

Bridging BDI Multi-Agent Systems and the Semantic Web Through the *Triples-to-Beliefs-to-Triples* Paradigm

Carmelo Fabio Longo¹^a, Rocco Paolillo¹^b, Misael Mongiovi²^c, Andrea Giovanni Nuzzolese¹^d,
Francesco Poggi¹^e, Michele Geremia Ceriani¹^f, Antonio Zinilli¹^g, Giusy Giulia Tuccari¹^h
and Corrado Santoro²ⁱ

¹National Research Council, Italy

²Department of Mathematic and Informatics, University of Catania, Italy

Keywords: Artificial Intelligence, BDI Agents, Semantic Web, Multi-Agent System.

Abstract: Well-established agent engineering frameworks from the state-of-the-art, due to their outdated designs, are not thought to work in the perspective of a shared semantics, nor do they provide an agent modeling language and environment that integrates seamlessly with them. This is especially challenging in dynamic, distributed environments where new concepts, data sources, and agents can emerge at runtime, potentially leading to semantic conflicts or inconsistencies. This paper proposes the novel paradigm *Triples-to-Beliefs-to-Triples* (T2B2T), which is being ontologically described, enabling multi-agent systems with seamless and consistent integration with the Semantic Web. In order to validate the approach, this paper proposes also a framework called SEMAS implementing the T2B2T paradigm, which provides a bridge between the *mental attitudes* Beliefs-Desire-Intentions (BDI) and triples describing a domain with an abstraction over the SPARQL language that feeds the inference process of agents. This enables more sophisticated forms of reasoning in the closed-world assumption, by supporting predicates without any limitation on arity and compositional structures, allowing also the employment of decentralized functions for the dynamic generation of new triples not included in the origin ontologies. As a case-study, SEMAS was employed on decision-making applied to academic mobility with real data coming from the SCOPUS database, demonstrating how the generated inferences can be tailored to specific conditions of individual agents, and how new triples can be inferred to capture the impact of agents' decisions on the evolution of the knowledge domain.

1 INTRODUCTION

A seamless integration with the Semantic Web plays a fundamental role in Multi-Agent Systems (MAS), whether they are virtual (as in social simulations), physical, or both (digital twins). MAS and the Semantic Web have traditionally evolved independently, each addressing distinct challenges within the realm

of Artificial Intelligence and Linked Open Data. The former focuses on the design and coordination of autonomous agents capable of decision-making, problem-solving, and interacting with other agents or the environment. The Semantic Web envision (Berners-Lee et al., 2001), on the other hand, emphasizes enriching web data with meaning, allowing machines to interpret and process information in a more intelligent, context-aware manner. In such a scope, the following research question arises: how to enable dynamic and semantically consistent interoperability between autonomous agents in heterogeneous Semantic Web ecosystems? More specifically, the integration of MAS with the Semantic Web faces the challenge of ensuring that autonomous agents, each referring to potentially distinct ontologies, vocabularies, and reasoning mechanisms, can dynamically interact and exchange knowledge in a meaningful and seman-

^a <https://orcid.org/0000-0002-2536-8659>

^b <https://orcid.org/0000-0001-9816-5839>

^c <https://orcid.org/0000-0003-0528-5490>

^d <https://orcid.org/0000-0003-2928-9496>

^e <https://orcid.org/0000-0001-6577-5606>

^f <https://orcid.org/0000-0002-5074-2112>

^g <https://orcid.org/0000-0001-8505-5040>

^h <https://orcid.org/0009-0008-3298-7168>

ⁱ <https://orcid.org/0000-0003-1780-5406>

tically consistent way. This is particularly difficult in open, distributed, and constantly evolving environments where new concepts, data sources, and agents may appear at runtime, potentially causing semantic conflicts or inconsistencies. Assuring scalable, dynamic, and reliable semantic interoperability in such contexts remains an open research problem.

