λ-kalkyle og domeneteori

Dag Normann
Universitetet i Oslo
Matematisk Institutt
Boks 1053 - Blindern
0316 Oslo

5. desember 2005

Innhold

Forord

Dette kompendiet ble skrevet høsten 2005 som lærestoff til et spesialemne i matematikk; “λ-kalkyle og domeneteori”.

Til hjelp for leseren er det opprettet et register bakerst i kompendiet.

Jeg har ikke gitt kreditt i teksten til kilder jeg har brukt, men noen av dem er nevnt i en referanseliste.

Pensum ble dekket av kapitlene 1-6, mens kapitlene 7-9 er lesestoff for spesiet interesserte. Stoffet i kapittel 8 og i avsnitt 9.1 har sålangt forfatteren kjenner til ikke vært publisert tidligere. Det var lenge planen å skrive et avsnitt 9.3 om bar-rekursjon over Kleene-Kreisel-funksjonalene, men de planene er i det minste midlertidig skrinlagt.

Jeg vil takke de som deltok på forelesningene, for deres konstruktive kommentarer til forelesningene og kompendiet og for et for meg hyggelig semester.

Dag Normann

Kapittel 1

Utypet λ-kalkyle

1.1 Innledning

λ-kalkyle representerer et tidlig forsøk på å lage en matematisk modell for det beregnbare. Grunntanken er at alt (vi snakker om i kalkylen) er funksjoner, og at alle funksjoner i prinsippet kan anvendes på alle objekter, vi kan bare ikke regne med at det kommer noe meningsfylt ut.

λ-kalkyle består av et termspråk, samt regler for hvordan termer kan skrives om til andre termer med den samme intuitive betydning. Vi skal her legge mer vekt på *intuitiv* enn på *betydning*, ettersom det går en god stund før vi får en forstandig semantikk for kalkylen.

Historisk sett har λ-kalkyle spilt en rolle både for utviklingen av beregnbarhetsteorien og tidlige programmeringsspråk, og selv før vi hadde rimelig gode matematiske modeller for λ-kalkyle, ble den brukt som et semantisk verktøy for å forstå programmer.

1.2 Kalkylen

I utypet λ-kalkyle opererer vi med en uendelig liste $x_1, x_2, \ldots$ av variable, og med to konstruksjoner av komplekse termer:

Definisjon 1.1 1. Alle variable x_i er termer i utypet λ-kalkyle.

2. Hvis s og t er termer, er (st) en term. Vi kaller dette en *applikasjonsterm.*

3. Hvis x_i er en variabel, og t er en term, er $(\lambda x_i t)$ en term. Vi kaller dette en *abstraksjonsterm* eller en λ-*abstraksjon*.

4. Mengden av λ-*termer* er den minste klassen av termer gitt ved 1. - 3.

I eksempler vil vi bruke x, y, z, u, v og andre bokstaver som variable. Intuisjonen er at en variabel på vanlig måte står for et vilkårlig objekt. En sammensatt term vil da stå for et mer bestemt objekt. Siden vi oppfatter alle objekter som funksjoner, gir det mening å gi termen (ts) den intuitive tolkningen av *funksjonen* t anvendt på *objektet* s.
Hvis s og t har variable i seg, betyr dette at de er termer som står for ubestemte funksjoner og objekter, og da blir også (ts) ubestemt.
(λxt) skal intuitivt sett oppfattes som den funksjonen som gir tolkningen av t fra en gitt tolkning av x.
Allerede nå må vi presisere at det her bare er snakk om intuisjon, utypet λ-kalkyle vil være en ren kalkyle for manipulasjon av λ-termer, uten at det ligger noen reell tolkning til grunn.

Eksempel 1.2 $\lambda x.x$.
Det er tre bemerkninger å knytte til dette eksemplet.

1. Vi har utelatt parentesene. Vi kommer ofte til å gjøre det for å øke lesbarheten av en term. Vi kommer tilbake til hvilke konvensjoner vi vil bruke når vi utelater parenteser.

2. Vi har skrevet en liten prikk mellom λx og x. Dette gjøres også for å få litt luft inn i termen, og dermed øke lesbarheten.

3. $\lambda x.x$ skal intuitivt tolkes som funksjonen som til en tolkning av x gir tilsvarende tolkning av x. $\lambda x.x$ betegner altså identitetsavbildningen i følge den intuisjonen vi legger til grunn.

Vi ser i eksemplet over at tolkningen av $\lambda x.x$ ikke lenger avhenger av tolkningen av variabelen x, bare av tolkningsområdet. Det betyr at vi binder variabelen når vi bruker abstraksjon, på samme måte som vi binder variable når vi bruker kvantorer i logikk. Dette gir oss et skille mellom *fri* og *bundne* forekomster av variable i en λ-term. Dette gir oss også den vanlige definisjonen av *substituerbarhet* og av *substitusjon* av en term for en variabel.

Definisjon 1.3 Vi lar t^s_x være termen vi får når vi erstatter alle fri forekomster av x i t med s.

Vi vil alltid anta at s er substituerbar for x i t når vi bruker denne notasjonen.

Vi skal gi substans til vår intuisjon om disse termene ved å gi to *omskrivningsregler*. Omskrivningsreglene vil være regler for å skrive en term om til en annen uten at det intuitive meningsinnholdet endres. Vi skal begrense oss til to regler, α-regelen og β-regelen. Husk at vi alltid antar substituerbarhet der det er relevant.

Definisjon 1.4

1. $\lambda x.t \to \lambda y.t_x^y$ (α-regelen).
2. $(\lambda x.t)s \to t_x^s$ (β-regelen).
3. Hvis s og t er termer, sier vi at $s \to t$ hvis vi kan skrive s om til t ved ingen, én eller flere gangers bruk av α og β-regelen på delterm-nivå.

Eksempler 1.5

1. La $t = (\lambda x.x)(\lambda x.x)$.
   Vi kan bruke α-regelen til å skrive om den første delen, og får
   $$t \to (\lambda y.y)(\lambda x.x).$$
   Her har vi da et λ-uttrykk anvendt på en annen term, og da kan vi bruke β-regelen (den kunne vi brukt med en gang), og vi får
   $$(\lambda y.y)(\lambda x.x) \to \lambda x.x$$
   ved å sette høyre siden inn for y i matriks til venstresiden. Vi får altså
   $$(\lambda x.x)(\lambda x.x) \to \lambda x.x.$$
   Anvender vi identitetsfunksjonen på identitetsfunksjonen, får vi identitetsfunksjonen som svar.
2. $(\lambda x.xx)(\lambda x.x)$.
   Her har vi også en λ-term anvendt på en annen term, så vi kan bruke β-regelen. Det betyr at vi skal sette inn $(\lambda x.x)$ for x i xx, og da får vi termen i første eksempel. Denne kan vi da skrive videre om til $\lambda x.x$, hvoretter det er umulig å bruke β-regelen.
3. $(\lambda x.xx)(\lambda x.xx)$.
   Igjen kan vi bruke β-regelen. I dette tilfellet innebærer det at vi setter inn $\lambda x.xx$ for x i xx, men da blir resultatet den termen vi startet med, så i en viss forstand kommer vi ikke videre.

Vi ser at β-regelen nettopp formaliserer intuisjonen om at $\lambda x.t$ betegner en funksjon, ved at regelen anvendt på $(\lambda x.t)s$ gir oss det vi får når vi setter inn s for x i t. β-regelen reflekterer derfor vår intuisjon om abstraksjonstermer. α-regelen er mer en bekvemmelighetsregel, og vi vil bruke den enten for å gjøre en term tilgjengelig for substitusjon, eller for å skrive mer lesbare termer ved ikke å belaste samme variable med for mange roller.

Definisjon 1.6 En term t er på *normalform* hvis det ikke er mulig å bruke β-regelen på termen.

I eksemplet over greide vi å skrive to av termene om til termer på normalform, mens det er umulig å skrive den tredje termen om på normalform. Hvis vi lar en λ-term stå for et *program*, vil det å finne en normalform svare til å kjøre programmet. Dette vil gi mer mening etter de tre neste avsnittene.

1.3 Church-Rosser-egenskapen

Vi skal nå gi oss i kast med det første skikkelige teoremet:

Teorem 1.7 *(Church-Rosser) La s, t og r være termer slik at $s \to t$ og $s \to r$.*
Da finnes det en term k slik at $t \to k$ og $r \to k$.

Det dette teoremet sier er at hvis to personer begynner å skrive om på en term uavhengig av hverandre, kan det ikke oppstå noen uopprettelige forskjeller mellom de to termene.
Beviset for Church-Rosser er omstendelig, og vil man være helt sikker på at man har fått tak i et fullstendig bevis, bør man gå til en av Henk Barendrechts bøker om λ-kalkyle. Argumentet under virker overbevisende på forfatteren i skrivende stund. Før vi gir oss i kast med et bevis, skal vi imidlertid se på to konsekvenser:

Korollar 1.8 *La t være en term, s og r to termer på normalform slik at $t \to s$ og $t \to r$.*
Da er s og r like modulo bruk av bundne variable.

Bevis
Ved Church-Rosser kan s og r skrives om til samme term. Siden både s og r er på normalform, kan vi bare bruke α-regelen. Men α-regelen innebærer

nettopp skifte av bundne variable.

Vi har sagt at omskrivningsreglene skal gi nye termer med den samme intuitive betydningen. Det å ha samme betydning bør være en ekvivalensrelasjon. Church-Rosser tillater oss å definere en ekvivalensrelasjon som svarer til

å dokumenterbart ha den samme intuitve betydningen.

Definisjon 1.9 La t og s være to λ-termer.
Vi sier at t og s er *ekvivalente*,

$$t \equiv s$$

hvis det finnes en term r slik at $s \to r$ og $t \to r$.

Korollar 1.10 *$\equiv$ er en ekvivalensrelasjon.*

Bevis
$\equiv$ er opplagt refleksiv og symmetrisk, så det er transitivitet som er utfordringen.
La $s \equiv t$ og $t \equiv r$. Da finnes termer a og b slik at $s \to a$, $t \to a$, $t \to b$ og $r \to b$.
Hvis vi bruker Church-Rosser på t, a og b får vi at det finnes en term k slik at $a \to k$ og $b \to k$.
Siden $\to$ er transitiv, følger det at $s \to k$ og $r \to k$, så $s \equiv r$.

Før vi gir oss i kast med beviset av Church-Rossers teorem, la oss gjøre en observasjon og diskutere utfordringen i beviset og arktekturen til beviset.

Observasjon La t være en λ-term, s og r to (forekomster av) deltermer av t.
Da er enten forekomstene disjunkte eller så er den ene en delterm av den andre.

Det tilsvarende fenomenet gjelder for termer og formler i logikk.

$\to$ er pr. definisjon den refleksive og transitive tillukningen til $\to^1$, hvor $\to^1$ svarer til én gangs bruk av α- eller β-regelen. Hvis vi hadde vært i stand til å vise Church-Rosser-egenskapen for $\to^1$ ville Church-Rosser egenskapen for $\to$ fulgt lett ved induksjon på lengden av den ene omskrivningen og subinduksjon på lengden av den andre omskrivningen. Nå er vi imidlertid ikke i stand til å bevise Church-Rosser-egenskapen for $\to^1$, fordi den ikke holder. Strategien vår vil derfor være å finne en relasjon $\to^*$ med Church-Rosser-egenskapen slik at

$$\to^1 \subseteq \to^* \subseteq \to .$$

Det vil følge at $\to$ er den refleksive og transitive tillukningen til $\to^*$, og $\to$ vil derfor arve Church-Rosser-egenskapen fra $\to^*$.
Resten av dette avsnittet vil være dedisert definisjonen av $\to^*$ og beviset for Church-Rosser egenskapen for $\to^*$. Begrepene innført her vil ikke bli brukt andre steder i kompendiet, og forekommer da heller ikke i registeret.

Definisjon 1.11 La t være en λ-term, $T(t)$ mengden av (forekomster av) deltermer til t, slik at $t \in T(t)$.
En *reduksjonsmengde* for t er en delmengde $X \subseteq T(t)$ slik at hver $s \in X$ enten er på formen $\lambda x.r$ eller på formen $(\lambda x.r)k$.
For hver reduksjonsmengde X for t kan vi utføre en serie omskrivninger av t ved å bruke α- og β-regelen på termene i X innenfra og utover. Her får vi bruk for den observasjonen vi gjorde.
Vi lar $t \to^* s$ hvis vi kan komme fra t til s ved én gangs bruk av en reduksjonsmengde på denne måten.

Det er opplagt at $\to^1 \subseteq \to^* \subseteq \to$, så det gjenstår å vise at $\to^*$ har Church-Rosser-egenskapen. For første inklusjon kan vi merke oss at én gangs bruk av en regel svarer til å bruke en reduksjonsmengde med ett element.
Vi vil skrive $s, t \to^* r$ som forkortelse for $s \to^* r \wedge t \to^* r$, og tilsvarende $s \to^* t, r$ som forkortelse for $s \to^* t \wedge s \to^* r$.

Lemma 1.12 *La s,t,r og k være λ-termer slik at $s \to^* t$ og $r \to^* k$.
Da vil $s^r_x \to^* t^k_x$.*

Bevis
Vi lager en reduksjonsmengde X for s^r_x fra reduksjonsmengdene Y for s og Z for r ved

1. Hvis $u \in Z$, lar vi alle forekomstene av denne u-en i s^r_x være i X, altså, vi får like mange forekomster av u som det er fri forekomster av x i s.

2. Hvis $y \in Y$, lar vi tilsvarende u^r_x være i X.

Ved å utføre $\to^1$-omskrivninger på deltermer i X innenfra og utover kommer vi fra t^r_x til s^k_x.

Bemerkning 1.13 Vi har ikke tatt hensyn til at k muligens ikke er substituerbar for x i t, ettersom vi faktisk mente alvor da vi skrev at vi alltid antar substituerbarhet når vi bruker substitusjonsnotasjonen. Hvis det oppstår problemer med substituerbarhet, må vi ta forbehold om noen mulige ekstraomganger hvor vi bare bruker α-regelen.

Lemma 1.14 *La s, t og r være λ-termer slik at $s \to^* t, r$.*
Da finnes det en λ-term k slik at $t, r \to^* k$.

Bevis
Vi bruker induksjon på oppbygningen av s.
Det vil være tre hovedtilfeller med en del undertilfeller.

Tlifelle 1: s er en variabel x.
Dette er et tomt tilfelle, ettersom s ikke er tilgjengelig for bruk av noen regel.

Tilfelle 2: $s = \lambda x.s'$.

Tilfelle 2.1: s inngår ikke i noen av reduksjonsmengdene.
Da er $t = \lambda x.t'$ og $r = \lambda x.r'$ hvor $s' \to^* t', r'$.
Ved induksjonsantagelsen finnes k' slik at $t', r' \to^* k'$ og hvis vi lar $k = \lambda x.k'$ får vi at $t, r \to^* k$.

Tilfelle 2.2: s inngår i minst én av reduksjonsmengdene.
Da vil $\to^*$-omskrivningene av s til t og r bestå av omskrivninger av $\lambda x.s'$ til hhv. $\lambda x.t'$ og $\lambda x.r'$ og deretter en ytterste bruk av α-regelen i det ene eller begge av tilfellene. Uten tap av generalitet kan vi anta at vi bruker α-regelen i begge tilfellene, så $t = \lambda y.(t')^y_x$ og $r = \lambda z.(r')^z_x$.
Ved induksjonsantagelsen finnes det da en term k' slik at $t', r' \to^* k'$.
Vi har da at $\lambda y.(t')^y_x \to^* \lambda u.(k')^u_x$ og $\lambda z.(r')^z_x \to^* \lambda u.(k')^u_x$, så la $k = \lambda u.(k')^u_x$.

Tilfelle 3: $s = s's''$.

Tilfelle 3.1: s inngår ikke i noen av reduksjonsmengdene.
Dette dekker spesielt de tilfellene hvor s' ikke er en abstraksjonsterm.
I dette tilfellet vil vi ha termer t', t'', r' og r'' slik at $s' \to^* t', r'$, $s'' \to^* t'', r''$, $t = t't''$ og $r = r'r''$.
Ved induksjonsantagelsen finnes det k' og k'' slik at $t', r' \to^* k'$ og $t'', r'' \to^* k''$, og da vil vi ha at $t, r \to^* k'k'' = k$.

Tilfelle 3.2: s forekommer i begge reduksjonsmengdene.

Da må $s' = \lambda x.s'''$, og omskrivningene må ha formen

$$(\lambda x.s''')s'' \to^* (\lambda x.t''')t'' \to^1 (t''')_x^{t''}$$

og

$$(\lambda x.s''')s'' \to^* (\lambda x.r''')r'' \to^1 (r''')_x^{r''},$$

hvor $s''' \to^* t''', r'''$ og $s'' \to^* t'', r''$.
Sammensetningene er i $\to^*$ fordi siste omskrivning foregår på ytterste nivå.
Ved induksjonsantagelsen vil det finnes k''' og k'' slik at

$$t''', r''' \to^* k''' \wedge t'', r'' \to^* k''.$$

Ved Lemma 1.12 vil da

$$(t''')_x^{t''}, (r''')_x^{r''} \to^* (k''')_x^{k''}.$$

Tilfelle 3.3: s forekommer bare i den ene av reduksjonsmengdene, for eksempel for reduksjon til t.
I dette tilfellet vil omskrivningene være

$$(\lambda x.s''')s'' \to^* (\lambda x.t'')t'' \to^1 (t'')_x^{t''}$$

og

$$(\lambda x.s''')s'' \to^* (\lambda x.r''')r'',$$

hvor vi fortsatt har at

$$t''', r''' \to^* k''' \wedge t'', r'' \to^* k''.$$

La k''' og k'' være som i tilfelle 3.2. Ved Lemma 1.12 har vi at

$$(t''')_x^{t''} \to^* (k''')_x^{k''}$$

som før. Videre har vi at

$$(\lambda x.r''')r'' \to^* (\lambda x.k''')k'' \to^* (k''')_x^{k''}$$

hvor sammensetningen er i $\to^*$ fordi den siste omskrivningen foregår på ytterste nivå.

Dette avslutter beviset, og dermed er også Church-Rossers teorem for utypet λ-kalkyle bevist.

1.4 En fikspunkt-operator

Vi skal nå begynne å bruke λ-termer til å "regne" med, og da er det viktig med konvensjoner for hvordan vi kan fjerne parenteser, og for å kunne lese underforståtte parenteser.

1. λ-abstraksjon deler mer enn sammensetning. Det betyr at $\lambda x.st$ betyr $(\lambda x(st))$ og IKKE $(((\lambda xs)t)$ (Husk at . bare er til lesehjelp).

2. Gjentatte sammensetninger skal grupperes mot venstre og ikke mot høyre. Det betyr at str skal stå for $((st)r)$ og IKKE for $(s(tr)$. Mener vi det siste, må vi skrive det.

Eksempel 1.15 La oss se på termen

$$\lambda x \lambda y.str$$

versus termen

$$(\lambda x \lambda y.s)tr.$$

I det første tilfellet har vi en abstraksjonsterm, og vi har ikke synliggjort noe sted vi kan bruke β-regelen.
I det andre tilfellet står det egentlig $((\lambda x.(\lambda y.s))t)r$, hvor vi ser bort fra den ytterste parentesen. Her er det mulig med én β-reduksjon i første omgang, vi kan sette t inn for x i $\lambda y.s$. Vi får altså at

$$((\lambda x.(\lambda y.s))t)r \to ((\lambda y.s)^t_x)r.$$

Under antagelse om at alt er substituerbart, kan vi nå gå videre, og får

$$((\lambda x.(\lambda y.s))t)r \to (s^t_x)^r_y.$$

Et viktig aspekt ved utypet λ-kalkyle er eksistensen av fikspunkt-termer.

Teorem 1.16 *Det finnes en lukket λ-term f slik at for alle andre λ-termer t vil*

$$ft \equiv t(ft).$$

Bevis
La f være termen

$$f = \lambda y.(\lambda x.y(xx))\lambda x.y(xx).$$

Hvis t er en vilkårlig term slik at vi ikke får substituerbarhetsproblemer, vil

$$ft = (\lambda y.(\lambda x.y(xx))\lambda x.y(xx))t.$$

Her kan vi bruke β-regelen, og sette t inn for y. Det gir

$$ft \to (\lambda x.t(xx))\lambda x.t(xx).$$

La $s = (\lambda x.t(xx))\lambda x.t(xx)$. Da vil $ft \to s$.
Nå kan vi bruke β-regelen på s. Det innebærer å sette $\lambda x.t(xx)$ inn for x i $t(xx)$, og resultetet blir

$$t((\lambda x.t(xx))\lambda x.t(xx)) = ts.$$

Vi har at $ft \to ts$, og siden $ft \to s$ vil $t(ft) \to ts$. Det følger at $ft \equiv t(ft)$.

Skal vi tenke intuitivt, har vi at $ft = t(ft)$, altså at ft er et fikspunkt for t. Eksistensen av slike fikpunkt-termer er en viktig årsak til at utypet λ-kalkyle brukes som et semantisk verktøy for studiet av programmer. Hvis P er en selvkallende prosedyre, finnes det egentlig et program R med en prosedyre-variabel p, og P vil være et fikspunkt for R, dvs $P = R(P)$. Hvis vi kan modellere R i utypet λ-kalkyle gir fikspunkt-termer oss også en mulighet til å modellere P.

Hvis vi skal benytte utypet λ-kalkyle til å tolke programmer hvor dataobjektene hentes fra gitte datatyper, kan det være aktuelt å utvide kalkylen med konstanter for disse objektene. Det er imidlertid slik at vi ofte kan representere slike objeker som termer i utypet λ-kalkyle. Det skal vi se på i det neste avsnittet.

1.5 Church-numeraler etc.

En viktig opprinnelig motivasjon for å studere utypet λ-kalkyle var at lukkede termer vil definere partielle funksjoner på de naturlige tallene (her i betydningen de ikke-negative hele tallene) $\mathbb{N}$.
For å kunne gjøre det, må tallene selv representeres. Det gjør vi ved hjelp av de såkalte *Church-numeralene*. I mengdelære vil vi representere tallet k som en utvalgt mengde med k elementer. I λ-kalkyle vil vi representere k som det å iterere en funksjon k ganger.
Det er også nyttig å kunne representere de Booleske verdiene $\mathbf{T}$ og $\mathbf{F}$ (sann

og usann). Intuitivt vil vi gjøre det ved to funksjoner tt og ff av to variable, hvor tt gir verdien av den andre variabelen og ff gir verdien av den første variabelen. Det gir oss følgende definisjon:

Definisjon 1.17

1. La $k_n = \lambda y \lambda x . y^n(x)$, hvor vi mener $y(y(\cdots(x)\cdots))$ og hvor y forekommer n ganger.
2. La $tt = k_0 = \lambda y \lambda x . x$.
3. La $ff = \lambda y \lambda x . y$.

Definisjon 1.18 La $f : \mathbb{N} \to \mathbb{N}$ være en partiell funksjon.
Vi sier at f er λ-*beregnbar* hvis det finnes en lukket λ-term $\hat{f}$ slik at for alle n og m,

$$\hat{f} k_n \to k_m \Leftrightarrow f(n) = m.$$

Siden Church-numeraler er på normalform, ser vi at alle lukkede termer t definerer en partiell funksjon f_t, hvor $f_t(n)$ er udefinert hvis tk_n ikke kan bringes på en normalform, eller hvis normalformen ikke er en numeral, og hvor $f_t(n) = m$ hvis m er slik at $tk_n \to k_m$. I dette siste tilfelle er m entydig bestemt, siden to omskrivninger av en term til normalform må gi det samme resultatet modulo skifte av bundne variable.

Church så på de λ-definerebare funksjonene som en matematisk modell for de beregnbare funksjonene, og den opprinnelige Church tese sier nettopp at alle beregnbare partielle funksjoner på $\mathbb{N}$ er λ-definerbare. Det har senere vist seg at Church's modell er ekvivalent med Turings modell, Gödels modell og alle andre eksisterende modeller som tar sikte på å fange opp det uformelle begrepet *beregnbarhet*. Vi skal ikke bevise dette, hovedsaklig fordi vi ikke skal innføre alle de modellene λ-definerbarhet er ekvivalent med. Vi skal imidlertid se hvordan vi kan utvikle en smule rekursjonsteori ved å bruke λ-termer.

La oss først observere at lukkede λ-termer definerer partielle funksjoner av en vilkårlig gitt aritet. Eksempelvis, hvis A er ariteten $\mathbb{N} \times \mathbb{N} \times \mathbb{B} \times \mathbb{B} \to \mathbb{N}$ og t er en lukket term, kan vi la $f_t^A(13, 17, \mathbf{F}, \mathbf{T}) = 19$ hvis og bare hvis $tk_{13}k_{17}ff tt \to k_{19}$.

Lemma 1.19 *Det finnes en term S slik at $Sk_n \to k_{n+1}$ for alle $n \in \mathbb{N}$.*

Bevis
La $S = \lambda z \lambda y \lambda x . y(zyx)$.

Ved en serie β-overganger som leseren utfordres til å verifisere selv får vi

$Sk_n = (\lambda z \lambda y \lambda x.y(zyx))\lambda y \lambda x.y^n(x)$

$\to \lambda y \lambda x.y((\lambda y \lambda x y^n(x))yx)$

$\to \lambda y \lambda x.y((\lambda x.y^n(x))x)$

$\to \lambda y \lambda x.y(y^n(x)) = \lambda y \lambda x.y^{n+1}(x) = k_{n+1}$.

Her bruker vi at $(\lambda x.t)x \to t$. Vi vil ofte komme i den situasjonen at samme variabel brukes mange steder i en term, og det kan være litt av en utfordring å finne ut av hvor det er mulig å bruke β-regelen, og hvordan den skal brukes riktig. Vi kaller S for *etterfølger* og bokstaven er entet fra den engelske varianten *successor*.

Vi kan også definere termer ved tilfeller:

Lemma 1.20 *a) Det finnes en term Z slik at for alle termer s og t vil*

$$Zk_0st \to s$$

og

$$Zk_{n+1}st \to t.$$

b) Det finnes en term $\supset$ slik at for alle termer s og t vil

$$\supset (tt)st \to s$$

og

$$\supset (ff)st \to t.$$

Bevis

a) La
$$Z = \lambda z \lambda y \lambda x.z(\lambda u.x)y.$$
Setter vi inn k_n, s og t for hhv. z, y og x, får vi at vi skal iterere konstantfunksjonen med verdi t n ganger på argumentet s. Det gir svar s hvis $n = 0$, mens det gir svar t for $n > 0$.

b) La
$$\supset \; = \lambda z \lambda y \lambda x.zxy.$$
Setter vi inn tt for z, s for y og t for x får vi $(tt)ts$ som reduseres til s, og setter vi inn ff for z, s for y og t for x, får vi $(ff)ts$ som reduseres til t.

I to oppgaver skal vi se på hvordan vi kan bruke fikspunkt-konstruksjonen til å simulere primitiv rekursjon og μ-operatoren. Vi må imidlertid først ta for oss forgjengerfunksjonen.

Definisjon 1.21 La $P(n) = n \dot{-} 1$, hvor $n \dot{-} m = n - m$ hvis $n \geq m$, mens $n \dot{-} m = 0$ om $n \leq m$.

Kleene viste at P er λ-definerbar. Vi skal gjennomgå Kleenes argument, og etter det ligger rekursjonsteorien åpen foran oss.

Definisjon 1.22 La (n, m) være et ordnet par av naturlige tall.
Vi lar $[n, m]$ være termen
$$[n, m] = \lambda z \lambda y \lambda x.z^n(y^m(x)).$$

Lemma 1.23 *Det finnes termer* Π_1 *og* Π_2 *slik at vi for alle naturlige tall* n *og* m *har at* $\Pi_1[n, m] \to k_n$ *og* $\Pi_2[n, m] \to k_m$.

Bevis
La $I = \lambda u.u$, la $\Pi_1 = \lambda v \lambda z.vzI$ og la $\Pi_2 = \lambda v.vI$.
Da har vi
$\Pi_1[n, m] = (\lambda v \lambda z.vzI)(\lambda z \lambda y \lambda x.z^n(y^m(x)))$
$\to \lambda z.(\lambda z \lambda y \lambda x.z^n(y^m(x)))zI$
$\to \lambda z.(\lambda y \lambda x.z^n(y^n(x)))I$
$\to \lambda z \lambda x.z^n(I^m(x))$
$\to \lambda z \lambda x.z^n(x)$,
som modulo et skifte fra z til y er det samme som k_n.
Vi har også
$\Pi_2[n, m] = (\lambda v.vI)(\lambda z \lambda y \lambda x.z^n(y^m(x)))$
$\to (\lambda z \lambda y \lambda x.z^n(y^m(x)))I$
$\to \lambda y \lambda x I^n(y^m(x))$
$\to \lambda y \lambda x.y^m(x) = k_m$
Vi overlater det til leseren å følge med på hvilke bruk av β-regelen vi gjør til enhver tid.

Lemma 1.24 *Det finnes en term Σ slik at for alle naturlige tall n og m vil*

$$\Sigma k_n k_m \to [n,m].$$

Bevis
La $\Sigma = \lambda u \lambda v \lambda z \lambda y \lambda x.uz(vyx)$.
Da er
$\Sigma k_n k_m =$
$(\lambda u \lambda v \lambda z \lambda y \lambda x.uz(vyx))(\lambda y \lambda x.y^n(x))(\lambda y \lambda x.y^m(x)) \to$
$\lambda z \lambda y \lambda x.(\lambda y \lambda x.y^n(x)z(\lambda y \lambda x.y^m(x)yx)) \to$
$\lambda z \lambda y \lambda x.(\lambda y \lambda x.y^n(x)zy^m(x) \to$
$\lambda z \lambda y \lambda x.(\lambda x.z^n(x))y^m(x)) \to$
$\lambda z \lambda y \lambda x.z^n(y^m(x)) = [n,m]$.
Det kan være vanskelig å følge med på hvilke β-omskrivninger som gjøre stil enhver tid i denne utregningen. Dette overlates til leseren som øvelse.
Merk at i den første overgangen setter vi inn både for u og for v.

Lemma 1.25 *Det finnes en term T slik at for alle n og m vil*

$$T[n,m] \to [n+1,n].$$

Bevis
Sett $T = \lambda x.\Sigma(S(\Pi_1 x))(\Pi_1 x))$.
Verifikasjonen er triviell.

Lemma 1.26 *P er λ-definerbar.*

Bevis
La $\hat{P} = \lambda z \Pi_2(zT[0,0])$.
Hvis vi regner på $\hat{P}k_n$ får vi

$$\hat{P}k_n = (\lambda z.\Pi_2(zT[0,0])k_n \to \Pi_2(k_n T[0,0]) \to \Pi_2(T^n([0,0])).$$

Fra egenskapen til T ser vi at $T^0([0,0]) = [0,0]$ og at $T^{n+1}([0,0]) \to [n+1,n]$. I begge tilfeller får vi den ønskede verdien når vi bruker Π_2.

Den videre utviklingen av rekursjonsteori vil vi gjøre i form av oppgaver.

1.6 Kombinatorer

Det finnes en alternativ tilnærming til λ-kalkyle, via de såkalte *kombinatorene*. En definisjon av en *kombinator* er at den er en lukket λ-term. Vi skal se på den mer algebraiske tilnærmingen, og la kombinatorene være termer i et språk med omskrivningsregler. Hovedpoenget er at vi eliminerer variable og abstraksjon fra λ-kalkyle, og står igjen bare med applikasjon.

Definisjon 1.27

1. Konstantene I, K og S er kombinatorer.
2. Hvis N og M er kombinatorer, er sammensetningen (NM) en kombinator.

Som i λ-kalkyle vil vi droppe parenteser, ved at vi kan skrive $M_1M_2\cdots M_n$ når vi mener $(\cdots((M_1M_2)\cdots M_n)$.

Definisjon 1.28 Vi har tre regneregler (omskrivningsregler) for kombinatorer, og de kan brukes på deluttrykk:

1. $IM = M$
2. $KMN = M$
3. $SMNL = ML(NL)$

Vi skal først se at alle kombinatorer kan realiseres som lukkede λ-termer:

Lemma 1.29 *Det finnes lukkede λ-termer $\hat{I}$, $\hat{K}$ og $\hat{S}$ slik at om s, t og r er vilkårlige λ-termer, så vil*

$$\hat{I}t \to t$$

$$\hat{K}st \to s$$

$$\hat{S}str \to sr(tr)$$

Bevis
La $\hat{I} = \lambda x.x$
La $\hat{K} = \lambda x\lambda y.x$
La $\hat{S} = \lambda x\lambda y\lambda z.xz(yz)$
Det er trivielt å se at disse termene "funker".

Det som er interessant er at opp til ekvivalens kan vi uttrykke alle lukkede λ-termer som kombinatorer.

