

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2009-173039

(P2009-173039A)

(43) 公開日 平成21年8月6日(2009.8.6)

(51) Int.Cl.
B 4 1 J 29/38 (2006.01)F I
B 4 1 J 29/38テーマコード (参考)
2 C 0 6 1

審査請求 有 請求項の数 3 O L (全 28 頁)

(21) 出願番号 特願2009-71308 (P2009-71308)
 (22) 出願日 平成21年3月24日 (2009.3.24)
 (62) 分割の表示 特願2000-242027 (P2000-242027)
 の分割
 原出願日 平成12年8月10日 (2000.8.10)
 (31) 優先権主張番号 370891
 (32) 優先日 平成11年8月10日 (1999.8.10)
 (33) 優先権主張国 米国 (US)

(71) 出願人 596170170
 ゼロックス コーポレイション
 XEROX CORPORATION
 アメリカ合衆国、コネチカット州 068
 56、ノーウォーク、ビーオーボックス
 4505、グローバー・アヴェニュー 4
 5
 (74) 代理人 100079049
 弁理士 中島 淳
 (74) 代理人 100084995
 弁理士 加藤 和詳
 (72) 発明者 リサ エス. パービス
 アメリカ合衆国 14450 ニューヨー
 ク州 フェアポート ファンリー パーク
 12

最終頁に続く

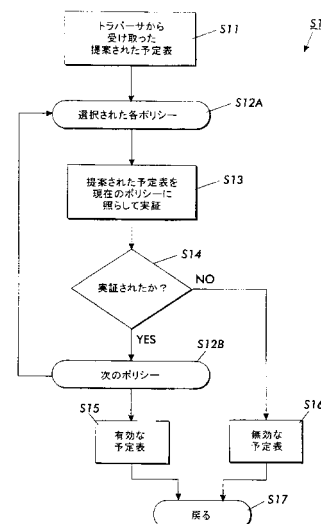
(54) 【発明の名称】 装置内のジョブを処理するために予定表を検出する装置

(57) 【要約】 (修正有)

【課題】 装置内のジョブを処理するために予定表を検出する。

【解決手段】 モデルトラバーサは、サーチアルゴリズムライブラリから選択される第1のサーチアルゴリズムを含み、第1のサーチアルゴリズムは、第1のデータメンバーとしての構成ポリシーと、第2のデータメンバーとしての装置のモデルと、第1のデータメンバーにもとづいて第2のデータメンバーを横断するロジックと、を含む。記憶部トラバーサは、サーチアルゴリズムライブラリから選択される第2のサーチアルゴリズムを含み、第2のサーチアルゴリズムは、第1のデータメンバーとしての選択ポリシーと、第2のデータメンバーとしての記憶された有効な予定表のリストと、第1のデータメンバーにもとづいて第2のデータメンバーを横断するロジックと、を含む。

【選択図】 図4



【特許請求の範囲】

【請求項 1】

(i) 少なくとも 1 つのトラバーサを選択し、(i i) 装置のモデルにもとづいてポリシーのライブラリから該装置に適用可能なポリシーを選択する、セクタを有し、

適用可能なポリシーは構成ポリシーおよび選択ポリシーの少なくとも一方を含み、

少なくとも 1 つの前記トラバーサは(i) モデルトラバーサおよび(i i) 記憶部トラバーサの少なくとも一方を含み、

前記セクタは、適用可能な前記ポリシーとして前記構成ポリシーを選択し、少なくとも 1 つの前記トラバーサとして前記モデルトラバーサを選択し、

前記モデルトラバーサは、前記構成ポリシーにもとづいて前記装置のモデルを横断し、有効な予定表のリストを記憶することができ、

前記セクタは、前記適用可能なポリシーとして前記選択ポリシーを選択し、少なくとも 1 つの前記トラバーサとして記憶部トラバーサを選択し、

前記記憶部トラバーサは、前記選択ポリシーおよびジョブの記述にもとづいて有効な予定表のリストを横断し、該ジョブを処理するために該予定表を探すことができ、

前記モデルトラバーサは、サーチアルゴリズムライブラリから前記セクタによって選択される第 1 のサーチアルゴリズムを含み、

前記第 1 のサーチアルゴリズムは、

第 1 のデータメンバーとしての前記構成ポリシーと、

第 2 のデータメンバーとしての前記装置のモデルと、

前記モデルトラバーサに有効な予定表を報告するために、前記第 1 のデータメンバーにもとづいて前記第 2 のデータメンバーを横断するロジックと、

を含み、

前記記憶部トラバーサは、前記サーチアルゴリズムライブラリから前記セクタによって選択される第 2 のサーチアルゴリズムを含み、

前記第 2 のサーチアルゴリズムは、

第 1 のデータメンバーとしての前記選択ポリシーと、

第 2 のデータメンバーとしての記憶された有効な予定表の前記リストと、

前記記憶部トラバーサに有効な予定表を報告するために、前記第 1 のデータメンバーにもとづいて前記第 2 のデータメンバーを横断するロジックと、

を含み、

前記装置はプリンタであり、

前記ポリシーは前記ジョブを処理するために満たさなければならない基準である、

装置内のジョブを処理するために予定表を検出する装置。

【請求項 2】

前記モデルトラバーサは、

前記構成ポリシーのすべてにしたがって、可能な予定表が有効であるか否かを判断する前記予定表バリデータと、

前記可能な予定表が有効であると判断された場合に、有効な予定表の前記リストに該可能な予定表を追加するモジュールと、

を備える、

請求項 1 に記載の装置。

【請求項 3】

前記記憶部トラバーサは、

有効な予定表の前記リストから選択された可能な予定表が前記選択ポリシーおよび前記ジョブの記述のすべてにしたがっているか否かを判断する予定表バリデータ、

を備える、

請求項 1 に記載の装置。

【発明の詳細な説明】

【技術分野】

10

20

30

40

50

【0001】

本発明は、プリント装置用の、プリント装置の複数のモジュールを通る用紙の処理をスケジューリングするためのスケジューラに関する。詳細には、本発明は、プリント装置を構成するために組合わされるプリンタモジュール及び機能の多様なセットに自動的に適応するスケジューラに関する。

【背景技術】

【0002】

プリント装置を通して処理される用紙は、スケジューリングソフトウェアによって制御されるスケジューラモジュールを用いて各ステージを通してスケジューリングされる。ある手法は、プリントジョブを処理するために満足されなければならない基準を指定するスケジューリングポリシー（例えば、“トレー2の用紙が不足している場合はトレー1を使用”）を用いる。既知の装置では、そのようなポリシーは、プリント装置開発のソフトウェア開発フェーズ中に、スケジューリングソフトウェアの論理として書かれる。ポリシーはスケジューリングソフトウェアの論理としてコード化され、プリント装置の出荷前に、スケジューリングソフトウェアはROMなどのメモリに記憶される。

【0003】

スケジューリング規則及びポリシーは、一般的に、開発中の装置のハードウェアに依存する。例えば、スケジューリングソフトウェアの開発者は、開発期間中に、特定の製品のデュプレックス・ペーパーパス（例えば、両面コピーのプリント用）が、最大で8枚の8.5インチ×11インチ用紙を保持できることを知るであろう。デュプレックス・ペーパーパスはノンストップ・ループなので、これは、デュプレックス・ペーパーパスに入る用紙が、8ピッチ後に再び現れることを意味する。1ピッチは、1つの画像を作成するのに要する時間である。デュプレックス・ループの長さが変わるようにハードウェアが変更された場合は、ソフトウェアも変更される必要がある。

【先行技術文献】

【特許文献】

【0004】

【特許文献1】米国特許第5,701,557号明細書

【特許文献2】米国特許第5,696,893号明細書

【特許文献3】米国特許第5,710,635号明細書

【特許文献4】米国特許第5,604,600号明細書

【特許文献5】米国特許第5,129,639号明細書

【非特許文献】

【0005】

【非特許文献1】フロイダー（Freuder）、ウォレス（Wallace）、「部分的な制約の満足（Partial Constraint Satisfaction）」、人工知能（Artificial Intelligence）、1992年、Vol. 58、頁21～70

【非特許文献2】フリーマン・ベンソン（Freeman-Benson）、マロニー（Maloney）、ボーニング（Borning）、「増分的制約解決器（An Incremental Constraint Solver）」、ACM通信（Communications of the ACM）、1990年、Vol. 33、No. 1、頁54～63

【非特許文献3】マックワース（Macworth）、「関係のネットワークにおける一貫性（Consistency in Networks of Relations）」、人工知能（Artificial Intelligence）、1997年、Vol. 8、頁99～118

【非特許文献4】ミントンら（Minton, S. et al.）、「矛盾の最小化：制約の満足及びスケジューリング問題のためのヒューリスティックな修復法（Minimizing Conflicts: A Heuristic Repair Met

10

20

30

40

50

hod for Constraint Satisfaction and Scheduling Problems)」、人工知能 (Artificial Intelligence)、1992年、Vol. 58、頁161~205

【非特許文献5】デクター (Dechter)、「制約処理のための強化スキーム：バックジャンプ、学習、及びカットセット分解 (Enhancement Schemes for Constraint Processing: Backjumping, Learning, and Cutset Decomposition)」、人工知能 (Artificial Intelligence)、1989年、Vol. 41、頁273~312

【非特許文献6】ガマら (Gamma, et al.)、「設計パターン：再使用可能なオブジェクト志向ソフトウェアの要素 (Design Patterns: Elements of Reusable Object-Oriented Software)」、アディソン・ウェズリー (Addison-Wesley Publishers)、1995年

【非特許文献7】ジェントら (Gent, et al.)、「制約満足問題のためのダイナミックに変化する順序付けヒューリスティックの経験的な研究 (An Empirical Study of Dynamic Variable Ordering Heuristics for the Constraint Satisfaction Problem)」、制約プログラミングの原理及び実践 (Principles and Practice of Constraint Programming)、1996年、LNAI Series #1, 118、頁179~193

【発明の概要】

【発明が解決しようとする課題】

【0006】

既知のシステムでは、スケジューリングポリシーは、スケジューリングソフトウェアの論理へと、予めコード化されるので、スケジューリングソフトウェアは静的で、更新が困難であり、他のプリント装置での再使用はできない。例えば、プリント装置はモジュール設計され、給紙モジュール、マーカ (マーキング) モジュール、及びスタッカモジュールを用いることがあるが、製品リリース後に、新たな機能(capability)を追加するためにマーカモジュールが再設計されるかもしれない。たとえ、モジュール自体は互換性があるように設計されていても、スケジューリングソフトウェアは書き直されなければならない。

【課題を解決するための手段】

【0007】

本発明の一態様は、装置内のジョブを処理するために予定表を検出する装置であって、(i)少なくとも1つのトラバーサを選択し、(ii)装置のモデルにもとづいてポリシーのライブラリから該装置に適用可能なポリシーを選択する、セレクトラを有し、適用可能なポリシーは構成ポリシーおよび選択ポリシーの少なくとも一方を含み、少なくとも1つの前記トラバーサは(i)モデルトラバーサおよび(ii)記憶部トラバーサの少なくとも一方を含み、前記セレクトラは、適用可能な前記ポリシーとして前記構成ポリシーを選択し、少なくとも1つの前記トラバーサとして前記モデルトラバーサを選択し、前記モデルトラバーサは、前記構成ポリシーにもとづいて前記装置のモデルを横断し、有効な予定表のリストを記憶することができ、前記セレクトラは、前記適用可能なポリシーとして前記選択ポリシーを選択し、少なくとも1つの前記トラバーサとして記憶部トラバーサを選択し、前記記憶部トラバーサは、前記選択ポリシーおよびジョブの記述にもとづいて有効な予定表のリストを横断し、該ジョブを処理するために該予定表を探すことができ、前記モデルトラバーサは、サーチアルゴリズムライブラリから前記セレクトラによって選択される第1のサーチアルゴリズムを含み、前記第1のサーチアルゴリズムは、第1のデータメンバーとしての前記構成ポリシーと、第2のデータメンバーとしての前記装置のモデルと、前記モデルトラバーサに有効な予定表を報告するために、前記第1のデータメンバーにもとづいて前記第2のデータメンバーを横断するロジックと、を含み、前記記憶部トラバーサ

