Helge Dyvik
Institutt for fonetikk og lingvistikk
Universitetet i Bergen
Sydnesplass 9
5000 Bergen

Knut Hofland
NAVFs EDB-senter for humanistisk forskning
Postboks 53
5014 Bergen-Universitet

Parsing basert på LFG: Et MIT/Xerox-system applisert på norsk

Det vi har å legge frem her idag, er ikke egne forskningsresultater, men snarere en rapport om et parsing-prosjekt for norsk som vi er i ferd med å sette i gang i Bergen, og en presentasjon av hovedtrekkene i det språkanalysesystemet prosjektet anvender. «Vi» i denne sammenheng er NAVFs EDB-senter for humanistisk forskning ved Knut Hofland, og Institutt for fonetikk og lingvistikk ved Helge Dyvik. Vi regner også med å kunne knytte flere personer til dette prosjektet i tiden som kommer.

Grunnlaget for parsing-prosjektet er det utviklingsarbeid som ble utført ved NAVFs EDB-senter av Per-Kristian Halvorsen, som hadde et forskerstipendium der frem til august 1983. Halvorsen implementerte et universelt språkanalysesystem som er utviklet i samarbeid mellom forskere ved MIT og Xerox Parc i California - bl.a. Halvorsen selv - og begynte arbeidet med å bygge opp en norsk parser innenfor rammen av dette systemet. Halvorsen er nettopp begynt i en ny stilling hos Xerox i California, og kunne derfor ikke selv være til stede her for å presentere sitt arbeid.

Det norsk-fragmentet som hittil er bygget opp, er meget begrenset. Det omfatter enkle aktive deklarative setninger som kan inneholde infinitivkomplementer kontrollert av subjekt eller objekt (f.eks. «Per lovet Kari å komme»), og et rudimentært leksikon. I det videre arbeidet vil vi i første omgang konsentrere oss om verbalsystemet, nærmere bestemt systemet av perifrastiske konstruksjoner og de modale og aspektuelle kategoriene de uttrykker, om utvidelse av leksikon, og dernest om langdistanse-avhengigheter av den typen vi finner i *hv*-spørsmål, relativsetninger og topikaliserte konstruksjoner. Siktemålet er å øke parserens dekningsgrad til å omfatte sentrale konstruksjonstyper og et større ordforråd, og etter hvert å aktivisere noen av fakultetets språkmiljøer i dette arbeidet. Analysesystemet burde ligge godt til rette for dette, som vi skal se. Vi håper også å kunne knytte arbeidet sammen med noe av den mer anvendelses-orienterte forskningen som skjer ved andre institutter ved Universitetet i Bergen. Det språkanalysesystemet som benyttes, kan karakteriseres som «lingvistvennlig». Det er utviklet under ledelse av Joan Bresnan ved MIT og Ronald M. Kaplan ved Xerox Parc. Den interne representasjonen av grammatikken er en nettverk-struktur, og parseralgoritmen bygger på Kaplans «General Syntactic Processor». Men systemet inneholder også en gramatikktolker som fritar brukeren fra å formulere sin grammatiske beskrivelse som transisjons-nettverk. Grammatiske beskrivelser kan skrives direkte inn i form av regler innenfor grammatikkmodellen «leksikalsk-funksjonell grammatikk» (LFG), og grammatikktolkeren oversetter så beskrivelsen til den mer maskin-motiverte nettverk-strukturen som parser-algoritmen refererer til. LFG er en lingvistisk motivert modell med formelle egenskaper som stort sett er kjent fra moderne lingvistisk tradisjon. Dette legger forholdene til rette for at språkforskere uten spesiell interesse for parsingteori kan knyttes til datalingvistiske prosjekter.

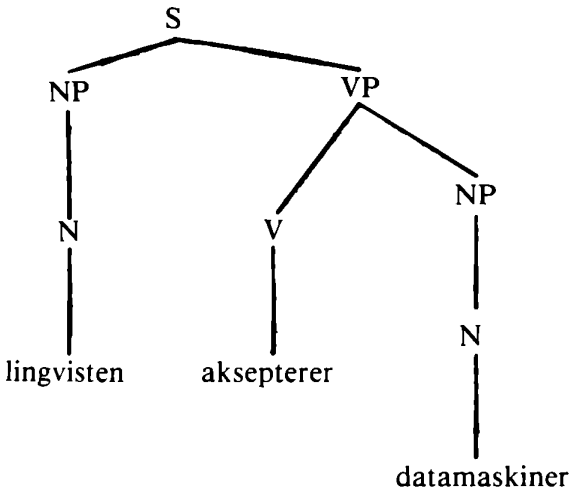
LFG er en transformasjonsfri grammatikk-modell. Modellen skiller dermed ikke mellom dypstruktur og overflatestruktur i konstituent-analysen. Fenomener som EQUI, eller PRO-kontroll i nyere versjoner av Chomskyansk syntaks - f.eks. identifikasjonen av subjektet i «Per lovet Kari å synge» som det underforståtte subjekt for infinitiven - beskrives i leksikon som en opplysning om kontrollegenskapene ved det overordnede verbet. (PRO-analysen i EST opererer riktignok heller ikke med noen transformasjon; men den antar et tomt PRO som det syntaktiske subjekt for infinitiven, noe LFG-analysen unngår.) På tilsvarende måte behandles *passiv* i leksikon som en redundansregel, som i praksis innebærer at aktiv og passiv form av samme verb blir to ulike leksikalske oppslag med identisk semantisk form, men med ulike valg av nominale ledd som argumenter. Langdistanse-avhengigheter som de vi finner i *hv*-spørsmål,

