

## オフセット付き CAN メッセージの正確な最大遅れ時間解析

倉地 亮<sup>†1</sup> 陳 暘<sup>†2</sup> 高田 広章<sup>†1,†2</sup>

これまでオフセット付き CAN メッセージの最大遅れ時間の解析手法が提案されている。しかしながら、従来の手法では計算量を抑え速く効率的に計算出来る一方、悲観的に解析される場合がある。このため、本論では従来手法の問題点を指摘し、より正確な最大遅れ時間を求めるための手法について提案する。提案する手法は、クリティカルフレームに着目し限定した範囲に限り網羅的に解析を行うことで、厳密な最大遅れ時間を導出する方法である。本提案手法の評価として自動車メーカーから提供されるメッセージセットに適用した結果、十分に少ない組み合わせでより正確な解析結果がえられた。

### Exact Schedulability Analysis for CAN Messages with offsets

RYO KURACHI,<sup>†1</sup> CHEN YANG<sup>†2</sup>  
and HIROAKI TAKADA<sup>†1,†2</sup>

The existing analysis for CAN messages with offsets can compute fast and efficient by reducing the amount of computation. However, it is pessimistic in some cases. The causes of the pessimism is that the cumulative accounting of maximum interference functions is overestimated much greater than a realistic situation. In this paper, we show the cause of this situation that existing analysis takes into account unrealistic interference in the cumulative accounting. Then, we propose the exact schedulability analysis which involves the partially exhaustive analysis. According to our experiments by using a actually-used message sets, we can conclude that our approach can significantly reduce the combination of the critical frames, and its can achieve an exact schedulability analysis.

### 1. はじめに

車載ネットワークの中でも高いリアルタイム性が要求される制御系ネットワークにおいて CAN(Controller Area network)<sup>1)</sup> が広く使われており、ハードリアルタイムシステムの研究分野においても、CAN メッセージの最大遅れ時間を解析する方法が研究されている。CAN メッセージの最大遅れ時間解析とは、CAN メッセージに対し静的優先度ベーススケジューリングである Rate Monotonic Analysis を適用し、各メッセージが送信要求されてから送信完了するまでの最大実行時間を計算することで、与えられたデッドラインを満たすかどうかを確認するための手法である。

まず、最初の CAN メッセージの最大遅れ時間の解析手法として、Tindell らは基本となるオフセットが付かない CAN メッセージの解析手法を提案し<sup>2)</sup> これまでに多くの研究が行われた<sup>3)</sup>。その後、各 CAN メッセージにオフセットと呼ばれる初回送信時までの待ち時間を持たせ、各 ECU から送信される CAN メッセージを分散することで、リアルタイム性を保証しながら CAN ネットワークの回線使用率を向上させる取り組みがなされており<sup>4),5)</sup> これらのオフセット付き CAN メッセージの最大遅れ時間の解析手法については、飯山らがマルチフレームタスクのモデルを適用した手法を提案した<sup>6)</sup>。しかしながら、従来手法は効率良く解析できる一方、実際よりも悲観的に解析されてしまうことがある。

そこで本論では、まず、従来手法が悲観的に解析されてしまう場合について、具体例を用いて説明する。その上で、正確な最大遅れ時間を求める手法として、限られた範囲で網羅的な解析を行う方法を提案する。具体的には、従来手法より求まるクリティカルフレームに着目し、限定されたクリティカルフレームに対し網羅的に組み合わせ解析を行う方法である。クリティカルフレームとは、文献 7) のマルチフレームタスクの解析において、ある時刻に最も邪魔するフレームのことを指し、文献 7) ではすべてのクリティカルフレームのみに着目し解析する手法である。しかしながら、CAN メッセージの解析の場合にはマルチフレームタスクの解析とは性質と対象が異なるため、文献 7) の手法をそのまま適用しても有効な手法とはならない。このため、我々が提案するクリティカルフレームを限定する手法を用い

†1 名古屋大学大学院情報科学研究科 附属組込みシステム研究センター  
Center for Embedded Computing Systems, Nagoya University

†2 名古屋大学大学院情報科学研究科情報システム学専攻  
Department of Information Engineering, Graduate School of Information Science, Nagoya University

