Friedrich-Alexander-Universität
Technische Fakultät

Lehrstuhl für Informatik 4 · Systemsoftware

Toller Student

Über das Verhältnis zwischen Bachelorarbeit und resultierender Note

Bachelorarbeit im Fach Informatik
BA-2024-42

29. Mai 2024

Please cite as:
Toller Student, „Über das Verhältnis zwischen Bachelorarbeit und resultierender Note“, Bachelor's Thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Dept. of Computer Science, May 2024.

Friedrich-Alexander-Universität Erlangen-Nürnberg
Department Informatik
Systemsoftware
Martensstr. 1 · 91058 Erlangen · Germany

Über das Verhältnis zwischen Bachelorarbeit und resultierender Note

Bachelorarbeit im Fach Informatik

vorgelegt von

Toller Student

geb. am 1. Dezember 1985
in Hier

angefertigt am

Lehrstuhl für Informatik 4
Systemsoftware

Department Informatik
Friedrich-Alexander-Universität Erlangen-Nürnberg

Betreuer:	**Dipl.-Inf. Marianne Mustermann** **Max Mustermann, M.Sc.**
Betreuender Hochschullehrer:	**Prof. Dr.-Ing. habil. Wolfgang Schröder-Preikschat**
Beginn der Arbeit:	**1. Februar 2014**
Abgabe der Arbeit:	**1. August 2014**

Erklärung

Ich versichere, dass ich die Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Declaration

I declare that the work is entirely my own and was produced with no assistance from third parties. I certify that the work has not been submitted in the same or any similar form for assessment to any other examining body and all references, direct and indirect, are indicated as such and have been cited accordingly.

(Toller Student)
Erlangen, 29. Mai 2024

ABSTRACT

about 1/2 page:
(1) Motivation (Why do we care?)
(2) Problem statement (What problem are we trying to solve?)
(3) Approach (How did we go about it)
(4) Results (What's the answer?)
(5) Conclusion (What are the implications of the answer?)

KURZFASSUNG

Gleicher Text in Deutsch

INHALTSVERZEICHNIS

1 INTRODUCTION

general motivation for your work, context and goals: 1-2 pages

- **Context:** make sure to link where your work fits in
- **Problem:** gap in knowledge, too expensive, too slow, a deficiency, superseded technology
- **Strategy:** the way you will address the problem

1.1 Sample Section

The following samples explain how to insert cross-references, figures and tables, how to set math, algorithms and program code, how to add references, and how to use acronyms.

1.1.1 Cross References

Use the `\label` and `\cref` commands for cross references, e.g. to Abschnitt 1.1.1.

1.1.2 Figures

Abbildung 1.1 shows the distribution of the nodes in the sample setup at time $t = 0$, as well as the initial coverage with a sensing radius of 30 m and the communication graph for a communication range of 50 m. Figure captions should always be placed at the bottom of the figure.

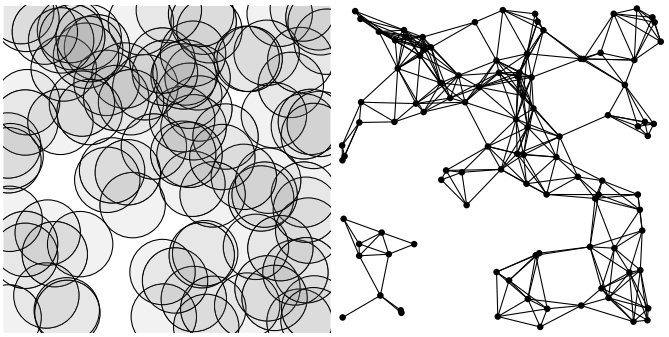


Abbildung 1.1 – Coverage and connectivity for a sample replication at time $t = 0$

1.1.3 Subfigures

Abbildungen 1.2a bis 1.2b show the distribution of the nodes in the sample setup at time $t = 0$, as well as the initial coverage with a sensing radius of 30 m and the communication graph for a communication range of 50 m.