Teorem 1.30 *La t være en λ-term med fri variable $x_1, \ldots, x_n$. Da er t ekvivalent til en λ-term som oppnås ved sammensetninger av termene $\hat{I}$, $\hat{S}$, $\hat{K}$ og variablene $x_1, \ldots, x_n$.*

Bevis
La L være språket av sammensetningstermer i konstantene I, S og K og variablene x_1, x_2, $\cdots$.
La M være en term i L. Ved rekursjon på oppbygningen av M definerer vi termen $\lambda^* x_i.M$ som en term i L hvor x_i ikke forekommer. Poenget er at i kombinator-kalkylen skal $(\lambda^* x_i.M)N = M_{x_i}^N$ for alle termer N.

Hvis $M = x_i$, lar vi $\lambda^* x_i.M = I$.

Hvis x_i ikke forkommer i M, lar vi $\lambda^* x_i.M = KM$.

Ellers lar vi $\lambda^* x_i.(MN) = S(\lambda^* x_i.M)(\lambda^* x_i.N)$

Vi viser så ved induksjon på oppbygningen av M at $(\lambda^* x_i.M)T = M_{x_i}^T$ for alle kombinatorer T i L. Beviset er trivielt, muligens untatt det siste tilfellet. Ved induksjonsantagelsen har vi imidlertid

$$(\lambda^* x_i.(MN))T = S(\lambda^* x_i.M)(\lambda^* x_i.N)T$$

$$= ((\lambda^* x_i.M)T)((\lambda^* x_i.N)T) = M_{x_i}^T N_{x_i}^T = (MN)_{x_i}^T.$$

Dette betyr at vi ved rekursjon kan eliminere alle forekomster av abstraksjon i en λ-term slik at omskrivningen er likeverdig med orginalen i forhold til β-regelen. Dette avslutter beviset. Merk at vi ikke har gjort noen antagelser om T, så T kan godt være en term hentet fra en alternativ kalkyle.
Vi kan merke oss at vi overbelaster = -tegnet, ved at vi bruker det både for syntaktisk likhet og for likhet via omskrivninger. Siden vi ikke vil arbeide mer med kombinatorer i dette kurset, er ikke det så farlig.

1.7 Oppgaver

Oppgave 1.1 Vis at hvis de partielle funksjonene $f : \mathbb{N} \to \mathbb{N}$ og $g : \mathbb{N} \to \mathbb{N}$ er λ-definerbare, vil sammensetningen

$$h(x) = g(f(x))$$

være λ-definerbar.
Formuler og generaliser dette resultatet til funksjoner av flere variable.

Oppgave 1.2 Vis at $(\lambda x_1 \lambda x_2 \lambda x_3 . x_2) t_1 t_2 t_3 \to t_2$.
Forklar hvordan dette kan generaliseres til at alle projeksjonsfunksjoner

$$I_{n,i}(x_1, \ldots, x_n) = x_i$$

er λ-definerbare.

Oppgave 1.3 Vis at addisjon og multiplikasjon er λ-definerbare.
Hint Addisjon og multiplikasjon kan defineres ved rekursjon, og ved hjelp av en fikspunkt-konstruksjon kan rekursjon simuleres i utypet λ-kalkyle.

Oppgave 1.4 Vis at $F(x, y) = x \dot{-} y$ er λ-definerbar.
Bruk dette til å vise at den karakteristiske funksjonen $K_<$ til ordningen av $\mathbb{N}$ er λ-definerbar.

Oppgave 1.5 La $f : \mathbb{N} \to \mathbb{N}$ være en partiell funksjon.
La

$$\mu(f)(n) = n$$

om $f(n) = 0$ og la

$$\mu(f)(n) = \mu(f)(n+1) + 1$$

om $f(n) \in \mathbb{N}$, men $f(n) \neq 0$.

a) Vis at vi kan finne en term $\hat{\mu}$ slik at for alle lukkede termer t og alle n vil

$$\hat{\mu} t k_n \to k_{\mu(f_t)(n)}$$

når $\mu(f_t)(n)$ er definert.

b) Vis at μ-operatoren er λ-definerbar, og diskuter hva dette måtte bety.

Bemerkning Vi har nå vist at alle Gödels μ-rekursive funksjoner er λ-definerbare.

Oppgave 1.6 [Trolig vanskelig]
Beskriv funksjonene f_{k_n} for hver n.

Oppgave 1.7 Uttrykk λ-termene k_0, k_1, k_2 og ff ved hjelp av kombinatorene K, S og I.

Oppgave 1.8 Vis at kalkylen med kombinatorer har Church-Rosser-egenskapen.
Hint Reduser problemet til Church-Rosser-egenskapen for λ-kalkyle.

Kapittel 2

En λ-kalkyle for endelige typer

2.1 Kalkylen

I utypet λ-kalkyle var alle objektene av samme karakter, det vil si, de var alle på en måte funksjoner og alle var i en viss forstand i definisjonsområdet til alle. Dette betyr spesielt at det er umulig å lage en mengdeteoretisk modell for utypet λ-kalkyle hvor objektene virkelig er funksjoner og hvor de virkelig tar hverandre som argumenter. Dette kommer i konflikt med det kjente mengdeteoretiske prinsippet om at $\in$ er en velfundert relasjon.

I tre av variantene av typet λ-kalkyle vi skal se på, er situasjonen en annen. Her vil situasjonen være at vi har en klassisk mengdeteoretisk semantikk for kalkylen, en semantikk som på den ene siden gir oss tryggere intuisjon og som på den andre siden kan brukes til å utforske kalkylen selv. Det må imidlertid presiseres at vi fortsatt snakker om formelle kalkyler, det vil si termspråk med omskrivningsregler.

Vi skal gi en generell definisjon, men konsentrere oss om et kjerneeksempel som svarer til typeteori over to grunntyper, en for de naturlige tallene og en for de booleske verdiene. Dette kjerneeksemplet vil bli utvidet etterhvert som vi beveger oss retning Gödels T og PCF.

Definisjon 2.1 La $\mathcal{B}$ være en mengde *basistyper*, det vil si en mengde *typekonstanter.*
Klassen av *endelige typer over* $\mathcal{B}$ er definert induktivt ved

1. Hvis $b \in \mathcal{B}$, vil b være en type.

2. Hvis σ og τ er typer, vil $(\sigma \to \tau)$ være en type.

Intuitivt vil en type være en term som betegner en mengde objekter. For hver $b \in \mathcal{B}$ vil vi derfor ha en mengde $||b||$ i tankene. Siden vi skal holde oss i en verden hvor funksjoner spiller hovedrollen, vil vi tenke på $||(\sigma \to \tau)||$ som en mengde funksjoner $f : ||\sigma|| \to ||\tau||$. Det vil være galt å tenke på det som alle funksjoner; det er bedre å tenke på funksjonsrommet som mengden av alle potensielt interessante funksjoner.

Kjerneeksempel La $\mathcal{B} = \{\iota, o\}$ (ι leses 'iota', o leses 'omikron') .
Intuitivt skal $||\iota||$ være mengden av naturlige tall, mens $||o||$ skal være mengden av booleske verdier. Vi skal ikke gi det som formelle definisjoner, ettersom vi underveis vil bruke varianter over dette temaet.

La oss allerede nå gi standardkonvensjonen for dropping av parenteser: Vi skriver

$$\sigma_1, \sigma_2, \ldots, \sigma_n \to \tau$$

for

$$(\sigma_1 \to (\sigma_2 \to \cdots \to (\sigma_n \to \tau) \cdots)).$$

Bakgrunnen for denne konvensjonen er at det er liten forskjell mellom en funksjon av to eller tre variable og en tilsvarende *funksjonal*: Gitt $f(x, y, z)$ kan vi definere $\hat{f}$ ved $\hat{f}(x)(y)(z) = f(x, y, z)$. Her er $\hat{f}(x)(y)$ en funksjon som tar z som argument. $\hat{f}(x)$ er derfor en funksjon som tar y som argument og gir en funksjon som tar z som argument til svar.
Beviset for følgende lemma gis som en oppgave:

Lemma 2.2 *La δ være en type.*
Da kan δ skrives entydig på formen

$$\delta = \sigma_1, \ldots, \sigma_n \to b$$

hvor $n \geq 0$ og $b \in \mathcal{B}$.

Ettersom typene er definert via en induktiv definisjon, kan de ordnes på forskjellige nivåer ut fra hvor langt ut i induksjonen de kommer. Det viser seg imidlertid at dette ikke er spesielt fruktbart. Det er stor prinsipiell forskjell mellom et tall $n \in \mathbb{N}$ og en funksjon $f : \mathbb{N} \to \mathbb{N}$, mens det i lys av rekursjonsteorien ikke er stor prinsipiell forskjell mellom en funksjon $f : \mathbb{N} \to \mathbb{N}$ og en funksjon $g : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$. Dette vil være objekter av tilnærmet samme mengdeteoretiske kompleksitet. Følgende definisjon av *nivå* viser seg å være den mest fruktbare:

Definisjon 2.3 Vi definerer *nivået* $niv(\sigma)$ et til en type σ ved rekursjon på oppbyggingen av typen ved

1. $niv(b) = 0$ hvis $b \in \mathcal{B}$
2. $niv(\sigma \to \tau) = max\{niv(\sigma) + 1, niv(\tau)\}$.

Foreløpig har vi bare snakket om typene. Nå er tiden moden til å gi typene et syntaktisk innhold:

Definisjon 2.4 Et språk L for endelig typeteori består av:

1. En mengde $\mathcal{B}$ av basistyper med den tilhørende mengden av endelige typer.
2. En uendelig liste av variable $\{x_i^\sigma\}_{i\in\mathbb{N}}$ for hver type σ
3. En mengde $\mathcal{C}$ av *typede konstanter*.
4. Alle variable og konstanter er *typede termer* av sine oppgitte typer.
5. Hvis s er en term av type $\sigma \to \tau$ og t er en term av type σ, så er (st) en term av type τ.
6. Hvis t er en term av type τ og x en variabel av type σ, er (λxt) en term av type $\sigma \to \tau$.

Vi vil ha de samme konvensjoner for å utelate parenteser som i utypet λ-kalkyle.
Vi vil ha to måter å uttrykke at t er en term av type σ på, enten å bruke σ som en øvre indeks, som i t^σ og mer formelt i x_i^σ, eller som en form for deklarasjon

$$t : \sigma.$$

Kjerneeksempel I utypet λ-kalkyle hadde vi Church-numeraler og λ-termer som svarte til booleske verdier. I vårt kjerneeksempel vil vi ha konstanter for disse og andre relevante funksjoner som vi kunne definere i utypet λ-kalkyle.
I kjerneeksemplet inngår konstantene $0 : \iota$, $tt : o$, $ff : o$, $S : \iota \to \iota$, $P : \iota \to \iota$, $\supset_\iota : o, \iota, \iota \to \iota$, $\supset_o : o, o, o \to o$ og $Z : \iota \to o$. .
Disse konstantene skal være de typede representantene for de tilsvarende termene vi definerte i utyped λ-kalkyle, og dette vil bli reflektert i de overgangsreglene vi vil gi.
Vi vil erstatte Church-numeralene k_n med kanoniske termer e_n hvor $e_0 = 0$ og $e_{n+1} = Se_n$. e står for *endelig*. Denne notasjonen vil følge oss til utvidelser som Gödels T og LCF/PCF.

Den siste ingrediensen i typet λ-kalkyle vil være omskrivningsreglene, eller reduksjonsreglene. Vi kan merke oss at α- og β-reglene gir mening også for disse språkene.

Definisjon 2.5 En *typet λ-kalkyle* består av et språk for typet λ-kalkyle samt en mengde omskrivningsregler som skal kunne bukes på deltermer, og som omfatter alle typede instanser av α-regelen og β-regelen. Der hvor en regel omfatter substitusjon eller omskrivning av en term med fri variable, forutsettes det at regelen bare brukes hvis man ikke binder eller frigjør variable under utøvelsen av regelen.

Siden det siste forbeholdet bare vil gjelde bruk av α- og β-regelen i våre eksempler, går vi ikke nærmere inn på det.
På vanlig måte vil vi si at $t \to s$ hvis vi kan komme fra t til s ved ingen, en eller flere lovlige bruk av overgangsregler på deltermnivå.

Kjerneeksempel Vi fortsetter kjerneeksemplet vårt ved å gi overgangsreglene knyttet til konstantene. Her vil b betegne en vilkårlig av de to grunntypene ι og o.
I kjerneeksemplet gjelder følgende overgangsregler i tillegg til alle typede instanser av α- og β-regelen:

1. $P0 \to 0$
2. $P(St) \to t$
3. $Z0 \to tt$
4. $Z(St) \to ff$

5. $\supset_b (tt)sr \to s$

6. $\supset_b (ff)sr \to r$.

hvor det alltid er underforstått at termene er riktig typet.

Bemerkning 2.6 I kjerneeksemplet har vi utelatt en viktig typekonstruksjon, nemlig *kartesisk produkt* med tilhørende paringsfunksjoner og projeksjonsfunksjoner. Hvis kjerne-eksemplet skal kunne brukes til å gi en analyse av fragmenter av tallteori, er dette en utilgivelig forglemmelse. Dette er imidlertid et aspekt av typeteorien som vi ikke skal komme inn på i dette kurset, og derfor nøyer vi oss med å ta med de ingrediensene som vi vil finne igjen i PCF, et av læringsmålene i kurset. Et av bruksområdene til typeteori er å gi karakteriseringer av klasser av beregnbare funksjoner.

Definisjon 2.7 La $f : \mathbb{N} \to \mathbb{N}$.
Vi sier at f er *definerbar i endelig typeteori* hvis det finnes en lukket term t av type $\iota \to \iota$ slik at $te_n \to e_{f(n)}$ i vårt spesialtilfelle av typeteori for alle $n \in \mathbb{N}$.

Vi skal etterhvert se at dette er et veldig svakt begrep, og at det er et naturlig mål å utvide spesialeksemplet med konstanter og overgangsregler som gir oss flere definerbare funksjoner.

2.2 Normalisering

Church-Rosser-egenskapen er en potensiell egenskap ved alle omskrivningsystemer. Hvis vi tar utgangspunkt i kjerneeksemplet vårt, kan vi erstatte alle termer i kjerneeksemplet med termer i utypet λ-kalkyle, se Oppgave 2.3. Dette betyr imidlertid ikke at vi arver Church-Rosser-egenskapen fra utypet λ-kalkyle. Derfor burde vi bevist den på nytt.
Ettersom det ikke kommer nye utfordringer inn i beviset, lar vi det være, men antar uten blygsel at alle λ-kalkylene vi vil se på fra nå av har Church-Rosser-egenskapen. Ettersom vi innfører flere overgangsregler, må vi redefinere hva vi mener med en *normalform*.

Definisjon 2.8 La t være en term i kjerneeksemplet av typet λ-kalkyle.

a) t er på *β-normalform* hvis det ikke finnes noen delterm på formen $(\lambda x.s)r$.
   En delterm på formen $(\lambda x.s)r$ er et eksempel på en *redeks* som vi kaller en *β-redeks.*

b) t er på *normalform* hvis det er umulig å bruke andre omskrivningsregler enn α-regelen på deltermer av t.
   En *redeks* er en delterm som tillater direkte omskrivning utenom α-regelen.

Fra nå av i dette avsnittet skal vi bare beskjeftige oss med kjerneeksemplet i endelig typeteori, og vil ikke presisere det hver gang.

Lemma 2.9 *La t være en term på β-normalform. Hvis vi anvender en av de andre omskrivningsreglene (utenom β-regelen) på t, får vi fortsatt en term på β-normalform.*

Bevis
Det blir ikke innført noen β-redekser hvis vi bruker noen av de andre reglene.

Lemma 2.10 *La t være en term på β-normalform. Da vil enhver fortsatt bruk av omskrivningsregler utenom α-regelen være endelig.*

Bevis
Vi ser at alle de øvrige omskrivningsreglene gir kortere termer, så dette følger trivielt ved induksjon på lengden av termen.

Definisjon 2.11 En term t er *sterkt normaliserbar* hvis enhver uendelig følge av enkeltoverganger

$$t \to^1 t_1 \to^1 t_2 \to^1 \cdots$$

ender med uendelig mange gangers bruk av α-regelen.

Vi vil fra nå av se bort fra α-regelen når vi snakker om lengden på en omskrivning. Siden vi til enhver tid bare har et endelig valg av mulige omskrivninger, gjelder

Lemma 2.12 *La t være sterkt normaliserbar. Da finnes det en øvre grense $v(t)$ for hvor lange sekvenser av omskrivninger vi kan utføre med t som utgangspunkt.*

Bevis
Beviset følger av Königs lemma. Lesere som ikke kjenner Königs lemma kan anta at det finnes vilkårlig lange omskrivningskjeder og utlede at det da må finnes en uendelig omskrivningskjede. Beviset kan ikke gjøres helt konstruktivt.

Vi skal nå gi oss i kast med å bevise at alle termer i spesialtilfellet vårt er sterkt normaliserbare. Dette beviset er på ingen måte enkelt, og dette er en type bevis som ikke kan formaliseres i en teori som ligger for tett opp til den kalkylen vi studerer, det er klare paralleller mellom et normaliseringsteorem og et konsistensteorem. Som en del av beviset vil vi gi noen definisjoner som bare brukes senere i beviset, og som derfor ikke beæres med nevnelse i registeret.

Definisjon 2.13 Vi sier at en term t av type σ er *reduserbar*, $t \in R_\sigma$, ved rekursjon på σ:

1. Hvis σ er en basistype er $t \in R_\sigma$ hvis t er sterkt normaliserbar.

2. Hvis t er av type $\sigma = \tau \to \delta$ vil $t \in R_\sigma$ hvis $tu \in R_\delta$ hver gang $u \in R_\tau$.

Definisjon 2.14 Vi sier at en term t er *nøytral* hvis t ikke er på formen $\lambda x.s$ og t ikke er noen av konstantene S, P, Z, $\supset_\iota$ eller $\supset_o$.

Lemma 2.15 *Følgende fire egenskaper holder for alle typer* σ

CR 1 Hvis $t \in R_\sigma$ *er* t *sterkt normaliserbar.*

CR 2 Hvis $t \in R_\sigma$ *og* $t \to t'$, *vil* $t' \in R_\sigma$.

CR 3 Hvis t *er nøytral og av type* σ, *og vi hver gang vi skriver om en redeks i* t *i tråd med en av reglene får en term* $t' \in R_\sigma$, *så er* $t \in R_\sigma$.

CR 4 Hvis t *er av type* σ, *nøytral og normal, så er* $t \in R_\sigma$

Bevis
Vi skal vise CR 1 - CR 4 ved simultan induksjon på oppbyggingen av σ. Vi kan merke oss at CR 4 er en logisk konsekvens av CR 3, så den bryr vi oss ikke om å bevise.

1. σ er en basistype.

CR 1 følger direkte fra definisjonen av R_σ i dette tilfellet.

CR 2 Hvis t er sterkt normaliserbar og $t \to t'$ vil t' være sterkt normaliserbar (med $v(t') \leq v(t)$). Dette følger derfor også rett fra definisjonen.

CR 3 La t være av type σ. (I dette tilfellet må t være nøytral, men det er uvesentlig for argumentet). Hvis alle éngangs omskrivninger av t gir oss en sterkt normaliserbar term, må t selv være sterkt normaliserbar. Påstanden følger derfor ved definisjonen.

2. $\sigma = \tau \to \delta$.

CR 1 La $t \in R_\sigma$ og la x være en variabel av type τ.
Da er x nøytral og på normalform, så ved CR 4 for τ er $x \in R_\tau$.
Det følger at $tx \in R_\delta$, og ved induksjonsantagelsen er tx sterkt normaliserbar.
Nå vil enhver uendelig følge

$$t \to t_1 \to \cdots$$

av omskrivninger gi en uendelig følge

$$tx \to t_1 x \to \cdots$$

av omskrivninger, men det er umulig siden tx er sterkt normaliserbar. Derfor må t være sterkt normaliserbar.

CR 2 La $t \to t'$ og anta at $t \in R_\sigma$.
La $s \in R_\tau$. Vi trenger å vise at $t's \in R_\delta$.
Men vi har at $tu \to t'u$, og vi har at $tu \in R_\delta$, så det følger fra CR 2 for δ at $t'u \in R_\delta$.

CR 3 Anta at t er nøytral og at vi hver gang vi utfører en enkelt omskrivning i t, så får vi en term $t' \in R_\sigma$.
La $u \in R_\tau$ vi må vise at $tu \in R_\delta$.
Siden t er nøytral, vil ikke tu selv være en redeks (poenget med definisjonen av nøytrale termer er at vi ekskluderer den muligheten). Det betyr at en ett-skritts omskrivning av tu vil være på en av formene

$tu \to t'u$ via $t \to t'$

$tu \to tu'$ via $u \to u'$

I det første tilfellet er $t'u \in R_\delta$ fordi $u \in R_\tau$ og vi har antatt at $t' \in R_\sigma$.
I det andre tilfellet bruker vi induksjon på $v(u)$ $v(u) < \infty$ fordi $u \in R_\tau$, og u vil derfor være sterkt normaliserbar. Hvis $v(u) = 0$ betyr det at u er på normalform, og da er dette tilfellet tomt. Hvis $v(u) > 0$ og u' er som ver, er $v(u') < v(u)$. Det betyr at enhver ettskrittsomskrivning av tu' leder til en term i R_δ, og ved CR 3 for δ vil $tu' \in R_\delta$. Da bruker vi CR 3 for δ en gang til, og får at $tu \in R_\delta$.
Som sagt innledningsvis i dette tilfellet, dette gir at $t \in R_\sigma$.

Dette avslutter beviset.

Lemma 2.16 *Alle konstantene S, P, Z, $\supset_b$, 0, tt og ff er reduserbare.*

Bevis
Vi ser på det ene av de to mest komplekse tilfellene, $\supset_\iota$. De andre tilfellene er enklere eller trivielle.
$\supset_\iota$ er av type $\iota, \iota, \iota \to \iota$, og det å vise at $\supset_\iota \in R_{\iota,\iota,\iota\to\iota}$ betyr å vise at $\supset_\iota tsr \in R_\iota$ hver gang t, s og r er i R_ι, det vil si at $\supset_\iota$ er sterkt normaliserbar hver gang t, s og r er sterkt normaliserbare. Det følger ved en enkel induksjon på

$$v(t) + v(s) + v(r),$$

ved at induksjonsantagelsen da sikrer at alle enkeltomskrivninger leder til en term på som er sterkt normaliserbar.

Lemma 2.17 *La $\sigma = \tau \to \delta$, la t være en term av type δ og x en variabel av type τ.*
Anta at $t^s_x \in R_\delta$ hver gang $s \in R_\tau$.
Da vil $\lambda x.t \in R_\sigma$.

Bevis
Vi må vise at $(\lambda x.t)s \in R_\delta$ hver gang $s \in R_\tau$.
Som bemerket før, er $x \in R_\tau$, og fra antagelsen følger det da at $t \in R_\delta$, så vi kan vise at $(\lambda x.t)s \in R_\delta$ ved induksjon på $v(t) + v(s)$.
$(\lambda x.t)s$ er nøytral, så vi kan bruke CR 3 og vise at alle termer vi får fra $(\lambda x.t)s$ ved en gangs bruk av en regel er i R_δ. Det er tre muligheter:

1. $(\lambda x.t)s \to t^s_x \in R_\delta$ ved antagelsen.

2. $(\lambda x.t)s \to (\lambda x.t)s' \in R_\delta$ ved induksjonsantagelsen.

3. $(\lambda x.t)s \to (\lambda x.t')s$.
Vi må vise at t' opfyller betingelsen i lemmaet. Det følger av CR 2 og av at $t \to t' \Rightarrow t^s_x \to (t')^s_x$.
Da kan vi bruke induksjonsantagelsen.

Dette avslutter beviset.

Vi har nå lagt grunnlaget for å bevise

Teorem 2.18 *Alle termer i kjerneeksemplet til endelig typeteori er sterkt normaliserbare.*

Bevis
Vi viser ved induksjon på oppbyggingen av t at hvis t er av type σ med fri variable blant $x_1, \ldots, x_n$ av typer $\sigma_1, \ldots, \sigma_n$ og $s_1, \ldots, s_n$ er termer i $R_{\sigma_1}, \ldots, R_{\sigma_n}$ respektive, så er

$$t^{s_1,\ldots,s_n}_{x_1,\ldots,x_n} \in R_\sigma$$

slik R_σ er definert i dette avsnittet.
Teoremet følger da fra Lemma 2.15 CR 1 og fra observasjonen at $x_i \in R_{\sigma_i}$ for alle i.
Vi skriver $\vec{x}$ for $x_1, \ldots, x_n$ og tilsvarende $\vec{s}$.
Hvis $t = x_i$ er $t^{\vec{s}}_{\vec{x}} = s_i \in R_\sigma$ pr antagelse ($\sigma = \sigma_i$ i dette tilfellet.
Hvis t er en konstant, bruker vi Lemma 2.16.
Hvis $t = rs$ hvor r er av type $\tau \to \sigma$ og s er av type τ, gir induksjonsantagelsen oss at $r^{\vec{s}}_{\vec{x}} \in R_{\tau\to\sigma}$ og at $s^{\vec{s}}_{\vec{x}} \in R_\tau$, og pr definisjon av $R_{\tau\to\sigma}$ gir dette oss at $t^{\vec{s}}_{\vec{x}} = r^{\vec{s}}_{\vec{x}} s^{\vec{s}}_{\vec{x}} \in R_\sigma$.
Hvis $t = \lambda x^\tau.r$ hvor $\sigma = \tau \to \delta$ og r er av type δ, gir induksjonsantagelsen for r oss at for alle $s \in R_\tau$ vil

$$r^{s,\vec{s}}_{x,\vec{x}} \in R_\delta.$$

Ved Lemma 2.17 er da $t^{\vec{s}}_{\vec{x}} \in R_\sigma$ Dette avslutter beviset.

Vi skal bruke normaliseringsteoremet til å vise at alle lukkede termer på nivå 1 definerer totale funksjoner.

Lemma 2.19 *La t være en lukket term på nivå n, og la t være på normalform.*
Hvis x er en variabel som forekommer i t, så er nivået til typen til x mindre enn n.

Beviset er gitt som en oppgave.

Lemma 2.20 *La t være en lukket term på normalform.*

a) Hvis typen til t er ι, er t syntaktisk identisk med en av numeralene e_n

b) Hvis typen til t er o, er $t \in \{tt, ff\}$.

Beviset er gitt som en oppgave.

Teorem 2.21 *La $\vec{\tau}$ være en endelig sekvens av basistyper (i kjerneeksemplet), la b være en basistype og la t være en lukket term av type $\sigma = \vec{\tau} \to b$. Da er funksjonen definert av t total.*

Bevis
La $s_1, \ldots, s_k$ være en sekvens av termer av type $\vec{\tau}$ hvor hver s_i enten er en numeral e_n eller i $\{tt, ff\}$.
Ved normaliseringsteoremet kan $ts_1 \cdots s_k$ skrives om til en normalform, som ved Lemma 2.20 vil være en numeral eller en sannhetsverdikonstant.
Hvis f_t er funksjonen definert av t, betyr dette at f_t er definert for verdiene tilsvarende termene $s_1, \ldots, s_k$.

2.3 En semantikk

Senere skal vi lage matematiske modeller for kalkyler hvor det ligger endel matematikk bak både det å beskrive modellen og bak det å verifisere at det vi konstruerer virkelig er en modell.
Nå skal vi gi en enkel modell for typet λ-kalkyle, enkel i den forstand at det ikke ligger noe vanskelig matematikk til grunn.
Årsaken til at vi gjør det, er at vi da kan se på hva det vil si å lage en modell, uten at målet drukner i middelet. Samtidig vil vi vise noen grunnleggende, enkle egenskaper ved slike modeller, egenskaper som vi ikke vil vise ved senere anledninger.

Definisjon 2.22 For hver type σ definerer vi en mengde $||\sigma||$ ved

1. $||\iota|| = \mathbb{N}$.

2. $||o|| = \mathbb{B} = \{\mathbf{T}, \mathbf{F}\}$.

3. $||\tau \to \delta||$ er mengden av alle funksjoner $f : ||\tau|| \to ||\delta||$.

Definisjon 2.23 La $\{x_i\}_{i\in\mathbb{N}}$ være en fast opptelling av alle variable, og la τ_i være typen til variabelen x_i.
En *tilordning* er en funksjon ν definert på $\mathbb{N}$ slik at $\nu(i) \in ||\tau_i||$ for alle i.
Vi lar V betegne mengden av alle tilordninger. .
Hvis $\nu \in V$ og $a \in ||\tau_i||$, vil ν_i^a være tilordningen ν' hvor $\nu'(i) = a$ og $\nu'(j) = \nu(j)$ når $i \neq j$.

Definisjon 2.24 La t være en term av type σ. Vi definerer $[\![t]\!]$, *tolkningen av* t som en funksjon

$$[\![t]\!] : V \to ||\sigma||$$

ved rekursjon på t som følger:

1. $[\![tt]\!](\nu) = \mathbf{T}$ og $[\![ff]\!](\nu) = \mathbf{F}$ for alle $\nu \in V$.

2. $[\![0]\!](\nu) = 0 \in \mathbb{N}$ for alle $\nu \in V$.

3. $[\![S]\!]$, $[\![P]\!]$, $[\![Z]\!]$, $[\![\supset_b]\!]$ vil også være funksjoner som uavhengig av $\nu \in V$ gir de intenderte tolkningene av disse konstantene.

4. $[\![(st)]\!](\nu) = ([\![s]\!](\nu))([\![t]\!](\nu))$.

5. Hvis $x = x_i$ er av type τ_i, og $a \in ||\tau_i||$, lar vi
$$[\![(\lambda x.t)]\!](\nu)(a) = [\![t]\!](\nu_i^a).$$

Leseren anbefales (av pedagogiske grunner) selv å kontrollere at $[\![t]\!](\nu) \in ||\sigma||$ når t er av type σ.
Når vi definerer $[\![t]\!]$ som en funksjon definert på mengden av tilordninger av verdier til alle variable (uansett om de forekommer i t eller ikke), så er det av bekvemmelightetsgrunner, det gir den enkleste definisjonen. Det er selvfølgelig slik at bare $\nu(i)$ hvor x_i forekommer fritt i t har noen innflytelse på $[\![t]\!](\nu)$. Beviset overlates til leseren:

Lemma 2.25 *La t være en term, og anta at ν og ν' er like for alle i hvor x_i forekommer fritt i t.*
Da vil $[\![t]\!](\nu) = [\![t]\!](\nu')$.

En viktig konsekvens er at hvis t er en lukket term, er $[\![t]\!](\nu)$ uavhengig av ν. Vi vil tillate oss å skrive $[\![t]\!]$ for denne felles verdien.

Vi har nå gitt en tolkning av alle termene i spesialtilfellet av typet λ-kalkyle. Før vi kan hevde at denne tolkningen gir en modell, må vi vise at overgangsreglene våre ikke endrer tolkningen.

Lemma 2.26 *La t og s være termer, x_i en variabel av samme type som s og anta at s er substituerbar for x_i i t.*
La ν være en tilordning.
Da vil $[\![t^s_{x_i}]\!](\nu) = [\![t]\!](\nu_i^{[\![s]\!](\nu)})$.

Bevis
Vi bruker induksjon på oppbygningen av t.
Hvis t er en av konstantene, eller t er en variabel forskjellig fra x_i, holder påstanden trivielt.
Hvis t er x_i er $t^s_{x_i}$ det samme som s, og da er også påstanden triviell.
For sammensatte typer følger påstanden trivielt fra induksjonsantagelsen.

Teorem 2.27 *La t være en term, og la $t \to t'$. Da er $[\![t]\!] = [\![t']\!]$*

Bevis
Det foregående lemmaet viser at teoremet holder for β-regelen, mens for andre regler er teoremet trivielt.
Merk! Ved det foregående lemmaet følger det at om vi skriver om en delterm til en annen delterm med den samme tolkningen, så endrer vi ikke tolkningen av hovedtermen.

2.4 Oppgaver

Oppgave 2.1 Bevis Lemma 2.2.
Drøft hva det betyr at $n = 0$ i Lemma 2.2.

Oppgave 2.2 Finn

$$niv(\iota, (\iota \to (o \to \iota)), ((o \to \iota) \to o) \to (o \to \iota)).$$

Oppgave 2.3 Vis at vi kan embedde kjerneeksemplet i endelig typeteori inn i utypet λ-kalkyle ved å erstatte alle konstantene med lukkede λ-termer og fjerne alle type-henvisninger på fri og bundne variable, slik at hvis to termer er ekvivalente i kjerneeksemplet, så er også oversettelsene ekvivalente i utypet λ-kalkyle.
Hint Sørg for at alle overgangsregler i kjerneeksemplet reflekteres som lovlige omskrivninger i utypet λ-kalkyle

Oppgave 2.4 Bevis Lemma 2.19

Oppgave 2.5 Bevis Lemma 2.20
Hint Vis a) og b) simultant ved induksjon på lengden av t. Bruk Lemma 2.19.