ることで、オフセット付き CAN メッセージの正確な最大遅れ時間解析が可能となる。

また、オフセットが付く CAN メッセージの解析においては critical instant が予めひとつに定まらないため、幾つかの critical instant candidate において最大遅れ時間を解析し、その候補の中から最も大きい最大遅れ時間をもつ状況を critical instant として決定する方法をとる。このため、本提案手法では、従来手法では導出することが出来ない critical instant を導出する方法も同時に検討する。仮に critical instant がわかれば、シミュレータ等を用い、その状況の再現が容易となるため、解析の妥当性について検証がしやすくなるという利点がある。

本論文の構成は以下のとおりである。第 2 章で従来の解析手法とその問題を示し、第 3 章では問題解決のためのアプローチと正確な最大遅れ時間を求める解析手法を提案する。第 4 章では自動車メーカーから提供されたメッセージセットを用いて評価を行い、本手法の有効性を示し、第 5 章で本論をまとめる。

## 2. 従来手法

本章では、最初に本論で扱うメッセージモデルの説明を行う。その上で、従来手法であるオフセット付き CAN メッセージの最大遅れ時間解析手法について説明し、従来手法において悲観的に解析される状況を指摘する。

### 2.1 メッセージモデル

メッセージセット  $S$  は、 $k$  個の ECU,  $ECU_1, \dots, ECU_k$  で構成される。各 ECU の送信メッセージは ECU 内のタイマーイベントなどにより相対的な時間で起動されるが、初回送信時に限ってはオフセットと呼ばれる初回送信時刻だけ待ってから送信され、以降は与えられた周期に従い、繰り返し送信が実行される。尚、CAN はプロトコル上、各 ECU は同期しておらず、ECU 間は互いに非同期として扱う必要がある。以降では、メッセージ  $\tau_i$  は、CAN メッセージの優先度  $P_i$ 、最大送信時間  $E_i$ 、送信周期  $C_i$ 、初回送信時のオフセット  $O_i$  の 4 つ組  $(P_i, E_i, C_i, O_i)$  で表すものとする。ここで、具体例として  $\tau_i = (1, 3, 12, 4)$  は図 1 のように表記できる。

尚、解析上、与えられるオフセットや周期、優先度については制限を与えるものではないが、デッドラインは周期と同じあるいはそれよりも小さいものとし、メッセージの各パラメータは静的に決められている必要がある。さらにシステム上メッセージのバス使用率は 100% を超えてはならないことだけ注意が必要である。

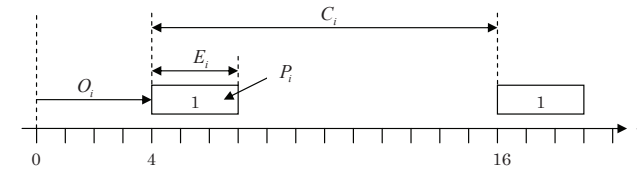


図 1 CAN メッセージの表記  
 Fig. 1 Notation of CAN messages

### 2.2 従来手法の概要

飯山らは、Tindell らの手法は解析対象となるメッセージの送信要求と同時にすべてのメッセージの送信要求が同時に不出される状況であることを前提としており、オフセットの付いた場合には正確に解析できないことを指摘した上で、オフセットの付いた CAN メッセージの最大遅れ時間を求める手法を提案した<sup>6)</sup>。飯山らが提案する解析手法では、オフセット付き CAN メッセージのモデルがマルチフレームタスクのモデルと類似しているということから、マルチフレームタスクのモデルにおける各フレームを CAN メッセージとして扱うことで、Maximum Interference Function(以降、MIF と呼ぶ)を用いて、最大遅れ時間を解析する手法である。

