マルチメディア通信と分散処理 80－16
グループウェア 21－16
（1997. 1. 30）

エージェント間交渉によるスケジュールの調整方式

喜田弘司 吉府研治 垂水浩幸

{kida,yoshifu,tarumi}@obp.cl.nec.co.jp

NEC

関西 C&C 研究所

オフィスワークにおいて、個人のスケジュールを管理するエージェント（Pochet）が互いに通信を行うことでスケジュールを調整するシステムの提案を行う。具体的に、Pochet は、(1)調整する必要のある作業を検出し、(2)調整案を作成し、(3)調整先の Pochet と調整の交渉を行う。本稿では、Pochet の基本機能とシステム構成を中心に説明した後、 Pochet 間の交渉によるスケジュール調整に関して、調査・説得・更新の３つのフェーズからなるプロトコルを提案する。Pochet は ORB をベースにしたエージェント通信基盤 INA/LI を用いて Windows 上で実装を行っている。

A Schedule Management System Based On Agent Negotiation

Koji Kida, Kenji Yoshifu, Hiroyuki Tarumi

NEC
Kansai C&C Reserch Labs

"Pochet" is an agent oriented schedule management system for reducing incidental tasks. Each user assigned a Pochet inputs personal/group schedules into his/her private calendar. When a Pochet finds problems, such as office work overload, Pochet semi-automatically negotiates with the other Pochets according to the private schedules. If the Pochets reach a compromise, it means that the Pochet have arranged a schedule plan. This system is implemented using our multi-agent platform called INA/LI.

1. はじめに

我々は、インターネットをはじめとする計算機ネットワーク環境を前提に、エージェントと呼ばれるソフトウェアにホワイトカラーが行っている作業を代行させることで、オフィスを可能な限り自動化することを目指したシステム（INA/LI）[1]を開発してきた。具体的に、エージェントはワークフロー管理エージェント[2]、情報のフィルタリングエージェント[3]、スケジュール管理エージェント（Pochet）[4]等から構成される。本稿はPochetに関するものである。

Pochetによるスケジュールの管理は、スケジュールを調整するといった間接的な作業をエージェントに代行させることで、ユーザが本来の業務に集中することができるようにすることが目的である。本稿では、Pochetの基本機能とシステム構成を中心に説明した後、Pochet間の交渉によるスケジュール調整のプロトコルを説明する。

2. Pochetとは

2.1. ねらい

従来のスケジュール管理システム[5-9]は、各ユーザのスケジュールデータをファイルサーバに共有し[1]、ユーザは、必要に応じて、他人のスケジュールデータを参照することで会議の開催日時の決定などを行っていた。このため、

1. 基本的に、先に入力された予定が優先される
2. 共通の空き時間がない場合には、先に入力された予定をキャンセルするといった交渉を、電話や電子メールでする必要がある
3. 会議の管理が中心であり、ToDo作業（締め切り日までに完了させる必要がある作業）の実行時間が考慮されていない
4. スケジュールデータを他人へオープンにすることが前提であるため、プライベートな予定の扱いが難しい

という課題がある。本稿では、前記課題を解決するために、各ユーザの個人スケジュールをユーザに代わってエージェント（Pochet）が管理し、各ユーザが持つPochetが、それぞれのユーザの都合を主張し合い、必要があれば妥協することで、スケジュールの調整を行う手法を提案する。

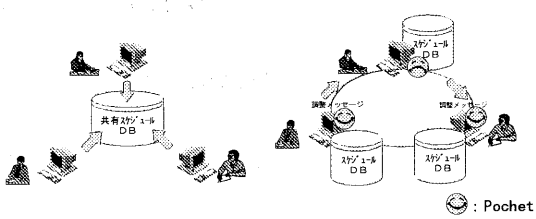

図1　従来手法とPochetによるスケジュール管理

以下に、Pochetの基本機能を示す。

【スケジュールデータの分散管理】（課題4）

ユーザ毎にPochetを持ち、各個人のスケジュールデータを各Pochetが管理する。Pochetはスケジュールデータの公開可能性を管理する[2]。

【忙しさの予測】（課題3）

Pochetはスケジュールのつまり具合から日毎の「忙しさ」を推定する。ここで「忙しさ」とは、データベースに登録された仕事をすべて予定どおりに完了するために、どれくらいの時間をさく必要があるのかを、日毎に推定したものである。具体的には、カレンダー上に濃淡で忙しさをユーザに表示したり（図3）、あるいは他のユーザのPochetからの忙しさの問い合わせに自動的に答える。