Oppgave 2.6 Denne oppgaven er noe mer åpen enn de andre.
Drøft hva slags funksjoner det er mulig å definere ved hjelp av termer på nivå 1.
Er det mulig å definere addisjon eller multiplikasjon?.
Drøft behovet for å se på sterkere kalkyler.

Oppgave 2.7 Bevis Lemma 2.25

Kapittel 3

Gödels T

I kapittel 2 beskrev vi en enkel typeteori, og vi måtte innse at det ikke var mange funksjoner vi kunne definere i den teorien.
Nå skal vi se på en sterkere teori, hvor vi tillater konstruksjoner ved rekursjon. Dette gir oss en mye sterkere kalkyle, men da blir også egenskaper som sterk normalisering vanskeligere å bevise.
Siden vi vil fokusere på det beregnbarhetsmessige, og ikke på bevisbarhet i tallteori o.l., sparer vi oss for endel bry ved ikke å ta med produkttyper.

3.1 Eksempler

Eksempel 3.1 La $+(n, m) = n + m$. Dette er en funksjon som vi ikke kan definere i typet λ-kalkyle slik vi beskrev den i Kapittel 2, men som har en opplagt rekursiv definisjon ved

1. $+(n, 0) = n$.
2. $+(n, S(m)) = S(+(n, m))$.

Vi ser her at vi uniformt for hver n definerer funksjonen $+_n(m) = n + m$, så det er snakk om å definere funksjoner av én variabel her.

Eksempel 3.2 På tilsvarende måte kan vi definere multiplikasjon som en funksjon av to variable via uniforme definisjoner av funksjoner av én variabel:

1. $\times(n, 0) = 0$.
2. $\times(n, S(m)) = +(n, \times(n, m))$.

I begge disse to eksemplene definerer vi en funksjon g fra en initialverdi a og en funksjon f ved rekursjon etter formatet

1. $g(0) = a$
2. $g(m+1) = f(g(m))$,

hvor den andre variabelen n kommer inn ved at a egentlig er en funksjon av n og f også har n som et argument.

Eksempel 3.3 Fakultetsfunksjonen er også definert rekursivt ved

1. $0! = 1$
2. $S(m)! = \times(S(m), m!)$.

Her inngår m som argument i rekursjons-likningen. Vi har altså formatet

1. $g(0) = a$.
2. $g(m+1) = f(m, g(m))$.

Dette eksemplet viser at hvis vi skal lage en generell kalkyle som fanger opp rekursive definisjoner, så bør initialfunksjonen f kunne ta argumentet for rekursjonen som argument, ikke bare resultatet av rekursjonen på et tidligere trinn.

Eksempel 3.4 Definer funksjonen $F(n, m)$ ved

1. $F(0, m) = S(m)$
2. $F(S(n), 0) = F(n, F(n, 0))$
3. $F(S(n), S(m)) = F(n, F(S(n), m))$

Dette er et eksempel på en dobbelrekursjon. Funksjonen F lar seg ikke definere ved hjelp av gjentatt rekursjon hvis vi bruker de formatene som vi har sett på over.
Typeteorien gjør det mulig for oss å splitte denne rekursjonen opp i enkeltrekursjoner på følgende måte:
La $F_n(m) = F(n, m)$. Vi ser at $F_0 = S$, så F_0 er definerbar sågar i typeteorien fra Kapittel 2.
Videre ser vi at F_{n+1} er definerbar ved rekursjon fra F_n ved følgende generelle definisjon av f^+ fra f:

1. $f^+(0) = f(f(0))$
2. $f^+(m+1) = f(f^+(m))$

Vi ser at omkostningen ved å skrive dobbelrekursjonen om til enkeltrekursjoner er at vi i den ene enkeltrekursjonen definerer en følge av funksjoner, ikke en følge av tall. Funksjonen

$$f \mapsto f^+$$

er en funksjon av type

$$(\iota \to \iota) \to (\iota \to \iota).$$

Siden vi ikke opererer med produkttyper, er funksjonen F av type

$$\iota \to (\iota \to \iota).$$

Ved rekursjon definerer vi en funksjon $n \mapsto F_n$ av type

$$\iota \to (\iota \to \iota)$$

ved

1. $F_0 = S$
2. $F_{S(n)} = F_n^+$.

hvor vi så at f^+ kunne defineres generelt for alle funksjoner f av type $\iota \to \iota$ ved bruk av en felles rekursjonslikning.

Vi kan utvide Eksempel 3.4 til å omfatte trippelrekursjon, oktippelrekursjon og i teorien hvilken som helst definisjon ved nestet rekursjon av endelig dybde. Omkostningen vil være at vi må gå opp i nivå på de typene vi trenger for å faktorisere rekursjonene i enkeltrekursjoner.
Gödels T er en kalkyle som nettopp tillater denne formen for rekursjon hvor objekter av høyere typer inngår.
Det er en sammenheng mellom bruk av nestet rekursjon og bevisteoretisk analyse av bevis hvor det forekommer nestet induksjon. Vi skal ikke komme inn på denne sammenhengen her.

3.2 Kalkylen

Definisjon 3.5 Vi utvider spesialtilfellet fra Kapittel 2 med nye konstanter Rec_σ for hver type σ.
Rec_σ vil ha type

$$\sigma, (\iota, \sigma \to \sigma) \to (\iota \to \sigma).$$

Ser vi litt intuitivt på dette ser vi at Rec_σ er en funksjon som tar to argumenter, x av type σ og f av type $(\iota, \sigma \to \sigma)$, og $Rec_\sigma(x, f)$ blir en følge av elementer av type σ. Den formelle kalkylen skal gi oss at $Rec_\sigma(x, f)$ er funksjonen g definert ved:

1. $g(0) = x$
2. $g(n + 1) = f(n, g(n))$.

Definisjon 3.6 Vi legger følgende omskrivningsregler til spesialtilfellet for endelig typeteori, hvor vi antar at alle termer er av rett type:

1. $Rec_\sigma st0 \to s$
2. $Rec_\sigma stS(r) \to tr(Rec_\sigma str)$.

Vi kaller denne kalkylen for *Gödels T*

Vi skal senere, når vi kommer til PCF, erstatte alle Rec_σ-konstantene med kraftigere saker, og i lys av de planene vil vi beholde P og Z med sine regler. Det er imidlertid mulig å eliminere disse i Gödels T i den forstand at vi kan finne alternative termer som respekterer overgangsreglene til P og Z.

Eksempler 3.7 a) Vi kan definere forgjengerfunksjonen $n \dot{-} 1$ ved

1. $0 \dot{-} 1 = 0$
2. $S(n) \dot{-} 1 = n$

Det betyr at vi kan erstatte P med den lukkede termen

$$\hat{P} = Rec_\iota 0(\lambda x \lambda y.x),$$

hvor både x og y er variable av type ι.

b) Vi kan erstatte Z med den lukkede termen

$$\hat{Z} = Rec_o(tt)(\lambda x \lambda y.(ff)),$$

hvor x har type ι og y har type o.

For begge eksemplene overlater vi det som en oppgave å vise at reglene for P of Z vil være de samme som de reglene vi kan utlede for henholdsvis $\hat{P}$ og $\hat{Z}$.

Som regel vil det være bedre å definere funksjoner ved rekursjon ved å bruke tradisjonell notasjon. Den formelle notasjonen har en tendens til å kamuflere hva som foregår, og egner seg mest til teoretiske studier. Det er en kjedelig øvelse å skrive ut de formelle definisjonene fra de tradisjonelle, en øvelse som desuten gir liten innsikt.

Eksempel 3.8 Vi kan definere $\dot{-}$ ved følgende rekursjon:

1. $n \dot{-} 0 = n$
2. $n \dot{-} S(m) = P(n \dot{-} m)$.

Definisjon 3.9 Et *predikat* på type σ er en lukket term t av type $\sigma \to o$. Intuisjonen er at t svarer til $\{x \in ||\sigma|| \; ; \; t(x) = tt\}$.

Vi legger ikke noen spesiell semantikk til grunn for denne intuisjonen. La oss likevel innrømme at vi vil få problemer med intuisjonen når vi innfører kalkyler uten sterk normalisering.

Eksempel 3.10 La $[\frac{n}{2}]$ betegne heltallsverdien av $\frac{n}{2}$.
Vi kan definere $[\frac{n}{2}]$ ved rekursjon som følger:

- La først *par* være definert ved
  1. $par(0) = 1$
  2. $par(n+1) = 1 \dot{-} par(n)$.

  par vil ta verdien 1 på partall og 0 på oddetall.

- vi definerer så g ved
  1. $g(0) = 0$

2. $g(n+1) = g(n) + par(n+1)$.

Vi skal ikke kjede leseren med å utvikle alt maskineriet til rekursjonsteorien. Poenget med dette siste eksemplet er at vi med dette kan definere en paringsfunksjon på $\mathbb{N}$ innenfor Gödels T ved

$$\langle n, m\rangle = \frac{1}{2}((n+m)^2 + 3m + n),$$

og det vil være innenfor vår kapasitet å definere de tilsvarende projeksjonsfunksjonene π_1 og π_2, se Oppgave 3.4

Simultanrekursjon

Eksempel 3.11 La $f(0) = 1$ og $g(0) = 0$.
La $f(n+1) = f(n) + g(n)$ og la $g(n+1) = f(n)$.
Her har vi definert f og g *simultant* ved induksjon på n.
Leseren bør, ved nærmere ettersyn, kunne se at f definerer Fibonacci-rekken $1, 1, 2, 3, 5, 8, 13, \cdots$, og at vi har brukt en helt generell metode for å omskrive en rekursjon som går to skritt tilbake til en simultanrekursjon.

Poenget med dette eksemplet er at det kan være behov for å fange opp mer generelle rekursjonsformer enn de vi har sett på til nå. Så lenge vi opererer med funksjoner over de naturlige tallene, trenger vi selvfølgelig ikke å bruke styrken til T, men følgende lemma gir en indikasjon på hvorfor det kan være aktuelt å utvide paringsfunksjonene til høyere typer:

Lemma 3.12 *La $f : \mathbb{N}^3 \to \mathbb{N}$ og $g : \mathbb{N}^3 \to \mathbb{N}$ være T-definerbare, a og b gitt. Da finnes det T-definerbare f_1 og g_1 fra $\mathbb{N}$ til $\mathbb{N}$ slik at*

1. *$f_1(0) = a$ og $g_1(0) = b$.*
2. *$f_1(n+1) = f(n, f_1(n), g_1(n))$ og $g_1(n+1) = g(n, f_1(n), g_1(n)$.*

Bevis
f_1 og g_1 er opplagt veldefinerte funksjoner. Vi viser at de er definerbare i T ved å vise at funksjonen h definert ved

$$h(n) = \langle f_1(n), g_1(n)\rangle$$

er definerbar i T. Vi definerer h rekursivt ved

1. $h(0) = \langle a, b \rangle$
2. $h(n+1) = \langle f(n, \pi_1(h(n)), \pi_2(h(n))), g(n, \pi_1(h(n)), \pi_2(h(n))) \rangle$

Da lar vi $f_1(n) = \pi_1(h(n))$ og $g_1(n) = \pi_2(h(n))$

Vi kan selvfølgelig gjennomføre tilsvarende konstruksjoner for simultanrekursjon av flere enn to funksjoner, men velger å stå over detaljene.
Hadde vi hatt T-definerbare parings- og projeksjonsfunksjoner på alle typer, kunne vi gjennomført tilsvarende konstruksjoner for simultan-rekursjon av objekter av høyere typer også. Vi ser imidlertid at allerede for o kan vi få et problem, i den klassiske tolkningen har $||o||$ to elementer, mens $||o|| \times ||o||$ har fire elementer, og en bijeksjon mellom disse mengdene er derfor umulig.
Det kan også være aktuelt å definere følger av funksjoner av forskjellige typer ved simultan rekursjon. Vi vil ikke gi noen eksempler.
I det neste avsnittet skal vi se hvordan vi i noen grad kan omgå disse problemene.

Rene typer

Vi beskrev en modell for spesialtilfellet av endelig typeteori i avsnitt 2.3, og vi kan umiddelbart utvide denne til en modell for T. Det vi trenger å gjøre er å tolke Rec_σ som en konstantfunksjon på V. Selv om det er opplagt hvordan dette skal gjøres, gir vi den formelle definisjonen for fullstendighets skyld. I denne definisjonen gjør vi bruk av at rekursive definisjoner kan gjennomføres i naiv mengdelære, og at rekursjonslikninger vil ha entydige løsninger:

Definisjon 3.13 Hvis σ er en type, $\nu \in V$ en tilordning, $a \in ||\sigma||$ og $f \in ||\iota \to (\sigma \to \sigma)||$, så vil $[\![Rec_\sigma]\!]$ være den entydige løsningen av følgende rekursjonslikning:

1. $[\![Rec_\sigma]\!](\nu)(a)(f)(0) = a$
2. $[\![Rec_\sigma]\!](\nu)(a)(f)(n+1) = f(n)([\![Rec_\sigma]\!](\nu)(a)(f)(n))$

Nå skal vi glemme kalkylen for en stakket stund, og la oss forklare hvorfor:

Eksempel 3.14 Vi har godtgjort at vi har lukkede λ-termer $\langle , \rangle$ av type $\iota \to (\iota \to \iota)$ og λ-termer π_1 og π_2 av type $\iota \to \iota$ slik at for alle numeraler e_n vil

$$\langle , \rangle(\pi_1 e_n)(\pi_2 e_n) \to e_n,$$

hvilket intuitivt betyr at kalkylen verifiserer at

$$\langle \pi_1(n), \pi_2(n)\rangle = n$$

for hver n.
Det betyr også at

$$[\![\lambda x^\iota.\langle , \rangle(\pi_1 x)(\pi_2 x)]\!] = [\![\lambda x^\iota.x]\!].$$

På den annen side har vi ikke at

$$\lambda x^\iota.\langle , \rangle(\pi_1 x)(\pi_2 x) \equiv \lambda x^\iota.x$$

siden det er umulig å regne på disse uttrykkene slik at de blir like. Det andre leddet er på normalform, og det er umulig å eliminere de skjulte forekomstene av *Rec* fra det første uttrykket uten at vi har termer på formen 0 eller $S(t)$ involvert.
Det betyr at alternative definisjoner av at termer er ekvivalente kan være aktuelle, og en slik definisjon kan være at de tolkes likt.

Definisjon 3.15 Vi definerer *de rene typene* ved rekursjon som følger:

1. ι er en ren type.
2. Hvis σ er en ren type, er $\sigma \to \iota$ en ren type.

Vi vil bruke notasjonen ι_0 for ι og ι_{n+1} for $\iota_n \to \iota$ på de rene typene

Vi skal se at vi for mange formål kan begrense våre undersøkelser til de rene typene, ved at ι_n "fanger opp" alle andre typer av nivå $\leq n$.

Definisjon 3.16 La X og Y være mengder.
Et *embedding-projeksons-par*, forkortet *ep-par* fra X til Y er et par (ϕ, ψ) hvor $\phi : X \to Y$, $\psi : Y \to X$ og $\psi(\phi(x)) = x$ for alle $x \in X$.

Vi skal utvikle et nettverk av ep-par definerbare i Gödels T. Ep-par forekommer under forskjellige navn mange steder i matematikken.

Lemma 3.17 *La X, Y og Z være mengder, la (ϕ_1, ψ_1) være et ep-par fra X til Y, og la (ϕ_2, ψ_2) være et ep-par fra Y til Z.*
Da er $(\phi_2 \circ \phi_1, \psi_1 \circ \psi_2)$ et ep-par fra X til Z.

Beviset overlates til leseren ettersom det er uhyggelig enkelt.

Lemma 3.18 *La* $n \leq m$. *Det finnes et* T*-definerbart ep-par* (ϕ_n^m, ψ_n^m) *fra* $||\iota_n||$ *til* $||\iota_m||$.

Bevis
Ved Lemma 3.17 er det nok å konstruere $(\phi_n^{n+1}, \psi_n^{n+1})$ for hver n.

1. La $\phi_0^1(n)(m) = n$.

2. La $\psi_0^1(f) = f(0)$.

3. La $\phi_n^{n+1}(f)(g) = f(\psi_{n-1}^n(g))$ når $n > 0$.

4. La $\psi_n^{n+1}(f)(g) = f(\phi_{n-1}^n(g))$ når $n > 0$.

Merk at typene på f og g ikke er de samme i de to siste linjene.
Vi viser ved induksjon på n at $\psi_n^{n+1}(\phi_n^{n+1}(x)) = x$ for alle x av rett type:

1. $\psi_0^1(\phi_0^1(n)) = \phi_0^1(n)(0) = n$.

2. La $n > 0$. Ved induksjonsantagelsen har vi at hvis f er av type ι_n og a er av type ι_{n-1} så er

$$\psi_n^{n+1}(\phi_n^{n+1}(f))(a) = (\phi_n^{n+1}(f))(\phi_{n-1}^n(a)) = f(\psi_{n-1}^n(\phi_{n-1}^n(a))) = f(a).$$

Dette avslutter beviset.

Siden vi ikke skal bruke for mange krefter på elementær rekursjonsteori, tar vi det for gitt at vi kan utvide paringsfunksjonen på $\mathbb{N}$ til sekvenstallfunksjoner som gir oss $\langle n_0, \ldots, n_{k-1} \rangle \in \mathbb{N}$ for alle ordnede sekvenser $(n_0, \ldots, n_{k-1})$ av lengde k av naturlige tall, at vi har projeksjonsfunksjoner $(\cdot)_i$ for hver koordinat i, og at alt dette er T-definerbare funksjoner.
Vi bruker den samme notasjonen, og løfter definisjonen til sekvenser av funksjoner av en fast ren type:

$$\langle f_0, \ldots, f_{k-1} \rangle(a) = \langle f_0(a) \ldots, f_{k-1}(a) \rangle,$$

med projeksjonsfunksjoner

$$(f)_i(a) = (f(a))_i.$$

Dette bruker vi i beviset for følgende lemma:

Lemma 3.19 *La σ være en type, og la $niv(\sigma) \leq n$.*
Da finnes det et T-definerbart ep-par $(\phi^n_\sigma, \psi^n_\sigma)$ fra $||\sigma||$ til $||\iota_n||$.

Bevis
Vi bruker induksjon på n.
For $n = 0$, må vi ha $\sigma = \iota$ eller $\sigma = o$. I det første tilfellet bruker vi identitetsfunksjonen begge veier. I det andre tilfellet lar vi $\phi(tt) = 0$, $\phi(ff) = 1$ og $\psi = Z$.
For $n > 0$ vil $\sigma = \sigma_0, \ldots, \sigma_{k-1} \to b$, hvor $b = \iota$ eller $b = o$.
Vi viser hvordan vi håndterer det første tilfellet og overlater til leseren å foreta de enkle justeringene som skal til i det andre tilfellet. Av notasjonsmessige grunner oppfatter vi $||\sigma||$ som $||\sigma_0|| \times \cdots \times ||\sigma_{k-1}|| \to \mathbb{N}$.

1. La
$$\phi^n_\sigma(F)(a) = F(\psi^{n-1}_{\sigma_0}((a)_0), \ldots, \psi^{n-1}_{\sigma_{k-1}}((a)_{k-1}))$$
hvor $F \in ||\sigma||$ og $a \in ||\iota_{n-1}||$.

2. La
$$\psi^n_\sigma(G)(a_0), \ldots, a_{k-1}) = G(\langle \phi^{n-1}_{\sigma_0}(a_0, \ldots, \phi^{n-1}_{\sigma_{k-1}}(a_{k-1}) \rangle)$$
hvor $F \in ||\iota_n||$ og hver $a_i \in ||\sigma_i||$.

Ved induksjon på n og rett frem regning får vi at vi har konstruert et ep-par.

Normalisering

Vi har holdt løftet vårt om ikke å vise Church-Rosser-egenskapen for T.
Dette løftet ga vi fordi det ikke kreves noen nye teknikker for å bevise Church-Rosser-egenskapen i dette tilfellet.
Det stiller seg anderledes med normaliseringsteoremet. Det kreves et betydelig tillegg i beviset for at vi skal kunne gi et teknisk sett holdbart bevis for full normalisering av T. Når vi likevel utelater beviset, er det fordi vi senere skal innføre Girards system F, vi skal vise at T kan reduseres til F, og vi skal vise full normalisering for F. Da kommer vi også til å vise full normalisering for T som et korollar.
Leseren utfordres til å kontrollere at vi ikke bruker full normalisering for T på noe delresultat som deretter brukes til å vise full normalisering av F. Med denne utfordringen i mente, tillater vi oss, midlertidig uten bevis, å slå fast

Teorem 3.20 *I Gödels T er alle termer fullt normaliserbare, det vil si alle omskrivningsekvenser er endelige (med untak av bruk av α-regelen).*

3.3 Konsistente, adekvate og fullt abstrakte modeller

Vi har gitt en tolkning av Gödels T ved å bruke det *fulle typehierarkiet*, det vil si at vi har tolket pil-typer som rommet av alle aktuelle funksjoner. For utypet λ-kalkyle er en slik modell umulig, og når vi senere skal se på typede kalkyler hvor vi har plass til å tolke programmer som ikke terminerer for visse inngangsverdier, er en slik modell uhensiktsmessig.
I dette avsnittet skal vi se på tre egenskaper som kan/kan ikke gjelde for modeller av λ-kalkyler generelt. Vi starter diskusjonen her for T, hvor verden er oversiktlig og uten store komplikasjoner.

Definisjon 3.21 En modell $\mathcal{M}$ for en kalkyle er *konsistent* hvis to termer t og s slik at $t \to s$ alltid tolkes likt.

Teorem 3.22 *Vår modell for Gödels T er konsistent.*

Bevis
Vi viste dette for kjerneeksemplet i Teorem 2.27, og omskrivningsreglene for Rec_σ er i full overenstemmelse med tolkningen, så utvidelsen til T bringer ingen vanskeligheter.

Definisjon 3.23 La C være en kalkyle med basistyper. Et *program* i C vil være en term av basistype.

Denne språkbruken antyder at det er verdier i basistypene vi har interesse av å få som svar, og at termer av andre typer bare er å betrakte som hensiktsmessige mellomledd under utformingen av programmer. Hvis et program har fri variable, kan disse oppfattes som variable for mulige inngangsdata, og dette kan godt være data av mer komplekse typer enn de typene vi ønsker at svarene våre skal ha. Når vi kommer til PCF og andre "programmeringsspråk", kan denne terminologien bli lettere å akseptere.

Definisjon 3.24 La $\mathcal{M}$ være en modell for en kalkyle med basistyper. Vi sier at $\mathcal{M}$ er en *adekvat* modell hvis hver gang t er et lukket program av type b,og t tolkes som et element i $||b||$, så vil t kunne skrives om til en normalform som representerer b ved å bruke omskrivningsreglene i kalkylen.

Når vi senere skal se på modeller hvor vi også tolker ikke-terminerende programmer, skal vi justere denne definisjonen noe.
Det at en modell er konsistent betyr bare at den respekterer omskrivningsreglene, og det er det minste vi kan vente av en modell. Det at modellen er adekvat, sier egentlig at kalkylen i en viss forstand er komplett med hensyn til modellen. Modellen kan ikke fortelle oss at et program burde terminere og gi oss et svar hvis ikke dette allerede kan erkjennes av kalkylen selv.

Teorem 3.25 *Vår modell for Gödels T er adekvat.*

Bevis
La t være et lukket program av type ι eller o. Da vil $[\![t]\!]$ være et naturlig tall eller en Boolesk verdi (hvor vi dropper henvisningen til en tilordning ν som jo ikke bidrar til tolkningen).
Siden t er sterkt normaliserbar, kan t skrives om til en normalform, og siden modellen er konsistent, må dette være normalformen til $[\![t]\!]$.

Det tilsvarende teoremet vil selvfølgelig være en større utfordring når vi innfører typede kalkyler som ikke tillater sterk normalisering. Sterk normalisering er en fin ting å ha når vi skal tolke logikker i typede λ-kalkyler, men er en stor begrensning hvis vi ønsker å tolke algoritmer. Vi vil jo ikke at alle algoritmer skal terminere, det begrenser programmerningsspråkene for mye.

Vi har sett eksempler på lukkede termer t og s av type $\iota \to \iota$ slik at $[\![t]\!] = [\![s]\!]$ uten at $s \equiv t$. Vi viste at vi kan definere Z-funksjonen rekursivt ved hjelp av en term $\hat{Z}$, men at kalkylen ikke kan vise likheten mellom Z og $\hat{Z}$.
Så lenge vi har sterk normalisering, må vi pent finne oss i at det er slik. Likhet mellom rekursivt definerte funksjoner er ikke avgjørbart, mens $\equiv$ er en avgjørbar relasjon, se oppgave 3.5. For termer s og t av typen $\iota \to \iota$ betyr likhet i modellen at $[\![t]\!](n) = [\![s]\!](n)$ for alle n, og det betyr at hvis $[\![t]\!] \neq [\![s]\!]$ kan vi avsløre det i kalkylen ved å finne en n slik at $te_n \not\equiv se_n$.
Hvis imidlertid termene t og s er av type $(\iota \to \iota) \to \iota$, vil $[\![t]\!] \neq [\![s]\!]$ bare bety at det finnes en $f : \mathbb{N} \to \mathbb{N}$ slik at $[\![t]\!](f) \neq [\![s]\!](f)$, og vi har i utgangspunktet ingen grunn til å tro at vi kan finne noen term r som representerer en slik funksjon f.
Vi skal bruke resten av dette avsnittet til å analysere dette problemet nærmere

Definisjon 3.26 La s og t være to lukkede termer av type σ.
Vi sier at s og t er *observabelt like* hvis vi for alle program p av basistype b

og med høyst variabelen $x = x^\sigma$ fri og alle lukkede termer a på normalform og av type b har at

$$p_x^s \to a \Leftrightarrow p_x^t \to a.$$

Når to termer er observabelt like kan den ene termen erstatte den andre termen i alle programmer uten at verdien av programmet endres. Dette svarer til at kalkylen vår ikke kan skille mellom de to termene.

Lemma 3.27 *La s og t være to observabelt like termer av type $\sigma \to b$, hvor b er en av de to basistypene. Hvis r er en lukket term av type σ vil $sr \equiv tr$.*

Bevis
La x være en variabel av type $\sigma \to b$ og la p være programmet xr. La a være normalformen til $tr = p_x^t \equiv p_x^s = sr$. Ekvivalensen holder fordi s og t er observabelt like.

Definisjon 3.28 La $\mathcal{M}$ være en modell for en λ-kalkyle med basistyper. $\mathcal{M}$ er *fullt abstrakt* hvis $[\![s]\!] = [\![t]\!]$ hver gang s og t er observabelt like.

Dette begrepet innebærer at modellen i en viss forstand ikke øver vold mot kalkylen, at den ikke innfører forskjeller som ikke ligger implisitt i kalkylen. I avsnitt 9.1 skal vi se at vår modell ikke er fullt abstrakt. Dette argumentet faller egentlig utenfor læringsmålet til dette kurset, så det er helt frivillig, men ganske opplærende, å sette seg inn i det.
Vi skal se på hvordan vi kan lage en alternativ semantikk som er fullt abstrakt. Den modellen vi skal konstruere er en term-modell, så det er begrenset hvor mye informasjon denne modellen faktisk gir oss om Gödels T. En grunn til at vi tar med denne konstruksjonen og dette beviset er at metodene vi bruker vil komme til nytte senere

Definisjon 3.29 La t_1 og t_2 være to lukkede termer av samme type i T. Vi skriver $t_1 \approx t_2$ for at t_1 og t_2 er observabelt like.

Lemma 3.30 *a) $\approx$ er en ekvivalensrelasjon.*

b) Hvis t og s er lukkede termer slik at $t \to s$, vil $t \approx s$.

c) Hvis $t_1 \approx t_2$ er av type $\sigma \to \tau$ og $s_1 \approx s_2$ er av type σ, vil $t_1 s_1 \approx t_2 s_2$ av type τ.

Bevis
Både a) og b) er trivielle, så vi nøyer oss med å vise c).
La r være en term av basistype med én fri variabel x, og x er av type τ.
Vi må vise at $r_x^{t_1 s_1} \equiv r_x^{t_2 s_2}$.
Vi ser på $r_x^{z s_1}$, bruker at $t_1 \approx t_2$ og får at $r_x^{t_1 s_1} \equiv r_x^{t_2 s_1}$.
Tilsvarende får vi ved å se på $r_x^{t_2 y}$ at $r_x^{t_2 s_1} \equiv r_x^{t_2 s_2}$.
Resten følger fra transitivitet av $\equiv$.

Lemma 3.31 *La t_1 og t_2 være to lukkede termer av type*

$$\sigma = \sigma_1, \ldots, \sigma_n \to b$$

hvor b er en basistype, slik at hver gang $s_1, \ldots, s_n$ er lukkede termer av typer $\sigma_1, \ldots, \sigma_n$ respektive, så vil

$$t_1 s_1 \cdots s_n \equiv t_2 s_1 \cdots s_n.$$

Hvis $s_1 \approx r_1, \ldots s_n \approx r_n$ er lukkede termer av typer $\sigma_n, \ldots, \sigma_n$ respektive, vil

$$t_1 s_1 \cdots s_n \equiv t_2 r_1 \cdots r_n.$$

Bevis
Ved antagelsen har vi $t_1 s_1 \cdots s_n \equiv t_2 s_1 \cdots s_n$.
Så bruker vi antagelsene $s_i \approx r_i$ til etter tur å bytte ut s'ene i det høyre uttrykket med r'er.

Dette lemmaet inspirerer oss til å se på en annen naturlig relasjon mellom lukkede termer av samme type, ekstensjonal likhet:

Definisjon 3.32 Vi definerer relasjonen $\sim$, *ekstensjonalt like*, mellom lukkede termer av samme type ved:

1. Hvis s og t er av basistype, vil $s \sim t$ hvis s og t har den samme normalformen.
2. Hvis s og t er av type $\sigma \to \tau$, vil $s \sim t$ hvis vi for alle s' og t' av type σ har at om $s' \sim t'$ vil $ss' \sim tt'$

Det er mulig å vise direkte ved induksjon på nivået til typen at $\sim$ er en ekvivalensrelasjon. Symmetri er trivielt og transitivitet er enkelt. Det er refleksivitet som skaper det største problemet. Det er imidlertid lett å vise at for alle konstanter C på nivå 0 eller 1, så vil $C \sim C$. Merk også at om $s \sim t$ og $s \to s'$, vil $s' \sim t$. Dette følger rett fra definisjonen.

Lemma 3.33 *La τ være en type. Da vil $Rec_\tau \sim Rec_\tau$*

Bevis
Rec_τ er av type $\delta = \tau, (\iota, \tau \to \tau), \iota \to \tau$, og følger vi definisjonen av $\sim$ for lukkede termer av type δ tre trinn tilbake, ser vi at utfordringen vil være å vise at

$$Rec_\tau s_1 s_2 s_3 \sim Rec_\tau t_1 t_2 t_3$$

når $s_1 \sim t_1$ er lukkede termer av type τ, $s_2 \sim t_2$ er lukkede termer av type $\iota, \tau \to \tau$ og $s_3 \sim t_3$ er lukkede termer av type ι.
s_3 og t_3 vil reduseres til den samme numeralen e_n. Deretter viser man påstanden ved induksjon på n. Induksjonstarten følger fra antagelsen om at $s_1 \sim t_1$ og induksjonsskrittet følger fra induksjonsantagelsen og antagelsen om at $t_2 \sim s_2$.

Lemma 3.34 *La p være en term med fri variable blant $x_1, \ldots, x_n$ av typer $\sigma_1, \ldots, \sigma_n$, og for hver $i = 1, \ldots, n$, la $s_i \sim t_i$ være lukkede termer av type σ_i. Vi bruker notasjonen $\vec{x}$, $\vec{s}$ og $\vec{t}$ for disse sekvensene.*
Da er

$$p^{\vec{s}}_{\vec{x}} \sim p^{\vec{t}}_{\vec{x}}.$$

Bevis
Vi bruker induksjon på oppbygningen av p.
Hvis p er n av konstantene $0, tt, ff, S, P, Z.\ \supset_\iota, \supset_o$ eller Rec_τ for en type τ, har vi allerede observert at $p \sim p$.
Hvis p er en variabel x_i følger påstanden fra antagelsen om at $s_i \sim t_i$
Det gjenstår å se på de to formene for sammensatte utsagn:

1. Tilfelle $p = qr$.
   Ved induksjonsantagelsen vil $q^{\vec{s}}_{\vec{x}} \sim q^{\vec{t}}_{\vec{x}}$ og $r^{\vec{s}}_{\vec{x}} \sim r^{\vec{t}}_{\vec{x}}$, og påstanden følger fra definisjonen av $\sim$.

