

# オフィス環境におけるネットワークシステムの 省電力化のためのプランニング手法

栗原 孝太<sup>1,a)</sup> 和泉 諭<sup>2,b)</sup> 阿部 亨<sup>3,1,c)</sup> 菅沼 拓夫<sup>3,1,d)</sup>

**概要:** オフィスビル等で消費される電力のうち、ネットワークシステムを構成する ICT 機器の消費電力が占める割合は益々増大してきており、ICT 機器やネットワークシステムの省電力化が喫緊の課題となっている。本研究では、オフィス環境におけるネットワークシステム全体の効果的な運用による電力の有効利用を実現するプランニング手法を提案する。本手法はフリーアドレスオフィス環境におけるネットワークシステムに焦点を当て、人間の作業効率やシステムの使用状況に基づいて人間の作業場所の配置や機器を自律的に制御する。本稿では、人間の作業効率やシステムの使用状況に基づいてシステム状態の最適化を行う目的関数と、最適状態への移行手順（プラン）を自動生成するためのプランニングモデルを設計する。また、プランナを用いて提案手法の実装を行い、プランニングの実用性の確認を行う。

## 1. はじめに

オフィスビル等で消費される電力のうち、ネットワークシステムによる消費電力が占める割合は益々増大してきており、オフィスや家庭における消費電力は 1990 年以降増加傾向にある [1]。ユビキタスコンピューティングやクラウドコンピューティングの発展により、ネットワークシステムの拠点集約化が進み、コスト削減や消費電力低減に一定の効果が得られているが、オフィスにおける消費電力の約 20% が ICT 機器による電力消費であり [2]、ICT 機器の省電力化が喫緊の課題となっている。

従来のネットワークシステムの省電力化では、個々の ICT 機器の省電力化の取り組みは積極的に行われてきた [5–12]。しかし、大規模なネットワークシステムであるデータセンタにおいて、センタ内のサーバの平均使用率は 20 % 程度であり [3]、サーバを十分に有効利用出来ていない。このことから、システムを構成する各機器を適切に運用し、有効利用することが省電力化・節電を実現するための重要な課題であると言える。

そこで、PC、ネットワーク機器、プリンタなどを含む系全体の挙動に配慮して運用を制御することで、より高い消費電力低減効果が得られると考えられるが、特にオフィス環境では機器の挙動のみを配慮して制御を行うと、人間の作業効率の低下を招く可能性がある。そのため、人間の作業効率やシステム全体の挙動を考慮して、最適なシステム構成の状態を導くことが必要となる。また、システム全体の制御を行う際、システムが複雑になると、システムの再構成の手順が複雑になり、適切な制御が困難になる。そのため、システム再構成の適切な手順（プラン）を導出することが必要である。

我々は、ネットワークシステムを構成する各機器を統合的かつ自律的に制御し、システム全体の最適化を行うことで電力の有効利用を実現する手法を提案している [4]。本手法では、目的関数に基づきシステム全体の消費電力の最適化を行い、システムの状態を導出する。そして、AI 技術の一つである「プランニング」を用いて、最適状態への移行プランを自動生成する。プランニングを用いることで、複雑なシステムの再構成に対しても適切な移行プランの導出が可能となる。

本稿では、上記の手法をオフィス環境に適用して、オフィス環境におけるネットワークシステム全体の省電力化の実現を目指す。具体的には、フリーアドレスオフィス環境を対象として、人間の作業効率やオフィス全体の消費電力を考慮して、最適化を行う目的関数を設計する。また、最適状態へ移行するために人間の配置や機器の制御を行うためのプランニングモデルを設計する。

<sup>1</sup> 東北大学大学院 情報科学研究科  
Graduate School of Information Sciences, Tohoku University  
<sup>2</sup> 東北大学電気通信研究所  
Research Institute of Electrical Communication, Tohoku University  
<sup>3</sup> 東北大学 サイバーサイエンスセンター  
Cyberscience Center, Tohoku University  
a) awahara@ci.isc.tohoku.ac.jp  
b) izumi@shiratori.riec.tohoku.ac.jp  
c) beto@isc.tohoku.ac.jp  
d) suganuma@isc.tohoku.ac.jp