This paper addresses the above challenge and proposes a novel paradigm for integrating MAS and the Semantic Web, called *Triples-to-Beliefs-to-Triples* (T2B2T), which is ontologically described in this paper. In this paradigm, RDF triples from a triple store are processed and turned into beliefs, by populating agents' Knowledge Base (KB) and interacting with *goal oriented* production rules systems to infer newly introduced beliefs. Such inferred beliefs will be possibly translated back into RDF triples to either update the origin triple store or build *new brand* derived Knowledge Graphs (KG). This cycle facilitates the seamless communication between agents and the semantic data layer, ensuring that agents operate based on a shared, evolving KB. The gain in such a two-way translation of the T2B2T paradigm lies in the chance of more sophisticated inferences involving computations out of SPARQL and OWL-based reasoners, shifting from open- to closed-world assumption. Moreover, this approach ensures results consistency by coordinating inferences of multiple agents in a deterministic way, relying on well-established and shared criteria of interoperability.

The contribution of our work is twofold: the formalization of the T2B2T paradigm through an ontology, and its implementation with the SEMAS BDI framework. The paper is organized as follows: Section 2 provides an ontology modeling the T2B2T paradigm; Section 3 delves into the framework modules; Section 4 provides an overview of the current state-of-the-art in the topic; Section 5 offers a comprehensive overview of a case-study based on the academic mobility; finally, Section 6 concludes the paper with some final considerations. The code of SEMAS is publicly available for research purposes through a dedicated GitHub repository¹.

2 THE ONTOLOGY

This Section shows how we ontologically modeled the here-proposed T2B2T paradigm, whose simplified functional schema is depicted in Figure 1. It is important to clarify that such ontology must be distinguished from external KGs we want agents to in-

teract with seamlessly. T2B2T could support in case agents need to make inference on KGs lacking required triples, through external functions outside of SPARQL, e.g. outcomes from classifiers, large language models, or even aggregated outcomes as in social simulation. The pipeline *Triple-to-Beliefs* (T2B) of T2B2T, first computes triples to produce beliefs interacting with symbolic axioms. Beliefs can either have symbolic notation or can be references to sub-symbolic information. In this configuration, we use Prolog-like engines to combine predicates with arity greater than two, overcoming the limitations of the SWRL (World Wide Web Consortium, 2004) language, and by leveraging on inference criteria based on the backward-chaining algorithm rather than the forward-chaining of OWL-based reasoners like Pellet (Sirin et al., 2007) or Hermit (Glimm et al., 2014).

After inference with axioms, inferred beliefs can be translated into triples and fed either the starting KG or newly introduced KGs, which is the *Beliefs-to-Triples* (B2T) pipeline.

Inference must also be supported by an event queue, in order to coordinate MAS and let them produce consistent/deterministic results in case of inter-related inferences, i.e., when an agent's inference (in terms of new triples) can affect inferences of other agents and subsequently their behavior.

We follow Rao et al. (Rao and Georgeff, 1995) framework of Belief-Desires-Intentions (BDI), building on *plans*, which are abstract specifications representing both the means for achieving certain desires and the options available to the agent. *Desires* are implicitly described by plans aimed at achieving the well-defined *goal*. Moreover, each *intention* that the system forms by adopting certain plans of action is represented implicitly by using a conventional runtime stack of hierarchically related plans, which is the reason we omit to report them in the ontology. Here's some details of classes and properties modeling the T2B2T paradigm (cf. Figure 2):

- **Agent.** Instances of this class represent a single agent aimed to make inference on KGs.
- **Belief.** Instances of this class are referenced by the object-property *hasBelief*, and refer to a piece of information that an agent considers to be true about the environment. Information can be (but not limited to) a predicate with arbitrary arity and compositional structures, and even a sub-symbolic representation in a vectorial space. Under the open-world assumption (which underpins the Semantic Web) missing information is treated as *unknown*. In contrast, the closed-world assumption interprets unknown information as *false*. A transition from the open- to

