Euler/X: A Toolkit for Logic-based Taxonomy Integration

Mingmin Chen[1], Shizhuo Yu[1], Nico Franz[2],
Shawn Bowers[3], and Bertram Ludäscher[1]

[1] Dept. of Computer Science, UC Davis, {michen,szyu,ludaesch}@ucdavis.edu
[2] School of Life Sciences, Arizona State University, nico.franz@asu.edu
[3] Dept. of Computer Science, Gonzaga University, bowers@gonzaga.edu

Abstract. We introduce Euler/X, a toolkit for logic-based taxonomy integration. Given two taxonomies and a set of alignment constraints between them, Euler/X provides tools for detecting, explaining, and reconciling inconsistencies; finding all possible merges between (consistent) taxonomies; and visualizing merge results. Euler/X employs a number of different underlying reasoning systems, including first-order reasoners (Prover9 and Mace4), answer set programming (DLV and Potassco), and RCC reasoners (PyRCC8). We demonstrate the features of Euler/X and provide experimental results showing its feasibility on various synthetic and real-world examples.

1 Introduction

Biological taxonomies are hierarchical representations used to specify formal classifications of organismal groups (e.g., species, genera, families, etc.) While the names used for organismal groups (i.e., *taxa*) are regulated by various *Codes* of nomenclature, it is widely recognized that names alone are not sufficiently granular to integrate taxonomic entities occuring in related classifications [10,6,2]. Thus additional information is required to relate taxonomic entities across taxonomies. These relationships can then be used to compare different taxonomies and integrate multiple taxonomies into a single hierarchical representation.

The first attempts to provide formal reasoning over taxonomies were made in the MoReTax project [1], which introduced the use of RCC-5 relations [12] for defining relationships (articulations) among taxonomic concepts. RCC-5 provides five basic relations for defining *congruence*, *proper inclusion*, *inverse proper inclusion*, *overlap*, and *exclusion* among pairs of sets or concepts. These comparative relations are intuitive to taxonomic experts who assert them and who may also express ambiguity in their assessment among concept pairs by using disjunctions of articulations: when the exact relation is unknown to the expert, she can choose disjunctions of the basic five relations, giving rise to up to 31 articulations, to capture partial knowledge. For example, A {*congruence, overlap*} B means the set A can be equivalent to or overlaps the set B. The MoReTax approach was formalized in first-order logic and implemented in CleanTax [14]. This system implemented RCC-5 reasoning using the first-order theorem provers Mace4

and Prover9 [11], but also adding three taxonomic covering assumptions—*(i) non-emptiness*, *(ii) sibling disjointness*, and *(iii) parent coverage*[4]—to achieve a working environment for taxonomic reasoning.

Here we demonstrate the Euler/X toolkit which offers a suite of interactive reasoning and visualization programs that extend the capabilities of CleanTax while improving scalability. Euler/X also adds new reasoning approaches to CleanTax including ASP (Answer Set Programming [8]) and a specialized RCC-8 reasoner [13]. The toolkit implements a comprehensive taxonomy import, merge, and visualization workflow, with new features such as (1) PostgreSQL input of the original taxonomies and expert-asserted articulations [5], (2) detection of alignment inconsistencies, (3) diagnosis of inconsistency provenance (based on provenance semirings [9]) and interactive repair, (4) alignment ambiguity reduction, and (5) visualization of merged taxonomies based on a set of inferred, *maximally informative relationships* (MIR) that reflect (6) one or multiple possible worlds scenarios for taxonomy integration. We illustrate these features using an abstract example that embodies various of the aforementioned challenges (inconsistency, ambiguity, multiple possible worlds) while maintaining close resemblance with real-life use cases [6,4].

Contributions. Euler/X encodes the input taxonomies, articulations, and constraints and feeds various inference problems to different reasoners (the "X" in Euler/X), then translates the output from those reasoners to "knowledge products" to suit user needs. The main technical contribution are the ASP and other logical encodings, the use of provenance, and result visualization, applied to real-world taxonomy integration problems. To the best of our knowledge, Euler is the first system to apply formal reasoning using ASP to such problems.