さらに、オープンソースのプランナを用いて提案手法を実装し、プランニングの有効性を確認するための予備実験を行う。実験結果から、プランニングの実用性について議論する。

## 2. 関連研究

### 2.1 オフィス環境の省電力化に関する関連研究

オフィス環境において、デスクトップ PC やモニタ、プリンタなど個々の ICT 機器の省電力化に関して様々な研究が存在する。Berl ら [5] は「仮想化オフィス環境」を提唱し、デスクトップ PC の環境を仮想化し、オフィスにおいて起動し続けているデスクトップ PC の環境を他の PC にマイグレーション・集約することで、オフィス全体の PC の消費電力を抑える手法を提案している。さらに、デスクトップ PC の電力モデルを定義し、解析的に省電力化効率の高さを示すとともに、シミュレーション実験を通して提案手法の有効性を評価している [6]。

Jaramillo ら [7] は PC のモニタに焦点を当てた省電力化を提案している。PC モニタの上部に超音波センサを設置することで利用者の在席状況を推定し、利用者が離席している、あるいはモニタを使わない作業をしていると推定された場合、モニタの電源を自動でオフにすることで消費電力の低減を実現している。

また、プリンタの省電力化のため、Managed Print Service(MPS) という概念が提唱されている。これはオフィスにおけるプリンタの使用状況を分析し、プリンタの台数を最適化して必要最小限にすることで省エネを目指すものであり、複合機メーカー各社から実用化され商品として提供されている [8] [9]。しかし、この概念は新たにプリンタを追加する場合を対象としているため、既設のプリンタの効率的な運用については考えていない。

一方で、近年一般的になりつつあるフリーアドレスオフィスを対象とし、作業員への座席の割り当てを行う製品が登場している [10]。フリーアドレスオフィスは作業員の座席を固定せず、共有の座席を自由に使用し、作業するオフィスである。この手法では夜間など作業員が少なくなった場合、作業員を一箇所にまとめて不要な箇所の証明を消灯することで、省電力化効果を促すことが可能である。しかし、夜間以外の通常時では座席の選択はランダムで行われ、また照明以外の要素については考慮していない。

これらに加え、オフィスの空調 [11] や照明 [12] の省電力化に関する研究も行われているが、いずれも各機器・設備個別の省電力化を焦点としており、複数の機器からなるシステム全体の省電力化を実現する技術は未だ確立されていない。

### 2.2 本研究のアプローチ

各機器個別の省電力化ではなくシステム全体の省電力化

のため、我々はグリーン指向ネットワーク管理技術の研究開発を行っている [13]。これは、ネットワーク管理技術を基盤として、そこに省電力化技術を導入することで、スマートタップなどの特別な計測機器を必要とせずにネットワークシステムの省電力化制御を行うことが可能である点が大きな特徴である。具体的には、ネットワーク管理技術により各機器の動作状況 (CPU 負荷やネットワーク負荷、動作しているタスクなど) やネットワークへの接続状態、接続時間、さらにはシステム全体のネットワーク構成を監視し、そこから各機器の消費電力やシステムの利用傾向を類推する。そして、可視化技術により、機器の利用状態や消費電力の内訳の分析や算出を行い、可視化表示する。さらに、自律化技術により、省電力化に向けたエネルギーポリシーやシステムの利用状況に基づいて、利用されていない端末の電源の ON/OFF を自律的に制御するとともに、ネットワーク構成やサーバ上で動作しているタスクを自律的に切り替えることで無駄な消費電力を削減する。

本論文では、グリーン指向ネットワーク管理技術における自律化技術の実現へ向けて、プランニング手法をネットワークシステムへ応用して、電力を有効利用するためにネットワーク構成を自律化に制御する手法を提案する。具体的にはネットワーク管理技術により収集した各機器の動作状況やシステム全体のネットワーク構成を利用して、電力を有効利用するための最適なネットワーク構成を導出し、自律的に制御する。このプランニング手法を自律化技術に用いることで、ネットワークの構成を変更する際、その構成変更の手順を自動的に導出することができる。

