

スキュー最適化を前提とするデータパス合成における スケジュール可能解空間の拡大

小畑 貴之[†] 金子 峰雄[†]

[†] 北陸先端科学技術大学院大学 情報科学研究科 〒923-1292 石川県能美市旭台 1-1

E-mail: †{t-obata,mkaneko}@jaist.ac.jp

あらまし RT データパス合成において、資源割り当て・演算順序・遅延時間を入力として制御信号のコントロールステップ割り当てを決定する問題がある。スキュー最適化を行う場合にコントロールステップ割り当てが存在するための資源割り当て、演算順序、遅延時間の満たすべき条件について考察を行い、コントロールステップ割り当てが存在するか否かの判定問題が NP 完全であることを確認した。一方、従来から零スキューの下でのコントロールステップ割り当て可能な条件が知られており、これがスキュー最適化を行う場合にコントロールステップ割り当てが存在するための一つの十分条件となっている。これに対して本稿ではこれとは別の十分条件を示すと共に、この条件を用いたスキュー・スケジュール同時最適化に対する発見的解法を示した。

キーワード 高位合成, スキュー, スケジュール

Enlarging The Solution Space For Schedulability Based On Skew Optimization

Takayuki OBATA[†] and Mineo KANEKO[†]

[†] School of Information Science, Japan Advanced Institute of Science and Technology

Asahidai 1-1, Nomi, Ishikawa, 923-1211 Japan

E-mail: †{t-obata,mkaneko}@jaist.ac.jp

Abstract In RT-Datapath synthesis, we sometime encounter a problem to find a control step assignment of control signals (schedule) for a given input instance which consists of DFG, resource binding and execution order of operations assigned to a same resource and delay information. This paper discusses the condition for an input instance (DFG, resource binding, execution order of operations, and delay value) to have a feasible solution (a control step assignment). Recently, it has been reported that skew control as well as control step assignment contributes to improving the performance of a datapath. Regarding the importance of skew control, we especially focus on simultaneous skew and control step assignment problem. One of our contributions of this paper is to show the NP-completeness of the decision problem whether an input instance has a feasible solution or not. The second contribution is a new condition for the input instance to have a feasible skew and control step assignments and a heuristic algorithm for simultaneous optimization of skew and control step assignments based on this condition.

Key words High-level synthesis, skew scheduling, control scheduling

1. はじめに

順序回路の論理レベル設計において、レジスタに供給されるクロック信号のスキュー調整とレジスタのリタイミングを行うことで回路を高速化する研究がなされてきた[1]–[3]。これと同様に、レジスタトランスファーレベルにおいて、レジスタやマルチプレクサに送られる制御信号のスケジュールとスキューを

調整することによる高速化が可能である[4], [5]。

高位合成とは集積回路路上に実装すべきアルゴリズムに対し、スケジュール及び資源割り当てを行って RT データパスに変換する操作であり、ASIC の性能を決定づける非常に重要な設計段階である。高位合成にスキュー最適化を統合する方法は複数考えられる。一つには演算器の遅延のミスマッチによるアイドルタイムを削減することが考えられる。また[6]–[10]に見られ

るようなデータパス合成とフロアプランを統合した最適化に組み込みパス遅延のミスマッチに対応することが考えられる。スキュー調整はクロック周期の短縮、コントロールステップ数の削減などの性能向上だけでなく、遅延変動に対処するにも有効である。

[12]では資源割当て、演算の順序、遅延時間、クロック周期を入力として与え、スケジュール長が最小となるようにスキューとスケジュールの最適化を行う手法を提案している。しかし、このような入力には実現不可能なものがあり、実現可能な入力を生成するためには、実現可能な入力の性質を明らかにする必要がある。本稿では、まず同じ入力インスタンスでもスキュー調整の有無によりスケジュール可能か否かが変わり得ることを示す。次にスケジュール可能か否かの判定問題がNP困難であることを示す。最後に任意のクロック周期に対してスケジュール可能な資源割当て・演算の順序・遅延時間の組み合わせであるか否かの判定法を示す。これは指定されたクロック周期に対してスケジュール可能であることの十分条件であり、従来から知られていたスキュー指定でスケジュール可能である条件と併せることでスケジュール可能解空間を拡張することができる。本稿ではこのような拡張された解空間に対する発見的な手法を提案する。

2. 背景と目的

2.1 回路の構造記述

本稿では、高位合成の入力となる計算アルゴリズムは、Fig.1(a)に示すようなDFG(データフローグラフ)(\mathcal{O}, \mathcal{D})によって表現されるものとする。DFGの頂点 \mathcal{O} はアルゴリズム中の演算の集合である。辺 \mathcal{D} は演算間のデータ依存関係を表す。