2 System Demonstration

Example. To demonstrate Euler/X, we introduce a simple example (Fig. 1) of two taxonomies T_1 (original) and T_2 (revised). Each taxonomy includes only two levels (genus and species) and ten constituent taxonomic concepts ($1.A$, $1.B$, ..., $2.A$, $2.B$, ...). Moreover there are six initial, expert-asserted articulations that connect the respective entities. Three of these include disjunctions ('or'), reflecting the expert's uncertainty as to the precise relationship among concept pairs, and one leads to an inconsistency (though the expert is not yet aware of this error). Comparable, real-life examples are provided in [4].

Workflow Overview. Euler/X will ingest the example input (Fig. 1) into PostgreSQL in the form of three simple spreadsheets: (1) a table that uniquely identifies each of the ten taxonomic concepts; (2) a table that incorporates each set of five concepts into its respective taxonomy (T_1, T_2) via *is_a* parent/child

[4] Denoting that (i) concepts/taxa are non-empty, i.e. have instances, (ii) sibling taxa are disjoint, (iii) the parent taxa is coverd by the union of child taxa, respectively.

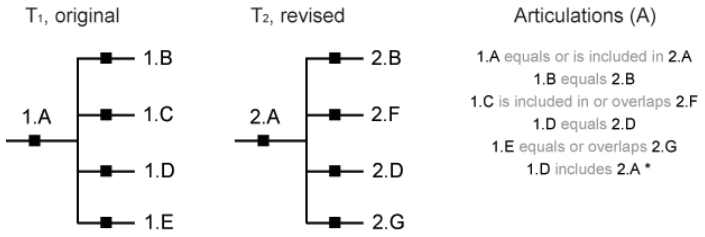

Fig. 1: Abstract example with two succeeding taxonomic classifications T_1, T_2 and a set of expert-asserted articulations (A) among taxonomic concepts. Three articulations are disjunctive; one ('*') leads to an inconsistency. T_2 (revised) builds on T_1 (original) but is a modification of T_1; it reuses T_1 entities but views and arranges them differently.

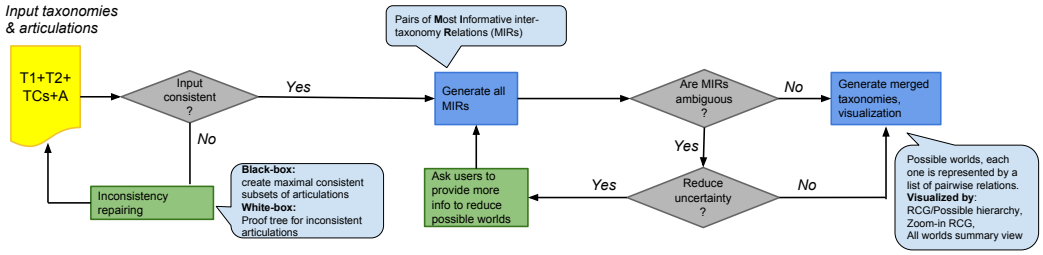

Fig. 2: EULER/X workflow overview: Input taxonomies $\mathsf{T}_1, \mathsf{T}_2$ together with expert articulations A and other taxonomic constraints TCs yield MIRs, merged taxonomies, and visualization products.

relationships (e.g., $1.B$ *is_a* $1.A$, etc.); and (3) a table with the six input articulations (A). The user also specifies a set of taxonomic constraints (TCs), e.g., *coverage*. The system then guides the user through an interactive workflow (Fig. 2) that includes the following major functions: consistency checking (including inconsistency explanation and repair), MIR generation, ambiguity representation (possible worlds[5]) and reduction, and lastly output of the merged taxonomies, including visualization and explanation of the newly inferred MIRs. Jointly, these functions enable the expert to obtain and comprehend a maximally consistent and unambiguous tabular and graphic representation of the merged taxonomy. Alternative reasoners—Prover9/Mace4 (FOL), DLV, Potassco (ASP), and PyRCC8 (RCC)—are integrated into the workflow to address specific reasoning challenges.