### 2.3 Maximum Interference Function

MIF とは、あるタスクがある時刻において他のタスクを邪魔しうる最大影響時間を表す時間関数であり、CAN メッセージの解析においては、あるメッセージグループから送信されるメッセージ群が他のグループに存在する対象メッセージを最大で邪魔する時間を表す<sup>8)</sup>。ここで、MIF の具体例として、2 つのメッセージ  $\tau_1 = (1, 1, 6, 0)$  と  $\tau_2 = (2, 2, 12, 3)$  が含まれるメッセージグループ  $J$  の MIF について、図 2 を用いて説明する。図 2 は、 $\tau_1$  と  $\tau_2$  の送信周期の最小公倍数(以降、LCM)である 12 までの間に発生する各送信要求時刻からの送信状況を表している。また、各メッセージの送信要求時刻から始まる状況はクリティカルリリース<sup>7)</sup>と呼ばれ、この各送信要求時刻から始まる Interference function(以降、IF と呼ぶ)を作成し、MIF を導出する。IF とは、メッセージ  $\tau_i$  よりも優先度が高いあるメッセージ  $\tau_j$  の送信要求時刻  $st$  を時刻 0 とする時間関数であり、 $IF_{P_j}^{st}(\tau_i, t)$  と定義できる。具体的には、図 2 の例において、 $\tau_1$  の送信要求時刻 0 から始まる状況を  $IF_1^0(\tau_i, t) = \{(0, 1), (3, 2), (6, 1), 12\}$ 、 $\tau_1$  の送信要求時刻 6 から始まる状況を  $IF_1^6(\tau_i, t) = \{(0, 1), (6, 1), (9, 2), 12\}$ 、 $\tau_2$  の送信要求時刻 3 から始まる状況を  $IF_2^3(\tau_i, t) = \{(0, 2), (3, 1), (9, 1), 12\}$  とする各凸点の集合とそ

の周期で表すものとする。また、各凸点の集合は図2に示すように、x軸を開始時刻からの相対的な経過時間、y軸を邪魔時間の総量とするグラフとして表記できる。

ここで、MIFとはグループに含まれるすべてのIFを重ね合わせたときの各時間における最大値をとり、<sup>8)</sup> 図2のグループJのMIFは  $M_J(\tau_i, t) = \max(IF_1^0(\tau_i, t), IF_1^6(\tau_i, t), IF_2^3(\tau_i, t))$  であり、各IFのy軸が最大となる線を繋いだ  $M_J(\tau_i, t) = \{(0, 2), (3, 1), (6, 1), 12\}$  となる。すなわち、MIFとはあるグループから送信されるメッセージが他のグループに存在する送信メッセージ  $\tau_i$  を最大で邪魔する状況を表す。

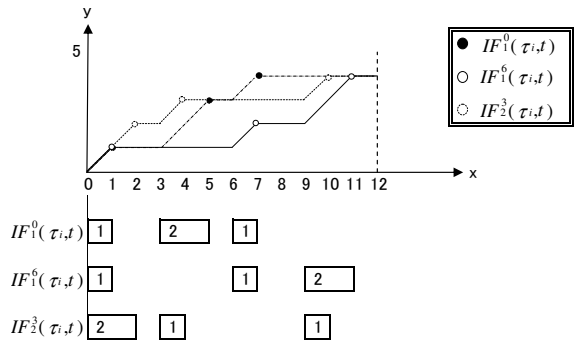


図2  $\tau_1$  と  $\tau_{au2}$  を含むグループJのIFとMIFの例  
Fig.2 Examples of MIF and IF of the group J.

## 2.4 飽和加算

従来手法では、すべての他グループの高優先度メッセージが対象メッセージに与える影響を累積的に計算するために、各グループごとに導出するMIFを飽和加算し、その影響時間の総和を導出している。飽和加算とは、傾きが1を超えないように累積的に積算する方法である<sup>9)</sup>。図3の例は、 $MIF_K(\tau_i, t) = \{(0, 2), (5, 1), (7, 1), 10\}$  と  $MIF_L(\tau_i, t) = \{(0, 1), 5\}$  を飽和加算し、 $MIF_{K,L}$ を生成する状況を表す。2つのMIFを飽和加算する場合、各MIFの周期のLCMまで飽和加算することで、飽和加算後のMIFを  $MIF_{K,L}(\tau_i, t) = \{(0, 3), (5, 3), 10\}$  と表すことができる。このように加算したいMIFの各凸点が重なる場合には、傾きが1になるよう加算することで、そのすべての影響時間を足し合わせる事が可能となる。

このような演算を用い、従来手法では解析対象となるメッセージが最大で邪魔された後に、その影響が最初に途切れる時間を算出し最大遅れ時間を求める。ここで、最初にその

影響が途切れる時間を Earliest Idle Time(以下、EITと呼ぶ)と定義すると、EITはMIFの傾きが最初に0になる時間を表す。すなわち、図3の例では、 $MIF_{K,L}(\tau_i, t)$ のEITは  $EIT(MIF_{K,L})(\tau_i, t) = MIF_{K,L}[0].y = 3$  となる。