入力アルゴリズムは、資源割当てを行うことでデータパスへと変換される。資源割当ては、演算に対する演算器の割当て $\rho: \mathcal{O} \rightarrow \mathcal{F}$ と演算の結果に対するレジスタの割当て $\xi: \mathcal{O} \rightarrow \mathcal{R}$ からなる。なお、 \mathcal{F} は演算器の集合、 \mathcal{R} はレジスタの集合であり、 $\xi(o) = r$ は演算 o の出力がレジスタ r に書き込まれることを表している。回路のデータパスを表現するためには演算器とレジスタの他に結線構造を記述しなければならない。 $(o_i, o_j) \in \mathcal{D}$ となる各演算 o_j について、入力を書き込まれているレジスタ $\xi(o_i)$ の出力端子と演算器 $\rho(o_j)$ の入力端子が接続され、 $\rho(o_j)$ の出力端子とレジスタ $\xi(o_j)$ の入力端子が接続される。簡単なデータパスの例をFig.1(b)に示す。

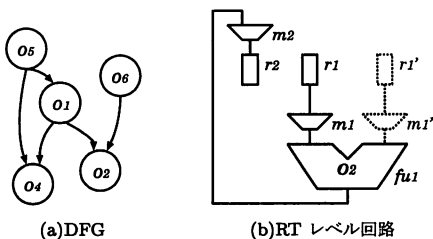


Fig. 1 DFG と RT レベル回路の例

2.2 制御信号とスケジュール

データパス中の全てのマルチプレクサ及びレジスタの集合を \mathcal{M} 、制御信号の集合を \mathcal{S} によって表す。また $c_o^s \in \mathcal{S}$ により演算 $o \in \mathcal{O}$ の実行に関して $x \in \mathcal{M}$ に送られる制御信号を表す。

データパスの動作は制御信号の到着時刻によって決定される。到着時刻は、制御信号の割り当てられたコントロールステップとスキューの値によって決定される。制御信号 $c_o^s \in \mathcal{S}$ に対して、 $\sigma(c_o^s)$ によって割り当てられたコントロールステップを表す。また、 $\sigma: \mathcal{S} \rightarrow \mathbb{Z}_+$ を制御信号のスケジュールと呼ぶ。 $x \in \mathcal{M}$ について、 $\tau(x)$ によって x に割り当てられたスキューを表す。最終的に制御信号 c_o^s の x への到着時刻は $\sigma(c_o^s) \cdot clk + \tau(x)$ と表される。ただし clk はクロック周期である。

2.3 スキュー調整によるスケジュール可能性の向上

Fig.2(a)のDFGに対し、レジスタ割当てを $\xi(a) = \xi(b) = r1, \xi(c) = r2, \xi(d) = r3$ とし、最大/最小遅延時間を有向辺の脇に書かれている値とするとき、スキュー零でスケジュールする事を考える。クロック周期が2のときにはFig.2(b)のようにスケジュール可能である。クロック周期を3とすると、Fig.2(c)のように、演算dの結果はbの結果をr1に書き込むより3ステップ以上後でなければならないが、一方でaの結果を用いてdを行う際の最小遅延が7であるため、bの結果によってaの結果が上書きされるよりも3ステップ以上後ではdの正確な結果はr3の入力端子に存在しない。ゆえにクロック周期3では実行可能なスケジュールが存在しない。これに対して、スキュー調整を行う場合、 $(r1, r2, r3)$ のスキューを $(0, 2, 0)$ とするとFig.2(c)のようにスケジュール可能である。この例では、クロック周期に応じてスキューを調整することにより、いかなるクロック周期でもスケジュール可能である。なおr2-r3間の最大遅延が3以下であればスキュー調整無しでもクロック周期3でスケジュール可能であるが、クロック周期を大きくするとどのような遅延の値であってもスケジュール不能である。

