

# ネットワークプリンタにおけるジョブスケジューリングのシミュレーション

2000MT006 有馬 良浩 2000MT045 小島 寛之 2001MT011 黄 京益  
指導教員 後藤 邦夫

## 1 はじめに

現在、企業や大学のような多数のプリンタ利用者に対して、限られた数のプリンタしか設置されていない環境ではネットワークプリンタを共有することが多い。プリンタサーバ設置の目的は、プリンタの状態把握や、印刷ジョブの管理である [3]。しかし、現在のプリンタサーバには複数のプリンタを統括しジョブをスケジュールする機能はない [2]。プリンタサーバは、システム側にとっては役立つが、利用者には全く役立っていない。利用者にとっては印刷までの待ち時間が短いことが重要で、プリンタ処理の向上が必要である。そのために、すべてのプリンタを有効に活用しなければならない。この分野において IPP(Internet Printing Protocol) などの印刷システムの研究はあるが、複数のネットワークプリンタを統括するスケジューリング機構は含まれていない [1]。

そこで本研究では、ジョブスケジューリング機能を持ったプリンタサーバを作成する。Web ブラウザで出力したいファイルを印刷サーバに転送する利用方法を前提に、印刷ジョブを出力先のプリンタに割り当てるスケジューリングを考案する。すなわち、ジョブの待ち行列の形成方法とジョブの配送ルールに改良を加え、ジョブを均等に割り当てることによりプリンタにかかる負荷を分散させる。負荷の分散はプリンタ処理を向上させ、利用者の印刷待ち時間を短くできる。実際に多数のプリンタ実機を用いる実験環境の用意は難しいので、現在のネットワークプリンタ利用モデルと改善後の利用モデルにおけるプリンタのジョブ処理数をシミュレーション実験で比較する。

有馬良浩は主にモデル、ジョブスケジューリングを担当し、小島寛之は主にジョブスケジューリング、シミュレーションを担当し、黄京益は主にシミュレータプログラムの作成を担当した。

## 2 ネットワークプリンタの利用環境モデル

現在のネットワークプリンタ利用における問題点を明らかにする。その上で、ネットワークプリンタを有効活用するための利用環境について説明する。

### 2.1 現在の利用環境

企業や大学では、各ユーザがあらかじめ自分の PC にネットワークプリンタを登録しておくことでどのプリンタも利用できる。利用者 (ユーザ) とシステム側の両方の視点から、現在のネットワークプリンタ利用環境について説明する。まず、利用者の視点では以下の特徴がある。

- 自分のよく知ったプリンタを利用する
- 自分の近くに設置されているプリンタを利用する

次に、システム側の視点では以下の特徴がある。

- 送られたジョブを各プリンタが保持している
- 印刷要求を受けた順に処理をする

また、最近のプリンタサーバ商品には、Web を通してプリンタの稼働状態や処理待ちの印刷ジョブ数を提供するものがあるが、ジョブをスケジュールするプリンタサーバはない [4]。

ユーザがプリンタの状態情報を正確に把握でき、印刷待ちジョブの保有数の少ないプリンタを選定して印刷要求をプリンタに出しているときや、プリンタ台数に対してジョブ数が飽和状態でないときは特に問題なく利用することができる。しかし、問題も多い。

### 2.2 問題点

現在のプリンタ利用を考えたとき、利用者の視点では以下の問題がある。

- 特定のプリンタにジョブが集中し印刷されるまでの待ち時間が長くなる
- プリンタトラブルによるプリンタ停止で印刷できない
- 印刷待ちのないプリンタを選んだが出力場所が遠い
- プリンタの状態情報が専門的な知識であるため理解できない

次に、システム側の視点では以下の問題がある。

- ユーザからの印刷要求が特定のプリンタに集中し過大にジョブが発生する
- プリンタが印刷能力を越えたジョブを保持している

このようなプリンタ利用における問題点を改善する新しいネットワークプリンタ利用モデルを次節で提案する。

### 2.3 提案する利用モデルとサーバの構成

私達の改善策は、ジョブスケジューリング機能を持ったプリンタサーバをネットワーク上に置くことである。まず、ネットワーク上に多数配置されたプリンタの前にプリンタサーバを設置する。その目的は、プリンタサーバでユーザの送るジョブの数や行き先を把握すること、ネットワーク上のプリンタを管理することである。新しいネットワークプリンタの利用モデルを図 1 に示す。

