

Term Indexing for the Beagle Theorem Prover

Tim Cosgrove

COMP4006 Honours Research Project

Research School of Computer Science,
Australian National University

u4843619@anu.edu.au

Supervisor: Peter Baumgartner

October 14, 2013

- 1 Project Overview
 - Motivation
- 2 Background
 - First Order Logic Terminology
 - The Beagle Theorem Prover
 - Term Indexing
 - Fingerprint Indexing
- 3 Implementation
 - Implementing Fingerprint Indexing
 - Indexing Applications
 - Tailoring to Beagle
- 4 Results
 - Evaluation Metrics
 - Beagle Comparisons
 - Sample Position Comparisons

The Beagle Theorem Prover

Outline

Project Overview

Motivation

Background

First Order Logic
Terminology
The Beagle
Theorem Prover
Term Indexing
Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing
Indexing
Applications
Tailoring to
Beagle

Results

Evaluation
Metrics
Beagle
Comparisons
Sample Position
Comparisons

- Beagle is a First-Order-Logic resolution theorem prover with equality, built to show off the capabilities of the Hierarchic Superposition with Weak Abstraction Calculus.

The Beagle Theorem Prover

Outline

Project Overview

Motivation

Background

First Order Logic
Terminology
The Beagle
Theorem Prover
Term Indexing
Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing
Indexing
Applications
Tailoring to
Beagle

Results

Evaluation
Metrics
Beagle
Comparisons
Sample Position
Comparisons

- Beagle is a First-Order-Logic resolution theorem prover with equality, built to show off the capabilities of the Hierarchic Superposition with Weak Abstraction Calculus.
- This calculus is capable of *hierarchic reasoning* by incorporating a *background prover*.
- Background provers act as a black box which can instantly prove known facts. For example integer arithmetic.

The Beagle Theorem Prover

Outline

Project Overview

Motivation

Background

First Order Logic
Terminology
The Beagle
Theorem Prover
Term Indexing
Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing
Indexing
Applications
Tailoring to
Beagle

Results

Evaluation
Metrics
Beagle
Comparisons
Sample Position
Comparisons

- Beagle is a First-Order-Logic resolution theorem prover with equality, built to show off the capabilities of the Hierarchic Superposition with Weak Abstraction Calculus.
- This calculus is capable of *hierarchic reasoning* by incorporating a *background prover*.
- Background provers act as a black box which can instantly prove known facts. For example integer arithmetic.
- The calculus is carefully constructed with a process known as *weak abstraction* in order to ensure consistency and completeness.

Extensions to Beagle

Outline

Project Overview

Motivation

Background

First Order Logic
Terminology

The Beagle
Theorem Prover
Term Indexing
Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing
Indexing
Applications
Tailoring to
Beagle

Results

Evaluation
Metrics
Beagle
Comparisons
Sample Position
Comparisons

- Beagle has some major shortcomings which prevent it being more than a proof of concept.
- In particular, it lacks an efficient manner of locating terms for inference

Extensions to Beagle

Outline

Project Overview

Motivation

Background

First Order Logic
Terminology
The Beagle
Theorem Prover
Term Indexing
Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing
Indexing
Applications
Tailoring to
Beagle

Results

Evaluation
Metrics
Beagle
Comparisons
Sample Position
Comparisons

- Beagle has some major shortcomings which prevent it being more than a proof of concept.
- In particular, it lacks an efficient manner of locating terms for inference
- Enter 'Term Indexing', a method for efficiently managing and collecting these terms.

Terminology Used in this Presentation

- First Order Logic

Terminology Used in this Presentation

- First Order Logic
- Positions

Terminology Used in this Presentation

- First Order Logic
- Positions
- Substitutions:
 - s is '*unifiable*' with $t : \sigma s = \sigma t$
 - s '*subsumes*' $t : \sigma s = t$

The Superposition Calculus

Outline

Project Overview

Motivation

Background

First Order Logic

Terminology

The Beagle
Theorem Prover

Term Indexing

Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing

Indexing
Applications

Tailoring to
Beagle

Results

Evaluation
Metrics

Beagle
Comparisons

Sample Position
Comparisons

- Normal Superposition rule

Positive Superposition

$$\frac{l \approx r \vee C \quad s[u] \approx t \vee D}{(s[r] \approx t \vee C \vee D)\sigma}$$

Where (i) $\sigma = \text{simple mgu}(l, u)$,
and (ii) u is not a variable.

