

III. Verfassen eines Papiers

1. Gliederung

Überschrift

Kurzfassung
(abstract)

Einführung

Was wurde gemacht?

Was kam dabei raus?

Schluss und Ausblick

Literaturverzeichnis

(Anhang)

2. Hauptteil

2.1 Was wurde gemacht? „Versuchsaufbau“

- **Definitionen** z.B. „stark zusammenhängend“, aber nicht „Graph“
- **Problem**
- **Algorithmus**
- **Implementation** „Design“
- **kein Code**
⇒ Pseudocode, Diagramme oder verbale Beschreibung

```

static LengthType step_length()
{
    /*
        Powell-Wolfe-rule:
        G(return)>=sigma
        P(return value)<=beta
    */
    const LengthType sigma = 0.25; /* ]0,0.5[ */
    const LengthType beta = 0.5; /* ]sigma,1[ */

    const LengthType alpha = 0.5; /* ]0,1[ */
    LengthType s = step_initsize; /* ]0,infty[ start value */
    LengthType a,b,m; /* [a,b] */

    if (step_G(s)>=sigma)
    {
        if (step_P(s)<=beta)
            { return s; }

        /* increase */
        a = s;
        do
            { s /= alpha; }
        while (step_G(s)>=sigma);
        b = s;
    }
    else
    {
        /* decrease */
        a = 0;
        b = s;
    }
}

```

```

/* half */
half_loop:

m = (a+b)/2;

if (b-a<EPSILON)
{ return a; } /* returning a asserts step_P(a)>beta */

if (step_G(m)>=sigma)
  if (step_P(m)<=beta)
    { return m; }
  else
    { a = m; }
else
  { b = m; }

goto half_loop;
}

```

2. Hauptteil

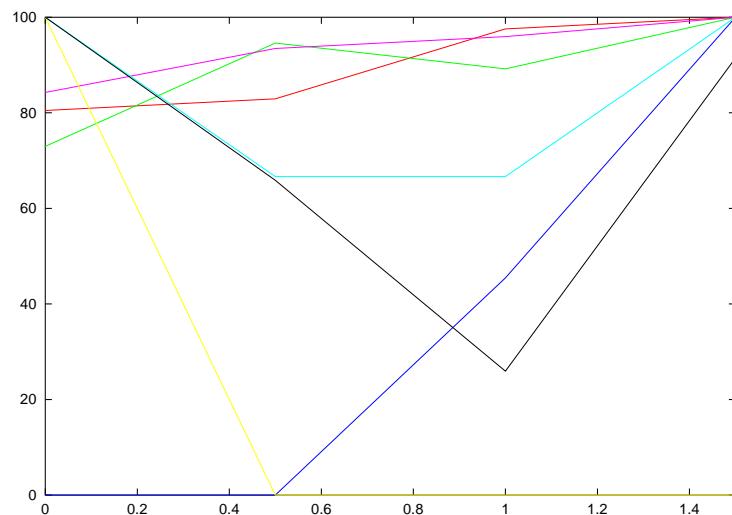
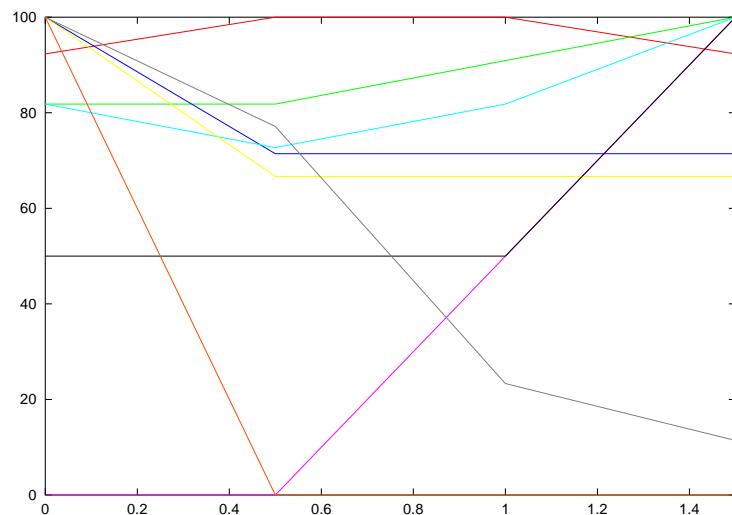
2.2 Was kam dabei raus? „Messergebnisse“

- Ergebnisse darstellen und interpretieren
- Anomalien nicht ignorieren (oder gar verheimlichen) sondern nach möglichen Erklärungen suchen

Die hohe Laufzeit für die ersten Datensätze lässt sich dadurch erklären, dass erst dort noch dynamische Bibliotheken nachgeladen werden. Diese These ließ sich dadurch erhärten, dass ...
- Nicht Rohdaten sondern aufgearbeitete Daten präsentieren.
- Bekannte statistische Methoden verwenden (z.B. Ausgleichsgerade, Mittelwert und Varianz)

