CEUR-WS.org/Vol-3227/Cai-et-al.PP6.pdf

Repairing Numerical Equations in Analogically Blended Theories Using Reformation

Cheng-Hao Cai, Alan Bundy

School of Informatics, University of Edinburgh, Edinburgh EH8 9AB, UK

Abstract

The ABC system supports analogical abduction algorithms for knowledge transfer, e.g., existing logical rules are adapted, by reformation, into new rules by substituting symbols for similar ones. Although such knowledge transfer method can easily expand knowledge sets, it is likely to produce inconsistent knowledge, e.g., equations that do not fit target data in the real world. To solve this problem, we extend the classic reformation algorithm to repair numerical equations that violate target data. Equation reformation is achieved by weakening equation parameters when unblocking failed SLD-resolution proofs. The feasibility of numerical equation reformation is demonstrated by the automated repair of a faulty electrostatic force equation that is analogically transferred from the gravity equation.

Keywords

Reformation, Theory repair, Analogical reasoning, Knowledge transfer

1. Introduction

Analogical reasoning is common in human thinking, ranging from knowledge acquisition and language learning, to scientific problem solving and program design. For instance, the analogy between natural and computational systems has led to various natural computing algorithms, e.g., genetic algorithms [1], ant colony optimisation [2] and particle swarm optimisation [3]. Existing studies [4, 5] have found that analogical reasoning can facilitate problem solving in both human mind and expert systems. The merger of analogical reasoning and formal methods has led to various approaches of human-like computing, e.g., conceptual blending [6], cased-based learning [7] and structure transfer learning [8]. These approaches can facilitate knowledge discovery and problem solving by introducing models from analogous domains. However, analogies may be faulty due to inevitable differences between two analogous domains. For instance, even though the movements of stars ruled by gravity are similar to the movements of charges ruled by electric forces, the theory of gravity cannot be directly transferred to describe the movements of charges unless the theory is adapted appropriately. This gives rise to the need for repairing the blended theories produced by analogical reasoning, e.g., using the reformation algorithm, to diagnose and repair inappropriate alignments between two ontologies [9].

Existing studies have focused on the formalism of analogical reasoning and theory repair. In the HDTP framework [10], analogies can be formalised by mapping terms from a source theory

HLC 2022: The 3rd International Workshop on Human-Like Computing, September 28-30, 2022, Windsor Great Park, UK

chenghao.cai@outlook.com (C. Cai); a.bundy@ed.ac.uk (A. Bundy)
0000-0001-6815-9091 (C. Cai); 0000-0002-0578-6474 (A. Bundy)

© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

to a target theory. Such mappings can be applied to existing logical rules in the source theory, leading to new rules in the target theory. Apart from HDTP, faulty theories may be repaired using the reformation algorithm supported by the ABC theory repair system [11, 12, 13, 14, 15], which revises logical rules by enabling expected rule applications and disabling unexpected ones. However, as reformation uses the pure symbolic unification [16] that deals with predicates rather than numbers, numerical expressions may not be appropriately repaired.

In this work, we intend to develop a model that formalises human analogical reason. As there are many prior studies [17, 18, 10, 8] on how to compute analogy mappings, our work focuses on a post-processing algorithm that repairs faulty analogy mappings. We start from extending reformation to automate the repair of faulty numerical equations produced by human analogy, whereas the automation of the whole analogy process is considered as future work. Numerical equations are of great importance in natural science and engineering as they can be used to formalise data, simplify problems and make calculations. The integration of reformation and numerical computation is likely to expand the application of theory repair in real world problems. Although our approach is work-in-progress and not fully implemented, at the first stage, we have the following contributions:

- The reformation algorithm is extended to repair numerical equations.
- The algorithm is used to automatically repair an analogical blend of gravity and electrostatic force.

2. The Reformation of Numerical Equations

Analogous numerical equations are often used in our life. For instance, when shopping online, people may compare products by functionalities and prices, i.e., people use equations of cost-performance ratio to score the products in mind. To compare two products, their information needs to be aligned, so that they can be scored by similar equations. Our earlier work [9] gives some insight on how to align two product models and use reformation to make the alignment consistent.

