











に、提示した問題に対して部品の要素から正しい振舞いと解答が保持されているため、それに基づく正誤判定や誤り箇所のフィードバックも可能となる。

次に問題の説明機能については、学習者に振舞いと構造を個別に解答させることで、学習者が **Decomposition** のプロセスにおいて機能から要求される振舞いへの変換が誤っているか、振舞いから表現する構造への変換が誤っているかを判断し、それに対してどこがどのように誤っているかを学習者に説明することができる。

ただし上記の機能により、**Decomposition** プロセスの支援を過剰に行ってしまうと、**Decomposition** で導き出された正しい構造を **Composition** で生成するだけの活動になってしまい、学習者が **Composition** のプロセスにおいて修正活動を十分に経験できず、構造の振舞いを観察して自身の構造を振り返るといった、試行錯誤のプロセスとして期待した活動が行われないといった問題点がある。これは、構造に基づく振舞いの生成と観察が実世界で行われる **Composition** でしか経験できないことに由来する。そのため、システムは **Decomposition** プロセスと **Composition** プロセスが存在することを把握しながらも、**Decomposition** での支援の量を調整することで、学習者の試行錯誤を促す **Composition** のプロセスを行わせる必要があるといえる。

さらに、学習者の知識状態の推定機能について説明する。システムが知識として部品の集合を持っていることで、ある部品に内包される部品同士の関係性が把握できているため、半順序構造として部品全体の関係性を把握している。これにより、学習者が解けた問題を獲得した部品とみなし、学習者がどの部品を持っているかを把握することで知識状態を推定することができる。また、この知識状態をもとに問題を提示して問題に解答できなかった場合は、誤った箇所の部品にバグがあるとみなし、その部品に対する理解のための補助問題を提示することが可能になる。

## 5. おわりに

本稿では、これまでに本研究で議論してきた内容 [6–8] をモデル化に基づいて整理した。まず、プログラミングにおいて部品が何であるかを、プログラミングの要素を言語独立に定義することで説明した。また一方で、部品の構築活動について検討し、**Decomposition** と **Composition** の活動を定義してこれを説明した。さらに、これらの定義を併せて構築した、部品の獲得・拡張活動プロセスを提案した。

以上の取り組みを通して、本稿ではこれまで暗黙

知とされていたプログラミングに関する諸要素に接近し、学習の観点から暗黙知を形式知化した。これにより、これまで整理されていなかったプログラミングにおける知識を形式知として定義し、知識を取り扱うためのモデルの一つを示した。

今後の課題としては、振舞いの表現として複雑となりうる if 文や for 文などの制御構文を取扱うことや、本モデルを実装したシステムの構築が挙げられる。特に振舞いの表現については、定性物理で「水飲み鳥のおもちゃ」の因果ネットワークモデル[1]が取り上げられていることが、「条件分岐」や「繰り返し」に対する今後の表現可能性を示唆している。

## 謝辞

本研究の一部は科研費・基盤研究(C)(18K11586)、科研費・基盤研究(B)(17H01839)の助成による。

## 参考文献

- [1] 西田豊明: 定性推論の諸相, 朝倉書店, (1993)
- [2] Kleer, J. de, and Brown, J. S.: A qualitative physics based on confluences, *Artificial Intelligence*, Vol. 24, No. 1, pp. 7–83, (1984)
- [3] 笹島宗彦, 來村徳信, 池田満, 溝口理一郎: 機能と振舞いのオントロジーに基づく機能モデル表現言語 FBRL の開発, 人工知能学会誌, Vol. 11, No. 3, pp. 420–431, (1996)
- [4] Shneiderman, B., and Mayer, R.: Syntactic/semantic interactions in programmer behavior: A model and experimental results, *International Journal of Parallel Programming*, Vol. 8, No. 3, pp. 219–238, (1979)
- [5] Mayrhauser, A. Von, and Vans, A. M.: Program comprehension during software maintenance and evolution, *Computer*, Vol. 28, No. 8, pp. 44–55, (1995)
- [6] 古池謙人, 東本崇仁: プログラミングにおける構造的な理解のための部品の段階的拡張手法の提案とそのシステムの開発, 教育システム情報学会誌, Vol. 35, No. 2, pp. 215–220, (2018)
- [7] Koike, K., Tomoto, T., Horiguchi, T., and Hirashima, T.: Proposal of a Framework for Stepwise Task Sequence in Programming, *International Conference on Human Interface and the Management of Information*, pp. 266–277, (2018)
- [8] Koike, K., Tomoto, T., Horiguchi, T., and Hirashima, T.: Proposal of an Adaptive Programming-Learning Support System Utilizing Structuralized Tasks, *Workshop proceedings of the International Conference on Computers in Education ICCE 2018*, pp. 278–287, (2018)