
高精度用紙搬送シミュレータTIMESの開発

TIMES:High-accuracy Sheet Handling Simulation

藤島 智子* 日吉 隆之** 長門 剛史*** 山下 哲央*
Tomoko FUJISHIMA Takayuki HIYOSHI Tsuyoshi NAGATO Tetsuo YAMASHITA

要 旨

プリンタや複写機の組み込みソフト開発では、用紙搬送制御を模擬するシミュレータを利用した開発が主流になっている。しかし、既存のシミュレータは、複雑なシステムでの用紙搬送制御を高速にかつ高精度に検証することはできなかった。そこで、開発プロセスの上流から下流まで利用可能なメカとソフトの協調シミュレータである“TIMES (Tool for Innovation of MEchatronics and Software co-development)”を開発した。TIMESは高精度な用紙搬送シミュレーションと、本体と数種類の周辺機とから成る複雑な組合せのシステムの検証を実現し、用紙搬送制御検証の効率を大幅に向上させるツールである。本論文では、実際にTIMESをプリンタや複写機の組み込みソフト開発に適用した事例を示し、その有効性を確認した。

ABSTRACT

In developing the embedded firmware for printers and copiers, it has become prevailing to adopt the simulators for sheet handling control. Conventional simulators were, however, unable to verify sheet handling control in a complicated system at high speed with high accuracy. As a solution to this problem, we developed TIMES, Tool for Innovation of MEchatronics and Software co-development, which is utilizable throughout the development process. TIMES substantially improves the efficiency of sheet handling control verification by providing the high-accuracy sheet handling simulation for the complex combinations of a printer/copier and several peripherals. This paper presents the case studies of TIMES effectively applied to actual firmware development.

* プロセスイノベーション本部 デジタルエンジニアリングセンター
Digital Engineering Center, Process Innovation Group

** GJ開発本部 GC開発センター
GC Development Center, GJ Design & Development Division

*** プロセスイノベーション本部 データインテリジェンス推進室
Data Intelligence Division, Process Innovation Group

1. 背景

近年、組み込みソフトの規模が肥大化、複雑化が進み、組み込みソフト産業において高品質ソフト開発技術が大きな課題となっている。特に電子機器製品出荷後の不具合の4割以上が組み込みソフトに起因する¹⁾ため、これらの不具合が市場へ流出しないような開発プロセス・技術が強く求められている。

プリンタや従来の複写機においては、実際の機械（以下、「実機」と呼ぶ）の代わりにPC上でメカの振舞いをする仮想メカモジュールを利用した用紙搬送制御の開発が主流となってきている。しかしながら、近年の高機能化が進んだ複写機・プロダクションプリンタでは、下記3つの要因があり、既存の用紙搬送シミュレータが十分に対応できなくなってきた。

(1) 複写機・プロダクションプリンタでは、制御設計プロセスの最上流の構想段階で、用紙搬送経路とローラ配置といった2次元的な構想設計が実施され、その後、3次元レイアウトを作成し、詳細設計を実施するプロセスが実施されている。3次元レイアウトを基にした用紙搬送制御を検証する力学的なシミュレーションは、用紙の詳細な挙動把握には効果的であるが、シミュレーションの準備や検証に時間が掛かるといった問題がある。また、用紙搬送は機械内部の基本的な部品の配置を決めてしまうため、もし制御設計プロセスの下流である詳細設計段階で変更が入ると上流の構想設計からやり直す必要があり、影響範囲が極めて大きい。例えば、DMU（デジタルモックアップ）の典型的なツールであるVPS²⁾のように3次元のCADデータを活用してシミュレーションする技術では、構想設計段階で何度も用紙搬送制御の検討を行うことは困難である。