We will show that similar techniques can also be used to develop a physical theory of a new domain from one for an old domain. For instance, in electrical engineering, power flows are considered similar to water flows, so that many equations, e.g., those of potential energy, work and wave, can be adapted to model both power grids and hydropower systems. The correctness of the adapted equations is judged using real-world data. Due to combinatorial explosions, adapting and judging equations is time-consuming for humans. We use reformation to automate the process.

We extend reformation [13] to equational reasoning by introducing a numerical constraint solver, e.g., the Z3 solver [19], into the unblocking function. During SLD-resolution [20], unblocking can repair faulty definite clauses by enabling wanted unifications, i.e., if the unification of two compound terms $f(s_1, \ldots, s_n)$ and $g(t_1, \ldots, t_m)$ fails, minimal changes will be applied to the terms or their sub-terms to achieve a successful unification. If the terms can be numerically evaluated, then the equation $f(s_1, \ldots, s_n) = g(t_1, \ldots, t_m)$ can be solved using the Z3 solver. If the solver fails, then the equation will be weakened by various repair operators, such as replacing constants with unknowns and introducing new functions with unknowns. The weakened equations can be solved by the solver to determine the values of unknowns. By replacing the original equation with the weakened equation, the unification becomes successful.

The proposed approach is implemented as a resolution module and an equation repair module.[1] The resolution module, based on SWI-Prolog [21], can recursively resolve goal problems into sub-problems by user-specified rules until no rules can be applied. The equation repair module, based on Z3 [19] and Python [22], can try to solve equations in the sub-problems and repair unsolvable equations. The suggested repairs are applied to the user-specified rules, resulting in a set of updated rules that will be checked by the resolution module. With the implementation, we suggest the following steps to achieve equation transfer and repair.

1. A source theory and a set of analogical relations are provided by humans to obtain a blended theory.
2. A set of target data is provided by humans to verify the blended theory.
3. If any target data results in failed proofs, then reformation is used to unblock the proofs, which causes any equations violating the target data to be repaired.

3. Case Study: Gravity and Electrostatic Force

Gravity and electrostatic force seem to be analogous as both of them act on objects without actual contact. Gravity is subject to masses and positions of objects, whereas electrostatic force is subject to charges and positions of objects. However, gravity only *attracts* objects, whereas electrostatic force can either attract or *repel* objects. The following process shows how to use reformation to adapt Newton's equation of gravity to Coulomb's equation of electrostatic force. Given the equation of gravity as the source theory:

$$\begin{aligned}&Mass(p_1,m_1) \wedge Position(p_1,x_1,y_1,z_1) \wedge Mass(p_2,m_2) \wedge Position(p_2,x_2,y_2,z_2) \implies \\ &Gravity(p_1,p_2) = \underbrace{\frac{6.67 \times 10^{-11} \cdot m_1 \cdot m_2}{(x_2-x_1)^2+(y_2-y_1)^2+(z_2-z_1)^2}}_{\texttt{Magnitude}} \cdot \underbrace{\frac{(x_2-x_1,\ y_2-y_1,\ z_2-z_1)}{\sqrt{(x_2-x_1)^2+(y_2-y_1)^2+(z_2-z_1)^2}}}_{\texttt{Direction}}\end{aligned} \tag{1}$$

