Topologische Dependenzgrammatik

Ralph Debusmann
Universität des Saarlandes
rade@coli.uni-sb.de

Zusammenfassung

In diesem Artikel erläutern wir eine neue Grammatiktheorie namens *topologische Dependenzgrammatik* oder kurz TDG. TDG vereint zwei traditionelle linguistische Theorien unter einem Dach. Zum einen werden syntaktische Relationen durch Dependenzbäume ähnlich denen in der *Dependenzgrammatik* ausgedrückt, zum anderen benutzt TDG die Theorie der *topologischen Felder*, um sogenannte topologische Dependenzen ausdrücken zu können. Mithilfe dieser Kombination von Theorien gelingt es, auch Sprachen mit relativ freier Wortstellung, wie z.B. das Deutsche, in den Griff zu bekommen, und das ohne allzu große Verrenkungen.

1 Einleitung

Dieser Artikel bietet eine Einführung in eine neue Grammatiktheorie mit dem Namen *topologische Dependenzgrammatik* oder kurz TDG. Teile der Theorie wurden bereits in mehreren Papieren beschrieben, aus denen besonders (Duchier & Debusmann 2001) herausticht. Einer der großen Vorteile von TDG ist, dass es auf eine Weise mathematisch axiomatisiert ist, dass man es quasi direkt in einen constraint-basierten Parser übersetzen kann. Auf diese Weise sind, unter Nutzung der Programmiersprache Mozart-Oz (*Mozart* 1998), schon mehrere Parser-Prototypen entstanden, die, trotz fehlender Optimierung, effizient arbeiten.

Eine TDG-Analyse besteht aus je zwei Baumstrukturen: einem *syntaktischen Dependenzbaum*, oder kurz *ID Baum*, und einem *topologischen Dependenzbaum*, oder kurz *LP Baum*.[1] Der ID Baum ist ein Dependenzbaum ähnlich wie in (Tesnière 1959) dessen Kanten mit syntaktischen Rollen beschriftet sind. ID Bäume sind ungeordnet (und deshalb auch nicht-projektiv), im Gegensatz zu LP Bäumen, die geordnet und projektiv sind. Der LP Baum ist durch die Theorie der topologischen Felder inspiriert. Der LP Baum ist eine flachere Version des dazugehörigen ID Baums, womit auch diskontinuierliche Konstruktionen in Sprachen mit freierer Wortstellung, wie z.B. dem Deutschen, beschrieben werden können.

[1] ID steht hier für *immediate dominance* und LP für *linear precedence*.

Der Ansatz von TDG hat Ähnlichkeiten mit anderen neueren dependenzbasierten Ansätzen, wie z.B. (Bröker 1998), (Kahane, Nasr & Rambow 1998) und (Gerdes & Kahane 2001). Die Grundidee, nämlich die Kopplung von Syntax bzw. Dependenz mit der Theorie der topologischen Felder, erinnert auch stark an den HPSG-basierten Ansatz von Andreas Kathol (Kathol 2000), der wiederum auf der Theorie von Mike Reape (z.B. Reape 1994) basiert.

Die Gliederung dieses Artikels sieht so aus: Wir beginnen mit einer Einführung in die Theorie der topologischen Felder in Abschnitt 2. Danach führen wir die Theorie der topologischen Dependenzgrammatik ein (Abschnitt 3) und zeigen dabei, wie die Theorie Verb-zweit-Sätze im Deutschen behandelt. In Abschnitt 4 blicken wir kurz in die Zukunft, um den Artikel abzurunden.

2 Die Theorie der topologischen Felder

Die Theorie der topologischen Felder hat eine lange Tradition in der deutschen Linguistik, und datiert zurück bis hin zu (Erdmann 1886) und (Herling 1821). Ein Beispiel ist der folgende Satz:

Einen Mann hat Maria geliebt. (1)

Die Theorie der topologischen Felder unterteilt (1) in vier Teile, die *Felder* genannt werden: Vorfeld, linke Satzklammer, Mittelfeld und rechte Satzklammer.[2]:

Vorfeld	linke Satzklammer	Mittelfeld	rechte Satzklammer
Einen Mann	*hat*	*Maria*	*geliebt.*

wobei das finite Verb *hat* in der linken Satzklammer und sein verbales Komplement *geliebt* in der rechten Satzklammer das nicht-verbale Material im Mittelfeld quasi *umklammern*. Das Vorfeld, links von der linken Satzklammer, kann von höchstens einer topikalisierten Konstituente belegt werden, wohingegen das Mittelfeld beliebig viele nicht-verbale Elemente enthalten kann. Die Ordnung der Elemente im Mittelfeld ist relativ freigestellt.