【約束型仕事依頼】（課題1、課題3）

Pochetにおける仕事の依頼は、依頼者のPochetと被依頼者のPochetの間で、締め切り日などの仕事の条件を通信しあい、被依頼者が依頼者に「いついつまでに仕事をする」という約束をかわすことで行う。この際、優先度や仕事の忙しさ等を考慮することで、課題1、課題3を解決できる。

【仕事調整の交渉】（課題1、課題2、課題3）

Pochetにおける、スケジュールの調整は、すでに約束を交わした予定の変更を交渉することで行う。例えば、最優先の緊急の会議を開きたい場合、

[1] [8]はスケジュールデータを分散管理している。

[2] 非公開を原則とする。

この会議と日時が重なっている優先度の低い予定のキャンセルの交渉をPochet間で行う。

2.2. 構成

以下にPochetの構成図を示す。

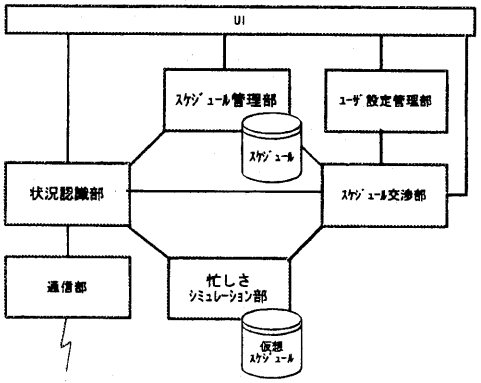

図 2　システム構成図

■スケジュール管理部

従来のスケジューラと同機能を提供するモジュール。スケジュールデータベースを持ち、仕事の登録、更新、削除を行い、カレンダー上に仕事を提示する。また、状況認識部へスケジュールが変更されたことを通知する。

■通信部

他のユーザのPochetや、ワークフローエージェント[2]などと通信を行うモジュール。受信したメッセージをあらかじめ定義されたプロトコルに従って解析し、解析結果を状況認識部に報告する。さらに送信メッセージの作成を行う。通信部が行うメッセージの送受信機能の実装は、INA/LIで開発したORBをベースにしたエージェント通信基盤[1]を用いている。

■状況認識部

スケジュール管理部からのスケジュール変更の報告や、通信部からの他のエージェントからのメッセージの送受信、さらに時間の経過などをきっかけに、締め切り日に間に合いそうにない仕事（以下 *問題タスク*）の検出を行う。状況認識部は問題タスクをユーザに提示したり、あるいは調整するためにスケジュール交渉部へ問題タスクを転送する。

■忙しさシミュレーション部

スケジュールデータベースの内容を仮に変更し、変更後の状況（忙しさ）をシミュレートするモジュール。例えば、ある新規の仕事Xを引き受けた場合をシミュレートする場合、まず、スケジュールデータベースのコピーをとり、このコピーに対して仕事Xを登録する。そして、コピーされたデータベースにおける忙しさを算出する。

■スケジュール交渉部

状況認識部からの問題タスクの検出をきっかけに、調整案の作成、調整案の他のエージェントへの提案、調整案の回答の作成、調整案回答の評価をユーザと対話を行いながらすすめる。また、交渉の履歴をデータベースに蓄え、調整の交渉先や交渉作業の決定を学習する。