$Mass(p,m)$ means that a point-like object p has mass m. $Position(p,x,y,z)$ means that the coordinate of the object p in a 3D space is (x,y,z). $Gravity(p_1,p_2)$ is the gravity that an object p_2 acts on another object p_1, which is a 3D vector with a magnitude and a direction. By analogical relations $Mass \mapsto Charge$ and $Gravity \mapsto EleForce$, a blended theory can be obtained:

$$\begin{aligned}&Charge(p_1,q_1) \wedge Position(p_1,x_1,y_1,z_1) \wedge Charge(p_2,q_2) \wedge Position(p_2,x_2,y_2,z_2) \implies \\ &EleForce(p_1,p_2) = \frac{6.67 \times 10^{-11} \cdot q_1 \cdot q_2}{(x_2-x_1)^2+(y_2-y_1)^2+(z_2-z_1)^2} \cdot \frac{(x_2-x_1,\ y_2-y_1,\ z_2-z_1)}{\sqrt{(x_2-x_1)^2+(y_2-y_1)^2+(z_2-z_1)^2}}\end{aligned} \tag{2}$$

$Charge(p,q)$ means that a point-like object p has charges q. $EleForce(p_1,p_2)$ is the electrostatic force that a charged object p_2 acts on another charged object p_1. The blended theory is expected to prove a goal that represents target data:

$$\begin{aligned}&Charge(P_1,\ 4 \times 10^{-8}) \ \wedge\ Position(P_1,\ 0.1,\ 0,\ 0) \ \wedge\ Charge(P_2,\ 4 \times 10^{-8}) \ \wedge \\ &Position(P_2,\ 0,\ 0,\ 0) \implies EleForce(P_1,\ P_2) = (1.4384 \times 10^{-3},\ 0,\ 0)\end{aligned} \tag{3}$$

[1] The source code is available via https://github.com/cchrewrite/Numerical-Equation-Repair.

During the resolution to the goal, the equation $EleForce(P_1,\ P_2) = (1.4384 \times 10^{-3},\ 0,\ 0)$ fails. To weaken the equation, the constant 6.67×10^{-11} is replaced with an unknown k and solved by the constraint solver, leading to the following repaired theory with $k = -8.99 \times 10^9$:

$$
\begin{aligned}
&Charge(p_1, q_1) \wedge Position(p_1, x_1, y_1, z_1) \wedge Charge(p_2, q_2) \wedge Position(p_2, x_2, y_2, z_2) \implies \\
&EleForce(p_1, p_2) = \frac{-8.99 \times 10^9 \cdot q_1 \cdot q_2}{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \cdot \frac{(x_2 - x_1,\ y_2 - y_1,\ z_2 - z_1)}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}}
\end{aligned}
\tag{4}
$$

The repaired theory fits Coulomb's equation of electrostatic force as the revised parameter -8.99×10^9 fits Coulomb's constant ($k_e \approx 8.99 \times 10^9$), which means that the equation transfer is successful. The parameter is negative because charges are signed, i.e., positive charges attract negative charges and repel positive charges, and vice versa.

4. Conclusions

We have demonstrated the feasibility for using reformation to repair numerical equations in analogical blends, i.e., a failed unification with equations may be unblocked by weakening the equations. This approach can repair a wide range of equations if they meet three requirements:

1. all variables in the equations can be instantiated by SLD-resolution and user-specified rules,
2. all operators in the equations are supported by the constraint solver, and
3. the equations can be made true by finding alternative constants.

As the capability of equation repair is limited by constraint solvers, satisfiability of equations, etc., we may introduce symbolic constraint solvers and explore how to generally repair equations in larger theories in the future. Additionally, combining with analogy mapping algorithms, our approach will have potential applications in scientific knowledge discovery. In other words, equations describing the characteristics of an object, e.g., a material, a species and a (natural or social) phenomenon, can be transferred to describe another similar object. Our approach might be used to automatically repair some transferred equations that violate real-world data, so that the efficiency of research can be improved.

Acknowledgments

For the purpose of open access, the author has applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising from this submission.

References

[1] S. Katoch, S. S. Chauhan, V. Kumar, A review on genetic algorithm: past, present, and future, Multimedia Tools and Applications 80 (2021) 8091–8126. doi:10.1007/s11042-020-10139-6.

[2] M. Dorigo, C. Blum, Ant colony optimization theory: A survey, Theoretical Computer Science 344 (2005) 243–278. doi:10.1016/j.tcs.2005.05.020.