このため、より上流で準備や検証に時間を掛け

ることなく用紙搬送制御のシミュレーションを実施できる技術が強く求められてきた。

(2) 単位時間あたりのプリント枚数を、複写機・プロダクションプリンタの生産性と呼ぶ。これは最も重要な製品仕様の1つであるが、特に高速機と呼ばれる高生産のプリンタや複写機では、その正確な推算は困難であった。生産性を決定付ける重要な要因には、用紙搬送速度とともに、先行紙の後端と後行紙の先端の間隔である用紙間隔があるが、用紙の先端位置が容易にシミュレーションできるのに対し、用紙の後端位置は、用紙の“たるみ”や“突っ張り”を考慮したより厳密なシミュレーション技術が不可欠である。従来のエンジンエミュレータ³⁾やG-VPM（Virtual Paper/Path Mechanics）⁴⁾では、用紙後端の“たるみ”や“突っ張り”を考慮することなく先端からの相対位置で計算しているため、生産性検証の精度が不十分であった。

(3) 特にプロダクションプリンタにおいては、本体機に接続する周辺機の種類が多岐にわたり、様々な組合せパターンが発生する。しかし、これらの周辺機の多様な組合せを、あらかじめ全て検証するのは困難である。このため組合せパターンごとに容易に動作を検証する環境構築が求められていた。

そこで、上記課題を解決するために、メカとソフトの協調シミュレータ“TIMES（Tool for Innovation of MEchatronics and Software co-development）”を開発した。TIMESは、近年の高機能化が進んだ複写機・プロダクションプリンタにおいても、開発プロセスの上流で生産性を検証できるような精度と、本体と数種類の周辺機から成る複雑な組合せの検証を実現したシミュレータである。

2. 用紙搬送制御設計プロセスとTIMES

2-1 用紙搬送制御設計プロセス

実機を用いた、従来の用紙搬送制御設計プロセスと手戻りのフローイメージをFig.1に示す。プリント画像を生成するユニットを有する本体機、および装着される周辺機が商品企画として定まると、まず、各機器でメカ制御設計が実施される。そして、メカ制御設計の結果が、用紙搬送制御の要求仕様（以下、「制御要求仕様」と呼ぶ）としてソフト設計者に伝えられる。その後、ソフト設計者は制御要求仕様に基づいてソフト制御設計・ソフト制御実装を実施する。最後に、ソフト制御実装結果を各機器の実機に搭載し、動作検証を実施する。

従来の用紙搬送制御設計プロセスでは、メカ制御設計の結果は、実機による動作検証によってでしか検証することができず、実機でメカ機構や制御の変更に関わる問題が見つかった場合には、メカ制御設計工程やソフト制御設計工程に戻る必要がある。また、3次元レイアウトを基にしたシミュレーションを用いた設計プロセスで開発した場合には、その準備と検証に膨大な時間が掛かるため、上流で何度もメカ制御検証を繰り返すことができない。

次に、TIMESを用いた新たな用紙搬送制御設計プロセスと手戻りのフローイメージをFig.2に示す。メカ制御設計の次工程と、ソフト制御実装の次工程として、シミュレーションによる制御検証の工程を導入する。

制御検証工程により、実機がない状態でも、メカ制御設計とソフト制御設計の妥当性を検証することができ、従来の用紙搬送制御設計プロセスに比べて圧倒的に短時間で、これまでの実機評価工程よりも上流工程での問題の検出と解消ができるようになった。

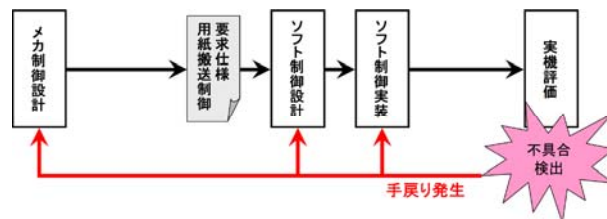


Fig.1 Conventional process to design sheet handling control with an actual machine.

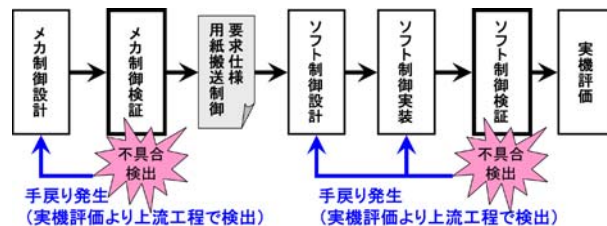


Fig.2 New process to design sheet handling control with TIMES.