■ユーザ設定管理部

エージェント自動動作の設定をユーザがカスタマイズする。例えば、「重要な仕事に関する自動動作はユーザに確認をする。」、「過去に何度も同様の調整を行っている場合はエージェントが自動的に動作する。」といった設定をユーザが行う。

■ＵＩ

通常のカレンダー形式のスケジュール入出力機能に加え（図 3）、交渉のための対話、エージェントが行った自動動作の説明、エージェント動作のカスタマイズ機能を有する。

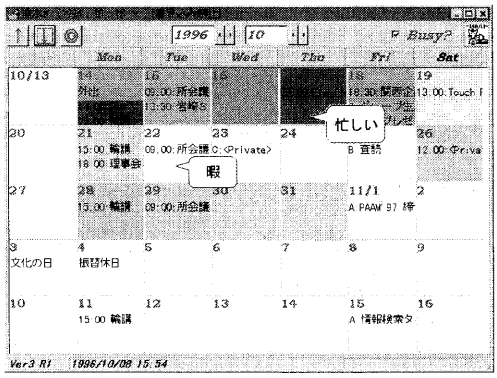

図 3　メイン画面例

3. 交渉の方式

本稿で扱う交渉とは以下のとおりである。

ひとつの交渉には、交渉を依頼する立場の Pochet (*交渉依頼 Pochet*) と交渉を受ける立場の Pochet (*交渉先 Pochet*) が存在する。問題タスクを抱えた Pochet は、交渉依頼 Pochet となって、適当な交渉先 Pochet へ調整案を提案する。調整案を提案された交渉先 Pochet は提案を受け入れるかどうかの判断をして回答する。交渉依頼 Pochet と交渉先 Pochet 間で同意を得るまで交渉を繰り返す。

本章は、交渉依頼 Pochet と交渉先 Pochet の間で交わされる交渉プロトコル関して説明する。

3.1. 効用の要件

各ユーザが自分の好み (効用) を自分の Pochet に与え、各 Pochet が単純に、この効用が高くなるように交渉したならば、以下の課題がある。

課題１：各エージェントが自己に有利になるように交渉を行うため、収束しない可能性がある。

課題２：非常に多くのユーザのスケジューリングの変更を強いられる可能性がある。

実際のオフィスワークの交渉を見てみると、

・あらかじめ、根回しして同意が得られそうな依頼をする。

・上司の意見を優先する。

などを行い、これら課題をうまく逃れている。すなわち、エージェントは、ユーザの好みだけではなく、他のエージェントとの提携も考え合わせた効用を持つことで、ある状況では個人の都合を妥協する仕組みが必要である。本稿では、こういった要件を満たした効用を定義するために、3 フェーズからなるプロトコルを提案する。