¹<https://github.com/cfabiolongo/Semas>

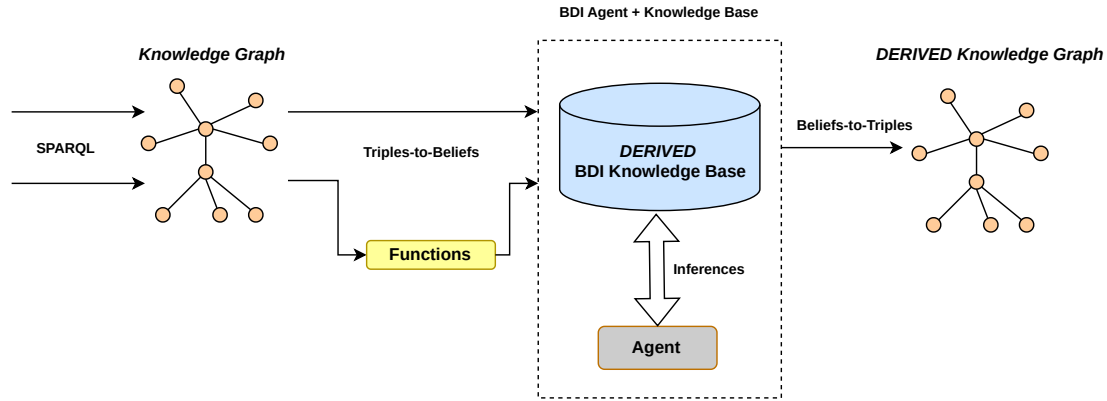


Figure 1: The simplified functional schema of the process behind the T2B2T paradigm.

the closed-world assumption is valid if and only if an equivalence is assumed between the predicates *Retrieved*(X)/*Non-Retrieved*(X) with *Asserted*(Y)/*Non-Asserted*(Y)², where X denotes a triple extracted from the Semantic Web and Y represents its corresponding belief. This equivalence holds under the condition that the mapping between X and Y is semantically coherent.

- **Action.** Instances of this class represent a primitive action or sub-goal that has to be achieved for defined plans execution to be successful, and they are referenced by the object-property *hasAction*.
- **Plan.** Instances of this class represent a set of intentions which link together triggering conditions (beliefs) to instance of *Action* aimed at achieving goals, and they are referenced by the object-property *hasPlan*.
- **Goal.** Instances of this class represents a desired state or outcome that agent aims to achieve. Goals drive agent's decision-making process and influence its intentions, and they are referenced by the object-property *hasGoal*.
- **Triple.** Instances of this class represent triples from KGs, whose properties, subjects and objects are referenced by the data-properties *hasProperty*, *hasSubject*, *hasObject*, respectively. The values of these data-properties are URIs pointing to Web resources, whose textual contents are to be extracted and utilized for the formulation of beliefs within the system.
- **Function.** Instances of this class represent functions computing instances of *DerivedBelief* (cf.

below), having (or not) in input beliefs translated from triples of the origin KG, and they are referenced by the object-property *hasFunction*.

- **Enricher.** This is a subclass of *Action*, being part of a specific plan aimed to enriching agents KB with beliefs computed by functions referenced by the object-property *hasFunction*. Among other *Action*'s instances related to *Goal*, before any inference by instances of *Agent*, an instance of the class *Enricher* represents the action of populating the KB with additional beliefs *derived* from triples not present in the origin KGs.
- **DerivedBelief.** This is a subclass of *Belief*, whose instances represent a new belief not directly related to any triple present in the origin KG, but a computation of triples by leveraging on instances of *Function*, referenced by the object-property *hasFunction* of *Enricher*. The latter carry out the task of asserting *derived beliefs* in the agent's KB, which is referenced by the object-property *assertBelief*.
- **Event.** Each instance of the class *Event* aims to model whatever kind of action aimed to change agent's KB content, and it is referenced by the object-property *hasEvent*.
- **Queue.** Instances of this class represent a queue to coordinate parallel agent's interactions with shared semantic resources, in multi-agent setting, in order to achieve deterministic outcomes. It is referenced by the object-property *isQueuedOn* of instances of the class *Event*.
- **Query.** Instances of this class are SPARQL queries that will feed agent's KB, after triples-to-beliefs translation, and they are referenced by the object-property *hasQuery*.
- **AgentType.** Instances of this class represent entities whose behavior is simulated within the frame-

²The *Non-Asserted*(Y) predicate is also subsumed by *Retracted*(Y), which indicates a belief Y that has been removed from the KB as a result of an inference, thus no longer present.

work where inference takes place, and they are referenced by the object-property *hasAgentType*.

- **RESTful.** Instances of this class represent remote RESTful services, and they are referenced by the object-property *hasRESTful*. Such class play an important role, as it provides shared-decentralize interfaces computing functions aimed to produce instances of *DerivedBelief*.
- **Code.** Instances of this class represent code fragment to compute locally new *DerivedBelief*, and they are referenced by the object-property *hasCode*.

