

第3世代ESシェル(ARES)スケジューリング問題向けタスク特化シェル

3D-8

- 知識獲得支援機構 - *

三條 水奈子 荒木 大 成松 克己 小島 昌一†
(株)東芝 研究開発センター‡

1 はじめに

スケジューリング型タスク特化シェル [1] [2] 上に構築した知識獲得支援機構について述べる。この機構は、タスク特化シェルに入力されている知識をユーザが洗練する過程、言い換えれば知識をデバッグする過程を支援するものである。

タスク特化シェルに入力される知識には [2] で述べたように、スケジューリング対象のドメインに関する定義、スケジューリング手順、スケジュール結果が満たすべき制約条件の3つに大きく分ける事ができる。また制約条件には、対象ドメインの定義として表現するものと、制約条件エディタで表現するものがあるが、本稿では、後者で表現する制約条件知識の洗練を支援する機構の概要を述べる。この機構は、ユーザがガントチャート上で修正を行なうスケジュール結果の変更に対して、制約条件知識の中で矛盾が発生する制約条件を検出し、その矛盾の解消方法を自動提案する機能である。

2 制約条件の矛盾とその解消

まず、シェルが出力したスケジュール結果をユーザが修正することによって発生する制約条件の矛盾と、これを解消するよう制約条件の記述を変更するという問題を定義する。

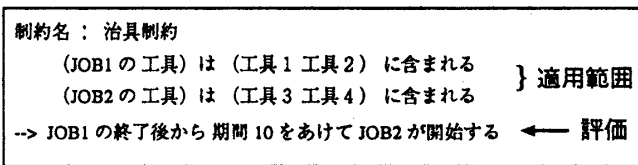


図1: 制約条件の記述例

図1は特定の工具を使用するジョブが連続する場合に、間隔をあけておく必要があることを意味する制約である。このように制約条件知識の中に蓄えられた個々の制約条件 C の記述は、制約条件の適用を受けるジョブあるいは休止期間に求められる条件を記した適用範囲 $Range$ と、適用範囲に合致したジョブ対ジョブあるいはジョブ対休止期間の時間的な位置関係に対して求められる条件を記した評価 $Eval$ からなる。

一般にあるスケジュール結果に対して制約条件 C を評価すると、適用範囲 $Range$ を満たしかつ評価 $Eval$ が真となるジョブあるいは休止期間の組の集合(正例集合 Pos)、適用範囲 $Range$ を満たすが評価 $Eval$ が偽となる組の集合(負例集合 Neg)、及

び適用範囲 $Range$ を満たさない組の集合(範囲外集合 Out) が得られる。

シェルが出力したスケジュール結果はすべての制約条件が満たされた状態になっている。いいかえれば、どの制約条件を評価しても負例集合は $Neg = \{\}$ である。一方で、ユーザがスケジュール結果を修正した後では $Neg \neq \{\}$ の制約条件 C_{conf} が現われる可能性がある。これを制約条件の矛盾と考える。矛盾の解消とは $Neg \neq \{\}$ となる制約条件 C_{conf} の記述を C'_{conf} に変更することによって $Neg' = \{\}$ とする問題であるといえる(図2)。

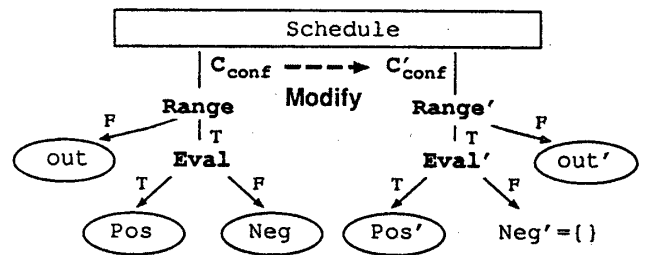


図2: 矛盾の解消

次に、 C_{conf} の記述を C'_{conf} に変更する方法について考察する。制約条件の記述は適用範囲 $Range$ と評価 $Eval$ からなるが、 $Eval$ の記述変更を行なう場合は、 $Eval$ の種別の変更は行なわずに期間の変更に限る。なぜならば、 $Eval$ の種別を変更することはその記述の意味を根本的に変えることになるからである。例えば図1では、「期間10」を変更することは考えるが、「間隔をあけて開始する」という制約種別を他の種別、例えば「期間の重なりがないようにする」に変更することは考えない。

また、 $Range$ の変更方法は、条件を強める方向への修正のみを考える。なぜならば、条件を弱める方向への修正を行なうと $Range$ を充足するインスタンスは増える方向に作用し $Neg' = \{\}$ は達成できないからである。具体的にどのような記述変更を考慮するかは次章で述べる。

図2の矛盾解消においては、記述の修正の前後で正例集合にできるだけ変化が起きない、すなわち $Pos \simeq Pos'$ が望ましいといえる。これと、 $Neg' = \{\}$ が望ましいという目標を加味して、本知識獲得機構では式(1)にある記述修正 mod に対する評価尺度 E として採用している。

$$E(mod) = \left\{ 1 - \frac{|Neg'|}{|Neg|} \right\} \left\{ \frac{|Pos \cap Pos'|}{|Pos|} \right\} \quad (1)$$

*Constraint Refinement on Task Specific ES Shell for Scheduling
†Minako SANJOH, Dai ARAKI, Katsumi NARIMATSU, Shoichi KOJIMA
‡Research & Development Center, TOSHIBA Corp.

