9

A Morphological Analyzer for Verbal Aspect in American Sign Language

AARON SHIELD AND JASON BALDRIDGE

1 Introduction

The study of phonology and morphology involves breaking down linguistic signs into successively smaller units (e.g. distinctive features, phonemes, and morphemes), examining how those units influence each other in context, and developing systems that account for the sound changes found universally in spoken languages. Despite the plethora of approaches which have been proposed, work in computational linguistics has shown that all known phonological and morphological processes – from simple concatenative processes to templatic and reduplicative morphology - can be treated as regular relations definable in terms of regular expressions (Beesley & Karttunen, 2003). Different theories organize the information flow in quite different manners, but the solutions they provide can all be encoded in a finite-state manner. This has the tremendous upshot that very efficient morphological analyzers can be produced by compiling a set of regular expressions into a finite-state tranducer, using tools such as the Xerox Finite State Toolkit (**xfst**). These transducers are *bi-directional*, and so can be used both for generation and analysis of word forms.

Signed languages do not have an auditory component, but they too exhibit phonological and morphological processes (see Sandler & Lillo-Martin 2006 for a review of much of the literature). While morphological

Texas Linguistics Society 10: Computational Linguistics for Less-Studied Languages.
Nicholas Gaylord, Stephen Hilderbrand, Heeyoung Lyu, Alexis Palmer and Elias Ponvert eds.

analyzers have been built for a wide variety of languages and language types, including most European languages, Turkish, Arabic, Korean, and Japanese (Karttunen, 2003), we are not aware of any for sign languages. Computationally oriented work aimed at creating computer characters capable of signing and/or providing machine translation for sign languages (Veale et. al., 1998; Speers 2001; Sáfár, É. and Marshall, I, 2002; Huenerfauth, 2006) has led to the development of feature-based representations of signs and the creation of computational syntactic grammars and other capabilities, but it has not utilized finite-state methods for handling sign language morphotactics. In general, sign languages have received little attention in computational linguistics.

In this paper, we present a morphological analyzer for American Sign Language (ASL) verbs that mediates underlying lemmas to abstract formal representations of their visual surface realizations. We focus on the morphological effects of aspectual distinctions and handle co-articulation constraints that rescue otherwise unpronounceable forms. Our solution is based on a series of transducers that are composed together in a cascade.

2 The Linguistics of American Sign Language

Signed languages have been an object of linguistic inquiry since the early 1960's, when William Stokoe (1960, 1965) first formalized the notion of a sign as a linguistic unit with internal structure. Previously, signed languages were considered to be rudimentary systems of pantomime having little in common with spoken languages. Stokoe's work established ASL and other signed languages around the world as complex, natural human languages.

After Stokoe, linguists sought to analyze the formal properties of sign languages. While much progress has been made, particularly in the areas of acquisition and syntax, sign linguistics has proved very challenging in other areas, particularly phonology and morpho-phonology. Since the modality of signed languages is visual-spatial, while that of spoken languages is auditory, the very nature of the two "phonological" systems is quite different. Signed languages are thought to reflect universals of human language, but they also differ in important ways from spoken languages. For example, signed languages tend to exhibit simultaneous (rather than sequential) morphology. Also, signed languages exhibit a high degree of iconicity, *contra* the classic Saussurian notion that linguistic symbols are arbitrary.

One problem that linguists (and others) have struggled with in the effort to formalize and understand sign language structure is the lack of a writing system. Various notations have been developed such as Stokoe notation and SignWriting (Sutton, 1974), but no one has succeeded in spreading a conventional system for representing signs in writing. This fact has the unfortunate consequence that signs are most often represented by analogous words in the ambient spoken language. Thus, a given sign is typically represented by the spoken language word that most closely fits its meaning, written in capital letters to differentiate it from the spoken word itself. For example, the sign for "dog" in ASL is transcribed as DOG; that particular combination of linguistic symbols (D-O-G) has no connection to the actual form of the sign, which is produced by the rubbing of thumb and third finger of the dominant hand in neutral space, palm facing out from the signer. This writing convention thus misrepresents the relationship between signifier and signified: the ASL sign DOG does not refer to the English word *dog*: both DOG and *dog* refer to the same concept, but the two linguistic symbols do not have a direct relationship.