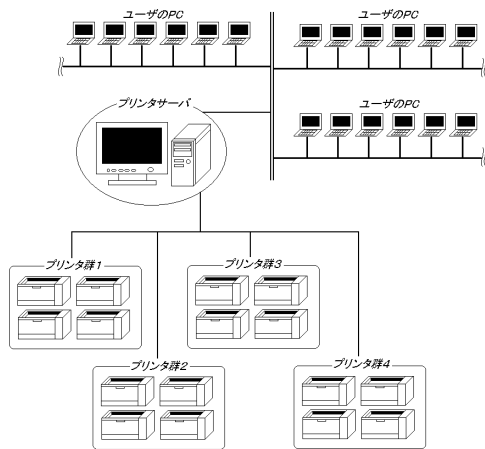


図 1：提案するネットワークプリンタ利用形態

最初に、この利用形態でプリンタに配送されるジョブの流れを示す。ユーザはプリンタサーバの Web 画面を参照し、プリンタ群 (出力場所) を選択、そしてサーバに印刷要求を出す。そして、プリンタサーバはそのジョブを一度スプールし、プリンタの負荷を軽減し、ジョブを均等に配分するようにスケジュールする。すなわち、ネットワークプリンタの稼働状態を把握し、適切なプリンタを選定して自動配送する。

次に、図 2 にプリンタサーバの仕組みを示す。

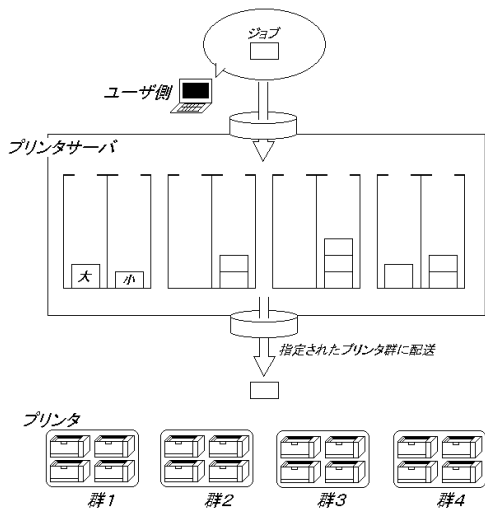


図 2：プリンタ群とスプール

プリンタ群は出力場所に対応する。利用者がプリンタを指定しなくても群を指定すれば、プリンタサーバが決められた場所のなかで、状態のよいプリンタを選定して出力することができる。すなわち、印刷要求のかたよりを防ぐ役割をする。さらに、カラー印刷、モノクロ印刷などの印刷方式により、プリンタを機能別にグループ化するという便利な利用法も考えられる。

スプールはプリンタサーバ中でキューを保持する場所である。本来プリンタキューにたまるジョブをプリンタ

サーバで保持することで、ユーザがプリンタトラブルに巻き込まれないようにする。キューをプリンタ群別に与え、さらに、サイズでジョブを 2 つのクラスに分け、各群にクラス別に 2 つのキューを用意する。