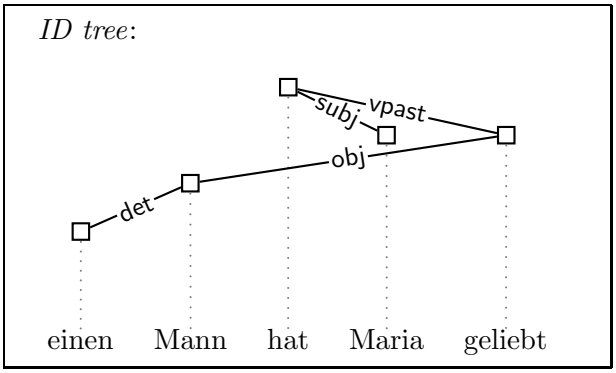
3 Topologische Dependenzgrammatik

3.1 ID und LP Bäume

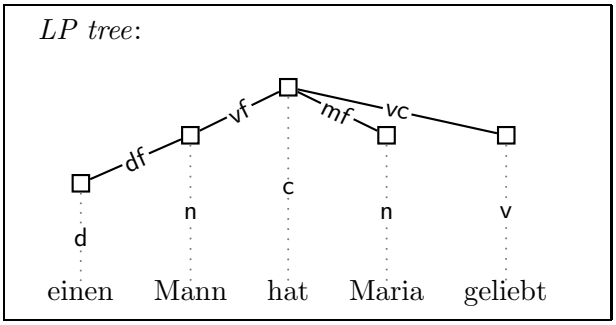
TDG unterscheidet wie schon gesagt zwei Baumstrukturen: den ungeordneten ID Baum und den partiell geordneten und projektiven LP Baum. ID und

[2]Eigentlich postuliert die Theorie noch ein weiteres Feld namens *Nachfeld* für Material rechts von der rechten Satzklammer. Für die Zwecke dieses Papiers ist dieses Feld jedoch unwichtig.

LP Bäume teilen sich dieselbe Menge von Knoten, die eins-zu-eins zu den dazugehörigen Wörtern korrespondieren. Die Menge von Kanten ist aber in ID und LP Baum verschieden. Untenstehend ist eine ID Baum-Analyse von (1). Weil ID Bäume ungeordnet sind, können wir uns eine beliebige Wortabfolge für die Grafik aussuchen. Im untenstehenden Bild bleiben wir bei der Wortabfolge in Satz (1):



Kanten in ID Bäumen sind mit syntaktischen Funktionen wie z.B. subj (für ein Nominativ-Subjekt), obj (Akkusativ-Objekt), vpast (für ein Partizip 2-Komplement) und det (Artikel) beschriftet. Die Mutter eines Knotens im ID Baum nennen wir *syntaktischer Kopf* und die Töchter *syntaktische Dependenten*. Hier ist die entsprechende LP Baum-Analyse:



Die Mutter eines Knotens im LP Baums nennen wir *topologischer Kopf* und die Töchter *topologische Dependenten*.

3.2 Ordnung von Wörtern im LP Baum

Um die durch die Theorie lizensierten Wortabfolgen zu bestimmen, benutzt TDG eine Menge $\mathcal{F}$ von Feldern. $\mathcal{F} = \mathcal{F}_{\text{ext}} \uplus \mathcal{F}_{\text{int}}$, wobei $\mathcal{F}_{\text{ext}} = \{\mathsf{df}, \mathsf{vf}, \mathsf{mf}, \mathsf{vc}\}$ die Menge von *externen Feldern* oder LP Kantenbeschriftungen ist. $\mathcal{F}_{\text{int}} = \{\mathsf{d}, \mathsf{n}, \mathsf{c}, \mathsf{v}\}$ ist die Menge von *internen Feldern* oder LP Knotenbeschriftungen.[3] $\mathcal{F}$ ist total geordnet, was für eine partielle Ordnung der LP Bäume sorgt:

[3]Natürlich enthält $\mathcal{F}$ nicht alle Felder, die für eine vollständige Abdeckung des Deutschen nötig wären, sondern nur einen kleinen Teil, um die Erklärungen zu erleichtern.

1. Die topologischen Dependenten jedes Knotens werden durch deren Kantenbeschriftungen (externe Felder) geordnet

2. Jeder Knoten wird in Relation zu seinen topologischen Dependenten durch seine Knotenbeschriftung (internes Feld) geordnet.

Die sich ergebende Ordnung ist partiell und nicht total: wenn zwei Wörter in demselben externen Feld landen[4], bleiben sie gegenseitig ungeordnet. Damit erklärt TDG u.a. die relativ frei Wortstellung im deutschen Mittelfeld: alle Elemente, die in demselben Feld mf landen, sind gegenseitig ungeordnet.