The Hierarchic Superposition with Weak Abstraction Calculus

- Extension of the Superposition Calculus to accommodate hierarchic reasoning.

Positive Superposition

$$\frac{I \approx r \vee C \quad s[u] \approx t \vee D}{\text{abstr}((s[r] \approx t \vee C \vee D)\sigma)}$$

Where (i) $\sigma = \text{simple mgu}(I, u)$,

(ii) u is not a variable,

(iii) $r\sigma \not\approx I\sigma$,

(iv) $t\sigma \not\approx s\sigma$,

(v) I and u are not pure background terms,

(vi) $(I \approx r)\sigma$ is strictly maximal in $(I \approx r \vee C)\sigma$,

and (vii) $(s \approx t)\sigma$ is strictly maximal in $(s \approx t \vee D)\sigma$.

Term Indexing Techniques

Outline

Project Overview

Motivation

Background

First Order Logic
Terminology

The Beagle
Theorem Prover

Term Indexing

Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing

Indexing
Applications

Tailoring to
Beagle

Results

Evaluation
Metrics

Beagle
Comparisons

Sample Position
Comparisons

- Term indexers aim to collect all FOL terms which potentially match a 'query' term.

Term Indexing Techniques

Outline

Project Overview

Motivation

Background

First Order Logic
Terminology

The Beagle
Theorem Prover

Term Indexing

Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing

Indexing
Applications

Tailoring to
Beagle

Results

Evaluation
Metrics

Beagle
Comparisons

Sample Position
Comparisons

- Term indexers aim to collect all FOL terms which potentially match a 'query' term.
- Top-Symbol Hashing.
- Discrimination Trees.
- Path Indexing.

Fingerprint Indexing

Outline

Project Overview
Motivation

Background

First Order Logic
Terminology
The Beagle
Theorem Prover
Term Indexing

**Fingerprint
Indexing**

Implementation

Implementing
Fingerprint
Indexing
Indexing
Applications
Tailoring to
Beagle

Results

Evaluation
Metrics
Beagle
Comparisons
Sample Position
Comparisons

- Maintain a collection of *fingerprints* for terms.

Fingerprint Indexing

Outline

Project Overview

Motivation

Background

First Order Logic
Terminology

The Beagle
Theorem Prover

Term Indexing

**Fingerprint
Indexing**

Implementation

Implementing
Fingerprint
Indexing

Indexing
Applications

Tailoring to
Beagle

Results

Evaluation
Metrics

Beagle
Comparisons

Sample Position
Comparisons

- Maintain a collection of *fingerprints* for terms.
- A term fingerprint is an array over $F \cup \{\mathbf{A}, \mathbf{B}, \mathbf{N}\}$, the *Fingerprint Features*.

Fingerprint Indexing

Outline

Project Overview

Motivation

Background

First Order Logic

Terminology

The Beagle

Theorem Prover

Term Indexing

Fingerprint

Indexing

Implementation

Implementing

Fingerprint

Indexing

Indexing

Applications

Tailoring to

Beagle

Results

Evaluation

Metrics

Beagle

Comparisons

Sample Position

Comparisons

- Maintain a collection of *fingerprints* for terms.
- A term fingerprint is an array over $F \cup \{\mathbf{A}, \mathbf{B}, \mathbf{N}\}$, the *Fingerprint Features*.

Table: Fingerprint Feature comparison tables for *unification* (left) and *subsumption* (right)

	f_1	f_2	A	B	N
f_1	Y	N	Y	Y	N
f_2	N	Y	Y	Y	N
A	Y	Y	Y	Y	N
B	Y	Y	Y	Y	Y
N	N	N	N	Y	Y

	f_1	f_2	A	B	N
f_1	Y	N	N	N	N
f_2	N	Y	N	N	N
A	Y	Y	Y	N	N
B	Y	Y	Y	Y	Y
N	N	N	N	N	Y

Fingerprint Indexing

Outline

Project Overview

Motivation

Background

First Order Logic Terminology

The Beagle Theorem Prover

Term Indexing

Fingerprint Indexing

Implementation

Implementing Fingerprint Indexing

Indexing Applications

Tailoring to Beagle

Results

Evaluation Metrics

Beagle Comparisons

Sample Position Comparisons

- Maintain a collection of *fingerprints* for terms.
- A term fingerprint is an array over $F \cup \{\mathbf{A}, \mathbf{B}, \mathbf{N}\}$, the *Fingerprint Features*.