2-2 TIMESの概要

TIMESの構成をFig.3に示す。TIMESは、仮想メカモジュールと制御プログラムから成る。

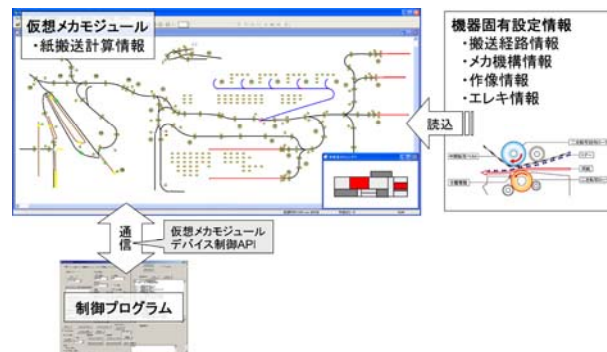


Fig.3 TIMES structure.

Fig.3上部に示す仮想メカモジュールとは、モータやセンサなどのメカ部品を仮想的に表現するためのツールで、仮想メカ部品を操作するための仮想デバイス制御APIを備えている。

Fig.3下部に示す制御プログラムとは、仮想デバイス制御APIを介して仮想メカモジュールを操作するプログラムのことで、各制御検証の工程に応じて構築している。

次節以降に、仮想メカモジュールおよび制御プログラムについて詳述する。

2-3 仮想メカモジュール

仮想メカモジュールには、各機器に共通する紙搬送計算情報が組み込まれており、Fig.3右に示すように機器固有設定情報を読み込むことによって、様々な機種（機器の組合せ）として振る舞うことができる。機器固有設定情報とは、用紙の搬送距離・ローラ位置・センサ位置などの搬送経路情報、モータ・クラッチなどのメカ機構情報、転写ベルト・トナー・駆動源情報などの作像情報、電源・ファン・トナーの内部状態を示すLEDなどのエレキ情報といった、機器ごとに定まる情報のことである。

仮想メカモジュールのGUIイメージをFig.4に示す。仮想メカモジュールは、機器固有設定情報に基づいて実機をある断面で切ったような仮想部品（以下、「断面図」と呼ぶ）や、実機をそのまま見たような外装部品（以下、「外装図」と呼ぶ）を表示することができる。断面図や外装図で表示されている部品を、仮想メカモジュールのGUIを利用してユーザが任意のタイミングで操作すると、その状態を変更させることができる。同様に、制御プログラムによって任意の部品の状態を変更できる。これにより、実機では部品を衝突させて壊してしまうような非現実的なモータの動作状態にしたり、手が届かない位置の部品でも容易に状態変更したりすることが可能となっている。

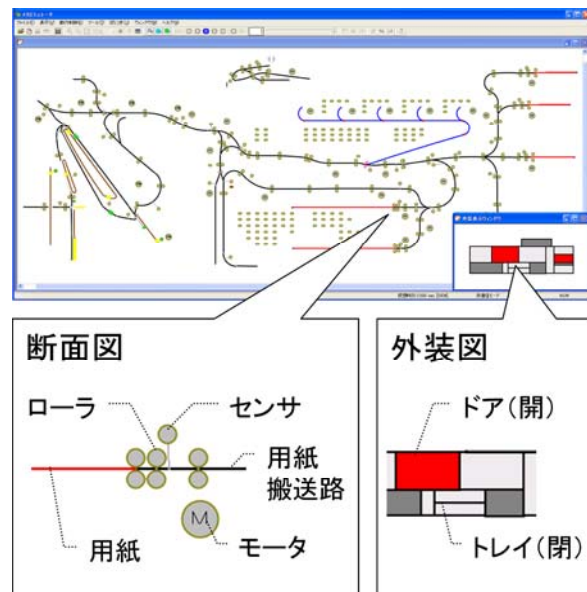


Fig.4 Virtual mechanical modules of TIMES and typical virtual components.