3 THE SEMAS FRAMEWORK

The SEMAS framework presented in this paper implementing the T2B2T paradigm, whose acronym stands for *SEmantic Multi-Agent System*, is built on top of the BDI architecture Phidias (D'Urso et al., 2019), which enables programs with the ability to perform logic-based reasoning (in Prolog-style). By following the T2B2T paradigm in Figure 1, triples retrieved from a KG via a SPARQL query are being turned into beliefs within the SEMAS KB, by interacting with its inference system made of production rules as follows:

```
[BELIEF] / [CONDS] >> [RULE_PLAN],
```

where the [BELIEF] placeholder refers to runtime events asserting a belief in the agent KB, which triggers the [RULE_PLAN] execution in case the content of [CONDS] is satisfied, i.e., the presence of other beliefs in the KB; [RULE_PLAN] contains a list of *actions* which implicitly implement the designed *goal*, where each action can either invoke functions in high level language or assert/retract beliefs. At any time, the whole content of the KB can be newly translated (with specific built-in functions) in triples, either to update the origin KG or to build locally novel *derived* KGs. The SEMAS framework enables efficient population of its KB according to the specific use case, also by feeding agents through sequential contextual-interrelated³ SPARQL queries (cf. Section 5), by preventing overpopulation with triples that are irrelevant to the inference process. As a result, it becomes possible to populate KBs directly derived from virtual *goal-focused* KGs of arbitrary size, by overcoming the current limitations of SPARQL in handling views⁴, which is a feature typically associated with re-

³Where results from one or more query are used as parameters to subsequent query.

⁴A view in a relational database is a virtual table defined by a saved SQL query: it doesn't store data but dynamically shows results from one or more tables each time it's queried.

lational databases. The mapping between RDF properties and beliefs is internally declared into the SEMAS's configuration, whereas each *desire* is mapped into the so-defined *procedure*, which can be used to trigger manually part of the production rules stack, taking in account (or not) of one or more arguments. Furthermore, purely reactive events linkable to intentions can be mapped into the so-called *reactors*, i.e. beliefs that after assertion do not remain resident in the KB, but interacting with the inference system as well as beliefs.

4 RELATED WORKS

A few attempts have been made to integrate MAS with ontologies. The authors of OASIS (Cantone et al., 2019; Cantone et al., 2022) proposed an OWL-based agent model language, endowed with fine-grained descriptions of agents' behaviors. However, their approach requires the executive grounding to be delegated to other frameworks. In contrast, in this paper we address both the semantic agents modeling and the reactive reasoning on the Semantic Web. The authors of SW-CASPAR (Longo et al., 2022) propose a BDI framework based on Natural Language Processing (NLP), capable of meta-reasoning in the Semantic Web. Although such a framework can be considered a move towards interoperability among NLP-based BDI agents, it does not provide templates to assist engineers in implementing multi-agent coordination protocols and is not compliant with the well-known FIPA⁵ agents interoperability guidelines. The authors of AJAN (Antakli et al., 2023) propose a modular framework for building Semantic Web-enabled intelligent agents, using Semantic Web standards and Behavior Tree technology. It supports SPARQL-extended behavior trees for agent scripting, multi-agent coordination, and is extensible with additional modules and communication layers. The advantage of using behavior trees over production rule systems (which is the core of the SEMAS inference) has not yet been documented. On one hand, production rule systems have historically been a foundational approach to artificial intelligence, such as in the General Problem Solver (GPS) (Newell and Simon, 1961); on the other hand, behavior trees are primarily designed for robotic applications (Ögren and Sprague, 2022), but in other domains, they may introduce significant overhead as their complexity increases. A special case of MAS is social simulation, where agents represent autonomous virtual entities, rather than physical ones as

