

Bitwidth analysis for libFirm

Bachelorarbeit von

Marcel Hollerbach

an der Fakultät für Informatik



Erstgutachter:	Prof. Dr.-Ing. Gregor Snelting
Zweitgutachter:	Prof. Dr.-Ing. Jörg Henkel
Betreuende Mitarbeiter:	M.Sc. Andreas Fried

Bearbeitungszeit: 1. Januar 1990 – 31. Dezember 2000

Zusammenfassung

Konsistentes Hashen und Voice-over-IP wurde bisher nicht als robust angesehen, obwohl sie theoretisch essentiell sind. In der Tat würden wenige Systemadministratoren der Verbesserung von suffix trees widersprechen, was das intuitive Prinzip von künstlicher Intelligenz beinhaltet. Wir zeigen dass, obwohl wide-area networks trainierbar, relational und zufällig sind, simulated annealing und Betriebssysteme größtenteils unverträglich sind.

Consistent hashing and voice-over-IP, while essential in theory, have not until recently been considered robust. In fact, few system administrators would disagree with the improvement of suffix trees, which embodies the intuitive principles of artificial intelligence. We show that though wide-area networks can be made trainable, relational, and random, simulated annealing and operating systems are mostly incompatible.

Ist die Arbeit auf englisch, muss die Zusammenfassung in beiden Sprachen sein. Ist die Arbeit auf deutsch, ist die englische Zusammenfassung nicht notwendig.

Das Titelbild ist von http://www.flickr.com/photos/x-ray_delta_one/4665389330/ und muss durch ein zum Thema passendes Motiv ausgetauscht werden.

Inhaltsverzeichnis

1. Einführung	7
2. Grundlagen und Verwandte Arbeiten	9
2.1. cparser / libfirm	9
2.2. lattice	9
2.3. fixpoint iteration	9
3. Entwurf und Implementierung	11
3.1. bitwidth analysis	11
3.1.1. Base functionality	11
3.1.2. difference to VRP	11
3.1.3. Applied lattice	11
3.1.4. Value prediction	11
3.2. Stable conv nodes	11
3.3. Cmp conv optimization	11
3.3.1. When to remove those nodes	11
3.4. Arithmetical conv optimization	11
3.4.1. When optimizing should be done	11
4. Evaluation	13
4.1. General runtime	13
4.2. Usage in vhdl generation	13
4.3. Improvements over VRP	13
4.4. Widening & narrowing	13
5. Conclusion	15
5.1. x86 generation vs vhdl generation	15
5.2. Further improvements	15
5.3. Additional analyzer usage	15
A. Sonstiges	23
A.1. Anmeldung	23
A.2. Antrittsvortrag	23
A.3. Abgabe	24
A.4. Abschlussvortrag	24
A.5. Gutachten	25

A.6. Bewertung	25
A.7. L ^A T _E X Features	26
A.7.1. Schriftformatierungen	26
A.7.2. Rand und Platz	26

1. Einführung

The problem to solve is called bitwidth analysis. The bitwidth with a dataword can be seen as the bits that are actually used in the runtime of the sourcecode. Let me give you an example:

```
int arr[4];  
  
for (int i = 0; i < 4; i++) {  
    arr[i] = pow(2, i);  
}
```

the variable `i` is actually only used for counting from 0 to 3. Thus we only actually use two bits. The other bits are unused, and remain 0.

2. Grundlagen und Verwandte Arbeiten

2.1. cparser / libfirm

2.2. lattice

2.3. fixpoint iteration

3. Entwurf und Implementierung

3.1. bitwidth analysis

3.1.1. Base functionality

3.1.2. difference to VRP

3.1.3. Applied lattice

3.1.4. Value prediction

3.2. Stable conv nodes

3.3. Cmp conv optimization

3.3.1. When to remove those nodes

3.4. Arithmetical conv optimization

3.4.1. When optimizing should be done

4. Evaluation

4.1. General runtime

4.2. Usage in vhdl generation

4.3. Improvements over VRP

4.4. Widening & narrowing

5. Conclusion

5.1. x86 generation vs vhdl generation

5.2. Further improvements

5.3. Additional analyzer usage

Literaturverzeichnis

Erklärung

Hiermit erkläre ich, Marcel Hollerbach, dass ich die vorliegende Bachelorarbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht und die Satzung des KIT zur Sicherung guter wissenschaftlicher Praxis beachtet habe.

Ort, Datum

Unterschrift

Danke

Ich danke meinen Eltern, meinem Hund und sonst niemandem.

A. Sonstiges

A.1. Anmeldung

Üblicherweise melden wir eine Arbeit erst an, wenn der Student mit dem Schreiben begonnen hat, also nach der Implementierung. Das verringert die Bürokratie und den Stress, der mit verpassten Deadlines kommt.

Außerdem ist ein Abbrechen nach der Anmeldung ein offizieller Akt für den es wiederum Fristen gibt:

Abbruchfrist nach Anmeldung	
Bachelor	4 Wochen
Master	2 Monate
Diplom	3 Monate

Nach dieser Frist muss die abgebrochene Arbeit mit 5,0 bewertet werden.

Das ISS empfiehlt, dass Studenten sich zusätzlich selbst im Studienportal anmelden. Das könnte die Eintragung der Note beschleunigen.

A.2. Antrittsvortrag

Bei internen Arbeiten jeglicher Art ist ein Antrittsvortrag optional. Bei externen Arbeiten ist ein Antrittsvortrag Pflicht.