Architecture. As shown in Fig. 3, the EULER/X toolkit wraps six modules: persistence module, taxonomy module, articulation module, alignment module, explanation module, and reasoning module. User input will be stored in the database (persistence module) after pre-processing; the taxonomy module and

[5] In each possible world, the relation of any two taxa is one of the RCC5.

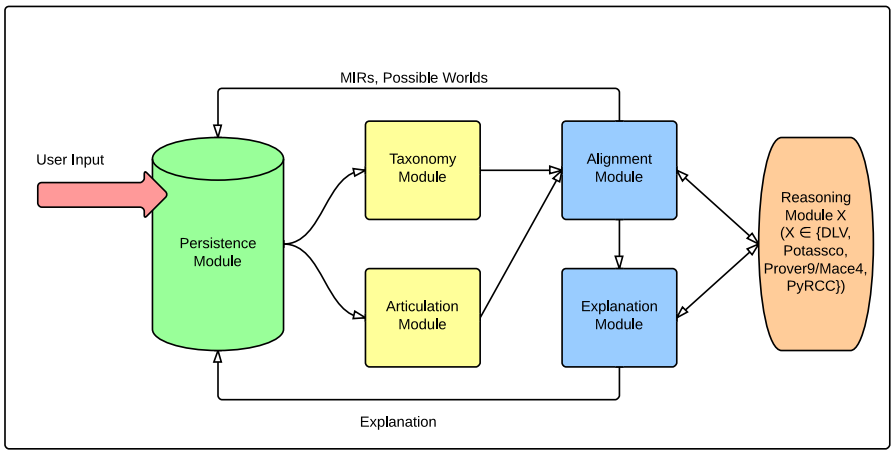

Fig. 3: EULER/X Toolkit Architecture.

articulation module load taxonomy and articulation data from the database, and pass to alignment module; alignment module then generates inputs for the reasoning module and determines the consistency and generate the possible worlds using the results from the reasoning module. In case there is inconsistency, explanation module will generate the provenance for the inconsistency based on the outputs from reasoning module. The MIRs, possible worlds, and explanation will then be passed to persistence module for storage. Reasoning module composes alternative reasoners, such as Prover9/Mace4 (FOL), DLV [3], Potassco [7] (ASP), and PyRCC8 (RCC).

Consistency Checking and Inconsistency Repair. The example (Fig. 1) is computable in EULER/X using either FOL or ASP reasoners (Fig. 2). The first processing step focuses on testing the consistency of the input alignment (A). In our use case, EULER/ASP and EULER/FO both infer that the input is inconsistent. In particular, EULER/FO provides a black-box explanation that "1.D includes 2.A" is inconsistent with the remaining articulations, and recommends removing this articulation to obtain a consistent alignment. In contrast, EULER/ASP offers a white-box explanation, stating that "1..D includes 2.A" (implying that 1.D is a high-level, inclusive taxonomic concept) is inconsistent with "1.A equals or is included in 2.A" and "1.D is_a 1.A" (jointly asserting that 1.D is a low-level, non-inclusive concept). Thus one can repair the inconsistency simply by deleting the articulation "1.D includes 2.A". Based on subsequent EULER/X reasoning (MIR), we will find that the correct 1.D/2.A articulation is "1.D is included in 2.A".