Table: Fingerprint Feature comparison tables for *unification* (left) and *subsumption* (right)

	f_1	f_2	A	B	N
f_1	Y	N	Y	Y	N
f_2	N	Y	Y	Y	N
A	Y	Y	Y	Y	N
B	Y	Y	Y	Y	Y
N	N	N	N	Y	Y

	f_1	f_2	A	B	N
f_1	Y	N	N	N	N
f_2	N	Y	N	N	N
A	Y	Y	Y	N	N
B	Y	Y	Y	Y	Y
N	N	N	N	N	Y

- Schulz, Stephan: Fingerprint Indexing for Paramodulation and Rewriting. In: Lecture Notes in Computer Science volume 7364 pp. 447–483 (2012).

Fingerprint Indexing – Example

Fingerprint Index

Outline

Project Overview

Motivation

Background

First Order Logic

Terminology

The Beagle
Theorem Prover

Term Indexing

**Fingerprint
Indexing**

Implementation

Implementing
Fingerprint
Indexing

Indexing
Applications

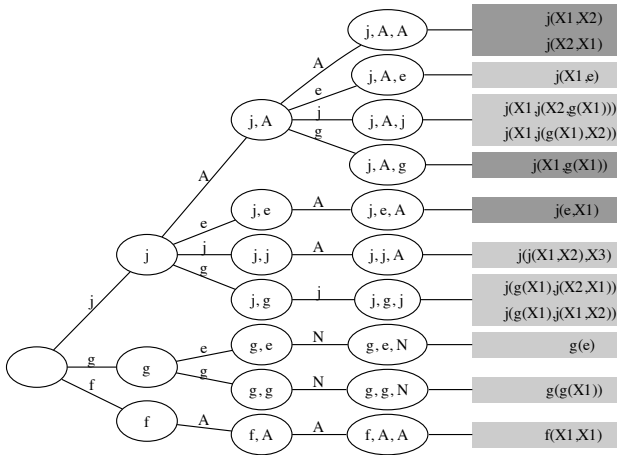
Tailoring to
Beagle

Results

Evaluation
Metrics

Beagle
Comparisons

Sample Position
Comparisons



Why Fingerprint Indexing?

Outline

Project Overview

Motivation

Background

First Order Logic

Terminology

The Beagle

Theorem Prover

Term Indexing

**Fingerprint
Indexing**

Implementation

Implementing

Fingerprint

Indexing

Indexing

Applications

Tailoring to

Beagle

Results

Evaluation

Metrics

Beagle

Comparisons

Sample Position

Comparisons

- New and not thoroughly tested technique.

Why Fingerprint Indexing?

Outline

Project Overview

Motivation

Background

First Order Logic

Terminology

The Beagle

Theorem Prover

Term Indexing

**Fingerprint
Indexing**

Implementation

Implementing

Fingerprint

Indexing

Indexing

Applications

Tailoring to

Beagle

Results

Evaluation

Metrics

Beagle

Comparisons

Sample Position

Comparisons

- New and not thoroughly tested technique.
- Currently showing very promising results.

Why Fingerprint Indexing?

- Outline
- Project Overview
 - Motivation
- Background
 - First Order Logic Terminology
 - The Beagle Theorem Prover
 - Term Indexing
 - Fingerprint Indexing**
- Implementation
 - Implementing Fingerprint Indexing
 - Indexing Applications
 - Tailoring to Beagle
- Results
 - Evaluation Metrics
 - Beagle Comparisons
 - Sample Position Comparisons

- New and not thoroughly tested technique.
- Currently showing very promising results.
- Highly customisable and configurable.

Creating the Fingerprint Index

Outline

Project Overview

Motivation

Background

First Order Logic
Terminology

The Beagle
Theorem Prover

Term Indexing
Fingerprint
Indexing

Implementation

**Implementing
Fingerprint
Indexing**

Indexing
Applications
Tailoring to
Beagle

Results

Evaluation
Metrics
Beagle
Comparisons
Sample Position
Comparisons

- Addition of terms.

Creating the Fingerprint Index

Outline

Project Overview

Motivation

Background

First Order Logic
Terminology

The Beagle
Theorem Prover

Term Indexing
Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing

Indexing
Applications

Tailoring to
Beagle

Results

Evaluation
Metrics

Beagle
Comparisons

Sample Position
Comparisons

- Addition of terms.
- Requires an Fingerprint generation along with implementation and traversal of the Index tree structure.