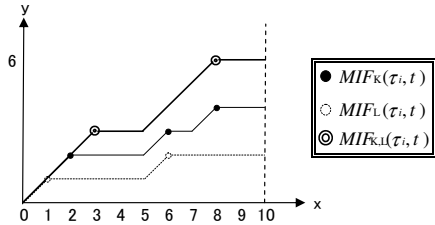


図3  $\tau_1$  と  $\tau_{au2}$  を飽和加算する例  
Fig.3 Examples of a cumulative function

## 2.5 オフセット付きCANメッセージの最大遅れ時間解析手法

従来解析手法では、解析対象となるメッセージ  $\tau_i$  に着目し、ノードと優先度を基準に、全てのメッセージを4つのグループに分類した上で、他ノードの高優先度メッセージから受ける影響を計算している。具体的には、図4で示すように対象メッセージと同じECUから送信される高優先度メッセージ群を  $ihp(i)$ 、低優先度メッセージ群を  $ilp(i)$ 、対象メッセージとは異なるECUから送信される高優先度メッセージ群を  $ohp(i)$ 、低優先度メッセージを  $olp(i)$  と4つに分類する。その上で、他ノードの高優先度メッセージから受ける影響  $ohp(i)$  について、各ECU単位でMIFを導出し、それらのMIFを飽和加算してその最大影響時間を求める。

ここで、対象メッセージとする  $\tau_i$  の最大遅れ時間を求める方法は以下の通りになる。尚、ここでは文献6)で提案されたオフセット付きCANメッセージの最大遅れ時間の解析手法に対して、ECU内の優先度逆転の効果を考慮した部分を削除した手法である。

- (1) 他ECUから送信される全ての高優先度メッセージの最大影響時間を求めるために、メッセージ  $\tau_i$  よりも優先度の高い  $ohp(i)$  に対し、各グループごとにMIFを求め、求められた各グループのMIFを飽和加算し、対象メッセージがすべての他ECUから送信される高優先度メッセージの影響時間の総和を求める。
- (2) 高優先度メッセージが送信要求された時に、すでに低優先度メッセージの送信が開始されている場合、低優先度メッセージ1メッセージ分に邪魔される。この影響を考

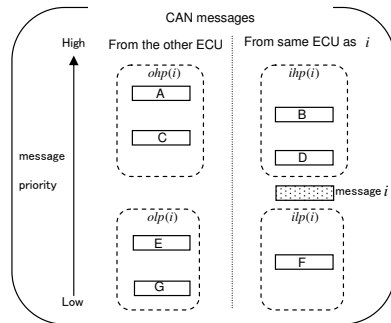


図4 対象メッセージ*i*におけるメッセージ分け  
Fig. 4 Groups of CAN messages based on priority with message *i*

慮するために、他 ECU の低優先度メッセージ  $olp(i)$  のうち最長送信時間分の影響を (1) に飽和加算する。

- (3) 最後に自 ECU 内の高優先度メッセージの影響を加算するため、対象メッセージ  $\tau_i$  の送信要求時刻から  $\tau_i$  より高い優先度のメッセージ  $ihp(i)$  の IF を作成し (2) まで求めた影響時間と飽和加算し、すべてのメッセージからの影響時間を求める。このすべてのメッセージからの影響時間の最初に傾きが 0 になる時刻まで対象メッセージは邪魔された後、送信を開始することができ、この対象メッセージ  $\tau_i$  が送信完了した時間が最大遅れ時間の候補となる。
- (4) 上記 (3) の計算について、対象メッセージ  $\tau_i$  と先行するすべての高優先度メッセージの送信要求時刻から計算を行うことで、すべての最大遅れ時間の候補を導出し、その中で最大の遅れ時間をもつ候補が対象メッセージ  $\tau_i$  の最大遅れ時間となる。

## 2.6 従来手法の問題点

オフセット付き CAN メッセージの最大遅れ時間を導出する際、各 MIF は必要十分条件を満たしているものの、MIF を飽和加算すると必ずしも必要十分条件を満たすとはならず、安全な解析ではあるものの悲観的に解析されてしまう場合がある。この具体例について、表 1 の簡単なメッセージセットを用いて説明する。

表 1 のメッセージセットは、 $G_1 = \{\tau_1, \tau_2, \tau_3\}$ ,  $G_2 = \{\tau_4, \tau_5\}$ ,  $G_3 = \{\tau_6\}$  の 3 つのグループからなる。ここで、対象メッセージを  $G_3$  に属するメッセージ  $\tau_6$  とした場合、他グループ (つまり、 $G_1$  と  $G_2$ ) の高優先度メッセージ群  $ohp(6) = \{\tau_1, \tau_2, \tau_3, \tau_4, \tau_5\}$  であり、

表 1 メッセージセット例  
Table 1 Example of Message Set.