自動配送ルールの詳細を図 3 に示す。

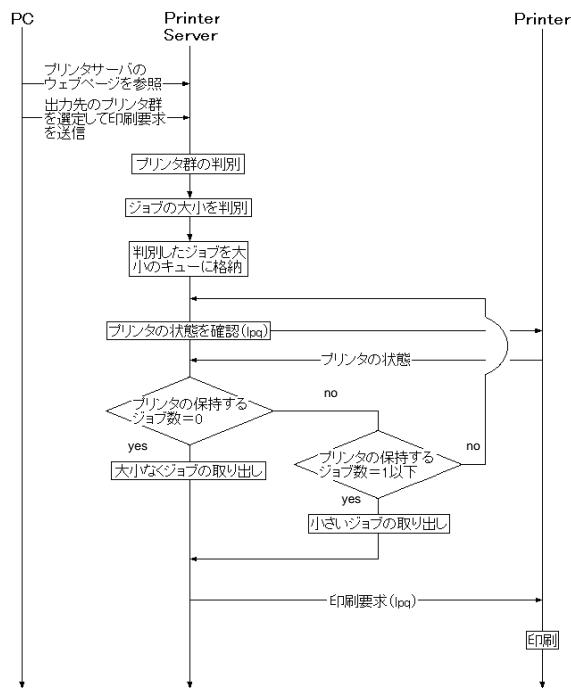


図 3：ジョブの自動配送ルール

本研究のプリンタサーバはジョブを多くためておくことができる。しかし、プリンタがアイドル状態、省電力モードになってしまえば印刷処理は向上しない。したがって、できるだけプリンタがジョブを常に 1 つ保持するようにする。

そこで図 3 のように、プリンタのジョブ保持数が 0 になったときは、各群でキューの先頭にあるジョブを大小のクラスに関係なく先着順に配送する。保持数が 1 つ以下のときは、小さなサイズのジョブのみ配送する。保持数が 2 つ以上のときは、ジョブを配送せず、同群の他のプリンタ状態を同様に確認する。以上を各群で繰り返す。この配送ルールで、プリンタはジョブを常に 1 つまたは 2 つ保持することになる。また、ジョブのサイズを大小に分けたことで、各プリンタにかかる負荷を軽減できる。結果プリンタはアイドルにならず、さらに待ち行列を持たなくても連続して印刷することができる。

#### 2.4 予想される結果

ここでは実験で与えたプリンタ群の 1 つを見て、現在のネットワークプリンタ環境と改善後のネットワークプリンタ環境を比較する。まず、現在の環境での予想される結果を図 4 に示す。

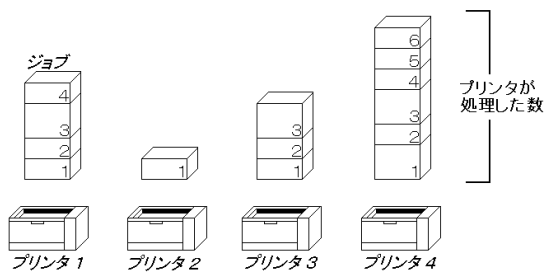


図 4: 現在の環境での結果イメージ

現在の環境では本研究のプリンタサーバを使用しないので、ジョブがユーザの好みに配信され、プリンタの状態にかたよりのある。

次に、改善後に予想される結果を図 5 に示す。

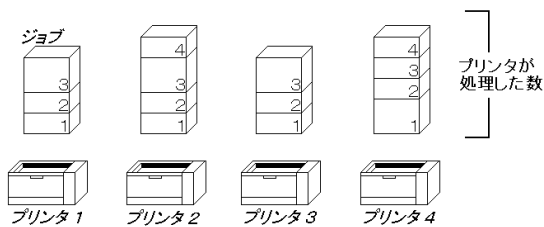


図 5: 改善後の環境でのイメージ

改善後の環境では、スケジューリング機能を持ったプリンタサーバを使用するので、ジョブは均等に割り当てられ、プリンタの状態にかたよりはなくなる、したがって、印刷時間の分散が減少し、アイドル時間が減少するので、平均印刷時間の短縮になる。

### 3 シミュレータの構造

現状のプリンタ利用モデルのシミュレータでは、lpr コマンドを用いてシミュレーション用ジョブを直接仮想プリンタへ送る。改善策を用いたプリンタ利用モデルでは、プリンタサーバにシミュレーション用ジョブを送る。この部分では lpr コマンドは用いず、PF\_INET ソケットを用いて通信をする。lpr コマンドはプリンタサーバから仮想プリンタへ印刷要求を出すときに使用する。

#### 3.1 仮想プリンタ

プリンタ実機の動作に近い仮想プリンタを作成した。すなわち、印刷動作は単にジョブ生成プログラムからランダムなサイズのファイルを受け取り、その実行時間を算出するものではない。