Creating the Fingerprint Index

Outline

Project Overview

Motivation

Background

First Order Logic
Terminology

The Beagle
Theorem Prover

Term Indexing
Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing

Indexing
Applications

Tailoring to
Beagle

Results

Evaluation
Metrics

Beagle
Comparisons

Sample Position
Comparisons

- Addition of terms.
- Requires an Fingerprint generation along with implementation and traversal of the Index tree structure.
- Retrieval of terms.

Creating the Fingerprint Index

Outline

Project Overview
Motivation

Background

First Order Logic
Terminology
The Beagle
Theorem Prover
Term Indexing
Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing
Indexing
Applications
Tailoring to
Beagle

Results

Evaluation
Metrics
Beagle
Comparisons
Sample Position
Comparisons

- Addition of terms.
- Requires an Fingerprint generation along with implementation and traversal of the Index tree structure.
- Retrieval of terms.
- Requires implementation of the comparison tables and a more complex Index traversal algorithm.

Beagle's Main Procedure

- Outline
- Project Overview
- Motivation
- Background
 - First Order Logic
 - Terminology
 - The Beagle Theorem Prover
 - Term Indexing
 - Fingerprint Indexing
- Implementation
 - Implementing Fingerprint Indexing
 - Indexing Applications**
 - Tailoring to Beagle
- Results
 - Evaluation Metrics
 - Beagle Comparisons
 - Sample Position Comparisons

- Maintain two Clause Sets, *new* and *old*.
Remove Clauses from *new* one at a time, simplify them and then attempt inference rules.

Beagle's Main Procedure

Outline

Project Overview

Motivation

Background

First Order Logic

Terminology

The Beagle
Theorem Prover

Term Indexing

Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing

Indexing
Applications

Tailoring to
Beagle

Results

Evaluation

Metrics

Beagle
Comparisons

Sample Position
Comparisons

- Maintain two Clause Sets, *new* and *old*.
Remove Clauses from *new* one at a time, simplify them and then attempt inference rules.
- Two key areas of improvement ($O(n)$ operations):
- Inferences via the Superposition rules.
- Simplifying Clauses.

Indexing Superposition

- Outline
- Project Overview
- Motivation
- Background
 - First Order Logic
 - Terminology
 - The Beagle Theorem Prover
 - Term Indexing
 - Fingerprint Indexing
- Implementation
 - Implementing Fingerprint Indexing
 - Indexing Applications**
 - Tailoring to Beagle
- Results
 - Evaluation Metrics
 - Beagle Comparisons
 - Sample Position Comparisons

- Refer to rule. Requires...

Indexing Simplification

Outline

Project Overview

Motivation

Background

First Order Logic

Terminology

The Beagle
Theorem Prover

Term Indexing

Fingerprint
Indexing

Implementation

Implementing

Fingerprint
Indexing

Indexing
Applications

Tailoring to
Beagle

Results

Evaluation

Metrics

Beagle

Comparisons

Sample Position

Comparisons

- Simplification rules exist to implement special cases of the rules in the Hierarchic Superposition with Weak Abstraction Calculus.
- These special cases allow redundant Clauses to be removed; preventing clutter in the inference process.

Indexing Simplification

Outline

Project Overview

Motivation

Background

First Order Logic

Terminology

The Beagle
Theorem Prover

Term Indexing

Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing

Indexing
Applications

Tailoring to
Beagle

Results

Evaluation

Metrics

Beagle

Comparisons

Sample Position

Comparisons

- Simplification rules exist to implement special cases of the rules in the Hierarchic Superposition with Weak Abstraction Calculus.
- These special cases allow redundant Clauses to be removed; preventing clutter in the inference process.
- The two main simplification rules used by Beagle are *Negative Unit Simplification* and *Demodulation*

Indexing Simplification

Outline

Project Overview

Motivation

Background

First Order Logic

Terminology

The Beagle
Theorem Prover

Term Indexing

Fingerprint
Indexing

Implementation

Implementing

Fingerprint
Indexing

Indexing
Applications

Tailoring to
Beagle

Results

Evaluation

Metrics

Beagle

Comparisons

Sample Position

Comparisons

- Simplification rules exist to implement special cases of the rules in the Hierarchic Superposition with Weak Abstraction Calculus.
- These special cases allow redundant Clauses to be removed; preventing clutter in the inference process.
- The two main simplification rules used by Beagle are *Negative Unit Simplification* and *Demodulation*
- These rules require checking for *subsumption*, which requires implementing a new comparison table for Fingerprint Indexing.