2. Tilfelle $p = \lambda x.q$ hvor x er av type σ.
   La $s \sim t$ være lukkede termer av type σ. Vi må vise at

   $$p^{\vec{s}}_{\vec{x}} s \sim p^{\vec{t}}_{\vec{x}} t.$$

   Men induksjonsantagelsen for q sier oss at

   $$q^{s,\vec{s}}_{x,\vec{x}} \sim q^{t,\vec{t}}_{x,\vec{x}},$$

   hvilket, via β-regelen, er det vi skal vise.

Teorem 3.35 *For alle lukkede termer t og s vil*

$$t \sim s \Leftrightarrow t \approx s.$$

Bevis
Først, la $t \sim s$ være lukkede termer av type σ og la p være en term av basistype med høyst variabelen x av type σ som fri.
Ved Lemma 3.34 vil $p_x^s \sim p_x^t$. For termer av basistype, betyr dette at de har samme normalform. Det følger at s og t er observabelt like, det vil si at $s \approx t$.
Anta nå at $s \approx t$. La $\sigma = \sigma_1, \ldots, \sigma_n \to b$ være typen til s, hvor b er en basistype, og la $s_1 \sim t_1, \ldots s_n, \sim t_n$ være lukkede termer av typer $\sigma_1, \ldots, \sigma_n$ respektive.
Ved første del av beviset vil vi for alle i med $1 \leq i \leq n$ ha at $s_i \approx t_i$, og ved Lemma 3.30 c) har vi da at $ss_1 \cdots s_n \equiv tt_1 \cdots t_n$.
Dette viser at $s \sim t$.

Når vi skal lage en modell for typet λ-kalkyle er det naturlig (men ikke strengt tatt nødvendig) å tolke piltyper $\sigma = \tau \to \delta$ som mengder av funksjoner. Teorem 3.35 gjør det mulig å konstruere en slik modell hvor alle objektene er tolkninger av lukkede termer, og hvor to lukkede termer tolkes likt hvis og bare hvis de er observabelt like. Dette er en type konstruksjon man finner mye av i semantikk for typeteori.

Definisjon 3.36 Ved en simultan rekursjon på oppbyggingen av en type σ definerer vi $|\sigma|$ og ρ_σ, hvor ρ_σ avbilder mengden av lukkede termer av type σ surjektivt på $|\sigma|$ som følger:

1. Hvis $\sigma = \iota$, lar vi $|\sigma| = \mathbb{N}$, og vi lar $\rho_\sigma(t) = n$ hvis normalformen til t er e_n.

2. Hvis $\sigma = o$, lar vi $|\sigma| = \mathbb{B}$, og vi lar $\rho_\sigma(t)$ være sannhetsverdien som svarer til normalformen til t.

3. Hvis t er en lukket term av type $\sigma \to \tau$ og $a \in |\sigma|$, lar vi s være en lukket term av type σ slik at $\rho_\sigma(s) = a$.
   Da lar vi $\rho_{\sigma\to\tau}(t)(a) = \rho_\tau(ts)$.
   Vi lar $|\sigma \to \tau|$ være mengden av $\rho_{\sigma\to\tau}$ definert på denne måten.

I oppgave 3.6 utfordres leseren til å vise hvordan vi kan tolke vilkårlige termer inn i denne modellen, og til å vise at modellen er fullt abstrakt for T. I avsnitt 9.1 skal vi se at den modellen vi så på i første omgang ikke er fullt abstrakt. Når vi har innført domeneteori vil vi være i stand til å beskrive en modell som både er fullt abstrakt og naturlig.

3.4 Oppgaver

Oppgave 3.1 Finn lukkede termer i Gödels T som svarer til addisjon og multiplikasjon.

Oppgave 3.2 Fullfør Eksemplene i 3.7 ved å vise at vi får de samme overgangsreglene.

Oppgave 3.3 a) Vis at det finnes et predikat på type ι som gir verdien tt nøyaktig for partallene.

b) Drøft følgende uformelle påstand:

> *Klassen av T- definerbare delmengder av $\mathbb{N}^k$ er lukket under Booleske operasjoner*

c) Drøft følgende uformelle påstand:

> *Klassen av T-definerbare delmengder av $\mathbb{N}^k$ er lukket under begrensede kvantorer.*

Oppgave 3.4 Vis at det finnes funksjoner π_1 og π_2 fra $\mathbb{N}$ til $\mathbb{N}$ slik at $n = \langle \pi_1(n), \pi_2(n) \rangle$ for alle n.
Vis at π_1 og π_2 er definerbare i Gödels T.

Oppgave 3.5 Bruk Church-Rosser-egenskapen og full normalisering til å vise at $\equiv$ i Gödels T er en avgjørbar relasjon.

Oppgave 3.6 Vis hvordan vi kan modifisere definisjonen av tilordninger til tolkninger av variable av type σ som elementer i $|\sigma|$ fra Definisjon 3.36.
Gi den alternative definisjonen av $[\![t]\!](\nu) \in |\sigma|$ for alle termer av type σ.
Det vil være en utfordring å vise at $[\![\lambda x^\sigma.s]\!](\nu) \in |\sigma \to \tau|$ når s er en term av type τ. Du må derfor formulere induksjonsantagelsen om tolkningen som gjør dette mulig.
Tilslutt, vis at modellen er fullt abstrakt.

Hvis du har valgt den rette definisjonen, vil du få at $[\![t]\!](\nu)$ er ekvivalent til termen vi får ved å substituere $\nu(i)$ inn for frie forekomster av x_i i t for alle $i \in \mathbb{N}$.

Kapittel 4

System F

4.1 Diskusjon

System F, eller *2. ordens typeteori*, er en formell kalkyle utviklet av Girard, og presentert første gang rundt 1970 på den skandinaviske logikkonferansen i Oslo. Opprinnelig var System F et bidrag til bevisteorien, en formell kalkyle som skulle muliggjøre bevisteoretisk analyse av sterke logiske systemer.
Ganske raskt ble System F gjenstand for oppmerksomhet fra teoretisk databehandling. Litt av grunnen til dette er at mange program eller prosedyrer kan være felles for data av mange forskjellige typer. For eksempel vil vi ha en identitetsfunksjon $\lambda x^\sigma.x^\sigma$ for hver type i endelig typeteori, og hver gang R er en binær relasjon på en mengde X, gir algoritmer for å sortere endelige delmengder av X ut fra relasjonen R mening, endog forstandig mening hvis R er en total ordning. Slike prosedyrer kan formuleres ved hjelp av termer hvor det forekommer variable for typen og for den ordningen vi vil sortere etter.

En annen motivasjon for å studere System F kommer fra observasjonen av at mange datatyper er induktivt bygget opp. Eksempelvis er $\mathbb{N}$ generert fra 0 ved hjelp av etterfølgeroperasjonen, mengden av binære uttrykk er generert fra det tomme ordet ved de to etterfølgeroperasjonene "føy til 0" og "føy til 1", og mengden av binære trær er generert fra det tomme treet ved hjelp av "ordnet par"-konstruksjonen. I alle disse tilfellene snakker vi om "den minste mengden slik at . . .", og System F vil være en formell kalkyle hvor vi kan uttrykke denne formen for definisjoner som konstruksjoner av nye typer. Dette skal vi komme tilbake til.

System F vil være så uttrykskraftig at numeraler, sannhetsverdier, konkatenering av sekvenser, addisjon, booleske operasjoner og mye annet kan beskrives ved hjelp av lukkede termer. Det er videre så uttrykskraftig at mange av de datatypene vi måtte bli interesserte i kan defineres i system F. Dette vil vi komme tilbake til gjennom eksempler. Derfor vil System F ikke ha konstanter, hverken for typer eller for objekter. Til tross for denne uttrykskraften vil vi vise at System F oppfyller sterk normalisering. Beviset for sterk normalisering må nødvendigvis være komplekst i den forstand at vi må bruke et rimelig stort fragment av mengdelæren for å kunne gjennomføre det. Det er faktisk slik at konsistens av annenordens tallteori følger som en konsekvens av sterk normalisering av System F, og da sier Gödels 2. ufullstendighetsteorem at sterk normalisering ikke lar seg bevise i 2. ordens tallteori.

4.2 Kalkylen

Språket til System F består av to deler, termer for typer og termer for typede objekter. I endelig typeteori vil vi ha typekonstanter for basistypene, typisk ι og o i kjerneeksemplet. I System F vil vi ikke trenge typekonstanter.

Definisjon 4.1 *Typene i System F* er gitt ved

1. En uendelig liste $X_1, X_2, \ldots\ldots$ av *typevariable.*
   (Vi vil bruke store latinske bokstaver X, Y langt bak i alfabetet som typevariavle.)

2. Hvis U og V er typer, er $U \to V$ en type.
   (Vi vil bruke store latinske bokstaver U, V etc. ikke fullt så langt bak i alfabetet for vilkårlige typer)

3. Hvis U er en type og X er en typevariabel, vil $\prod X.U$ være en type.

Vi må bygge litt intuisjon rundt denne definisjonen.
Vi er vant til å arbeide med variable i andre sammenhenger. Intuitivt skal en type være en mengde av en viss interesse for oss, uten at vi ønsker å presisere hva denne "interessen" skal være. En typevariabel vil da betegne en ubestemt mengde av interesse.
Hvis U og V er typer, vil U og V betegne bestemte eller ubestemte mengder. $U \to V$ betegner da den tilsvarende bestemte eller ubestemte mengden av funksjoner fra tolkningen av U til tolkningen av V. Eksempelvis vil $X \to Y$

være en type som beskriver et generelt funksjonsrom.
For hver mengde X har vi en identitetsfunksjon $id_X : X \to X$ som halvformelt kan defineres som

$$id_X = \lambda x \in X.x.$$

Vi ser at vi kan oppfatte id som en funksjon som til en type X gir oss identitetsfunksjonen id_X på denne typen. Dette er poenget med

$$\prod X.X \to X.$$

Denne typetermen har som intuitiv tolkning alle funksjoner som til en vilkårlig type X gir oss en funksjon fra X til X.

La oss se på et annet eksempel, Church-numeralen $k_2 = \lambda y \lambda x.y(y(x))$.
Hvis vi nå lar x variere over en mengde X, må y variere over $X \to X$ for at dette skal gi mening. For alle typer X kan vi imidlertid se på k_2 begrenset til X definert ved

$$\lambda y \in (X \to X)\lambda x \in X.y(y(x)).$$

Siden vi har beskrevet et objekt $k_2^X \in (X \to X) \to X$ for hver type X har vi i prinsippet beskrevet et objekt i $\prod X.(X \to X) \to X$.
Språket for typede termer i System F skal fange opp denne typen konstruksjoner, og kalkylen skal respektere den intuisjonen vi har lagt til grunn.

Vi kan også legge merke til at $\prod$ er det matematiske symbolet vi bruker når vi skal uttrykke et produkt, det brukes ofte helt analogt med summetegnet $\sum$. Typevariabelen X vil være lukket i $\prod X.U$. Dette gir oss igjen en standard definisjon av fri og lukkede variable i typer, og med det, definisjonen av at en type V er substituerbar for en variabel X i typen U. Vi overfører vår tidligere notasjon til denne situasjonen, vi lar

$$U_X^V$$

stå for resultatet av å erstatte alle fri forekomster av X i U med typen V, hvor vi alltid antar substituerbarhet uten alltid å presisere det.

Det er nå på tide å definere *termene* i System F:

Definisjon 4.2 De *typede termene* i System F er gitt som følger:

1. For hver type U har vi en uendelig liste $x_1^U, x_2^U, \ldots$ av variable av type U.

(Vi vil bruke x^U, y^U etc. for typede variable. Vi kan merke oss at U forekommer som et deluttrykk av x^U slik at en typevariabel kan forekomme i en typet variabel i fri eller bunden form.)

2. Hvis t er en term av type $U \to V$ og s er en term av type U, er (ts) en term av type V.
   (Vi vil ikke sette typen til termen i superskript på termen, ettersom superskript er en formell del av språket her, og bare brukes i tilknytning til de variable.)

3. Hvis U er en type, x^U er en variabel og t er en term av type V, så er $(\lambda x^U.t)$ en term av type $U \to V$.

4. Hvis v er en term av type V, vil $\Lambda X.v$ være en term av type $\prod X.V$ såfremt variabelen X ikke er fri i typen til en fri variabel i v.
   (Vi skal kommentere denne delen av definisjonen senere.)

5. Hvis t er en term av type $\prod X.V$ og U er en type, så er (tU) en term av type V^U_X.

Vi vil bruke de samme konvensjonene for å utelate parenteser som tidligere. Merk at vi har to typer applikasjonstermer, i noen tilfeller gir det mening å anvende et objekt på et annet objekt, mens i andre tilfeller gir det mening å anvende objektet på en type.

Eksempler 4.3 a) Vi har sett på hvordan identitetsfunksjonen *id* kan oppfattes som en funksjon som til enhver mengde X gir oss identitetsfunksjonen på X. Dette uttrykker vi i språket til system F ved

$$v = \Lambda X.\lambda x^X.x^X \text{ er en term av type } V = \prod X.X \to X$$

Her er V en *lukket type*, det vil si en type uten fri variable og v er en lukket term. Merk at en term kan inneholde både fri typevariable og fri objektvariable. De fri typevariablene vil alltid være fri i typene til fri eller bundne objektvariable.

b) Hva skal vi mene med applikasjon av typer, det vil si, hvordan skal vi forstå punkt 5?
Hvis vi nå lar $v = \Lambda X.\lambda x^X.x^X$ og $U = Y \to Y$, så skal vU være en term av type $(X \to X)^{Y \to Y}_X$, det vil si av type $(Y \to Y) \to (Y \to Y)$. Den intuitive tolkningen skal være at vU skal betegne identitetsfunksjonen på $(Y \to Y)$, og det skal vi fange opp ved hjelp av omskrivningsreglene.

c) I punkt 4. hadde vi et krav om at variabelen X ikke er fri i typen til en fri variabel i v. Det betyr at vi må lukke alle objektvariable x^U som inneholder en typevariabel X før vi kan lukke X.
La oss se hva som kan gå galt hvis vi gjør det omvendt:
La t være den syntaktisk ulovlige termen $\Lambda X.x^X$. Intuisjonen vår sier at dette skal betegne en funksjon som til enhver type X plukker ut det ubestemte elementet x^X i X. Hva er så typen til variabelen x^X i dette uttrykket? Den er ikke-eksisterende, ettersom den er abstrahert bort. I en viss forstand vil x^X forekomme i halvfri tilstand i denne termen, som objektvariabel er den fri, mens vi har bundet variasjonsområdet til objektvariabelen ved å binde typevariabelen X.

Til punkt c) over er det å bemerke at hvis vi tillater abstraksjon av typevariable som forekommer fritt i fri objektvariable, og benytter oss uhemmet av β-regelen, så vil vi kunne embedde utypet λ-kalkyle inn i denne kalkylen, og da er det ikke noe poeng med den. Vi skal ikke gjennomføre beviset for dette.

Det gjenstår å gi omskrivningsreglene. Vi får to varianter av α-regelen og to varianter av β-regelen. Flere regler trenger vi ikke:

Definisjon 4.4 Omskrivningsreglene for System F er gitt ved

1. $\lambda x^U.t \to \lambda y^U.t^{y^U}_{x^U}$.
2. $\Lambda X.t \to \Lambda Y.t^Y_X$.
3. $(\lambda x^U.t)u \to t^u_{x^U}$.
4. $(\Lambda X.t)U \to t^U_X$.

Omskrivningsreglene skal kunne brukes på subtermnivå, men vi forutsetter at om en av omskrivningsreglene brukes på en subterm vil ingen fri variable i den termen vi setter inn ende som bundne variable i den større sammenhengen. *System F* er definert som samlingen av språk og omskrivningsregler gitt over.

I det neste avsnittet skal vi implementere Gödels T i System F. Dette avsnittet vil da også tjene som en eksempel-samling for hvordan vi regner innenfor System F.

4.3 Implementering av Gödels T

For å kunne implementere Gödels T i System F må vi finne to lukkede typer som svarer til de naturlige tallene og til de booleske verdiene. Videre må vi finne lukkede termer som svarer til konstantene 0, tt, ff, S, P, Z, $\supset_\iota$ og $\supset_o$. Til slutt bør vi konstruere en lukket term Rec av en type som vi vil presisere når tiden er inne.
Disse tolkningene må være slik at alle enkeltomskrivninger i T kan realiseres, via et endelig antall omskrivninger, i F. Dette vil ha sterk normalisering for T som konsekvens av sterk normalisering for F, noe vi skal se på i det neste avsnittet.

Vi minner om Church-numeralene $k_n = \lambda y \lambda x.y^n(x)$. Dette er en term i utypet λ-kalkyle, men vi ser at om vi begrenser x til en vilkårlig mengde X, så gir dette mening når y varierer over funksjonsrommet $X \to X$.
Vi kan derfor si at k_n har instanser av type $(X \to X), X \to X$ for alle mengder, eller typer, X.
Dette frister oss til følgende definisjon

Definisjon 4.5 La

$$NAT = \prod X.(X \to X), X \to X.$$

La

$$k_n = \Lambda X.\lambda y^{X \to X} \lambda x^X.y^n(x)$$

for alle naturlige tall n.

Det er et par bemerkninger å komme med til denne definisjonen. For det første skriver vi $(X \to X), X \to X$ i stedet for det syntaktisk riktigere $(X \to X) \to (X \to X)$. Dette er fordi vi overfører konvensjonen om å erstatte visse forekomster av $\to$ med komma-notasjon.
For det andre har vi droppet typesupskripten på noen forekomster av variablene. Så lenge vi bruker forskjellige latinske bokstaver for de forskjellige variablene, er det tilstrekkelig å presisere typen ved første gangs nevnelse, og det vil øke lesbarheten i det minste noen ganger hvis vi utelater typen enkelte steder.
Det er fullt mulig å realisere etterfølgerfunksjonen ved å tilpasse konstruksjonen fra utypet λ-kalkyle til System F:

Lemma 4.6 *La S være den lukkede termen av type $NAT \to NAT$ definert ved*

$$S = \lambda u^{NAT}.\Lambda X.\lambda y^{X \to X} \lambda x^X.y(uXyx).$$

Da vil $Sk_n \to k_{n+1}$ for alle n.

Bevis
Siden dette også skal tjene som et eksempel, kontrollerer vi at typingen av S er riktig. Senere vil vi ikke være så nøye med det, siden feil typing vil bli avslørt når vi prøver å evaluere en term på en annen term.
Siden vi starter med en abstraksjon av en NAT-variabel u, er S av type $NAT \to V$ hvor V er typen til $\Lambda X.\lambda y^{X \to X} \lambda x^X.y(uXyx)$. Vi ser at prefikset $\Lambda X.\lambda y^{X \to X} \lambda x^X$ til V stemmer overens med prefikset til NAT.
Vi trenger at matriks $y(uXyx)$ er riktig typet og av type X. La X være en vilkårlig type. Siden u er av type NAT vil uX være av type $(X \to X), X \to X$. y er av type $X \to X$ og x av type X. Det gir at typen til uXy er $X \to X$ og at typen til $uXyx$ er X. Til slutt ser vi at typen til $y(uXyx)$ da må bli den ønskede X.

Nå er det bare å verifisere lemmaet ved direkte utregning. Hver gang er det én bruk av β-regelen som kan brukes, i form av regel 3. eller regel 4.
Vi får
$Sk_n = (\lambda u^{NAT}.\Lambda X.\lambda y^{X \to X} \lambda x^X.y(uXyx))k_n$
$\to \Lambda X.\lambda y^{X \to X} \lambda x^X.y(k_n Xyx)$
$= \Lambda X.\lambda y^{X \to X} \lambda x^X.y((\Lambda X.\lambda y^{X \to X} \lambda x^X.y^n(x))Xyx)$
$\to \Lambda X.\lambda y^{X \to X} \lambda x^X.y(y^n(x)) = k_{n+1}$.

Definisjon 4.7 La $BOOLE = \prod X.X, X \to X$
La $k_{\mathbf{T}} = \Lambda X.\lambda y^X \lambda x^X.y$ og la $k_{\mathbf{F}} = \Lambda X.\lambda y^X \lambda x^X.x$.

Vi kan merke oss at Churchnumeralen for 0 og λ-termen for $\mathbf{T}$ var like, mens nå er typene NAT og $BOOLE$ forskjellige, så de kan ikke ha sammenfallende termer.
Vi ser at hvis U er en type, s og t termer av type U, så vil

- $k_{\mathbf{T}}Ust \to s$
- $k_{\mathbf{F}}Ust \to t$

Dette bruker vi til å definere

$$\supset = \Lambda X.\lambda z^{BOOLE} \lambda y^X \lambda x^X.zXyz.$$

Vi må fortsatt finne termer i F som svarer til Z, P og Rec_U for hver type U. Ettersom vi faktisk kan definere P ved rekursjon, vil vi nøye oss med Z og Rec_U, og vi vil se på sammenhengen mellom vår behandling av Rec og Kleene's definisjon av P i utypet λ-kalkyle.

Vi definerte en versjon av Z for utypet λ-kalkyle og vår definisjon i System F vil være en tilpasning av den definisjonen.

Definisjon 4.8 Vi definerer termen Z av type $NAT \to BOOLE$ som følger:

$$Z = \lambda z^{NAT} \Lambda X.z(X, X \to X)c_{k_{\mathbf{F}}X} k_{\mathbf{T}}$$

hvor $c_{k_{\mathbf{F}}X} = \lambda w^{X,X \to X}.k_{\mathbf{F}}X$.

Det krever litt regning å se at denne termen gjør det den skal, poenget er fortsatt at om vi itererer konstantfunksjonen **F** ingen ganger på **T** får vi **T**, mens hvis vi bruker denne konstantfunksjonen en eller flere ganger får vi **F**.

La oss nå gå over til rekursjon. Numeraler er termer som representerer iterasjon, og hvis vi se på den enkleste formen for rekursjon er det direkte iterasjon vi bruker. La oss se igjen på

$$k_n = \Lambda X.\lambda y^{X \to X} \lambda x^X.y^n(x).$$

Hvis U er en mengde, $f : U \to U$ og $a \in U$, er den intuitive tolkningen av

$$n \mapsto k_n U f a$$

funksjonen $g : \mathbb{N} \to U$ definert ved

- $g(0) = a$
- $g(n+1) = f(g(n))$

Problemet i forhold til den generelle definisjonen av Rec er at her kommer også n inn som argument. For å håndtere det, trenger vi å iterere etterfølgerfunksjonen som en del av den funksjonen vi bruker til rekursjon, og for å få til det, trenger vi ordnede par som i Kleene's beskrivelse av P.
Selv om vi ikke har noe behov for å representere kartesiske produkter for å kunne representere typene i T, har vi altså behov for å gjøre det for å kunne representere Rec.

Definisjon 4.9 La U og V være typer i F.
La $U \times V$ være typen

$$U \times V = \prod X.(U, V \to X) \to X.$$

Vi definerer *projeksjonsfunksjonene* π^1 og π^2 ved

$$\pi^1 t = tU(\lambda x^U \lambda y^V.x)$$

og

$$\pi^2 t = tV(\lambda x^U \lambda y^V.y).$$

Vi definerer *paringsfunksjonen* ved

$$\langle u, v\rangle = \Lambda X.\lambda x^{U,V\to X} xuv.$$

Vi kunne selvfølgelig uttrykt π^1 og π^2 som lukkede termer $\lambda z^{U\times V}.\cdots$ og ikke via disse omskrivningsreglene. En tilsvarende kommentar gjelder for paringsfunksjonen.
Ved å sette inn og regne ut får vi

- $\pi^1\langle u, v\rangle \to u$
- $\pi^2\langle u, v\rangle \to v$.

Utregningen overlates til leseren som en passelig vanskelig Oppgave 4.3.
Omvendingen gjelder ikke, vi kan ikke utlede t fra $\langle \pi^1 t, \pi^2 t\rangle$. Vi kan tolke dette som at $\langle\cdot,\cdot\rangle$ sammen med π^1 og π^2 er en form for *ep*-par fra det virkelige kartesisske produktet til det formelle. Denne mangelen på formell bijektivitet gjør imidlertid at vi må være litt forsiktige når vi resonnerer rundt paring og projeksjoner, ikke alle resonnementer vi intuitivt vil synes virker gjør det.

Vi kan merke oss at $\langle\cdot,\cdot\rangle$ er definert for hver U og V, men på en uniform måte. Skal vi gjøre dette mer korrekt enn i [1] bør vi enten sette på en indeks U, V eller definere lukkede termer som alternativer til paring og projeksjoner. I Oppgave 4.4 utfordres leseren til å gjøre dette selv.

Vi skal nå gå tilbake til tolkning av rekursjonsoperatoren, og vi vil definere Rec_U for en vilkårlig type U. Den lukkede termen Rec blir da $\Lambda Z.Rec_Z$ av type

$$\prod Z.Z, (NAT, Z \to Z), NAT \to Z.$$

Vi tar utgangspunkt i rekursjonslikningen

- $Rec_U(a, f, 0) = a$
- $Rec_U(a, f, n + 1) = f(n, Rec_U(a, f, n))$

Hvis vi ser på $\Phi(a, f, n) = \langle n, Rec_U(a, f, n)\rangle$, ser vi at vi kan bruke følgende rekursjonslikning:

- $\Phi(a, f, 0) = \langle 0, a\rangle$
- $\Phi(a, f, n + 1) = \langle \pi^1(\Phi(a, f, n)), f(\pi^1(\Phi(a, f, n)), \pi^2(\Phi(a, f, n)))\rangle$.

Det vi har oppnådd med omskrivningen er å lage en rekursjonslikning hvor argumentet n ikke brukes aktivt.
Hvis vi nå lar a være en term av type U og f en term av type $NAT, U \to U$, lar vi t være termen

$$t = \lambda x^{Nat\times U}\langle \pi^1 x, f(\pi^1 x)(\pi^2 x)\rangle$$

og la

$$Rec_U a f k_n = k_n\langle k_0, a\rangle t.$$

Igjen overlater vi det til leseren å skrive om disse uttrykkene til en lukket term av passende type som evalueres i tråd med definisjonen over.

Ytterligere verifikasjoner bør gjennomføres av den leseren som ønsker en grundig forståelse av hva som foregår. I Oppgave 4.5 tilbys leseren å regne på et par eksempler, hvor det vil fremgå at vi alltid kombinerer paring og projeksjoner i den rette rekkefølgen. Her nøyer vi oss med å konkludere

Teorem 4.10 *Alle typer og typede termer i Gödels T kan omskrives til lukkede typer og til tilsvarende typede termer i System F slik at alle omskrivninger i T kan realiseres som omskrivninger i System F.*

Korollar 4.11 *Hvis alle termer i F er sterkt normaliserbare, vil det samme gjelde for T.*

4.4 Normalisering

Vi minner om at en term i en kalkyle er sterkt normaliserbar hvis enhver kjede av omskrivninger av termen er endelig, dersom vi ser bort fra utidig bruk av α-regelen. I dette avsnittet skal vi vise at alle termene i System F

er sterkt normaliserbare.
Vi har vist sterk normalisering for elementær typet λ-kalkyle. I det beviset definerte vi en klasse R_σ ved induksjon på typen. Definisjonen av R_σ var at R_σ består av alle sterkt normaliserbare termer når σ er en basistype, mens for funksjonstyper $\sigma = \tau \to \delta$ består R_σ av alle termer som "sender" et element i R_τ på et element i R_δ via den syntaktiske applikasjonsoperatoren. I en språkbruk som vi skal bli mer fortrolig med senere innebærer det at R_σ består av alle "totale" objekter, hvor det er de sterkt normaliserbare termene av basistype som regnes som de totale objektene i de typene.
Deretter viste vi egenskapene CR 1, CR 2, CR 3 og den logiske konsekvensen CR 4 av CR 3 for R_σ ved induksjon på oppbyggingen av σ, og brukte det til å vise at alle termer er sterkt normaliserbare.
Intuitivt sett skulle ikke utvidelsen til Gödels T bringe noen ekstra prinsipielle problemer her, typene er de samme, vi kunne brukt den samme definisjonen av R_σ og siden et poeng med rekursjon er at vi har full kontroll over hvordan vi evaluerer et rekursjonsuttrykk, vil sluttføringen av beviset kreve mer teknisk enn idémessig innsats.

Når det gjelder System F er situasjonen adskillig mer kompleks. Her er ikke typene bygget opp induktivt i samme forstand. Hvis vi ser på typen $U = \prod X.X \to X$ og termen $t = \Lambda X.\lambda x^X.x$ av type U, tillater kalkylen uttrykket tU. Det betyr at det finnes termer som kan appliseres på sin egen type, eller endog på typer som er langt mer komplekse enn typen til termen. I dette eksemplet er det lett å se at termen t er sterkt normaliserbar,

$$tU \to \lambda x^U.x$$

er den eneste mulige omskrivningskjeden, men det er ingenlunde opplagt at vi ikke kan snike inn uendelige omskrivningskjeder ved å applisere termer på stadig mer komplekse typer. Det som i alle fall er opplagt er at vi ikke kan definere noen form R_σ for totalitet på typene, vi har ikke basistyper, og vi har ikke at applikasjon reduserer kompleksiteten på typene.

Idéen bak beviset er å studere klassen av alle kandidater til en slik totalitet, alle utvalg av termer som oppfyller CR 1 - CR 3.
Vi minner om at en term er *nøytral* dersom den ikke starter med en abstraksjon. I tilfelle av System F er det termer på formen x, tu og tU. Siden resten av dette avvsnittet bare omhandler System F, vil vi ikke presisere det hver gang.

Definisjon 4.12 La U være en type.
En *reduserbarhetskandidat for typen* U er en mengde R av termer av type U slik at

CR 1 Hvis $t \in R$ er t sterkt normaliserbar.

CR 2 Hvis $t \in R$ og $t \to t'$ vil $t' \in R$

CR 3 Hvis t av type U er nøytral, og hver gang vi bruker en av β-reglene én gang på t får vi en term t' i R, så er $t \in R$.

Som før får vi at om R er en reduserbarhetskandidat for typen U vil

CR 4 Hvis t av type U er nøytral og normal, er $t \in R$.

Denne siste observasjonen gir oss at en reduserbarhetskandidat aldri er tom, variablene av type U er alltid med.

Lemma 4.13 *La T være en type, og la N_T være mengden av sterkt normaliserbare termer av type T. Da er N_T en reduserbarhetskandidat for T.*

Bevis
CR 1 er trivielt.
CR 2 er nesten like trivielt, ettersom om vi skriver om en sterkt normaliserbar term, er resultatet også sterkt normaliserbart.
CR 3 er også å betrakte som trivielt, ettersom dersom enhver enkeltomskrivning leder til en sterkt normaliserbar term, må utgangspunktet være sterkt normaliserbart.

Vi minner om at når en term t er sterkt normaliserbar, finnes det en øvre grense for hvor mange omskrivninger (utenom α-reglene) vi kan bruke før vi når en normalform.

Ved CR 1 er dette det største mulige valget av reduserbarhetskandidater for de forskjellige typene. Hvis vi derfor kan vise at alle termer t av type T ligger i minst en reduserbarhetskandidat for T, er vi fremme. Vi skal gjøre dette ved å definere en reduserbarhetskandidat for hver type T som vil inneholde alle termene av type T. Dette blir nødvendigvis N_T, noe som ikke er opplagt fra start av. I det følgende kan det hjelpe å tenke på en reduserbarhetskandidat som et forslag til hvilke objekter av type T som skal gjelde som totale. Den første definisjonen følger da formatet til definisjonen av totalitet for funksjonsrom.

Definisjon 4.14 La S og R være reduserbarhetskandidater for typene U hhv. V.
La t være en term av type $U \to V$.
Vi lar $t \in S \to R$ hvis $ts \in R$ hver gang $s \in S$.

Lemma 4.15 *$S \to R$ som definert over vil være en reduserbarhetskandidat.*

Bevis
Vi kan kopiere beviset for 2. i Lemma 2.15.