プリンタ実機では、受け取ったファイルサイズにより印刷時間に差が出る。このことをプログラムの実行を遅延させることで表した。

印刷要求の出し方 (lpd の使い方) はプリンタを使用するときの手順に準ずる。lpd での出力先リモートプリンタのキューエントリを記述する/etc/printcap 内に 16 台の仮想プリンタを追加した。

プリンタ実機は故障するので、故障する確率、故障から復旧する確率を与え、プリンタが動作している状態と動作していない状態を仮想プリンタで実現して、シミュレーションを実行した。

#### 3.2 ジョブ生成プログラム

実験のためにジョブを生成するプログラムを作成した。現状のプリンタ利用では、印刷要求を送るプリンタは、ユーザの好みで決められる。しかし、本研究ではジョブ生成プログラムから仮想プリンタへの印刷要求は、一様乱数で無作為に振り分けることとした。

#### 3.3 プリンタサーバ

プリンタサーバシミュレータの構造を図 6 に示す。

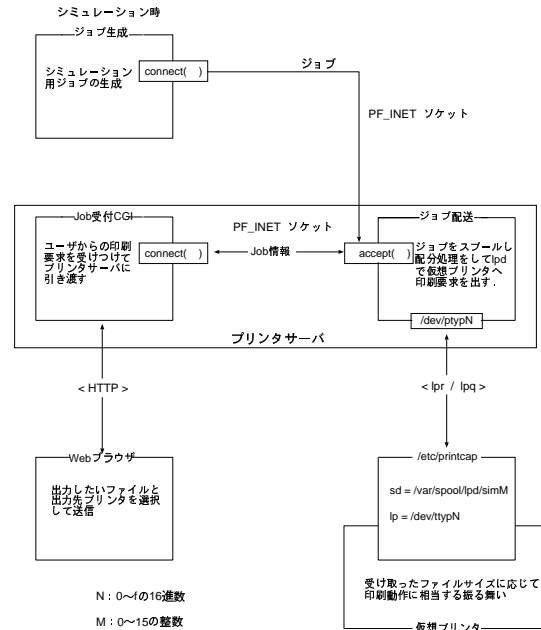


図 6: 改善後のシミュレータ構造

作成したプリンタサーバは 2 つのプログラムからなる。サーバ側でユーザからの印刷要求を受け付ける CGI スクリプト、その情報を受けジョブの均等配分を行う自動配送プログラムである。

サーバの主機能である自動配送プログラムは、常に 1 つだけ働いている必要がある。Web ブラウザからジョブを受け付ける CGI スクリプトは、PF\_INET ソケットを用いて自動配送プログラムへ connect() し、印刷要求の情報を受け渡す。自動配送プログラムは、ソケットからジョブを受け取りキューに格納する。その後、配送ルールに従い仮想プリンタへジョブを自動配送する。このプリンタサーバはジョブを均等に配分し、プリンタにかかる負荷を分散し、プリンタの稼働率を上げる役割を果たす。

### 4 シミュレータを用いた実験結果

実験は、プリンタの故障率 3%、障害復旧率 12%、平均発生間隔 100 ミリ秒、発生ジョブ数 200 個で行った。仮

想プリンタは16台である。ジョブのサイズは平均1500バイトの指数乱数で決定した。

先に、プリンタサーバ設置前のシミュレーション結果を図7に示す。横軸はプリンタ、縦軸は処理したジョブの個数である。

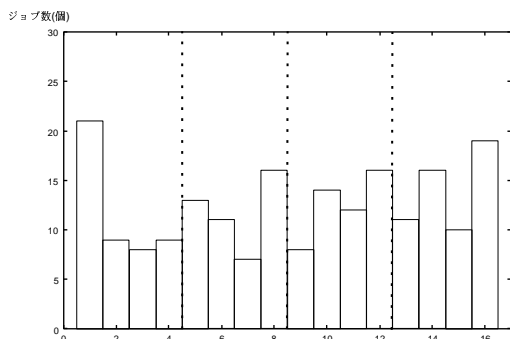


図7: プリンタサーバ設置前の結果 (発生ジョブ数 200 個)

図7では、一様乱数を用いて出力先を決定した。しかし、発生個数が少なかったためかたよりが出た。本研究では、2.2節で述べたような問題点を表すためにこの結果をそのまま使用した。このような状態では、特定のプリンタにジョブが集中し、印刷処理の負荷に大きなばらつきが見られ、印刷待ち時間に大きな差が出る。