また、仮想メカモジュールは、モータの回転速度などの多値情報やセンサのON/OFF 情報などの2値情報をタイミングチャートとして、用紙の位置情報をダイアグラムチャートとして表示することができる。

2-4 制御プログラム

制御プログラムは、仮想メカモジュールのデバイス制御APIを用いて仮想メカモジュールを操作することができるが、このAPIは、実機のデバイス制御と同等の抽象度で作成してある。例えば、実機ではあるレジスタに値を書き込むと、その値を回転速度としてあるモータを回転させることができるが、その制御に相当するAPIとして、SimMotorOn (double velocity) という関数を提供している。さらに、機器によって組み込みソフトのデバイス制御レイヤの構成が異なるため、機器のソフトウェアアーキテクチャに応じて、種々のAPIを提供している。

実機に搭載されるソフトは、OS (iTron) やASIC (Application Specific Integrated Circuit, カスタムIC) 上で動作しているが、PC上にOSシミュレータやASICエミュレータを用意することで、実機に搭

2-4-2 ソフト制御検証用の制御プログラム

Fig.7に、TIMESをソフト制御検証に利用するための構成を示す。ソフト制御検証においては、前工程で実装した組み込みソフトが、仮想メカモジュールを操作する制御プログラムとなる。具体的には、実装した組み込みソフトのデバイス制御レイヤを仮想メカモジュールのAPIに置き換えることで、実際の組み込みソフトを制御プログラムとして動作させることができる。

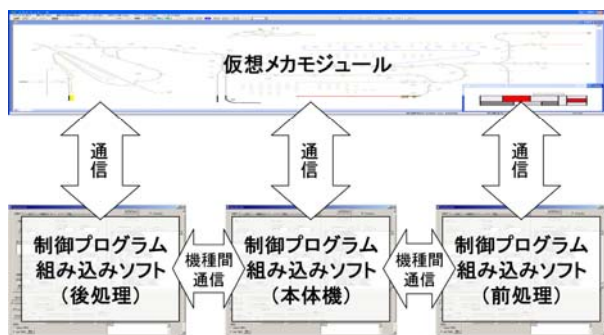


Fig.7 TIMES configuration for design and verification of software control.

Fig.7の例では、複数機器のソフトを組合せた状態でのソフト制御検証にTIMESを利用する構成を示している。各機器の組み込みソフトは、ソフト間での専用通信を行いながら、仮想メカモジュールを操作して動作することができる。

2-5 仮想時刻による同期処理

組み込みソフトは、実時間処理において同期性が保たれているが、シミュレーションにおいては、その速度を下げて制御プログラムのステップ実行を行ったり、シミュレーションを一時停止して仮想メカモジュールの内部状態（用紙の位置、センサのON/OFF状態、モータの回転状態など）を確認したりできることが求められる。

そのため、仮想メカモジュールは仮想時刻という概念で時刻処理を実現した。仮想時刻は、サーバである仮想メカモジュールが管理するタイマ機構を利用する。このタイマ機構に対して、クライアントで

ある各機器用の制御プログラムが、時刻を進めるAPIを呼び出す。全てのクライアントからの命令が揃うと、サーバが単位時刻だけ仮想時刻を進むように設計されている。

Fig.8に、2つの制御プログラムが仮想メカモジュールと同期をとりながらシミュレーションを進めていく様子を示す。Fig.8では、本体機の制御プログラムと周辺機の制御プログラムが、単位時間あたりの自身の処理を行った後時刻を進めるAPIを呼び出している。仮想メカモジュールは、2つの制御プログラムからのAPI呼び出しが揃うまで待機し、揃った段階で時刻を進める処理を実施する。

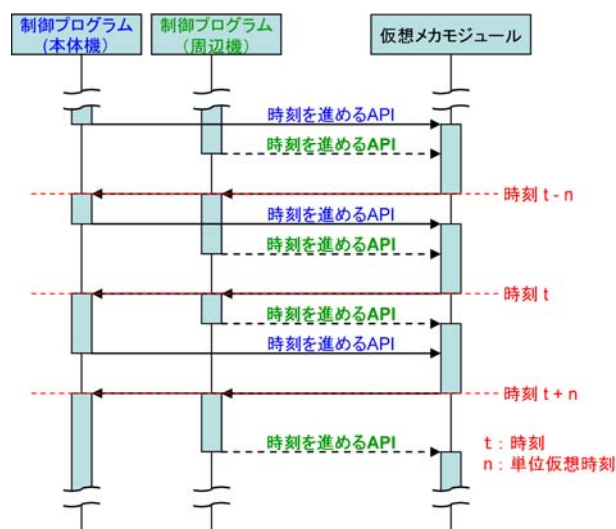


Fig.8 Sequence diagram for virtual time management.