Parsing basert på LFG: Et MIT/Xerox-system applisert på norsk
Helge Dyvik, Knut Hofland
Proceedings of NODALIDA 1983, pages 62-73

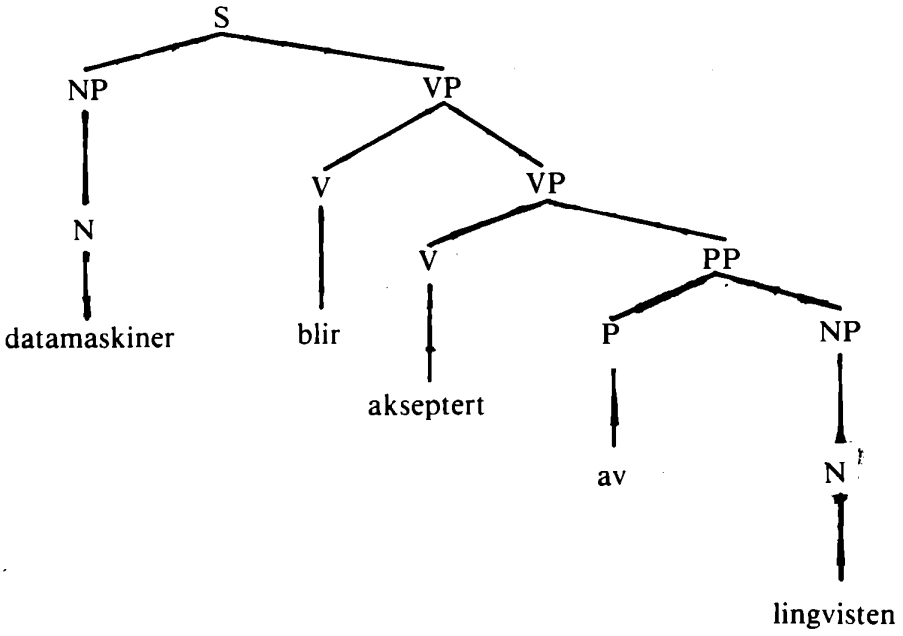
relativsetninger osv. (f.eks. «Hvem påstod Per at Kari ikke likte at han kjente?») lar seg neppe behandle leksikalsk; de ivaretas ved hjelp av en spesiell type korresponderende variabler på den kontrollerende konstituenten («hvem») og den kontrollerte tomme plassen (etter «kjente»).

Dermed kan grammatikkens kontekstfrie frasestrukturregler generere konstituentstrukturer som direkte korresponderer med den observerte streng av former. Analysesystemet tillater at grammatikken skrives direkte inn i form av slike regler. Vi får da konstituentstrukturer av vanlig type:

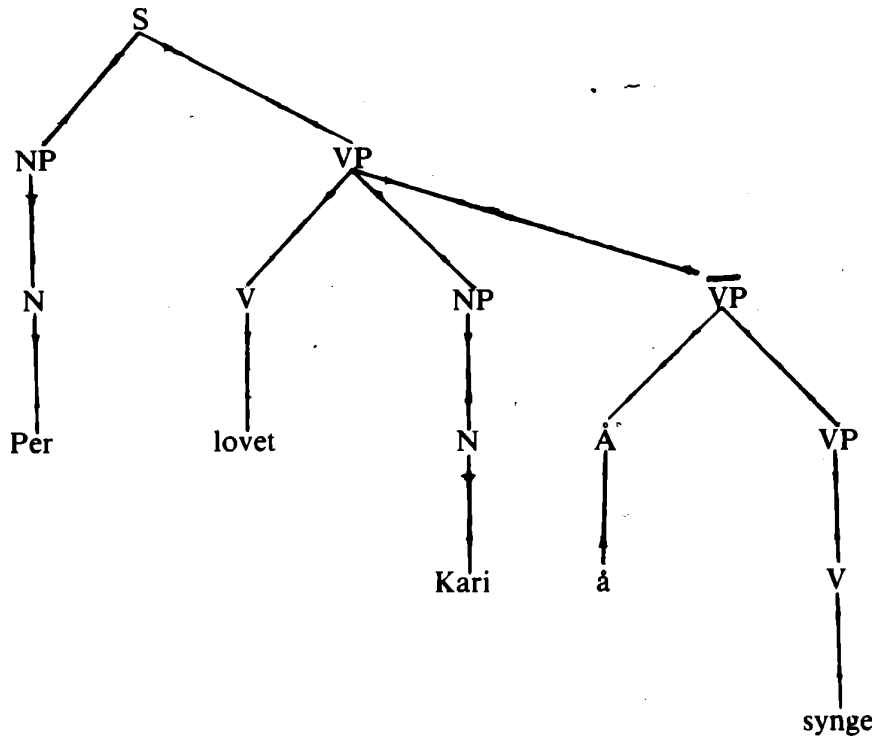
(1)



(2)



(3)