3 Representing Signs Formally

Although spoken languages have simultaneous dimensions such as phonetic articulation, stress and intonation contours, the most important properties can be abstractly characterized with sequential written formats. Thus, when creating a morphological transducer, the usual task is to mediate between abstract underlying forms (the lemmas) and surface forms that are are phonetic transcriptions or conventionalized written words. For example, an English transducer might map underlying forms such as *leaf+Noun+Plural* and *leave+Verb+3rdSingular* to the surface form *leaves*. In **xfst**, regular expressions map abstract features to surface realizations (such as *+Plural* to the string *s*) and handle sound alternations (such as ensuring that the output form is *leaves* and not *leafs*). Though it is tempting to view such transducers as enacting a sequence of rules, they are actually the result of composing a series of transducers (each defined by an individual regular expression) into a *single* transducer that implements a regular relation. Because of this property, finite-state morphological transducers are bi-directional and can thus produce the lemma representations from the surface forms as well as producing surface forms from lemmas (Beesley & Karttunen, 2003).

Because sign language has little in the way of sequential morphology, an even more abstract formal representation than transcriptions or written words is necessary. Each parameter of the sign must be represented with a shorthand system that is fairly transparent and comprehensible. Various phonological models of how signs are composed have been proposed, including the Move-Hold model (Liddell & Johnson 1989) and the Hand Tier model (Sandler 1989). These models differ in the way they represent abstract sign features and architecture. We do not commit to a particular underlying phonological form of the sign, but acknowledge that basic parameters must be represented in order to be able to recover sign forms from a text format. We therefore specify the parameters of sign type, handshape, location, palm orientation and (simplified) movement. The following schema represents the basic parameters that combine to form signs:

- *Types*: 1-Handed (1H), 2-Handed Symmetrical (2HS), 2-Handed Dominant (2HD)
- *Handshapes*: A, B, C, 5, E, F, G, H, 3, O, R, V, W, X, Y, 8
- *Locations*: face, neutral, torso, neck, shoulders, chest, trunk, upper arm, elbow, forearm, wrist
- *Palm orientations*: up, down, out, in, base
- *Movements*: touch, twist, reduplication, arc, slow (all +/- values)

Signs can be one-handed, two-handed symmetrical (in which both hands form the same shape and make the same movement), or two-handed dominant (one hand is "dominant" - the non-dominant hand has a limited number of possible handshapes and exhibits no independent motion). For the sake of uniformity, our analysis specifies every sign as having a dominant hand (DH) shape and a non-dominant hand (NDH) shape.

The locations represent all of the possible contrastive locations for signs in ASL. Handshapes are represented by the corresponding ASL number or letter. Palm orientation refers to the way the palm of the signer's hand faces: up, down, base (i.e., the way the palms face while hanging at rest), outwards from the signer, or inwards toward the signer. Finally, movement has been simplified to five essential distinctive features: touch (whether or not there is contact between the two hands during articulation of the sign), twist (whether or not the articulating hand reverses orientation during the performance of the sign), reduplication (whether or not the sign is iterated more than once), arc (whether or not the sign follows a path through space), and slow (whether or not the articulation of the sign is produced at a rate slower than normal). Touch, reduplication, and arc are all well-attested in the literature; twist and slow are novel features which we have found useful in characterizing certain phonological phenomena. It should be noted that this characterization of movement is simplified, specifying only the bare bones of movement necessary for producing forms. However, we believe that this description adequately captures the morpho-phonological problems we address.

Using this notation, the verb SEE is represented as follows:

```
<Type:1H DH:Vin NDH:none Loc:face -Touch -Twist -Redup +Arc -
Slow>
```

The goal of our analyzer is to produce such representations from underlying word signs like SEE, and vice versa.