は、前記サーチアルゴリズムライブラリから前記セレクトタによって選択される第2のサーチアルゴリズムを含み、前記第2のサーチアルゴリズムは、第1のデータメンバーとしての前記選択ポリシーと、第2のデータメンバーとしての記憶された有効な予定表の前記リストと、前記記憶部トラバーサに有効な予定表を報告するために、前記第1のデータメンバーにもとづいて前記第2のデータメンバーを横断するロジックと、を含み、前記装置はプリンタであり、前記ポリシーは前記ジョブを処理するために満たさなければならない基準である。

【図面の簡単な説明】

【0008】

【図1】本発明の機能ブロック図である。

10

【図2】本発明に従った代表的装置グラフ（装置モデル）の図である。

【図3】本発明に従ったモニタのフロー図である。

【図4】本発明の予定表実証処理のフロー図である。

【図5】本発明のモデル横断処理のフロー図である。

【図6】本発明の記憶部横断処理のフロー図である。

【図7】本発明の戦略選択処理のフロー図である。

【図8】本発明のパワーアップシーケンスのフロー図である。

【図9】本発明の選択時シーケンスのフロー図である。

【図10】本発明に従った簡略化された装置モデルを示す図である。

【図11】本発明に従ったバックトラックアルゴリズムの前方サーチ部のフロー図である

20

。

【図12】本発明に従ったバックトラックアルゴリズムの後方サーチ部のフロー図である

。

【図13】本発明に従った構成可能な横断アルゴリズムを有するモデルトラバーサのフロー図である。

【図14】戦略設計パターンに従って構成可能なアルゴリズムフラグメントを有する記憶部トラバーサのフロー図である。

【発明を実施するための形態】

【0009】

添付の図面を参照して本発明を説明する。

30

【0010】

本発明のスケジューラは、プリンタを通るプリントジョブを、柔軟且つ拡張可能な方法でスケジューリングする。最もシンプルな実施の形態では、本発明は、パワーアップ時に操作され、装置モデルを処理して有効な用紙予定表リストを生成するモデルトラバーサ（横断器）と、有効な用紙予定表リストを処理して所望のプリントジョブの処理に適した1つの予定表を選択する記憶部トラバーサとを含む。モデルトラバーサ及び記憶部トラバーサは両方とも、予定表バリデータ（実証器）を用いて、一般的にデータファイルとして記憶されているプリントスケジューリングポリシーに照らして予定表をチェックする。

【0011】

プリントスケジューリングポリシーは、選択された予定表がプリントジョブを処理するために満たさなければならない基準であり、用紙1枚のプリントジョブに対する予定表は、用紙の処理に必要な装置モジュール並びにモジュールの機能及び制約のスケジュールである。ポリシーは、装置非依存型又は装置依存型であり得る。装置非依存型ポリシーは、全製品にわたって共通となるポリシーである。予定表から出力される用紙の特性は要求されるプリント用紙の特性とマッチしなければならないというマッチングの要件は、装置非依存型基準の一例である。装置依存型ポリシーは、特定の製品の所望の特徴を優先するポリシーである。例えば、大容量給紙装置を有することも有さないこともあるプリンタは、大容量給紙装置を有する場合には大量のジョブに対して大容量給紙装置の使用を優先するが、少量のジョブに対しては優先しないポリシーを有するように設計され得る。しかし、他のプリンタは、大量のジョブに対して、たとえ有していても大容量プリンタを優先しな

40

50

いポリシーを有するように設計され得る。そのような場合には、各製品のポリシーが異なることがある。

【0012】

本発明は、パワーアップ時スケジューリング処理、及びジョブ実行時スケジューリング処理を提供する。パワーアップ時スケジューリング処理（例えば図8）は、戦略セクタ (Strategy Selector) を用いてモデルトラバーサを選択し、目標とする装置の構成及び環境に基いてポリシーのライブラリから構成ポリシーを選択し、モデルトラバーサを用いて、選択されたポリシーに従って装置モデルを処理し、有効な予定表のリストを作成する。

【0013】

ジョブ実行時スケジューリング処理（例えば図9）は、記憶部トラバーサ（ステップS52）、プリントジョブ明細、及び可能な実行時選択ポリシー（例えば実行時基準）を用いて、装置モデルに基き、予定表記憶部に記憶されている有効予定表リストをサーチし、目標の装置でプリントジョブを処理できる予定表を見つける。予定表は、プリントジョブの処理に必要な特定のモジュール及び機能を含む。パワーアップ時に、戦略セクタによって特定の記憶部トラバーサ（ライブラリには複数の記憶部トラバーサが存在し得る）が選択される（ステップS51）か、又は可能であれば、実行時に、特別な戦略セクタが選択される。

【0014】

本発明は、特に、プリント装置におけるプリントジョブのための柔軟且つ拡張可能なスケジューラ構成処理に適用できる。しかし、本発明はより広範な用途を有することが認識されよう。例えば、"プラグアンドプレイ (plug and play)" モジュール互換性を達成しつつ、ジョブの実行のために個々のモジュールをコーディネートする必要がある、自動化された工場生産技術から自動化された農業生産技術までの任意の装置に、本発明が好ましく用いられよう。本発明は、（1）装置の自動的な記述を取得し、（2）一般的には装置モジュールのコーディネーションによって記述されるが本明細書では予定表と呼ぶ、操作の装置モードのリストを（適用可能なポリシーに基いて）決定し、（3）装置によって実行されるジョブに適用可能な装置モードを選択するための、処理を提供する。

【0015】

プリント装置の装置コントローラは、電子的装置モデル (M o M) を作成及び記憶する。M o M は、装置モジュール及びそれらの機能等の表現を含む階層的データ構造である。

【0016】

本明細書に参照として援用するウェブスター (Webster, et al.) の特許文献1は、コントローラ及び複数のモジュールを有する画像処理装置並びに画像処理装置の構成を定義する方法を述べている。本明細書に参照として援用するフロムハーツラ (Fromherz, et al.) の特許文献2は、物理的要件、ソフトウェアインタフェースの要件、及び内部リソースの要件、並びに挙動モデルで構成モデルを指定して、ワークユニットの記述、ワークユニットの変換、時限式イベント、リソース割当て、制約、及び制限で、構成要素の機能を記述する、プリント装置のモデリング方法を述べている。

【0017】

プリントジョブ明細は、用紙サイズ及び、片面又は両面プリントのどちらが要求されているか等の、そこにプリントされるタイプを記述する。本明細書に参照として援用するウェブスターの特許文献3は、プリントジョブ又はドキュメントの表現、及びその表現をコントローラがジョブのプリントに用いることが可能なものに変換し得る方法を述べている。本明細書に参照として援用するウェブスターの特許文献4は、装置非依存型で"即プリント可能 (print-ready)" なプリントジョブ又はドキュメントの表現を述べている。本明細書に参照として援用するデホリティ (DeHority) の特許文献5は、プリントジョブの要件をプリンタの機能と比較するプリンタ構成制御システムを述べている。

【0018】

図1では、プリント装置10は、複数のモジュール12、14、16、マークファシリティ・インタフェース (Mark Facility Interface) 18（概括的にオペレータインタフェ

10

20

30

40

50

ース)、及び装置コントローラ20を含む。例えば、一般的な装置は、装置に供給する用紙を提供する給紙モジュール12と、用紙に画像をマーキングするマーカーモジュール14と、出力された用紙をスタックするスタッカモジュール16とを含む。プリント装置は、装置モデル(MoM)によって記述される。各モジュールは1つ以上の機能によって特徴づけられ、各機能は対応する制約によって特徴づけられる。全てのモジュール、機能、及び制約は、コンポーネント記述言語(CDL)を用いて記述及びコード化されるのが好ましい。CDLは、装置コントローラの再プログラムを必要とせずに、異なる装置間の異なるモジュールの互換を容易にするように、標準化されている。各モジュールは、好ましくは、読取り専用メモリ(ROM)24、及び再プログラム可能揮発性フラッシュメモリ又は消去可能プログラム可能読取り専用メモリ(PROM)26を有するメモリである、モジュール記述メモリ22を含む。各モジュールは、メモリ22へのアクセスを制御するための装置モジュールインタフェース(Machine Module Interface:MMI)と呼ばれるインタフェース28(例えばメモリコントローラ又は汎用プロセッサ若しくはマイクロプロセッサ)を含み、各モジュールは、モジュールの状態を感知するための(例えば"用紙不足"インジケータ)、インタフェース28に接続された1つ以上のマイクロスイッチ30又はセンサを含み得る。各モジュールはCDLを用いて完全に記述され、記述は、装置コントローラによるアクセスのために、モジュール記述メモリ22に記憶される。

【0019】

図2では、装置モデル(MoM)は、装置モジュール#1及び装置モジュール#2を含む。装置モジュール#1は、それが処理可能な消耗品のリストと、入力ポート及び出力ポートのリストを含む。装置モジュール#1は機能のリストも含み、各機能は、処理できる各消耗品に対する制約(例えば寸法)、並びに入力ポート及び出力ポートに参照される通信の入力及び出力の仕様を含む。

【0020】

給紙モジュールは、一般的に、給紙モジュールが供給できる用紙を保持するための複数のトレイを含み、総称的にそれらを機能と呼ぶ。例えば、給紙モジュールは、8.5インチ×11インチ用紙用のトレイ1と、A4用紙用のトレイ2と、8.5インチ×13インチから8.5インチ×14インチまでのサイズの用紙に合うように調節可能なトレイ3とを含み得る。この例示的なケースでは、給紙モジュールはこれらの3つの機能を有することになる。

【0021】

各機能は制約に関して記述される。例えば、トレイ1(8.5インチ×11インチ用紙)については、メモリ22は、"DIM(寸法).x=850; DIM.y=1100"又は同等のコードとして、その中に、制約をコード化してもよい。トレイ2(A4用紙)については、メモリ22は、"DIM.x=826; DIM.y=1171"又は同等のコードとして、制約をコード化できよう。調節可能なトレイ3(8.5インチ×13インチ又は8.5インチ×14インチのいずれかの用紙)については、メモリ22は、長さyが13インチから14インチの間に制限されることを示すために、"DIM.x=850; 1300<DIM.y<1400"又は同等のコードとして、制約をコード化できよう。より普遍的なトレイの制約は、長さyが11インチから14インチの間に制限されることを示すために、"DIM.x=850; 1100<DIM.y<1400"又は同等のコードとして、コード化できよう。トレイの寸法がある範囲の値をとることが可能であることを示す制約に遭遇した場合は、装置コントローラ20は、どこか他の場所を見て最終的に決定された値を見つけなければならない。例えば、コントローラ20は、モジュール記述メモリ22のフラッシュメモリ部26から報告される情報を見ることができる。オペレータが、トレイ内の選択可能フェンスを、正しいサイズの用紙を適切なアラインメントで保持するように設定する場合には、オペレータは、プリント装置の別のモジュール(PSIPモジュール)を用いて、用紙の属性(例えば用紙のサイズ又は色)を指定する。PSIPモジュールは、これらの属性を、装置モジュールインタフェース28(MMI28)を介して給紙モジュール12に報告する。そこから、MMIは情報をフラッシュメモリ26に記憶し、装置コントローラ20に"消耗品"(この場合は用紙)の属性の更新に関する情報を与える。或いは、マイクロスイッチ30が可動フェンスを感知してトレイ内の用紙サイズをMMI28に示し、MMI