第二の例としてFig.3(a)のDFGをスキュー調整ありでスケジュールすることを考える。資源割当ては $\xi(a) = \xi(b) = \xi(g) =$

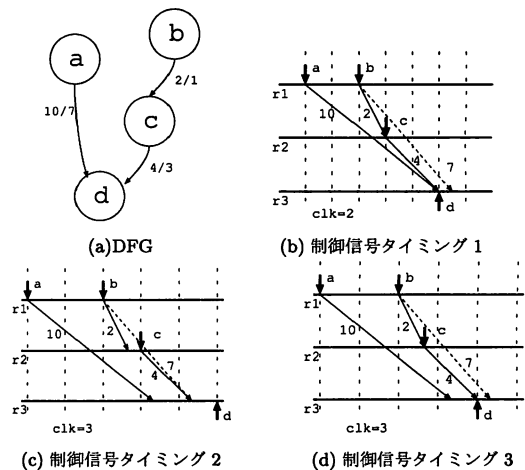


Fig. 2 スキュー調整の有無とクロック周期によるスケジュール可否

$r1, \xi(c) = \xi(f) = r2, \xi(d) = \xi(e) = r3$ とする。クロック周期 3 では Fig. 3(b) のようにスケジュール可能である。次にクロック周期 4 に対応するスキューを考える。DFG の {a,b,c,d} からなる誘導部分グラフによって $r1, r2, r3$ のスキューは $\tau(r2) \leq \tau(r3) + 4k_1 \leq \tau(r2) + 1$ かつ $\tau(r1) + 2 \leq \tau(r3) + 4k_2 \leq \tau(r1) + 3$ かつ $\tau(r1) + 2 \leq \tau(r2) + 4k_3 \leq \tau(r1) + 3$ (但し k_1, k_2, k_3 は任意の整数) に制限されるのに対し, {d,e,f,g} からなる誘導部分グラフからは $\tau(r3) + 2 \leq \tau(r2) + 4k_4 \leq \tau(3) + 3$ かつ $\tau(r3) + 2 \leq \tau(r1) + 4k_5 \leq \tau(3) + 3$ かつ $\tau(r2) \leq \tau(r1) + 4k_6 \leq \tau(3) + 1$ (但し k_4, k_5, k_6 は任意の整数) に制限される。これらの制約を同時に満たすことは不可能であり, クロック周期 4 ではどのようにスキューの値を定めてもスケジュール不能である。

以上のように, 資源割り当て, 演算の順序, 遅延時間, クロック周期によってスケジュールの可否が異なり, また, スキューを零とするとスケジュール不能な場合であってもスキュー調整によりスケジュール可能となる場合がある。

3. 準備

3.1 セットアップ条件とホールド条件

本稿で扱うスキューとスケジュールの同時決定問題は DFG G , 資源割当て ρ 及び ξ , 演算の順序 (関数 $next$ によって同じレジスタ又は演算器に割り当てられた演算同士の順序関係を表す), クロック周期 clk を入力として制御信号のスケジュール σ 及びスキュー τ を出力する。

Fig.4 は演算 o_j に関する制御信号の到着タイミング同士の関係を表している。 o_j の出力はレジスタ r_j に書き込まれるも

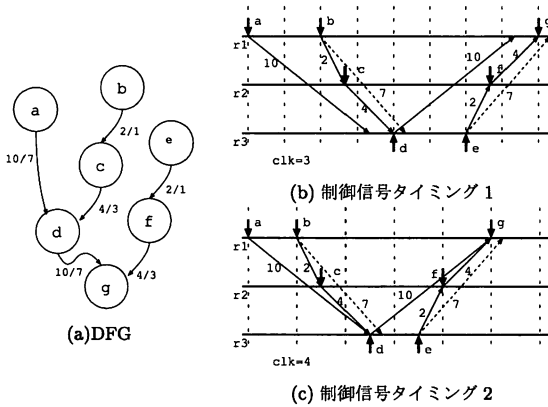


Fig. 3 スキュー調整有りでもクロック周期によりスケジュール不能な例

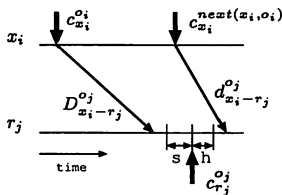


Fig. 4 セットアップ/ホールド条件

のとする ($\xi(r_j) = r_j$). 資源 x_i はレジスタ又はマルチプレクサである。レジスタであるときは, o_i の出力は x_i に書き込まれ, これは o_j の入力の一つである。また $next(x_i, o_i)$ は o_i の次に x_i に結果が書き込まれる演算である。マルチプレクサである場合は o_i は o_j 自身である。 $next(x_i, o_i)$ は, x_i が演算器 $\rho(o_j)$ の入力マルチプレクサであれば o_j の次に同じ演算器で実行される演算であり, r_j の入力マルチプレクサであればその結果が o_j の次に r_j に書き込まれる演算である。

制御信号 $c_{r_j}^{o_j}$ の r_j への到着は o_j の正しい結果が到着した後でなければならないが, この制約はセットアップ条件と呼ばれ, 次の式で表される。

$$\sigma(c_{x_i}^{o_i}) \cdot clk + \tau(x_i) + t_{err} + D_{x_i-r_j}^{o_j} + s \leq \sigma(c_{r_j}^{o_j}) \cdot clk + \tau(r_j) \quad (1)$$

但し $D_{x_i-r_j}^{o_j}$ は o_j を実行するときの x_i から r_j への最大遅延, t_{err} はタイミングマージン, s はレジスタのセットアップ時間である。

また $c_{r_j}^{o_j}$ の r_j への到着は o_j の正しい結果が r_j の入力端子上で書き換えられる前でなければならないが, この制約はホールド条件と呼ばれ, 次の式で表される。

$$\sigma(c_{r_j}^{o_j}) \cdot clk + \tau(r_j) + t_{err} \leq \sigma(c_{x_i}^{next(x_i, o_i)}) \cdot clk + \tau(x_i) + d_{x_i-r_j}^{o_j} - h \quad (2)$$

