

複製を含む双方向変換とその応用に関する研究

46227 松田 一孝

指導教員 胡 振江 助教授

2006年2月8日

1 背景

双方向変換 [1, 2] は, 2つの異なるデータ間の同期を行う技術であり, ソースと呼ばれる元のデータからビューと呼ばれる別のデータを得る順方向の変換と, ビューの変更をソースに書き戻す逆方向の変換から構成される. 近年, この双方向変換の枠組みを拡張し, 複製を導入することにより, データ内の複製された要素の同期を行うことができることが示された [2]. それにより, データ内の依存性を表現することができ, 様々なアプリケーションの作成に有用であると考えられている.

複製を含まない双方向変換においてはその満たすべき性質に対し十分な議論が行われているにもかかわらず, 複製を含む双方向変換については, 十分な議論が行われていない. たとえば, 複製を含まない双方向変換では, 変更されたビューと変更が書き戻されたソースのビューが同じであるという性質が変換の組合せについて保存されるが, 複製を含む双方向変換についてはこの性質は必ずしも成り立たない. また, 双方向変換を応用したアプリケーションもあまりない.

そこで本研究では, 複製を含む双方向変換について, その満たすべき性質について議論する. そして, 応用例として, 双方向変換を利用したファイルマネージャ「梅林」の実現を示す.

2 複製を含まない双方向変換

双方向変換は, ソースと呼ばれるデータからビューと呼ばれるデータを得る順方向変換とビューに加えた変更を書き戻す逆方向変換から構成される [1]. つまり, 以下のように定式化される.

定義 1 (双方向変換). ソース S からビュー V への双方向変換 t は, 順方向変換 $[[t]]_G :: S \rightarrow V$ と逆方向変換 $[[t]]_P :: (S, V) \rightarrow S$ の2つから構成される. \square

ここで, 逆方向変換が変更前のソースを取ることによって, 射影変換

$$\begin{aligned} [[fst]]_G (s_1, s_2) &= s_1 \\ [[fst]]_P ((s_1, s_2), v) &= (v, s_2) \end{aligned}$$

などを表現することができる.

しかし, 全ての双方向変換が我々にとって必要なものではない. ビューに変更がない場合にソースは変更されるべきではないし, ビューに変更があった場合は, 変更されたソースのビューが変更されたビューと同じになるようにソースが変更されてほしい. この2つの性質を満たすことを正当性として定式化すると以下ようになる [1].

定義 2 (正当な双方向変換). 次の2つを満たす双方向変換を正当な双方向変換と呼ぶ.

- 反射性:
 $\forall s \in S, [[t]]_P (s, [[t]]_G s) \sqsubseteq s$
- 変更保存性:
 $\forall (s, v) \in (S, V), [[t]]_G ([[t]]_P (s, v)) \sqsubseteq v$

但し, $a \sqsubseteq b$ は a が未定義 \perp でなければ $a = b$ であることを表す. \square

このとき, 以下のように定義される双方向変換の合成

$$\begin{aligned} [[t_1 \hat{\;} t_2]]_G s &= [[t_2]]_G ([[t_1]]_G s) \\ [[t_1 \hat{\;} t_2]]_P (s, v) &= [[t_1]]_P (s, [[t_2]]_P ([[t_1]]_G s, v)) \end{aligned}$$

について, 次の定理が成り立つ [1].

定理 3. 双方向変換が t_1 と t_2 がともに反射的なら, $t_1 \hat{\;} t_2$ も反射的, とともに変更保存的なら $t_1 \hat{\;} t_2$ も変更保存的. \square

3 複製を含む双方向変換とその性質に関する機能

複製は、 $[[\delta]]_G s = (s, s)$ と定義される変換である。この逆方向変換を

$$[[\delta]]_P (s, (v_1, v_2)) = \text{if } v_1 = v_2 \text{ then } v_1 \text{ else } \perp$$

と定義すると変更保存性を満たすが、複製された箇所に同じ変更しなければならず使いづらい。一方で、

$$[[\delta]]_P (s, (v_1, v_2)) = \text{if } v_1 = s \text{ then } v_2 \text{ else if } v_2 = s \text{ then } v_1 \text{ else } \perp$$

と定義すると、ビューにおける片方の変更をソースとビューのもう一方に伝播でき、ビュー中の依存性を表現でき便利 [2] であるが、変更保存性を満たさない。複製変換による利便性を享受しつつ、双方向変換による「よい」性質の議論を行うために、変更保存性に代わる性質を定めなければならない。

複製はビューを 2 つ作成することであると考えると、そのそれぞれに対する局所的な変更保存性について考える。このとき、変換の種類を制限した場合に変更保存性に代わる以下のような性質を定義できる。

定義 4 (局所変更保存性). ソース S からビュー \mathcal{V} への双方向変換 t が局所変更保存性を満たすとは、

- $\mathcal{V} = (\mathcal{V}_1, \mathcal{V}_2)$ であるとき $t \hat{;} fst$ と $t \hat{;} snd$ が局所変更保存性を満たす、
- そうでない場合 t が変更保存性を満たす、

ことをいう。□

論文中では、この性質を満たすかどうかを構文的かつ静的にある程度解析できることを示した。

4 双方向変換の応用

双方向変換の応用例としてファイルマネージャ「梅林」を作成した [3]。ディレクトリ構造をソースとし、ユーザは変換によって得られたビュー上で、ファイルの作成などの操作を行う。

双方向変換をファイルマネージャに応用することにより以下の利点が得られる。

対称な同期されたファイル参照の実現 ショートカットと異なり、一方の改名、削除をもう一方に反映することができる。

双方向変換による統一的な取り扱い ソート、お気に入りリスト、検索などを双方向変換を通して統一的に扱うことができる。

ビューの切り替え 音楽ファイルの時代ごとのソートや、作曲家ごとのソートなどのビューを適宜切り替えることができる。また、複製変換によりそれらを同時に扱うこともできる。

5 まとめ

複製を含む双方向変換の満たすべき性質として、局所変更保存性を提案し、またその性質に対し解析を行った。さらに、複製を含む双方向変換の応用例としてファイルマネージャを作成した。ここでは、ショートカットのようなファイル参照の同期が対称に表現され、そのような同期構造が双方向変換を通じて構造化される。

参考文献

- [1] J. N. Foster, M. B. Greenwald, J. T. Moore, B. C. Pierce, and A. Schmitt. Combinators for bi-directional tree transformations: a linguistic approach to the view update problem. In *POPL '05: Proceedings of the 32nd ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, pp. 233–246, New York, NY, USA, 2005. ACM Press.
- [2] Z. Hu, S.-C. Mu, and M. Takeichi. A programmable editor for developing structured documents based on bidirectional transformations. In *PEPM '04: Proceedings of the 2004 ACM SIGPLAN symposium on Partial evaluation and semantics-based program manipulation*, pp. 178–189, New York, NY, USA, 2004. ACM Press.
- [3] 松田一孝, 大川徳之, 野村芳明, 森田直幸, 箕一彦, 胡振江, 武市正人. 木上の双方向変換を利用したファイルマネージャの実現. 情報処理学会論文誌: トランザクション「プログラミング」, 2005. 掲載予定.