3.2. 交渉プロトコル

図４に交渉のフローチャートを示す。以下の３つのフェーズからなる。

【調査フェーズ】

交渉依頼 Pochet→交渉先 Pochet

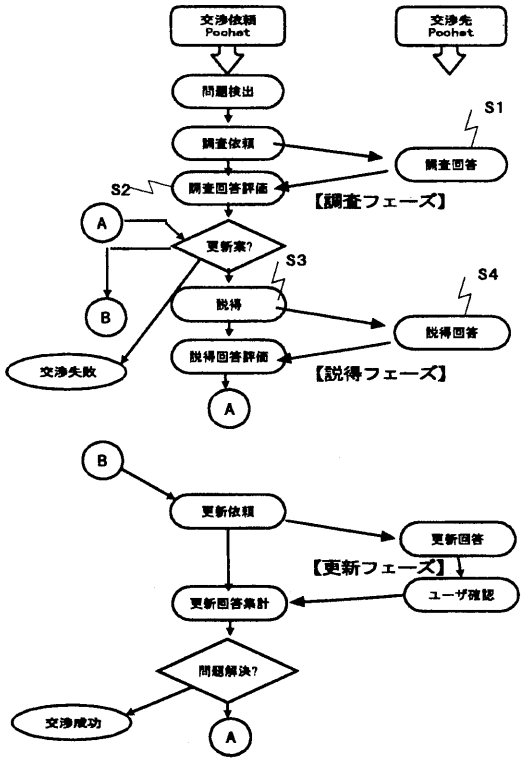

図4　交渉フローチャート

調整案を提案するフェーズ。複数の調整案をそれぞれの Pochet へ依頼する。

交渉依頼 Pochet←交渉先 Pochet

提案が受け入れられるかどうか (OK/NG) を効用値を付加して回答する (S1) [1]。この回答は他の Pochet との提携を考慮せずに、自分の都合で回答する。例えば、表 1 に示したような調査回答ルールを登録しておき、(OK を回答する効用) = (マッチしたルールでアクションが OK の数) − (マッチしたルールでアクションが NG の数) とする。

表１　調査回答ルールの例

条件部	*アクション*
(優先度 <= B) and (完了予測日 < 最終締め切り日)	OK
(仕事名 = 社外論文) and (完了予測 > 最終締め切り日)	NG
:	:

[1] 交渉先 Pochet は原則としてユーザへは確認はしない。

■回答の評価

交渉依頼 Pochet は調査フェーズの結果を集計し、すべての交渉先 Pochet からの回答が NG であった場合、説得案の作成を行う（説得フェーズ）。一つの交渉先Pochet からOKの回答が得られた場合、この Pochet へ正式に調整案を依頼する（更新フェーズ）。複数の交渉先 Pochet からOKの回答が得られた場合、OKの回答に関して、以下に示した三つの基準で評価し(S2)、評価値の最も高い調整案を正式に依頼する。評価値は0以上1以下の実数で表し、値が大きいほど評価が高いことを表す。評価値の計算方法の一例を以下に示す。

問題タスクとの関連性

-調整候補タスクが問題タスク自身の場合は、「1.0」

-調整候補タスクが問題タスクとは異なり、調整候補タスクの依頼者が問題タスクの依頼者と同じ場合は、「0.5」

交渉先の効用

交渉先 Pochet が算出した効用を評価値とする。

過去の交渉履歴

エージェント自身の経験から算出する評価値であり、交渉履歴を検索し、同じ仕事の交渉が成功した例と、失敗した例の履歴数から求める。例えば、同じ仕事の交渉が常に成功していれば、評価値は「1」であり、成功した履歴と失敗した履歴が同数ならば、「0.5」となる。

各回答評価基準に対する重みをユーザがあらかじめ登録しておき、これを基に評価値を算出する。

【説得フェーズ】

交渉依頼 Pochet→交渉先 Pochet

NG を回答してきた交渉先Pochet にOKを返すように、理由をつけて説得するフェーズ。

交渉依頼 Pochet←交渉先 Pochet

説得理由を検討して、説得案を受け入れるかどうか（OK/NG）を交渉先 Pochet が自動的に回答するフェーズ。

■説得案の作成（S3）

交渉先 Pochet からの回答がすべて NG であった場合、NG の回答に関して、前記の評価を行い、評価値がある一定以上であるユーザを説得する。すなわち、問題タスクと関連が深く、NG の回答の効用が低く、以前に調整した実績があるユーザに対して説得を試みる。説得案は調査の内容に評価結果を付加して説得する。評価値がある一定以上のものがない場合には、交渉失敗である。

■説得回答（S4）

説得に対する回答を行うために、以下の項目を使って説得回答ルールを記述しておく。

交渉回数：交渉先 Pochet がカウントしておく

評価結果：交渉依頼 Pochet から送られてくる値

交渉時間：交渉先 Pochet が調査開始時間を記憶しておくことで算出する

説得回答ルールの例を以下に示す。

表2　説得回答ルールの例

【更新フェーズ】

条件部	*アクション*
(優先度 <= B) and (評価結果 > 0.8)	OK
(仕事名 = 社外論文) and (交渉回数 < 2)	NG
:	:

交渉依頼 Pochet→交渉先 Pochet

調査フェーズの結果、あるいは説得フェーズの結果がOKであったものに対して、調整を正式に依頼するフェーズ。

交渉依頼 Pochet←交渉先 Pochet

調整の依頼に対して、ユーザの確認をとって回答するフェーズ。

4. 検討

4.1. 交渉プロトコル

交渉先 Pochet を説得する前に、調査フェーズで交渉先 Pochet の都合を調査するのは、以下の理由からである。

・妥協するために交渉の状況が知りたい（課題1）

交渉先 Pochet が妥協するには、他人が妥協するのではなく、自分が妥協すべきである理由が必要である。本手法は、交渉依頼 Pochet が

調査結果を評価することで妥協すべきPochetの決定を行っている。ただし、この決定は交渉依頼Pochetの一方的な決定であり、交渉先Pochetは断っても構わない。

・無理な説得を行わないため（課題2）

例えば、非常に高い効用値でNGを回答してきたPochetはNGを回答する明確な理由があると考えられ、このPochetに説得を試みるのはナンセンスである。

4.2. 今後の課題

【交渉のためのルールに関して】

通常、人間の場合、妥協するかどうかの判断は、仕事の内容の重要性や、交渉相手との人間関係（上司・部下）や、交渉の状況などを統合的に考え合わせて回答を行っている。これらをルールとしてPochetに与えるために、調査回答ルール、説得回答ルールに要求される記述力を評価する予定である。

【交渉の並列実行】

ある作業の調整を交渉中であっても、別の作業の調整がはじまることがあるため、同時に複数の調整を行うための仕組みが必要である。例えば、ある仕事の交渉中であっても、さらに優先度の高い仕事の交渉が開始された場合、前の仕事の交渉を一時中断させるような仕組みが必要である。

【ユーザの入力手間の問題】

各ユーザがきちんと自分のスケジュールをPochetに入力することは、ユーザに負担であると予測される。本研究では、入力のユーザインタフェースの改善、入力負担を上回るサービスの提供という二点からアプローチをとる。今後、システムがうまく動作するために、ユーザにどの程度まで大雑把な入力を許せるのかを評価する予定である。

5. おわりに

本論文では、スケジュールの調整をエージェント間の交渉で行うシステムの提案を行った。これは、いわゆる根回しをエージェントに代行させていることに相当する。エージェント間交渉は調査、説得、更新の3つのフェーズからなり、個人の都合を主張するだけでなく、他のエージェントとの提携も考え合わせる仕組みを提案した。

現在、本システムはスケジュールの問い合わせを行う等の簡単な利用を所内で開始している。今後、本稿で述べた交渉機能等の実装を行い、より多くのユーザでの試用を通して、システムを評価する予定である。

参考文献

[1]Yoshihide Ishiguro, et al.: An Agent Architecture for Personal and Group Work Support, ICMAS'96, pp. 134-141(1996).

[2]垂水ほか:「ワークウェブシステムの実現」, 情報処理学会グループウェア研究会, GW-15-22(1996).

[3]朝倉ほか:「エージェントによる情報フィルタリング」, 情報処理学会 情報メディア研究会, IM-20-9(1995).

[4]喜田ほか:「仕事依頼交渉を支援するスケジューラエージェント」, 情報処理学会第52回全国大会論文集, 1X-7(1996).

[5]Microsoft Windows NT Workstation 3.51 システムガイド Schedule+, pp. 285-304(1996).

[6]ノベル GroupWise カタログ(1996).

[7] Thomas Kreifelts, et al.: Sharing To-Do Lists with a Distributed Task Manager, ECSCW-'93, pp. 31-45(1993).

[8] Yuji WADA, et al.: An Agent Oriented Schedule Management System -IntelliDiary, PAAM96, pp. 655-667(1996).

[9] Sandip Sen, et al.: Unsupervised Surrogate Agents and Search Bias Change in Flexible Distributed Scheduling, ICMAS-95, pp. 336-343(1995).