但し $d_{x_i-r_j}^{o_j}$ は o_j を実行するときの x_i から r_j への最小遅延時間である。

スケジュールを調整できる状況下ではスキューの値は $x \in \mathcal{M}$ について $0 \leq \tau(x) < clk$ と制限しても差し支えなく, 以下このような制限を置く。

3.2 スキュー制約グラフ

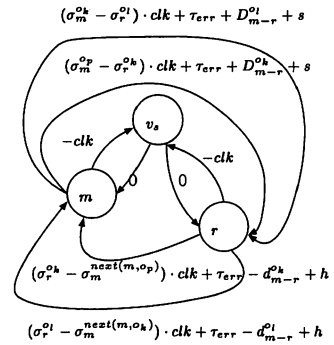


Fig. 5 スキュー制約グラフ

制御信号の到着時刻同士の制約のうち, スキューの制約を表すために有向マルチグラフであるスキュー制約グラフ $G_\tau = (V_\tau, E_\tau)$ を用いる。まず式 (1)-(2) を次のように変形する。

$$\tau_r - \tau_m \geq (\sigma_m^{o_m} - \sigma_r^{o_r}) \cdot clk + \tau_{err} + D_{m-r}^{o_m} + s \quad (3)$$

$$\tau_m - \tau_r \geq (\sigma_r^{o_r} - \sigma_m^{next(m, o_p)}) \cdot clk + \tau_{err} - d_{m-r}^{o_r} + h \quad (4)$$

G_r 頂点は M およびソース頂点 v_s であり、辺はセットアップ条件及 (3) 及びホールド条件 (4) に対応している。(3) に対応する有向辺 $(\tau(r_j), \tau(x_i))$ には重み $(\sigma(c_{x_i}^{O_j}) - \sigma(c_{r_j}^{O_j})) \cdot clk + t_{err} + D_{x_i-r_j}^{O_j} + s$, (4) に対応する有向辺 $(\tau(x_i), \tau(r_j))$ には重み $(\sigma(c_{r_j}^{O_j}) - \sigma(c_{x_i}^{next(x_i, O_i)})) \cdot clk + t_{err} - d_{x_i-r_j}^{O_j} + h$ が付される。また全ての $m \in M$ についての $0 \leq \tau(m) < clk$ という制約を表すために v_s から他の頂点へ重み 0 の辺、各頂点から v_s へ重み clk の辺が付けられる。

スケジュール σ が与えられたとき、このグラフ上で、 v_s から各頂点 m までの最長パス長は、制約式を満たす $\tau(m)$ の値となる。また、正サイクルが存在するとき、即ち最長パス長が計算できないときはその clk 及び σ に対しては制約を満たすスキュー割り当てが存在しないことになる。

3.3 スケジュール制約グラフ

スキュー制約グラフと同様にスケジュールの制約を表すグラフを構成する。 σ が整数であることに注意して、式 (1) と (2) を次のように変形する。

$$\sigma(c_{r_j}^{O_j}) - \sigma(c_{x_i}^{O_j}) \geq \left\lfloor \frac{\tau(x_i) - \tau(r_j) + t_{err} + D_{x_i-r_j}^{O_j} + s}{clk} \right\rfloor, \quad (5)$$

$$\sigma(c_{x_i}^{next(x_i, O_i)}) - \sigma(c_{r_j}^{O_j}) \geq \left\lfloor \frac{\tau(r_j) - \tau(x_i) + t_{err} - d_{x_i-r_j}^{O_j} + h}{clk} \right\rfloor. \quad (6)$$

スケジュール制約グラフ G_σ の頂点 V_σ は制御信号 S 及びソース頂点 v_s である。辺集合 E_σ は式 (5), (6) に対応する有向辺と v_s から他の頂点へ向かう重み 0 の有向辺がある。 τ 及び clk が与えられたとき、 V_σ 上での V_σ から各頂点 x への最長パス長は、制約を満たす $\sigma(x)$ の値となり、最長の最長パス長は CS となる。 CS を決定するパスは、クリティカルパスと呼ばれる。スキューの場合と同様に、 G_σ に正サイクルが存在する場合には制約を満たすスケジュールは存在しない。

4. スケジュール可能性

ある DFG G 、資源割当て ρ 及び ξ 、演算の順序 $next$ 、クロック周期 clk に対して、セットアップ条件及びホールド条件を満たす σ 及び τ の組を求めることができるとき、スケジュール可能であるという。本節では、スケジュールが可能な条件を考察する。

4.1 スケジュール可能性判定問題の計算複雑さ

スケジュール可能性判定問題の NP 完全性を示すにあたって次の補題を証明する。

補題 1. クロック周期を clk とし、最大/最小遅延が全て整数 k を用いて $k \cdot clk$ または $(k + 0.5)clk$ の形で表記できるとき、スキューの値が 0 又は $0.5clk$ となる最適なスキュー及びコントロールステップ割り当てが存在する。