## 3. 提案手法の概要

### 3.1 提案する省電力化手法

提案する省電力化手法を用いたシステムの概略を図 1 に示す。

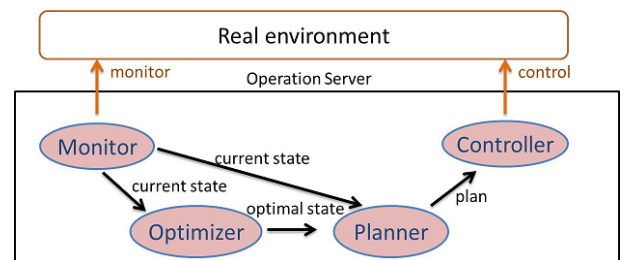


図 1 提案する省電力化手法を用いたシステム概略図

このシステムは、Monitor, Optimizer, Planner, Controller の 4 つのモジュールから構成されている。Monitor は PC の使用状況などシステムの状態を観測する。観測した値を基に Optimizer が最適化を行い、最適状態を推定する。次に、Monitor による観測値と Optimizer により得られた最適状態を基に、Planner が最適状態へ移行するた

めのアクション系列 (プラン) を作成 (プランニング) し、Controller によって実環境の状態を制御する。

### 3.2 プランニング・制御対象

PC だけでなくネットワーク機器やプリンタなども含め統合的に制御する際、系全体が複雑になり、適切な手順で機器を制御することが必要になる。そこで、本手法では AI 技術の一つである「プランニング」を用いる。プランニングとは「与えられた目標を達成するために必要な行為の系列を自動生成する」ことであり、これを用いることで、複数の機器を含むシステムの構成変更や人の移動の手順を自動生成することができる。

本稿ではオフィス環境におけるネットワークシステムを想定し、議論する。想定するネットワークシステムは、以下の機器から構成されるものとする。

- PC
- ネットワークスイッチ
- プリンタ

これらの機器に対する制御に加え、作業員に対する座席の割り当てを行う。オフィス環境における省電力化を実現するうえで、単純に消費電力の最小化のみを目標とした場合、その場における人の日常生活に支障をきたす場合がある。その為、作業員の作業効率も考慮に入れる。

### 3.3 期待される効果

先行研究 [14] では、対象として大学における研究室の居室を想定し、居室内機器の消費電力最適化に対する提案プランニング手法の有効性を議論した。具体的には消費電力を考慮する機器類として、作業用 PC、照明、空調を対象とし、研究室のプロジェクト (研究グループ) に所属するメンバー (学生) が部屋で作業用 PC を用いて作業する場面を想定した。Monitor は実環境から各機器の消費電力を観測し、その値から利用者の生活状況の推定を行う [15]。推定した生活状況を基に、Optimizer が利用者の作業効率も考慮に入れて消費電力を最適化し、最適状態を推定する。その結果、作業効率を維持したまま電力を有効利用するプランが生成可能であることを検証した。

本稿では、制御対象を作業用 PC からネットワークシステム全体に拡張し、システム全体の省電力化を行う。本提案手法による効果を以下の図 2 に示す。

2つの異なるエリアでメンバが作業している場合、1つのエリアに集約して配置することで、不必要となったネットワークスイッチやプリンタの電源を OFF にし、消費電力を低減することができる。また、これらの機器の制御や、メンバの移動などを適切な順序で行うためのプランを自動生成する。

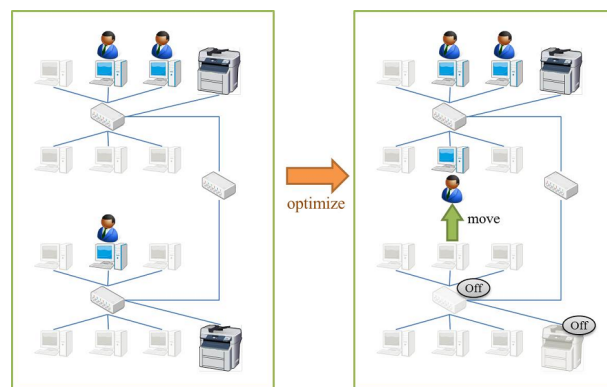


図 2 本提案手法による効果

## 4. 設計・実装

### 4.1 Optimizer

#### 4.1.1 最適化のための目的関数の設計

オフィスにおいて、新たにメンバが追加されたときや既存のメンバの再配置を行う際の座席の割り当て方を決めるため、目的関数  $F(\mathbf{s})$  を以下の式 (1) のように定義する。