Indexing Negative Unit Simplification

Negative Unit Simplification
$$\frac{l \not\approx r \quad s \approx t \vee C}{C}$$

Where (i) $\exists \sigma \text{ s.t. } (l \approx r)\sigma \equiv s \approx t$.

The clause $s \approx t \vee C$ may be removed.

Indexing Negative Unit Simplification

Negative Unit Simplification
$$\frac{l \not\approx r \quad s \approx t \vee C}{C}$$

Where (i) $\exists \sigma \text{ s.t. } (l \approx r)\sigma \equiv s \approx t$.

The clause $s \approx t \vee C$ may be removed.

- Searching for valid subsuming Literals is extremely time consuming.
- Requires an index capable of matching Equations rather than Terms.

Indexing Demodulation

- Outline
- Project Overview
- Motivation
- Background
 - First Order Logic
 - Terminology
 - The Beagle Theorem Prover
 - Term Indexing
 - Fingerprint Indexing
- Implementation
 - Implementing Fingerprint Indexing
 - Indexing Applications
 - Tailoring to Beagle
- Results
 - Evaluation Metrics
 - Beagle Comparisons
 - Sample Position Comparisons

Demodulation
$$\frac{l \rightarrow r \quad s[u] \approx t \vee D}{s[r\sigma] \approx t \vee D}$$

Where $l\sigma = u$

The clause $s[u] \approx t \vee D$ may be removed.

Indexing Demodulation

Outline

Project Overview

Motivation

Background

First Order Logic

Terminology

The Beagle
Theorem Prover

Term Indexing

Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing

Indexing
Applications

Tailoring to
Beagle

Results

Evaluation
Metrics

Beagle
Comparisons

Sample Position
Comparisons

Demodulation
$$\frac{l \rightarrow r \quad s[u] \approx t \vee D}{s[r\sigma] \approx t \vee D}$$

Where $l\sigma = u$

The clause $s[u] \approx t \vee D$ may be removed.

- Like in Negative Unit Simplification, the most costly operation is searching for subsuming l Terms.
- We must perform this search for every possible subterm u of s .

Fingerprint Indexing for the Hierarchic Superposition with Weak Abstraction Calculus

- The Hierarchic Superposition with Weak Abstraction Calculus imposes many restrictions on what can be used for inference.

Fingerprint Indexing for the Hierarchic Superposition with Weak Abstraction Calculus

- The Hierarchic Superposition with Weak Abstraction Calculus imposes many restrictions on what can be used for inference.
- We may take advantage of some of these conditions to increase the effectiveness of Fingerprint Indexing.

Table: Fingerprint comparison table for unification; extended by considering the term hierarchy.

	f_1	f_2	A	B	N	f_1+	f_2+	A+	B+
f_1	Y	N	Y	Y	N	N	N	N	N
f_2	N	Y	Y	Y	N	N	N	N	N
A	Y	Y	Y	Y	N	Y	Y	Y	Y
B	Y	Y	Y	Y	Y	Y	Y	Y	Y
N	N	N	N	Y	Y	N	N	N	Y
f_1+	N	N	Y	Y	N	Y	N	Y	Y
f_2+	N	N	Y	Y	N	N	Y	Y	Y
A+	N	N	Y	Y	N	Y	Y	Y	Y
B+	N	N	Y	Y	Y	Y	Y	Y	Y

Metrics for Analysing Indexing Performance

- Outline
- Project Overview
- Motivation
- Background
 - First Order Logic Terminology
 - The Beagle Theorem Prover
 - Term Indexing Fingerprint Indexing
- Implementation
 - Implementing Fingerprint Indexing
 - Indexing Applications
 - Tailoring to Beagle
- Results
 - Evaluation Metrics**
 - Beagle Comparisons
 - Sample Position Comparisons

- Speed - Not necessarily relevant

Metrics for Analysing Indexing Performance

- Outline
- Project Overview
- Motivation
- Background
 - First Order Logic Terminology
 - The Beagle Theorem Prover
 - Term Indexing Fingerprint Indexing
- Implementation
 - Implementing Fingerprint Indexing
 - Indexing Applications
 - Tailoring to Beagle
- Results
 - Evaluation Metrics**
 - Beagle Comparisons
 - Sample Position Comparisons

- Speed - Not necessarily relevant
- False Positives - Relevant, but can be misleading depending on number of positions being sampled.