2-6 周辺機の組合せのシミュレーション

TIMESが対象としている複写機・プロダクションプリンタでは、本体機に大量用紙積載共有ユニットや、インサータ（印刷物への表紙の挟み込み）、ソータ（排紙口での仕分け）など、様々な周辺機を追加することができる。周辺機の組合せは、プロダクションプリンタなどの大きな機種では300パターンにも上り、事前に全ての組合せパターンを作成しておくことは現実的ではない。

そこで、複数の機器間で用紙の受け渡しを表現できるようにするために、異なる機器間で用紙が搬送

される境界部分を結合点という仮想的な位置情報として定義し、本体機や周辺機それぞれの機器固有設定情報は別々に作成できる機能を実現した。

結合点を介して、用紙の位置情報に加え、次章で詳述する紙の突っ張りやたるみなどの力学情報も受け渡すことで、紙の精密な搬送速度のシミュレーションを実現している。

本体機・原稿周辺機・前処理周辺機・後処理周辺機の機器固有情報を連結した状態を、Fig.9に示す。結合点を用いた組合せによって、ある周辺機の機器固有情報を入れ替えるだけで、利用者は実機で周辺機を付け替えるように、任意の組合せパターンを仮想メカモジュールを起動することができる。実際にTIMESを利用して開発したプロダクションプリンタの実際の機種イメージと、TIMESの用紙搬送シミュレーションイメージを重ね合わせたものをFig.10に示す。

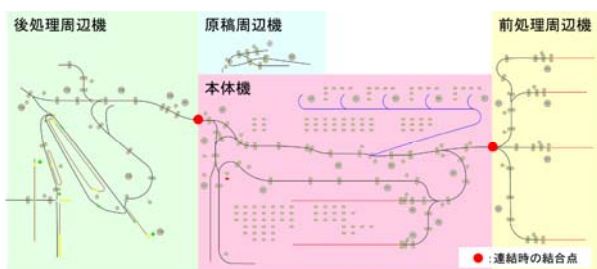


Fig.9 Combination mechanism of TIMES sheet handling simulation.

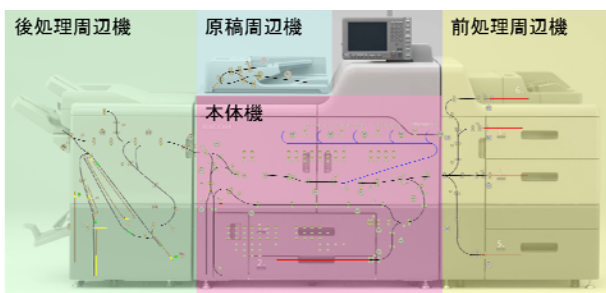


Fig.10 Production printing machine configuration overlaid with TIMES sheet handling simulation.

3. 用紙搬送モデル

生産性の高い複写機・プロダクションプリンタの用紙搬送制御を効率良く検証するためには、高精度かつ高速な用紙先端・後端の位置のシミュレーションが必要となる。

柔軟体である用紙は、複数のローラや搬送ガイド板からの相互作用により変形やスリップを行いながら搬送されるといった特徴がある。このような用紙を力学的に厳密にシミュレーションするためには、竹平^{5,6)}の研究のように有限要素法などを用いた繰返し計算を行う必要があるが、これには膨大な計算時間が必要となり、用紙搬送制御の効率的な検証には利用できない。このため、TIMESでは用紙のスリップを考慮した搬送速度計算において、単純な1次元の実験式を基に搬送路内における用紙のたるみ状態による計算の場合分けを行うことにより、用紙搬送制御の検証に耐えうる高速性と搬送精度を両立したシミュレーションを可能にした。