4 ASL Verbal Morphology

Although ASL lacks morphological tense, it has a complex system of inflectional morphology to show aspect. The kind of *movement* exhibited by the verb changes depending on the type of aspect, while handshape, location, and palm orientation (typically) remain the same. Klima & Bellugi (1979) individuated the following aspectual distinctions in ASL: protractive, incessant, habitual, continuative, iterative, facilitative, inceptive, and augmentative. In this paper, we consider two of the most common aspectual inflections -- habitual and continuative.

The underlying forms of our transducer are taken to be the sign plus aspect. For example, for the verb STUDY, our lexicon will contain the following entries: (a) `STUDY Aspect:None`, (b) `STUDY Aspect:Hab`, and (c) `STUDY Aspect:Cont` (for no aspect, habitual aspect, and continuative aspect, respectively). We create this lexicon (here, of five different ASL verbs) in the standard way that such distinctions are produced for spoken language morphology – through rules of word formation (Karttunen, 2006):

```
define WordSigns [COOK | FORCE | PLAY | SEE | STUDY ];

define AspectFeature [Aspect ":" [None | Hab | Cont ] ] ;

define Lexicon WordSigns " " AspectFeature ;
```

These regular expressions create a lexicon with 15 entries – three for each verb. We follow Karttunen's notation for encoding Realizational Morphology in **xfst** (Karttunen, 2003), which is convenient for a feature-based representation such as ours. Note that certain characters in our representation, such as the colon and whitespace and others used below, are operators in **xfst**. They thus need to be surrounded by double quotes in order to be used as literal strings.

These underlying forms must be mapped to their correct surface forms, such as that given for `SEE Aspect:None` in the previous section. To do this, we start by encapsulating the base forms in the lexicon in brackets (to facilitate later processing) and then producing the base realizations for all verbs. For example, the basic form of the sign STUDY is a two-handed sign: the base hand is a 5-handshape with the palm facing up, and the active hand is a 5-handshape with the palm facing inwards[1]. The bracketing rule and the base form rule for STUDY are the following:

```
define BracketedLexicon 0:"[" Lexicon 0:"]" ;

define StudySign [. .] ->
  "<" Type ":" 2HD " " DH ":" 5in " " NDH ":" 5up " "
  Loc ":" neutral " " "-" Touch " "
  "-" Twist " " "-" Redup " " "-" Arc " " "-" Slow " >" ||
$STUDY "]" _ ;
```

In words, the StudySign rule says that the empty string is replaced by the given features when preceded by a string that contains STUDY and a right bracket immediately in front of it. In essence, such rules define a secondary lexicon that retrieves the feature representation associated with the basic sign stem.

By composing the transducers from these regular expressions together, all three STUDY entries are enriched with brackets and the correct feature specifications. We define similar rewrite rules for the other five verbs. The transducer now contains elements such as:

```
[STUDY Aspect:None]<Type:2HD DH:5in NDH:5up Loc:neutral -Touch
-Twist -Redup -Arc -Slow>
```

This representation forms the basis for producing the correct surface forms for each aspectual type.

Continuative aspect indicates that a particular action happens through time and is characterized by a prolonged, lengthened path movement (`+Arc + Slow`). Habitual aspect indicates an action happening repeatedly and is characterized by a reduplicated path movement (`+Redup +Arc`). For regular verbs, habitual and continuative aspectual inflections can be represented by simply changing the movement features appropriately. When the sign STUDY is inflected for habitual aspect, both hands move repeatedly in a circle, while retaining their original handshapes, palm orientations, and locations. The addition of habitual aspect thus changes the `Redup` and `Arc` features from − to +. Continuative aspect changes only `Arc`:

```
define Continuative  "-" -> "+" || $[Aspect ":" Cont] _ Arc ;
```

[1] STUDY also contains an *internal* movement: the fingers of the active hand wiggle. We only account for path, not internal movement, in our analysis. The internal movement does not appear to change with changes in aspectual inflection.

```
define Habitual
  "-" -> "+" || $[Aspect ":" Hab] _ [Redup | Arc] ;
```