Die Menge $\mathcal{F}$ von Feldern ist durch die Theorie der topologischen Felder motiviert. So modelliert vf das Vorfeld, c die linke Satzklammer, mf das Mittelfeld und vc die rechte Satzklammer (die Abkürzung vc steht eigentlich für *verb cluster*). df und n werden benutzt, um die Wortstellung innerhalb von Nominalphrasen einzuschränken: so steht df für *determiner field* (Artikel-Feld) und n für *noun field* (Nomen-Feld). Die totale Ordnung von $\mathcal{F}$ sieht nun wie folgt aus:

$$\mathsf{d} \prec \mathsf{df} \prec \mathsf{n} \prec \mathsf{vf} \prec \mathsf{c} \prec \mathsf{mf} \prec \mathsf{vc} \prec \mathsf{v} \tag{2}$$

Diese globale Ordnung kann in lokale Ordnungen dekomponiert werden: so verlangt die lokale Ordnung $\mathsf{df} \prec \mathsf{n}$, dass Artikel vor Nomen in derselben Nominalphrase stehen müssen. Die Abfolge $\mathsf{vf} \prec \mathsf{c} \prec \mathsf{mf} \prec \mathsf{vc}$ wiederum verlangt, dass das Vorfeld vor der linken Satzklammer vor dem Mittelfeld vor der rechten Satzklammer stehen muss.

In unserem Beispiel wird die gewünschte Wortabfolge folgendermaßen erreicht:

1. *Mann* landet im vf, *Maria* im mf und *geliebt* im vc von *hat*. Durch $\mathsf{vf} \prec \mathsf{mf} \prec \mathsf{vc}$ in (2) muss *Mann* vor *Maria* und *Maria* vor *geliebt* stehen.

2. Das interne Feld von *hat* ist c. Durch $\mathsf{vf} \prec \mathsf{c} \prec \mathsf{mf}$ in (2) muss *hat* zwischen *Mann* im vf und *Maria* im mf stehen.

3.3 Beispiellexikon

TDG beschreibt die Wohlgeformtheits-Bedinungen für ID und LP Bäume in einer lexikalisierten Art und Weise: ein Lexikoneintrag schreibt vor, welche externen Felder das Wort für potenzielle topologische Dependenten *anbietet* und welche externen Felder es selbst *akzeptiert*. Ein Knoten w' kann in einem externen Feld f eines topologischen Kopfes w nur dann landen, wenn w f anbietet und w' f akzeptiert. Ein Lexikoneintrag ordet außerdem jedem Wort

[4]Ein Knoten *landet* in einem externen Feld f gdw seine eingehende Kante mit f beschriftet ist.

eine Menge von möglichen internen Feldern zu. Hier sind die Lexikoneinträge für unser Beispiel:[5]

	bietet an	akzeptiert	internes Feld
einen	{}	{df}	{d}
Mann	{df}	{vf, mf}	{n}
Maria	{}	{vf, mf}	{n}
hat	{vf?, mf∗, vc?}	{}	{c}
geliebt	{}	{vc}	{v}

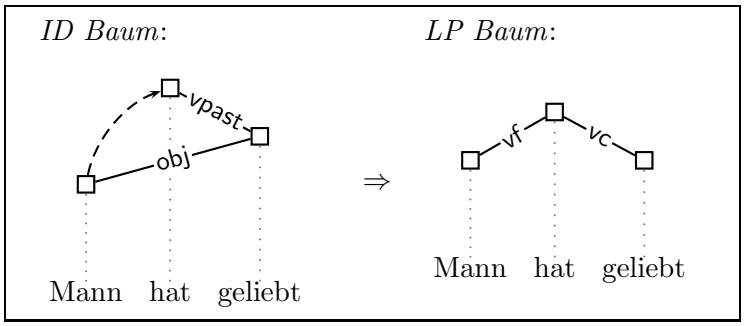
Die Menge von angebotenen externen Feldern ist in Joker-Schreibweise gegeben: vf? bedeutet z.B., dass es höchstens einen topologischen Dependenten im vf geben darf (wie von der Theorie der topologischen Felder vorgeschrieben), und mf∗ dass eine beliebige Anzahl von topologischen Dependenten im mf landen kann. Eine Feldangabe ohne ? oder ∗ bedeutet, dass dieses Feld von genau einem topologischen Dependenten belegt werden muss.