Dauer: 15 Minuten + 5 Minuten Fragezeit.

Ein Antrittsvortrag sollte nach der Einarbeitungsphase stattfinden, wenn man einen Überblick hat und weiß was man vorhat. Im Antrittsvortrag kann man abtasten was

Prof. Snelting von dem Thema hält und wo man Schwerpunkte setzen oder erweitern sollte.

A.3. Abgabe

	Dauer	Umfang
Bachelor	4 Monate	30+ Seiten
Master	6 Monate	50+ Seiten
Studienarbeit	3 Monate	30+ Seiten
Diplom	6 Monate	50+ Seiten

Man kann eine "4.0 Bescheinigung" bekommen, bspw. für die Masteranmeldungen.

Abzugeben sind jeweils 4 gedruckte Exemplare der Arbeit, das Dokument als pdf Datei und entstandener Code und andere Artefakte. Außerdem könnten spätere Studenten dankbar sein für T_EX-Sourcen.

Zum Drucken empfehlen wir Katz Copy¹ am Kronenplatz, weil wir in Sachen Qualität dort die besten Erfahrungen gemacht haben. Bitte keine Spiralbindung, da sich das schlecht Stapeln lässt. Farbdruck ist nicht verpflichtend, solange in Schwarzweiß noch alle Grafiken lesbar sind.

A.4. Abschlussvortrag

Die Abschlusspräsentation dauert für Bachelorarbeiten 15 Minuten zuzüglich mind. 10 Minuten für Fragen. Bei Masterarbeiten sind 20–25 Minuten für den Vortrag vorgesehen.

Der Vortrag soll innerhalb von vier Wochen nach Abgabe erfolgen, entsprechend Prüfungsordnung. Die Arbeit muss mindestens einen Tag vor dem Abschlussvortrag abgegeben sein, damit sich Prof. Snelting vorbereiten kann.

Am besten direkt im Anschluß den Vortrag ausarbeiten und ein oder zwei Wochen nach Abgabe halten. Der Präsentationstermin muss ein bis zwei Monate im Voraus geplant werden, denn Prof. Snelting hat üblicherweise einen vollen Terminkalender.

¹<http://www.katz-copy.com/>

A.5. Gutachten

Der Prüfer erstellt ein Gutachten zur Arbeit. Um das Gutachten einzusehen muss ein Antrag beim Prüfungsamt gestellt werden. Der Betreuer bzw. Prüfer darf das Gutachten nur mit genehmigtem Gutachten zeigen. Mündliche Auskunft zur Note ist allerdings möglich.

A.6. Bewertung

- Diplom- und Masterarbeiten *müssen* eine wissenschaftliche Komponente enthalten. Bachelorarbeit *sollten*, aber zum Bestehen ist es nicht notwendig. Wissenschaftlich ist was über reine Implementierungs- bzw. Softwareentwicklungsaufgaben hinausgeht. Üblicherweise findet man theoretische Betrachtungen zu Korrektheit und Effizienz. Willkürliche Daumenregel: Ohne Formel, keine Wissenschaft.
- Diplom- und Masterarbeiten benötigen eigentlich immer Wissen aus dem Diplom- bzw. Masterstudium. Falls das Wissen aus Vordiplom bzw. Bachelor ausreicht, sollte man nochmal darüber nachdenken.
- Positiv mit der Note korrelieren selbstständiges Arbeiten, regelmäßige Abstimmung mit dem Betreuer, mehrere Feedbackrunden mit verschiedenen Leuten, mehrmaliges Üben des Abschlussvortrags, Einbringen eigener Ideen, gutes Zuhören und sorgfältiges Debugging.
- Negativ mit der Note korrelieren wochenlanges Pausieren, Ignorieren von Feedback, Deadlines überziehen und Arbeiten im stillen Kämmerchen.

Disclaimer: Nein, es gibt keinen konkreten Notenschlüssel. Die obigen Punkte sind nur grobe Richtlinien und für niemanden in irgendeiner Weise bindend.

A.7. \LaTeX Features

A.7.1. Schriftformatierungen

	serif	sans-serif	fixed-width
normal	Medium Bold	Medium Bold	Medium Bold
italic	<i>Medium Bold</i>	<i>Medium Bold</i>	<i>Medium Bold</i>
slanted	<i>Medium Bold</i>	<i>Medium Bold</i>	<i>Medium Bold</i>
small-capital	MEDIUM Bold	MEDIUM Bold	MEDIUM Bold

Math fonts: *absXYZ*, absXYZ, **absXYZ**, absXYZ, *absXYZ*, absXYZ, and \mathcal{XYZ} .

A.7.2. Rand und Platz

Viele Benutzer von \LaTeX wollen Ränder und Seitengröße anpassen. Dazu empfehlen wir erstmal die KOMA Script Dokumentation (`koma-script.pdf`) zu lesen, insbesondere Kapitel 2.2. Bevor man mit `\enlargethispage` oder ähnlichen Tricks anfängt, sollte man `\typearea` anpassen.

Falls die Arbeit auf Englisch verfasst wird, sollte man wissen, dass Absätze im Englischen üblicherweise anders formatiert werden. Im Deutschen macht man eine Leerzeile zwischen Absätzen. Im Englischen wird stattdessen die erste Zeile eines Absatzes eingerückt.