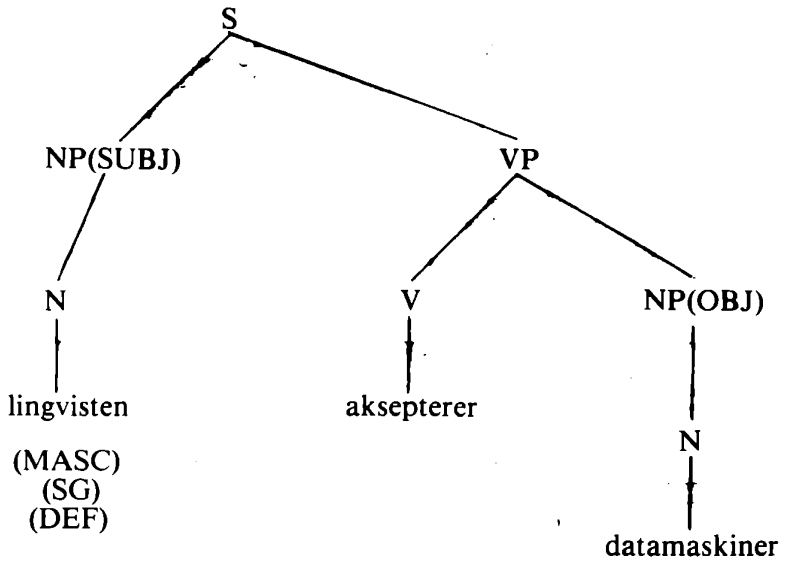


Disse strukturene er da alle generert av kontekstfrie frasestrukturregler. Det innebærer f.eks. at (1) og (2) ikke er relatert gjennom den syntaktiske derivasjonen, og at de syntaktiske reglene heller ikke relaterer «Per» i (3) til noen tom subjektplass foran «å synge». For å uttrykke disse relasjonene, og dermed få et brukbart utgangspunkt for en semantisk fortolkning, må disse enkle strukturrepresentasjonene suppleres med ytterligere uttrykksmidler. Dette er også nødvendig for å eliminere ugrammatikalske setninger som f.eks. *«Per aksepterer Kari å synge», en setning som frasestrukturreglene alene vil generere hvis de først genererer (3). Disse ytterligere uttrykksmidlene er *grammatiske funksjoner* og *grammatiske trekk*.

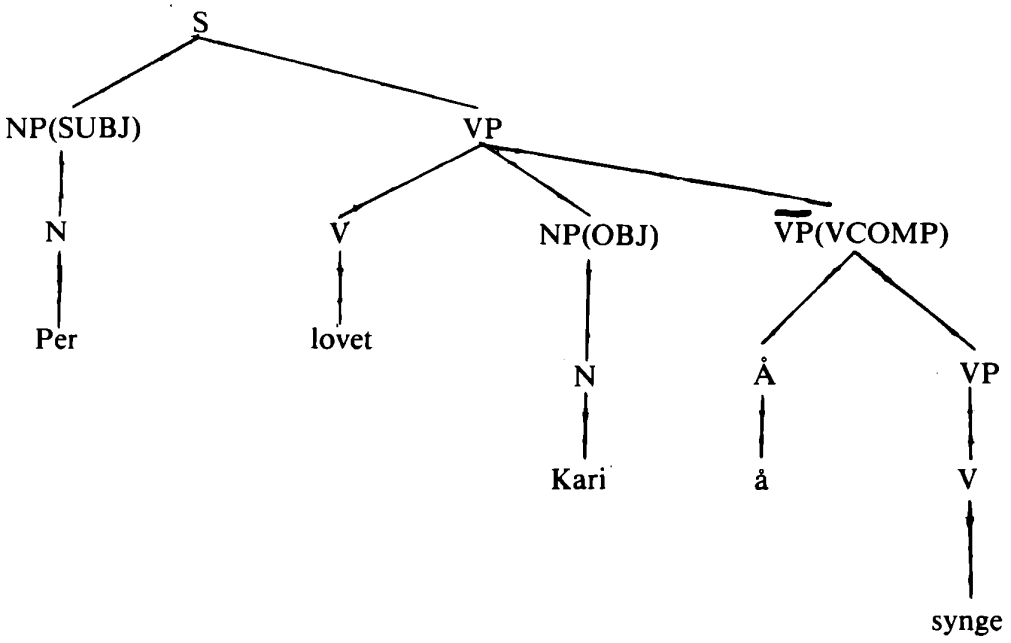
LFG behandler grammatiske funksjoner som SUBJEKT, OBJEKT osv. som primitiver, og ikke som størrelser som nødvendigvis skal være konfigurasjonelt definerbare, slik tilfellet er innenfor EST. De funksjonelle termene SUBJEKT, OBJEKT, osv. i LFG har ikke noen selvstendig interpretasjon, men fungerer bare som et grunnlag for oversettelsen fra syntaktisk til semantisk representasjon - det vil si, de bidrar til å knytte forbindelsen mellom syntaktiske konstituenter og semantiske argumentposisjoner. Dessuten har de en viktig funksjon i å filtrere bort ugrammatikalske frasestrukturer, som vi skal se.