Vi skal nå prøve å gi mening til at en term av $\prod$-type er total. Intuitivt vil t av type $\prod X.U$ være total hvis vi for alle typer V og alle totaliteter R på V har at tU blir total i den nedarvede totaliteten på U_X^V. Det er dette vi vil fange opp i punkt 3. under:

Definisjon 4.16 La T være en type med fri typevariable blant $\vec{X} = X_1, \ldots, X_n$
La $\vec{U}$ være en sekvens av typer av samme lengde som $\vec{X}$ og la $\vec{R}$ være reduserbarhetskandidater for typene i $\vec{U}$.
Ved induksjon på T definerer vi en mengde termer $T_{\vec{X}}^{\vec{R}}$ av type $T_{\vec{X}}^{\vec{U}}$ son følger:

1. Hvis $T = X_i$ er $T_{\vec{X}}^{\vec{R}} = R_i$
2. Hvis $T = U \to V$ er $T_{\vec{X}}^{\vec{R}} = U_{\vec{X}}^{\vec{R}} \to V_{\vec{X}}^{\vec{R}}$ slik som definert over.
3. Hvis $T = \prod Y.V$ lar vi $t \in T_{\vec{X}}^{\vec{R}}$ hvis t er en term av type $T_{\vec{X}}^{\vec{U}}$ og vi for alle typer W og alle reduserbarhetskandidater S for W har at $tW \in V_{Y,\vec{X}}^{S,\vec{R}}$

Bemerkning 4.17 Det er denne definisjonen som ikke kan formaliseres i 2. ordens tallteori. For hver type T er $T_{\vec{X}}^{\vec{R}}$ definerbar, men for hver forekomst av $\prod$ må vi til med en ekstra kvantor over mengden av reduserbarhetskandidater, så det er ingen begrensning på hvor lange 2. ordens prefikser vi trenger i denne definisjonen.

Lemma 4.18 *$T_{\vec{X}}^{\vec{R}}$ som definert over vil alltid være en reduserbarhetskandidat.*

Bevis
Vi trenger bare å se på tilfelle 3, hvor $T = \prod Y.V$.

CR 1 La $t \in T^{\vec{R}}_{\vec{X}}$. La W være en vilkårlig type og la $S = N_W$, det vil si reduserbarhetskandidaten av sterkt normaliserbare termer.
Pr. definisjon av $T^{\vec{R}}_{\vec{X}}$ vil $tW \in V^{S,\vec{R}}_{Y,\vec{X}}$, og ved induksjonsantagelsen, CR 1 for V, er tW sterkt normaliserbar. Siden enhver reduksjonsfølge fra t kan skrives om til en like lang reduksjonsfølge fra tW, må t være sterkt normaliserbar.

CR 2 La $t \in T^{\vec{R}}_{\vec{X}}$ og anta at $t \to t'$. Vi skal vise at $t' \in T^{\vec{R}}_{\vec{X}}$.
Det betyr at vi skal velge en vilkårlig type W og en vilkårlig reduserbarhetskandidat S og vise at $t'W \in V^{S,\vec{R}}_{Y,\vec{X}}$.
Men $tW \to t'W$ og $tW \in V^{S,\vec{R}}_{Y,\vec{X}}$, så dette følger fra CR 2 for V, som skal holde i følge induksjonshypotesen.

CR 3 La t være nøytral av type T og anta at alle enkeltomskrivninger t' av t er i $T^{\vec{R}}_{\vec{X}}$. La W og S være som før. Vi må vise at $tW \in V^{S,\vec{R}}_{Y,\vec{X}}$, og vi vil bruke CR 3 for instanser av V.
Siden t er nøytral, vil enhver ettskrittsomskrivning av tW være på formen $t'W$ hvor t' er en ettskrittsomskrivning av t. Pr. antagelse er $t' \in T^{\vec{R}}_{\vec{X}}$, så $t'W \in V^{S\vec{R}}_{Y,\vec{X}}$. CR 3 for instanser av V gir oss da at $tW \in V^{S,\vec{R}}_{Y,\vec{X}}$, og vi er fremme.

$T^{\vec{R}}_{\vec{X}}$ kan oppfattes som en form for substitusjon, vi setter konkrete typer $\vec{U}$ med reduserbarhetskandidater $\vec{R}$ inn for $\vec{X}$ i T, og vi får en type med en reduserbarhetskandidat ut. Det neste lemmaet forteller oss at rekkefølgen vi utfører substitusjon på ikke spiller noen rolle:

Lemma 4.19 *La T være en type med fri typevariable blant $Y, \vec{X}$ hvor Y ikke forekommer i $\vec{X}$.*
La V være en type med fri variable blant $\vec{X}$ og la $\vec{U}$ med reduserbarhetskandidater $\vec{R}$ være gitt. Da er

$$(T^V_Y)^{\vec{R}}_{\vec{X}} = T^{V^{\vec{R}}_{\vec{X}},\vec{R}}_{Y,\vec{X}}.$$

Bevis
Vi bruker induksjon på oppbyggingen av T. Induksjonsbeviset er rett frem, og overlates til leseren som en oppgave.

Vårt mål er å vise at alle termene i en type er i minst en reduserbarhetskandidat. Det innebærer at vi må vise en generalisering av Teorem 2.18. Vi trenger et reformulert Lemma 2.17 og to nye lemmaer for å kunne gjennomføre beviset.

Lemma 4.20 *La $U = V \to W$ være typer, la S være en reduserbarhetskandidat for V og R en reduserbarhetsskandidat for W.*
La t være en term av type W og anta at $t_x^s \in R$ hver gang $s \in S$, hvor x er en variabel av type V.
Da er $\lambda x.t \in S \to R$.

Bevis
Ved å skrive V for τ, W for δ, $V \to W$ for σ, R for R_δ og S for R_τ kan vi bruke beviset for Lemma 2.17 direkte.

Lemma 4.21 *La W være en type med fri variable blant $Y, \vec{X}$ og la Y være en typevariabel. La $T = \prod Y.W$.*
La $\vec{U}$ med reduserbarhetskandidater $\vec{R}$ være som over.
La w være en term av type $W_{\vec{X}}^{\vec{U}}$ slik at for alle typer V og alle reduserbarhetskandidater S for V vil $w_Y^V \in W_{Y,\vec{X}}^{S,\vec{R}}$.
Da er $\Lambda Y.w \in (\prod Y.W)_{\vec{X}}^{\vec{R}}$.

Bevis
Vi viser først at w vil være sterkt normaliserbar:
Fra antagelsen følger det at $w = w_Y^Y \in W_{Y,\vec{X}}^{N_Y,\vec{R}}$, og ved CR 1 for sistnevnte er w sterkt normaliserbar.
Pr. definisjon må vi vise at $(\Lambda Y.w)V \in W_{Y,\vec{X}}^{S,\vec{R}}$ for alle typer V og alle reduserbarhetskandidater S for V. Vi skal bruke induksjon på maksimal lengde på en omskrivningfølge fra w.
Vi skal bruke $CR3$, og se på resultatet av én omskrivning:

- $(\Lambda Y.w)V$ skrives om til $(\Lambda Y.w')V$ ved at w skrives om til w'.
  Analogt til i beviset for Lemma 2.17 må vi sjekke at w' oppfyller betingelsen i teoremet, noe som vil følge fra CR 2 for hver $W_{Y,\vec{X}}^{S,\vec{R}}$.
  Da bruker vi induksjonsantagelsen for w' og får at $(\Lambda Y.w')V \in W_{Y,\vec{X}}^{S,\vec{R}}$.

- Vi bruker β-regelen, så $(\Lambda Y.w)V$ reduseres til w_Y^V.
  Denne termen er i $W_{Y,\vec{X}}^{S,\vec{R}}$ pr. antagelse.

Ved CR 3 for $W^{S,\vec{R}}_{Y,\vec{X}}$ får vi da at $(\Lambda Y.w)V \in W^{S,\vec{R}}_{Y,\vec{X}}$ for alle V og S, og vi er fremme.

Lemma 4.22 *La W, $\vec{U}$ og $\vec{X}$ være som over, og anta at $t \in (\prod Y.W)^{\vec{R}}_{\vec{X}}$ Da er $tV \in (W^V_Y)^{\vec{R}}_{\vec{X}}$ for alle typer V med fri typevariable blant $\vec{X}$.*

Bevis
Ved antagelsen er $tV \in W^{S,\vec{R}}_{Y,\vec{X}}$ for alle reduserbarhetskandidater S for V. Velg $S = V^{\vec{R}}_{\vec{X}}$ og bruk Lemma 4.19

Alt dette bygger opp mot et induksjonsbevis for

Lemma 4.23 *La T være en type og t en term av type T.*
La $\vec{x} = x_1, \ldots, x_m$ av typer $\vec{V} = V_1, \ldots, V_m$ omfatte alle fri variable i t og la $\vec{X} = X_1, \ldots, X_n$ omfatte alle fri typevariable i $T, \vec{V}$.
La $\vec{U} = U_1, \ldots, U_n$ være typer med reduserbarhetskandidater $\vec{R} = R_1, \ldots, R_n$ og la $\vec{v} = v_1, \ldots, v_m$ være termer slik at hver $v_j \in (V_j)^{\vec{R}}_{\vec{X}}$.
Da er

$$(t^{\vec{U}}_{\vec{X}})^{\vec{v}}_{\vec{x}} \in T^{\vec{R}}_{\vec{X}}.$$

Bevis
Lemmaet er notasjonsmessig krevende både formuleringsmessig og i utskrivningen. Vi bruker induksjon på oppbyggingen av t, og alle induksjonsskrittene er dekket av de foregående lemmaene. Induksjonstarten, $t = x_j$ er en direkte konsekvens av antagelsen om at $v_j \in (V_j)^{\vec{R}}_{\vec{X}}$.

Teorem 4.24 *Alle termer i System F er sterkt normaliserbare.*

Bevis
La t være en term av type T, la $\vec{x}$ av typer $\vec{V}$ omfatte de fri variableme i t og la $\vec{X}$ omfatte alle fri typevariable i T og i $\vec{V}$. La R_i være mengden av sterkt normaliserbare termer av type X_i. Ved CR 4 vil $x_j \in (V_j)^{\vec{R}}_{\vec{X}}$, og vi setter inn X_j for V_j og x_j for v_j i lemmaet over.
Det følger at $t \in T^{\vec{R}}_{\vec{X}}$ og ved CR 1 at t er sterkt normaliserbar.

Bemerkning 4.25 Vi lovte innledningsvis at vi skulle finne en reduserbarhetskandidat for en type T som omfattet alle termene av type T. $T^{\vec{R}}_{\vec{X}}$, slik den er definert i beviset over, er uavhengig av t, og vil derfor ha denne egenskapen. Det er selvfølgelig en konsekvens av normaliseringsteoremet at N_T omfatter alle termene av type T, og vi har at $N_T = T^{\vec{R}}_{\vec{X}}$.

4.5 Oppgaver

Oppgave 4.1 Forklar hvordan $\supset$ generaliserer både $\supset_\iota$ og $\supset_o$

Oppgave 4.2 La Z være som i avsnitt 4.3.
Vis at hvis t er en term av type NAT, så vil $Z(St) \to k_{\mathbf{F}}$.

Oppgave 4.3 Vis

- $\pi^1\langle u, v\rangle \to u$
- $\pi^2\langle u, v\rangle \to v$

ved å bruke omskrivningsreglene til System F.

Oppgave 4.4 Gi alternative, men likeverdige, definisjoner av paring og projeksjoner i form av lukkede termer i System F.

Oppgave 4.5 La U være en type, a en term av type U og t en term av type $NAT, U \to U$.
Vis at

1. $Rec_U atk_0 \to a$
2. $Rec_U atk_1 \to tk_0 a$
3. $Rec_U atk_2 \to tk_1(tk_0 a)$.

Bruk induksjon på n til å vise at $Rec_U atk_{n+1} \to tk_n(Rec_U atk_n)$.

Oppgave 4.6 Bevis Lemma 4.19

Kapittel 5

Domeneteoriens ene opprinnelse

5.1 Scott's dilemma

Utypet λ-kalkyle kan virke som et forholdsvis esoterisk forsøk på å lage en matematisk modell for beregnbarhet, og når vi i dag argumenterer for at kalkylen er en god modell, er det fordi vi kan vise at den er ekvivalent til andre modeller basert på en konseptuell analyse av beregningsfenomenet, for eksempel Turingmaskiner.

Når λ-kalkyle likevel var et populært tema på 50- og 60-tallet var det fordi mange programmeringsspråk lot seg oversette til λ-kalkyle, og enkelte programmeringsspråk hadde λ-kalkyle som sitt ideale, det var bare praktiske hensyn som gjorde at man måtte bruke rikere språk enn λ-kalkylen selv.

Dana Scott var kritisk til dette. Han mente at i utgangspunktet brukte man logiske systemer eller kalkyler for å støtte undersøkelsen av visse matematiske strukturer, og at studiet av en kalkyle uten noen form for mengdeteoretisk modell var lite verd. Sommeren 1969 var han gjesteforsker ved Universitetet i Oxford, og han arbeidet frem en typet kalkyle, LCF som han mente kunne tjene som en alternativt kilde til semantikk for programmeringsspråk. Scott argumenterte for at selv om den typefrie kalkylen er egnet til å modellere selvkallende prosedyrer, så vil de fleste program ta data av visse typer som inngangsdata og svarene vil også være data av visse typer. Som hjelp i å formulere program som fra data av type A gir oss data av type B kan vi ha behov for selvkallende prosedyrer, eksempelvis, definere $f : A \to B$ som et fikspunkt av en programmerbar funksjon $(A \to B) \to (A \to B)$. Hvis vi nå i stedet skulle definert en følge av funksjoner $\{f_n\}_{n\in\mathbb{N}}$ fra A til B kunne dette (i teorien) skjedd ved en rekursiv definisjon av

$$G : \mathbb{N} \to ((A \to B) \to (A \to B))$$

hvor vi deretter tar fikspunkt av hver $G(n)$. Skal vi simulere rekursjon ved hjelp av selvkallende prosedyrer, må vi opp en type til, så vi kan i værste fall risikere å måtte skrive et program for et objekt av type

$$(\mathbb{N} \to ((A \to B) \to (A \to B))) \to (\mathbb{N} \to ((A \to B) \to (A \to B))).$$

Scotts poeng var at selv om disse programmene blir komplekse så vil vi intuitivt arbeide med objekter av endelig type over basistypene.

Fortsatt gjenstår det to problemer:

1. Hvordan skal vi formulere kalkylen slik at vi faktisk fanger opp de programmene vi er interessert i å modellere?

2. Hvordan skal vi lage en mengdeteoretisk modell for en slik kalkyle?

Scotts svar på det første spørsmålet visste han at lå i doktoravhandlingen til Platek. Platek var ikke opptatt av semantikk for programmer, men av å utvikle en alternativ teori for beregnbarhet i funksjonaler av høyere orden. Kleene hadde allerede i 1959 kommet med en mulig definisjon av algoritmer hvor funksjonaler av endelig type (slik vi definerte dem i modellen vår for kjerneeksemplet) kunne tas som inngangsdata. Kleenes definisjon var basert på en form for Gödelnummerering av algoritmene, med en universell algoritme som en av de definerende ingrediensene. Plateks alternativ var over en rikere mengdeteoretisk tilkning av typene, han innførte de *hereditært konsistente funksjonalene* av endelig type, og han viste hvordan en form for typet λ-kalkyle, med konstanter for fikspunktoperatoren for hver type, kunne tolkes i dette typehierarkiet.
Nå er ikke Plateks avhandling lett tilgjengelig, hverken fysisk eller åndelig, og Scott bestemte seg for å utvikle sin egen versjon, LCF. Han tok utgangspunkt i kalkylen for kombinatorer, tilpasset en typet verden, og la til noen konstanter for funksjoner og objekter han trengte, typisk etterfølger, 0, tt og ff, og en fikspunktkonstant Y_σ av type $\sigma \to \sigma$ for hver type σ. En annen ny ingrediens er konstanten Ω_σ for det totalt ikketerminerende objektet av type

σ.
Han utviklet samtidig en formell logikk for en ordning $t \sqsubseteq s$ mellom termer av samme type. Vi skal ikke gå inn på denne logikken her, ettersom vi skal velge å se på et ekvivalent alternativ til LCF, nemlig Plotkins PCF, se avsnitt 6. Logikken i LCF hadde imidlertid innflytelse i informatikk, ved at Robin Milner og andre så at måten Scott definerte et logisk system på egnet seg godt for automatisk bevisførsel, og for eksempel det generelle programmeringsspråket ML er utviklet med LCF som modell.

Scotts svar på det andre spørsmålet var at han prøvde å begrense Plateks hereditært konsistente funksjonaler til en kontinuerlig situasjon. Litt inspirasjon for dette fant han i en artikke av Kreisel fra 1959, hvor Kreisel definerer de totale, kontinuerlige funksjonalene av endelig type. Scott hadde imidlertid behov for systematisk å innarbeide partielle funksjonaler av endelig type i sin matematiske modell, det vil si funksjonaler som ikke nødvendigvis terminerer på gitte input, og i tilfelle av høyere typer, hvor input selv kan være partielle funksjonaler.

Scott publiserte ikke sitt manuskript [3] fra sommeren 1969 før lenge etter, men det kan nå finnes som [4].

Når vi kaller dette kapitlet for domeneteoriens ene opprinnelse, er det fordi den russiske matematikeren Yu. Ershov uavhengig av Scott utviklet et matematisk maskineri ekvivalent med domeneteorien. Ershovs utgangspunkt var egenskapen til visse topologiske rom som han kom over i studiet av opptellinger av r.e. mengder. Vi skal ikke komme inn på Ershovs tilnærming i dette kurset.

5.2 Partielle, kontinuerlige funksjonaler

Når vi lager algoritmer for å regne med naturlige tall, kan vi komme i skade for å lage algoritmer som ikke terminerer, det vil si at de underliggende (teoretiske) maskinene vil arbeide og arbeide og $\cdots$ og $\cdots$ og $\cdots$, men aldri komme med noe svar. I utgangspunktet kan det virke som om bare dumme programmerere vil komme i en slik situasjon, eksempelvis slike som skriver

$$\mathit{while}\ \ x > 0\ \ \mathit{do}\ \ x := x + 1\ \ \mathit{od}$$

men en analyse av problemet viser at det ikke er mulig å kombinere ønsket om at vi bare skal kunne skrive programmer som alltid terminerer, at alle

beregnbare funksjoner skal kunne programeres og at det skal være avgjørbart om et program er syntaktisk korrekt. I denne umuligheten ligger det underforstått at språket vårt må kunne implementeres på en datamaskin, slik at hvis maskinen får programmet og inngangsdataene, så produserer den de ønskede utgangsdataene.
Dette betyr at dersom vi vil lage matematiske modeller for programmer som kan gi naturlige tall eller booleske verdier som svar, så må vi også gi en verdi til det udefinerte eller ikke-terminerende, og vi vil bruke $\perp$ for å betegne det udefinerte av basistype.
Vi vil derfor se på mengdene $\mathbb{N}_\perp = \{\perp\} \cup \mathbb{N}$ og $\mathbb{B}_\perp = \{\perp\} \cup \mathbb{B}$.

Hvis vi nå oppfatter elmentene i $\mathbb{N}_\perp$ og $\mathbb{B}_\perp$ som inneholdende informasjon om en inngangsverdi eller en utgangsverdi, ser vi at $\perp$ vil inneholde mindre informasjon enn alle de andre objektene, mens de innbyrdes ikke er sammenliknbare. Opplysninger om at "svaret er 17" og "svaret er 18" bør ikke være opplysninger om det samme regnestykket. Samtidig oppfatter vi det som at 17 og 18 er forskjellige inngangsdata i et program. Hvis vi derimot prøver å la $\perp$ være inngangsdata for et program ved å sette verdien til en variabel x til $\perp$ når vi starter programmet, så er det mulig at vi får et resultat av programmet hvis verdien av x ikke brukes aktivt under eksekveringen av programmet. På den annen side, hvis dette ikke er tilfelle, kan det hjelpe å gi mer informasjon om hvordan x skal tolkes, for eksempel ved å bestemme at $x = 0$ i utgangspunktet.
Mer generelt kan vi si at informasjon tolkes som en partiell ordning $(D, \sqsubseteq)$ hvor $\sqsubseteq$ skal svare til "mer informasjon".

Definisjon 5.1 La $D(\iota) = (\mathbb{N}_\perp, \sqsubseteq_\iota)$ og la $D(o) = (\mathbb{B}_\perp, \sqsubseteq_o)$ hvor $\perp \sqsubseteq_o a$ for alle $a \in \mathbb{B}_\perp$, $\perp \sqsubseteq_\iota a$ for alle $a \in \mathbb{N}_\perp$, $\mathbf{T} \sqsubseteq_o \mathbf{T}$, $\mathbf{F} \sqsubseteq_o \mathbf{F}$ og $n \sqsubseteq_\iota n$ for alle $n \in \mathbb{N}$.

Vårt neste problem vil være å gi tolkninger av typer på nivå 1 som slike partielle ordninger av informasjonsmengder.

Eksempel 5.2 La $f : \mathbb{N}_\perp \to \mathbb{N}_\perp$ være definert ved

1. $f(\perp) = 17$
2. $f(0) = f(13) = f(2345) = \perp$
3. $f(n) = 0$ for alle andre n.

Vi ser at hvis vi oppfatter $\perp$ som at vi ikke har fått informasjon om inngangsdata, men alle tallene som at vi har den tilsvarende informasjonen, blir det umulig å oppfatte f som resultatet av en reell eller abstrakt prosedyre; så lenge ingen forteller oss hva input skal være er svaret 17, mens etter at noen har fortalt oss det er svaret 0 eller udefinert.
Dette eksemplet viser at et rimelig krav er at vi begrenser interessen til *monotone* funksjoner.

Definisjon 5.3 La $(X, \sqsubseteq_X)$ og $(Y, \sqsubseteq_Y)$ være to partielle ordninger.
$f; X \to Y$ er *monoton* hvis vi for alle x_1 og x_2 i X har

$$x_1 \sqsubseteq x_2 \Rightarrow f(x_1) \sqsubseteq_Y f(x_2).$$

Mengden av monotone funksjoner fra $\mathbb{N}_\perp$ til $\mathbb{N}_\perp$ har en naturlig ordning, ved

$$f \sqsubseteq g \Leftrightarrow \forall a \in \mathbb{N}_\perp(f(a) \sqsubseteq_\iota g(a).$$

Denne definisjonen kan selvfølgelig utvides til mengden av monotone funksjoner fra en vilkårlig partiell ordning til en annen. Det var dette Platek gjorde da han definerte sitt typehierarki, se oppgave 5.1.

Definisjon 5.4 La σ være en type på nivå 1. Da er $\sigma = b \to \tau$ hvor b er en basistype og τ er av nivå 0 eller 1. Vi lar $D(\sigma)$ bestå av alle monotone funksjoner fra $D(b)$ til $D(\tau)$ ordnet ved punktvis ordning.

Det var gode grunner for at Scott ville begrense sine tolkninger av typene i forhold til Platek, men de grunnene ser vi først på nivå 2:

Eksempel 5.5 Definer

$$E : (\mathbb{N}_\perp \to \mathbb{N}_\perp) \to \mathbb{N}_\perp$$

ved

1. $E(f) = 0$ hvis $\exists n \in \mathbb{N}(f(n) = 0)$
2. $E(f) = 1$ hvis $\forall n \in \mathbb{N}(f(n) \in \mathbb{N} \wedge f(n) > 0$
3. $E(f) = \perp$ ellers.

Det er lett å se at E er monoton. Vi ser imidlertid at hvis $E(f) = 1$, så kan dette bare bekreftes ved å finne verdien av $f(n)$ for alle $n \in \mathbb{N}$, og hvis vi har en funksjon g som bare er definert for endelig mange argumenter, og der gir verdien 1, så er $E(g) = 0$ samtidig som det finnes det en f slik at $g \sqsubseteq f$ og $E(f) = 1$.
Uansett hvordan hva slags prosedyrer vi ønsker å modellere, så er det et rimelig krav at selv om input kan inneholde uendelig mye informasjon, så kan hver enkelt utregning i tråd med prosedyren ikke utnytte mer enn endelig mye av den informasjonen. Dette kan vi gjøre matematisk presist gjennom en definisjon av kontinuitet.

Vi skal vokte oss vel for å innføre topologi i disse forelesningene, men omkostningen vil være at vi må snakke om grensebegrepet i litt mer generell forstand enn vanlig.

Definisjon 5.6 La $(X, \sqsubseteq)$ være en partiell ordning.

a) $A \subseteq X$ kalles *rettet* hvis vi for alle a og b i A har en $c \in A$ slik at $a \sqsubseteq c$ og $b \sqsubseteq c$.

b) En partiell ordning kalles *komplett* hvis enhver rettet delmengde A har en minste øvre skranke $\sqcup A$.
En komplett partiell ordning omtales som en *cpo*.

Bemerkning 5.7 Noen forfattere velger å bruke betegnelsen *rettet komplett*, eller *dcpo* for det engelske *directed complete* på det vi kaller en *cpo*.

Den tomme mengden $\emptyset$ oppfyller kravene til å være rettet, så i en *cpo* skal $\sqcup\emptyset$ finnes. Dette må være det minste elementet i X, og vil bli omtalt som $\perp$ eller $\perp_X$ når $(X, \sqsubseteq_X)$ er en *cpo*.

La nå $(X, \sqsubseteq_X)$ og $(Y, \sqsubseteq_Y)$ være to *cpo*-er, og la $f : X \to Y$.
Hvis vi nå tenker intuitivt på elementene i X og Y som mengder av forenlig informasjon, hvor ordningen svarer til at vi utvider informasjonsmengden, hva skal da være rimelige krav til f ut fra forestillingen om at f prosseserer informasjon i Y fra informasjon i X?
Det første naturlige kravet vil være monotonitet, hvis $x_1 \sqsubseteq_X x_2$ bør $f(x_1) \sqsubseteq_Y f(x_2)$.
Hvis f er monoton, ser vi at bildet av en rettet mengde i X er en rettet

mengde i Y. Hvis en rettet mengde $A \subset X$ har et største element a, vil $a = \sqcup_X A$, og vi har at

$$f(\sqcup_X A) = \sqcup_Y f[A]$$

(hvor $f[A]$ betegner bildet av A under f).
Hvis, imidlertid, A ikke har et største element, tenker vi som følger:

- For hver enkelt-informasjon c i $f(\sqcup A)$ vil prosessen som definerer f bare bruke endelig mange informasjonsbiter fra $\sqcup A$. Hver av disse bitene b må ligge i en $a_b \in A$, og i en rettet mengde er alle endelige mengder begrenset. Derfor vil det finnes en $a \in A$ slik at $a_b \sqsubseteq_X a$ for alle relevante b. Det betyr at c bør finnes allerede i $f(a)$.

Disse vurderingene har ledet til følgende begrep:

Definisjon 5.8 La $(X, \sqsubseteq_X)$ og $(Y, \sqsubseteq_Y)$ være *cpo*-er.

a) Vi sier at $f : X \to Y$ er *kontinuerlig* dersom

   1. f er monoton
   2. Hvis $A \subset X$ er rettet vil

$$f(\sqcup_X A) = \sqcup_Y f[A].$$

b) Vi lar $X \to Y$ betegne mengden av kontinuerlige funksjoner fra X til Y.
Vi lar $f \sqsubseteq_{X \to Y} g$ hvis $f(x) \sqsubseteq_Y g(x)$ for alle $x \in X$.

Lemma 5.9 *$(X \to Y, \sqsubseteq_{X \to Y})$ som konstruert over er en cpo.*

Bevis
La $C \subset X \to Y$ være rettet.
For hver $x \in X$, la $B_x = \{f(x) \mid f \in C\}$. Da er B_x rettet for hver $x \in X$, og vi kan definere

$$g(x) = \sqcup_Y B_x.$$

Det er opplagt tilstrekkelig å vise at $g \in X \to Y$.

g er monoton:
Hvis $x_1 \sqsubseteq_X x_2$ er $f(x_1) \sqsubseteq f(x_2)$ for alle $f \in C$. Det følger at enhver øvre grense for B_{x_2} også vil være en øvre grense for B_{x_1}, og pr. definisjon av minste øvre grense, vil $g(x_1) \sqsubseteq_Y g(x_2)$.

g respekterer grenser av rettede mengder:
La $A \subset X$ være rettet. La

$$B = \{f(x) \mid f \in C \wedge a \in A\}.$$

Hvis $f_1(x_1) \in B$ og $f_2(x_2) \in B$, kan vi velge $f \in C$ som ligger over f_1 og f_2 og $x \in A$ som ligger over x_1 og x_2, og da vil $f(x)$ ligge over $f_1(x_1)$ og $f_2(x_2)$.
Det viser at B er rettet og har en minste øvre grense b.
Da har vi $g(\sqcup_X A) = \sqcup_Y \{f(\sqcup_X A) \mid f \in C\} =$
$\sqcup_Y \{\sqcup_Y f[A] \mid f \in C\} =$
$\sqcup_Y \{\sqcup_Y \{f(x) \mid x \in A\} \mid f \in C\} = b =$
$\sqcup_Y \{\sqcup_Y \{f(x) \mid f \in C\} \mid x \in A\} =$
$\sqcup_Y \{g(x) \mid x \in A\} = \sqcup_Y g[A]$.
Dette viser at g er kontinuerlig, og lemmaet er vist.

Lemma 5.10 *La $(X, \sqsubseteq)$ være en cpo. La $f : X \to X$ være kontinuerlig. Da finnes det en minste $x \in X$ slik at $f(x) = x$.*

Bevis
La $x_0 = \bot$ være det minste elementet i X.
La $x_{n+1} = f(x_n)$ definere følgen $\{x_n\}_{n \in \mathbb{N}}$ rekursivt.
Siden $x_0 \sqsubseteq x_1$ følger det ved induksjon og monotoniteten av f at $x_n \sqsubseteq x_{n+1}$ for alle n.
Da er $\{x_n \mid n \in \mathbb{N}\}$ en rettet mengde med en minste øvre grense x.
Vi har at $f(x) = \sqcup\{f(x_n) \mid n \in \mathbb{N}\} = \sqcup\{x_{n+1} \mid n \in \mathbb{N}\} = x$, så x er et fikspunkt for f.
Hvis y er et annet fikspunkt for f, ser vi at $x_0 \sqsubseteq y$, og vi viser ved induksjon at $x_{n+1} = f(x_n) \sqsubseteq f(y) = y$, så x må være det minste fikspunktet.

Det er i prinsippet lett å vise at sammensetningen av kontinuerlige funksjoner er kontinuerlig. Beviset overlates leseren som en passe trening i å håndtere begreper som monotonisitet og rettede mengders minste øvre skranker, se Oppgave 5.2.

Lemma 5.11 *La X, Y og Z være cpo-er, $f : X \to Y$ og $g : Y \to Z$ være kontinuerlige.*
Da er sammensetningen $h = g \circ f; X \to Z$ definert ved

$$h(x) = g(f(x))$$

også kontinuerlig.

Definisjon 5.12 Vi definerer *Scotts hierarki av partielle kontinuerlige funksjonaler* ved

1. $D(\iota) = \mathbb{N}_\perp$
2. $D(o) = \mathbb{B}_\perp$
3. $D(\sigma \to \tau) = D(\sigma) \to D(\tau)$

hvor vi underforstått har tatt med de partielle ordningene, og bruker *cpo*-definisjonen av funksjonsrom.

Bemerkning 5.13 Vi har bygget inn mye mindre innsikt omkring disse ordningene i disse definisjonene enn det som lå i Scotts første arbeide [3] på området. Vi velger å ta med denne innsikten under senere avsnitt, selv om denne innsikten lå til grunn for at Scott senere konstruerte en mengdeteoretisk modell for utypet λ-kalkyle.

5.3 Domener

Domeneteori er det abstrakte studiet av informasjonsmengder, gjennomført som studiet av *cpo*-er med endel tilleggsegenskaper. I hele dette avsnittet vil D, E og eventuelt C være *cpo*-er med ordning $\sqsubseteq$, hvor vi bruker en indeks på $\sqsubseteq$ når det er nødvendig.
Det finnes ikke noen omforent definisjon av begrepet *domene*. Vi skal etterhvert arbeide med det vi vil kalle *Scott domener*, men domeneteorien studerer mange andre typer ordninger enn dette.