Proof. 任意の最適解 (σ, τ) から、その最適解とコントロー

ルステップ数が同じであってスキューの値が 0 又は $0.5clk$ であるような解へ変換可能であることを示す。コントロールステップ割り当て σ を保ったまま、それに対応するスキュー割り当て τ' を、スキュー制約グラフを構成しその上で最長パス長を計算することで求める。遅延が $0.5clk$ の倍数となっているので、スキュー制約グラフの枝重みもまた $0.5clk$ の倍数であり、したがって最長パス長も $0.5clk$ の倍数である。 (σ', τ') を $\sigma'(c_s^O) = \sigma(c_s^O) + \lfloor \tau(x)/clk \rfloor$, $\tau'(x) = \tau(x) - clk \lfloor \tau(x)/clk \rfloor$ とすると、これは (σ, τ) とコントロールステップ数が同じであってスキューの値が 0 又は $0.5clk$ であるような解である。□

定理 1. スケジュール可能性判定問題は NP 完全である

Proof. 3SAT 問題がスキューとスケジュールの同時最適化問題へ多項式還元できることを示す。3SAT のインスタンス (X, C) に対してスキュー・スケジュール同時最適化問題のインスタンス $(G, \rho, \xi, next, D, d, clk)$ を次のように生成する。ただし $X = \{x_1, x_2, \dots, x_n\}$ は変数、 $C = c_1 \wedge c_2 \wedge \dots \wedge c_m$ は論理式であり、各節 $c_i = (l_{i1} \vee l_{i2} \vee l_{i3})$ について l_{ij} は x_k 又は $\neg x_k$ のいずれかである。Fig.6 に示すように、データフローグラフ G の頂点は、各節 c_i に対応する演算 O_{c_i} 及びリテラルに対応する演算 $O_{l_{ij}}$ と開始演算 O_{ini} 及び O_{c_0} 、終了演算 O_t である。各節 c_i に対して、五つの枝 $(O_{c_{i-1}}, O_{l_{i1}})$, $(O_{l_{i1}}, O_{l_{i2}})$, $(O_{l_{i2}}, O_{l_{i3}})$, $(O_{l_{i3}}, O_{c_i})$, $(O_{c_{i-1}}, O_{c_i})$ を G に加える。また、枝 (O_{c_m}, O_t) , (O_{ini}, O_t) を加える。資源割当ては $\xi(O_{ini}) = \xi(O_{c_i}) = y_1, \xi(O_t) = y_2$ であり、 l_{ij} が x_k 又は $\neg x_k$ であるとき、 $\xi(O_{l_{ij}}) = x_k$ である。また全ての $O_1 \neq O_2$ に対し $\rho(O_1) \neq \rho(O_2)$ となるように ρ を定める。データフローグラフ上で O_{ini} 以外の全ての二点間に有向パスが存在するので、このように ρ を定めれば、 $next$ は O_{ini} を除いて一意に定まる。 $next(y_1, O_{ini}) = O_{c_0}$ とする。 clk を 1 とする。

以上のように構成したスキュー・スケジュール同時最適化問題に対して、スキューの値が 0 または 0.5 であるような最適解を求める。スキュー・スケジュール同時最適化問題の最適解において $\tau(x_i) = 0.5$ となるときに 3SAT 問題において $x_i = 1$, $\tau(x_i) = 0$ のときに $x_i = 0$ と対応させる。 $c_i = (x_j \vee x_k \vee x_l)$ の形になっている C の節の個数を α とする。 $D_{y_1-y_2}^{O_t} = 4m - \alpha + 1$ 及び $d_{y_1-y_2}^{O_t} = 4m - \alpha + 0.5$ とすると、 $\sigma(c_{y_1}^{O_{c_m}}) - \sigma(c_{y_1}^{O_{c_0}}) \leq 4m - \alpha$ となる。また残りの遅延を後述するように定めると、最適解において $\sigma(c_{y_1}^{O_{c_m}}) - \sigma(c_{y_1}^{O_{c_0}}) \geq 4m - \alpha$ であり、等号が成立するとき、かつそのときに限り $C = 1$ となる。この二つを合わせると、スケジュール可能ならば $\sigma(c_{y_1}^{O_{c_m}}) - \sigma(c_{y_1}^{O_{c_0}}) = 4m - \alpha$ となり、この時に限り $C = 1$ となる X への割り当てが存在する。すなわち、スケジュールの可否が $C = 1$ となる X への割り当てが存在するか否かを示す。