2.2 Was kam dabei raus?

- Schaubilder statt Tabellen



No. ref.	neighbours pot.	weight 0.0	weight 0.5	weight 1.0	weight 1.5
1	55	1025	4	4	4
2	210	7860	0	0	0
3	676	35134	12	33	13
4	128	2304	0	0	0
5	573	23338	0	0	0
6	321	11525	18	72	18
7	823	48689	7	27	5
8	222	8669	0	0	35
9	1205	129596	0	0	0
10	1280	198055	0	0	0
11	40	567	0	0	0
12	409	20537	9	749	831
13	321	28860	3	3	9
14	446	29072	2	2	2
15	341	19085	3	13	13
16	261	17695	3	0	0
35	293	27	193	76	4
7	0	2	2	2	4
4	0	4	4	4	4
268	0	0	0	0	0

3. Einleitung

- (anschauliche) Einführung in das Thema
Wir betrachten das Problem, in einem Graphen starke Zusammenhangskomponenten zu finden. Diese sind wichtig, da ...
- Zusammenfassung des Papiers
Es wird empirisch gezeigt, dass der Algorithmus funktioniert ...
- Struktureller Überblick
In Abschnitt 2.2 stellen wir das Design unserer Implementierung vor. In Abschnitt 2.3 werden die statistischen Verfahren beschrieben, die wir verwendet haben ...

4. Kurzfassung

- wenige Sätze
- Inhalt des Papiers (incl. Problemstellung)
- schwierig

5. Überschrift

Titel

Eine Heap-basierte Implementierung
der Bestimmung
starker Zusammenhangskomponenten

Autor

Maria Musterfrau und Martin Mustermann

Ort

Winzigweich-Institut, Rotmund, Nirgendland

Datum

31. Februar 2000

5. Schluss und Ausblick

- Gesamtergebnis

Wir haben gesehen, dass wir das TSP in linearer Laufzeit lösen können ...

- Offene Fragestellungen

Bisher konnten wir noch nicht klären, ob sich dieses Ergebnis auf alle NP-vollständigen Probleme erweitern lässt.

6. Literaturverzeichnis

- [1] K. Weihe and T. Willhalm. Reconstructing the Topology of a CAD model – a discrete approach. *Algorithmica*, Volume 26, Issue 1, p. 126-147, 2000
- [2] K. Weihe, U. Brandes, A. Liebers, M. Müller-Hannemann, D. Wagner, T. Willhalm. Empirical Design of Geometric Algorithms. *Proceedings of the Fifteenth Annual ACM Symposium on Computational Geometry*, SCG 99, Miami Beach, Florida, p. 86-94.
- [3] K. Weihe and T. Willhalm. Why CAD Data Repair Requires Discrete Algorithmic Techniques. *2nd Workshop on Algorithm Engineering WAE 1998*, Saarbrücken, Germany.
<http://www.mpi-sb.mpg.de/~wae98/PROCEEDINGS/>

5. Literaturverzeichnis

- Autor(en)
- Titel des Artikels
- Titel der Konferenz, der Zeitschrift oder des Buches
- Seitenzahlen
- Verlag
- Erscheinungsjahr
- (Url)

schlecht: x ist daher immer größer Null.

besser: Der Wert von x ist daher immer größer Null.

⇒ Keine Variablen am Satzanfang

schlecht: Since $u \in U$, u is blue.

besser: Since the node u is in U , it is blue.

⇒ Nie zwei Variablen direkt hintereinander

schlecht: Man nennt einen Knoten $v \in V$ *isoliert*, wenn v keine Nachbarn hat.

besser: Man nennt einen Knoten *isoliert*, wenn er keine Nachbarn hat.

⇒ Keine unnötigen Variablennamen

schlecht: Seien v_1, \dots, v_n mit $v_i \subset V$ für alle $i = 1, \dots, n$ und $v_i \cap v_j = \emptyset$ für $i \neq j$, $i = 1, \dots, n$ und $j = 1, \dots, n$.

besser: Seien v_1, \dots, v_n paarweise disjunkte Teilmengen von V .

⇒ Nicht durch Formalismen verkomplizieren

schlecht:

Sei $(x_i | i \in \mathbb{N})$ eine Folge mit $x_i \in \mathbb{R} \forall i \in \mathbb{N}$.

besser: Sei $(x_i | i \in \mathbb{N})$ eine Folge mit $x_i \in \mathbb{R}$ für alle $i \in \mathbb{N}$.

⇒ Nicht zwischen Fließtextmodus und Mathe-modus hin- und herspringen

noch besser: Sei $(x_i | i \in \mathbb{N})$ eine reelle Folge.

Aufgabe:

- Papier schreiben
- \LaTeX verwenden
- deutsch
- 8-12 Seiten
- Style-File kommt nächste Woche
- Quell-Code in ein jar-File verpacken
- bis zum Beginn der vorlesungsfreien Zeit fertig werden