$G_i$	$\tau_i$	$P_i$	$E_i$	$C_i$	$O_i$
$G_1$	$\tau_1$	1	1	25	0
	$\tau_2$	2	2	25	5
	$\tau_3$	3	3	25	16
$G_2$	$\tau_4$	4	4	25	0
	$\tau_5$	5	5	25	7
$G_3$	$\tau_6$	6	6	25	0

前述する方法より、各グループごとに MIF を求め、その上で各 MIF を飽和加算することで、他グループの高優先度メッセージの最大影響時間を導出する。具体的には、グループ  $G_1$  およびグループ  $G_2$  の MIF はそれぞれ図 5 と図 6 に示すとおりであり、 $MIF_{G_1}(\tau_6, t) = \{(0, 3), (9, 1), (13, 1), (15, 1), 25\}$ ,  $MIF_{G_2}(\tau_6, t) = \{(0, 5), (8, 4), 25\}$  となる。このとき、 $MIF_{G_1}(\tau_6, t)$  と  $MIF_{G_2}(\tau_6, t)$  を飽和加算すると、 $MIF_{G_1G_2}(\tau_6, t) = \{(0, 14), (15, 1), 25\}$  となり、対象メッセージ  $\tau_6$  は最初に傾きが 0 になる時間  $EIT(MIF_{G_1G_2})(\tau_6, t) = MIF_{G_1G_2}[0] \cdot y = 14$  まで邪魔されてから送信を開始することができる。

一方、実際に発生しうる最悪状況を調べるためには、文献 7) に定義されるように、すべてのグループにおける全メッセージの送信要求時刻から始まる組み合わせにより解析する必要があり、その組み合わせ数は各グループに存在する送信要求時刻の数の直積により導出できる。つまり、表 1 の例においては、 $G_1$  に存在する送信要求時刻の数  $\times$   $G_2$  に存在する送信要求時刻の数の組み合わせから、そのすべての状況に対し最大遅れ時間の候補を求め、候補の中で最大値を持つ候補が実際に起こりうる最大遅れ時間となる。ここで、表 1 のメッセージセットにおいて対象メッセージ  $\tau_6$  を最も邪魔する状況とは、 $G_1$  の  $\tau_3$  と  $G_2$  の  $\tau_4$  の送信要求時刻から始まる  $IF_3^{16}(\tau_6, t) = \{(0, 3), (9, 1), (14, 2), 25\}$  と  $IF_4^0(\tau_6, t) = \{(0, 4), (7, 5), 25\}$  を飽和加算した場合であり、実際には  $EIT(\max(IF_{G_1 \times G_2}))(\tau_6, t) = 13$  だけ対象メッセージ  $\tau_6$  を邪魔する場合は最悪となる。このように、MIF を飽和加算した結果が実際の最悪状況よりも悲観的に解析されるのは、 $MIF_{G_1}(\tau_6, t)$  の 2 つ目の凸点である (13, 1) は  $IF_2^5(\tau_6, t) = \{(0, 2), (11, 3), (20, 1), 25\}$  の影響から発生する状況であり、実際の最悪状況である  $IF_3^{16}(\tau_6, t)$  では起こりえない影響から導出されているためである。このことより、MIF は各時間における最大影響時間を表しているものの、MIF を飽和加算する場合には、MIF の凸点が実際に起こりうる影響であるかを考慮する必要があるといえる。