10

20

30

40

50

が装置コントローラ 20 に"消耗品" 属性の更新に関する情報を与える。

【0022】

ホームオフィスはトレイが1つか2つのプリンタを有するかもしれない。中規模のプリントショップはトレイが3つか4つの給紙モジュールを有するかもしれない。しかし、大規模のプリントショップは、20個以上のトレイ、又は、個々に複数のトレイを有する5、6個の給紙モジュールを望むかもしれない。装置におけるモジュール数及び給紙モジュールにおけるトレイ数は様々であろう。用紙のサイズ、重量、及び用紙タイプ（例えば、透明フィルム又は紙、用紙の色、レターヘッド、タブ付きシート、封筒等）によって、たちまち20個以上のトレイが必要になろう。用紙サイズなどといったトレイ機能に対する幾つかの制約は固有の特徴であり、これらは、モジュール記述メモリ 22 の ROM 部 24 に記憶される機能及び制約に記述される。

10

【0023】

しかし、トレイの固有の特徴は、装置オペレータがどのようにトレイに（用紙を）入れるかを定めないのである。例えば、トレイに入れられる用紙の色はトレイの固有の特徴ではない。これらの、オペレータが機能に対して課す制約については、オペレータがモジュールの制約をパーソナル化しなければならない。オペレータは、用紙重量及び用紙のタイプ（例えば、透明フィルム又は紙、用紙の色、レターヘッド、タブ付きシート、封筒等）をパーソナル化できる。装置のオペレータは、これらの選択をプリント装置の別のモジュール（PSIPモジュール）を介して装置に入力して、トレイ内の用紙のパーソナル化属性を指定する。

20

【0024】

PSIPは、装置のオペレータが、プリンタのパーソナル化、診断の開始、ジョブの提出等のために用いるインタフェースである。PSIPモジュールは、装置コントローラの全体的なワークフローを上流に向かって処理する、プリントシステムのソフトウェアコンポーネントである。PSIPモジュールは、装置モジュールインタフェース 28（MMI 28）を介して給紙モジュール 12 にパーソナル化属性を報告する。そこから、MMI は情報をフラッシュメモリ 26 に記憶し、装置コントローラ 20 に"消耗品"の属性の更新に関する情報を与える。

【0025】

同様に、スタッカモジュール 16 の機能及び制約は、スタッカモジュールにおいて使用可能な丁合い、及び綴じ又はホチキス止めパターンを定義するであろう。機能は、サイズの制約を有する特定のスタッカトレイを含み得る。スタッカモジュールは、左上の角のホチキス止め、又は左余白若しくは上余白に2回のホチキス止めが可能であるかもしれない。シートの向きを横又は縦にしてホチキス止めするバリエーションも可能であろう。先進的な機能としては、他の綴じ技術、シュリンク包装、様々な折り方の形式（新聞セクションのように）等が挙げられよう。スタッカモジュールの機能及び制約は、CDLで記述され、モジュール記述メモリに記憶される。

30

【0026】

マーカモジュール 14 の機能及び制約は、マーカモジュールから入手可能なプリントオプションを定義するであろう。例えば、マーカモジュールは、8.5インチ×11インチ又はA4の用紙を片面モード（表面のみ）、両面モード（左余白に沿って綴じたときに読み易いようにアレンジされた表面及び裏面）、及び逆転両面モード（上余白に沿って綴じたときに読み易いように裏面が逆さまになった表面及び裏面）でプリントできるかもしれない。しかし、マーカモジュールは、11インチ×17インチを片面モードだけでプリントできるということもある。マーカモジュールの機能は、拡大又は縮小の量の制約によって定義される拡大及び縮小フィーチャ（機能）を含むこともある。これらの機能も、CDLを用いて、それらの制約によって記述され、モジュール記述メモリに記憶される。

40

【0027】

各モジュールは、ポイント対ポイント接続で、又は好ましくは全モジュールが接続される共通バス 32 を介して、装置コントローラ 20 に接続される。装置が最初にパワーアッ

50

プされる時に、全モジュールはそれらの機能及び制約をコントローラに伝え、コントローラは装置モデルを構築する。例えば、給紙モジュールのモデルは、

(TRAY1)"DIM.x=850; DIM.y=1100"

(TRAY2)"DIM.x=826; DIM.y=1171"

(TRAY3)"DIM.x=850; 1300<DIM.y<1400"

となり得、装置コントローラ内に記憶される。装置がパワーアップされる度ごとに装置モデル(MoM)が構築されるので、同じコントローラを有する装置に異なる給紙モジュールを用いてもよい。モジュール記述メモリの内容をコントローラに転送するために、コントローラは、幾つかの許容モジュールステーションアドレスに対してポーリングして、モジュールステーションが応答したら内容をアップロードしてもよい。或いは、パワーアップ時に、各モジュールがその記述メモリの内容を転送するためのアクセスを要求してもよく、コントローラが要求の競合を解消してから転送を命令してもよい。装置のトータル又はダイナミックモデルは、"給紙トレイ1が空"、マーカの準備ができていない、又は"スタックトレイ3が満杯"、などといった装置状態パラメータも含み、それらのパラメータは、個々のモジュール又は装置コントローラの動作中に、センサによってピックアップされるか又は制御信号(例えば準備完了又は準備できていない)によって生成される。

【0028】

装置コントローラ20は、好ましくは、プロセッサ34(例えば、マイクロプロセッサか又は1つ以上のASIC即ち特定用途向け集積回路に実装され得る特殊な論理のいずれか)と、オペレータインタフェース18と通信するためのインタフェース36及びバス32と通信するためのインタフェース38とを含む。装置コントローラ20はメモリ40も含み、メモリ40は、ROM、RAM、及び消去可能PROM即ちフラッシュメモリを含み得る。メモリ40は、好ましくはROMに、複数の名前付きモジュールとして組織された制御プログラムを記憶する。装置コントローラ20は、独立型モジュール内であってもよく、他のモジュール(例えばモジュール12、14、16)のうちの1つに収容された回路基板であってもよく、又は装置内のどこかに好ましく実装された回路基板の一部であってもよいことを、当業者は認識するであろう。

【0029】

メモリ40は、1つ、又は他の形式のデータも記憶する。永久的なデータはROMに記憶され、一次的なデータはRAMに記憶される。装置の作動中、装置モデル(MoM)が構築され、好ましくはRAMに、又は、一部がRAMに、一部が消去可能PROMに記憶される(パワーアップ)。装置オペレータのパーソナル化操作によって生じたパラメータは、一般的に、消去可能PROMに記憶される。更に、本明細書により詳細に述べるように、装置は、好ましくはROMに記憶されるポリシーのライブラリを有して出荷される。しかし、戦略セレクトと呼ばれるプログラムモジュールは、装置モデルに適したポリシーを識別し、選択したポリシーへの参照を好ましくは消去可能PROMに記憶する。

【0030】

これらの教示に照らして、本発明をサポートするためにメモリ40の他の構成を構成してもよいことを、当業者は認識するであろう。例えば、プログラム及びデータの一部又は全てが、CD-ROM又は磁気フロッピー(登録商標)ディスクに記憶され、装置コントローラ20内部の磁気ディスクに転送され、そこからRAMにロードされてもよい。

【0031】

戦略セレクトは、本明細書で更に説明するように、他のモジュール及びポリシーを条件的に追加又は削除する条件ステートメント(例えばif-then-elseステートメント)を含むソフトウェアモジュールである。例えば、パワーアップ(時)スケジューリング処理(例えば図8のステップS42)では、戦略セレクトは、モデルトラバーサを選択して、目標とする装置の構成及び環境に基き、ポリシーのライブラリから構成ポリシーを選択する。一方、選択時スケジューリング処理(例えば図9のステップS51)では、選択セレクトは記憶部トラバーサを選択して、選択ポリシーを選択する。1つの戦略セレクトを、(i)パワーアップ時に用いるモデルトラバーサ及び構成ポリシーの選択、及び(ii)選択

10

20

30

40

50

時に用いる記憶部トラバーサ及びパワーアップ時の選択ポリシーの選択の両方に用いることができる。

【0032】

戦略セレクトタを、例えばモデルトラバーサ（ソフトウェアモジュール）及び構成ポリシー（if-then-elseステートメントでできている同じくソフトウェアモジュール）のメモリへのロードをメモリローダに指示するものとして、考えることもできよう。モデルトラバーサは、特定のメモリロケーションで始まり、特定数のバイト分だけ続いてロードされる。同様に、各構成ポリシーは、特定のメモリロケーションにロードされ、特定数のバイト分だけ続く。戦略セレクトタのメモリローダ機能（例えば、ユーティリティソフトウェアモジュール）は、ロードされたモジュールがどこに記憶されたかを記録し、構成ポリシー及び装置モデル（M o M）がどこに記憶されているかをモデルトラバーサが知るように、戦略セレクトタは必要なロケーションでモデルトラバーサを構成する。このようにして、モデルトラバーサは構成ポリシーに基き装置モデルを横断できる。記憶部トラバーサ及び選択ポリシーのロードにも同様のロード手順が用いられる。

【0033】

本発明の異なる実施の形態は、モニタと協働して、スケジューラを装置の現在の状態に適応させるための情報を取得する。モニタは、装置状態（例えば、マーカの準備ができていない）、装置性能（例えば、最初のコピー出力に対して経過した時間）及び/又は装置環境（即ち、割増し料金顧客ジョブのプリント）をモニタするハードウェア要素及び/又はソフトウェアモジュールである。モニタがソフトウェアモジュールである場合は、本発明の実施の形態は、戦略セレクトタを用いて、本明細書に述べるようにモニタモジュール及びモニタポリシー、又は他のモジュールを選択する。

【0034】

次に、まだパワーアップシーケンスにある間に、装置コントローラはモデルトラバーサを用いて、選択された構成ポリシーに従って装置モデルを処理し、有効予定表リストを作成する。後で、ジョブ明細を受け取ったときに、装置コントローラは記憶部トラバーサを用いて、選択ポリシー及びプリントジョブ明細に基き有効予定表リストをサーチし、目標とする装置でプリントジョブを処理できる特定のモジュール及び機能を含む予定表を決定する。

【0035】

一般的に、トラバーサは、装置コントローラ20を制御して1つ以上の有効予定表をサーチするプログラムモジュールのうちの1つのモジュールである。本発明は、例えばメモリ40（図1）に記憶されるライブラリに含まれる複数のタイプのトラバーサを用いる。1つのタイプのトラバーサはパワーアップ時に用いられるモデルトラバーサである。モデルトラバーサは、本明細書で述べるように複数のタイプのアルゴリズムを用いて実装されてもよく、パワーアップトラバーサ（又は構成トラバーサ）と呼ばれることもあり、装置モデル（M o M）に基き全ての可能な予定表のリストを作成し、構成ポリシー及びM o Mに基き、どの予定表が有効であり、どの予定表が無効であるかを決定するために用いられる。

【0036】

一般的なパワーアップバージョン（即ち、構成バージョン）では、モデルトラバーサは、モジュール及び機能の制約及び相互接続によって各サイクルについて異なる可能な予定表（即ち、モジュール及び機能の独自の組み合わせ）を特定することを可能とされた、モジュール及びそれらの機能の全ての組合せ及び順列を通してループする。各サイクルでは、モデルトラバーサは予定表バリデータ（本明細書で述べる）を用いて、可能な予定表が有効であるか無効であるかを決定する。

【0037】

好ましいバージョンでは、モデルトラバーサは、実際に横断を行うために用いられるアルゴリズムから分離できる。この分離により、本明細書で更に述べるように、システム内への異なるアルゴリズムの入れ替えが容易になる。モデルトラバーサは、戦略セレクトタに

よって、特定のサーチアルゴリズムを用いるか、又は含むように構成される（本明細書で述べる）が、モデルトラバーサ自体は、いかなる特定のアルゴリズムへの依存関係からも分離される。