次に、プリンタサーバ設置後のシミュレーション結果を図8に示す。

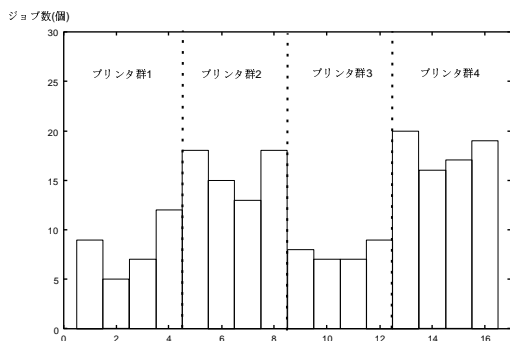


図8: プリンタサーバ設置後の結果 (発生ジョブ数 200 個)

図8では、本研究のプリンタサーバが予想どおりの動作をした結果、ジョブは各プリンタ群に自動配送され、同群のプリンタの処理数に大きな差が出なかった。群単位でプリンタが処理したジョブ数を見ると、すべての印刷要求が完了する直前まですべてのプリンタが稼働している。すなわち、印刷処理を受けるジョブの待ち時間のばらつきを減少させた。したがって、問題点を改善するために置いたプリンタサーバは十分に機能したと言える。

## 5 考察

シミュレーションの結果、本研究でのプリンタサーバのジョブスケジューリング機能は、2.2節で示した問題点の解決に役立った。

## 5.1 仮想プリンタの動作

実際のプリンタでは紙づまり、インク切れなどのトラブルがあり、ユーザが簡単に復旧できるものとそうでないものがある。作成した仮想プリンタではトラブルの種類を区別していないので、実用に向けてはプログラムを改良するか、プリンタ実機を十数台用意しテストをする必要がある。

## 5.2 プリンタトラブル

プリンタサーバのスケジューリング機能は、プリンタ情報を確認してから印刷要求を出すため稼働していないプリンタへジョブを送ることはない。シミュレーションにおいて、現状のプリンタ利用では故障したプリンタにも印刷要求を出しており、プリンタは障害回復を待って印刷動作を再開する。通常のプリンタ利用では、ジョブを多数保持しているプリンタがトラブルによって稼働ができなくなり修復が難しいときに、やむを得ず電源を落としプリンタを初期状態にしてしまう解決策を取ることがある。この場合、プリンタの保持していたジョブは消失する。本研究のサーバ機能は、この状態になったとき印刷処理されず消失するジョブを最小限にとどめることができる。

## 6 おわりに

プリンタ群の考え方は、ユーザが出力場所を指定するが、プリンタまで指定しないので、確定した出力先をユーザへ通知する機能が欲しい。本研究では印刷要求の混雑の改善だけを行ったが、別の視点からネットワークプリンタ利用環境の向上を図ることも必要である。例えば、ユーザ認証とアカウント管理でユーザの管理をすることが重要である [6]。また、本研究における仮想プリンタを改良することにより、印刷性能の異なるプリンタを表現することができ、プリンタの種類による性能差の比較をすることができる。今後の課題は、仮想プリンタの性能やジョブの発生間隔、平均ジョブサイズに与える値を変えて、さまざまなシミュレーションをすることである。

## 参考文献

- [1] Jana Koehler, IPP, <http://www.informatik.uni-freiburg.de/~koehler/ipp.html>, 2000年5月1日
- [2] Novell, Novell NetWare6, <http://www.novell.co.jp/products/netware6/details.html>(accessed2004)
- [3] 柴田良治, オープンプリンティングプロジェクト, <http://www.epkowa3.on.arena.ne.jp/OpenPrintingProject/>, 2004年10月29日
- [4] 片岡巖, ネットワークプリンタ新時代到来!, Software Design, P-1-P-4, 技術評論社, 2004年7月号
- [5] 尾崎俊治, 確率モデル入門, 朝倉書店, 2000年4月
- [6] Gerald Carter, LDAP—設定・管理・プログラミング, オーム社, 2003年