Dels gjennom leksikon og den morfologiske analysen og dels gjennom frasestrukturreglene blir da de syntaktiske trærne supplert med funksjoner og trekk som f.eks. opplyser om at «lingvisten» er SUBJEKT i (1) mens «datamaskiner» er OBJEKT, og videre at «lingvisten» er MASKULINUM, SINGULARIS, DEFINITT. Denne informasjonen kunne vi føye inn i trærne:

(1')

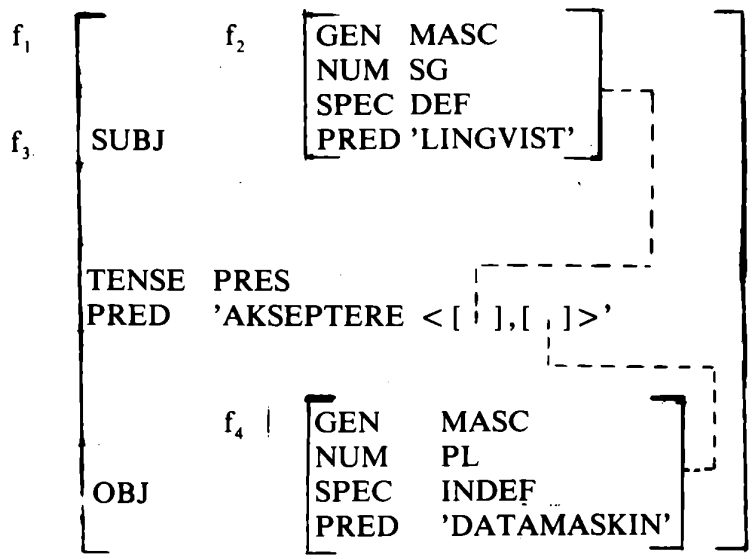

S
NP(SUBJ)
VP
N
V
NP(OBJ)
lingvisten
(MASC)
(SG)
(DEF)
aksepterer
N
datamaskiner

(3')


S
NP(SUBJ)
VP
N
V
NP(OBJ)
VP(VCOMP)
Per
lovet
N
Å
VP
Kari
å
V
synge

På grunnlag av disse funksjonene og trekkene kan man bygge opp en representasjon av setningens *funksjonelle struktur*, der man abstraherer bort fra den syntaktiske tre-konfigurasjonen. En funksjonell struktur er et hierarkisk arrangement av attributter og verdier, der attributtene er grammatiske funksjoner som SUBJEKT osv. og trekk-dimensjoner som NUMERUS osv., og verdiene er enkle symboler som SINGULARIS, semantiske former, eller nye funksjonelle strukturer med sine egne attributter. Den funksjonelle struktur til (1') kan da se slik ut:

(4)



Her har f.eks. attributtet SUBJEKT en ny funksjonell struktur som verdi, nemlig dem som tilsvarer NPen «lingvisten», og den har i sin tur sine egne attributter GEN, NUM osv., med verdier. Hvert attributt har bare en verdi. Hvis den funksjonelle strukturen benevnes f_1, tillater den hierarkiske strukturen oss da å referere konsist til dens enkelte elementer: f_1 (SUBJ) «f_1's SUBJEKT» blir da en funksjon med verdi lik den funksjonelle strukturen til «lingvisten» (f_2), og f_1's (SUBJ)(NUM) «f_1's SUBJEKTs NUMERUS» blir en funksjon med verdien SG. Attributtet PRED har en semantisk form som verdi, og den danner grunnlag for den videre oversettelse til en semantisk representasjon. For substantiver representeres verdien bare med substantivets grunnform. For verb angis argumentstrukturen og hvilke elementer i den funksjonelle strukturen som fyller argumentplassene. I diagrammet er dette angitt ved stiplede linjer fra de strukturene som fyller argumentplassen. Dette er gjort for å understreke at den funksjonelle strukturen som fyller SUBJEKT-attributtet og den som fyller første argumentplass til 'akseptere' er *en og den samme*, og ikke bare to strukturer med samme form. Det samme kunne man oppnå gjennom ko-indeksering av strukturene og argumentplassene.

Som nevnt innføres de grammatiske funksjonene og trekkene dels gjennom frasestrukturreglene og dels gjennom leksikon. I reglene skjer dette ved at de ulike konstituentene får *funksjonelle ligninger* knyttet til seg:

(5)

S → NP VP

(↑SUBJ) = ↓ (↑ = ↓)

VP → V (NP)

(↑OBJ) = ↓

I leksikon er bl.a. følgende informasjon knyttet til formen «aksepterer»:

(6)