The rules both utilize **xfst**'s containment operator "$", which saves us from having to specify what else might occur in the string context preceding the replacement point. This is particularly useful with our non-sequential representations – typically, rules for spoken language morphology act in a very local fashion in which the context that makes the rule fire is string-adjacent to the change. Even in the case of vowel harmony rules that set off a cascade of vowel changes, each replacement is still locally determined (Beesley & Karttunen, 2003). Our representations are not order dependent, so adjacency is irrelevant. What is necessary instead is the ability to test whether a value exists somewhere in the string, so the "$" operator is perfect for this. Also note that the +/- notation allows both the arc and reduplication changes to be encoded with a single rule.

With these rules, we obtain the following forms for STUDY:

```
[STUDY Aspect:Hab]<Type:2HD DH:5in NDH:5up Loc:neutral -Touch
-Twist +Redup +Arc -Slow>

[STUDY Aspect:Cont]<Type:2HD DH:5in NDH:5up Loc:neutral -Touch
-Twist -Redup +Arc +Slow>
```

Not all verbal inflection can be modeled so easily; there are constraints in some configurations which we turn to next.

5 Co-articulation Constraints in ASL Verb Inflection

Several verbs show complications in their phonological form when inflected with continuative or habitual aspect -- parameters other than `Redup`, `Arc`, and `Slow` features change because some aspects of signs cannot be co-articulated. Here, we give examples for COOK, PLAY, and FORCE and show how they are handled by our transducer.

PLAY in its base form includes a `+Twist` value, indicating that palm orientation reverses during the enunciation of the sign (specifically, a forearm twist produces oscillations between `up` and `down` palm orientations):

```
<Type:2HS DH:Ybase NDH:Ybase Loc:neutral -Touch +Twist +Redup
 -Arc -Slow>
```

Adding continuative aspect changes `Arc` and `Slow` to `+` (via the `Continuative` rule). The `Arc` feature cannot be pronounced simultaneously on a two-handed symmetrical sign like PLAY, resulting in a `-Twist` value. The following rewrite rule enacts this change:

```
define NoTwistWith2HSArc
  "+" -> "-" || $[Type ":" 2HS ] _ Twist $["+" Arc] ;
```

Applying the rule gives the correct surface form for `PLAY Aspect:Cont`:

```
<Type:2HS DH:Ybase NDH:Ybase Loc:neutral -Touch -Twist +Redup
 +Arc +Slow>
```

COOK in its base form includes a `+Touch` value:

```
<Type:2HD DH:5down NDH:5up Loc:neutral +Touch +Twist[2] -Redup
 -Arc -Slow>
```

Adding habitual aspect changes both `Redup` and `Arc` to `+` (via the `Habitual` rule). The `+Arc` and `+Twist` values make it impossible to retain `+Touch`; we encode this with the following rule and obtain the correct surface form for `COOK Aspect:Hab`:

```
define NoTouchWithArcTwist
  "+" -> "-" || _ Touch [$["+" Arc] & $["+" Twist]] ;
```

```
<Type:2HD DH:5down NDH:5up Loc:neutral -Touch +Twist +Redup
 +Arc -Slow>
```

Note the use of the **xfst** intersection operator "&" in the rule. The expression `[$["+" Arc] & $["+" Twist]]` describes all strings which contain both `+Arc` and `+Twist` values. The order in which they are encoded in the representation is not important – the rule would match both "`+Twist -Redup +Arc`" and "`+Arc -Redup +Twist`". This makes the approach extensible, since the rule will continue to work even as more features are added or their order in the representation changes.