【0038】

戦略セレクトタを、例えば、モデルトラバーサ（いかなる特定のアルゴリズムへの依存関係からも分離されたソフトウェアモジュール）及びモデルサーチアルゴリズム（同じくソフトウェアモジュール）のメモリへのロードを、メモリローダに指示するものとして、考えることもできよう。モデルトラバーサは、特定のメモリロケーションで始まり、特定数のバイト分だけ続けて、ロードされる。モデルサーチアルゴリズムは、特定のメモリロケーションに、特定数のバイト分だけ続けてロードされる。戦略セレクトタのメモリローダ機能（例えば、ユーティリティソフトウェアモジュール）は、ロードされたモジュールがどこに記憶されたかを記録し、モデルサーチアルゴリズム、構成ポリシー、及び装置モデル（M o M）がどこに記憶されているかをモデルトラバーサが知るように、戦略セレクトタは必要なロケーションでモデルトラバーサを構成する。このようにして、モデルトラバーサはいかなる特定のサーチアルゴリズムからも分離されることができると同時に、依然としてモデルサーチアルゴリズムを用いて構成ポリシーに基づいて装置モデルを横断できる。記憶部トラバーサ及び選択ポリシーのロードにも同様のロード手順が用いられる。

【0039】

モデルサーチアルゴリズムは、1つのデータメンバー（例えばM o M）を、別のデータメンバー（例えば構成ポリシー）に基づいて横断するように記述されるのが好ましい。同様に、記憶部サーチアルゴリズムは、1つのデータメンバー（例えば記憶されている有効予定表リスト）を、別のデータメンバー（例えば選択ポリシー及びジョブ明細）に基づいて横断するように記述されるのが好ましい。このようにして、戦略セレクトタは、トラバーサがいかなる特定のアルゴリズムへの依存関係からも分離されるように、トラバーサ、関連アルゴリズム、並びに第1及び第2データ要素を構成する。トラバーサの最良の期待性能を達成するために、トラバーサは、ライブラリ内の置換可能なアルゴリズムから最適なアルゴリズムを選択することによって、M o Mに適合される。

【0040】

代替のバージョンでは、アルゴリズムは、アルゴリズムシェル及び複数の置換可能なアルゴリズムフラグメント（例えば、置換可能サブルーチン）を含んでもよい。戦略セレクトタは、戦略設計パターン（本明細書で論じる）に基づいて、アルゴリズムシェル及び1組のアルゴリズムフラグメントを、トラバーサ用アルゴリズムとして選択する。

【0041】

予定表は、有効又は無効であり得る。例えば、マーカモジュールに、11インチ×17インチの用紙を片面モードだけでマーキングする機能がある場合は、両面モードでマーキングされる11インチ×17インチの用紙をマーカモジュールを通して送ることを指定する予定表は、無効な予定表となる。

【0042】

モデルトラバーサは全ての可能な予定表をサーチし、有効な全予定表を後の使用のために記憶する。

【0043】

プリントジョブが実行される時（即ち選択時）、プリント装置の装置コントローラは、ジョブ明細をプリント予定表と照合する。1枚の用紙のためのプリント予定表は、用紙の処理に必要な、必要モジュール及び各モジュールの必要機能のスケジュールである。例えば、特定の予定表は、用紙が給紙モジュールのトレイ1から選択され、マーカモジュールを介して片面モードで処理され、綴じられずに、スタッカモジュールのトレイに送出されることを指定し得る。別の予定表は、用紙が給紙モジュールのトレイ2から選択され、マーカモジュールを介して両面モードで処理され、プリントジョブの最後に左上の角をホチキス止めされて、スタッカモジュールのトレイに送出されることを指定し得る。ジョブ明細と予定表との照合タスクは、記憶部トラバーサが行う。

【0044】

記憶部トラバーサは、ジョブ明細（又はアセンブリツリー）を解析し、記憶されている有効予定表のリストをサーチして、プリントジョブ明細を処理できる予定表を選択する。

【0045】

例えば、プリント装置は、A4又はA5サイズ of 用紙を給紙できる給紙モジュールと、A4又はA5 of 用紙に片面プリントできるマーカモジュールと、A4又はA5 of 用紙をスタックできるスタッカモジュールとを含む（即ち、Feed（給紙）A4及びFeedA5機能を有する給紙モジュールと、Simplex（片面）A4及びSimplexA5機能を有するマーカモジュールと、Stack（スタック）A4及びStackA5機能を有するスタッカモジュール）。モデルトラバーサはパワーアップ時に、1. FeedA4-SimplexA4-StackA4、及び2. FeedA5-SimplexA5-StackA5の2つの予定表を見出すことになる。 10

【0046】

モデルトラバーサは、FeedA4機能はA4サイズ of 用紙を出力するが、SimplexA5機能はA5用紙だけを受け入れ可能なことが（MOM内の機能自体の記述によって）わかるので、FeedA4-SimplexA5-StackA4のような他の予定表を除外することになる。

【0047】

ジョブ明細は、装置非依存型の、プリントされることを所望される用紙の指定である。装置コントローラ20の上流のシステムモジュール（マークファシリティインタフェース又はMFIと呼ばれる）は、ジョブ明細のユーザが指定したジョブの属性（例えば、横置きモード、2コピー、ブルーの表紙、等）を、アセンブリツリーとして再定式化する。アセンブリツリーは、MFIから装置コントローラ20に入力される装置非依存型の記述である。ジョブ明細及びアセンブリツリーは、本明細書に参照として援用する特許文献3及び特許文献4に述べられている。ジョブ明細のパラメータは、その最もシンプルな形態で、MFIを介して装置コントローラへと送られる。例えば、1つのジョブは、A4用紙が片面モードでプリントされ、スタック及びホチキス止めされることを指定し得る。 20

【0048】

プリントジョブを実行するには、まず、そのジョブ明細を処理できる予定表を見つける必要がある。ジョブが装置コントローラに到達し、まず、用紙が例えばA4用紙であることを要求すると、装置コントローラは記憶部トラバーサを用いて記憶されている各予定表を巡回し、各サイクルで、記憶部トラバーサは予定表バリデータを用いて、現在選択されている予定表を選択ポリシー及びジョブ明細に照らしてチェックし、それが所望のA4用紙を生成できるか否かを確認する。 30

【0049】

予定表バリデータは、各ポリシーにわたって、現在選択されている予定表をポリシーに照らしてチェックしながら反復するモジュールである。予定表バリデータはこのように1つ以上のポリシーによってサポートされる。予定表バリデータは、モデルトラバーサによって、記憶部トラバーサによって、又はそれらの両方によって用いられ得る。例えば、予定表バリデータは、FeedA5-SimplexA5-StackA5という予定表を最初に試みてもよい。バリデータが有し得る1つのポリシー（ポリシー1と呼ぶ）は、要求されている用紙特性が、予定表によって生成できる用紙の用紙特性とマッチする（即ち"適合する"）か否かをチェックして確認する。ポリシー1は、FeedA5-SimplexA5-StackA5予定表の出力の記述をチェックし、それを要求されている用紙の用紙記述と照合する。ポリシー1でのチェックの結果は、特性の幾つかはマッチしない（即ち、予定表の出力用紙サイズはA5であるが、要求されている用紙特性はA4である）というものである。従って、この予定表はポリシー1をパスしないので、選択されない。 40

【0050】

次に、記憶部トラバーサは前進して、予定表バリデータを用いて次の予定表である予定表2（FeedA4-SimplexA4-StackA4）をチェックし、予定表2はポリシー1をパスする。予定表バリデータが検討しなければならない、予定表の機能が全て"準備完了"状態でなければならないことを指定し得る別のポリシー（ポリシー2）があるかもしれない。ポリシー 50

2 もパスした場合には、予定表 2 が、要求されている用紙を生成する予定表として選択される。

【0051】

選択ポリシーは、選択される予定表がジョブ明細を処理するために満足しなければならない基準である。ポリシーは、一般的に、if-then-elseステートメントの形態である。ポリシーは、そのif-then-elseステートメントを用いて、装置（又は装置モデル若しくは処理すべきジョブ）に関する決定を行う。例えば、ポリシーを用いて、装置が所望される働きを実行するのに必要な機能及び制約を有するか否かを決定してもよい。ポリシーを用いて、装置モジュールの機能が準備完了状態にあるか否かを解釈してもよい。或いは、ポリシーを用いて、装置モジュールの機能が存在するか（例えば、給紙モジュールが、用紙が入った8.5インチ×11インチトレイを含むか）否かを解釈してもよい。ポリシーは、プログラムがサブルーチンに分解されるのと同じ方法で分解される下位ポリシーを含んでもよい。

10

【0052】

これらのポリシーは、プリント装置ごとに、及び同じプリント装置内でもその時々で、異なり得る。各プリント装置は、選択されるための可能なポリシーの"ライブラリ"を有して出荷される。パワーアップ時に、戦略セレクトは、この特定の装置構成に対してどのポリシーが妥当であるかを、装置モデルに照らして判別する。例えば、戦略セレクトは、本明細書に更に述べるように、装置モジュールの数が所定の数（例えば7）より大きいか否かに応じて異なるポリシーを選択し得る。戦略セレクトは、選択されたポリシーだけにわたって反復するために、予定表バリデータを構成する装置コントローラの一部である。

20

【0053】

ポリシーを装置構成に基いて構成できるだけでなく、別個のモニタモジュールが装置の1つ以上の操作をモニタでき、戦略セレクトに、装置状態の検出、モニタされた性能、及び動作環境に基いて、選択された構成へ及び選択された構成から、ポリシーをダイナミックにスワップすることを要求できる。モニタモジュールは、装置のダイナミックモデルの一部である情報を用いる。装置状態が変わると、モニタモジュールは戦略セレクトに、装置の新たなダイナミックモデルに基いてポリシーを再構成するよう要求する。このようにして、1つのジョブの処理中に予定表バリデータが"予定表はスタッカAで終了しなければならない"というポリシーAを有するが、次のジョブを処理する際、戦略セレクトによってポリシーAが外されて、その代わりに"予定表はスタッカBで終了しなければならない"というポリシーBが加えられるように、処理中にポリシーをダイナミックに迅速にスワップできる。モニタ自体が、戦略セレクトによって選択される、サポートするモニタポリシーを用いてもよい。

30

【0054】

例えば、戦略セレクトによって診断モニタが選択され、装置の性能をモニタしており、その日はマーカ2の片面機能の画質が悪かった場合には、診断モニタは戦略セレクトに、マーカ2を無視せよというポリシー32を加えるように伝える。別の例では、ジョブモニタは、自分のプリントのために割増し料金を払った顧客に関する情報を有するモジュールである。

40

【0055】

図3のS02では、ジョブモニタは、ジョブが送られてきたときに、これは"優先される顧客"のジョブであるか否かを確認する。肯定される場合は、ステップS03で、ジョブモニタは戦略セレクトに、優先される顧客のジョブが他のマーカよりも良好な画質を有するマーカ4でプリントされることを保証するポリシー33を加えるよう求める。普通の顧客のジョブが送られてきたときには、スケジューラが低コストのマーカによるプリントを再開するように、ステップS04でジョブモニタは戦略セレクトに、この"割増し料金"ポリシーを外すように求める。

【0056】

一般的に、モニタを用いることで、スケジューラが特定の動作環境を学習して適応し、

50

それに従って処理を調整することが可能になる。モニタモジュールは、モニタした状況（例えば、装置状態、装置性能、及び/又は装置環境）及び処理構成（例えば、その時々を選択される構成及び選択ポリシー）を記録して覚える。

【0057】

例えば、モニタは、ある用紙のプリントに要する時間を測定及び記録する。時間を超過すると、選択ポリシーは、記憶部トラバーサに、同じ用紙タイプを処理するために異なる予定表を選択させる（例えば、ポリシーの採択によって、準備完了しているプリンタモジュール機能を選択するため）。モニタは、異なる予定表を用いて同じ用紙をプリントするのに要する時間を測定及び記録する。一定時間経過後、予定表と、その予定表を用いた用紙のプリントに要する時間との間の、統計的な関係が明らかになる。この統計的な関係は、より速い予定表を優先する選択ポリシーとして記録される。新たなポリシーを記録する際、モニタモジュールは、戦略セレクトに、このポリシーをアクティブな選択ポリシーに追加するよう要求する。