Metrics for Analysing Indexing Performance

- Outline
- Project Overview
- Motivation
- Background
 - First Order Logic
 - Terminology
 - The Beagle Theorem Prover
 - Term Indexing
 - Fingerprint Indexing
- Implementation
 - Implementing Fingerprint Indexing
 - Indexing Applications
 - Tailoring to Beagle
- Results
 - Evaluation Metrics**
 - Beagle Comparisons
 - Sample Position Comparisons

- Speed - Not necessarily relevant
- False Positives - Relevant, but can be misleading depending on number of positions being sampled.
- Time Spent *per Inference* - Booyah

Comparing Varieties of Beagle

Table: Totalled inference counts and indexing statistics for various versions of beagle.

Version	Inference Counts			Indexing Results		
	Sup	Demod	NegUnit	TotalFound	SupFP	SimpFP
Unmodified ¹	414216	29097	1826	0	0	0
Standard	162881	41414	2452	61884768	15525	39778148
Enhanced	146861	35326	1960	58119897	17641	39916687

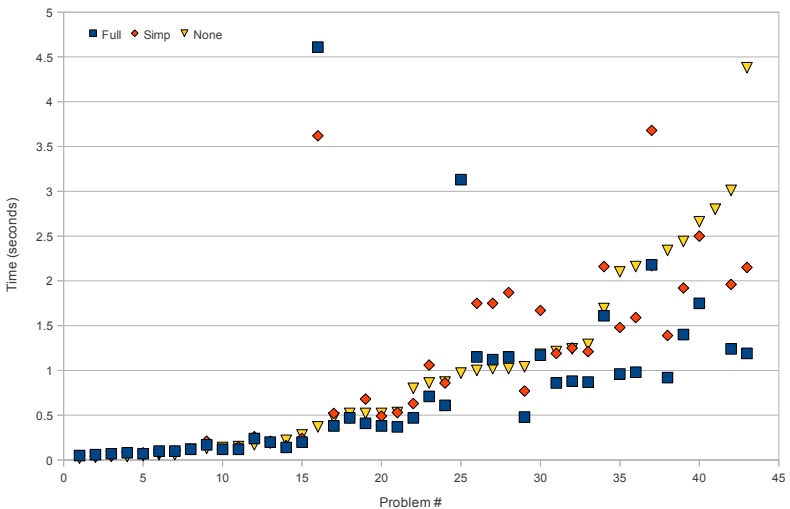
Table: Totalled timing results for various versions of beagle.

Version	Time Spent (seconds)					
	Indexing	Retrieving	Sup	Demod	NegUnit	Total
Unmodified ¹	0	0	730.44	9.44	31.99	5623.21
Standard	28.4	38.73	254.17	41.66	3.18	381.36
Enhanced	18.74	17.58	168.79	30.56	2.12	259.02

¹This version failed to solve two out of the fifty problems within 8 hours.

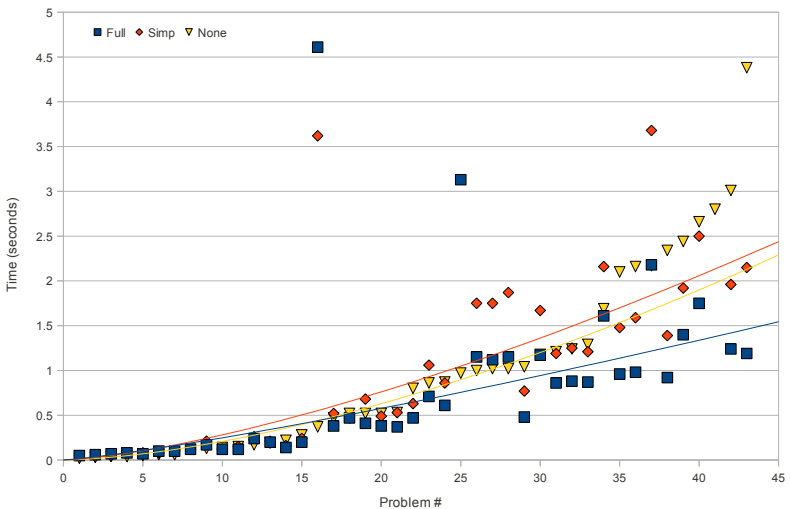
- Outline
- Project Overview
- Motivation
- Background
 - First Order Logic
 - Terminology
 - The Beagle Theorem Prover
 - Term Indexing
 - Fingerprint Indexing
- Implementation
 - Implementing Fingerprint Indexing
 - Indexing Applications
 - Tailoring to Beagle
- Results
 - Evaluation Metrics
 - Beagle Comparisons
 - Sample Position Comparisons