FORCE in its base form also includes a `+Touch` value:

---

[2] Note that the +twist feature only applies to the dominant hand in a two-handed dominant (2HD) sign, while it applies to both hands in a two-handed symmetrical (2HS) sign.

```
<Type:2HD DH:Cout NDH:5down Loc:neutral +Touch -Twist -Redup
 -Arc>
```

Adding habitual aspect to FORCE leads to the non-dominant hand being dropped completely. Unsurprisingly, this change also renders `+Touch` impossible. We encode this with the following rules and obtain the correct surface form:

```
define NoNDHWith5DownRedup
  5down -> none || NDH ":" _ $["+" Redup] ;
define NoTouchWoutNDH "+" -> "-" || $[NDH ":" none ] _ Touch ;

<Type:2HD DH:Cout NDH:none Loc:neutral -Touch -Twist +Redup
 +Arc -Slow>
```

Having defined the lexicon and the rules, all that remains is to compose the individual transducers together and strip off the word sign and aspectual information to obtain a final transducer that maps underlying representations to the appropriate feature representations that describe the surface visual forms themselves. Using **xfst**, we can apply down the network to get surface forms from underlying forms, and up it to do the reverse:

```
xfst[1]: apply down "PLAY Aspect:Cont"
<Type:2HS DH:Ybase NDH:Ybase Loc:neutral -Touch -Twist +Redup
 +Arc +Slow>
xfst[1]: apply up "<Type:2HD DH:Cout NDH:none Loc:neutral
 -Touch -Twist +Redup +Arc +Slow>"
FORCE Aspect:Hab
```

The feature representations on the lower side of the network could be given to another application, such as a virtual signing avatar, to pronounce it (visually). The upper side provides lemma and aspectual information, which would be useful for ASL dialog systems.

6 Conclusion

We have presented a morphological analyzer for ASL that represents base forms for several ASL verbs and handles morpho-phonological changes when continuative and habitual aspect are added to them. For some verbs, the addition of aspect leads to forms that are impossible to pronounce. These co-articulation constraints are mediated by a cascade of transducers that correct the forms appropriately. By virtue of being a finite-state transducer that implements a regular relation, our analyzer is bi-directional, and thus can be used both for generation and analysis of ASL signs. This makes it potentially useful for a number of practical applications including educational tools for learners of ASL, sign language dialog systems, and machine translation (Speers, 2001; Huenerfauth, 2006). Handling aspect is particularly relevant for educational goals – it is often difficult for (hearing) sign learners to master, yet it is an essential part of sign language grammar.

To have broader applicability, the coverage of the analyzer would need to be greatly expanded. There are several other interesting morpho-phonological alternations in ASL and other signed languages that could be represented with such an analyzer; one broadly-attested and important phenomenon is verb agreement. Many verbs in ASL change their location and directionality depending on the argument structure (so-called verb agreement). For example, the verb GIVE can inflect by moving from the spatial location of the agent to the spatial location of the recipient. Thus, the path movement of the utterance I-GIVE-YOU starts at the body of the signer and moves outwards, while YOU-GIVE-ME does the opposite. These locations could be included in the encoding of agreeing verbs in **xfst**. This could be useful, for example, with a grammar for ASL such as that described by Wright (2006, this volume), which includes lexical entries that encode such path movements.

Such an analyzer would also need to account for the sequential aspects of ASL signs – as entire sentences are uttered, the features of one sign lead to assimilation in another, much like nasal assimilation in spoken languages. Additionally, and unlike spoken languages, the epenthetic movements between individual signs are always visible, and would thus need to be represented. Our representations and rules, with their heavy use of the containment and intersection operators in **xfst** (similar to Karttunen's (2003) rules for Lingala), should extend straightforwardly to this context.

It bears noting as well that we have not accounted for facial marking morphology in our analysis. Facial markings are an important part of sign language grammar (used, e.g., in question-marking, negation, verb agreement marking, and adverbial and aspectual manner marking) and must be included in eventually more elaborate descriptions of signs. However, this fact does not present problems for our analyzer: facial markings can be represented in the form of the sign or in the rules, and alternations can be handled with transducers in **xfst**, just as manual signs can.