【0058】

代替例では、ハードウェアモニタが、ある特定のジョブだけに対して（例えば両面ジョブ）、マーカモジュール（例えばマーカ4）の悪い画質を見つける。モニタモジュールは、この観察を、両面ジョブの処理にマーカ4を選択することを嫌う選択ポリシーとして記録する。このようにして、スケジューラは、過去の性能がどうであったかを認識して覚えることができ、性能目標をより良く達成するように適応する。

【0059】

より詳細には、予定表バリデータは、装置コントローラ20を制御するプログラムモジュールのうちの1つのモジュールであり、このモジュールは、選択された各ポリシーにわたって、提案された予定表（即ち候補予定表）を1つ又は複数のポリシーに照らしてチェックしながら反復するためのものである。この選択されたポリシーは、戦略セレクトによって、本明細書で更に述べる可能なポリシーのライブラリから選択されたポリシーである。図4では、予定表バリデータS1はステップS11からステップS17を含む。ステップS11では、バリデータは、選択されたポリシーとの一致をチェックされるべき提案された予定表を入力する。ステップS12A及びステップS12Bの間では、バリデータはステップS13及びステップS14の周りでループを形成し、選択された各ポリシーを巡回する。ステップS13では、バリデータは現在のポリシーのif-then-elseステートメントを用いて、提案された予定表が装置内で処理され得るか否かを、装置モデル(MoM)に含まれる記述に基いて評価する。ステップS14で、バリデータは分岐し、予定表がMoMに対して有効であり現在のポリシーと一致すると評価が決定した場合には、次のポリシーの評価を続ける。しかし、予定表がMoMに対して無効及び/又は現在のポリシーと一致しないと評価が決定した場合には、バリデータはステップS16へと分岐して、ステップS17の呼出しモジュールに戻る前に、そこで無効な予定表を報告する。しかし、バリデータがステップS12A及びステップS12B間における選択された全ポリシーを通るループを完了した場合には、ステップS15で、バリデータは有効な予定表を報告し、ステップS17の呼出しモジュールに戻る。従って、予定表バリデータS1は、選択された全ポリシーに照らして、可能な予定表の有効性をチェックする手段を提供する。

【0060】

モデルトラバーサ又は記憶部トラバーサのいずれかであるトラバーサは、装置コントローラ20を制御するプログラムモジュールのうちの1つのモジュールである。モデルトラバーサは、全ての可能な予定表にわたって反復し、装置モデル及び選択された構成ポリシーの両方と一致する予定表のリストを記憶するためのものである。記憶部トラバーサは、記憶されている全ての有効な予定表にわたって反復し、選択ポリシー及びジョブ明細の両方と一致する有効な予定表を見つけるためのものである。

【0061】

図5では、モデルトラバーサS2はステップS21からステップS26を含む。ステップS21では、モデルトラバーサは、装置モデルを受け取るか、入力するか、又は見る。

ステップ S 2 2 A 及びステップ S 2 2 B の間では、モデルトラバーサはステップ S 2 3、ステップ S 2 4、及びステップ S 2 5 の周りでループを形成し、可能な各予定表を巡回する。

【0062】

モデルトラバーサ用には、1組の可能な予定表を生成するための多くの可能なアルゴリズムがある。本明細書では、一例としてバックトラック（後戻り）アルゴリズムを説明する。ステップ S 2 3 では、モデルトラバーサは、予定表バリデータ S 1 を用いて、提案された各予定表が装置内で処理され得るか否かを、装置モデル（M o M）に含まれる記述に基いて評価する。予定表バリデータ S 1 が、予定表が M o M に対して無効及び/又は選択された全ポリシーと一致しないことを決定した場合には、ステップ S 2 4 でモデルトラバーサはステップ S 2 2 B へと分岐して、次の可能な予定表の評価を続ける。しかし、予定表バリデータ S 1 が、予定表が M o M に対して有効であり選択された全ポリシーと一致すると決定した場合には、モデルトラバーサはステップ S 2 5 へと分岐し、ステップ S 2 2 B で、及びそこからステップ S 2 2 A へと、次の可能な予定表の評価を続ける前に、そこで、新たに評価された予定表を有効予定表リストに追加する。全ての可能な予定表が評価されたら、トラバーサはステップ S 2 6 の呼出しモジュールに戻る。このように、モデルトラバーサ S 2 は、装置モデル及び選択されたポリシーに基いて、有効な全予定表のリストを作成する手段を提供する。

10

【0063】

図 6 では、記憶部トラバーサ S 2' はステップ S 2 1' からステップ S 2 4' 及びステップ S 2 6' を含む。ステップ S 2 1' では、記憶部トラバーサは、有効予定表リストを受け取るか、入力するか、又は見る。ステップ S 2 2 A' 及びステップ S 2 2 B' の間では、記憶部トラバーサはステップ S 2 3' 及びステップ S 2 4' の周りでループを形成し、記憶されている各予定表を巡回する。

20

【0064】

記憶部トラバーサの動作は、有効予定表リストの予定表がサーチされる順序によって異なり得る。ステップ S 2 3' では、記憶部トラバーサは、予定表バリデータ S 1 を用いて、記憶されている各予定表がジョブ明細を処理し得るか否かを評価する。予定表バリデータ S 1 が、予定表がジョブ明細を処理できないこと及び/又は選択された全ポリシーと一致しないことを決定した場合には、ステップ S 2 4' でトラバーサはステップ S 2 2 B' へと分岐して、次の可能な予定表の評価を続ける。しかし、予定表バリデータ S 1 が、予定表がジョブ明細を処理できるとともに選択された全ポリシーと一致すると決定した場合には、記憶部トラバーサは予定表が見つかったことを宣言し、ステップ S 2 2 A' 及びステップ S 2 2 B' にセットアップされたループを終え、ステップ S 2 2 B' へと分岐し、そこで、ステップ S 2 6' の呼出しモジュールに戻る。このように、記憶部トラバーサ S 2' は、選択ポリシーに基いて、ジョブ明細を処理できる有効な予定表を見つける手段を提供する。

30

【0065】

代替の記憶部トラバーサを図 1 4 に関して説明すると、記憶部トラバーサは、戦略設計パターンに従って構成される選択可能アルゴリズムフラグメントを有する。

40

【0066】

戦略セレクトは、装置コントローラ 20 を制御するプログラムモジュールのうちの 1 つのモジュールであり、このモジュールは、モデルトラバーサと、記憶部トラバーサと、任意の関連アルゴリズム及び設計パターンテーブル（本明細書で述べる）と、目下構成されている装置に適用可能な又は装置の動作環境に従って、ポリシーライブラリからのポリシーとを選択するためのものである。

【0067】

図 7 では、戦略セレクト S 3 はステップ S 3 1 からステップ S 3 6 を含む。ステップ S 3 1 では、戦略セレクトは、装置モデル及びポリシーライブラリ内の全ポリシーを受け取るか、入力するか、又は見る。戦略セレクトは、装置に関して知るためのそこにある全て

50

を知っている。戦略セレクトは、装置がパワーアップフェーズにあるか否かを知っている。オペレータがトレイに用紙を再充填するために給紙モジュールのトレイを開けたか否かを知っている。現在処理中の用紙の、予定表によって指定されるパス内のロケーションを知っている。ステップ S 3 2 A 及びステップ S 3 2 Bの間では、戦略セレクトはステップ S 3 3、ステップ S 3 4、及びステップ S 3 5の周りでループを形成し、ポリシーライブラリ内の各ポリシー（及び、同様のライブラリに記憶されているトラバーサルアルゴリズムなどの複数のアルゴリズムの各々）を巡回する。メモリ 4 0（図 1）の一部である ROM 内にライブラリを記憶でき、戦略セレクトはポリシー及びアルゴリズムのサブセットを選択し、このサブセットをメモリ 4 0の一部である RAM に記憶する。好ましくは、RAM 内には間接アドレスベクトルだけが記憶され、選択されたポリシー又はアルゴリズムを指し示すアドレスベクトルは ROM に記憶される。ステップ S 3 3では、戦略セレクトは、ライブラリ内の各ポリシー又はアルゴリズムが装置に適用可能か否かを、装置モデル（MoM）に含まれる記述及び装置の現在の環境に基づいて評価する。

【0068】

例えば、装置がパワーアップ時の構成フェーズにあるか、又は現在のジョブの実行時の選択フェーズにあるかに応じて、異なるトラバーサルアルゴリズムが用いられることが望ましいことは、あり得るとともに、よくあることである。構成フェーズでは、モデルトラバーサの 1 つを、バックトラックアルゴリズムの 1 つのバージョン及びあるポリシー（例えばポリシー 1 及びポリシー 3）とともに用いることが望ましい。しかし、選択フェーズでは、記憶部トラバーサルアルゴリズム及び異なるポリシー（例えばポリシー 2 3 及びポリシー 4 4）を用いることが望ましいかもしれない。

【0069】

戦略セレクトは、その時点で何が進行中かに関する知識に基き、使用するアルゴリズム及びポリシーを選択するソフトウェアモジュールである。上述の例に対しては、戦略セレクトは次のように記述され得る。

```
if(configuration_phase),
    add BacktrackTraverser
    add policy1
    add policy3
```

```
else if(selection_phase),
    add StoreTraverser
    add policy23
    add policy44
```

この例示的な戦略セレクトの拡張は、装置構成の性能に関して集められた又は集められる統計的な情報を用いる。例えば、プリント装置の出荷前に、様々なサイズの装置構成、様々な機能等の選択を行うのに要する時間に関する情報を作成して ROM に記憶できる。或いは、装置の実際の使用中に情報を集めて消去可能 PROM に記憶できる。この場合は、戦略セレクトも、この計量的な知識を利用する if-then-else ステートメントを含むことになる。上述の例に対しては、戦略セレクトは次のように記述され得る。

```
if(number of machine modules in MoM > 7) and (configuration_phase),
    add Combined_Traverser
    add policy1B
```

```
else if(number of machine Module <= 7) and (configuration_phase),
    add BacktrackTraverser
    add policy1
    add policy3
```

この戦略セレクトは、装置構成が 7 個以上の装置モジュールを有する場合には、"全ての"可能な予定表をサーチして記憶するには、時間が長くかかりすぎるか、又は使用する

メモリが多すぎるということがわかっているという事実を活用する。むしろ、構成フェーズでは、"幾つか"の可能な予定表を識別して記憶する"組合わせた"トラバーサ（本明細書で述べる別のトラバーサアルゴリズム）が、選択される。そして、選択フェーズでは、記憶されている有効予定表リストのサーチには記憶部トラバーサが用いられるが、有効な予定表が見つからない場合は、トラバーサが、装置モデルに従って、可能なより有効な予定表をサーチできるように、バックトラクトラバーサアルゴリズムの1バージョンが選択される。また、ある装置構成に対して"最適化された"幾つかのポリシーもあるかもしれず、特定の種類の構成に遭遇した場合にはそれらのポリシーを用いることができる。

【0070】

ステップS34（図7）では、戦略セレクトが現在のポリシーは適用不可であると決定した場合は、戦略セレクトはステップS32Bへと分岐し、次のポリシーの評価を続ける。しかし、戦略セレクトがそのポリシーが適用可能であると決定した場合は、戦略セレクトはステップS35へと分岐し、ステップS32Bで、及びそこからステップS32Aへと、ポリシーライブラリからの次のポリシーの評価を続ける前に、そこで、新たに評価されたポリシーを選択されたポリシーのリストに追加する。ライブラリ内の全ポリシーが評価されたら、戦略セレクトはステップS36の呼出しステップに戻る。このように、戦略セレクトS3は、ポリシーライブラリからポリシーを選択するとともに現在の装置に適応可能な横断アルゴリズム又は他のアルゴリズムを選択する手段を提供する。