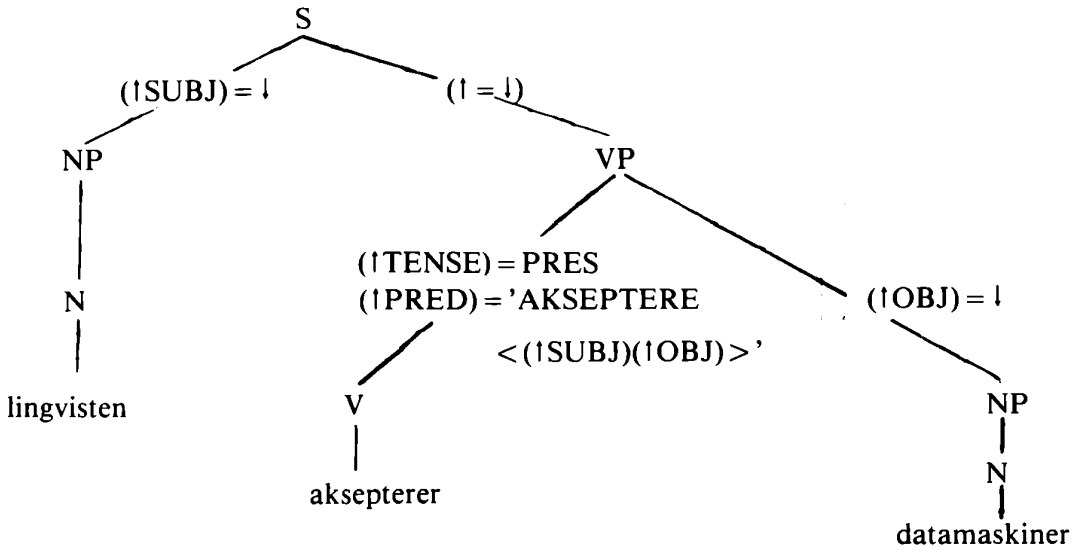
aksepterer: V

(↑TENSE) = PRES

(↑PRED) = 'AKSEPTERE < (↑SUBJ)(↑OBJ) > '

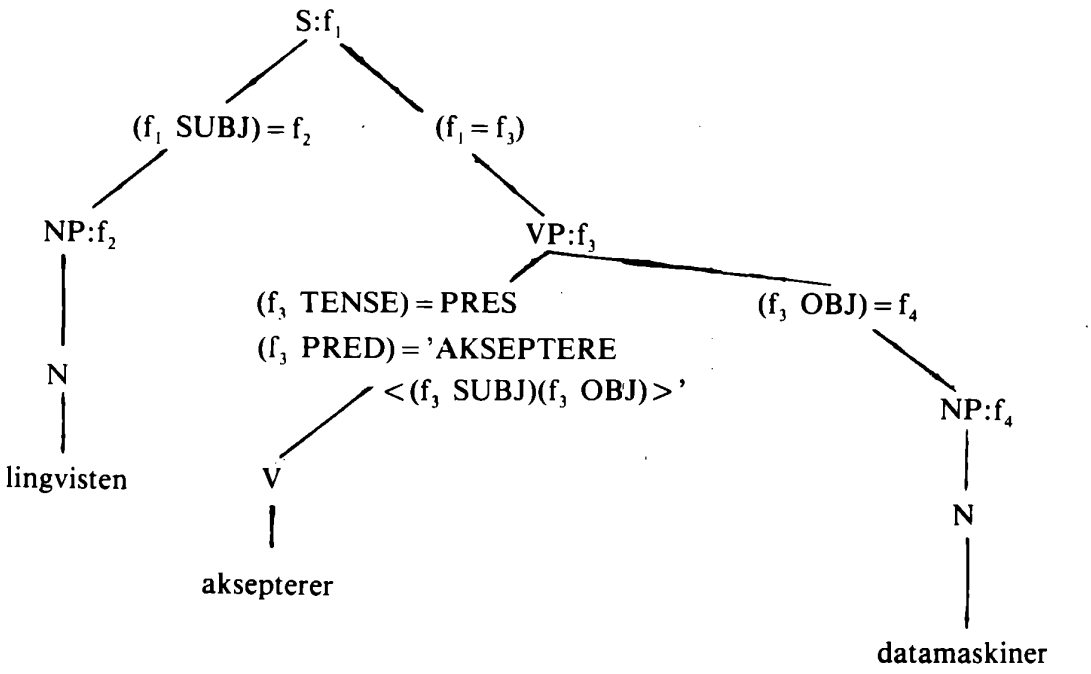
For intuitivt å forstå hva pilene innebærer, kan vi tenke oss en slik ligning plassert i treet på den grenen som fører ned til den konstituenten ligningen tilhører:

(7)



Pilene er *metavariabler* som tar som verdier indekserte variabler som blir knyttet til de enkelte knutene i treet. Piler som peker opp, får som verdi den variabelen som tilhører knuten over, og piler som peker ned, får som verdi den variabelen som tilhører knuten under. Disse variablene (f_1, f_2, f_3 osv.) tar i sin tur som verdier de funksjonelle strukturene som tilsvarer de respektive knutene. Etter at de aktuelle knutene har fått hver sin variabel, kan vi dermed erstatte pilene (metavariablene) med variabler slik at resultatet blir som følger:

(8)



«(f_1 SUBJ)» osv. kan vi lese «f_1's SUBJEKT» osv. Disse ligningene kan så løses med en funksjonell struktur som (4) som resultat. Legg merke til at ligningen under VP identifiserer den funksjonelle struktur for VP med den for S, slik at f-strukturen får færre nivåer enn treet.

På grunn av f-strukturens formelle egenskaper, og det forhold at syntaktiske regler aldri *endrer* tilordningen av funksjoner, kan de bygges opp additivt, det vil si, det spiller ingen rolle i hvilken rekkefølge ligningene løses. Dette er av betydning for løsningsalgoritmen, som da blir enklere enn den ville ha vært hvis en viss rekkefølge hadde måttet bli observert.