[3] A. R. Jordehi, J. Jasni, Parameter selection in particle swarm optimisation: a survey, Journal of Experimental and Theoretical Artificial Intelligence 25 (2013) 527–542. doi:10.1080/0952813X.2013.782348.

[4] K. J. Holyoak, K. Koh, Surface and structural similarity in analogical transfer, Memory and Cognition 15 (1987) 332–340. doi:10.3758/BF03197035.

[5] L. Eliot, Analogical problem-solving and expert systems, IEEE Expert 1 (1986) 17–28. doi:10.1109/MEX.1986.4306951.

[6] M. Eppe, E. Maclean, R. Confalonieri, O. Kutz, M. Schorlemmer, E. Plaza, K. Kühnberger, A computational framework for conceptual blending, Artificial Intelligence 256 (2018) 105–129. doi:10.1016/j.artint.2017.11.005.

[7] T. Liang, Analogical reasoning and case-based learning in model management systems, Decision Support Systems 10 (1993) 137–160. doi:10.1016/0167-9236(93)90035-2.

[8] T. R. Hinrichs, K. D. Forbus, Transfer learning through analogy in games, AI Magazine 32 (2011) 70–83. doi:10.1609/aimag.v32i1.2332.

[9] A. Bundy, E. Maclean, The use of reformation to repair faulty analogical blends, in: the 5th UK Ontology Network, Newcastle upon Tyne, UK, April 14, 2016.

[10] H. Gust, K. Kühnberger, U. Schmid, Metaphors and heuristic-driven theory projection (HDTP), Theoretical Computer Science 354 (2006) 98–117. doi:10.1016/j.tcs.2005.11.009.

[11] X. Li, Automating the repair of faulty logical theories, Doctoral Dissertation, the University of Edinburgh (2021).

[12] M. Urbonas, A. Bundy, J. Casanova, X. Li, The use of MAX-SAT for optimal choice of automated theory repairs, in: Proceedings of the 40th SGAI International Conference on Artificial Intelligence, Cambridge, UK, December 15-17, 2020, pp. 49–63. doi:10.1007/978-3-030-63799-6_4.

[13] A. Bundy, B. Mitrovic, Reformation: A domain-independent algorithm for theory repair, Technical report, the University of Edinburgh (2016).

[14] C. Cai, The application of reformation to repair faulty analogical blends, MSc Dissertation, the University of Edinburgh (2016).

[15] X. Li, A. Bundy, A. Smaill, ABC repair system for Datalog-like theories, in: Proceedings of the 10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, Seville, Spain, September 18-20, 2018, pp. 333–340. doi:10.5220/0006959703330340.

[16] J. H. Siekmann, Unification theory, Journal of Symbolic Computation 7 (1989) 207–274. doi:10.1016/S0747-7171(89)80012-4.

[17] D. Gentner, Structure-mapping: A theoretical framework for analogy, Cognitive Science 7 (1983) 155–170. doi:10.1207/s15516709cog0702_3.

[18] B. Falkenhainer, K. D. Forbus, D. Gentner, The structure-mapping engine: Algorithm and examples, Artificial Intelligence 41 (1989) 1–63. doi:10.1016/0004-3702(89)90077-5.

[19] L. M. de Moura, N. S. Bjørner, Z3: an efficient SMT solver, in: Proceedings of the 14th International Conference on Tools and Algorithms for the Construction and Analysis of Systems, Budapest, Hungary, March 29-April 6, 2008, pp. 337–340. doi:10.1007/978-3-540-78800-3_24.

[20] R. A. Kowalski, D. Kuehner, Linear resolution with selection function, Artificial Intelligence 2 (1971) 227–260. doi:10.1016/0004-3702(71)90012-9.

[21] J. Wielemaker, T. Schrijvers, M. Triska, T. Lager, SWI-prolog, Theory and Practice of Logic Programming 12 (2012) 67–96. doi:10.1017/S1471068411000494.

[22] G. Van Rossum, F. L. Drake Jr, Python Tutorial, Python Software Foundation, 2012.