TIMESでは、用紙は搬送方向に剛体と仮定し、用紙のたわみによる反発力などの力学的影響は無視する。また、用紙をローラに挟まれた用紙区間に区切って考える。各用紙区間には用紙の状態を示すたるみ量 δ というプロパティを設定する。たるみ量 δ は、その用紙区間の両端のローラがその用紙区間に送り込む送り量を、シミュレーション単位時間ごとに加算することで計算される値である。また、少なくとも用紙区間の片方のローラに用紙が接していないときは、たるみ量 $\delta=0$ である。

また、各搬送路区間には、用紙のたるみ、突っ張りの限界を示す、たるみ上限値 (B)、たるみ下限値 (A) というプロパティを設定する。以降、Fig.11を用いて説明する。

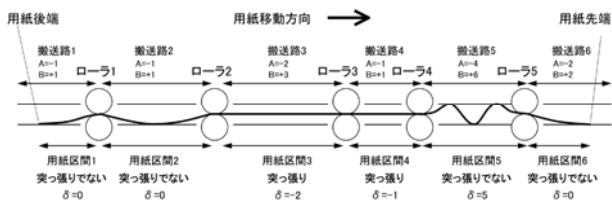


Fig.11 Status of sheets handled between rollers.

たるみ量 δ が0の場合は、用紙区間1, 2, 6のように、用紙が自由に搬送されている状態である。

ローラ2よりローラ1の搬送速度が大きいと、用紙区間2の用紙は用紙区間5のような状態に向けてたるむことになり、用紙区間2のたるみ量 δ は+に増加する。たるみ量 δ が搬送路区間のたるみ上限値に到達すると、用紙は搬送路内でこれ以上たるむことができない状態を意味し、紙詰まり通知などの処理を行う。

また、ローラ2よりローラ1の搬送速度が小さいと、用紙区間2の用紙は用紙区間3のような状態に向かって突っ張ることになり、用紙区間2のたるみ量 δ は-に減少する。たるみ量 δ が搬送路区間のたるみ下限値に到達すると、用紙は搬送路内でピンと張った突っ張り状態を意味し、両端のローラの搬送速度が互いに影響を及ぼす状態となる。

仮想時間ごとの用紙位置は、各ローラによる用紙の搬送速度から算出する。単一のローラによる用紙搬送は、用紙に掛かる搬送抵抗とローラの搬送力とローラの回転線速（ローラ外周の速度）に影響されて、スリップを伴いながら搬送される。

用紙の搬送速度を v 、用紙に掛かる搬送抵抗を R_t 、ローラの搬送力を F 、ローラの回転線速を V とし、実験的に求めた定数 a 、 b 、 c を用いると、用紙の搬送速度は下記のように単純な1次元式で表される。ここで、 R_t 、 F は用紙の種類やローラに依存するパラメータである。

$$v = \{[a-b/(c-R_t/F)]\} V \quad (1)$$

上式は実験結果を同定した式であり、様々な種類の用紙、およびローラで実験した結果をよく表現していることを確認している。

ただし、Fig.11の用紙区間3, 4のように、用紙がローラ間でピンと張った状態（突っ張り状態）では、お互いのローラが影響し合った上、搬送速度が同じになるため、式をそのまま用いることができない。このためTIMESでは、連続した突っ張り状態にある区間に接するローラの用紙搬送速度は、力学的に一体のものとして上記実験式に基づいて近似計算を行う。すなわちFig.11における突っ張り状態である用紙区間3, 4に隣接しているローラ2, 3, 4について、それぞれの単体の用紙搬送速度がローラ2 \leq ローラ3 \leq ローラ4の関係であれば、用紙区間3, 4の突っ張り状態が維持されるので、式の各パラメータを以下のようにまとめてローラ2, 3, 4の搬送速度を計算する。

