果となり、スロットマルチプレクシングの効果を実験的に確認できた。また、通信信号数が 159 個を超える規模のベンチマークに対しては、既存手法では、FlexRay の規格上許される 10Mbps を超える結果となり、FlexRay に準拠したシステムを構築することができないことが理解される。

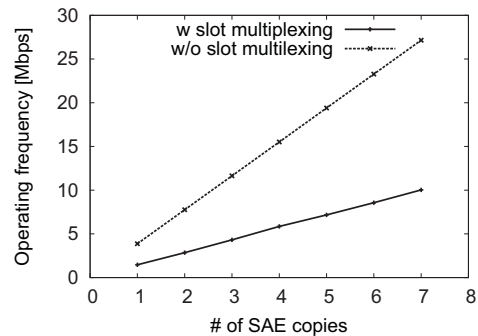


図 7 通信信号集合の規模と動作周波数の関係.

## 5. おわりに

本稿においては、スロットマルチプレクシング、及びフレームのペイロード長を最適化し、通信バスの動作周波数を最小化する TDMA 通信機構の設計手法を提案した。SA 法を採用することで、大規模な数の通信信号を含むシステムに対しても、品質の高い TDMA スケジュールを実用時間内に得ることが可能となった。スロットマルチプレクシングを採用しない既存手法 [10, 11] と比べ、提案手法は動作周波数を 62.3~63.2%削減できることを実験的に確認した。提案手法によって、より低速で安価なワイヤハーネスを用いることができることが期待される。

## 参考文献

- [1] J. Bisschop, AIMMS –optimization modeling–, Apr. 2011.
- [2] Bosch, “Control area network,” <http://www.can.bosch.com/>, Jan. 2013.
- [3] G. Cena and A. Valenzano, “Achieving round-robin access in controller area networks,” *IEEE Trans. Ind. Electron.*, Vol. 49, No. 6, pp.1202–1213, Dec. 2002.
- [4] FlexRay Consortium, FlexRay communications system protocol specification version 3.0.1, Oct. 2010.
- [5] 岩永明人, 杉原真, “信頼性要求に基づいた対故障性を実現する TDMA スケジューリング手法,” 情報処理学会組込みシステムシンポジウム, pp.39–47, 2012 年 10 月.
- [6] A. Kutlu, H. Ekiz, and E. T. Powner, “Performance analysis of MAC protocols for wireless control area network,” *Proc. International Symposium on Parallel Architectures, Algorithms, and Networks*, pp. 494–499, Jun. 1996.
- [7] I. Park and M. Sunwoo, “FlexRay network parameter optimization method for automotive applications,” *IEEE Trans. on Industrial Electronics*, Vol. 58, No. 4, pp. 1449–1459, Apr. 2011.
- [8] K. Schmidt and E. G. Schmidt, “Message scheduling for the FlexRay protocol: the static segment,” *IEEE Trans. Vehicular Technology*, Vol. 58, No. 5, pp. 2170–2179, Jun. 2009.
- [9] 白石洋一 訳 Sadiq M. Sait 著, 組合せ最適化アルゴリズムの最新手法, 丸善株式会社, 2002 年 3 月.
- [10] 杉原真, 岩永明人, “ハードリアルタイム性を考慮した FlexRay のバス帯域幅の最小化技術,” 情報処理学会研究報告, Vol. 2012-EMB-24, No. 19, pp. 1–6, 2012 年 3 月.
- [11] M. Sugihara and A. Iwanaga, “Minimization of FlexRay bus bandwidth for hard real-time applications,” *Journal of Information Processing*, Vol. 21, No. 1, pp.46–52, Jan. 2013.
- [12] H. Zeng et al., “Schedule optimization of time-triggered systems communicating over the FlexRay static segment,” *IEEE Trans. Industrial Informatics*, Vol. 7, No. 1, pp. 1–17, Feb. 2011.