遅延割り当ては、まず $D_{y_1-y_2}^{O_t} = 0.5$ とする。すると、最適解では必ず $\tau(y_1) = 0, \tau(y_2) = 0.5$ となる。なぜなら y_1 には G 中の唯一のソース O_{c_0} が割り当てられているので、 $\tau(y_1) = 0$ となる最適解が存在し、加えて $\tau(y_1) = 0, \tau(y_2) = 0.5$ の場合のみ $\sigma(c_{y_1}^{O_{c_m}}) = \sigma(c_{y_2}^{O_t})$ でありそれ以外の場合は $\sigma(c_{y_1}^{O_{c_m}}) + 1 = \sigma(c_{y_2}^{O_t})$ となるからである。次に各節 c_i に

ついて次の四通りに分けて考える。

(1) $c_i = (x_j \vee x_k \vee x_l)$ のとき

$$D_{x-y}^{O_{i,j}} = D_{x-y}^{O_{i,j}} = 0.5, D_{y_1-y_1}^{O_{i,j}} = 3.$$

(2) $c_i = (x_j \vee x_k \vee \neg x_l)$ のとき

$$D_{y_1-y_1}^{O_{i,j}} = D_{x_j-x_k}^{O_{i,j}} = 0.5, D_{x_j-x_k}^{O_{i,j}} = D_{x_l-\neg x_l}^{O_{i,j}} = 1, D_{y_1-y_1}^{O_{i,j}} = 4.$$

(3) $c_i = (x_j \vee \neg x_k \vee \neg x_l)$ のとき

$$D_{y_1-y_1}^{O_{i,j}} = D_{x_j-x_k}^{O_{i,j}} = 0.5, D_{x_j-x_k}^{O_{i,j}} = D_{x_l-\neg x_l}^{O_{i,j}} = 1, D_{y_1-y_1}^{O_{i,j}} = 4.$$

(4) $c_i = (\neg x_j \vee \neg x_k \vee \neg x_l)$ のとき

$$D_{y_1-y_1}^{O_{i,j}} = D_{x_l-\neg x_l}^{O_{i,j}} = 1, D_{x_j-x_k}^{O_{i,j}} = D_{x_j-x_k}^{O_{i,j}} = 0.5, D_{y_1-y_1}^{O_{i,j}} = 4.$$

最適解における $\sigma(c_{y_1}^{O_{i,j}}) - \sigma(c_{y_1}^{O_{i,j}-1})$ の値は、1. については、 $x_j = x_k = x_l = 0$ のとき、すなわち $c_i = 0$ のとき 4、それ以外のとき 3 であり (Fig.7 参照)、2.-4. については、 $c_i = 0$ のとき 5、 $c_i = 1$ のとき 4 となる。

なお最大遅延の制約を満たす σ 及び τ の割り当ては $(O', O) \in G$ となる全ての (O', O) について $\sigma(c_{\xi(O)}^{O'}) \leq \sigma(c_{\xi(O)}^{O(O')})$ となるので、 $d_{y_1-y_2}^{O_{i,j}}$ 以外の最小遅延 d_{σ}^O は $0 \leq d_{\sigma}^O < D_{\sigma}^O$ の範囲でどのように定めてもよい。 $c_{m_{\xi(O)}^O}$ および $\tau(m_{\xi(O)}^O)$ については $D_{m_{\xi(O)}^O}^O - \xi(O) - d_{m_{\xi(O)}^O}^O - \xi(O) < 1$ となるようにすれば上記の性質を満たすように割り当て可能である。

□

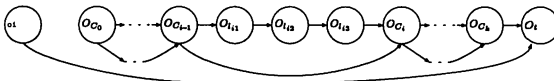


Fig. 6 3SAT に対応するデータフローグラフ

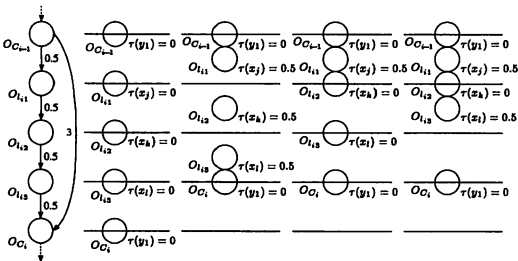


Fig. 7 節 $c_i = (x_j \vee x_k \vee x_l)$ に対応する四つのコントロールステップ割り当て

4.2 任意のクロック周期に対するスケジュール可能条件

任意のクロック周期に対してスケジュール可能であるための条件に関係して、以下に示す補題、定理が得られている [13].