1.1.4 Plotting

Plotting is an important part of every evaluation. In general, the user has to major options there: Either using an external application to generate plots, or to use an integrated, LaTeX-native plotting solution.

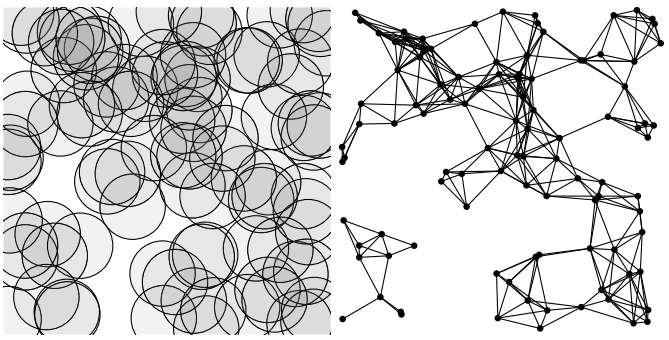
Popular external applications include gnuplot[1], matplotlib[2] or ggplot2[3]. If using one of those, please make sure to export the plots as either TeX (if available) or at least vector graphics such as PDF, and do not use bitmap graphics.

You can also make LaTeX generate plots automatically using `gnuplot`. Just put your data into a `csv` file in the `plots` subdirectory and create a `gnuplot` script ending in `.gplt`. Follow the example from `plots/ass_granularity.gplt`. To enable this feature, two steps are required

1. Enable shell-escape in the file `.latexmkrc`. See the comments there for details
2. Pass the gnuplot option to the `i4thesis` documentclass in `thesis.tex`, i.e. `\documentclass[...,gnuplot]{i4thesis}`
3. See the example code below (hidden as long as gnuplot is disabled)

Finally, the pgfplots[4] package constitutes a fully LaTeX-native solution to plot measurement data, and thus does not require external applications or the shell-escape feature. As this plotting happens in the context of your current document, it ensures a consistent styling and allows you to directly annotate datapoints by using Ti*k*Z. However, due to the limits of the TeX-engine, it might struggle with particularly large datasets.

[1] `http://www.gnuplot.info/`
[2] `https://matplotlib.org/stable/`
[3] `https://ggplot2.tidyverse.org/`
[4] `http://pgfplots.sourceforge.net/gallery.html`



(a) coverage **(b)** connectivity

Abbildung 1.2 – Subfigures showing coverage and connectivity for a sample replication at time $t = 0$

1.1.5 Tables

Tabelle 1.1 gives an overview of the discussed application classes. In contrast to figure captions, table captions are always placed above the table.

1.1.6 Math

Simple inlined equations: $\zeta(t) = \min(\zeta_{**}(t))$. The same in a numbered equation, i.e. Gleichung (1.1):

$$\zeta(t) = \min(\zeta_{**}(t)) \tag{1.1}$$

Equations covering multiple lines should be aligned. Note that the numbering is added automatically, independent of whether the equation is actually referenced or not:

$$sd_{max} = max((t_{i+1} - t_i) : \zeta(t_i) < 1, i \in [0, |T| - 1]) \tag{1.2}$$

$$\psi_{sd}(t) = \begin{cases} \frac{\Delta t_{sd}}{sd_{max}} & sd_{max} > 0 \\ 1 & sd_{max} = 0 \end{cases} \tag{1.3}$$

$$\zeta_{sd}(t) = \frac{\psi_{sd} - cl_{sd}}{c_{sd} - cl_{sd}} \tag{1.4}$$

1.1.7 Units

Units should be set using the `\SI` command: the measurements show that the car was accelerating at 5 $\frac{\mathrm{m}}{\mathrm{s}^2}$ until it reached its final speed of 100 $\frac{\mathrm{km}}{\mathrm{h}}$. Longer unitless numbers or ranges can be typeset using the `\num` and `\numrange` commands, respectively: The number 12,345,678 lies in the range of 10,000,000…20,000,000. The value $4.1 \pm 0.2\,\mathrm{m}$ is not exactly measured. Tabelle 1.2 gives an example of how to typeset numbers and units in tables.