【0071】

装置10（図1）は、パワーアップ時にパワーアップモジュールのシーケンスを実行する。図8では、パワーアップシーケンスS4はステップS41からステップS45を含む。ステップS41では、パワーアップシーケンスはモジュール記述メモリを読んで装置モデルを構築する。ステップS42では、パワーアップシーケンスは、戦略セレクトS3を用いて、装置に適用可能なポリシー並びにトラバーサアルゴリズム及び/又は他のアルゴリズムを選択する。これは、パワーアップモデルトラバーサの選択を含むとともに、記憶部トラバーサの選択を含み得る。ステップS43では、パワーアップシーケンスは、トラバーサS2を用いて装置モデルを横断し、有効予定表のリストを作成する。そして、ステップS44では、パワーアップシーケンスは、ステップS45の呼出しモジュールに戻る前に、有効予定表を記憶する。このように、パワーアップシーケンスS4は、装置コントローラ20（図1）に、パワーアップシーケンス中に有効予定表リストを作成する手段を提供する。

【0072】

ジョブがプリントされる時には、装置10（図1）は選択時モジュールのシーケンスを実行する。図9では、選択時シーケンスS5はステップS51からステップS52を含む。ステップS51では、選択時シーケンスは、戦略セレクトS3を用いて、記憶部トラバーサ（パワーアップ時の間に選択されない場合）及び装置に適応可能なポリシーを選択する。例えば、戦略セレクトS3は、前に記憶された有効予定表リストから予定表を選択する記憶部トラバーサを選択するとともに、現在のジョブ明細に照らした予定表の実証に適したポリシーを選択する。ステップS52では、選択時シーケンスは、記憶部トラバーサを用いて、有効予定表リストに記憶されている各予定表を横断する。そして、記憶部トラバーサは予定表バリデータを用いて、現在の予定表がジョブ明細を処理できるか否かを確認する。予定表バリデータが、特定の予定表がジョブ明細を処理できることを確認したら、記憶部トラバーサは停止し、選択された予定表がタイミングのために使用可能となる。

【0073】

記憶部トラバーサは、ジョブ明細を処理できる予定表を求めて有効予定表リストをサーチする。戦略セレクトは、記憶部トラバーサ及び選択ポリシーを前に選択しており、それは予定表バリデータをサポートし、ひいては選択された記憶部トラバーサをサポートする。記憶部トラバーサは、記憶されている有効予定表を通してループして可能な予定表を選択し、予定表バリデータを用いてジョブ明細（又はアセンブリツリー）が選択された予定表によって処理され得るか否かをチェックする。否である場合は、記憶されている次の予

定表を試みる。予定表バリデータが、選択された全ポリシーに基づいて、ジョブ明細（又はアセンブリツリー）が1つの予定表によって処理され得ると決定した場合には、その予定表が選択され、更に別のタイミングされるべきシステムコンポーネントに渡される。スケジューラの目的は、プリントジョブを処理できる有効な予定表を見つけること、及び、後の装置による実行のために、その予定表をタイミングすることである。

【0074】

ジョブ明細を処理できる予定表が2つ以上あるということが生じ得る。そのような場合には、トラバーサは、最初に見つけた予定表を選択する。両方ともそのプリントジョブを処理する能力があり、一方の予定表を用いた方が他方の予定表を用いるよりも何かしらの利益があるという場合には、その利益が特徴づけられ、特徴づけられたパラメータが条件

10

【0075】

或いは、選択ポリシーはハードポリシーとソフトポリシーとに区別できる。ハードポリシーは、"満足しなければならない"ポリシーであり、ソフトポリシーは"望ましい"ポリシーである。例えば、ソフトポリシーは"入手可能であれば8.5インチ×11インチの用紙にプリントし、入手できなければ次に入手可能なより大きなサイズにプリントする"、又は"白又はパフ（淡黄色）にプリントする。但し入手可能であれば白へのプリントを優先する"というものである。このタイプの状況では、トラバーサは、単なる"有していなければならない"ハードポリシー以上のものを探している。即ち、トラバーサは"好ましい"ポリシ

20

【0076】

ハードポリシー及びソフトポリシーの使用は、機能選択をより複雑な最適化の問題にする。本明細書に参照として援用する非特許文献1は、"十分"な解決法が許容可能な場合に、問題を部分的に解決できる様々なアルゴリズムの調査を述べており、調査された技術の部分的問題解決における有効性を示す実験的な結果も提供している。本明細書に参照として援用する非特許文献2は、新たな制約が追加されたとき、又は古い制約が取り除かれたときの、制約の解決法を維持するための方法を述べており、必要且つ好ましい制約の階層表現を述べており、ユーザインタフェースアプリケーションにおけるそのような制約の階層を解決するためのアルゴリズムを提案している。本発明の柔軟性及び様々な戦略を扱う能力は、ダイナミックポリシー選択及びソフトポリシー選択によって呈示される最適化問題が、同じアーキテクチャの枠組み内で解決されることを可能にする。

30

【0077】

ソフトポリシー評価基準を用いる特定の実施の形態は、選択された各ポリシーに重みを割当てて、予定表は、予定表バリデータ内で、その予定表によっていくつのポリシーが満足されるか、それらのポリシーがどのくらい重要か（重み）のグローバルな尺度である、トータルの重みに基づいて評価される。トラバーサは、ポリシーを満足する最初の予定表を見つける代わりに、1つの予定表を見つけるとそのトータルの重み（即ちスコア）を計算し、トラバーサが閾値を超える予定表を見つけるまで、時間切れまで、又はトラバーサが最適な予定表を見つける（可能な予定表が全てサーチされたとき）まで、サーチを続ける

40

【0078】

パワーアップ戦略セレクトは、プリントジョブが要求されたときに、パワーアップ時に用いるモデルトラバーサ及び可能であれば後で用いる記憶部トラバーサを選択する。プリント装置の1つのバージョンでは、パワーアップ時に用いられるモデルトラバーサはバックトラックトラバーサである。

【0079】

本明細書に参照として援用する非特許文献3は、"制約満足問題(constraint satisfaction problems)"（以降、CSPと呼ぶ）として定式化される問題の解決に適用できる、公知のバックトラックサーチを述べている。CSPは、変数、各変数に対する1組の可能な

50

値、及び値間の制約を用いて記述される問題である。C S Pに対する解決法は、全ての制約を満足する、変数に対する値の1つ又は全ての割当てを見つけることを含む。例えば、一般的なC S Pは、隣接する州(state)が同じ色にならないように、所定数（例えば3）の色でマップを色づけることが望まれる、公知の"マップ色づけ問題"である。この問題をC S Pとして定式化すると、変数は州であり、値は各州に対する可能な色（例えば、赤、緑、及び青）であり、2つの隣接する州に同じ色が割当てられ得ることがないように制約が定式化される（例えば、各州の近傍が識別されて、異なる色になるように制約される）。解決法は、隣り合う全ての州が異なる色を有するように、各州に1つの色を割当てることである。

【0080】

バックトラックは、C S Pに対する1つ又は全ての解を見つけることができるサーチアルゴリズムである。一般的なバックトラックアルゴリズムは、前方にサーチするルーチン（図11のステップS6を参照）と後方にサーチするルーチン（図12のステップS7を参照）との2つのルーチンを含む。図10は、本明細書で例として用いる簡略化したプリンタの装置モデルM o Mを示している。本発明のスケジューラでは、バックトラックアルゴリズムの変数はM o Mの3つの装置モジュールである（即ち、給紙モジュール、マーカモジュール、及びスタッカモジュールであり、個々に1、2、及び3の番号をつける）。各変数に対して2つの可能な値が存在し、これらの可能な値は個々のモジュールの機能である（例えば、M o M内の各モジュールがA4及びA5の用紙を処理する機能を有する）。満足されなければならない制約は、1つのモジュールのどの機能が他のモジュールの機能と両立するかに関する装置の制約に基く。解は、どの制約にも違反しないように組合わされた各モジュールにおける1つの機能として定式化された予定表である。

【0081】

図11では、ステップS61A及びステップS61Bの間で、バックトラックトラバーサは、予定表が見つかるまで、全モジュールを用いてループする。まず、ステップS62で、バックトラックトラバーサは、値を割当てられる次の変数として1つのモジュールを選択し、この変数をNextVarと呼ぶ。予定表は給紙モジュールで始まるので、給紙モジュールが最初のモジュールとして選択される。ステップS63では、給紙モジュールに対して有効な候補の値が、その機能、feedA4及びfeedA5となるように定式化される。ステップS64では、バックトラックトラバーサは、NextVar（現在は給紙モジュール）に対する値が存在するか否かを尋ねる。存在する値が2つある（feedA4及びfeedA5）ので、バックトラックトラバーサは、ステップS65で値を順序付けする（例えば、feedA4の後にfeedA5）。ステップS66では、バックトラックトラバーサは、現在のNextVar（即ち給紙モジュール）に、最初の値（即ちfeedA4）を割当てることによって、ラベル付けする。次に、ステップS61A及びステップS61Bを介してループを繰り返すことにより、サーチを続ける。

【0082】

バックトラックトラバーサが選択する、ラベル付けされるべき次の変数は、マーカモジュールである。今回のNextVarはマーカモジュールである（ステップS62）。ステップS63では、給紙モジュールは選択されたfeedA4機能からA4用紙を出力するので、マーカモジュールに対する唯一の有効な候補の値はmarkA4機能であることがわかる。ステップS64は、NextVarに対する値が存在するか否かをテストし、1つの値が存在するので、ステップS65で、その値が順番に配置される。1つの値だけが存在するので、ステップS65はトリビアルである。ステップS66では、バックトラックトラバーサは、現在のNextVar（即ちマーカモジュール）に、最初の値（即ちmarkA4）を割当てることによって、ラベル付けする。

【0083】

同様に、バックトラックトラバーサは、ラベル付けすべき次の変数がスタッカモジュールであることを見出し、給紙モジュール及びマーカモジュールに対して現在選択されている値と両立するのはstackA4機能であることを見出す。バックトラックトラバーサは、ス

テップ S 6 2 でラベル付けされていない変数が残っていないことを見出し、従って、feed A4-markA4-stackA4 という予定表を見つけた。ラベル付けされていない変数がないので、ステップ S 6 3 及びステップ S 6 4 で次の変数に対して見つけるべき値もなく、従って、ステップ S 6 7 で、バックトラクトラバーサはGoBack（即ち、後方サーチ）部に進む。

【0084】

バックトラクトラバーサは、"バックトラック"によって更に多くの予定表を見つけ続ける。ステップ S 7 2 では、バックトラクトラバーサは、スタッカモジュールに対して有効な値が更に存在するか否かを決定する。この例では、stackA5は、既にラベル付けされた給紙モジュール及びマーカモジュール（即ち、マーカモジュールからの A 4 用紙出力）と両立しないので、有効な値ではない。従って、ステップ S 7 3 では、最後の変数（即ち、スタッカモジュール）のラベルが外され、Gobackは更に後戻りする。次に、ステップ S 7 2 で、マーカモジュールに対する有効な値（給紙モジュールに対して選択されたfeed A4機能と両立する値）が残っていないことがわかり、ステップ S 7 3 で、マーカモジュールのラベルが外され、Gobackは更に後戻りする。次の回のステップ S 7 2 が処理され、他の全てのラベル付けされた変数と両立する値（即ち機能）が見つけられる。実際には、マーカモジュール及びスタッカモジュールのラベルは外されているので、給紙モジュールの機能が両立しなければならない他のラベル付けされた変数はない。ステップ S 7 4 で、最初の変数（即ち、給紙モジュール）が、その機能の中から見つけられた唯一の他の値で再度ラベル付けされる。他の値とはfeedA5機能である。

【0085】

ステップ S 7 5 で、トラバーサのGoBack部は、バックトラクトラバーサの始めに進む。次に、バックトラクトラバーサは前方にサーチして迅速に検出し、マーカモジュールをmarkA5機能で、及びスタッカモジュールをstackA5機能でラベル付けし、このように、feedA5-markA5-stackA5という第 2 の最後の有効な予定表を見つめる。

【0086】

第 2 の最後の予定表が識別されたら、ステップ S 6 4 及びステップ S 6 7 で、トラバーサはトラバーサのGoBack部に進む（最初の予定表を見つけた後と同様）。次に、ステップ S 7 1 で、最後にラベル付けされた変数は、全てのモジュール及びそれらの機能の検討を尽くし、見つけるべき更なる解がないので、ステップ S 7 6 でトラバーサは終了する。