3.4 Klettern

Die Wohlgeformtheits-Bedingungen für LP Bäume werden neben den Bedingungen, die aus dem Lexikon abgeleitet werden, durch *grammatikalische Prinzipien* ergänzt. In diesem Artikel beschreiben wir allerdings nur das erste von dreien in (Duchier & Debusmann 2001):

Prinzip 1 *Ein Knoten muss auf einem transitiven Kopf landen.*

Das Prinzip sagt aus, dass der topologische Kopf w eines Knotens w' im LP Baum *über* w' im ID Baum sein muss. Wenn ein Knoten über seinem syntaktischen Kopf landet, sagen wir, dass er *geklettert* ist. In der untenstehenden Abbildung illustrieren wir, wie *Mann* ins Feld vf von *hat* klettert (angedeutet durch den gestrichelten Pfeil):



Mann muss unbedingt klettern, weil es die Wohlgeformtheits-Bedingungen aus dem Lexikon so verlangen: *Mann* kann kein topologischer Dependent

[5]Wir zeigen nur den Teil des Lexikons, der die LP Bäume betrifft. Vollständige Lexikoneinträge beinhalten noch dazu ID Baum-Merkmale wie z.B. Subkategorisierungs- und Kongruenzinformation.

von *geliebt* sein, weil *geliebt* im Lexikon kein einziges Feld für topologische Dependenten anbietet.

4 Ausblick

Mithilfe der beiden Theorien der Dependenzgrammatik (zur Syntaxbeschreibung) und der Theorie der topologischen Felder versucht TDG, vielleicht irgendwann einmal eine Alternative zu bestehenden etablierten Grammatikformalismen wie LFG (Bresnan & Kaplan 1982) oder auch HPSG (Pollard & Sag 1994) zu werden. Die Vorteile der Theorie liegen (bisher zumindest) in deren Einfachheit, und vor allem auch in der vollständigen mathematischen Axiomatisierung, die direkt zu effizienten Parser-Implementationen führen kann und auch schon geführt hat.

Ein großes Manko ist allerdings bisher die fehlende linguistische Abdeckung: hier sind HPSG-basierte Theorien z.B. von Müller (1999) oder auch Kathol (2000) viel viel weiter. Allerdings plagt diese Theorien der überkomplizierte HPSG-Apparat, der das Verständnis der Theorien vor allem en detail arg erschwert.

TDG kann nur wachsen, wenn möglichst viele Leute die Theorie verstehen und damit arbeiten. Einen Prototyp samt Grammatikformalismus gibt es bereits, d.h. TDG-Grammatiken können ohne Kenntnis der Programmiersprache Mozart-Oz ziemlich einfach eingegeben und ausprobiert werden. Wer Interesse hat: zögert nicht und fordert den Prototypen unter der obengenannten Email-Adresse vom Autor an.

Literatur

Bresnan, J. & Kaplan, R. (1982), Lexical-functional grammar: A formal system for grammatical representation, *in* J. Bresnan, ed., 'The Mental Representation of Grammatical Relations', The MIT Press, Cambridge/MA, pp. 173–281.

Bröker, N. (1998), Separating surface order and syntactic relations in a dependency grammar, *in* 'COLING-ACL 98 - Proc. of the 17th Intl. Conf. on Computational Linguistics and 36th Annual Meeting of the ACL.', Montreal/CAN.

Duchier, D. & Debusmann, R. (2001), Topological dependency trees: A constraint-based account of linear precedence, *in* '39th Annual Meeting of the Association for Computational Linguistics (ACL 2001)', Toulouse/FRA. To appear.

Erdmann, O. (1886), *Grundzüge der deutschen Syntax nach ihrer geschichtlichen Entwicklung dargestellt*, Erste Abteilung, Stuttgart/FRG.

Gerdes, K. & Kahane, S. (2001), Word order in german: A formal dependency grammar using a topological hierarchy, *in* '39th Annual Meeting of the Association for Computational Linguistics (ACL 2001)', Toulouse/FRA. To appear.

Herling, S. (1821), 'Über die Topik der deutschen Sprache'.

Kahane, S., Nasr, A. & Rambow, O. (1998), Pseudo-projectivity: a polynomially parsable non-projective dependency grammar, *in* '36th Annual Meeting of the Association for Computational Linguistics (ACL 1998)', Montreal/CAN.

Kathol, A. (2000), *Linear Syntax*, Oxford University Press.

Mozart (1998). http://www.mozart-oz.org/.

Müller, S. (1999), *Deutsche Syntax deklarativ. Head-Driven Phrase Structure Grammar für das Deutsche*, Linguistische Arbeiten 394, Max Niemeyer Verlag, Tübingen/FRG.

Pollard, C. & Sag, I. (1994), *Head-Driven Phrase Structure Grammar*, University of Chicago Press, Chicago.

Reape, M. (1994), Domain union and word order variation in german, *in* J. Nerbonne, K. Netter & C. Pollard, eds, 'German in Head-Driven Phrase Structure Grammar', CSLI, Stanford/CA, pp. 151–197.

Tesnière, L. (1959), *Eléments de Syntaxe Structurale*, Klincksiek, Paris/FRA.