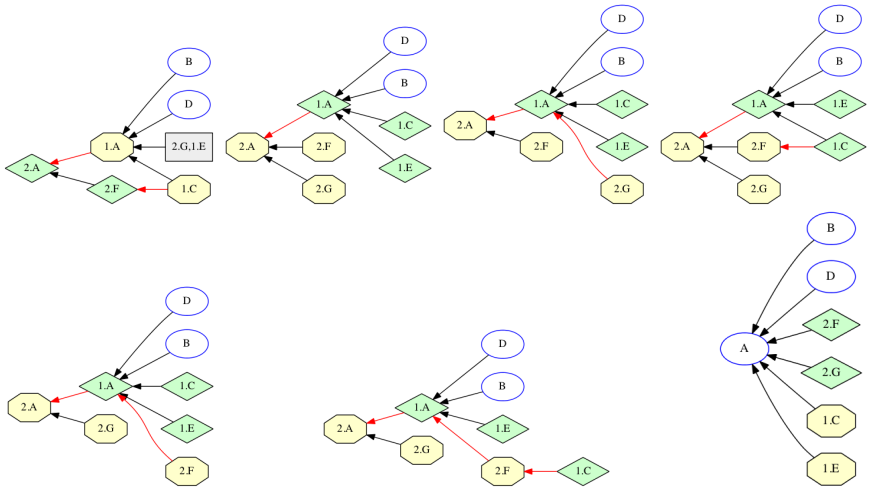

Fig. 4: Set of possible worlds $w_0, \ldots, w_6$ (i.e., concept hierarchies or RCGs) resulting from the MIR inferred by EULER/X based on the repaired input example: white ellipse (grey box) nodes show congruent (merged) concepts in both taxonomies; green diamond/yellow octagon nodes show concepts unique to each taxonomy; black edges show input *is_a* relations; and red edges show newly inferred *is_a* relations. RCGs do not represent concept overlap.

Generating MIR and Possible World Visualizations. Once the input example's inconsistency is repaired, EULER/X will proceed to generate all *maximally informative relations* (MIR; see Thau *et al.*, 2009 [14]) among taxonomic concept pairs. The interaction of the three articulations involving disjunction (Fig. 1) form an inherently ambiguous input alignment, which results in a total of seven equally consistent "possible world" solutions. These possible worlds can be displayed using a simple "Reduced Containment Graph" – a transitively reduced directed graph in which an edge represents proper inclusion. (Fig. 4).

Interactive Ambiguity Reduction. Although the seven possible worlds (Fig. 4) accurately reflect the resolving power of the input alignment (Fig. 1), the user may now have the ability and desire to reduce the inherent ambiguity by selectively eliminating certain (apparently improbable) possible worlds. This is facilitated by the EULER/X feature of ambiguity reduction. At runtime, EULER asks the user more questions (generated by a decision tree function) via pop-out

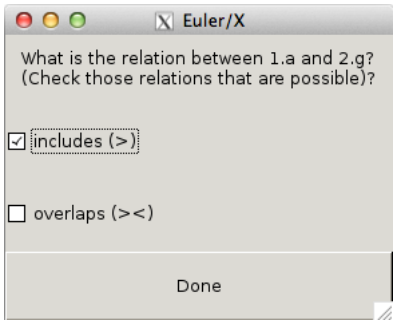

Fig. 5: Pop-out interactive window showing ambiguity reduction.

interactive windows allowing the user to select the preferred answer, e.g., by specifying that the current articulation in the query instance is "`1.A` > `2.G`", i.e., `1.A` properly includes `2.G` (Fig. 5). Based on the responses EULER/X can reduce the number of possible worlds from seven to three, filtering out four possible worlds in which `1.A` and `2.G` and overlap.

Visual clustering of similar possible worlds. We can expect some use cases with larger sized input taxonomies and multiple inherent ambiguities to yield large numbers of possible worlds. EULER/X offers a visual representation of the cumulative possible worlds "universe" via a distance matrix (Fig. 6). As shown in Fig. 4, our input example has seven possible worlds. We can compute pairwise distances among these by integrating the numbers of MIRs in which they differ and thereby generating a network that summarizes the similarities and differences.