⁵<http://www.fipa.org/>

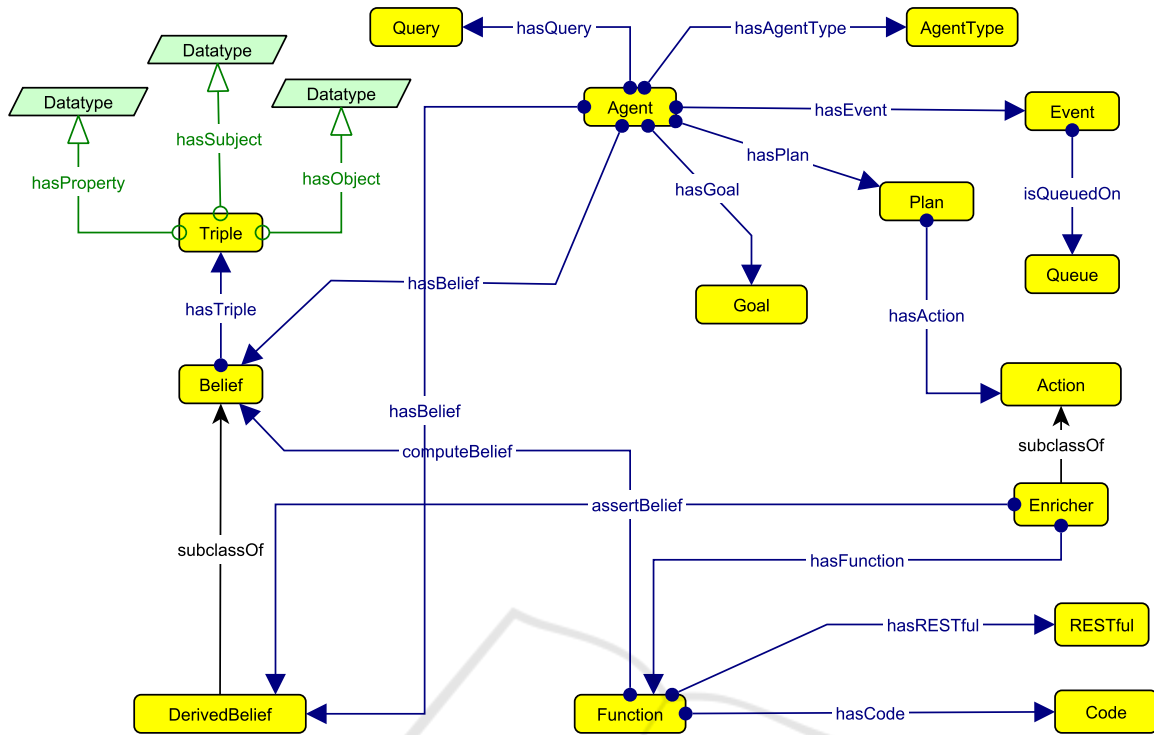


Figure 2: The ontology modeling the T2B2T paradigm.

in FIPA, capable of processing information and interacting with their surrounding environment, by replicating collective behavior and changes in the system due to the decision of individual agents. The application of T2B2T in Section 5 aligns with this viewpoint. Farrenkopf and colleagues used ontologies to model agents' cognitive architecture through BDI and their communication in a model of business decisions (Farrenkopf et al., 2016). They classify between an Individual Domain Layer (IDL) of knowledge of local agents, a Specific Domain Layer (SDL) on the market sector, and an Abstract Domain Layer (ADL) of knowledge shared by all agents. The authors focus on updating of the last two layers through direct communication between agents. Our contribution with the implementation of T2B2T goes beyond these studies, using BDI to formalize the cognitive architecture of agents but also modeling the processes of elaboration of inferences from semantic ontologies and associated effects through update of shared knowledge domain. Fostering the interoperability and integration between ontologies and MAS, we show how to interrogate a remote graph empirically grounded through a remote SPARQL. In the case-study in Section 5, we apply T2B2T and SEMAS to a scenario of academic career decision making, empirically grounded in the Italian

landscape.⁶

5 CASE-STUDY

As case-study, we first initialized a KG about Italian scholars in 2024, extracting raw data from the SCOPUS database through its Search API, including authors' name, affiliations, and topic of articles. Data were elaborated to identify co-authorships through shared publications and top-authors. Co-authorship is counted as one, regardless of the number of shared publications. A scholar is considered a top-author if published at least 10 papers and have a *h-index* equal or higher than 10. Data collected were translated into RDF triples and stored in a GRAPHDB⁷ triple store to communicate with SEMAS via URIs, resulting in a KG containing up to 100.000 triples. The beliefs obtained from co-authorships and top-authorships can be ascribed to instances of the class *DerivedBelief* in Figure 2. In Figure 3, we show the SEMAS production rules applied to the case, where rules are grouped in three distinct stages: Acquiring Triples Stage (ATS), Inference Stage (IS), and Updat-