補題 2. clk_1 以上の全てのクロック周期についてスケジュールが存在するとき、全てのクロック周期に対してスケジュールが存在する。

定理 2. $clk > 3|M| \max_{\sigma, \tau} D_{x_1-x_2}^{\sigma}$ となるクロック周期 clk に対してスケジュールが存在するとき、いかなるクロック周期に対してもスケジュールが存在する。

以下 $\max_{\sigma, \tau} D_{x_1-x_2}^{\sigma}$ を D_{max} とおく。

補題 3. $clk > 3|M|D_{max}$ のとき、スケジュール可能ならば、 G_{σ} 中のいかなる強連結成分 C に対しても、 $c_1, c_2 \in C$ となる全ての c_1, c_2 について $\sigma(c_1) = \sigma(c_2)$ となるスケジュール σ が存在する。

補題 4. G_{σ} の強連結成分 C に含まれる 2 頂点 $\sigma(c_{\xi}^O), \sigma(c_{\xi'}^O)$ について、 $|\left(\sigma(c_{\xi}^O)clk + \tau(m)\right) - \left(\sigma(c_{\xi'}^O)clk + \tau(m')\right)| < |M|D_{max}$ である。

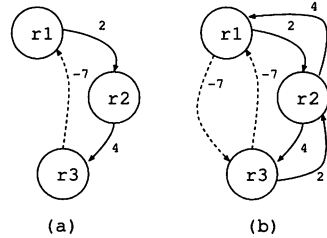


Fig. 8 Fig. 2, 3 に対応する変形スキュー制御グラフ

変形スキュー制御グラフは (V_r, E) と表される。 E は $G_{\sigma} = (V_{\sigma}, E_{\sigma})$ において強連結成分に含まれる辺に対応する G_r の辺からなり、重みは E_r の重みから σ を除去した値である。

定理 3. 全てのクロック周期に対してスケジュールが存在する必要十分条件は、変形スキュー制御グラフにおいて正サイクルが存在しないことである。

Fig. 2 および Fig. 3 に対応する変形スキュー制御グラフを Fig. 8 に示す。Fig. 2 に対応する変形スキュー制御グラフには正サイクルが存在せず、いかなるクロック周期に対してもスケジュールが存在する。Fig. 3 に対応するものには正サイクルが存在し、スケジュール不能なクロック周期がある。

5. 最適化アルゴリズム

(σ, τ) 同時最適化問題は NP-困難 [11] であり、発見的解法 [12] が提案されている。この解法は零スキューのスケジュールを初期解として、スキューの導入とスケジュールの変更により、コントロールステップ数を削減するものである。この解法は、指定されたクロック周期に対して零スキューのスケジュールが存在する入力に対してのみ有効であり、解空間に強い制限があった。本稿では、全てのクロック周期に対してスケジュール可能であるような入力に対する解法を提案する。

σ, clk を固定したとき、スキュー制御グラフにおいて最長パス長を計算することによって τ を求めることができる。また、このとき CS も clk も変化しないから、スキュー制御グラフにおける最長パス長と一致するスキュー割り当てをもつ最適解が存在することになる。このスキュー割り当ては、最長パス長を決定するような枝の両端点の間の相対値としてみると、 $(t_{err} + D_{x_i-x_j}^{\sigma} + s) \bmod clk$ もしくは $(t_{err} - d_{x_i-x_j}^{\sigma} + h) \bmod clk$ となっている。また、最長パス長を決定するような枝集合は、全

域木を構成する。結局 σ, τ 同時最適化問題は、スキュー制約グラフ上において、各頂点に対してひとつの入力枝を選択し、入力点に対して、出力点スキューの相対値を、その枝の重みに応じて $(t_{err} + D_{x_i-r_j}^{\sigma_j} + s) \bmod clk$ もしくは $(t_{err} - d_{x_i-r_j}^{\sigma_j} + h) \bmod clk$ と定めて CS を最小化する問題となる。

スケジュール制約グラフの枝重みを決定する式 (5), (6) の右辺に切り上げがあるので、スキューが $(t_{err} + D_{x_i-r_j}^{\sigma_j} + s) \bmod clk$ もしくは $(t_{err} - d_{x_i-r_j}^{\sigma_j} + h) \bmod clk$ となるときに重みは最小となる。したがって CS を最小化するためには、クリティカルパスのできるだけ多くの枝についてスキューを $(t_{err} + D_{x_i-r_j}^{\sigma_j} + s) \bmod clk$ もしくは $(t_{err} - d_{x_i-r_j}^{\sigma_j} + h) \bmod clk$ とするのが良い。

定理 3 の証明 [13] より、変形スキュー制約グラフに正サイクルがないとき、任意のクロック周期に対してスケジュール及びスキュー割り当てを求めることができる。提案手法では、このようなスケジュールに対するスキュー割り当てを与える G_r 上の全域木 T を初期解とし、 T の枝と G_σ 上のクリティカルパスに含まれる枝を入れ替えながら解を改良する。解の循環を防ぐために、各枝について T の枝として用いられたことがあるか否かの情報を保存し、一度 T から取り除かれた枝は再び加えることを禁止する。また、枝 (u, v) を T に加えることができる必要十分条件は、 v から u へのパスが T に存在しないことである。 T に枝 (u, v) を加えた木を構成するには、 $(w, v) \in T$ となる枝 (w, v) を T から取り除き、 (u, v) を挿入すればよい。提案する解法は Fig.9 のようになる。