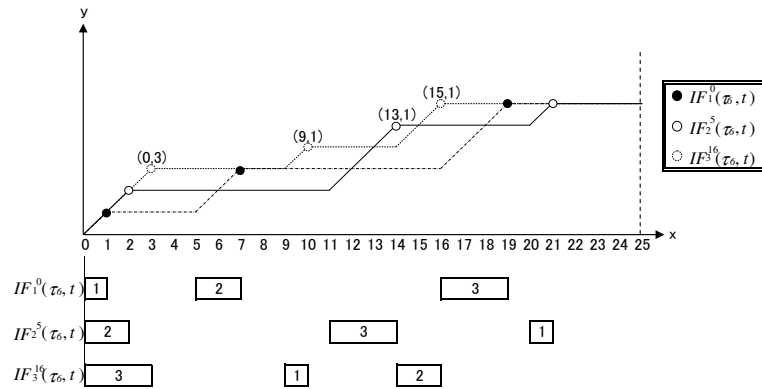


図 5  $G_1$  の MIF  
 Fig. 5 MIF of  $G_1$ .

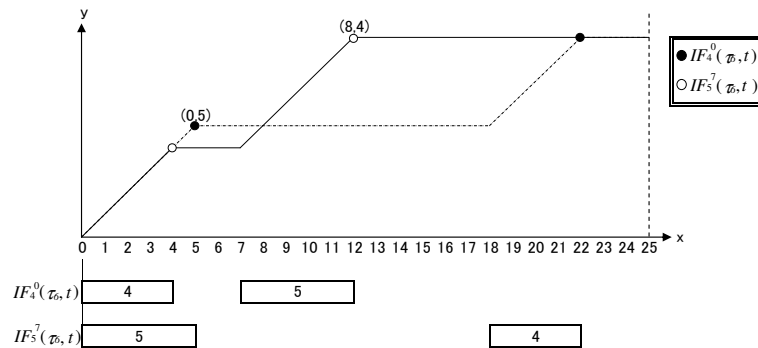


図 6  $G_2$  の MIF  
 Fig. 6 MIF of  $G_2$ .

### 3. 提案手法

本章では、正確なオフセット付き CAN メッセージの最大遅れ時間を導出するためのアプローチについて説明し、その上で正確な最大遅れ時間解析アルゴリズムを提案する。

#### 3.1 アプローチ

従来手法をベースとして最悪状況を特定するためには、前述するように各 MIF の凸点がどの IF から受ける影響かを考慮する必要がある。しかしながら、従来手法では IF から MIF を生成する際に各 IF の凸点の情報は単なる最大影響時間として集約されてしまうため、MIF の各凸点（すなわち、クリティカルフレーム）がどの IF の影響を表すかはわからない。このため、MIF の各凸点がどの IF からの影響を表すものが特定する方法を検討する。仮にこのどの IF からの影響を表すものが特定できれば、特定された IF に対してのみ網羅的に組み合わせ解析を行うことにより、正確な最大遅れ時間を求めることができる。

ここで、IF および MIF を生成する以下の手順に対して、IF を特定する情報を埋め込む方法を提案する。

- (1) 与えられたメッセージ情報から IF を生成する際、IF のすべての凸点には開始時刻情報を付与する。開始時刻情報とは、どのメッセージを開始時刻として生成した IF を表すための優先度（CAN メッセージの識別子）とそのメッセージの送信要求時刻を指す。具体的には、 $IF_{P_j}^{st}(\tau_i, t)$  における  $P_j$  と  $st$  のことであり、各凸点は  $(x, y, P_j, st)$  の 4 つ組で表すことができる。このため、図 5 の各 IF は  $IF_1^0(\tau_6, t) = \{(0, 1, 1, 0), (5, 2, 1, 0), (16, 3, 1, 0), 25\}$ 、 $IF_2^5(\tau_6, t) = \{(0, 2, 2, 5), (11, 3, 2, 5), (20, 1, 2, 5), 25\}$ 、 $IF_3^{16}(\tau_6, t) = \{(0, 3, 3, 16), (9, 1, 3, 16), (14, 2, 3, 16), 25\}$  と表される。
- (2) 各 IF から MIF を生成する際、すべての IF を重ね合わせ、MIF の凸点となりうる開始時刻情報のみを残し、それ以外の開始時刻情報はすべて破棄する。この操作により、図 5 の  $MIF_{G_1}$  は  $MIF_{G_1}(\tau_6, t) = \{(0, 3, 3, 16), (9, 1, 3, 16), (13, 1, 2, 5), (15, 1, 3, 16), 25\}$  となり、図 6 の  $MIF_{G_2}$  は  $MIF_{G_2}(\tau_6, t) = \{(0, 5, 5, 7), (8, 4, 4, 0), 25\}$  となる。
- (3) MIF および IF を飽和加算する場合には、MIF や IF の各凸点に含まれるすべての開始時刻情報を残したまま、凸点を生成する。すなわち、前述の例において、 $MIF_{G_1}(\tau_6, t)$  と  $MIF_{G_2}(\tau_6, t)$  を飽和加算すると、 $MIF_{G_1G_2}(\tau_6, t) = \{(0, 14, \langle(3, 16), (2, 5), (5, 7), (4, 0)\rangle), (15, 1, 3, 16), 25\}$  となり、 $MIF_{G_1G_2}(\tau_6, t)$  の最初の凸点に含まれる開始時刻情報から最大遅れ時間に影響を与える IF を抽出するこ