R_t = 用紙区間2, 3, 4, 5の搬送抵抗の合計

F = ローラ2, 3, 4における搬送力の合計

V = ローラ2, 3, 4の搬送力に応じた回転線速の加重平均

また、単体の用紙搬送速度がローラ2 \leq ローラ3 \leq ローラ4の関係であったならば、用紙区間3でのみ突っ張り状態が維持されるので、ローラ2, 3のみについて上記のようにまとめて計算する。

以上の仮定、アルゴリズムを用いて、高精度かつ高速な用紙先端・後端の位置のシミュレーションを実現した。シミュレーション精度の検証結果は次章に示す。

4. 開発への適用事例

TIMESは、リコー社内だけではなく、国内外のグループ会社の全ての開発に展開され、設計開発の必須ツールとなっている。

シミュレーションの準備や検証に時間を掛けない事例として、3次元の用紙搬送シミュレータとTIMESの準備や検証時間を比較する。3次元の用紙搬送シミュレータでは、準備に約1週間、検証に約2

週間もの時間が掛かっていたが、TIMESでは準備に約1時間、検証に2時間の時間で実施することができるため、大幅に時間短縮を実現した。

シミュレーションの精度の検証として、実機の生産性とTIMESの生産性を比較した結果をTable 1に示す。このTIMESの評価には、試作前の情報のみを用いている。実機とたるみを考慮していない従来のシミュレータの生産性と比較すると、1.5ppm (paper per minutes) の誤差が生じている。実機とTIMESの生産性を比較すると、0.1ppm単位での誤差が生じていない。これにより、TIMESの精度の高さが伺える。

Table 1 Comparison among actual machine, TIMES, and conventional simulator.

	paper per minutes
規格値	55.0
実機	55.3
TIMES	55.3
従来のシミュレータ	53.8

次に、2-6節に示した組合せの場合の効果検証として、事前に全ての組合せパターンを作成した作業者の要した時間(a)と、独立したレイアウトを作成する作業に要した時間(b)を比較した。本体機、原稿周辺機、前処理機、後処理機の各機器が全部で15機器あり、それらを組合せることによって、2-6節の組合せでは通常約300パターンの組合せが存在する。したがって、(a)の作業時間は1パターン平均2時間×300パターン=600時間掛かる。一方、(b)では(a)と同等の評価には15機器分を作成すればよい。このため、(b)の作業時間は1機器平均2時間×15機器=30時間となり、9割程度の作業時間削減を実現した。

障害解析の改善事例として、障害解析手順を比較する。Fig.1に示すような実機を用いたプロセスでは、Fig.12のように、ジャムなどの現象が発生した後にログを取得し、実機で用紙がどのような状態になっているかを推測しながら解析を行っており、用紙搬送状態の推測に長けた熟練の技術者であっても、障害解析に1時間以上の時間が掛かっていた。しか

し、Fig.2のTIMESを用いたプロセスでは、Fig.13のように用紙搬送の状態とそのときのソースコードを確認しながら解析することができるため、ログや機械についての知識がなくても10分程度で障害解析が完了する。このように、TIMESによって障害解析に掛かる工数を大幅に削減することができた。

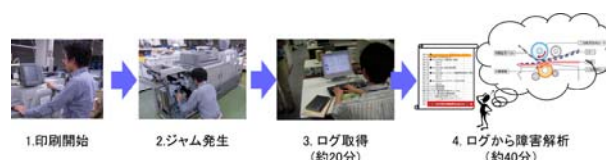


Fig.12 Failure analysis procedure with an actual machine.

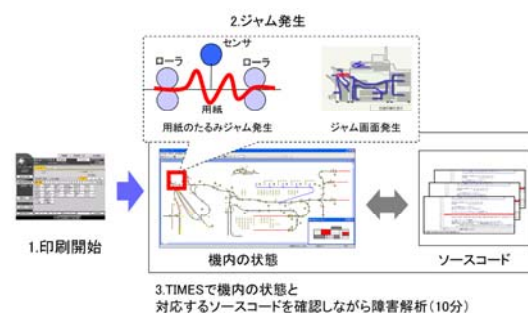


Fig.13 Failure analysis procedure with TIMES.