Tabelle 1.1 – Sensor network applications

Class	application examples	lifetime aspects
Critical, coverage	Forest fire detection, flood detection, nuclear/chemical/biological attack detection, battlefield surveillance, intrusion detection	$c_{ca}/c_{ct}/c_{cb}$, c_{ln}, c_{la}, c_{lo}
Critical, no coverage	Monitoring human physiological data, military monitoring of friendly forces, machine monitoring	c_{cc}, c_{ln}, c_{la}, c_{lo}
Noncritical, coverage	Agriculture, smart buildings, habitat monitoring (sensors monitor the inhabitants in a region)	$c_{ac}/c_{tc}/c_{bc}$, c_{cc}, c_{sd}
Noncritical, no coverage	Home automation, habitat monitoring (sensors are attached to animals and monitor their health and social contacts)	c_{cc}, c_{sd}

Tabelle 1.2 – EMIT factors for a category 9 vehicle

factor		value	unit
M	vehicle mass	1.3250×10^3	kg
g	gravitational constant	9.81	$\frac{\mathrm{m}}{\mathrm{s}^2}$
ϑ	road grade	0	$^\circ$
α		1.1100	$\frac{\mathrm{g}}{\mathrm{s}}$
δ		1.9800×10^{-6}	$\frac{\mathrm{g\,s}^2}{\mathrm{m}^3}$

1.1.8 Algorithms

Based on the periodically transmitted `hello` messages, the joining node gets information about its physical neighbors and their adjacent nodes. Algorithmus 1.1 depicts the handling of `hello` messages.

Require: Locally stored state of all neighbors in set N
Ensure: Maintain neighbor set N and set virtual address
1: Receive neighbor information from node N_i
2: **if** $N_i \notin N$ **then**
3: $N \leftarrow N_i$
4: **else**
5: Update $N_i \in N$
6: **end if**
7: **if** $P == -1$ AND (Time() − OldTime) $> T_{ps}$ **then**
8: OldTime ← Time()
9: SetMyPosition()
10: **end if**

Algorithmus 1.1 – Handle `hello` messages

1.1.9 Program Code

Program code should be omitted, but if absolutely necessary, it should be set as seen in Listing 1.1.

```
1  APPLICATION("printU", 192, arg)
2  {
3      // Set Priority
4      NutThreadSetPriority(16);
5      // main() loop
6      for (;;) {
7          putchar('U');
8          NutSleep(125);
9      }
10 }
```

Listing 1.1 – Sample application

1.1.10 References

To further evaluate the applicability of our definition, we analyzed sensor network applications as surveyed in [Aky+02; ALM05; KDM05]. Concerning the importance of different lifetime criteria, most of the application scenarios can be grouped into two main classes with two sub-classes each [DD09].

1.1.11 Acronyms

Acronyms should be explained when first used. Latex helps, e.g. Mobile Ad Hoc Networks (MANETs) have been frequently used as examples for the development of Wireless Sensor Network (WSN) applications.

1.1.12 References to Data

With dataref, LaTeX provides a package to annotate data symbolically within the text. The data is declared in `data.tex` and can be used with the `\dref` macro and its companions. See the dataref documentation for further examples.

We concluded 105 experiments. 20 percent of all experiments were successful.

1.1.13 TODOs and FIXMEs

You can use the `\TODO` command to add short "sticky notes" to your document.

This is what a TODO looks like

This will also trigger generation of a list-of-TODOs at the end of the document. The same goes for the `\FIXME` command. Be careful when using `\TODO` or `\FIXME` near figures when using the gnuplot feature.