$$F(\mathbf{s}) = \alpha E(\mathbf{s}) - (1 - \alpha)W(\mathbf{s}) \quad (1)$$

新たにメンバが追加されたとき、この目的関数を最小化する座席への割り当てや機器の制御が行われる。メンバがある座席に割り当てた際のシステムの状態 (各 PC の動作状況やネットワークスイッチの ON/OFF など) や、各メンバの配置を状態ベクトル  $\mathbf{s}$  で表す。  $E(\mathbf{s})$  はシステム全体の消費電力、  $W(\mathbf{s})$  はメンバの作業効率を高める要素を数値化したものである。省電力化の実現において、消費電力  $E(\mathbf{s})$  が可能な限り小さく、作業効率  $W(\mathbf{s})$  が可能な限り大きくなるのが望ましい。従って、式 (1) で定義した目的関数  $F(\mathbf{s})$  が最小となる状態ベクトル  $\mathbf{s}$  を求めることで最適状態を推定する。一般に消費電力と作業効率の間にはトレードオフの関係があり、どちらを重視するかは利用者の志向やプロジェクトの状況等により異なる。その為、利用者が予め重み係数  $\alpha$  を適当に設定することで、消費電力と作業効率それぞれの重みを自由に設定できるものとした。例えば  $\alpha = 1$  の場合、作業効率は考慮せず単純にシステム全体の消費電力の最小化を行い、  $\alpha = 0$  の場合、消費電力は無視して作業効率を最大化する。

#### 4.1.2 消費電力の定式化

消費電力  $E(\mathbf{s})$  を以下の式 (2) のように定義する。

$$E(\mathbf{s}) = \sum_{pc \in PC} P(pc, \mathbf{s}) + \sum_{sw \in Switch} P(sw, \mathbf{s}) + \sum_{pr \in Printer} P(pr, \mathbf{s}) \quad (2)$$

ここで、  $PC$ 、  $Switch$ 、  $Printer$  はそれぞれ作業用デスク

トップ PC, ネットワークスイッチ, プリンタの集合である. デスクトップ PC の消費電力  $P(pc, \mathbf{s})$  は以下の式 (3) から算出される.

$$P(pc, \mathbf{s}) = S(pc, \mathbf{s}) \times P_{pc} \quad (3)$$

$S(pc, \mathbf{s})$  は PC の状態によって変化する値であり, 以下の式 (4) のように表される.

$$S(pc, \mathbf{s}) = \begin{cases} 0 & \text{if } pc \text{ is off} \\ 0.01 & \text{if } pc \text{ is in sleep mode} \\ 0.1 & \text{if } pc \text{ is idle} \\ \vdots & \\ 1 & \text{if } pc\text{'s utilization is max} \end{cases} \quad (4)$$

$P_{pc}$  は  $pc$  の最大消費電力を表す定数であり, 各 PC のスペックや個体差によってそれぞれ異なる.

ネットワークスイッチの消費電力  $P(sw, \mathbf{s})$  は, ON の時については一定値  $P_{sw}$  であり, 以下の式 (5) で表される.

$$P(sw, \mathbf{s}) = S(sw, \mathbf{s}) \times P_{sw} \quad (5)$$

$$S(sw, \mathbf{s}) = \begin{cases} 0 & \text{if } sw \text{ is off} \\ 1 & \text{if } sw \text{ is on} \end{cases} \quad (6)$$

プリンタの消費電力  $P(pr, \mathbf{s})$  も同様に, ON の時の消費電力  $P_{pr}$  を用いて以下の式 (7) のように表される.