Our analysis could help in the understanding of the formal properties of verb morpho-phonology in ASL (and perhaps other signed languages). The encoding of the rules in **xfst** allows us to straightforwardly test their predictions on all forms, and in fact it did highlight errors in the original paper-and-pencil analysis. Additionally, the overall architecture of representations and rules may also have implications for accounts of simultaneous morphophonological phenomena in spoken languages.

The rules as defined are admittedly quite specific to each of the changes they enact. In the interest of creating a more cross-linguistically applicable analysis, it would be preferable to instead be able to state general constraints on what are pronounceable sign representations. For example, Pfau & Steinbach (2004, 2005) provide an analysis of reciprocals and plurals in German Sign Language which uses language-specific as well as general constraints in the framework of Optimality Theory (OT; Prince & Smolensky, 1993). As **xfst** is capable of implementing OT analyses (via lenient composition), we intend to explore the possibility of using such cross-linguistic constraints to create analyzers that are less language-specific than the rule-based one presented here. Should these constraints turn out to be robustly attested, an OT-based analyzer could be specified in **xfst** that could be more easily extended to handle phenomena from signed languages other than ASL.

7 Appendix

We include here the full XFST script.

```
# XFST Script for handling ASL verbal aspect.

# Define what the basic signs are -- basically underlying
# concepts communicated by the sign so that we can refer to
# them as macros to produce the actual feature descriptions of
# the signs.

define WordSigns [ COOK | FORCE | PLAY | SEE | STUDY ];
define AspectFeature [ Aspect ":" [None | Hab | Cont ] ] ;
define Lexicon WordSigns " " AspectFeature;

# Place brackets around lexical entries to facilitate later
# cleanup.

define BracketedLexicon 0:"[" Lexicon 0:"]" ;

# Map the WordSigns to their Sign descriptions.

define CookSign [. .] ->
  "<" Type ":" 2HD " " DH ":" 5down " " NDH ":" 5up " " Loc ":"
neutral " " "+" Touch " " "+" Twist " " "-" Redup " " "-" Arc "
" "-" Slow ">" || $COOK "]" _ ;

define ForceSign [. .] ->
```

```
  "<" Type ":" 2HD " " DH ":" Cout " " NDH ":" 5down " " Loc
":" neutral " " "+" Touch " " "-" Twist " " "-" Redup " " "-"
Arc " " "-" Slow ">" || $FORCE "]" _ ;

define PlaySign [. .] ->
  "<" Type ":" 2HS " " DH ":" Ybase " " NDH ":" Ybase " " Loc
":" neutral " " "-" Touch " " "+" Twist " " "+" Redup " " "-"
Arc " " "-" Slow ">" || $PLAY "]" _ ;

define SeeSign [. .] ->
  "<" Type ":" 1H " " DH ":" Vin " " NDH ":" none " " Loc ":"
face " " "-" Touch " " "-" Twist " " "-" Redup " " "+" Arc " "
" "-" Slow >" || $SEE "]" _ ;

define StudySign [. .] ->
  "<" Type ":" 2HD " " DH ":" 5in " " NDH ":" 5up " " Loc ":"
neutral " " "-" Touch " " "-" Twist " " "-" Redup " " "-" Arc "
" "-" Slow ">" || $STUDY "]" _ ;


# Compose these rules together to create a transducer that will
# map the Englishized word forms to their sign representations.

define AddSigns CookSign .o. ForceSign .o. PlaySign .o. SeeSign
.o. StudySign;

# Create rules to add morphological changes due to aspect and
# compose them.

define Habitual  "-" -> "+" || $[Aspect ":" Hab] _ [Redup|Arc];

define Continuative  "-" -> "+" || $[Aspect ":" Cont] _ [Arc |
Slow] ;

define AspectualMorph Habitual .o. Continuative ;

# Create rules for handling irregular morphology.

define NoTouchWithArcTwist "+" -> "-" || _ Touch [$["+" Arc] &
$["+" Twist]] ;

define NoTwistWith2HSArc "+" -> "-" || $[ Type ":" 2HS ] _
Twist $["+" Arc] ;

define NoNDHWith5downRedup 5down -> none || NDH ":" _ $["+"
Redup] ;

define NoTouchWoutNDH "+" -> "-" || $[NDH ":" none ] _ Touch ;

define Irreg NoTouchWithArcTwist .o. NoTwistWith2HSArc
  .o. NoNDHWith5downRedup .o. NoTouchWoutNDH ;

# Compose all morphotactics together.

define Morph AspectualMorph .o. Irreg ;
```

```
# Strip the Signs.

define StripSigns "[" ?* "]" -> 0 || _ "<";

# Compose it all together.

define ASLVerbalAspect BracketedLexicon .o. AddSigns .o. Morph
.o. StripSigns ;

# Now use it.

push ASLVerbalAspect
```

