

ALC Tableau Implementation for Knowledge Base Entailment

Chetan Reddy, CS19B012

Moideen Nihal, AE19B108

1 Introduction

This report provides an in-depth description of an implementation of the ALC Tableau algorithm for determining whether a given query in Negation Normal Form (NNF) is entailed by a Knowledge Base (KB) represented in ALC. The ALC Tableau algorithm is a popular method for reasoning over ALC KBs due to its soundness, completeness, and termination guarantees.

2 Algorithm Overview

1. Convert the NNF query into a concept in ALC Description Language.
2. Prepare TBox and ABox from KB and query.
3. Initialize the completion graph from the ABox.
4. Apply all TBox axioms to all individuals as well as unblocked variables.
5. If there is a contradiction, then backtrack and try the remaining \cup -branches.
6. The tableau terminates when all branches are either CLOSED or COMPLETE.
7. If all branches are CLOSED, then the NNF query is not entailed by the KB.
8. If all branches are COMPLETE, then the NNF query is entailed by the KB.

3 Assumptions

1. Both the NNF query and KB are in ALC Description Language.
2. Both have been converted into their respective forms before applying them to ALC Tableau.
The query should be in Negation Normal Form (NNF).

4 Implementation

The provided code is well-structured and consists of several components that work together to implement the ALC Tableau algorithm. The key components and their roles in the implementation are as follows:

1. **ALCConcept class:** This class represents concepts in the ALC language, including atomic concepts, conjunctions, disjunctions, negations, existential, and universal restrictions. The class also includes methods for representation (`--repr--`) and equality (`--eq--`) to facilitate debugging and comparison operations.
2. **Node class:** This class is responsible for representing a node in the tableau tree. It contains a set of ALCConcepts, a label, a list of parent nodes, and a count of children. The (`--repr--`) method is also implemented for easier debugging.
3. **RoleNode class:** This class is designed to represent a relationship between two Node objects connected by a role. It holds the role and references to the left and right nodes involved in the relationship.
4. **Parsing functions** (`parse_alc_concept_from_xml`, `parse_individual_from_xml`, `parse_xml_file`): These functions are dedicated to parsing ALC concepts and ABoxes from XML files. They use the `xml.etree.ElementTree` and `lxml.etree` libraries to read and extract information from the XML format.
5. **Tableau expansion rules:** The code implements functions for each of the four tableau expansion rules (AND, OR, EXISTS, FORALL). These functions are named `apply_rule_and`, `apply_rule_or`, `apply_rule_exists`, and `apply_rule_forall`, respectively. Each function takes a node and a concept as input and returns a list of new nodes generated by applying the rule.
6. **Expand function:** This function serves as the main driver for applying the tableau expansion rules. It iteratively calls the expansion rule functions for each concept in a node until no further expansion is possible.
7. **Entailment and satisfiability functions** (`isEntailed`, `IsSatisfiable`): The `isEntailed` function checks whether a query is entailed by the KB by negating the query and checking for satisfiability. The `IsSatisfiable` function is the core of the ALC Tableau algorithm. It uses an open list to store nodes that need to be expanded and a closed list to store processed nodes. The function iteratively processes nodes from the open list, applying expansion rules, and checks for closure (i.e., the presence of contradictory concepts).

5 Results

The implemented ALC Tableau algorithm for determining whether a given query in Negation Normal Form (NNF) is entailed by a Knowledge Base has been successfully tested and found to work correctly on the given input. The algorithm demonstrates the expected behavior on a variety of constructed inputs.

6 Conclusion

ALC Tableau algorithm, as seen in the submitted code, works excellently in determining the entailment of a Negation Normal Form (NNF) query by a Knowledge Base.