$$P(pr, \mathbf{s}) = S(pr, \mathbf{s}) \times P_{pr} \quad (7)$$

$$S(pr, \mathbf{s}) = \begin{cases} 0 & \text{if } pr \text{ is off} \\ 1 & \text{if } pr \text{ is on} \end{cases} \quad (8)$$

#### 4.1.3 作業効率の定式化

作業効率  $W(\mathbf{s})$  を以下の式 (9) のように定義する.

$$W(\mathbf{s}) = \sum_{p \in Project} W_{\text{dist}}(p, \mathbf{s}) + \sum_{m \in Member} W_{\text{PC}}(m, \mathbf{s}) \quad (9)$$

ここで,  $Project$  はプロジェクトの集合を表す. プロジェクトとは同じ仕事をしているグループであり, メンバはいずれか必ず一つのプロジェクトに属しているものとする. 一般に, 同プロジェクトのメンバが近くで作業している時, 作業効率が上がると考えられる. そこで, オフィスを複数のエリアに区切り, 同プロジェクトのメンバが同じエリアで作業している時, 作業効率  $W(\mathbf{s})$  が上がるものとする.

同プロジェクトのメンバが同じエリアにいることにより加算されるパラメータ  $W_{\text{dist}}(p, \mathbf{s})$  を以下の式 (10) のように定義する.

$$W_{\text{dist}}(p, \mathbf{s}) = \beta \times N_{\text{pair}}(p, \mathbf{s}) \quad (10)$$

ここで,  $N_{\text{pair}}(p, \mathbf{s})$  は同一プロジェクトで同じエリアにいるペアの数を表し,  $\beta$  は比例係数である.

また,  $W_{\text{PC}}(m, \mathbf{s})$  は PC のスペック差が作業効率に与える影響を数値化したものであり,  $Member$  はメンバの集合である. PC は一般にハイスペックであれば消費電力は高いが, その利用者の作業効率は高まると考えられる. そこで, メンバ  $m$  が使用する PC を  $pc_m$  とし, その最大消費電力  $P_{pc_m}$  を用いて,  $W_{\text{PC}}(m, \mathbf{s})$  を以下の式 (11) のように定義する.

$$W_{\text{PC}}(m, \mathbf{s}) = \gamma \times P_{pc_m} \quad (11)$$

ここで,  $\gamma$  は比例係数である.

以上の式により, 電力の有効利用のための最適状態を推定する.

## 4.2 Planner

本研究では, 実環境の状態 (初期状態) から最適状態 (ゴール) に至るまでのアクション系列 (プラン) をプランニングを用いて生成する. 本稿では, Planner の設計・実装に DLPlan [16] を用いる. 以下, プランニングを行うために必要な, オブジェクト, ステート, アクションから構成されるプランニングモデルの設計について述べる.

プランニングを行う対象として, オフィスで作業するメンバやネットワークシステムを構成する各機器をオブジェクトとして定義した. 表 1 に各オブジェクトの定義を示す.

表 1 オブジェクトの定義

Member	オフィスで作業するメンバ
PC	デスクトップ PC
Switch	ネットワークスイッチ
Printer	プリンタ
Area	オフィスを区切ったエリア

次に, オブジェクトの状態 (ステート) として表 2 の通り定義する.

最後に, ステートを変化させるために行う動作であるアクションを定義する. DLPlan におけるアクションの構文規則を以下に示す.

- oper (アクション名)
- vars: (関係するオブジェクト)
- pre: (前提条件)
- del: (削除するステート)
- add: (追加するステート)

表 2 ステートの定義

inArea(?member, ?area)	メンバ?member がエリア?area にいる
using(?member, ?pc)	メンバ?member が PC?pc を使っている
notUsing(?member)	メンバ?member が PC を使っていない
switchOn(?switch)	ネットワークスイッチ?switch が ON になっている
switchOff(?switch)	ネットワークスイッチ?switch が OFF になっている
printerOn(?switch)	プリンタ?printer が ON になっている
printerOff(?switch)	プリンタ?printer が OFF になっている
hasOrder(?switch, ?pc)/ hasOrder(?switch, ?printer)	ネットワークスイッチ?switch に PC?pc/プリンタ?printer が繋がっている