【0087】

本発明の代替の実施の形態では、パワーアップシーケンス時間の短縮又はメモリの節約が望まれる場合、パワーアップ時に予定表を記憶する必要がないように、パワーアップモデルトラバーサが完全に削除される。そして、選択時に、ジョブをプリントする際、記憶部トラバーサを用いる代わりに、戦略セレクトは装置モデルを横断する選択時モデルトラバーサを用いて、まず、推定上の予定表が全ての構成ポリシーを満足するか否かをチェックし、次に、その予定表が選択ポリシーを満足してプリントジョブを処理できるか否かをチェックして確認する。選択時モデルトラバーサは、プリントジョブを処理できる予定表を見つめるために装置モデルを"迅速に"横断するために、完全なバックトラクトラバーサを用いてもよい。この実施の形態の使用は、迅速なパワーアップが必要で、装置が異なるモジュール及びモジュール内の機能を有し得る場合に、好ましいことがある。

【0088】

本発明の更に別の実施の形態では、パワーアップトラバーサは、パワーアップ時に、装置モデルに従って、可能な幾つかの有効な予定表のリストを生成する、修正されたモデルトラバーサである。この予定表の短いリストは、任意の所望の手段によって（例えば、メモリ記憶を最小にする、最も頻繁に用いられる、等）選択され得る。しかしながら、この手段は、この装置を用いるプリントショップで最も用いられそうな予定表を選択するのが好ましい。そして、選択時には、修正された記憶部トラバーサが、この有効な予定表の短いリストを横断し、プリントジョブを処理できる予定表を求めてサーチする。ジョブを処理できる予定表が見つからない場合は、記憶部トラバーサの操作の後に、装置モデルを"迅速に"横断するための第 2 の修正されたモデルトラバーサの操作が続き、パワーアップ

シーケンス中にサーチされなかった残りの可能な予定表の中から、プリントジョブを処理できる予定表を見つける。

【0089】

この"迅速な"選択時モデルトラバーサの変形では、モデルトラバーサの選択時操作中に識別された有効な予定表が、メモリに記憶されるか、又は既存の有効予定表を記憶しているメモリに追加される。その後、プリントのために別のジョブが提出されたときに、選択時記憶部トラバーサは、まず、その時点で"迅速な"選択時モデルトラバーサの操作中に識別されて記憶された予定表を含む記憶されている予定表の中から使用可能な予定表を求めてサーチする。このようにして、記憶されている予定表は、最も頻繁に用いられる予定表を反映する。

10

【0090】

有効な予定表を記憶するためのメモリスペースが限られているアプリケーションでは、"迅速な"トラバーサは、ある基準（例えば、最も長時間使われずに記憶部に残っている予定表を削除）に基いて記憶されている予定表を削除するためのモジュールを含む。このようにして、予定表の記憶は、最も頻繁に用いられる予定表を反映する。このようにして、スケジューラは、プリントショップで最も頻繁に用いられる予定表を"学習"する。

【0091】

通常、パワーアップ時とは、1日に1回、1週間に1回、又はメンテナンス後にプリンタを再びオンにするときに、プリンタに最初に電力を加える時のことである。しかし、予定表の記憶部は不揮発性メモリ（例えば、磁気ディスク、フラッシュメモリ、又は電氣的に消去可能な不揮発性メモリ）であってもよいので、電力が新たに加えられる度ごとにパワーアップ処理を開始する必要はないかもしれない。不揮発性の予定表記憶部は、既に、電源が切られる前から記憶されている予定表を有しているであろう。しかし、プリント装置構成の修正後（即ち、修正されたM o M）は、新たなM o Mに合わせて調整するために、パワーアップ処理を開始する必要があるだろう。例えば、夜にスイッチを切って翌朝にスイッチを入れる装置に対しては、構成の修正がない限り、毎朝、全てのパワーアップ処理を実行する必要はない。そのような場合には、朝、装置のスイッチが入れられたときに、装置は装置構成の監査を行い、構成の変更を検出する。しかし、変更が検出されない場合は、装置は、不揮発性メモリ記憶部から、記憶されている有効予定表を再ロードする。新たなモジュールがインストールされるか、又は既存のモジュールが修正された場合は、技術者が"リセット"ボタンを押してリブートし、完全なパワーアップシーケンスを開始させる。

20

30

【0092】

代替の実施の形態では、モデルトラバーサは、予定表を見つけるために横断アルゴリズムを用いることだけを知っており、どのように予定表を見つけるかは知らないように構成される。このようにして、モデルトラバーサは、横断アルゴリズム自体をモデルトラバーサの他の機能から切り離す。従って、モデルトラバーサは、装置非依存という意味での実際の横断アルゴリズムの分離を利用でき、異なる横断用アルゴリズムをシステム内に容易に入れ替えできる。例えば、モデルトラバーサは、バックトラクタラバーサアルゴリズムを用いて装置モデル（装置グラフ）上を反復して、予定表を見つけることができよう。上述したように、非特許文献3は、制約満足問題として表現される問題の1つ又は全ての解を見つけるために使用可能なバックトラクタラバーサアルゴリズムを述べている。

40

【0093】

バックトラクタラバーサアルゴリズムを遺伝的トラバーサアルゴリズムと入れ替えることによって、"遺伝的"アルゴリズムを用いる別の手法を実施することもできよう。たとえ基本アルゴリズムが異なっても、いずれの場合にも、モデルトラバーサは同じ機能性を提供するであろう。これは、新しいアルゴリズム（例えば、最小矛盾アルゴリズム及びバックジャンプアルゴリズム）が開発され、経験的にテストされ、性能強化を提供することが示されるときに、モデルトラバーサを継続して改良するための、柔軟性及び機会を可能にする。値の順序付け(value-ordering)ヒューリスティックを用いて、バックトラッ

50

クアルゴリズムを、サーチ中の制約違反を最小化する解を優先するように強化できる方法を述べている、本明細書に参照として援用する非特許文献4を参照されたい。また、本明細書に参照として援用する非特許文献5は、基本的なバックトラックアルゴリズムの性能を高めるための種々のアルゴリズムを述べている。

【0094】

図13では、モデルトラバーサは、選択セクタによって、特定の横断アルゴリズム（例えば、ある特定の設計のバックトラックアルゴリズム、遺伝的アルゴリズム等）を用いるように構成されている。モデルトラバーサは横断ステップS8を実行する。ステップS8は、ステップS81で、予定表を見つけるために特定の横断アルゴリズムを呼び出し、ステップS82で、横断アルゴリズムによって見つめられた予定表を記憶する。ステップS83で、モデルトラバーサは、見つめるべき更なる予定表があるか否かをチェックして確認する。更なる予定表がない場合は、モデルトラバーサはステップS84の呼出しプログラムに戻る。更なる予定表がある場合は、ステップS81で再び横断アルゴリズムが呼び出される。これにより、モデルトラバーサ及びモデルトラバーサを用いる任意のシステムコンポーネントが、予定表の検出に用いられる特定の横断アルゴリズムに依存しないことが可能になる。このようにして、異なる状況に対して異なるアルゴリズムを開発できるとともに、戦略セクタがパワーアップ時に適切なアルゴリズムを選択できる。

【0095】

更に別の代替の実施の形態では、モデルトラバーサ内の横断アルゴリズムは、その戦略設計パターンとしての表現に因って複数の代替の実施の形態を可能とするように構成される。本明細書に参照として援用する非特許文献6は、基本的な戦略設計パターンの概念を述べている。本発明では、横断アルゴリズムのインプリメンテーションは、戦略設計パターンに従って組み合わされるアルゴリズム部分の小さな構築ブロックを用いるように設計される。これにより、横断アルゴリズムの様々な部分を柔軟にすることができる。アルゴリズムの（及びそれに対応してサーチ性能における）可変性は、アルゴリズムに追加される異なる変数及び値の順序付けヒューリスティックに起因して生じる。本明細書に参照として援用する非特許文献7は、様々な可変順序付けヒューリスティックの有効性をサーチ効率において比較している。横断アルゴリズムは、制約満足問題（CSP）のある変数（例えば、本発明の定式化におけるモジュール）が処理される順序（例えば、図11のステップS62におけるモジュール選択の順序）に応じて異なる効率性で性能を発揮し得る。リアルタイムモニタを用いて統計を集めることにより（又は単にヒューリスティックにより）、又はおそらくは"オフライン"でシミュレーションにより、又は出版された比較（例えばジェントらによる論文）を検討することにより、図11のステップS62でモジュール（一般的に、CSPにおける変数）を検討する好ましい順序に関する戦略（戦略設計パターンアーキテクチャと呼ぶ）を開発可能である。

【0096】

戦略設計パターンアーキテクチャは、他のファクター（例えば、小さなプリントショップ対大きなプリントショップ）に応じて異なり得る。横断アルゴリズムの戦略設計パターンアーキテクチャは、アルゴリズムが様々な変数及び値の順序付けヒューリスティックで構成されるのを可能にする。これらのヒューリスティック及び/又は戦略は、様々な装置構成及びジョブ要求に対して開発可能である。戦略セクタを用いて、モデルトラバーサのアルゴリズムによって用いられる所望のヒューリスティックを選択できる。戦略セクタは、現在の装置構成の複雑さを評価し、そのような構成において効果的又は効率的であることが示されているいずれかの戦略を選択する。

【0097】

バックトラックアルゴリズムの要素は、評価が終了されるときに終了条件を定義する(EndCondition)ステップS61A及びステップS61B（図11）と、評価される変数の順序を定義する(VariableOrderer)ステップS62（図11）と、制約満足問題の値が評価される順序を定義する(ValueOrderer)ステップS63（図11）と、を含む。戦略設計パターンとして定式化される際、バックトラックアルゴリズムは、どのようにステップが実

行されるかを正確に知らずに、単に、EndConditionステップ、VariableOrdererステップ、及びValueOrdererステップが実行されることを求める。例えば、横断アルゴリズムはVariableOrdererを呼出して、検討されるべき次の変数を与えるように要求する。ValueOrderer自体は、多くの異なるインプリメンテーションのうちの任意のインプリメンテーションであり得る（例えば、次の装置モジュールについてチェックするもの、又は最も制約の数が多い装置モジュールについてチェックするもの等）。戦略設計パターンは、アルゴリズムの基礎をなすステップが異なるインプリメンテーションと互換可能であっても、バックトラックアルゴリズムの骨組み（例えば図11）は同じであり続けることを可能にする。

【0098】

10

別の実施の形態では、記憶部トラバーサは、最初に検討する予定表を決定するために記憶部トラバーサが用いる構成可能な検索基準又は選択ポリシーに従って、記憶されている有効予定表リストを通して反復する。例えば、記憶部トラバーサは、"準備完了した機能を有する最初の全予定表を検討する"ことを指定する検索基準1を用いてもよく、次に、記憶部トラバーサは、"内部で保留中の機能を有する次の全予定表を検討する"ことを指定する検索基準2に移る、等。検索基準は、バックトラックアルゴリズムの部分又はポリシーと同様に、戦略セクタ又はモジュールによって記憶部トラバーサの構成から追加及び削除される。

【0099】

20

図14では、記憶部トラバーサは、処理ステップS9に二重ループ構成を含む。内側ループはステップS93A及びステップS93Bによって制御される。内側ループでは、記憶されている全予定表が基準（例えば、"準備完了"である予定表）に照らしてチェックされ、基準に見合う有効な予定表が検出される。外側ループは、ステップS91A及びステップS91Bによって制御される。外側ループでは、基準リストに載っている全ての基準が選択されて、"予定表が見つかりませんでした"という条件が宣言されるまで、基準に見合う有効な予定表を見つけるために内側ループによって処理される。複数の異なる基準が記憶されてもよい（例えば、"準備完了"であるモジュール機能だけを用いる予定表を選択、大容量給紙モジュールを用いる基準を選択、高画質マーカモジュールを用いる基準を選択、等）。