とができる。

ここで上記3つのオペレータを使うことで、組み合わせが減る効果について説明する。具体的には、前述の表1の例において、すべてのIFの組み合わせを解析する場合には、その組み合わせ数は $G_1$ に存在する送信要求時刻の数 $\times G_2$ に存在する送信要求時刻の数であり、 $3 \times 2 = 6$ 回実行が必要になる。一方、Zuhilyらの手法であるクリティカルフレームのみに限定する場合には、 $MIF_{G_1}$ の各凸点に含まれる開始時刻情報は $IF_2^5(\tau_6, t)$ と $IF_3^6(\tau_6, t)$ のみに限定され、 $IF_1^0(\tau_6, t)$ の場合を計算する必要がなくなり、その組み合わせ数は $2 \times 2 = 4$ 回と減らすことができる。

しかしながら、一般的な自動車に使われるメッセージセットでは、全体で数十個の送信メッセージが存在し、各メッセージの送信周期はデッドラインとして与えられており、その優先度に応じて送信周期がゆるやかに大きくなるよう与えられていることが多い。このため、各ECUの送信メッセージのLCMは表1の例のように送信周期と同一になることは稀で、実際には各メッセージの送信周期よりも非常に大きいLCMになることが多く、そのLCMまでのすべてのクリティカルフレームに対して網羅的に解析することは計算量の観点から無駄が多い。このため、我々は最大遅れ時間に影響を与えるクリティカルフレームのみに限定する方法を提案する。具体的には、上記3つのオペレータを有効にし従来手法で最大遅れ時間を求め、その後、最大遅れ時間に影響を与える開始時刻情報のみに限定し、網羅的に解析する方法である。

### 3.2 提案手法

オフセット付きCANメッセージの正確な最大遅れ時間を解析するアルゴリズムについて、以下に示す。

- (1) ある対象メッセージに対し、第2章で提示する従来手法を用いて最大遅れ時間を求める。このとき、3.1で定義する3つのオペレータを用いて最大遅れ時間を求めることで、対象メッセージの最大遅れ時間に影響を与える高優先度メッセージの開始時刻情報を抽出する。
- (2) 抽出された開始時刻情報に対して、網羅的な組み合わせ解析を実行する。網羅的な組み合わせ解析とは、抽出された開始時刻情報を各送信グループに分類した上で、各グループの開始時刻情報のIFを作成し、すべてのIFの組み合わせから最大遅れ時間を導出する方法である。この網羅的な組み合わせ解析の過程で、(1)からえられた最大遅れ時間と一致する場合には、そのIFの組み合わせ状況をcritical instantとして出力して、次のメッセージの解析へと移る。一方、(1)からえられた最大遅れ時間と一

致しない場合には、(2)でえられる最大遅れ時間のうち、最も大きい最大遅れ時間をもつIFの組み合わせ状況をcritical instant candidateとして保持し、すべてのIFの組み合わせに対して最大遅れ時間を導出する。ここですべてのIFの組み合わせに対して最大遅れ時間を導出した結果、(1)の最大遅れ時間と一致しない場合には、(2)で導出されたcritical instant candidateを真の最大遅れ時間として出力する。

- (3) 解析するすべてのメッセージに対して上記(1)と(2)を繰り返し、解析するメッセージが無くなったら終了する。

## 4. 評価

本手法の評価として、自動車メーカーから提供されたメッセージセットに対して本手法を適用し、各メッセージの最大遅延率を従来手法と比較した。最大遅延率とはメッセージの最大遅れ時間を送信周期で割り百分率で表したものである。