また、開発プロセスの下流で実施するソフト制御検証は実機を用いて手作業で行っていたため、全ての検証項目を実施するのに3ヶ月もの期間が掛かっていた。また、周辺機の組合せ対応を実施していない従来のシミュレータを用いた検証では、現実には全検証項目のうち1割以下しか実施できないため、品質保証検証期間が長期化していた。TIMESでは、組合せ対応が可能となったことで、8,000程度の項目数の9割以上を実施することができるようになり、不具合が流出しないような開発プロセスを確立し、早期品質確保を実現した。

以下に、TIMESを展開することで、QCD (Quality : 品質向上, Cost : コスト削減, Delivery : 期間短縮) に与える効果の概要を示す。

Q：構想設計段階から利用することで、設計・評価の手戻りを未然に防ぎ、上流での品質を向上させることができた。TIMESでは、メカのない設計値での検証はもちろん、実機のようにメカがばらつくような状態も再現させることで、様々な条件での検証も可能となった。実際、メカのばらつきは、仮想メカモジュールの個々の部品のパラメータを変更することで対応する。例えば、ローラが用紙を搬送する際、実機では用紙の厚みや光沢紙などの種類によって搬送抵抗が変わってくるが、TIMESでは、式(1)のパラメータを変更して表現している。

また、Fig.13に示すように、用紙搬送の状態が一目で分かるため、メカとソフトの設計者がコンカレントに開発するための不可欠なツールとなっている。

C：実機を手配できないことで実施できなかった遠隔拠点での開発も可能としている。特に海外拠点開発では、実機や輸送費用のほか、梱包作業や手続きなどにも費用が掛かっていたが、これらを削減することで、年間あたり10億円以上掛かる海外拠点での開発コストを1億円以上削減できている。

D：実機がなくても検証を開始することができるため、検証開始時期の前倒しを可能とした。また、従来時間が掛かっていた機内の動作や、あるタイミングでのソフトの状態が瞬時に確認できるため、効率的に開発可能になり、実際に開発の全工数を比較した結果、TIMES導入後は導入前の2/3の工数で開発可能となった。また、不具合解析に掛かる工数を大幅に削減した。

5. おわりに

本論文では、複写機・プロダクションプリンタ全体を対象として、不具合が流出しないようなプロセス改革の必須シミュレータであるTIMESを開発した。

TIMESは用紙搬送を高精度で予測し、様々な周辺機との組合せを可能とする機能を実現した。実際の

製品開発に利用することで、9割以上のテスト項目の検証を可能とした。また、実際にTIMESを利用した設計における効果について示した。

TIMESは、汎用性が高く、複雑なメカ機構を持っている様々な機器などへの応用も可能であり、今後は高信頼性な組み込みソフト開発の基幹的ツールとして広く活用していきたい。

謝辞

本システムの開発・展開、ならびに有意義なご助言と議論をしてくださった関係者各位に厚く御礼申し上げます。

参考文献

- 1) 八尋俊英: 組込みソフトウェア産業の課題と政策展開, *Embedded Technology* 2008 (2008).
- 2) 千田陽一, 橋間正芳, 佐藤祐一: 3D CAD データを用いた組込み用ソフトウェア開発支援システムの構築 - 第一報システムの概要と基本設計 -, *情報処理学会研究報告グラフィクスとCAD*, Vol.2002, No.109, pp.1-6 (2002).
- 3) Sky 株式会社: デジタル複合機開発, <http://www.skygroup.jp/software/dcopy/> (参照 2013-07-05).
- 4) ガイオ・テクノロジー株式会社: G-VPM ソフト技術者向け仮想搬送路シミュレータ, http://www.gaio.co.jp/product/dev_tools/pdt_gvpm.html (参照 2013-07-05).
- 5) 竹平修: シート類の運搬シミュレーション, *Ricoh Technical Report*, No.25, pp.125-131 (1999).
- 6) 竹平修: 用紙の挙動搬送シミュレーション, *日本画像学会誌*, Vol.43, No.3, pp.193-201 (2004).