Definisjon 5.14 La $(D, \sqsubseteq)$ være gitt og la $x \sqsubseteq y$.

a) Vi sier at $x \ll y$, (x er *langt under* y, x *is way below* y på engelsk) hvis hver gang $A \subseteq D$ er rettet og $y \sqsubseteq \sqcup A$, så finnes det $a \in A$ slik at $x \sqsubseteq a$.

b) $(D, \sqsubseteq)$ er et *kontinuerlig domene* hvis

   1. $\{x \mid x \ll y\}$ er rettet for alle $y \in D$
   2. $y = \sqcup\{x \mid x \ll y\}$ for alle $y \in D$

Eksempler 5.15 Vi skal se på en rekke eksempler på kontinuerlige domener:

a) $\mathbb{N}_\perp$ og $\mathbb{B}_\perp$ er eksempler på kontinuerlige domener. Her faller $\ll$ sammen med $\sqsubseteq$.

b) La D være potensmengden til $\mathbb{N}$ ordnet ved inklusjon.
La $A \subset B \subset \mathbb{N}$. Da har vi at $A \ll B$ hvis A er en endelig delmengde av B. Dette følger av at B er unionen av mengden av endelige delmengder av B, og definisjonen av $\ll$ tilsier at om $A \ll B$, så må A være inneholdt i en endelig delmengde av B.

c) La D være mengden av lukkede intervaller $[a, b]$ på den reelle talllinja sammen med hele $\mathbb{R}$.
Vi lar $I \sqsubseteq J \leftrightarrow J \subseteq I$, altså ved *omvendt inklusjon.*.
Denne ordningen er en *cpo* ettersom hvis $A \subseteq D$ er rettet, så vil $\bigcap\{I \mid I \in A\} \neq \emptyset$ og snittet blir et element i D.
Vi har at $I \ll J$ hvis J er inneholdt i det indre av I. Verifikasjonen av dette overlates til leseren. Vi overlater også til leseren å verifisere at dette betyr at D er et kontinuerlig domene.

d) La D være mengden av lukkede disker med radius $\leq \infty$ i planet, det vil si mengder på formen

$$\{(x, y) \in \mathbb{R}^2 \mid (x - x_0)^2 + (y - y_0)^2 \leq r^2\}$$

hvor r varierer fra og med 0 til og med ∞ og (x_0, y_0) varierer over $\mathbb{R}^2$.
La D være ordnet ved omvendt inklusjon.
Igjen har vi at $A \ll B$ hvis B er inneholdt i det indre av A og at $(D, \sqsubseteq)$ er et kontinuerlig domene.

e) La D være mengden av monotone funksjoner fra $\mathbb{N}_\perp$ til $\mathbb{N}_\perp$ ordnet ved punktvis ordning.
Da vil $f \ll g$ hvis $f \sqsubseteq g$ og $f(n) \in \mathbb{N}$ for bare endelig mange $n \in \mathbb{N}$.
Vi ser at $D(\iota \to \iota)$ er et kontinuerlig domene.

f) La D være mengden av konsistente teorier over et førsteordens språk L hvor to teorier anses for like hvis de har de samme teoremene.
Vi lar $T \sqsubseteq S$ hvis vi for alle utsagn ϕ i L har at

$$T \vdash \phi \Rightarrow S \vdash \phi.$$

Da vil $T \ll S$ hvis T er en endelig aksiomatiserbar delteori av S, og vi får et kontinuerlig domene.

I studiet av kontinuerlige domener er relasjonen $\ll$ sammen med prosessen av å ta minste øvre grenser sentrale elementer. I eksemplene over ser vi at vi normalt ikke trenger alle x slik at $x \ll y$ for å finne en rettet mengde med y som minste øvre grense. La oss se på eksemplene 5.15 a) - e):
I eksempel a) vil vi trenge alle elementene for å kunne generere hele området via rettede øvre grenser.
I eksempel b) vil vi klare oss med de endelige delmengdene.
I eksempel c) kan vi begrense oss til lukkede intervaller med rasjonale endepunkter og i eksempel d) til lukkede disker med rasjonal radius og rasjonale koordinater i sentrum. I begge disse tilfellene må vi ta med bunnelementet ($\mathbb{R}$ og $\mathbb{R}^2$). I begge disse tilfellene vil en vilkårlig y være den minste øvre grensen til den rettede mengden av elementer i den utpekte delmengden som ligger langt under y.
I eksempel e) vil enhver funksjon være den minste øvre grensen av de endelige subfunksjonene.

Alle disse observasjonene leder oss til et naturlig begrep *basis*:

Definisjon 5.16 La $(D, \sqsubseteq)$ være et kontinuerlig domene, $B \subseteq D$.
Vi kaller B en *basis* for $(D, \sqsubseteq)$ hvis

1. $\{x \in B \mid x \ll y\}$ er en rettet mengde for alle $y \in D$.

2. $y = \sqcup\{x \in B \mid x \ll y\}$ for alle $y \in D$.

Ethvert kontinuerlig domene vil ha en basis, nemlig D selv. Det interessante er når vi kan plukke ut små basiser med en viss effektiv representasjon. Da kan vi bruke basisen til å løfte fenomener fra rekursjonsteorien til domeneteorien. Dette er selvfølgelig av interesse når vi skal bruke domener til å gi tolkninger av programmer i programmeringsspråk. I alle eksemplene over har vi tellbare basiser for de aktuelle domenene, og relasjoner som $\sqsubseteq$ og $\ll$ vil med ett untak være avgjørbare for basiselementer. Untaket er eksemplet med teorier over et første ordens språk.

I noen av eksemplene viste vi at $x \ll y$ hvis $x \sqsubseteq y$ og tilhører en på forhånd gitt mengde uavhengig av y. Dette gjelder eksemplene a), b), e) og f) i 5.15. I de andre eksemplene vil y i større grad bestemme hvilke x som vil være langt under y.
I eksempel b) vil vi for eksempel ha at $\{0, 4, 5\} \ll \{0, 4, 5\}$ i følge den formelle definisjonen, og dette fenomenet er felles for alle basiselementene i eksemplene

a), b), e) og f). Dette er selvfølgelig i konflikt med hva vi umiddelbart tenker på når vi sier “langt under”, men det får vi bare finne oss i inntil videre. Denne diskusjonen leder oss til følgende definisjoner:

Definisjon 5.17 La $(D, \sqsubseteq)$ være et domene.

a) La $c \in D$. Vi sier at c er *kompakt* hvis $c \ll c$.

b) D kalles *algebraisk* hvis kompaktene i D utgjør en basis.
Vi skriver gjerne D_0 for mengden av kompakter.

c) Et algebraisk domene D er *separabelt* hvis mengden av kompakter er høyst tellbar.

Hvis vi ser på eksemplene i 5.15 ser vi at i a) er alle objektene kompakte, i b) utgjør de endelige delmengdene av $\mathbb{N}$ kompaktene, i c) og d) er det bare bunnelementet $\perp$ som er kompakt (Advarsel! Som delmengder av de overliggende metriske rommene er alle elementer unntatt bunnelementet kompakte. Det er derfor viktig i enkelte sammenhenger å holde sammenhengen klart for seg.). I eksempel d) er det de endelige partielle funksjonene som er kompakter, mens i eksempel f) er det de endelig aksiomatiserbare teoriene som utgjør kompaktene.
a), b), e) og f) er eksempler på algebraiske domener, mens c) og d) er eksempler på kontinuerlige domener som ikke er algebraiske.

Det er en smaksak om man ønsker å begrense seg til algebraiske domener, eller om man ønsker å arbeide med de mer generelle kontinuerlige domenene. Vi skal være litt mer presis i hva vi mener med “smaksak”.

Definisjon 5.18 La $(D, \sqsubseteq_D)$ og $(E, \sqsubseteq_D)$ være to *cpo*-er.
Et *kontinuerlig ep-par* fra D til E er et par (η, π) av kontinuerlige funksjoner $\eta : D \to E$ og $\pi : E \to D$ slik at

1. $\pi \circ \eta = id_D$

2. $\eta \circ \pi \sqsubseteq_{E \to E} id_E$

Bemerkning 5.19 Vi bruker η, “eta” for å markere at det er en “embedding”, mens π markerer at det er en “projeksjon”.

Vi kan vise at den ene delen av et kontinuerlig ep-par (η, π) bestemmer den andre via følgende karakteristikker:

Lemma 5.20 *La* $(X, \sqsubseteq_X)$ *og* $(Y, \sqsubseteq_Y)$ *være to cpo-er og la* (η, π) *være et ep-par fra* X *til* Y. *Da gjelder*

- *For alle* $y \in Y$ *er* $\pi(y)$ *den* $\sqsubseteq_X$*-største* $x \in X$ *slik at* $\eta(x) \sqsubseteq_Y y$.
- *For alle* $x \in X$ *er* $\eta(x)$ *den* $\sqsubseteq_Y$ *minste* y *slik at* $\pi(y) = x$.

Beviset er gitt som Oppgave 5.3

Teorem 5.21 *La* $(X, \sqsubseteq_X)$ *være et kontinuerlig domene.*
Da finnes det et algebraisk domene $(Y, \sqsubseteq_Y)$ *og et kontinuerlig ep-par* (η, π) *fra* X *til* Y.

Bevis
La $B \subseteq X$ være en basis for X.
$\alpha \subseteq B$ kalles et *ideal* hvis

- $\perp_X \in \alpha$. [Merk at $\perp_X \in B$.]
- Hvis $b \in \alpha$ og $c \in B$ slik at $c \sqsubseteq_X b$ så er $c \in \alpha$.
- Hvis $a \in \alpha$ og $b \in \alpha$ finnes det en $c \in \alpha$ slik at $a \sqsubseteq_X c$ og $b \sqsubseteq_X c$.

[Merk at definisjonen av et ideal gir mening for alle partielle ordninger med et minste element.]
Vi lar Y være mengden av idealer i B og vi lar $\sqsubseteq_Y$ være delmengdeordningen på klassen av idealer.
For alle $x \in X$ lar vi $\alpha_x = \{b \in B \mid b \ll x\}$, og vi lar $\eta(x) = \alpha_x$.
For $b \in B$ lar vi $\beta_b = \{c \in B \mid c \sqsubseteq_X b\}$. Merk at $b \in \beta_b$, mens vi ikke automatisk har at $b \in \alpha_b$. Det er en rekke forhold som må verifiseres her.

1. α_x er et ideal.
   Dette følger av definisjonen av kontinuerlige domener.

2. Y er en *cpo* og β_b er kompakt i Y når $b \in B$. La C være en rettet mengde av elementer i Y. Da er $\bigcup C$ et ideal, så $\bigcup C \in Y$. $\bigcup C$ er opplagt den minste øvre grensen for C. Dette viser at Y er en *cpo*.
   Anta nå videre at $\beta_b \sqsubseteq_Y \bigcup C$. Da finnes det en $\alpha \in C$ slik at $b \in \alpha$, hvilket videre betyr at $\beta_b \sqsubseteq_Y \alpha$.
   Dette viser at β_b er kompakt.

3. Y er et algebraisk domene.
   For alle $\alpha \in Y$ har vi
$$\alpha = \bigcup \{\beta_b \mid b \in \alpha\}$$
   så mengden av β_b når b varierer over B utgjør en basis av kompakter.

4. η er en del av et ep-par.
   La $\alpha \in Y$ og la $\pi(\alpha) = \bigsqcup_X \alpha$.
   For alle $x \in X$ har vi
$$x = \bigsqcup_X \{b \in B \mid b \ll x\} = \pi(\eta(x)).$$
   Hvis $\alpha \in Y$ vil
$$\eta(\pi(\alpha)) = \{b \in B \mid b \ll \bigsqcup_X \alpha\}.$$
   Men hvis α er et ideal i B, er spesielt α en retted delmengde av X, så hvis $b \ll \bigsqcup_X \alpha$ betyr det, fra definisjonen av $\ll$, at $b \in \alpha$.
   Det gir at $\eta(\pi(\alpha)) \subseteq \alpha$.
   Det gjenstår å vise at η og π er kontinuerlige. det overlates leseren som oppgave.

Dette avslutter beviset for Teorem 5.21

Nå skulle vi gjerne vist at hvis X og Y er kontinuerlige domener, så er $X \to Y$ også kontinuerlig, eller at hvis X og Y er algebraiske domener, så er $X \to Y$ også algebraisk. Grunnen til at vi ikke gjør det er at da vil det trolig være en feil i beviset ettersom det finnes moteksempler.
Vi skal ikke gå inn på hvordan vi skal konstruere slike moteksempler. Vi skal i stedet beskrive en ekstra tilleggsbetingelse som både er naturlig og som vil sikre at disse to kategoriene er lukket under funksjonsromdannelsen.

Bemerkning 5.22 For den som kan litt kategoriteori gir følgende bemerkning mening:

> Kategoriene av kontinuerlige eller algebraiske domener med de kontinuerlige funksjonene som morfismer er ikke kartesisk lukkede kategorier. Vi tar sikte på å finne relevante underkategorier som er kartesisk lukket.

I 5.15 så vi på en rekke algebraiske og kontinuerlige domener. Med ett unntak har disse eksemplene den egenskapen at hvis x og y har en felles øvre grense, så har paret en minste øvre grense. Unntaket er eksempel d), hvor to lukkede disker har en øvre grense hvis snittet er ikketomt, mens hvis snittet har et indre, så finnes det ingen minste øvre grense her. Leseren anbefales å tegne en figur for å reflektere over denne påstanden.

Definisjon 5.23 La $(X, \sqsubseteq)$ være en partiell ordning.

a) Vi sier at X er *begrenset komplett* hvis alle begrensede mengder $\{x, y\} \subseteq X$ har en minste øvre grense.

b) En *cpo* X er et *Scott-domene* hvis X er et begrenset komplett og separabelt algebraisk domene.

Vi kommer til å arbeide med Scott-domener i fortsettelsen, men vil formulere lemmaer og teoremer så generelt som mulig, der dette ikke krever ekstra innsats i bevisføringen.

Lemma 5.24 *La $(X, \sqsubseteq)$ være en begrenset komplett cpo.*

a) Hvis $A \subseteq X$ er endelig og begrenset, så vil A ha en minste øvre grense $\bigsqcup A$.

b) Hvis $B \subseteq X$ er slik at alle endelige delmengder $A \subseteq B$ er begrenset, så har B en minste øvre grense $\bigsqcup B$.

Bevis
a) vises ved induksjon på antall elementer i A sammen med observasjonen at

$$\bigsqcup\{a_1, \ldots a_n\} = \bigsqcup\{a_1 \ldots, a_{n-1}\} \sqcup a_n.$$

For å vise b) bruker vi notasjonen $A \subseteq_f B$ som betyr at A er en endelig delmengde av B.
Vi har da at $\{\bigsqcup A \mid A \subseteq_f B\}$ er en rettet mengde, og at den minste øvre grensen av denne, som finnes ettersom X er en *cpo*, også vil være den minste øvre grensen for B.

Lemma 5.25 *La $(X, \sqsubseteq)$ være en begrenset komplett cpo og la x og y være kompakter. Da er $x \sqcup y$ også en kompakt.*

Beviset er enkelt og overlates leseren som Oppgave 5.5

Når det gjelder det neste resultatet er det ikke bare teoremet som er viktig, men også de konstruksjonene vi gjennomfører i beviset for teoremet. Det finnes et tilsvarende resultat for begrenset komplette algebraiske domener og kontinuerlige domener generelt, kravet til at domenet skal være separabelt er uvesentlig.

Teorem 5.26 *La* $(X, \sqsubseteq X)$ *og* $(Y, \sqsubseteq Y)$ *være Scott-domener.*
Da er $(X \to Y, \sqsubseteq_{X \to Y})$ *også et Scott-domene.*

Bevis
Vi skal bevise dette teoremet ved å vise en rekke påstander. Vi kommer systematisk til å la x variere over X og y over Y.

Påstand 1
La $p \in X$ og $q \in Y$ være kompakter.
La $f_{\langle p,q\rangle}(x) = q$ hvis $p \sqsubseteq_X x$ og $\bot_Y$ ellers.
Da er $f_{\langle p,q\rangle}$ en kompakt i $X \to Y$.

Bevis for påstand 1
$f_{\langle p,q\rangle}$ er opplagt monoton.
Hvis $A \subseteq X$ er en rettet mengde, vil $p \sqsubseteq_X \bigsqcup_X A$ hvis og bare hvis $p \sqsubseteq_X a$ for en $a \in A$.
Dette betyr at

$$f(\bigsqcup_X A) = q \Leftrightarrow \exists a \in A(f(a) = q)$$

hvor alternativet er at $f_{\langle p,q\rangle}(\bigsqcup_X A) = \bot_Y$.

Påstand 2
$f_{\langle p,q\rangle}$ er kompakt i $X \to Y$.

Bevis for påstand 2.
La $C \subseteq X \to Y$ være rettet slik at $f_{\langle p,q\rangle} \sqsubseteq_{X\to Y} \bigsqcup_{X\to Y} C$. La $f = \bigsqcup_{X\to Y} C$.
Da er $q \sqsubseteq_Y f(p)$, og siden $\{g(p) \mid g \in C\}$ er en rettet mengde med $q \sqsubseteq_Y \bigsqcup_Y \{g(p) \mid g \in C\}$, finnes det $g \in C$ slik at $q \sqsubseteq_Y g(p)$.
Da vil $f_{\langle p,q\rangle} \sqsubseteq_{X\to Y} g$.
Dette viser at $f_{\langle p,q\rangle}$ er kompakt.

Påstand 3
$X \to Y$ er begrenset komplett.

Bevis for påstand 3

La f og g være et begrenset par. Da er h definert ved

$$h(x) = f(x) \sqcup g(x)$$

den minste øvre grensen.
Merk at vi bare trenger at Y er begrenset komplett her.
Påstand 4
La $f \in X \to Y$. Da er

$$f = \bigsqcup_{X \to Y} \{f_{\langle p,q \rangle} \mid q \sqsubseteq_Y f(p)\}.$$

Bevis for påstand 4
Vi har opplagt at om $q \sqsubseteq_Y f(p)$ så vil $f_{\langle p,q \rangle} \sqsubseteq_{X \to Y} f$, og det viser ulikheten den ene veien.
Hvis $f_{\langle p,q \rangle} \sqsubseteq_{X \to Y} g$ hver gang $q \sqsubseteq_Y f(p)$ skal vi se at $f(x) \sqsubseteq g(x)$ for alle $x \in X$:
Vi bruker at f er kontinuerlig og får

$$f(x) = \bigsqcup_Y \{f(p) \mid p \sqsubseteq_X x\} = \bigsqcup_Y \{q \mid p \sqsubseteq_X x \wedge q \sqsubseteq_Y f(p)\}.$$

Påstand 5
Hvis $f \in X \to Y$ er kompakt, finnes det kompakter $p_1, \ldots, p_n$ i X og $q_1, \ldots, q_n$ i Y slik at

$$f = f_{\langle p_1, q_1 \rangle} \sqcup \ldots \sqcup f_{\langle p_n, q_n \rangle}.$$

Det følger at $X \to Y$ er separabelt.
Bevis for påstand 5
Vi har sett at $f = \bigsqcup\{f_{\langle p,q \rangle} \mid q \sqsubseteq f(p)$, og siden f er kompakt, må f være begrenset av den minste øvre grensen for endelig mange $f_{\langle p,q \rangle}$ slik at $q \sqsubseteq_Y f(p)$. Dette viser påstanden.
Hvis antall kompakter i X og i Y begge er tellbare, betyr det at antall kompakter i $X \to Y$ er tellbart.

Påstandene 1-5 gir samlet konklusjonen i teoremet, og beviset er dermed fullendt.

Korollar 5.27 *Hvis σ er en endelig type, er $D(\sigma)$ et Scott-domene.*

Bevis
Det følger direkte fra erkjennelsen av at $D(\iota)$ og $D(o)$ er Scott-domener ved

induksjon på σ.

Vi skal nå se at vi kan gi en presis beskrivelse av når en endelig mengde av funksjoner $f_{\langle p,q\rangle}$ er begrenset, og av hvordan den minste øvre grensen er definert.

Definisjon 5.28 La X og Y være Scott-domener.
Funksjoner på formen $f_{\langle p,q\rangle}$ kalles *basale kompakter* i $X \to Y$.

Lemma 5.29 *La X og Y være Scott-domener. $C = \{f_{\langle p_1,q_1\rangle}, \ldots, f_{\langle p_n,q_n\rangle}\}$ være en endelig mengde basale kompakter i $X \to Y$.*
Da er C begrenset hvis vi for alle endelige mengder $K \subseteq \{1, \ldots, n\}$ har

Hvis $\{p_i \mid i \in K\}$ er begrenset i X, så er $\{q_i \mid i \in K\}$ begrenset i Y.

I så fall er

$$f_C(x) = \bigsqcup_Y \{q_i \mid p_i \sqsubseteq_X x\}$$

den minste øvre grensen til X.

Bevis
La C være begrenset av f La $K \subseteq \{1, \ldots, n\}$ være slik at $\{p_i \mid i \in K\}$ er begrenset, og la x være en øvre grense. Da er $q_i \sqsubseteq_Y f(x)$ når $i \in K$, så C oppfyller kriteriet over.
På den annen side, hvis kriteriet er oppfyllt er f_C veldefinert, monoton og kontinuerlig, og det er lett å se at f_C er den minste øvre grensen for C.

Vi trenger å behandle kartesiske produkter av domener også:

Definisjon 5.30 La $(X, \sqsubseteq_X)$ og $(Y, \sqsubseteq_Y)$ være to partielle ordninger.
Vi definerer det *kartesiske produktet* $Z = X \times Y$ ved

$$Z = \{(x, y) \mid x \in X \wedge y \in Y\}$$

$(x, y) \sqsubseteq_Z (x', y')$ hvis $x \sqsubseteq x'$ og $y \sqsubseteq y'$

Følgende påstander er trivielle, og overlates til leseren som øvelser, se Oppgave 5.6:

Lemma 5.31 *La X og Y være partielle ordninger, $Z = X \times Y$.*

a) Hvis X og Y er cpo-er, er Z en cpo.

b) *Hvis X og Y er kontinuerlige domener, er Z et kontinuerlig domene. Vi kan finne en basis for Z ved å ta produktet av en basis for X og en basis for Y.*

c) *Hvis X og Y er algebraiske domener, er Z et algebraisk domene. Mengden av kompakter i Z er produktet av mengdene av kompakter i X og kompakter i Y.*

d) *Hvis X og Y er separable, er Z separabelt.*

Vi trenger kartesiske produkter for å slå fast at funksjonsanvendelse er kontinuerlig i begge koordinater:

Lemma 5.32 *La X, Y og Z være Scott-domener.*

a) *La $Ap : (X \to Y) \times X \to Y$ være definert ved*

$$Ap(f, x) = f(x).$$

Da er Ap kontinuerlig.

b) *La $Ab : (X \times Y \to Z) \to (X \to (Y \to Z))$ være definert ved*

$$Ab(f)(x)(y) = f(x, y).$$

(Ab står for "abstraksjon".)
Da er Ab kontinuerlig.

Bevis
Vi formulerte lemmaet for Scott-domener, men trenger bare at X, Y og Z er *cpo*-er.

a) Ap er opplagt monoton, det ligger i definisjonen av ordningen på $X \to Y$.
Hvis $C \subseteq (X \to Y) \times X$ er en rettet mengde, er

$$B = \{f \in X \to Y \mid \exists x \in X (f, x) \in C\}$$

og

$$A = \{x \in X \mid \exists f \in (X \to Y)(f, x) \in C\}$$

begge rettede mengder. Hvis $f = \bigsqcup_{X \to Y} B$ og $x = \bigsqcup_X A$ vil $(f, x) = \bigsqcup_{(X \to Y) \times X} C$.
Detaljene overlates leseren.

b) La $G(f, x, y) = f(x, y)$. Ved a) er G kontinuerlig, dvs. G er monoton og respekterer minste øvre grense av rettede mengder.

   - For å vise at $Ab(f)(x) \in Y \to Z$ når f og x er gitt, trenger vi at G er kontinuerlig i siste koordinat.
   - For å vise at $Ab(f) \in X \to (Y \to Z)$ trenger vi at G er kontinuerlig i midterste koordinat når f og y holdes fast.
   - For å vise at Ab er kontinuerlig trenger vi at G er kontinuerlig i første koordinat når x og y holdes fast.

   Detaljene overlates leseren.

Vi skal også se hvordan vi kan konstruere disjunkt union av to Scott-domener. En tilsvarende konstruksjon kan brukes for å konstruere disjunkte unioner av flere enn to domener.

Definisjon 5.33 La $(X, \sqsubseteq_X)$ og $(Y, \sqsubseteq_Y)$ være to Scott-domener.
Vi definerer den *disjunkte unionen* $Z = X \oplus Y$ ved:

- $Z = \{\bot\} \cup \{(1, x) \mid x \in X\} \cup \{(2, y) \mid y \in Y\}$
- $\bot \sqsubseteq_Z (1, x)$ og $\bot \sqsubseteq_Z (2, y)$ når $x \in X$ eller $y \in Y$.
- $(1, x) \sqsubseteq_Z (1, x') \Leftrightarrow x \sqsubseteq_X x'$
- $(2, y) \sqsubseteq_Y (2, y') \Leftrightarrow y \sqsubseteq_Y y'$
- $(1, x)$ og $(2, y)$ er ikke ordnet seg imellom når $x \in X$ og $y \in Y$.

Se Oppgave 5.7 for å finne de vesentlige egenskapene.

Scott gjennomførte konstruksjonen av hver enkelt $D(\sigma)$ med den innsikten vi nå har skaffet oss, og brukte dette til å gi en modell for LCF. Han oppdaget imidlertid at han i tillegg til å kunne tolke hver endelig type som *idealkompletteringen* av en partiell ordning, så kunne han fortsette konstruksjonen til en typefri struktur.
Vi følger en litt modernisert vei til de samme typer modeller:

Definisjon 5.34 Et *applikasjonsdomene* vil være et Scott-domene $(X, \sqsubseteq)$ sammen med en ep-par (η, π) fra $X \to X$ til X.

I det neste avsnittet skal vi se at vi kan bruke ethvert applikasjonsdomene til å gi en tolkning av utypet λ-kalkyle. Vi skal avslutte dette avsnittet med å konstruere to slike applikasjonsdomener, den late og den energiske konstruksjonen. La oss starte med den late.

Lemma 5.35 *La X være $\mathcal{P}(\mathbb{N})$ ordnet ved inklusjon.*
La $\lceil \ldots \rceil$ være en nummerering av de endelige delmengdene av $\mathbb{N}$.
Vi lar A og B variere over de endelige delmengdene av $\mathbb{N}$ og x over alle elementer i X. La $\subset$ betegne strikt inklusjon.
La $f : X \to X$ være kontinuerlig. La

$$\eta(f) = \{\langle \lceil A \rceil, n\rangle \mid n \in f(A) \wedge (B \subset A \to n \notin f(B))\}.$$

Hvis $x \in X$ lar vi

$$\pi(x) = \{n \mid \exists A \subseteq_f x(\langle \lceil A \rceil, n\rangle \in x)\}.$$

Da vil (η, π) utgjøre et ep-par fra $X \to X$ til X

Vi kaller dette det late applikasjonsdomenet fordi vi ikke har gjort noe forsøk på å fange opp noen struktur. Når det gjelder å tolke utypet λ-kalkyle, så er dette godt nok, for utypet λ-kalkyle er strukturløs i seg selv. Det gir for eksempel ikke mening å si at to termer i utypet λ-kalkyle definerer uforenlige objekter. Hvis vi derimot ønsker modeller hvor den utypede kalkylen er konstruert som en utvidelse av typede kalkyler, kan den opprinnelige konstruksjonen til Scott være mer relevant.
Vi trenger å kunne ta grenser av voksende følger av Scott-domener i denne konstruksjonen.

Teorem 5.36 *La $\{(X_n, \sqsubseteq_n)\}_{i\in\mathbb{N}}$ være en følge av Scott-domener, med en følge kontinuerlige ep-par $(\eta_{n,n+1}, \pi_{n,n+1})$ fra X_n til X_{n+1}.*
Da finnes det et Scott-domene $(X, \sqsubseteq)$ og for hver n kontinuerlige ep-par (η_n, π_n) fra X_n til X slik at

1. *$\eta_n = \eta_{n+1} \circ \eta_{n,n+1}$ og $\pi_n = \pi_{n,n+1} \circ \pi_{n+1}$ for alle n*

2. *Hvis $(X', \sqsubseteq')$ og $\{(\eta'_n, \pi'_n\}_{n\in\mathbb{N}}$ oppfyller kravet i 1., finnes det et kontinuerlig ep-par (η, π) fra X til X' slik at $\eta'_n = \eta \circ \eta_n$ og $\pi'_n = \pi_n \circ \pi$ for alle n.*

Vi oppfatter $(X, \sqsubseteq)$ som *grensen* av den *voksende følgen* $\{(X_n, \sqsubseteq_n)\}_{n\in\mathbb{N}}$.

Bevis
La p være kompakt i X_n, og la $A \subseteq X_{n+1}$ være rettet slik at $\eta_{n,n+1}(p) \sqsubseteq_{n+1} \bigsqcup_n A$. Da er $p = \pi_{n,n+1}(\eta_{n,n+1}(p)) \sqsubseteq_n \bigsqcup_n \pi_{n,n+1}[A]$, og siden p er kompakt, finnes det $a \in A$ slik at $p \sqsubseteq_n \pi_{n,n+1}(a)$.
Det betyr at $\eta_{n,n+1}(p) \sqsubseteq_{n+1} \eta_{n,n+1}(\pi_{n,n+1}(a)) \sqsubseteq_{n+1} a$.
Dette viser at $\eta_{n,n+1}(p)$ er kompakt i X_{n+1} når p er kompakt i X_n.
Vi har også at en mengde av kompakter i X_n er begrenset i X_n hvis og bare hvis $\eta_{n,n+1}$-bildet er begrenset i X_{n+1}.
Denne observasjonen gjør at vi kan forenkle bildet noe, vi kan anta at $X_n \subseteq X_{n+1}$ og at $\eta_{n,n+1}$ er inklusjonsavbildningen.

La unionen av kompaktene i hver X_n, med sin ordning arvet fra ordningene $\sqsubseteq_n$, utgjøre kompaktene i X, og la X være domenet av idealer av slike kompakter.
Da lar vi $\eta_n(x)$ være idealet vi får ved å lukke mengden av kompakter begrenset av $x \in X_n$ nedover. Hvis $x \in X$, lar vi $\pi_n(x)$ være den minste øvre grensen i X_n av mengden av kompakter i X_n som ligger i x.
Det er lett å se at egenskapene i 1. er oppfyllt, så la $(X', \sqsubseteq')$ med ep-par (η'_n, π'_n) for hver n også oppfylle 1.
Vi definerer

$$\eta(x) = \bigsqcup{}'\{\eta'_n(y) \mid n \in \mathbb{N} \wedge y \in X_n \wedge \pi_n(y) \sqsubseteq x\}.$$

Vi definerer

$$\pi(x) = \bigsqcup\{\eta_n(\pi'_n(x)) \mid n \in \mathbb{N}\}.$$

Vi overlater verifikasjonene til leseren.

Vi er nå klar til å fullføre den energiske konstruksjonen. Poenget er at vi ønsker oss et Scott-domene X som tilfredstiller følgende likning

$$X = \mathbb{N}_\perp \oplus \mathbb{B}_\perp \oplus (X \to X).$$

La $X_0 = \{\perp\}$. For hvert domene X har vi et kontinuerlig ep-par (η_0, π_0) fra X_0 til X ved $\eta(\perp) = \perp_X$ og $\pi(x) = \perp$ for alle $x \in X$.

La $\Gamma(X) = \mathbb{N}_\perp \oplus \mathbb{B}_\perp \oplus (X \to X)$ hvor X er et domene, og hvor vi bruker paring med 1, 2 og 3 for å markere de tre bitene.
Hvis (η, π) er et ep-par fra et domene X til et domene Y, definerer vi

$$\Gamma(\eta, \pi) = (\Gamma(\eta), \Gamma(\pi))$$

ved;

- $\Gamma(\eta)(\bot) = \Gamma(\pi)(\bot) = \bot$
- $\Gamma(\eta)(\langle i, x\rangle) = \Gamma(\pi)(\langle i, x\rangle) = \langle i, x\rangle$ når $i = 1$ eller $i = 2$.
- $\Gamma(\eta)(\langle 2, f\rangle) = \langle 2, \lambda y \in Y\eta(f(\pi(y)))\rangle$
- $\Gamma(\pi)(\langle 2, f\rangle) = \langle 2, \lambda x \in X\pi(f(\eta(y)))\rangle$

Vi lar nå X_0 være det trivielle Scott-domenet, $X_{n+1} = \Gamma(X_n)$ for alle n.
Vi lar (η_0, π_0) være det trivielle ep-paret fra X_0 til X_1 og vi lar $(\eta_{n+1}, \pi_{n+1}) = \Gamma(\eta_n, \pi_n)$ være det avledede ep-paret fra X_{n+1} til X_{n+2} for alle n.
Dette definerer en voksende følge av domener slik at vi kan ta grensen X_∞.
Slik vi konstruerte X_∞ vil alle kompakter i $\Gamma(X_\infty)$ allerede være kompakter i X_∞, så (opp til isomorfi, om man insisterer) vil $X_\infty = \Gamma(X_\infty)$.
Dette betyr at X_∞ er et applikasjonsdomene.

5.4 En modell for utypet λ-kalkyle.

I dette avsnittet lar vi X være et applikasjonsdomene og vi lar (η, π) være et ep-par fra $X \to X$ til X.