### 4.1 対象と前提

本評価に用いたメッセージセットの概要は、CANの転送速度500kbps、ECU数14個、メッセージ数60メッセージ、バス負荷率は約60%である。尚、本解析では各ノードをそれぞれ1グループとして扱うものとした。

### 4.2 評価結果

自動車メーカーより提供されたメッセージセットを提案手法と従来手法で解析した結果、従来手法で解析する場合には60メッセージ中10メッセージが悲観的に解析されていた。また、悲観的に解析されたメッセージ中、最大で約7%も遅れ時間が過大に評価されているメッセージも存在した。

さらに、我々の手法の有効性を示すために、他ノードの高優先度メッセージを網羅的に解析する際の組み合わせ数について、Zuhilyらの手法(つまり、すべてのクリティカルフレームの組み合わせ数)と我々の手法とで比較した。この結果、Zuhilyらの手法ではあるひとつのCANメッセージの最大遅れ時間を解析するための組み合わせ数が最大で $2.98 \times 10^{20}$ 通り存在する一方、我々の手法を用いた場合には、最大で30720通りに減少させることができ、計算量の観点から本手法が有効であるといえる。

## 5. まとめ

本論文では、従来手法では悲観的に解析される場合の反例について提示した上で、正確な遅れ時間を解析するための手法を提案した。具体的には、限られた範囲のクリティカルフ

レーンに着目し、計算量を減らした上で網羅的な組み合わせ解析を行う方法である。この手法の評価として、自動車メーカーから提供されるメッセージセットに対して適用した結果、従来の手法では悲観的に見積られるメッセージが 60 メッセージ中 10 メッセージ存在しており、本論で提案する手法を用いることで、計算量を抑えつつ正確な最大遅れ時間をえることができた。これより、従来手法に比べ計算量は増えるものの、正確な最大遅れ時間が解析でき、各メッセージの critical instant が得られることができるため、本手法が有用であるといえる。

今後の課題として、より多くのメッセージセットに対して本手法を適用し、その有用性を示す実験が必要である。

謝辞 本研究を進めるにあたり、貴重なご意見をいただいたスズキ株式会社の堂畑克彦氏、増田達彦氏に厚く御礼申し上げます。

#### 参 考 文 献

- 1) International Organization for Standardization, Road vehicles . Controller area network (CAN) . Part1:Data link layer and physical signaling, ISO IS11898-1, (2003).
- 2) Tindell, K. and Burns, A.: Guaranteed Message Latencies for Distributed Safety-Critical Hard Real- Time Networks, Technical Report YCS 229, Department of Computer Science, University of York (1994).
- 3) 飯山真一, 高田広章: システム構成を考慮した CAN の最大遅れ時間解析手法, 情報処理学会論文誌: コンピューティングシステム, Vol.45, No.SIG1(ACS 4), pp.66-76 (2004) .
- 4) M. Grenier, J. Goossens, and N. Navet. Near-optimal fixed priority preemptive scheduling of offset free systems. In Proc. of the 14th International Conference on Network and Systems (RTNS '2006), Poitiers, France, May 30-31, 2006., 2006.
- 5) M. Grenier, L. Havet, and N. Navet. Pushing the limits of CAN - scheduling frames with offsets provides a major performance boost. Proc. of the 4th European Congress Embedded. Real Time Software (ERTS 2008), 2008.
- 6) 飯山真一, 富山宏之, 高田広章, 城戸正利, 細谷伊知郎: オフセット付き CAN メッセージの最大遅れ時間解析, 情報処理学会論文誌: コンピューティングシステム, Vol.45, No.SIG11(ACS 7), pp.455-464, (2004) .
- 7) Zuhily, A. and Burns, A.: Exact schedulability analysis of non-accumulatively monotonic multiframe tasks. Real-Time Syst, 43: pp.119-146, (2009) .
- 8) Takada, H. and Sakamura, K.: Schedulability of generalized multiframe task sets under static priority assignment, Proc. Real-Time Computing Systems and Appli-

cations, pp. 80.86 (1997).

- 9) 飯山真一, 高田広章, 菅沼英明: エンジン制御システムのためのリアルタイム性検証手法, 情報処理学会論文誌, vol.43, no.6, pp.1715-1724, Jun . (2002) .