Additional features. EULER/X also provides information on the *provenance* of a newly generated MIR relation. Moreover the toolkit can provide users with a consensus perspective of all possible worlds, i.e., specifying what is true in all of them, or how often a particular MIR occurs across all possible worlds.

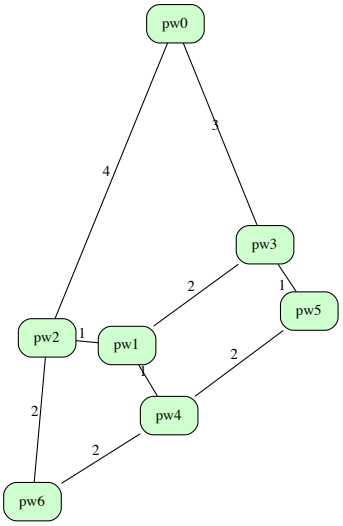

Fig. 6: Distance matrix-based visualization of the seven possible worlds of the input example. See also Fig. 4. The absolute distance between two possible worlds is the shortest distance traceable in the graph; e.g. the distance between possible worlds 5 and 6 is 4 steps.

3 Performance Results

We tested the performance and scalability of different reasoning approaches, including EULER/FO (Prover9/Mace4), EULER/ASP (DLV and Potassco), and EULER/PyRCC (PyRCC8). Tests used both real-life and simulated examples as well as performed both consistency checks and MIR and possible worlds computation. The running time was measured using increasingly larger input datasets. All examples were tested on an 8-core, 32GB-memory Linux server.

While EULER/FO checks consistency by calling Mace4 once and then generates each MIR by calling Prover9[6] (for $m * n$ MIR's assuming there are m, n entities in each taxonomy), the other EULER tools only invoke the reasoner once

[6] To get a MIR, Prover9 is called to answer "yes" or "no" to the five base relation questions.

to check consistency and merge taxonomies (MIR and possible world generation). This is why EULER/FO is faster for consistency checking (specifically, EULER/FO is slower than EULER/ASP (Potassco) when the number of entities in each taxonomy is less than 100, but faster when it is more than 100), but very slow in MIR generation as shown in Fig. 7. For taxonomy merge, PyRCC8 is faster than Potassco, Potassco is faster than DLV, and DLV is much faster than our FO-based approach. However, note that EULER/PyRCC is not capable of applying the same merge as the other tools since the coverage constraints cannot be asserted using RCC-5. When considering all three taxonomic constraints, the Potassco-based EULER is the fastest and reasonably good overall, since it can perform taxonomy merge for realistic taxonomies of 100 entities in half a minute.

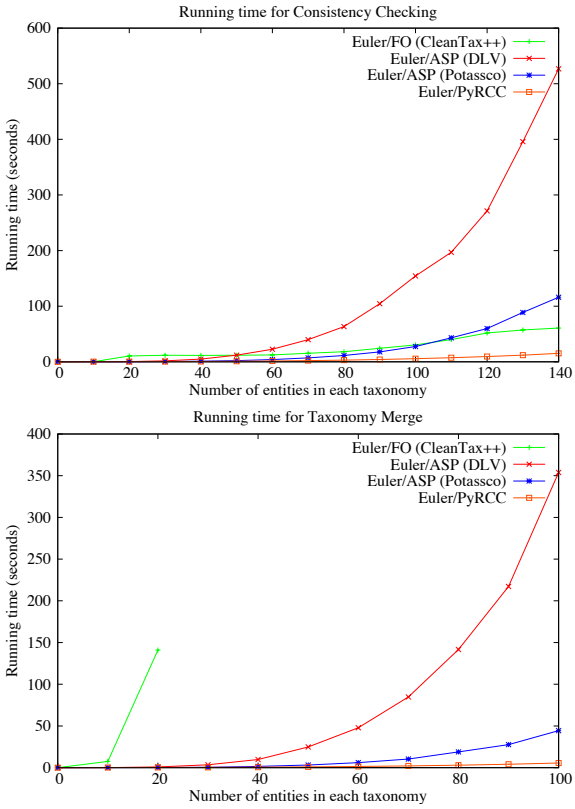

Fig. 7: Running times for consistency checking (left) and taxonomy merge (right) on synthetic taxonomies (balanced taxonomy trees of depth 8 with "is included in" articulations, resulting in a single possible world).