Vi definerer $App : X \times X \to X$ som

$$App(y, x) = \pi(y)(x).$$

App er opplagt kontinuerlig.

La $x_1, \dots, x_n$ være variable i utypet λ-kalkyle. I dette avsnittet vil en *tilordning* være et element i X^n. som vi oppfatter som en tilordning av verdier til variablene $x_1, \dots, x_n$. Hvis $a \in X$ og ν er en tilordning, lar vi ν_i^a som før stå for tilordningen vi får fra ν ved å endre verdien av koordinat i til a. Notasjonsmessig lar vi $\nu : \{1, \dots, n\} \to X$.

Definisjon 5.37 La t være en term med alle variable blant $x_1, \dots, x_n$.
Vi tolker t som en kontinuerlig funksjon

$$[\![t]\!]; X^n \to X$$

ved induksjon på oppbyggingen av t som følger:

1. Hvis t er x_i er $[\![t]\!](\nu) = \nu(i)$.
2. Hvis $t = sr$ lar vi $[\![t]\!](\nu) = App([\![s]\!](\nu), [\![r]\!](\nu))$.
3. Hvis $t = \lambda x_i.s$ lar vi $[\![t]\!](\nu) = \eta(f)$ hvor $f : X \to X$ er definert ved
$$f(a) = [\![s]\!](\nu_i^a).$$

Vi må vise at dette er en velfungerende definisjon:

Lemma 5.38 $[\![t]\!]$ *er veldefinert og kontinuerlig for alle* t.

Bevis
Vi bruker induksjon på oppbygningen av t

1. Hvis t er en variabel, er dette trivielt.
2. Hvis $t = sr$ og $[\![s]\!]$ og $[\![r]\!]$ begge er kontinuerlige, får vi $[\![t]\!]$ ved sammensetninger av kontinuerlige funksjoner, og slike sammensetninger er kontinuerlig, se Lemma 5.32a).
3. Hvis $t = \lambda x_i.s$ og $[\![s]\!]$ er kontinuerlig, vil funksjonen $g : X^n \to (X \to X)$ definert ved
$$g(\nu)(a) = [\![s]\!](\nu_i^a)$$
være kontinuerlig ved Lemma 5.32b). Da er $[\![t]\!](\nu) = \eta(g(\nu))$, så $[\![t]\!]$ er definert som en sammensetning av kontinuerlige funksjoner.

Dette viser at hele tolkningen kan gjennomføres kontinuerlig i alle parametre, og at den derfor er veldefinert.

Vi kan legge merke til at vi av notasjonsmessige grunner lot tilordningene omfatte alle variable, ikke bare de frie. Det er selvfølgelig slik at det bare er verdien på de fri variablene som har noen innflytelse på tolkningen. Det betyr spesielt at enhver lukket term har en entydig tolkning som et element i X.

Hvorvidt modellen er interessant, avhenger av applikasjonsstrukturen. Legg merke til at $X_0 = \{\bot\}$ er en applikasjonsstruktur ettersom X_0 er isomorf til $X_0 \to X_0$. Denne gir imidlertid ikke noen spennende modell for utypet λ-kalkyle. Før vi fortsetter slike spekulasjoner kan det imidlertid være greit å vise at vi faktisk har definert en modell, det vil si, vår tolkning respekterer omskrivningsreglene.
Det er selvfølgelig β-regelen vi må kontrollere.

Lemma 5.39 *La t være en term og la s være substituerbar for x_i i t. La ν være en tilordning. Da vil*

$$[\![t]\!](\nu_i^{[\![s]\!](\nu)}) = [\![t^s_{x_i}]\!](\nu).$$

Bevis
Beviset er ved en triviell induksjon på oppbyggingen av t.

Lemma 5.40 *For alle termer t og s, alle variable x_i slik at s er substituerbar for x_i i t og alle tilordninger ν har vi*

$$[\![(\lambda x_i.t)s]\!](\nu) = [\![t^s_{x_i}]\!](\nu).$$

Bevis
Vi bruker foregående lemma og alle definisjonene. La $f_\nu : X \to X$ være definert ved

$$f_\nu(a) = [\![t]\!](\nu_i^a).$$

Da har vi

$$[\![(\lambda x_i.t)s]\!](\nu) = App([\![\lambda x_i.t]\!](\nu), [\![s]\!](\nu)) = App(\eta(f_\nu), [\![s]\!](\nu))$$

$$= \pi(\eta(f_\nu))([\![s]\!](\nu)) = f_\nu([\![s]\!](\nu)) = [\![t]\!](\nu_i^{[\![s]\!](\nu)}) = [\![t^s_{x_i}]\!](\nu).$$

5.5 Oppgaver til Kapittel 5

Oppgave 5.1 Definer *Plateks hierarki* ved

1. $P(\iota) = \mathbb{N}_\perp$ med ordning som i teksten.
2. $P(o) = \mathbb{B}_\perp$ med ordning som i teksten.
3. $P(\sigma \to \tau)$ vil bestå av alle monotone funksjoner fra $P(\sigma)$ til $P(\tau)$ ordnet ved den punktvise ordningen.

a) Vis at hvis $\{a_n\}_{n\in\mathbb{N}}$ er en voksende følge i $P(\sigma)$, så vil følgen ha en minste øvre grense $a \in P(\sigma)$.

b) Vis at hver $P(\sigma)$ har et minste element, som vi kan kalle $\perp_\sigma$.

c) [Vanskelig uten kunnskap om mengdelære] Vis at hvis $F \in P(\sigma \to \sigma)$, så finnes det en minste $a \in P(\sigma)$ slik at $F(a) = a$.

Oppgave 5.2 Vis Lemma 5.11.

Oppgave 5.3 Vis Lemma 5.20.
Legg merke til at eksistensen av den største x hhv. den minste y er en underforstått del av påstanden i Lemma 5.20

Oppgave 5.4 Vis at funksjonene η og π definert i beviset for Teorem 5.21 er kontinuerlige

Oppgave 5.5 Bevis Lemma 5.25.

Oppgave 5.6 Bevis Lemma 5.31.

Oppgave 5.7 La X og Y være to Scott-domener.
Vis at det finnes kontinuerlige ep-par fra både X og Y til både $X \times Y$ og $X \oplus Y$.

Oppgave 5.8 La X være et applikasjonsdomene hvor (η, π) er et ep-par fra $(X \to X)$ til X. La $[\![\cdot]\!]_X$ være tolkningen av termer over domenet X.

a) Vis at hvis det finnes to lukkede termer t og s slik at $[\![t]\!]_X \neq [\![s]\!]_X$, så vil $[\![tt]\!]_X \neq [\![ff]\!]_X$ hvor tt og ff er termene i utypet λ-kalkyle som representerer sannhetsverdiene.
Hint: Bruk termen som representerer kondisjonalen $\supset$.

b) Vis at hvis $[\![tt]\!]_X \neq [\![ff]\!]_X$, så er $[\![k_0]\!]_X \neq [\![k_1]\!]_X$.

c) Vis at hvis det finnes to lukkede termer som tolkes forskjellig, såvil alle $[\![k_n]\!]_X$ være forskjellihe når n varierer.

d) Vis at hvis relasjonen $[\![t]\!]_X = [\![s]\!]_X$ er avgjørbar som en relasjon mellom lukkede termer i utypet λ-kalkyle, så vil alle lukkede termer r tolkes likt i X.

Kapittel 6

Plotkins PCF

I 1977 kom det to artikler i den (den gang) forholdsvis nye journalen *Theoretical Computer Science*, av Robin Milner i nr. 4 og av Gordon Plotkin i nr. 5. Milner tok utgangspunkt i Scotts LCF og det faktum at det finnes kompakter i Scott-hierarkiet som ikke kan defineres i LCF, og viste at det finnes ett og bare ett hierarki over $\mathbb{N}_\perp$ og $\mathbb{B}_\perp$ av algebraiske domener som kan tjene som en modell for LCF og hvor alle kompaktene er LCF-definerbare. Plotkin på sin side skrev LCF om til en typet λ-kalkyle PCF og viste noen meget viktige sammenhenger mellom LCF og standardmodellen via Scotts hierarki av algebraiske domener.
Ettersom vi ikke vil innføre LCF, velger vi å gå gjennom hovedresultatene i disse to artiklene i omvendt rekkefølge.

6.1 Kalkylen

La oss minne om ingrediensene i kjerneeksemplet på en typet λ-kalkyle, fra før vi gikk over til Gödels T.
Vi hadde konstanter 0, tt, ff, S, P, Z, $\supset_o$ og $\supset_\iota$ med tilhørende omskrivningsregler. PCF, eller det som i litteraturen ofte kalles *den spesielle* PCF, er en tilnærmet utvidelse av vårt kjerneeksempel med sine omskrivningsregler. Vi gir den fulle definisjonen, ettersom det er en del praktiske endringer vi vil gjøre i forhold til endelig typeteori.

Definisjon 6.1 Vi lar PCF være den typede λ-kalkylen definert som følger:

1. ι og o er basistyper, og hvis σ og τ er typer, er $\sigma \to \tau$ en type.
   Alle typer fremkommer på denne måten.

2. PCF har typede variable x_i^σ for alle typer σ og naturlige tall i. x_i^σ er av type σ.

3. PCF har følgende typede konstanter:
   Ω_ι og k_n av type ι for alle $n \in \mathbb{N}$,
   Ω_o, tt og ff av type o,
   S og P av type $\iota \to \iota$, Z av type $\iota \to o$, $\supset_\iota$ av type $o, \iota, \iota \to \iota$, $\supset_o$ av type $o, o, o \to o$
   og for hver type σ, konstanten Y_σ av type $(\sigma \to \sigma) \to \sigma$.

4. De *typede termene* i PCF er bygget opp fra variablene og konstantene ved sammensetning og λ-abstraksjon.

5. PCF er underlagt følgende omskrivningsregler:

   - α- og β-reglene for typet λ-kalkyle.
   - $S\Omega_\iota \to \Omega_\iota$.
   - $Sk_n \to k_{n+1}$ for alle $n \in \mathbb{N}$.
   - $P\Omega_\iota \to \Omega_\iota$.
   - $Pk_0 \to \Omega_\iota$.
   - $Pk_{n+1} \to k_n$ for alle n.
   - $Z\Omega_\iota \to \Omega_o$.
   - $Zk_0 \to tt$.
   - $Z(k_{n+1}) \to ff$ for alle n.
   - $\supset_b \Omega_o sr \to \Omega_b$ (b basistype).
   - $\supset_b (tt)sr \to s$ (b basistype).
   - $\supset_b (ff)sr \to r$ (b basistype).
   - $Y_\sigma t \to t(Y_\sigma t)$ for alle typer σ.

Konstantene Y_σ og omskrivningsreglene for dem er nye ingredienser i PCF i forhold til endelig typeteori, og de kommer som en erstatning for, og utvidelse av, konstantene for rekursjon i Gödels T.
Vi ser at ingen term som inneholder en delterm på formen $Y_\sigma t$ kan være sterkt normaliserbar, ettersom vi får omskrivningen

$$Y_\sigma t \to t(Y_\sigma t) \to t(t(Y_\sigma t)) \to \cdots.$$

Det beste vi derfor kan håpe på er at noen slike termer er normaliserbare, og at vi kan finne omskrivningsstrategier som finner normalformen til normaliserbare termer.
Konstantene Ω_ι og Ω_o skal betegne den udefinerte størrelsen, som i modellen vil betegne bunnelementet. I LCF aksiomatiserte Scott en ordning mellom termer, og da var et av aksiomene at Ω_σ er minst for hver type σ. Vi har ikke innført noen ordning. Likevel vil vi definere termen Ω_σ for alle typer σ, og tenke på Ω_σ som det minste objektet av type σ. Dette blir klarere når vi ser på modellen for PCF.

Definisjon 6.2 Hvis $\sigma = \delta \to \tau$ og Ω_τ er definert, lar vi

$$\Omega_\sigma = \lambda x^\delta.\Omega_\tau$$

Det neste vi skal se på er at vi kan løfte $\supset_\iota$ og $\supset_o$ til typer generelt.

Definisjon 6.3 Anta at $\supset_\tau$ er definert, og av type $o, \tau, \tau \to \tau$
Da lar vi

$$\supset_{\sigma\to\tau} = \lambda x^o \lambda y^{\sigma\to\tau} \lambda z^{\sigma\to\tau} \lambda w^\sigma. \supset_\tau x(yw)(zw).$$

Lemma 6.4 *For alle typer $\sigma = \tau_1, \ldots, \tau_n \to b$ hvor b er en basistype og for alle termer t og s av type σ vil*

- $\supset_\sigma (tt)ts \to \lambda w_1^{\tau_1} \cdots \lambda w_n^{\tau_n}.tw_1 \ldots w_n$
- $\supset_\sigma (ff)ts \to \lambda w_1^{\tau_1} \cdots \lambda w_n^{\tau_n}.sw_1 \ldots w_n$

Bevis
Vi viser dette ved induksjon på n.
For $n = 0$ er dette omskrivningsreglene for de to kondisjonalene, så induksjonstarten er triviell.
Anta at lemmaet holder for $\sigma_1 = \tau_1, \ldots, \tau_n \to b$. Vi vil vise at da holder det også for $\sigma = \tau, \tau_1, \ldots, \tau_n \to b$. Vi begrenser oss til det ene tilfellet, ettersom bevisene i de to tilfellene er tilnærmet like. Vi dropper typebetegnelser der de bare er i veien.
Vi har da

$\supset_\sigma (tt)ts = \lambda x \lambda y \lambda z \lambda w^\tau. \supset_{\sigma_1} (tt)ts$

$\to \lambda w^\tau \supset_{\sigma_1} (tt)(tw)(sw)$
$\to \lambda w^\tau \lambda w^{\tau_1} \dots \lambda w^{\tau_n}.tww_1 \dots w_n.$

I den siste overgangen bruker vi induksjonsantagelsen.

Bemerkning 6.5 Ideelt sett skulle vi gjerne hatt at $\supset_\sigma (tt)ts \to t$ og at $\supset_\sigma (ff)ts \to s$, men for å få til det trenger vi en ekstra generell omskrivningsregel kjent som η-regelen. η-regelen sier at hvis t er av type $\sigma \to \tau$, så vil $\lambda x^\sigma tx \to t$. I Oppgave 6.1 skal vi se at semantisk så har vi η-regelen, så vi kunne godt lagt den til PCF uten at vi endret sunnhet av semantikken.

Før vi gir oss i kast med en semantikk for PCF, la oss se at vi kan programmere alle μ-beregnbare funksjoner i PCF. Dette vil delvis bli gjentagelse av beviset for at alle μ-beregnbare funksjoner er λ definerbare, og leseren som innser dette, kan droppe deler av resten av dette avsnittet.
Først vil vi vise at Gödels T kan implementeres i PCF. Vi får ikke et fullt så sterkt resultat som vi kunne ønsket, vi må bruke ekvivalensrelasjonen $\equiv$ som vi så på i utypet λ-kalkyle, men som her vil bety at to termer kan skrives om til samme term i henhold til reglene for PCF. Videre kunne rekursjon i T gjennomføres ved hjelp av latere evalueringsstrategier enn de vi må bruke i PCF.

Lemma 6.6 *For hver type σ finnes det en term R_σ av samme type som Rec_σ slik at for alle termer s og t av relevant type vil*

$$R_\sigma stk_0 \to s$$

$$R_\sigma st(k_{n+1}) \equiv tk_n(R_\sigma stk_n) \textit{ for alle } n.$$

Bevis
La
$$R_\sigma = \lambda x^\sigma \lambda y^{\iota,\sigma\to\sigma} Y_{\iota\to\sigma}(\lambda f^{\iota\to\sigma} \lambda z^\iota \supset_\iota (Zz)x(y(Pz)f(Pz))).$$

Vi dropper indeksene i den videre utregningen. Vi viser at de to omskrivningene for Rec kan realiseres i PCF ved hard og brutal utregning:

- Ved å sette inn og foreta to gangers bruk av β-regelen kan vi skrive $R_\sigma stk_0$ om til
$$Y(\lambda f \lambda z. \supset (Zz)s(t(Pz)f(Pz)))k_0.$$

Bruker vi omskrivningsregelen for Y får vi at dette skrives om til

$$\lambda f \lambda z. \supset (Zz)s(t(Pz)f(Pz))(Y(\lambda f \lambda z. \supset (Zz)s(t(Pz)f(Pz))))k_0$$

som jo begynner å bli en smule uoversiktlig. For å lette oversikten, innfører vi forkortelsen Φ for $Y(\lambda f \lambda z. \supset (Zz)s(t(Pz)f(Pz)))$ og får

$$\lambda f \lambda z. \supset (Zz)s(t(Pz)f(Pz))(\Phi)k_0.$$

Ved å bruke β-regelen to ganger får vi

$$\supset (Zk_0)s(t(Pk_0)\Phi(Pk_0)).$$

Siden Zk_0 skrives om til tt og $\supset (tt)s(t(Pk_0)\Phi(Pk_o))$ skrives om til s er vi fremme.

- Vi lar Φ være som over.
  Vi skriver $R_\sigma st(k_{n+1})$ om til

  $$Y(\lambda f \lambda z. \supset (Zz)s(t(Pz)f(Pz)))(k_{n+1})$$

  som etter at vi har brukt omskrivningsregelen for Y og brukt forkortelsen Φ blir til

  $$\lambda f \lambda z. \supset (Zz)s(t(Pz)f(Pz))(\Phi)(k_{n+1}).$$

  To gangers bruk av β-regelen gir oss da

  $$\supset (Z(k_{n+1}))s(t(P(k_{n+1}))(\Phi(P(k_{n+1})))),$$

  som, når vi bruker $Z(k_{n+1}) \to ff$, $P(k_{n+1}) \to k_n$ og regelen for $\supset$, kan skrives om til $tk_n(\Phi k_n)$.

  På den annen side, ser vi på $R_\sigma stk_n$ og bruker β-regelen to ganger, får vi Φk_n, så dette viser at $R_\sigma st(k_{n+1}) \equiv tk_n(\Phi k_n) \equiv tk_n(R_\sigma stk_n)$ og vi er fremme.

Den andre ingrediensen vi trenger for å vise at alle generelt beregnbare funksjoner fra $\mathbb{N}$ til $\mathbb{N}$ er programmerbare i PCF er μ-operatoren. Vi skal innføre en term M og bevise de vesentlige egenskapene for M:

Lemma 6.7 *Det finnes en lukket term M av type $(\iota \to \iota) \to \iota$ slik at for enhver lukket term t av type $\iota \to \iota$ og hver numeral k_n har vi at følgende er ekvivalente:*

1. $Mt \to k_n$

2. $tk_n \to k_0$ *og for alle* $m < n$ *finnes* $i \neq 0$ *slik at* $tk_m \to k_i$

Bevis
Vi har gitt intuisjonen rundt det å beskrive μ-operatoren som en fikspunkt-konstruksjon før, så vi gyver rett løs på definisjonen:

$$M = \lambda f^{\iota\to\iota}.Y_{\iota\to\iota}(\lambda g^{\iota\to\iota}\lambda x^{\iota}. \supset_\iota (Z(fx))x(S(g(Sx))))k_0.$$

Vi ser at Mt vil reduseres til

$$Y_{\iota\to\iota}(\lambda g^{\iota\to\iota}\lambda x^{\iota}. \supset_\iota (Z(tx))x(S(g(Sx))))k_0.$$

Vi viser ved induksjon på m at for alle n vil

$$Y_{\iota\to\iota}(\lambda g^{\iota\to\iota}\lambda x^{\iota}. \supset_\iota (Z(tx))x(S(g(Sx))))k_n \to k_{n+m} \tag{6.1}$$

hvis og bare hvis vi for alle $i \leq m$ har at tk_{n+i} kan skrives om til en numeral, og m er minimal slik at denne numeralen er k_0. Lemmaet vil følge fra 6.1.
For å bevise "hvis"-retningen av 6.1 bruker vi først omskrivningsregelen for $Y_{\iota\to\iota}$ en gang. For $m = 0$ ser vi da at påstanden holder direkte, mens for $M > 0$ må vi bruke at påstanden holder for $n + 1$.
For å vise "bare hvis"-retningen må vi bruke at termen til venstre i 6.1 bare kan bringes på normalform i forbindelse med nullpunkter for t, ellers blir vi tvunget til å bruke omskrivningsreglene for $Y_{\iota\to\iota}$ igjen og igjen.
Utregningene blir ganske uoversiktlige, og gjøres best i enerom på eget papir.

6.2 Scott-modellen for PCF

Vi skal nå bruke Scott-hierarkiet som base for en tolkning av termene i PCF. Vi gjennomførte mye av det tekniske da vi viste at applikasjonsdomener kan brukes til å gi modeller for utypet λ-kalkyle, men vi trenger et nytt generelt lemma fra domeneteori før vi går i gang.

Lemma 6.8 *La* X *være en cpo og la* $FP : (X \to X) \to X$ *være funksjonen som til en kontinuerlig* $f : X \to X$ *gir oss det minste fikspunktet til* f.
Da er FP *kontinuerlig.*

Bevis
Hvis $f : X \to X$ er kontinuerlig, vil $FP(f) = \bigsqcup\{f^n(\bot) \mid n \in \mathbb{N}\}$. Hvis $f \sqsubseteq_{X\to X} g$ vil $f^n(\bot) \sqsubseteq g^n(\bot)$ for hver n, så $FP(f) \sqsubseteq FP(g)$.
Hvis $A \subset X \to X$ er rettet, og $g = \bigsqcup_{X\to X} A$ har vi

$$FP(g) = \bigsqcup\{g^n(\bot) \mid n \in \mathbb{N}\} = \bigsqcup\{f^n(\bot) \mid n \in \mathbb{N} \wedge f \in A\} = \bigsqcup\{FP(f) \mid f \in A\}$$

ettersom en dobbelgrense kan tas som to enkeltgrenser i den rekkefølge man vil.
Dette avslutter beviset.

Definisjon 6.9 La t være en PCF-term av type σ med variable blant $x_1, \dots, x_n$ av typer $\tau_1, \dots, \tau_n$.
For alle typer τ, la $D(\tau)$ være Scott-domenet av type τ.
Vi tolker t som en kontinuerlig funksjon $[\![t]\!]$, hvor vi bruker bokstavene a, b, c, d og f for å betegne domeneelementer. $\bot$ vil alltid være bunnelementet i det aktuelle domenet, og n vil som normalt betegne et vilkårlig naturlig tall.

1. Hvis $t = x_i$ lar vi $[\![t]\!](a_1, \dots, a_n) = a_i$.
2. $[\![\Omega_\iota]\!] = \bot_\iota$ og $[\![\Omega_o]\!] = \bot_o$.
3. Hvis t er en av konstantene k_n. tt, ff, S, P, Z, $\supset_\iota$ eller $\supset_o$ lar vi $[\![t]\!]$ være den kanoniske tolkningen av denne konstanten, hvor vi dropper henvisningen til argumentene:

   $[\![k_n]\!] = n$, $[\![tt]\!] = \mathbf{T}$, $[\![ff]\!] = \mathbf{F}$
   $[\![S]\!](n) = n + 1$, $[\![S]\!](\bot_\iota) = \bot_\iota$
   $[\![P]\!](n + 1) = n$, $[\![P]\!](0) = \bot_\iota$, $[\![P]\!](\bot_\iota) = \bot_\iota$
   $[\![Z]\!](0) = \mathbf{T}$, $[\![Z]\!](n + 1) = \mathbf{F}$, $[\![Z]\!](\bot_\iota) = \bot_o$
   $\supset_{\iota/o} (\mathbf{T}, a, b) = a$, $\supset_{\iota/o} (\mathbf{F}, a, b) = b$ og $\supset_{\iota/o} (\bot_o, a, b) = \bot_{\iota/o}$.

4. Hvis $t = Y_\sigma$ lar vi $[\![t]\!]$ være "den minste fikspunkt"-operatoren på $D_{\sigma\to\sigma}$, se Lemma 6.8.
5. Hvis $t = rs$ lar vi $[\![t]\!](a_1, \dots, a_n) = [\![r]\!](a_1, \dots, a_n)([\![s]\!](a_1, \dots, a_n))$
6. Hvis $t = \lambda x_i^\tau.s$ er av type $\tau \to \sigma$, lar vi $[\![t]\!](a_1, \dots, a_n)$ være funksjonen fra $D(\tau)$ til $D(\sigma)$ definert ved

   $$[\![t]\!](a_1, \dots, a_n)(a) = [\![s]\!](a_1, \dots, a_{i-1}, a, a_{i+1}, \dots, a_n).$$

I lys av Lemma 6.8 og tilsvarende konstruksjon for utypet λ-kalkyle ser vi at $[\![\cdot]\!]$ er veldefinert.

6.3 Konsistens - adekvathet - full abstraksjon

For ordens skyld bør vi slå fast at Scott-hierarkiet gir oss en konsistent modell for PCF:

Lemma 6.10 *La t og s være termer i PCF slik at $t \to s$.*
Da vil $[\![t]\!] = [\![s]\!]$.

Bevis
Beviset for at bruk av α-regelen og β-regelen ikke endrer verdien av tolkningen er som for konsistensbeviset for modellene for utypet λ-kalkyle. Det gjenstår derfor å sjekke at tolkningene av konstantene respekterer omskrivningsreglene, og det er et møysommelig men trivielt arbeid, møysommelig fordi det er så mange konstanter.

Bemerkning 6.11 Hvis vi hadde beholdt regelen $Z(St) \to ff$ ville vi ikke kunne bevist konsistens av modellen. Da skulle vi nemlig på den ene siden hatt at $[\![Z]\!]([\![S]\!](\bot_\iota)) = \mathbf{F}$, mens vi på den annen side vil ha at $[\![Z]\!](\bot_\iota) = \bot_o$ og at $[\![S]\!](\bot_\iota) = \bot_\iota$

6.3.1 Plotkins adekvathetsteorem

Da vi studerte Gödels T, beviste vi en egenskap vi kalte "adekvathet". Det var at hvis t er en lukket term av basistype, og $[\![t]\!]$ er et objekt i tolkningsrommet, så vil t kunne skrives om til den tilsvarende normalformen. Den gang var beviset trivielt, det bygget på at vi har sterk normalisering i T. Vi har ikke noe i nærheten av sterk normalisering for PCF, men Plotkin har vist at PCF også har denne egenskapen. Vi skal nå gi oss i kast med *Plotkins adekvathetsteorem*:

Teorem 6.12 *La t være en lukket PCF-term av basistype og anta at $[\![t]\!] \neq \bot$. Da kan t skrives om til normalformen som svarer til $[\![t]\!]$.*

Beviset for Teorem 6.12 vil også gi oss en strategi for hvordan vi kan normalisere en term.
Vi trenger å jobbe litt før vi kan bevise adekvathetsteoremet.

Definisjon 6.13 Vi definerer $[\![t]\!]_n$, den *n'te tilnærmingen til* $[\![t]\!]$ som følger:

1. Hvis t er x_i^τ eller t er en av konstantene Ω_ι, k_m, Ω_o, tt, ff, S, P, Z, $\supset_\iota$ eller $\supset_o$, lar vi $[\![t]\!]_n(\vec{a}) = [\![t]\!](\vec{a})$ for alle n.

2. $[\![Y_\sigma]\!]_n(\vec{a})(f) = f^n(\perp_\sigma)$ når $f \in D(\sigma \to \sigma)$.

3. Hvis $t = sr$ hvor s er av type $\sigma \to \tau$ og r er av type σ, lar vi

   - $[\![t]\!]_0(\vec{a}) = \perp_\tau$
   - $[\![t]\!]_{n+1}(\vec{a}) = [\![s]\!]_n(\vec{a})([\![r]\!]_n(\vec{a}))$.

4. Hvis t er av type $\lambda x_i^\sigma.s$ hvor s er av type τ, lar vi

   - $[\![t]\!]_0(\vec{a})$ være konstantfunksjonen $\perp_\tau$ definert på $D(\sigma)$.
   - $[\![t]\!]_{n+1}(\vec{a})(a) = [\![s]\!](a_1, \ldots, a_{i-1}, a, a_{i+1}, \ldots, a_n)$.

Lemma 6.14 *La t være en PCF-term av type σ med fri variable blant $\vec{x}$ av typer $\vec{\tau}$ og la $\vec{a}$ være objekter av typer $\vec{\tau}$ i Scotts hierarki.*

a) For alle n vil
$$[\![t]\!]_n(\vec{a}) \sqsubseteq_\sigma [\![t]\!]_{n+1}(\vec{a}) \sqsubseteq_\sigma [\![t]\!](\vec{a}).$$

b) Hvis p er en kompakt i $D(\sigma)$ og $p \sqsubseteq_\sigma [\![t]\!](\vec{a})$ finnes det en $n \in \mathbb{N}$ slik at $p \sqsubseteq_\sigma [\![t]\!]_n(\vec{a})$.

Bevis
a) bevises trivielt ved induksjon på oppbyggingen av t, og vi går ikke inn på detaljene.
Vi viser også b) ved induksjon på oppbygging av termen.

1. Hvis t er en variabel eller t er en av konstantene forskjellig fra alle Y_σ følger lemmaet fra definisjonen.

2. Hvis $t = Y_\sigma$ er $[\![t]\!] = \bigsqcup_{(\sigma\to\sigma)\to\sigma}\{\lambda f \in D(\sigma \to \sigma) f^n(\perp_\sigma) \mid n \in \mathbb{N}\}$.
   La $h \in D((\sigma \to \sigma) \to \sigma)$ være kompakt slik at $h \sqsubseteq_{(\sigma\to\sigma)\to\sigma} [\![Y_\sigma]\!]$.
   Da finnes det kompakter $p_1, \ldots, p_k$ i $D(\sigma \to \sigma)$ og kompakter $q_1, \ldots, q_k$ i $D(\sigma)$ slik at
   $$h(f) = \bigsqcup_\sigma \{q_i \mid p_i \sqsubseteq_{\sigma\to\sigma} f\}.$$

Vi kan uten tap av generalitet anta at $h(p_i) = q_i$ for alle $i = 1, \ldots, k$.
Siden $h \sqsubseteq Y_\sigma$ har vi spesielt for hver i at

$$q_i = h(p_i) \sqsubseteq [\![Y_\sigma]\!](p_i) = \bigsqcup\{p_i^n(\bot_\sigma) \mid n \in \mathbb{N}\}.$$

For hver $i = 1, \ldots, k$ finnes det da n_i slik at $q_i \sqsubseteq p_i^{n_i}(\bot_\sigma)$.
La $n = max\{n_i \mid 1 \leq i \leq k\}$. Da er $h \sqsubseteq [\![Y_\sigma]\!]_n$.

3. La $t = sr$ og la $p \sqsubseteq [\![s]\!](\vec{a})([\![t]\!](\vec{a}))$.
Da finnes det kompakt $p_1 \sqsubseteq [\![s]\!](\vec{a})$ og kompakt $p_2 \sqsubseteq [\![r]\!](\vec{a})$ slik at $q \sqsubseteq p_1(p_2)$.
Ved induksjonsantagelsen finnes n_1 og n_2 slik at $p_1 \sqsubseteq [\![s]\!]_{n_1}(\vec{a})$ og $p_2 \sqsubseteq [\![r]\!]_{n_2}(\vec{a})$.
La $n > max\{n_1, n_2\}$.
Da vil $q \sqsubseteq [\![t]\!]_n(\vec{a})$.

4. La $t = \lambda x_i^\tau.s$ hvor s er av type σ, og la $p \sqsubseteq [\![t]\!](\vec{a})$ være kompakt.
Da finnes det kompakter $p_1, \ldots, p_k$ av type τ og $q_1, \ldots, q_k$ av type σ slik at vi for alle $a \in D(\tau)$ har at

$$p(a) = \bigsqcup\{q_i \mid p_i \sqsubseteq a\}.$$

Som under punkt 2. antar vi at $p(p_i) = q_i$.
Da er

$$q_i \sqsubseteq [\![s]\!]((\vec{a})_i^{p_i})$$

for hver $i = 1, \ldots, k$, og ved induksjonsantagelsen finnes n_i slik at

$$q_i \sqsubseteq [\![s]\!]_{n_i}((\vec{a})_i^{p_i}).$$

La $n > max\{n_i \mid 1 \leq i \leq k\}$. Da vil $p \sqsubseteq [\![t]\!]_n(\vec{a})$.

Dette avslutter beviset for lemmaet.

Vi vil nå innføre en alternativ kalkyle som vi kaller $w - PCF$:

Definisjon 6.15 La $w - PCF$ (*weak* PCF) være som PCF hvor hver konstant Y_σ erstattes av en sekvens av konstanter $\{Y_{n,\sigma}\}_{n \in \mathbb{N}}$ med regler

$$Y_{n+1,\sigma}t \to t(Y_{n,\sigma}t).$$

Vi lar $[\![Y_{n,\sigma}]\!](f) = f^n(\bot_\sigma)$ når $f \in D(\sigma \to \sigma)$
Hvis t er en term i PCF, lar vi $t[n]$ være termen vi får ved å erstatte alle forekomster av Y_σ for alle σ med $Y_{n,\sigma}$ i t.