De funksjonelle strukturene inneholder da dels informasjon som er nødvendig ved den semantiske fortolkning, og dels sørger de for å filtrere ut som ugrammatikalske visse strukturer som tillates av frasestrukturreglene. Som et eksempel på det første kan vi betrakte hvordan subjektkontrollen i (3) («Per lovet Kari å synge») blir uttrykt. Under 'love' i leksikon finner vi blant annet disse ligningene:

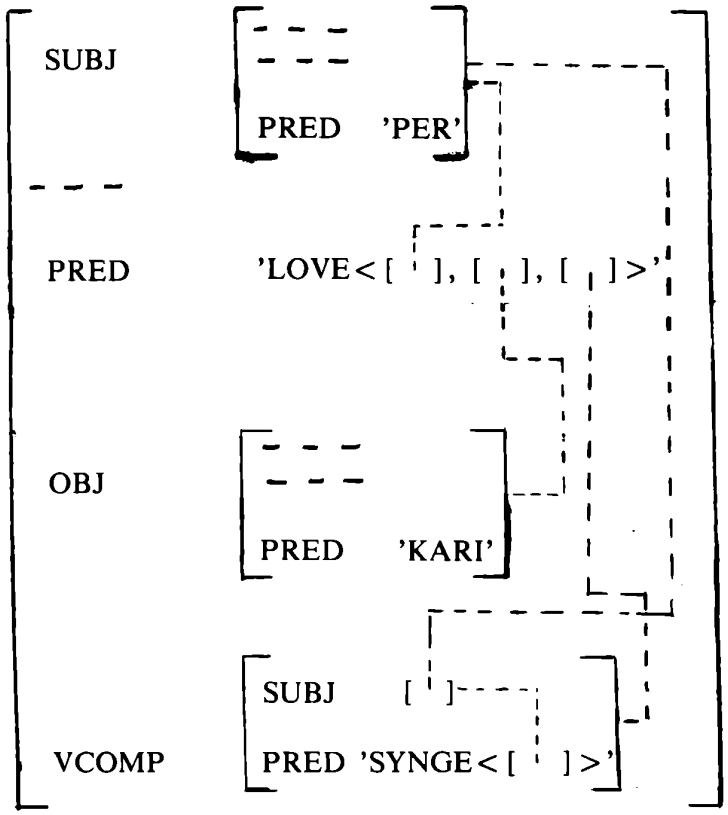
(9)

love: (↑PRED) = 'love<(↑SUBJ),(↑OBJ),(↑VCOMP)>'

(↑VCOMP SUBJ) = (↑SUBJ)

Den siste av disse ligningene stipulerer da at verbalkomplementets subjekt er identisk med setningens subjekt, på samme måte som første ligning stipulerer at første argument for 'love' er identisk med setningens subjekt. Resultatet av at denne ligningen er med, blir en funksjonell struktur som denne:

(10)



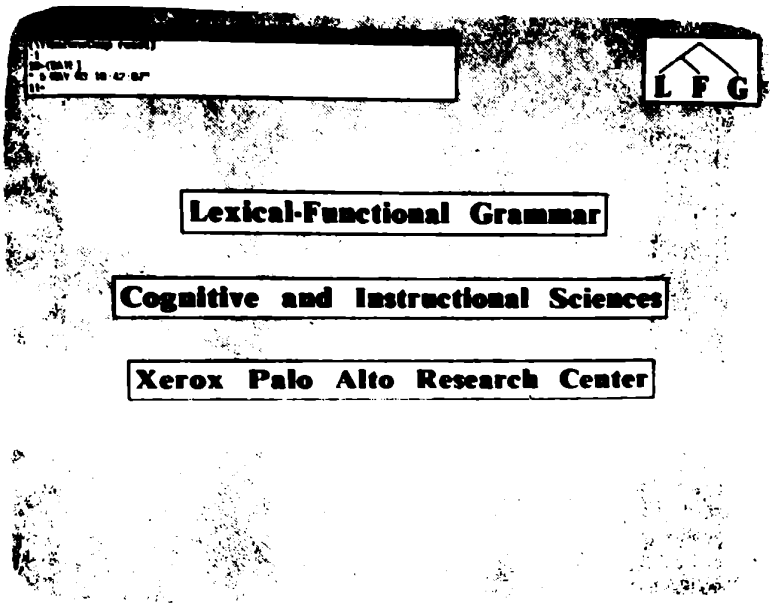
Her fremgår det at setningens subjekt 'Per' også er å oppfatte som subjekt for verbalkomplementet 'å synge', og dermed også som argument til predikatet 'synge'.

Den filtrerende effekt ser vi f.eks. hvis vi tenker oss at vi sløyfet infinitivfrasen og bare genererte «Per lovet Kari.». Da ville det ikke være noe til å fylle tredje argumentplass i den semantiske formen til 'love', og f-strukturen ville være ufullstendig. Omvendt, hvis vi skiftet verbet *lovet* ut med *akseptere* («Per aksepterte Kari å synge»), ville verbets semantiske form bare ha to argumentplasser, og funksjonen VCOMP ville ikke finne noen plass i setningens semantiske form. Resultatet ville altså være en usammenhengende f-struktur, og dette markerer setningen som ugrammatikalsk. Noe lignende gjelder behandlingen av utillatelige langdistanseavhengigheter, f.eks. at ett kontrollerende ledd korreleres med mer enn en tom plass: Frasestrukturreglene muliggjør det, men den funksjonelle strukturen filtrerer ut resultatet.