⁶The repository for this extract is available at <https://github.com/RoccoPaolillo/Semas/tree/webist>

⁷<https://graphdb.ontotext.com/>

```

1 # Acquiring Triples Stage (ATS)
2 BeTopAuthorship(X) >> [show_line('Planning to be top-author in ', X, '...'),
3 load_obj('acad:isTopAuthorIn', X), FindRelated(), Publicationship(X)]
4
5 FindRelated() / ConsiderTopAuthor(X,Y) >> [-ConsiderTopAuthor(X, Y), +TopAuthorship(X,Y),
6 show_line('Finding triples related with ', X, '...'), load_subj('acad:hasAffiliationWith', X),
7 load_subj('acad:coAuthorWith', X), load_obj('acad:coAuthorWith', X), FindRelated()]
8 FindRelated() >> [show_line('Related triples retrieved.')]
9
10 # Inference Stage (IS)
11 Publicationship(X) / (CoAuthorship(Z,Y) & TopAuthorship(Y,X) & Affiliation(Z,U) & Selectionship(S,U)
12 & Affiliation(S,T)) >> [-CoAuthorship(Z,Y), +ProposeCoauthorship(Z,U,Y,X,S,T), Publicationship(X)]
13
14 # Updating Triples Stage (UTS)
15 +ProposeCoauthorship(Z,U,Y,X,S,T) >> [show_line(Z, ' at Organization ', U, ' is co-author with ', Y,
16 ' top-author in the topic ', X), -Affiliation(S,T), -Selectionship(S,U), +Affiliation(S,U),
17 DeleteAlternative(S)]
18 DeleteAlternative(S) / (Selectionship(S,P)) >> [-Selectionship(S,P), DeleteAlternative(S)]

```

Figure 3: The SEMAS production rules applied to SCOPUS database.

```

1 # assertion beliefs for the hypothetical scenario, selection of identified scholar S by two
   organizations S and P, here reported with the corresponding URIs
2 eShell: main >
3 +Selectionship('BASE-URI/authors/57201117401', 'BASE-URI/organizations/60000481')
4 +Selectionship('BASE-URI/authors/57201117401', 'BASE-URI/organizations/105937250')
5
6 # knowledge graph showing the affiliation of scholar S and selectionship by research organizations
7 eShell: main > kb
8 Affiliation('BASE-URI/authors/57201117401', 'BASE-URI/organizations/60024690')
9 Selectionship('BASE-URI/authors/57201117401', 'BASE-URI/organizations/60000481')
10 Selectionship('BASE-URI/authors/57201117401', 'BASE-URI/organizations/105937250')
11
12 # The desire BeTopAuthorship() in a specific topic is invoked, which enacts ACS, IS and UTS stages.
13 eShell: main > BeTopAuthorship('BASE-URI/topics/2003')
14
15 # Results of IS stage reporting co-authors to top-authors found in the two organizations.
16 # In this case only one organization matches the conditions required, so to be chosen by scholar S
17 BASE-URI/authors/16021198600 at Organization BASE-URI/organizations/60000481 is co-author
18 with BASE-URI/authors/55904339600 top-author in the topic BASE-URI/topics/2003
19
20 BASE-URI/organizations/60000481 is co-author with BASE-URI/authors/48061340500 top-author in the topic
21 BASE-URI/topics/2003
22
23 # knowledge graph showing the new affiliation of scholar S after UTS stage
24 eShell: main > kb
25 Affiliation('BASE-URI/authors/57201117401', 'BASE-URI/organizations/60000481'),...

```

Figure 4: Initialized case study with SCOPUS data and running of SEMAS.

ing Triples Stage (UTS). Figure 4 shows the initialization and running of SEMAS with SCOPUS data. To ensure consistency and determinism each group can be run accordingly to the stage; afterward, the SEMAS' built-in event queue (inherited from Phidias) will do implicitly the rest of the work, taking into account of stage's priorities of other agents through messages exchange.

```

SELECT ?subj ?prop ?obj WHERE {
  <[SUBJ]> [PROP] ?obj .
  BIND([PROP] AS ?prop)
  BIND(<[SUBJ]> AS ?subj)
}

```

Listing 1: The parameterized SPARQL query executed by the action load_obj([PROP], [SUBJ]), with property [PROP] and subject [SUBJ] as placeholders.