Step1. 変形スキュー制約グラフからスキュー r' を求め、クロック周期 $clk' = \lceil 3|M|D_{max}/clk \rceil clk$ に対してスケジュール制約グラフよりスケジュール σ' を求める。全ての $c_m^0 \in C$ に対し $\sigma(c_m^0) = \lceil 3|M|D_{max}/clk \rceil \sigma'(c_m^0) + \lceil r'(m)/clk \rceil$ となるスケジュール σ を求める。

Step2. G_r 及び G_σ を構成する。

Step3. G_r において、 V_s から各頂点への最長パス長を求め、これを与える全域木 T を初期解とする。

Step4. T に対応する τ を求め、 G_σ から σ を求める。これに対応するコントロールステップ数を CS とする。また G_σ において、 CS を決定するクリティカルパスを P とする。

Step5. 各枝 $(u, v) \in P$ に対して、 T に枝 (u, v) を加えた木 $T_{(u,v)}$ を作れるならば、対応する $\tau_{(u,v)}$ を求め、 $CS_{(u,v)}$ を計算する。

Step6. Step5. において、 $T_{u,v}$ が一つも作れないとき、また得られた最小の $CS_{(u,v)}$ が CS より大きいとき、 T に対応する τ 及び σ を出力して終了する。それ以外の時、最小の $CS_{(u,v)}$ と与える $T_{(u,v)}$ を T として Step4. へ。

Fig. 9 発見的解法

6. まとめ

資源割当て、演算の順序、遅延時間、クロック周期が与えられた場合にスキューとスケジュールを求める問題について検討し、この問題に解が存在するか否かの判定問題は NP 完全であることを示した。また、どのような遅延・クロック周期に対してもスケジュール可能であるための必要十分条件を示し、これ

を用いた発見的解法を提案した。必要十分条件は原問題に対する十分条件であり、従来から知られていた別の十分条件と併せることでスケジュール可能解空間を拡張することができる。

今後の課題として、提案手法の有効性を実験によって確認すること、定理 3 を満たすような資源割当て、演算の順序、遅延時間を生成する問題があげられる。

References

- [1] John P. Fishburn, "Clock Skew Optimization", IEEE Trans. on Computers, pp. 945-951, Vol.39, No. 7, 1990.
- [2] R. B. Deokar and S. S. Sapatnekar, "A graphtheoretic approach to clock skew optimization," Proc. IEEE Int. Symp. Circuits and Systems, pp. 1.407-410, 1994.
- [3] Xun Liu, Marios C.Papaefthymiou, Eby G. Friedman, "Retiming and Clock Scheduling for Digital Circuit Optimization", IEEE Trans. on Computer Aided Design, pp.184-203, Vol.21, No. 2, 2002.
- [4] Takayuki Obata and Mineo Kaneko "Control Signal Skew Scheduling for RT Level Datapaths," Proc. of IEICE 18th Karuizawa Workshop on Circuits and Systems, pp. 521-526, April 2005.
- [5] Shih-Hsu Huang, Chun-Hua Cheng, Yow-Tyng Nieh, Wei-Chieh Yu, "Register binding for clock period minimization", Proc. of the 43rd annual Conference on Design Automation, pp.439-444, July 2006.
- [6] J. P. Weng, A. C. Parker, "3D scheduling: high-level synthesis with floorplanning," Proc. Design Automation Conf., pp.668-673, 1991.
- [7] V. G. Moshnyaga, K. Tamaru, "A placement driven methodology for high-level synthesis of sub-micron ASIC's," Proc. Int. Symp. on Circuits and Systems, vol. 4, pp572-575, 1996.
- [8] S. Tarafdar, M. Leeser, Z. Yin, "Integrating floorplanning in data-transfer based high-level synthesis," Proc. Int. Conf. on Computer Aided Design, pp.412-417, 1998.
- [9] K. Ohashi, M. Kaneko, S. Tayu, "Assignment-space exploration approach to concurrent datapath/floorplan synthesis," Proc. Int. Conf. on Computer Design, pp.370-375, 2000.
- [10] M. Kaneko, K. Ohashi, "Assignment Constrained Scheduling under Max/Min Logic/Interconnect Delays for Placed Datapath", Proc. APCCAS2004, vol.2, pp.545-548, 2004.
- [11] Takayuki Obata, Mineo Kaneko, "Computational Complexity of Simultaneous Optimization of Control Schedule and Skew in Datapath Synthesis", IEICE technical report, VLD2006-65, DC2006-52 pp. 83-88, November 2006.
- [12] Takayuki Obata, Mineo Kaneko, "Concurrent Skew and Control Step Assignments in RT-Level Datapath Synthesis", Proc. the 2008 IEEE International Symposium on Circuits and Systems (ISCAS 2008), pp. 2018-2021, May 20, 2008.
- [13] 小畑貴之, 金子峰雄, "スキュー最適化を前提とした実行可能な資源割り当て及び演算順序", IEICE technical report, VLD2008-50, pp.19-24 September 2008.