8 References

Beesley, K. and Karttunen, L. 2003. *Finite State Morphology*. CSLI Publications, Palo Alto, CA.

Huenerfauth, M. 2006. Representing Coordination and Non-coordination in American Sign Language Animations. Behaviour & Information Technology 25:4, pp. 285-295.

Karttunen, L. 2003. Computing with Realizational Morphology. In Computational Linguistics and Intelligent Text Processing, Alexander Gelbukh (ed.), Lecture Notes in Computer Science, Volume 2588, pp. 205-216, Springer Verlag, Heidelberg.

Karttunen, L. 2006. Numbers and Finnish Numerals. In: M. Suominen, A. Arppe, A. Airola, O. Heinämäki, M. Miestamo, U. Määttä, J. Niemi, K. Pitkänen and K. Sinnemäki (eds.), *A Man of Measure: Festschrift in Honour of Fred Karlson on his 60th Birthday*. Special Supplement to SKY Journal of Linguistics 19. pp. 407-421.

Klima, E. & Bellugi, U. 1979. The signs of language. Cambridge, MA: Harvard University Press.

Liddell, S. & Johnson, R.E. 1989. American Sign language: The phonological base. *Sign Language Studies* 64:195-277.

Padden, C. 1988. Interaction of morphology and syntax in ASL. Ph.D. dissertation, UC-San Diego.

Pfau, R. & Steinbach, M. 2004. Optimal reciprocals in German Sign Language. In *Sign Language & Linguistics* 6 (1).

Pfau, R. & Steinbach, M. 2005. Plural formation in German Sign Language: Constraints and strategies. In: Leuninger, H. & D. Happ (eds.), *Gebärdenspra-*

chen: Struktur, Erwerb, Verwendung (*Linguistische Berichte* Special Issue 15). Hamburg: Buske, 111-144.

Prince, A. & Smolensky, P. 1993. Optimality Theory: Constraint interaction in generative grammar.

Sáfár, É. & Marshall, I. 2002. Sign language translation via DRT and HPSG. In A. Gelbukh (Ed.) Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics, CICLing, Mexico, Lecture Notes in Computer Science 2276, pages 58-68, Springer Verlag, Mexico.

Sandler, W. 1989. Phonological representation of the sign: linearity and nonlinearity in American sign language. Dordrecht: Foris.

Sandler, W. & Lillo-Martin, D. 2006. Sign language and linguistic universals. Cambridge: Cambridge University Press.

Speers, d'A.L. 2001. Representation of American Sign Language for Machine Translation. PhD Dissertation, Dept. of Linguistics, Georgetown University.

Stokoe, W. 1960. Sign language structure: an outline of the visual communication systems of the American deaf. In *Studies in Linguistics: Occasional Papers*. Buffalo: University of Buffalo.

Stokoe, W. 1965. A dictionary of American Sign Language on linguistic principles. Washington, DC: Gallaudet University Press.

Sutton, V. 1974. SignWriting. www.signwriting.org

Veale, T., Conway, A., & Collins, B. 1998. The challenges of cross-modal translation: English-to-Sign-Language translation in the Zardoz system. Machine Translation 13:1.

Wright, T. 2006. A Combinatory Categorial Grammar of a Fragment of American Sign Language. This volume - Proceedings of the 10th Meeting of the Texas Linguistics Society. Austin, TX.