The production rules leverage on the objects *procedures* and *reactors* introduced in Section 3, to implement the cognitive qualities of agents' mental attitudes. We build on the recognized role of co-authoring with top-authors and networking as a driver to academic career (Letina, 2016).

In line 2 of Figure 3, the procedure `BeTopAuthorship(X)` first starts the pipeline *triples-to-beliefs* (T2B) of T2B2T with the action `load_obj('acad:isTopAuthorIn', X)`, which executes a parameterized SPARQL query (cf. Listing 1)⁸ to filter all triples' subjects having `acad:isTopAuthorIn` as property and topic `X` as object, that are used to assert the symbolic belief `ConsiderTopAuthor`, which is a special belief used to avoid an infinite loop in line 5. In this line, the procedure `FindRelated()` executes more SPARQL queries through the action `load_subj('acad:hasAffiliationWith', X)`, which filters all triples having `hasAffiliationWith` as property and author `X` as subject, in order to assert beliefs with `Affiliation` as label; similarly, the actions `load_subj('acad:coAuthorWith', X)` and `load_obj('acad:coAuthorWith', X)` aim to extract triples with property `coAuthorWith`, where author `X` is either subject or object of the triple, both to assert beliefs with `CoAuthorship` as label. Likewise to `load_obj`, `load_subj(X, Y)` executes parameterized SPARQL queries (cf. Listing 2) to extract all triples having `X` as property and `Y` as subject.

```
SELECT ?subj ?prop ?obj WHERE {
  ?subj [PROP] <[OBJ]> .
  BIND([PROP] AS ?prop)
  BIND(<[OBJ]> AS ?obj)
}
```

Listing 2: The parameterized SPARQL query executed by the action `load_subj([PROP], [OBJ])`, with property `[PROP]` and object `[OBJ]` as placeholders.

The pipeline thus composes the KG used for the inferences upon beliefs assertions. Line 11 shows the actual reactive plan in the inference stage (IS) activated by the procedure `Publicationship()` with top-authors, to which a set of condition beliefs is applied. In the scenario, a scholar `S` who is already affiliated with the university `T`, is offered a position by universities `U` and `P` with neither of them hosting a top-author in the elective field. Applying a strategy based on small world networks (Koseoglu, 2016), the inference identifies the orga-

nization `U` with co-authors `Z` to a top-author `Y` they could connect `S` with in case the offer is accepted. The proposed *intention* is implemented by the reactor `ProposeCoauthorship()` (line 15), which implies accepting the offer by the organization matching the conditions. This translates into the retraction of the belief of scholar `S` selected by the chosen organization `U` (`-Selectionship(S,U)`), and retraction of the previous affiliation of `S` with the organization `T` (`-Affiliation(S,T)`), updating the assertion belief to the new affiliation (`+Affiliation(S,U)`). Also the information of `S` being selected by the alternative organization `P` is eliminated with the retraction of `-Selectionship(S,P)` activated by the procedure `DeleteAlternative()`. In Figure 4, `authors/57201117401`⁹ is selected from SCOPUS database as scholar `S`, with lines 8 to 10 showing their real affiliation to an Italian university and the two alternative institutions (equivalent to `U,P` in Figure 3) we selected. Neither of them hosts a top-author in the elective field `topics/2003` (finance), but `organizations/60000481` hosts coauthors. Line 17 shows the inference produced by SEMAS based on scanning the SCOPUS database, while line 25 shows the new hypothetical affiliation of the scholar based out of the inference.

6 CONCLUSIONS

In this paper, we addressed the challenge of enabling dynamic and semantically consistent interoperability between Multi-Agent Systems (MAS) and heterogeneous Semantic Web environments. We proposed the T2B2T paradigm as a novel integration model, aimed at overcoming the limitations of traditional MAS frameworks in processing and interacting with semantic data. By ontologically formalizing the T2B2T cycle and implementing it within the SEMAS framework, we demonstrated how agents can reason upon RDF KGs, update their internal beliefs accordingly, and propagate new inferences back into the Semantic Web ecosystem. In our work we showed the soundness of SEMAS by feeding its KB with empirical data from SCOPUS database, by adding new beliefs computed from existing triples, which take part in the inference process to produce newly introduced triples. The proposed paradigm goes beyond conventional SPARQL-based querying by allowing agents to perform richer symbolic inference, while ensuring consistency and determinism across distributed

⁸The PREFIX declarations in the query have been intentionally omitted.