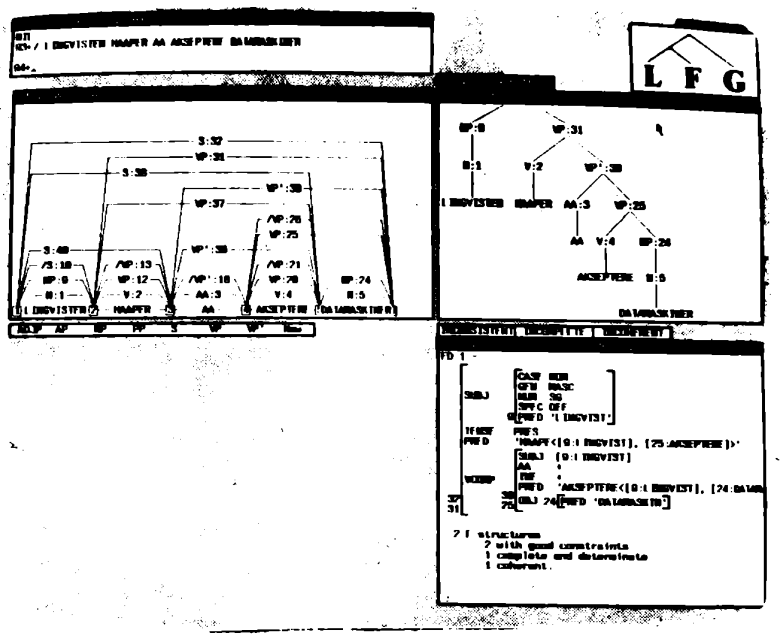
Fra et parsingsynspunkt har denne grammatikkmodellen interessante egenskaper. Ved at den funksjonelle strukturen med sine ulike avhengigheter kan bygges opp på etterskudd, så å si, på grunnlag av de funksjonelle ligningene, kan selve parsingen skje uavhengig av dem. Det innebærer at algoritmen kan forholde seg til en kontekst-fri frasestrukturgrammatikk, dvs. et rekursivt transisjonsnettverk, med de fordeler det innebærer. Resultatet er at algoritmen vil finne trær også for ugrammatikalske setninger, og

også eventuelle uakseptable alternativer som reglene måtte tillate; men disse gir da opphav til inkonsistente, ufullstendige eller usammenhengende f-strukturer, og filtreres dermed ut. En ulempe er at algoritmen da kan tenkes å bruke tid på å regne ut ubrukelige analyser, men i praksis er det naturligvis mulig å interkalere løsning av ligninger i selve parseralgoritmen, i den grad det er ønskelig. I denne sammenheng er det en fordel at man ikke er bundet til noen bestemt rekkefølge i løsningen av ligningene.

Systemet har også implementert en algoritme som oversetter f-strukturer til semantiske strukturer, dvs. en representasjon av setningen i høyere ordens intensjonal predikatslogikk. Denne algoritmen er utarbeidet av Per-Kristian Halvorsen. I de semantiske strukturene uttrykkes f.eks. mulighetene for kvantor-rekkevidde.

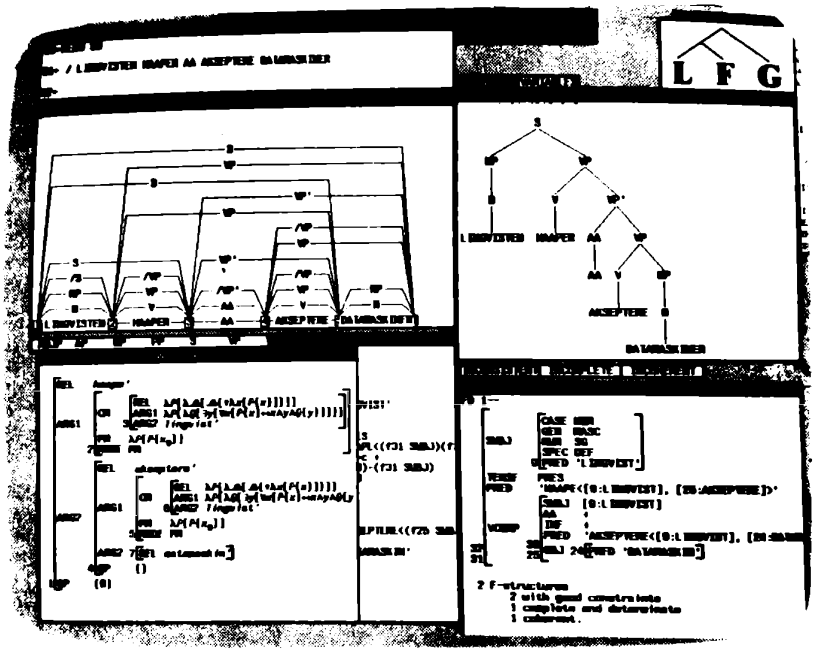


Analysesystemet er implementert i INTERLISP, først på DEC-20 ved MIT og Xerox og senere på LISP-maskiner ved Xerox. En LISP-maskin er en en-bruker maskin med stort primærlager (1.5 Mb) og 10-30 Mb masselager. Denne kan stå tilknyttet et større nett (Ethernet). Maskinen har en stor grafisk skjerm (bitmap display 1000 × 1000 punkter) og en «mus» som pekeinnretning til skjerm. Mye av interaksjonen mellom maskin og bruker skjer via denne. «Musen» har to knapper, en for å kalle frem en meny og en for å utføre en ordre som blir pekt på. Videre har systemet en avansert vinduspakke som gjør det mulig å skrive og redigere data i forskjellige områder på skjermen. Eksemplene i fortsettelsen er hentet fra kjøringer på en slik maskin.