この構文規則に従い、それぞれのアクションを表現すると以下ようになる。

- メンバ?member がエリア?fromarea からエリア?toarea へ移動する

```
oper moveArea
vars: ?member, ?fromarea, ?toarea
pre: dom:inArea(?member, ?fromarea),
     dom:notUsing(?member)
del: dom:inArea(?member, ?fromarea)
add: dom:inArea(?member, ?toarea)
```

- メンバ?member が PC?pc を使い始める

```
oper usePC
vars: ?member, ?pc, ?switch
pre: dom:notUsing(?member),
     dom:hasOrder(?switch, ?pc),
     dom:switchOn(?switch)
del: dom:notUsing(?member)
add: dom:using(?member, ?pc)
```

- メンバ?member が PC?pc を使うのを止める

```
oper unusePC
vars: ?member, ?pc, ?switch
pre: dom:using(?member, ?pc),
     dom:hasOrder(?switch, ?pc),
     dom:switchOn(?switch)
del: dom:using(?member, ?pc)
add: dom:notUsing(?member)
```

- プリンタ?printer を ON にする

```
oper turnPrinterOn
vars: ?printer, ?switch
pre: dom:printerOff(?printer),
     dom:hasOrder(?switch, ?printer),
     dom:switchOn(?switch)
```

```
del: dom:printerOff(?printer)
add: dom:printerOn(?printer)
```

- プリンタ?printer を OFF にする
 

```
oper turnPrinterOff
vars: ?printer, ?switch
pre: dom:printerOn(?printer),
     dom:hasOrder(?switch, ?printer),
     dom:switchOn(?switch)
del: dom:printerOn(?printer)
add: dom:printerOff(?printer)
```
- ネットワークスイッチ?switch を ON にする
 

```
oper turnSwitchOn
vars: ?switch
pre: dom:switchOff(?printer)
del: dom:switchOff(?printer)
add: dom:switchOn(?printer)
```
- ネットワークスイッチ?switch を OFF にする
 

```
oper turnSwitchOff
vars: ?switch
pre: dom:switchOn(?printer)
del: dom:switchOn(?printer)
add: dom:switchOff(?printer)
```

## 5. 実験

### 5.1 実験の概要

本章では特に Planner に焦点を当て、プランニングの有効性を確認するための予備実験を行う。具体的には、あるシナリオに基づいたオフィスにおける作業状況を設定し、初期状態と最適状態を Planner に与え最適状態へと至るプランを生成する。この実験を通し、最適状態へと至る適切なプランが生成されることを確認する。

本実験の実行環境を以下の表 3 に示す。

表 3 実行環境

PC	Dell Optiplex 980
OS	Windows 7 Professional 64bit
CPU	Intel(R) Core(TM) i7 870 @ 2.93GHz
RAM	4.00GB
Planner	DLPlan

シナリオとして、図 3 のように 2 人のメンバが 2 つのエリアからなるオフィスで作業している環境を想定する。

それぞれのエリアでは PC が 6 台ずつと、ネットワークスイッチ、プリンタがそれぞれ 1 台ずつ稼働している。初期状態として 2 人のメンバはそれぞれ別のエリアで作業しており、2 つのエリアのネットワークスイッチとプリンタがそれぞれ ON になっている状態を与える (図 3 中左図)。また、最適状態 (ゴール) として、2 人のメンバが同じエリアで作業し、無人となったエリアのネットワークスイッ