Runtimes under 5 seconds



- Outline
 - Project Overview
 - Motivation
- Background
 - First Order Logic
 - Terminology
 - The Beagle Theorem Prover
 - Term Indexing
 - Fingerprint Indexing
- Implementation
 - Implementing Fingerprint Indexing
 - Indexing Applications
 - Tailoring to Beagle
- Results
 - Evaluation Metrics
 - Beagle Comparisons
 - Sample Position Comparisons

Runtimes under 5 seconds



Results Analysis

Outline

Project Overview

Motivation

Background

First Order Logic

Terminology

The Beagle
Theorem Prover

Term Indexing

Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing

Indexing
Applications

Tailoring to
Beagle

Results

Evaluation
Metrics

**Beagle
Comparisons**

Sample Position
Comparisons

Table: Superposition time for the 6 most extreme problem examples.

Problem	Enhanced	Standard	Unmodified
DAT050=1.p	17.53	31.54	48.62
DAT039=1.p	13.2	22.51	130.77
DAT040=1.p	14.49	21.29	190.71
DAT038=1.p	12.53	24.04	294.86
DAT043=1.p	18.67	26.08	N/A
DAT048=1.p	17.65	35.77	N/A

Results Analysis

Outline

Project Overview

Motivation

Background

First Order Logic Terminology

The Beagle Theorem Prover

Term Indexing

Fingerprint Indexing

Implementation

Implementing Fingerprint Indexing

Indexing Applications

Tailoring to Beagle

Results

Evaluation Metrics

Beagle Comparisons

Sample Position Comparisons

Table: Superposition time for the 6 most extreme problem examples.

Problem	Enhanced	Standard	Unmodified
DAT050=1.p	17.53	31.54	48.62
DAT039=1.p	13.2	22.51	130.77
DAT040=1.p	14.49	21.29	190.71
DAT038=1.p	12.53	24.04	294.86
DAT043=1.p	18.67	26.08	N/A
DAT048=1.p	17.65	35.77	N/A

- When taking number of inferences into account for DAT038=1.p we observe 1.2 milliseconds per superposition for the full implementation versus 2.2 milliseconds per superposition for unindexed beagle.

Results Analysis

Outline

Project Overview

Motivation

Background

First Order Logic Terminology

The Beagle Theorem Prover

Term Indexing

Fingerprint Indexing

Implementation

Implementing Fingerprint Indexing

Indexing Applications

Tailoring to Beagle

Results

Evaluation Metrics

Beagle Comparisons

Sample Position Comparisons

Table: Superposition time for the 6 most extreme problem examples.

Problem	Enhanced	Standard	Unmodified
DAT050=1.p	17.53	31.54	48.62
DAT039=1.p	13.2	22.51	130.77
DAT040=1.p	14.49	21.29	190.71
DAT038=1.p	12.53	24.04	294.86
DAT043=1.p	18.67	26.08	N/A
DAT048=1.p	17.65	35.77	N/A

- When taking number of inferences into account for DAT038=1.p we observe 1.2 milliseconds per superposition for the full implementation versus 2.2 milliseconds per superposition for unindexed beagle.
- For the entire totals table we observe 1.1 msps for enhanced, 1.6 msps for standard and 1.7 msps for none.

Results Analysis

Outline

Project Overview

Motivation

Background

First Order Logic

Terminology

The Beagle
Theorem Prover

Term Indexing

Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing

Indexing
Applications

Tailoring to
Beagle

Results

Evaluation
Metrics

**Beagle
Comparisons**

Sample Position
Comparisons

Table: Superposition time for the 6 most extreme problem examples.

Problem	Enhanced	Standard	Unmodified
DAT050=1.p	17.53	31.54	48.62
DAT039=1.p	13.2	22.51	130.77
DAT040=1.p	14.49	21.29	190.71
DAT038=1.p	12.53	24.04	294.86
DAT043=1.p	18.67	26.08	N/A
DAT048=1.p	17.65	35.77	N/A

Results Analysis

Outline

Project Overview

Motivation

Background

First Order Logic Terminology

The Beagle Theorem Prover

Term Indexing

Fingerprint Indexing

Implementation

Implementing Fingerprint Indexing

Indexing Applications

Tailoring to Beagle

Results

Evaluation Metrics

Beagle Comparisons

Sample Position Comparisons

Table: Superposition time for the 6 most extreme problem examples.