【0100】

30

記憶されている基準は選択ポリシーであり、戦略セクタは1つ以上の基準を選択して、戦略セクタが用いる基準のリストを作成する。ステップS92では、次の基準が処理のために選択される。ステップS94では、チェックのために次の予定表が選択され、ステップS95では、選択された予定表が現在選択されている基準に照らしてチェックされる。ステップS96では、予定表が現在の基準の要件に見合う場合には、その予定表が有効であると宣言される。予定表が有効であると宣言された場合には、ステップ97で、予定表が見つかったと言われ、記憶されている全予定表が全基準に照らしてチェックされた後に予定表が有効であると宣言されない場合は、ステップS98で、予定表が見つからなかったと言われる。

【0101】

40

予定表のリストが長い場合もあり、記憶部トラバーサの効率性は予定表リストがサーチされる順序に依存し得ることが認識されよう。また、基準のリストが存在することもあり、記憶部トラバーサの効率性は基準リストがサーチされる順序に依存し得ることも認識されよう。モデル横断アルゴリズムの場合のように、ステップS92、ステップS94、及びステップS95が単にアルゴリズムのフラグメントである他のルーチンを呼び出すように、図14に示されている記憶部横断アルゴリズムが分離されてもよい。これらのアルゴリズムフラグメントの代替バージョンを開発してもよく、モデル横断アルゴリズムの場合のように、戦略セクタは所望の戦略設計パターンに基き特定のアルゴリズムフラグメントで記憶部トラバーサを構成する。

【0102】

50

開示された本発明の特定の実施の形態に、添付の特許請求の範囲によって定義される本発明の範囲及び精神の範囲内の変更がなされ得ることを理解されたい。

【0103】

以上のように本発明を説明したが、特許請求され、特許証による保護が望まれるものを、添付の特許請求の範囲に記載する。

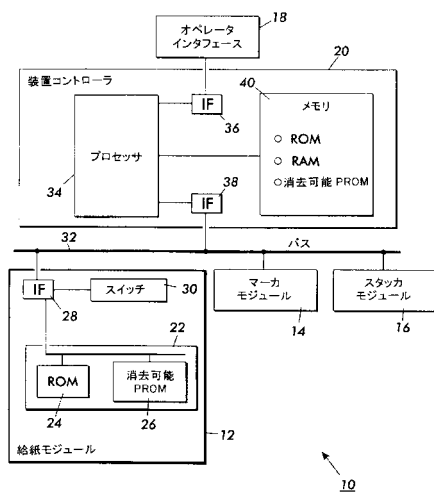
【符号の説明】

【0104】

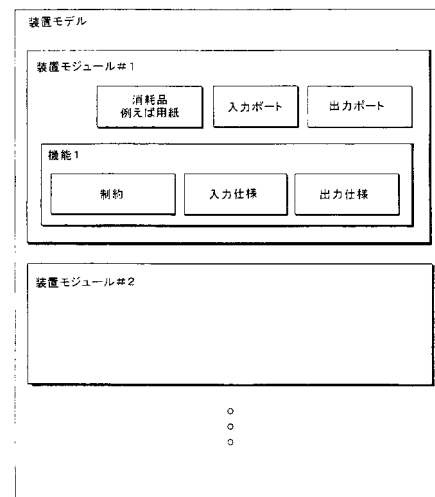
- 10 プリント装置
- 12 給紙モジュール
- 14 マーカモジュール
- 16 スタッカモジュール
- 20 装置コントローラ
- 34 プロセッサ

10

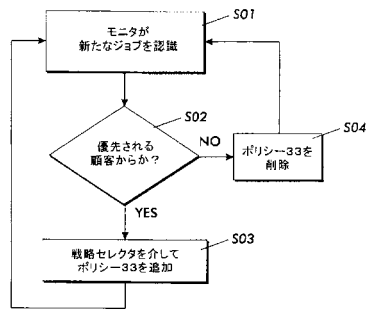
【図1】



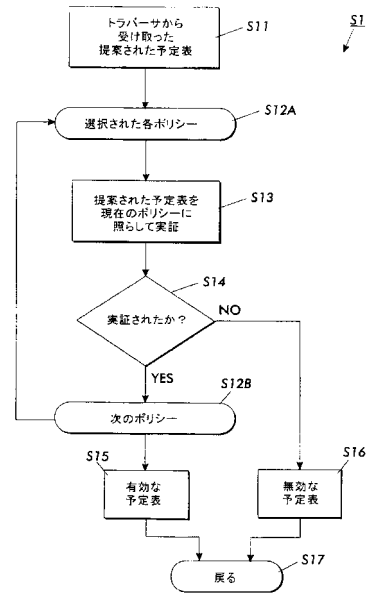
【図2】



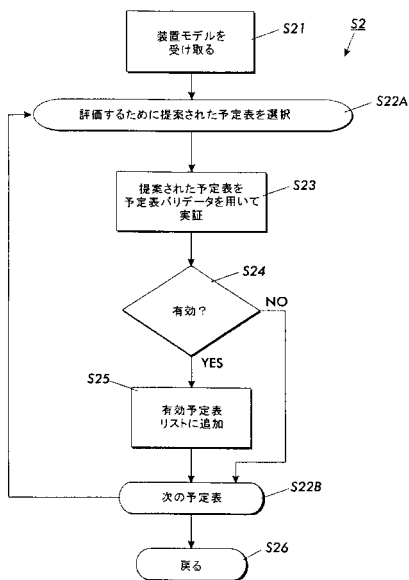
【図 3】



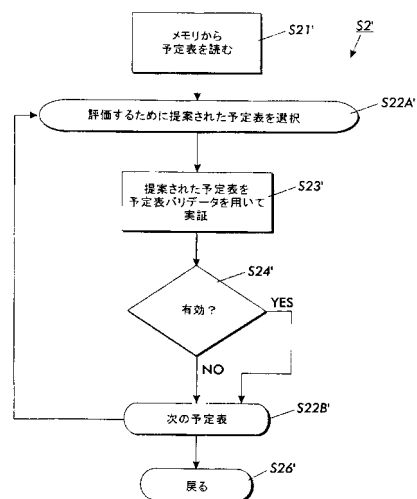
【図 4】



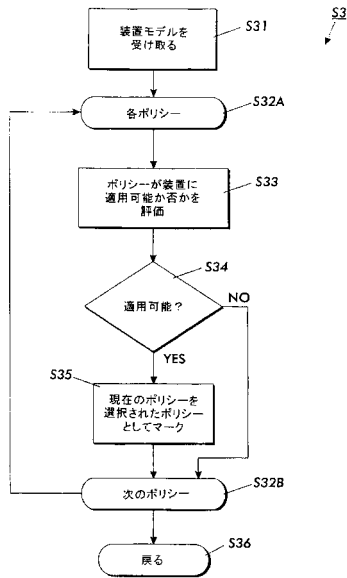
【図 5】



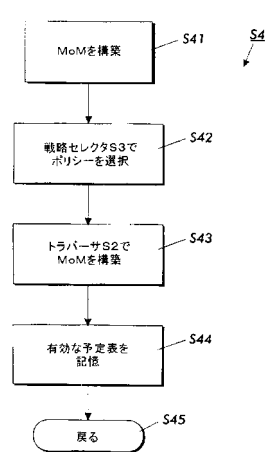
【図 6】



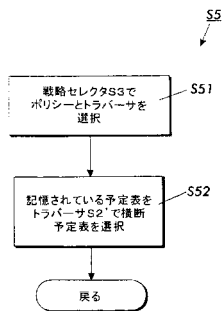
【図 7】



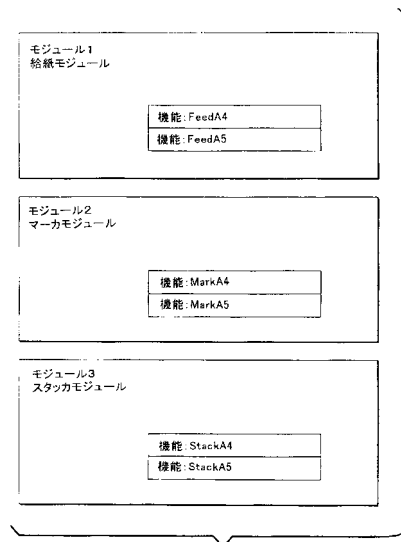
【図 8】



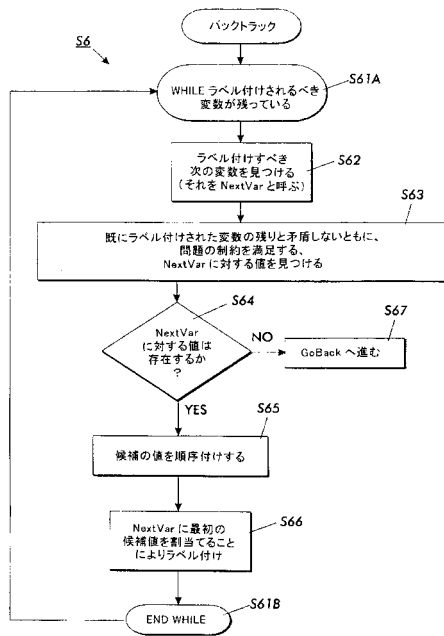
【図 9】



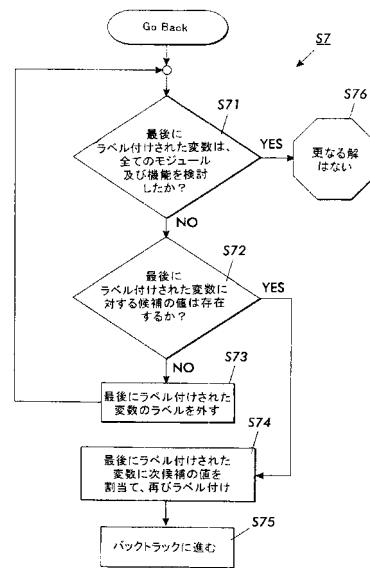
【図 10】



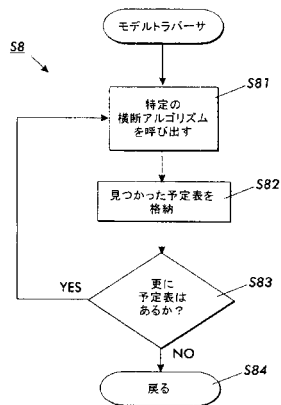
【図 1 1】



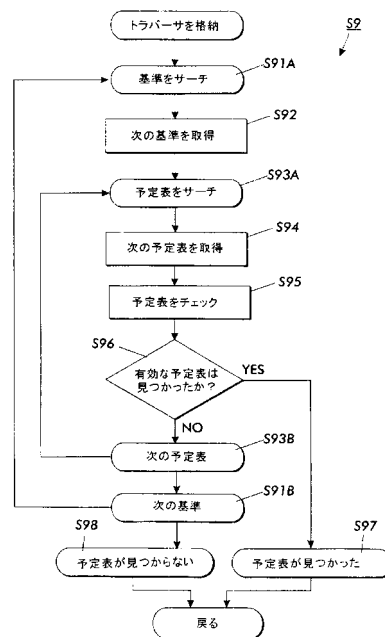
【図 1 2】



【図 1 3】



【図 1 4】



フロントページの続き

(72)発明者 ジョン エイチ. コンリー
アメリカ合衆国 1 4 6 1 9 ニューヨーク州 ロチェスター ジェネシー パーク ブルバード
2 7 4

(72)発明者 カーチス アール. マケルヴィ
アメリカ合衆国 1 4 6 0 9 ニューヨーク州 ロチェスター マーチャンツ ロード 8 3 2

(72)発明者 クリストファ ラ ローザ
アメリカ合衆国 1 4 6 2 6 ニューヨーク州 ロチェスター レッチワース アベニュー 1 7
8

Fターム(参考) 2C061 AR03 AS02 HH03 HJ03 HN15 HN17