3 制約条件の修正

制約条件 C_{conf} の適用範囲 $Range$ の記述を修正する方法について述べる。なお評価 $Eval$ の期間の変更については、以下で述べる (b-2) のしきい値の決め方と同じ手法を用いる。

3.1 適用範囲 $Range$ の記述形式

制約条件 C の適用範囲 $Range$ は、論理式の列で表現される。たとえば図1は、「(JOB 1の工具) は (工具1 工具2) に含まれる」と「(JOB 2の工具) は (工具3 工具4) に含まれる」の2つの論理式からなる。

論理式には次の3種類がある。

(1) リストに対する包含関係を調べる論理式

「(JOB 1の工具) は (工具1 工具2) に含まれる」

これは、(JOB 1の工具) に代入される値が、(工具1 工具2) というリストに含まれるかどうかを評価する論理式である。これ以外に、リストに「含まれない」関係を調べる論理式もある。

(2) 値の大小比較を行なう論理式

2つの値の大小関係を評価する論理式である。

「(JOB 1の優先度) は (5) より大きい」

これは、(JOB 1の優先度) に代入される値が、比較対象値 (5) より大きいかどうかを評価する。

(3) 変数への代入を行なう論理式

次の例は、(JOB 1の納期) に代入される値を変数 X に代入する論理式である。(注: 変数は区別するため () で括弧していない)

「(JOB 1の納期) を X に代入する」

3.2 制約条件の修正規則

制約条件 C_{conf} を修正する方法を「修正規則」と呼ぶ。修正規則には以下の種類がある。

(a) 既存の論理式の内容を変更する方法

(a-1) リストへの包含関係を調べる論理式の変更

論理式が「含まれる」型の場合を説明する。例えば、

「(JOB 1の工具) は (工具1 工具2) に含まれる」

の場合、負例集合 Neg のインスタンスにおいて (JOB 1の工具) に代入された値をリストから削除することにより、論理式が負例に対して充足されないようにする。「含まれない」型の論理式の場合は、負例に対して代入された値をリストに追加する。

(a-2) 値の大小比較を行なう論理式の変更

例えば

「(JOB 1の優先度) は (5) より大きい」

の場合、負例集合 Neg のインスタンスに対してこの論理式の評価が偽になるように、比較対象値 (5) を変更する。例えば、負例において (JOB 1の優先度) に代入された値が 6 の場合は、比較対象値を (6) に変更する。

(b) 論理式を追加する方法

論理式追加については様々な方法が考えられるが、ここでは次の2つの戦略を用いている。

(b-1) 包含関係を調べる論理式を追加

例えば、

「(JOB 1の種別分類) は (A 分類) に含まれない」

という論理式を追加する。ここで、どの属性を参照するかは、ユーザが予め候補として指定しておいた属性から選択する。リストに入れる値は、負例集合 Neg のインスタンスにおいて (JOB 1の種別分類) に代入されたすべての値とする。この戦略は、「ジョブの種別分類」のような列挙型の属性に対して用いる。

(b-2) 大小比較を行なう論理式を追加

例えば、

「(JOB 1の優先度) は (6) より大きい」

という論理式を追加する。ここで、どの属性を参照するかは、ユーザが予め候補として指定しておいた属性から選択する。比較対象値は、負例集合 Neg のインスタンスにおいて (JOB 1の優先度) に代入された値の集合と正例集合 Pos のインスタンスにおいて代入された値の集合との間のしきい値とし、その大小関係を論理式の種別として決定する。この戦略は、「優先度」「処理時間」といった連続値型の属性に対して用いる。

4 知識獲得支援機構の動作

知識獲得支援機構は以下の順で動作する。

- (1) 修正対象制約条件の発見
スケジューリング結果と矛盾する制約条件を修正対象制約条件 C_{conf} として発見する。
- (2) 修正方法の作成
 C_{conf} に対して、3.2節で述べた修正規則を順次適用して C'_{conf} の候補を作成する。この際に、どの修正規則を試みるか、どの属性に対する修正を試みるかはユーザが指定できるようにしている。
- (3) 修正方法の評価
それぞれの C'_{conf} を式 (1) で評価し、評価尺度 E がユーザがあらかじめ指定した基準値より大きい C'_{conf} を表示する。

5 まとめ

タスク特化シェル上に構築した知識獲得支援機構について述べた。この機構は、タスク特化シェルが出力したスケジュール結果に対して、ユーザがガントチャート上で修正を行なった際に利用し、矛盾制約条件の検出と、その制約条件の修正方法の自動提案を行なう。これにより、ユーザの知識洗練過程を支援する事が可能になった。

参考文献

- [1] 荒木, 他: 第3世代ESシェル (ARES) における特定問題向けタスク特化シェル群, 情報処理学会第46回全国大会, 3D-4, 1993.
- [2] 成松, 他: 第3世代ESシェル (ARES) スケジューリング問題向けタスク特化シェル-推論エンジン構築メカニズム-, 情報処理学会第46回全国大会, 3D-7, 1993.