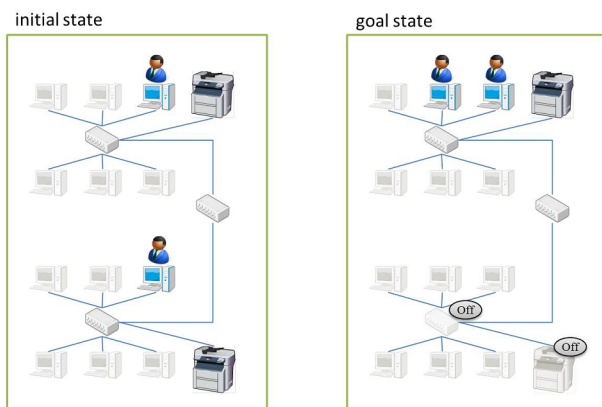


図 3 想定する環境

チとプリンタが OFF になっている状態を与える (図 3 中右図)。このシナリオを 4.2 節で設計したプランニングモデルを用いて表す。まずオブジェクトは以下の通り表せる。

```
Member(prob:m1)
Member(prob:m2)
PC(prob:pc11)
PC(prob:pc12)
PC(prob:pc13)
PC(prob:pc14)
PC(prob:pc15)
PC(prob:pc16)
PC(prob:pc21)
PC(prob:pc22)
PC(prob:pc23)
PC(prob:pc24)
PC(prob:pc25)
PC(prob:pc26)
Switch(prob:sw1)
Switch(prob:sw2)
Printer(prob:pr1)
Printer(prob:pr2)
Area(prob:a1)
Area(prob:a2)
```

次に、初期状態およびゴールは以下の通り表せる。

- initial state
  - dom:inArea(prob:m1, prob:a1),
  - dom:inArea(prob:m2, prob:a2),
  - dom:using(prob:m1, prob:pc11),
  - dom:using(prob:m2, prob:pc21),
  - dom:switchOn(prob:sw1),
  - dom:switchOn(prob:sw2),
  - dom:printerOn(prob:pr1),
  - dom:printerOn(prob:pr2),
  - dom:hasOrder(prob:sw1, prob:pc11),
  - dom:hasOrder(prob:sw1, prob:pc12),
  - dom:hasOrder(prob:sw1, prob:pc13),
  - dom:hasOrder(prob:sw1, prob:pc14),
  - dom:hasOrder(prob:sw1, prob:pc15),
  - dom:hasOrder(prob:sw1, prob:pc16),
  - dom:hasOrder(prob:sw2, prob:pc21),
  - dom:hasOrder(prob:sw2, prob:pc22),
  - dom:hasOrder(prob:sw2, prob:pc23),
  - dom:hasOrder(prob:sw2, prob:pc24),
  - dom:hasOrder(prob:sw2, prob:pc25),

- dom:hasOrder(prob:sw2, prob:pc26),
- dom:hasOrder(prob:sw1, prob:pr1),
- dom:hasOrder(prob:sw2, prob:pr2)

- goal state
  - dom:inArea(prob:m1, prob:a1),
  - dom:inArea(prob:m2, prob:a1),
  - dom:using(prob:m1, prob:pc11),
  - dom:using(prob:m2, prob:pc12),
  - dom:switchOn(prob:sw1),
  - dom:switchOff(prob:sw2),
  - dom:printerOn(prob:pr1),
  - dom:printerOff(prob:pr2)

## 5.2 実験結果および考察

初期状態とゴールを入力として、DLPlan を用いてプランを生成した。出力されたプランを図 4 に示す。

```
10 Loading initial state
11 Looking for a valid plan...
12 Time solving problem: 6225 ms
13 Total time solving problems: 6225 ms
14
15 Problem: problem1
16 Plan: [unusePC{?member=prob:m2, ?switch=prob:sw2, ?pc=prob:pc21},
17 moveArea{?member=prob:m2, ?fromarea=prob:a2, ?toarea=prob:a1},
18 usePC{?member=prob:m2, ?pc=prob:pc12, ?switch=prob:sw1},
19 turnPrinterOff{?printer=prob:pr2, ?switch=prob:sw2},
20 turnSwitchOff{?switch=prob:sw2}]
21
22
```

図 4 生成されたプラン

ここでは、メンバの移動後に、不必要となったプリンタ、その後にネットワークスイッチの順でオフにしている。このように、メンバの移動や各機器の制御が正しい順序でプランが生成されることがわかった。また、この際の計算時間は約 6 秒であり、充分実用的であるといえる。今後は現実のオフィスにより近い規模、シナリオにおける実験を進める必要がある。