⁹We report URIs and not actual name to guarantee anonymity

agents. This approach also provides a practical mechanism for moving between open- and closed-world assumptions depending on the context of reasoning, thereby enhancing agents' cognitive capabilities in dynamic and evolving environments.

ACKNOWLEDGEMENTS

This work was supported by FOSSR (Fostering Open Science in Social Science Research), funded by the European Union - NextGenerationEU under NRRP Grant agreement n. MUR IR0000008.

REFERENCES

- Antakli, A., Kazimov, A., Spieldenner, D., Rojas, G. E. J., Zinnikus, I., and Klusch, M. (2023). Ajan: An engineering framework for semantic web-enabled agents and multi-agent systems. In Mathieu, P., Dignum, F., Novais, P., and De la Prieta, F., editors, *Advances in Practical Applications of Agents, Multi-Agent Systems, and Cognitive Mimetics. The PAAMS Collection*, pages 15–27, Cham. Springer Nature Switzerland.
- Berners-Lee, T., Hendler, J., and Lassila, O. (2001). Web semantic. *Scientific American*, 284(5):34–43.
- Cantone, D., Longo, C. F., Nicolosi-Asmundo, M., Santamaria, D., and Santoro, C. (2019). Towards an Ontology-Based Framework for a Behavior-Oriented Integration of the IoT. In *Proceedings of the 20th Workshop From Objects to Agents, 26-28 June, 2019, Parma, Italy, CEUR Workshop Proceeding Vol. 2404*, pages 119–126.
- Cantone, D., Longo, C. F., Nicolosi Asmundo, M., Santamaria, D. F., and Santoro, C. (2022). Ontological smart contracts in oasis: Ontology for agents, systems, and integration of services. In Camacho, D., Rosaci, D., Sarné, G. M. L., and Versaci, M., editors, *Intelligent Distributed Computing XIV*, pages 237–247, Cham. Springer International Publishing.
- D'Urso, F., Longo, C. F., and Santoro, C. (2019). Programming intelligent iot systems with a python-based declarative tool. In *The Workshops of the 18th International Conference of the Italian Association for Artificial Intelligence*.
- Farrenkopf, T., Guckert, M., Urquhart, N., and Wells, S. (2016). Ontology based business simulations. *Journal of Artificial Societies and Social Simulation*, 19(4).
- Glimm, B., Horrocks, I., Motik, B., Stoilos, G., and Wang, Z. (2014). HermiT: An OWL 2 Reasoner. *Journal of Automated Reasoning*, 53(3):245–269.
- Koseoglu, M. A. (2016). Growth and structure of authorship and co-authorship network in the strategic management realm: Evidence from the strategic management journal. *BRQ Business Research Quarterly*, 19(3):153–170.
- Letina, S. (2016). Network and actor attribute effects on the performance of researchers in two fields of social science in a small peripheral community. *Journal of Informetrics*, 10(2):571–595.
- Longo, C. F., Santoro, C., Nicolosi-Asmundo, M., Cantone, D., and Santamaria, D. F. (2022). Towards ontological interoperability of cognitive iot agents based on natural language processing. *Intelligenza Artificiale*, 16(1):93–112.
- Newell, A. and Simon, H. A. (1961). Gps, a program that simulates human thought. In Billing, H., editor, *Lernen und automatische Informationsverarbeitung*, pages 109–124. Springer.
- Rao, A. and Georgeff, M. (1995). BDI agents: From theory to practice. In *Proceedings of the first international conference on multi-agent systems (ICMAS-95)*, pages 312–319. San Francisco, CA.
- Sirin, E., Parsia, B., Grau, B. C., Kalyanpur, A., and Katz, Y. (2007). Pellet: A practical OWL-DL reasoner. *Web Semantics*, 5(2):51–53.
- World Wide Web Consortium (2004). SWRL: A Semantic Web Rule Language Combining OWL and RuleML.
- Ögren, P. and Sprague, C. I. (2022). Behavior trees in robot control systems. *Annual Review of Control, Robotics, and Autonomous Systems*, 5(Volume 5, 2022):81–107.