This is what a FIXME looks like

1.1.14 TikZ

TikZ ist kein Zeichenprogramm

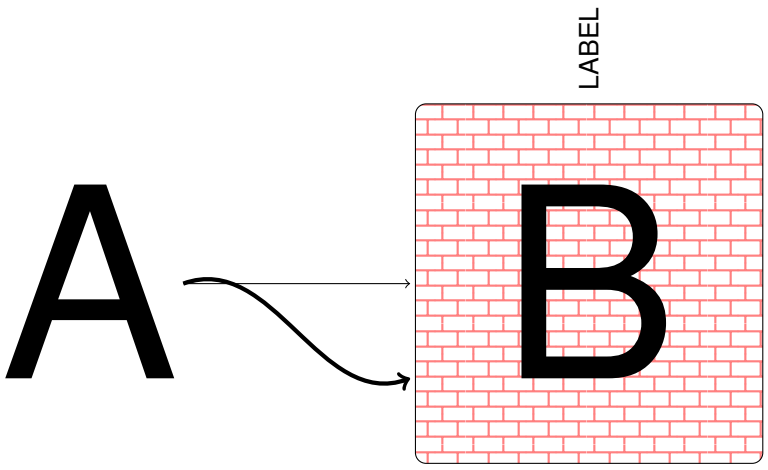


Abbildung 1.3 – This is a useful caption for a very useful TikZ picture

2 BACKGROUND

Provide the background information needed to understand your work. Goal should be that any random cs student on the campus should be able to understand your work. You don't need to explain the very basis, e.g., how a linked list works, but should explain concepts you use that are not taught in the basic courses.

- comment on employed hardware and software
- describe methods and techniques that build the basis of your work

3 DESIGN

Describe the high-level design of your system. It makes sense to start with a subchapter for the problem statement. Mention any attacker models, assumptions, and constraints of your system.

- start with a theoretical approach
- describe the developed system/algorithm/method from a high-level point of view

4 IMPLEMENTATION

Describe how you implemented your design into a real system.

- go ahead in presenting your developments in more detail

5 EVALUATION

Measurement results / analysis / discussion

- whatever you have done, you must comment it, compare it to other systems, evaluate it
- usually, adequate graphs help to show the benefits of your approach
- caution: each result/graph must be discussed! What's the reason for this peak or why have you observed this effect

6 RELATED WORK

Describe related work. Don’t just list papers, but describe what they did and how you compare to them. What is missing in their approach that you now address?

7 CONCLUSION

Conclusion: 1 page

- summarize again what your paper did, but now emphasize more the results, and comparisons
- write conclusions that can be drawn from the results found and the discussion presented in the paper
- future work (be very brief, explain what, but not much how)

ABKÜRZUNGSVERZEICHNIS

WSN	Wireless Sensor Network
MANET	Mobile Ad Hoc Network

ABBILDUNGSVERZEICHNIS

TABELLENVERZEICHNIS

QUELLCODEVERZEICHNIS

ALGORITHMENVERZEICHNIS

LITERATUR

[Aky+02] Ian F. Akyildiz u. a. „A Survey on Sensor Networks“. In: *IEEE Communications Magazine* 40.8 (Aug. 2002), S. 102–116.

[ALM05] Th. Arampatzis, J. Lygeros und S. Manesis. „A Survey of Applications of Wireless Sensors and Wireless Sensor Networks“. In: *13th Mediterrean Conference on Control and Automation*. Limassol, Cyprus, Juni 2005, S. 719–724. ISBN: 0-7803-8937-9. DOI: 10.1109/.2005.1467103.

[DD09] Isabel Dietrich und Falko Dressler. „On the Lifetime of Wireless Sensor Networks“. In: *ACM Transactions on Sensor Networks (TOSN)* 5.1 (Jan. 2009), S. 1–39. DOI: http://doi.acm.org/10.1145/1464420.1464425.

[KDM05] I. Khemapech, I. Duncan und A. Miller. „A Survey of Wireless Sensor Networks Technology“. In: *6th Annual PostGraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting*. Hrsg. von M. Merabti und R. Pereira. Liverpool, UK, Juni 2005.

OFFENE PUNKTE