## 6. おわりに

本稿では、オフィス環境におけるネットワークシステム全体の省電力化を行うため、消費電力最適化のためのプランニング手法をオフィス環境に適用し、システムを構成する機器や作業者の配置を最適化する手法を提案した。具体的には、フリーアドレスオフィスを対象として、人間の作業効率やオフィス全体の消費電力を考慮した最適化のための目的関数と、最適状態へ移行するためのプランニングモデルの設計について述べた。さらに、オープンソースのプランナを用いた予備実験を通してプランニングの実用性についての確認を行った。今後は、システム全体の実装を行い提案手法の有用性を検証し、実用化に向けた検討を行う。謝辞 本研究の一部は、総務省 SCOPE(旧 PREDICT)「情報システムの省電力化を実現する次世代ネットワーク管理技術の研究開発」の支援を受けて実施している。

参考文献

- [1] 資源エネルギー庁, <http://www.enecho.meti.go.jp/>
- [2] 一般財団法人省エネルギーセンター, [http://www.eccj.or.jp/office\\_bldg/](http://www.eccj.or.jp/office_bldg/)
- [3] Lefurgy C, Xiaorui Wang, and Ware M.: Server-Level Power Control, *Proc. Fourth International Conference on Autonomic Computing (ICAC 2007)*, pp.11–15 (2007).
- [4] 栗原孝太, 和泉諭, 阿部亨, 菅沼拓夫: ネットワークシステムにおける電力の有効利用のための自律的制御手法, 第75回情報処理学会全国大会, Vol.3, pp.89–90 (2013)
- [5] Andreas B., Nicholas R., Johnathan I., and Hermann de Meer: Network virtualization in energy-efficient office environments, *The International Journal of Computer and Telecommunications Networking*, Vol.54, No.16, pp.2856–2868 (2010).
- [6] Andreas B. and Hermann de Meer, An energy consumption model for virtualized office environments, *Future Generation Computer Systems*, Vol.27, No.8, pp.1047–1055 (2011)
- [7] Jaramillo, P. and Amft, O.: Improving energy efficiency through activity-aware control of office appliances using proximity sensing - A real-life study, *Proc. the 5th International Workshop on Smart Environments and Ambient Intelligence*, pp.664–669 (2013)
- [8] 株式会社リコー, “マネージド・プリント・サービス (MPS),” <http://www.ricoh.co.jp/outsourcing/mps/>
- [9] 富士ゼロックス株式会社, “出力環境最適化,” <http://www.fujixerox.co.jp/product/multifunction/promotion/optimize/>
- [10] コクヨファニチャー株式会社, “Office DARTS (座席アシストシステム),” <http://www.kokuyo-furniture.co.jp/solution/service/cd/cd3a.html>
- [11] 高木康夫, 米沢憲造, 村山大, 西村信孝, 花田雄一, 山崎謙一: 快適性と省エネを両立させる連携省エネ空調制御技術, 計測自動制御学会論文集 Vol.46, No.8, pp.430–438 (2010)
- [12] 小野景子, 三木光範, 米澤基: 知的照明システムのための自律分散最適化アルゴリズム, 電気学会論文誌 C, Vol.130, No.5, pp.750–757 (2010)
- [13] Inaba, T., Ogasawara, T., Kita, N., Nakamura, N., Suganuma, T., and Shiratori, N.: Green-oriented Never Die Network Management: The Concept and Design, *Proc. of the 2012 International Conference on Systems and Informatics (ICSAI2012)*, pp.529–535 (2012)
- [14] 栗原孝太, 吉野太郎, 和泉諭, 寺邊正大, 橋本和夫, 菅沼拓夫: 電力の有効利用を実現する生活行動プランニング手法, 電子情報通信学会技術研究報告, Vol.112, No.88, IN2012-25, pp.7–12 (2012)
- [15] 吉野太郎, 栗原孝太, 和泉諭, 寺邊正大, 橋本和夫, 菅沼拓夫: 消費電力観測値に基づく生活状況推定システムの設計, 電子情報通信学会技術研究報告, Vol.112, No.88, IN2012-24, pp.1–6 (2012)
- [16] Antonio A. Sánchez-Ruiz, et al. “DLPlan,” <http://dlplan.sourceforge.net/>