4 Conclusions and Future Directions

EULER/X is open source and can be downloaded from BitBucket[7]. Planned future developments include: (1) support for incremental changes to alignments; (2) an improved ASP-based tool, using the results from PyRCC8; (3) development of a user-friendly GUI; and (4) further exploration of other reasoners, e.g., those developed for OWL.

Acknowledgements. We thank the anonymous reviewers for their helpful comments. Work supported in part by NSF awards IIS-1118088 and DBI-1147273.

[7] https://bitbucket.org/eulerx/euler-project

References

1. W. G. Berendsohn. *MoReTax: Handling Factual Information Linked to Taxonomic Concepts in Biology*. Schriftenreihe für Vegetationskunde 39:1-113, 2003.
2. B. Boyle, N. Hopkins, Z. Lu, J. A. R. Garay, D. Mozzherin, T. Rees, N. Matasci, M. L. Narro, W. H. Piel, S. J. Mckay, et al. The taxonomic name resolution service: an online tool for automated standardization of plant names. *BMC Bioinformatics*, 14:16, 2013.
3. S. Citrigno, T. Eiter, W. Faber, G. Gottlob, C. Koch, N. Leone, C. Mateis, G. Pfeifer, and F. Scarcello. The dlv system: Model generator and application frontends. In *Proceedings of the 12th Workshop on Logic Programming*, pages 128–137, 1997.
4. N. M. Franz and J. Cardona-Duque. Description of two new species and phylogenetic reassessment of *Perelleschus* Wibmer & O'Brien, 1986 (Coleoptera: Curculionidae), with a complete taxonomic concept history of *Perelleschus* sec, 2013.
5. N. M. Franz and R. K. Peet. Towards a language for mapping relationships among taxonomic concepts. *Systematics and Biodiversity*, 7(1):5–20, 2009.
6. N. M. Franz, R. K. Peet, and A. S. Weakley. On the use of taxonomic concepts in support of biodiversity research and taxonomy. *Systematics Association Special Volume*, 76:63, 2008.
7. M. Gebser, B. Kaufmann, R. Kaminski, M. Ostrowski, T. Schaub, and M. Schneider. Potassco: The potsdam answer set solving collection. *AI Communications*, 24(2):107–124, 2011.
8. M. Gelfond. Answer sets. In F. van Harmelen, V. Lifschitz, and B. Porter, editors, *Handbook of Knowledge Representation*, pages 285–316. Elsevier, 2008.
9. T. J. Green, G. Karvounarakis, and V. Tannen. Provenance semirings. In *PODS*, 2007.
10. J. Kennedy, R. Kukla, and T. Paterson. Scientific names are ambiguous as identifiers for biological taxa: Their context and definition are required for accurate data integration. In *Data Integration in the Life Sciences (DILS)*, LNCS 3615, 2005.
11. W. McCune. Prover9 and Mace4, 2005–2010. www.cs.unm.edu/~mccune/prover9.
12. D. A. Randell, Z. Cui, and A. G. Cohn. A spatial logic based on regions and connection. In *Knowledge Representation and Reasoning (KR)*, 1992.
13. M. Sioutis. A RCC8 based qualitative spatial reasoner written in pure python. pypi.python.org/pypi/PyRCC8, 2012.
14. D. Thau, S. Bowers, and B. Ludäscher. Merging Sets of Taxonomically Organized Data Using Concept Mappings under Uncertainty. In *Ontologies, DataBases, and Applications of Semantics*, LNCS 5871, 2009.