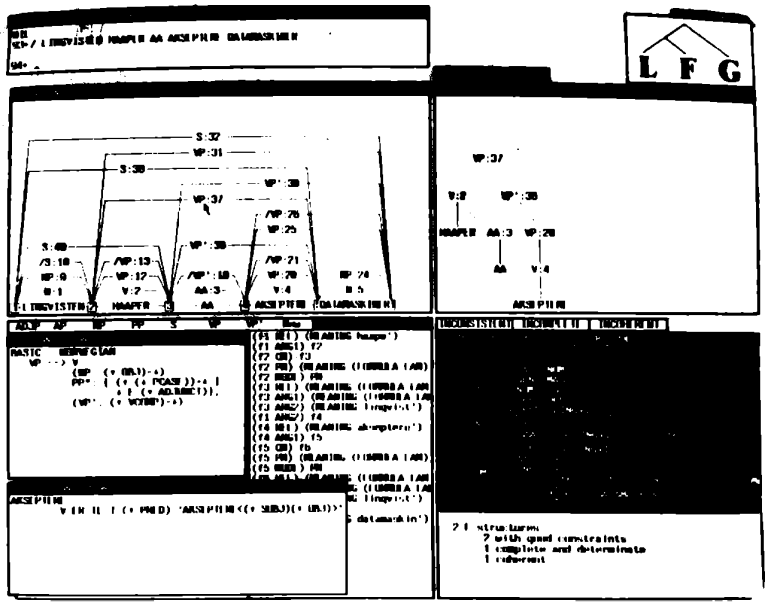
Eksempel 1.

Vi har skrevet inn setningen øverst til venstre og får ut et antall frasestrukturtrær (c-structure) og funksjonelle strukturer (f-structure). For de funksjonelle strukturer får vi angitt hvor mange som er konsistente, sammenhengende og fullstendige. Ved å peke på merkelappen CHART får vi ut et vindu som viser hvoledes analysen har blitt foretatt.



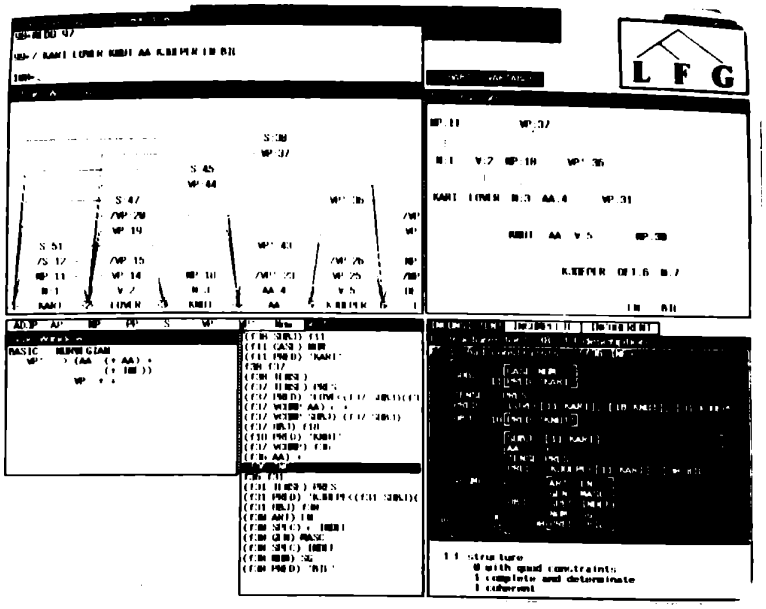
Eksempel 2.

I eksempel 2 har vi også fått ut den semantiske struktur. Ved å peke på forskjellige nivåer i frasetrukturtreet og den funksjonelle struktur vil vi få ut deler av den funksjonelle struktur og den semantiske struktur. Dersom vi peker på et ord vil vi få ut dette ordets innførsel i leksikon. Denne kan vi eventuelt rette og deretter kjøre analysen på ny. Tilsvarende kan vi få ut og rette de grammatiske reglene ved å peke på regelnavnet midt på venstre skjermhalvdel. Dette er vist i eksempel 3 (rule window og lexicon window).



Eksempel 3.

I eksempel 4 ser vi hvorledes det går med en ugrammatisk setning. Vi får her et frasestrukturtre, men ingen konsistent f-struktur. Ved å peke på merkelappen INCONSISTENT viser systemet den inkonsistente f-struktur og angir hvilken funksjonell ligning som ikke er oppfylt (f36 INF). Utsnitt av de funksjonelle ligninger vises i midterste vindu nederst på skjermen. Vi har også kalt opp regelen VP' og ser hvor den aktuelle ligningen står i grammatikken.



Eksempel 4.

LITTERATUR

Bresnan, J. (red.): *The mental reprensentation of grammatical relations*. Cambridge, Mass.: The MIT Press (1982). (Inneholder en rekke artikler om LFG, bl.a. en introduksjon til systemet av R.M. Kaplan og J. Bresnan.)

Halvorsen, P.-K.: Semantics for Lexical-Functional Grammar. *Linguistic Inquiry* 14(4).567-615 (1983).