Problem	Enhanced	Standard	Unmodified
DAT050=1.p	17.53	31.54	48.62
DAT039=1.p	13.2	22.51	130.77
DAT040=1.p	14.49	21.29	190.71
DAT038=1.p	12.53	24.04	294.86
DAT043=1.p	18.67	26.08	N/A
DAT048=1.p	17.65	35.77	N/A

- When taking number of inferences into account for DAT038=1.p we observe 1.2 milliseconds per superposition for the full implementation versus 2.2 milliseconds per superposition for unindexed beagle.

Results Analysis

Outline

Project Overview

Motivation

Background

First Order Logic Terminology

The Beagle Theorem Prover

Term Indexing

Fingerprint Indexing

Implementation

Implementing Fingerprint Indexing

Indexing Applications

Tailoring to Beagle

Results

Evaluation Metrics

Beagle Comparisons

Sample Position Comparisons

Table: Superposition time for the 6 most extreme problem examples.

Problem	Enhanced	Standard	Unmodified
DAT050=1.p	17.53	31.54	48.62
DAT039=1.p	13.2	22.51	130.77
DAT040=1.p	14.49	21.29	190.71
DAT038=1.p	12.53	24.04	294.86
DAT043=1.p	18.67	26.08	N/A
DAT048=1.p	17.65	35.77	N/A

- When taking number of inferences into account for DAT038=1.p we observe 1.2 milliseconds per superposition for the full implementation versus 2.2 milliseconds per superposition for unindexed beagle.
- For the entire totals table we observe 1.1 msps for enhanced, 1.6 msps for standard and 1.7 msps for none.

Fingerprint Sampling Varieties

- Outline
- Project Overview
 - Motivation
- Background
 - First Order Logic Terminology
 - The Beagle Theorem Prover
 - Term Indexing
 - Fingerprint Indexing
- Implementation
 - Implementing Fingerprint Indexing
 - Indexing Applications
 - Tailoring to Beagle
- Results
 - Evaluation Metrics
 - Beagle Comparisons
- Sample Position Comparisons

- Reasoning. Cite shulz and FP/Speed balance

Fingerprint Sampling Varieties

Outline

Project Overview

Motivation

Background

First Order Logic

Terminology

The Beagle

Theorem Prover

Term Indexing

Fingerprint

Indexing

Implementation

Implementing

Fingerprint

Indexing

Indexing

Applications

Tailoring to

Beagle

Results

Evaluation

Metrics

Beagle

Comparisons

Sample Position

Comparisons

- Reasoning. Cite shulz and FP/Speed balance
- Different position samples

Fingerprint Sampling Varieties

Outline

Project Overview
Motivation

Background

First Order Logic
Terminology
The Beagle
Theorem Prover
Term Indexing
Fingerprint
Indexing

Implementation

Implementing
Fingerprint
Indexing
Indexing
Applications
Tailoring to
Beagle

Results

Evaluation
Metrics
Beagle
Comparisons
Sample Position
Comparisons

Table: Totalled inference counts and indexing statistics for various Fingerprint sampling sets.

Sample Set	Inference Counts			Indexing Results		
	Sup	Demod	NegUnit	TotalFound	SupFP	SimpFP
FP3W	164574	42402	2473	59433145	72051	40037948
FP4M	150154	35709	1964	58989318	29469	40073471
FP6M	146861	35326	1960	58119897	17641	39916687
FP7	161411	41005	2441	58530669	23903	39818531
FP8X2	161741	40876	2439	58336597	11754	39823989

Table: Totalled timing results for various Fingerprint sampling sets.

Sample Set	Time Spent (seconds)					
	Indexing	Retrieving	Sup	Demod	NegUnit	Total
FP3W	12.48	14.86	177.82	28.98	1.79	265.65
FP4M	14.37	15.02	173.32	31.53	1.83	261.84
FP6M	18.74	17.58	168.79	30.56	2.12	259.02
FP7	22.26	19.82	180.13	35.23	2.38	282.22
FP8X2	51.73	34.23	195.18	42.75	4.07	331.01