Lemma 6.16 *Vi har følgende sammenhenger mellom PCF og $w - PCF$*

a) *Hvis t og s er termer av samme type i PCF hvor s ikke har noen forekomster av noen Y_σ er følgende ekvivalente:*

   b) *$t \to s$ i PCF.*

   1. *Det finnes en n slik at $t[n] \to s$ i $w - PCF$.*

- *Hvis t er en lukket term av basistype er følgende ekvivalente:*

   1. *$[\![t]\!] \neq \perp$*
   2. *Det finnes en n slik at $[\![t[n]]\!] \neq \perp$.*

Bevis
Vi viser a) først.
2. $\Rightarrow$ 1.: Enhver omskrivningsfølge fra $t[n]$ i henhold til reglene for $w - PCF$ resulterer i en omskrivningsfølge fra t ved for alle m og σ å erstatte alle forekomster av $Y_{m,\sigma}$ i omskrivningskjeden med Y_σ. Det følger at hvis $t[n] \to_{w-PCF} s$ hvor s er som over, vil $t \to_{PCF} s$.
1. $\Rightarrow$ 2.: Anta at $t \to_{PCF} s$ hvor s ikke har noen forekomst av fikspunktkonstanten. La m være lengden på omskrivingskjeden. Ved induksjon på m ser vi at om $n \geq m$ vil $t[n] \to_{w-PCF} s$ via en omskrivning av samme lengde.
La oss så vise b).
1. $\Rightarrow$ 2. er en konsekvens av Lemma 6.14 og 2. $\Rightarrow$ 1. følger av at $[\![Y_{n,\sigma}]\!] \sqsubseteq [\![Y_\sigma]\!]$ for alle n og σ.
Dermed er lemmaet bevist.

Fra Lemma 6.16 følger det at det er tilstrekkelig å vise Teorem 6.12 for $w - PCF$. For å gjøre det skal vi erstatte $w - PCF$ med en ekvivalent kalkyle $vw - PCF$.

Definisjon 6.17 La $vw - PCF$ (*very weak* PCF) være PCF hvor vi fjerner alle konstantene Y_σ og alle tilhørende omskrivningsregler.

Lemma 6.18 *$w - PCF$ kan reduseres til $vw - PCF$ både formelt og semantisk.*

Bevis
Vi erstatter $Y_{n,\sigma}$ med $\lambda y^{\sigma\to\sigma}.y^n\Omega_\sigma$.
Omskrivningsreglene fra $w - PCF$ kan da realiseres via regler for $vw - PCF$.

Vi vil også ha at tolkningene er de samme.

Det er nå tilstrekkelig å vise adekvathetsteoremet for $vw - PCF$. Det vil følge av sterk normalisering av $vw - PCF$, noe som vises like lett som sterk normalisering av endelig typeteori.
Vi har imidlertid bare behov for svak normalisering av lukkede termer av basistype, og det kan vi oppnå ved et enklere bevis. Vårt bevis er i hovedsak Plotkins argument.

Definisjon 6.19 La t være en term i $vw - PCF$. Vi sier at t er *utregnbar* hvis

1. t er lukket, av basistype og normaliserbar.
2. t er lukket, av type $\sigma \to \tau$ og for alle lukkede utregnbare termer s av type σ er ts utregnbar av type τ
3. Hvis de fri variablene i t er blant $\vec{x}$ og $\vec{u}$ er lukkede, utregnbare termer av typene til $\vec{x}$, så er $t^{\vec{u}}_{\vec{x}}$ utregnbar.

Lemma 6.20 *Hvis t er en term og t kan skrives om til en utregnbar term, er t selv utregnbar.*

Bevis
Beviset er trivielt ut fra definisjonen ved induksjon på typen. Merk at hvis t er lukket, så er ikke 3. i konflikt med 1. eller 2. Dette vil forenkle utskrivningen av det neste beviset:

Lemma 6.21 *Alle termer er utregnbare.*

Bevis
Vi bruker induksjon på oppbyggingen av termen. Beviset deler seg opp i tilfeller ut fra den syntaktiske formen til t:

1. t er en variabel. Da følger påstanden trivielt fra definisjonen.
2. t er en av konstantene k_n, tt, ff, Ω_ι eller Ω_o. Da er t på normalform og av basistype, så t er utregnbar.
3. t er en av konstantene S, P, Z, $\supset_\iota$ eller $\supset_o$.
   Disse tilfellene er nokså like, og siden Plotkin velger å se på P, velger vi å se på $\supset_\iota$.

Da krever definisjonen at vi skal vise at $\supset_\iota t_1 t_2 t_3$ er normaliserbar når t_1 er en lukket, normaliserbar term av type o og t_2 og t_3 er lukkede normaliserbare termer av type ι Det er tre deltilfeller:

i) $t_1 \to tt$: Da vil $t \to t_2$ som er normaliserbar pr. antagelse.

ii) $t_1 \to ff$: Da vil $t \to t_3$ som er normaliserbar pr. antagelse.

iii) $t_1 \to \Omega_o$. Da vil $t \to \Omega_\iota$ som er på normalform.

4. $t = sr$ hvor s er av type $\sigma \to \tau$ og r er av type σ.
Nå lar vi $\vec{x}$ og $\vec{u}$ være som over.
Ved induksjonsantagelsen er $s^{\vec{u}}_{\vec{x}}$ og $r^{\vec{u}}_{\vec{x}}$ utregnbare, og ved del 2. av definisjonen er sr utregnbar.

5. $t = \lambda x^\sigma.s$ hvor s er av type τ. La $\vec{x}$ og $\vec{u}$ være som over.
Vi skal vise at $(\lambda x.s)^{\vec{u}}_{\vec{x}}$ er utregnbar, og ved del 2 av definisjonen betyr det at vi skal vise at $(\lambda x.s)^{\vec{u}}_{\vec{x}} u$ er utregnbar hver gang u er en lukket, utregnbar term av type σ.
En gangs omskriving av $(\lambda x.s)^{\vec{u}}_{\vec{x}} u$ gir oss $s^{u,\vec{u}}_{x,\vec{x}}$ som er utregnbar ved induksjonsantagelsen. Det følger fra Lemma 6.20 at $(\lambda x.s)^{\vec{u}}_{\vec{x}} u$ er utregnbar.

Dette avslutter beviset.

Vi har nå lagt alt til rette for

Bevis for Teorem 6.12
La t være en lukket term i PCF av basistype, og anta at $[\![t]\!] = a \neq \bot$.
Da finnes det en n slik at $[\![t[n]]\!] = a$. Husk at $t[n]$ er en term i $w - PCF$.
La $\hat{t}[n]$ være termen i $vw - PCF$ som svarer til $t[n]$.
Siden $\hat{t}[n]$ er av basistype og utregnbar, er $\hat{t}[n]$ normaliserbar i $vw - PCF$.
Ved konsistens av modelen vår for $vw - PCF$ vil normalformen til $\hat{t}[n]$ ha a som tolkning.
Hvis vi nå går baklengs, ser vi at dette betyr at $t[n]$ kan normaliseres til den samme normalformen, og det følger at t kan normaliseres til normalformen som svarer til a.
Dette avslutter beviset.

Vi kan bruke den innsikten vi har vunnet til å beskrive en *normaliseringsstrategi* for lukkede termer av basistype.
La t være en lukket term av basistype. Da kan t normaliseres hvis og bare

hvis det finnes en n slik at $t[n]$ kan normaliseres til en term $\neq \bar{\Omega}_b$ innenfor $w - PCF$.
Hvis vi nå ser på argumentet for at $\hat{t}[n]$ er utregnbar i $vw - PCF$ fungerer det som følger: t vil være på formen $Kt_1 \cdots t_m$ hvor K ikke er en sammensetning. I utgangspunktet bruker vi at K, $t_1, \ldots, t_m$ alle er utregnbare som en induksjonshypotese. Hvis vi imidlertid beveger oss enda ett skritt ned, ser vi at den videre analysen avhenger av hvordan K ser ut.
Hvis K er en konstant av basis type, er $m = 0$ og vi er ved foten av induksjonen.
Hvis K er en konstant av en type på nivå 1, bruker vi at $t_1, \ldots, t_m$ alle er normaliserbare.
Hvis K er Y_σ står det egentlig en $Y_{n,\sigma}$ i termen vi vil vise er utregnbar, og induksjonsantagelsen der vil være ekvivalent med at $t_1(Y_{n-1,\sigma}t_1)$ er utregnbar.
I tilfelle $K = \lambda y.s$ viser vi at K er utregnbar fordi den videre induksjonsantagelsen gir oss at $s_y^{t_1}$ er utregnbar.

Dette gir oss følgende strategi for normalisering av lukkede termer av basistype

1. Hvis $t = k_n$, $t = tt$ eller $t = ff$ er t på normalform.

2. Hvis $t = St_1$, $t = Pt_1$ eller $t = Zt_1$, bruk strategien på t_1 og eventuell regel for P eller Z i de to siste tilfellene.

3. Hvis $t = \supset_b t_1t_2t_3$ bruk først strategien til å finne normalformen til t_1. Bruk deretter omskrivningsregelen for $\supset_b$ og så strategien igjen.

4. Hvis $t = Y_\sigma t_1 t_2 \cdots t_m$ skriv t om til $t_1(Y_\sigma t_1)t_2 \cdots t_m$ og følg strategien videre.

5. Hvis $t = (\lambda y.s)t_1 t_2 \cdots t_m$, skriv t om til $(s_y^{t_1})t_2 \cdots t_m$ og bruk strategien videre.

Vi skriver ikke ut beviset for at denne strategien virker, men poenget er at for hver omskrivning vi foretar oss, så beveger vi oss nedover i induksjonsargumentet for at alle termer i $vw - PCF$ er utregnbare, og så må vi ende opp med normalformer i $vw - PCF$. Hvis da t kan normaliseres i PCF er det denne normalformen vi finner.

6.3.2 Udefinerbare objekter

Definisjon 6.22 La σ være en type, og la $a \in D(\sigma)$.
Vi sier at a er *PCF-definerbar* hvis det finnes en lukket term t av type σ slik at $a = [\![t]\!]$.

Siden vi bare har tellbart mange PCF-termer og for eksempel $D(\iota \to o)$ er overtellbar, finnes det opplagt elementer i modellen vår som ikke er PCF-definerbare.
Det som imidlertid kan være overaskende på noen er at det finnes kompakter på et svært lavt nivå som ikke er PCF-definerbare.

Definisjon 6.23 Definer $\bigvee : \mathbb{B}_\perp^2 \to \mathbb{B}_\perp$ ved

1. $\bigvee(\mathbf{T}, \mathbf{T}) = \bigvee(\mathbf{T}, \mathbf{F}) = \bigvee(\mathbf{F}, \mathbf{T}) = \bigvee(\mathbf{T}, \perp) = \bigvee(\perp, \mathbf{T}) = \mathbf{T}$
2. $\bigvee(\mathbf{F}, \perp) = \bigvee(\perp, \mathbf{F}) = \bigvee(\perp, \perp) = \perp$
3. $\bigvee(\mathbf{F}, \mathbf{F}) = \mathbf{F}$

$\bigvee$ er en rimelig domeneteoretisk tolkning av $\vee$, men problemet er at $\bigvee$ ikke kan defineres i PCF. Før vi gir beviset, la oss peke på hva som egentlig er problemet: Det finnes to måter å verifisere at $\bigvee(a, b) = \mathbf{T}$, nemlig å verifisere at $a = \mathbf{T}$ og å verifisere at $b = \mathbf{T}$. PCF vil måtte velge en av disse, og hvis det mislykkes, vil PCF gi opp. Det er faktisk det at PCF tillater en sekvensiell evalueringsstrategi som umuliggjør at vi kan programmere $\bigvee$ i PCF.
Det faktum at det finnes et naturlig element i modellen vår som ikke er PVF-definerbart er et så viktig resultat at vi kaller det

Teorem 6.24 *$\bigvee$ er ikke PCF-definerbar.*

Bevis
Teoremet vil være en konsekvens av følgende mer generelle påstand:

> La b være en basistype, og la t være en lukket term av type b med to fri variable x og y av type o slik at både $t_{x,y}^{\Omega_o,tt}$ og $t_{x,y}^{tt,\Omega_o}$ kan skrives om til en normalform $\neq \Omega_b$.
> Da kan også $t_{x,y}^{\Omega_o,\Omega_o}$ skrives om til en normalform $\neq \Omega_b$.

Husk at i PCF vil en normalform representere et skikkelig tall eller en skikkelig sannhetsverdi. Merk også at siden $t_{x,y}^{tt,tt}$ vil ligge over både $t_{x,y}^{\Omega_o,tt}$ og $t_{x,y}^{tt,\Omega_o}$, vil de to normalformene være like.
Vi bruker induksjon på lengden av omskrivningen av $t_{x,y}^{tt,\Omega_o}$, hvor vi bruker strategien gitt i forrige avsnitt. Dette vil dele beviset opp i flere tilfeller:

1. $t = x$ eller $t = y$. Da vil ikke premissen i påstanden holde.

2. $t = \Omega_\iota$ eller $t = \Omega_o$. Da vil ikke premissen i påstanden holde.

3. $t = k_n$, $t = tt$ eller $t = ff$. Da holder konklusjonen i påstanden.

4. $t = St_1$, $t = Pt_1$, $t = Zt_1$, $t = \supset_\iota t_1t_2t_3$ eller $t = \supset_o t_1t_2t_3$.
Vi ser på det siste tilfellet, de andre er like vanskelige eller enklere. Vi må ha at $(t_1)^{\Omega_o,tt}_{x,y}$ og $(t_1)^{tt,\Omega_o}_{x,y}$ kan normaliseres, og til samme normalform $\neq \Omega_o$. Anta at denne er tt. Den andre muligheten gir et tilsvarende argument.
Ved induksjonsantagelsen vil $(t_1)^{\Omega_o,\Omega_o}_{x,y}$ også normaliseres til tt.
Videre kan i dette tilfellet $(t_2)^{\Omega_o,tt}_{x,y}$ og $(t_2)^{tt,\Omega_o}_{x,y}$ normaliseres til en term $\neq \Omega_o$, og ved induksjonsantagelsen kan $(t_2)^{\Omega_o,\Omega_o}_{x,y}$ også skrives om til samme normalform.
Det er imidlertid det som skal til for at påstanden holder i dette tilfellet.

5. $t = Y_\sigma t_1t_2 \cdots t_m$.
Vi kan bruke induksjonsantagelsen på termen
$$t_1(Y_\sigma t_1)t_2 \cdots t_m.$$

6. $t = (\lambda y.s)t_1t_2 \cdots t_m$.
Vi kan bruke induksjonsantagelsen på termen
$$s^{t_1}_y t_2 \cdots t_m.$$

Dette formatet på bevis kan brukes til så mangt, noe vi skal se et eksempel på om litt.
Enkelte hevder at det er mer et svakhetstegn ved PCF at $\bigvee$ ikke er PCF-definerbar enn et problem ved Scott-modellen. På den annen side liker informatikere sekvensielle evalueringsstrategier, slik at man slipper å teste i parallell langs et voksende antall grener. En mulig evalueringsstrategi for $\bigvee$ vil involvere en slik forgrening: For å bestemme $\bigvee t_1t_2$ evaluerer vi t_1 og t_2 i parallell. Får vi normalformen tt i et av tilfellene, er vi fornøyd, mens får vi normalformen ff i et av tilfellene fortsetter vi med evalueringen langs den andre grenen.
Vi kan formalisere dette gjennom en formell utvidelse av PCF:

Definisjon 6.25 La PCF^+ være PCF utvidet med en ny konstant $\bigvee$ av type $o, o \to o$ og følgende overgangsregler:

1. $\bigvee(tt)s \to tt$

2. $\bigvee s(tt) \to tt$

3. $\bigvee(ff)(ff) \to ff$

Plotkin viste at alle kompaktene i Scott-modellen er definerbare i PCF^+ (se Oppgave 6.2), men at det finnes beregnbare uendelige elementer i Scott-modellen som ikke er PCF^+ definerbare. Siden vi ikke har anledning til å gå inn på sammenhengen mellom domeneteori og beregnbarhet, skal vi la det ligge.

6.3.3 Full abstraksjon

Da vi studerte Gödels T, innførte vi et begrep "observabelt like". Dette er egentlig et begrep som er interessant for PCF og beslektede kalkyler, så vi definerer begrepet igjen. Vi minner om at et *program* er en term av basistype.

Definisjon 6.26 La t og s være to lukkede PCF-termer av samme type σ.

a) Vi sier at s og t er *observabelt like* hvis vi for alle program p med høyst en variabel x^σ fri og alle normalformer $e \neq \Omega$ av typen til p har

$$p^t_x \to e \Leftrightarrow p^s_x \to e.$$

b) En modell M for PCF er *fullt abstrakt* hvis vi har at $[\![t]\!]_M = [\![s]\!]_M$ hver gang s og t er observabelt like.

Vi skal avslutte dette avsnittet med å vise at Scott-modellen for PCF ikke er fullt abstrakt.

Teorem 6.27 *Det finnes to lukkede termer s og t som er observabelt like, men som tolkes forskjellig i Scott-modellen.*

Bevis
Vi vil bruke at likhet mellom elementer $\neq \bot$ i basistyper er avgjørbart i PCF. Når vi skriver et Boolesk uttrykk, mener vi den tilsvarende funksjonen som tar booleske verdier.
La s og t være termene av type $(o, o \to o) \to o$ definert ved

- $t = \lambda x.(x\Omega_o(tt) = x(tt)\Omega_o = tt)$

- $s = \lambda x.(x\Omega_o\Omega_o) = tt$

Vi har $[\![s]\!] \sqsubseteq [\![t]\!]$.
La p være et program med en fri variabel y av type $(o, o \to o) \to o$. La $e \neq \Omega$ være en normalform av basis type. Vi har opplagt at hvis $p^s_y \to e$ vil $p^t_y \to e$.
Den som likevel ikke synes at dette er opplagt bruker adekvathetsteroemet og observasjonen over.
For å bevise at s og t er observabelt like, må vi vise omvendingen, dvs.

$$p^t_y \to e \Rightarrow p^s_y \Rightarrow e.$$

Her bruker vi induksjon på lengden av utledningen av e fra p^t_y hvor vi følger strategien vår.
Alle induksjonsskrittene er trivielle, og det eneste tilfellet som er nytt i forhold til tidligere argumenter av samme slag er tilfellet hvor $p = yr$.
Da er antagelsen at $t(r^t_y) \to e$. For at dette skal være mulig må $(r^t_y)(tt)\Omega_o \to tt$ og $(r^t_y)\Omega_o(tt) \to tt$.
Fra beviset for at $\bigvee$ ikke er PCF-definerbar får vi da at $r^t_y\Omega_o\Omega_o \to e$.
Ved induksjonsantagelsen vil da $r^s_y\Omega_o\Omega_o \to e$, så $p^s_y \to t$.
Dette avslutter beviset.

6.4 Milner's modell

Robin Milner viste at det finnes et hierarki av Scott-domener som kan brukes til å gi en fullt abstrakt modell for PCF. Det vil gå for langt å gi en fullstendig definisjon av Milners modell, så dette blir langt påvei et "tegne og fortelle"-avsnitt.
Milners utgangspunkt var en term-modell, som igjen tar utgangspunkt i en naturlig preordning av de lukkede termene:

Definisjon 6.28 La s og t være to lukkede termer av samme type σ.
Vi sier at

$$s \prec t$$

hvis vi for alle programmer p med høyst en fri variabel x^σ og alle normalformer $e \neq \Omega$ av basistype har

$$p^s_x \to e \Rightarrow p^t_x \to e.$$

Vi sier at s er *observabelt mindre eller lik* t hvis $s \prec t$

Vi ser at s og t er observabelt like hvis og bare hvis $s \prec t$ og $t \prec s$, så dette er en naturlig relasjon å ta utgangspunkt i. Vi kan merke oss at $\prec$ ikke er en ordning, ettersom det ikke er en antisymmetrisk relasjon. Hvis vi imidlertid deler ut med observabelt likhet, som er en ekvivalensrelasjon, får vi en ordning. Milners tolkning $M(\sigma)$ av type σ er et Scott-domene vi får ved først å ta kompaktene i $\prec_\sigma$-ordningen av de lukkede termene av type σ, og deretter idealkompletteringen. Milners konstruksjon går via modeller for de begrensede versjonene $PCF(n)$ av PCF, hvor vi begrenser oss til konstantene $k_0, \ldots, k_n$ av type ι. Han viste at hvis vi gjennomfører konstruksjonen over for $PCF(n)$, får vi et hierarki av endelige Scott-domener hvor hvert element er definerbart i $PCF(n)$. Han brukte videre et system av ep-par for hver type σ fra $PCF(n)$-tolkningen av σ til $PCF(n+1)$-tolkningen av σ, og hans endelige modell får vi da ved å ta de rettede grensene for hver type.
I to artikler, den ene av Abramsky, Jagedeesan og Malacaria og den andre av Hyland og Ong, har man studert alternative tilnærminger til Milners modell. De karakteriserte kompaktene i Milners modell som det vi kan kalle *endelige sekvensielle prosedyrer*. Dette er et utplukk av PCF-termer av hver enkelt type som vil definere kompakter både i Scotts modell og i Milners modell.
Ralph Loader viste at Milners modell ikke er effektiv, i den forstand at det er umulig å representere kompaktene som en beregnbar partiell ordning på de naturlige tallene. En tolkning av dette resultatet er at ønsket om å studere fullt abstrakte modeller for PCF og beslektede kalkyler er for optimistisk, det er ikke mulig å kombinere dette ønsket med ønsket om å arbeide med beregnbare strukturer. Derfor stopper vi diskusjonen av Milners modell her.

6.5 Oppgaver til Kapittel 6

Oppgave 6.1 La t være en term av type $\sigma \to \tau$. Vis at $[\![t]\!] = [\![\lambda x^\sigma.tx]\!]$ hvor vi bruker Scotts modell for PCF.

Oppgave 6.2 Vis at hvis p er en kompakt i $D(\sigma)$, så er p PCF^+ definerbar.
Hint:
Bruk induksjon på nivået til typen.
Det kan lønne seg å vise simultant at relasjonen $p \sqsubseteq a$ er PCF^+ definerbar.
Minner om at kompakter p av type $\sigma \to \tau$ er definerbare fra par $(p_1, q_1), \ldots (p_n, q_n)$ av kompakter fra hhv. $D(\sigma)$ og $D(\tau)$. Det kan lønne seg å bruke subinduksjon på n.

Oppgave 6.3 Vis at relasjonen $\prec$ tilnærmet er en nedre semilattice i den forstand at hvis s_1 og s_2 er to lukkede PCF-termer av samme type, finnes det en term t slik at

1. $t \prec s_1 \wedge t \prec s_2$.
2. Hvis $t' \prec s_1 \wedge t' \prec t_2$, så vil $t' \prec t$.

Hint: La t simulere at s_1 og s_2 kjøres i serie.

Kapittel 7

Kleene-Kreisels funksjonaler

7.1 De hereditært totale funksjonalene

Det er en kjent sak at hvis vi har en opplisting av beregnbare funksjoner fra $\mathbb{N}$ til $\mathbb{N}$ så vil enten opplistingen selv ikke være beregnbar, eller så vil det finnes beregnbare funksjoner som ikke er fanget opp av opplistingen. Ved å se på

$$f(n) = f_n(n) + 1$$

ser vi at vi kan diagonalisere oss vekk fra enhver opptellbar mengde av funksjoner, og hvis opptellingen er beregnbar er dette diagonalelementet også beregnbart. Konsekvensen er at hvis vi ønsker å studere de beregnbare funksjonene, er det matematisk gunstig å ta med et studium av de partielle beregnbare funksjonene også.
Det er likevel slik at de totale funksjonene, de som er definerte med verdier i $\mathbb{N}$ hver gang argumentet hentes fra $\mathbb{N}$, er en mengde funksjoner av spesiell interesse. Videre vil det være slik at hvis F er av type $(\iota \to \iota) \to \iota$, så kan det knytte seg spesiell interesse til F hvis $F(f) \in \mathbb{N}$ hver gang f er total.
Det er derfor ikke underlig at følgende begrep har vært vist interesse i studiet av Scott-modellen:

Definisjon 7.1 La $T(\iota) = \mathbb{N}$ og $T(o) = \mathbb{B}$.
Hvis $\sigma = \tau \to \delta$ og $f \in D(\sigma)$ lar vi $f \in T(\sigma)$ hvis $f(a) \in T(\delta)$ hver gang $a \in T(\tau)$.
Vi kaller elementene i $T(\sigma)$ de *hereditært totale funksonalene* av type σ.

Det er lett å se at hvis $a \in T(\sigma)$ og $a \sqsubseteq_\sigma b$, så er $b \in T(\sigma)$, se oppgave 7.1.

Vi kan bruke T-hierarkiet til å gi en alternativ modell for Gödels T, se Oppgave 7.3

Dette hierarkiet vårt har en liten svakhet, det er ikke ekstensjonalt. Det betyr at det finnes forskjellige objekter f og g i $T(\sigma \to \tau)$ slik at $f(a) = g(a)$ for alle $a \in T(\sigma)$. Matematikerens måte å viske ut slike uønskede forskjeller på er å se på ekvivalensklasser av objekter i stedet for objektene selv. For å kunne gi en pen karakterisering av disse ekvivalensklassene, trenger vi et enkelt lemma som vi kunne vist på et meget tidligere stadium i utviklingen av domeneteorien, men som vi ikke har fått bruk for før nå:

Lemma 7.2 *La $(X, \sqsubseteq)$ være en begrenset komplett cpo og la $A \subseteq X$ være en ikketom mengde.*
Da finnes det en største nedre grense for A i X.

Bevis
La $B = \{x \in X \mid \forall a \in A(x \sqsubseteq a)\}$.
Det er lett å se at $\perp \in B$, at B er lukket nedover og siden X er begrenset komplett og $A \neq \emptyset$ ser vi at B er rettet. Da finnes den minste øvre grensen $\bigsqcup B$, som samtidig vil være den største nedre grensen til A.

Vi vil la $\sqcap$ betegne den største nedre grensen når den finnes.
La oss nå gå tilbake til de hereditært totale elementene:

Definisjon 7.3 Vi definerer relasjonen $\sim_\sigma$ på $T(\sigma)$ ved rekursjon på σ som følger:

1. $\sim_\iota$ og $\sim_o$ er identitetsrelasjonene på henholdsvis $\mathbb{N}$ og $\mathbb{B}$.
2. $f \sim_{\tau \to \delta} g \Leftrightarrow \forall a \in T(\tau) \forall b \in T(\tau)(a \sim_\tau b \Rightarrow f(a) \sim_\delta g(b))$.

Lemma 7.4 *For alle typer σ har vi*

a) Hvis a og b er i $T(\sigma)$ vil

$$a \sim_\sigma b \Leftrightarrow a \sqcap b \in T(\sigma).$$

b) $\sim_\sigma$ er en ekvivalensrelasjon.

Bevis
Vi bruker simultan induksjon på σ, og beviset er trivielt når σ er en basistype. Hvis $\sigma = \delta \to \tau$ og f og g er i $D(\sigma)$, bruker vi at

$$(f \sqcap g)(a) = f(a) \sqcap g(a)$$

for alle $a \in D(\delta)$. La $f \in T(\sigma)$, $g \in T(\sigma)$. Anta først at $f \sqcap g \in T(\sigma)$ og la $a \sim_\delta b$ være i $T(\delta)$. Ved induksjonsantagelsen vil $c = a \sqcap b \in T(\delta)$, så $(f \sqcap g)(c) \in T(\tau)$.
Vi har at $(f \sqcap g)(a \sqcap b) \sqsubseteq f(a) \sqcap g(b)$ (dette gjelder generelt for monotone funksjoner og ordninger som har største nedre grenser), og siden $T(\tau)$ er lukket oppover, vil $f(a) \sqcap g(b) \in T(\tau)$. Ved induksjonsantagelsen for τ er da $f(a) \sim_\tau g(b)$, så $f \sim_\sigma g$.
Anta nå at $f \sim_\sigma g$ La $a \in T(\delta)$. Ved punkt b) i induksjonsantagelsen vil $a \sim_\delta a$, så $f(a) \sim_\tau g(a)$. Ved induksjonsantagelsen får vi da at

$$(f \sqcap g)(a) = f(a) \sqcap g(a) \in T(\tau).$$

Dette viser at $f \sqcap g \in T(\sigma)$, og del a) er vist.
For å vise b) ser vi at relasjonen $f \sqcap g \in T(\sigma)$ opplagt er symmetrisk og refleksiv. Transitivitet følger av den andre karakteriseringen: La $f \sim_\sigma g \sim_\sigma h$.
La $a \sim_\delta b$.
Siden $a \sim_\delta a$ har vi

$$f(a) \sim_\tau g(a) \sim_\tau h(b)$$

så $f(a) \sim_\tau g(b)$ siden $\sim_\tau$ er transitiv.
Dette avslutter beviset for lemmaet.

Vi har beskrevet en familie av ekvivalensrelasjoner $\sim_\sigma$ som er kongruensrelasjoner med hensyn til applikasjon. Da ligger alt til rette for å se på det typede hierarkiet av mengder av ekvivalensklasser. Poenget med å gjøre dette er at resultatet blir et ekstensjonalt hierarki av funksjoner.

7.2 De tellbare/kontinuerlige funksjonalene

Definisjon 7.5 For hver type σ skal vi definere *Kleene-Kreisel-funksjonalene* $Ct(\sigma)$ av type σ, også kjent som de *tellbare* eller *kontinuerlige* funksjonalene av type σ, som følger. Simultant vil vi definere projeksjonsfunksjonene $\rho_\sigma : T(\sigma) \to Ct(\sigma)$ slik at

- for alle a og b i $T(\sigma)$ vil $a \sim_\sigma b \Leftrightarrow \rho_\sigma(a) = \rho_\sigma(b)$.
- Hvis $\sigma = \delta \to \tau$ har vi for alle $f \in T(\sigma)$ og $a \in T(\delta)$ at $\rho_\sigma(f)(\rho_\delta(a)) = \rho_\tau(f(a))$.

Vi lar $Ct(\iota) = \mathbb{N}$ og $\rho_\iota(n) = n$ for alle $n \in \mathbb{N}$, og vi lar $Ct(o) = \mathbb{B}$ og $\rho_o(a) = a$ for alle $a \in \mathbb{B}$.
Egentlig bestemmer spesifikasjonene våre nå $Ct(\sigma)$ og ρ_σ fullstendig ved rekursjon over σ, men for tydelighetens skyld gir vi definisjonen i mer detaljert form:
Anta at $\sigma = \delta \to \tau$, at $Ct(\delta)$,ρ_δ, $Ct(\tau)$ og ρ_τ er definerte slik at ρ_σ identifiserer nøyaktig $\sim_\sigma$-ekvivalente objekter og det tilsvarende holder for τ.
La $F : Ct(\delta) \to Ct(\tau)$. Vi lar $F \in Ct(\delta \to \tau)$ om det finnes $f \in T(\delta \to \tau)$ slik at vi for alle $a \in T(\delta)$ har at $F(\rho_\delta(a)) = \rho_\tau(f(a))$
Vi lar $\rho_\sigma(f) = F$ for alle aktuelle $f \in T(\sigma)$.
Alle $f \in T(\sigma)$ vil bestemme en $F \in Ct(\sigma)$, så ρ_σ er definert på hele $T(\sigma)$.
Spesifikasjonene er oppfylt pr. konstruksjon.

Vårt hierarki $\{Ct(\sigma)\}_{\sigma \text{ type}}$ er det samme som Kreisel definerte, og som han kalte *de kontinuerlige funksjonalene*, selv om hans definisjon var litt anderledes, han hadde ikke språket til domeneteorien. Kleenes definisjon av *de tellbare funksjonalene* var anderledes både i form og innhold. Kleenes tellbare funksjonaler var elementer i det fulle typehierarkiet. Poenget er at en funksjonal er tellbar hvis vi kan beskrive virkemåten av funksjonalen på et vilkårlig tellbart argument via tellbar mye informasjon. Hvis vi systematisk skreller bort alle ikketellbare argumenter og identifiserer Kleene-funksjonaler som er like for tellbare argumenter, vil vi ende opp med det samme hierarkiet som vi har definert.
Kreisel brukte $Ct(\sigma)$-hierarkiet til å gi en *konstruktiv tolkning* av alle utsagn i 2. ordens tallteori. Et viktig hjelpemiddel var teoremet som, ettersom det, med en litt anderledes formulering, ble uavhengig vist av Kleene, er kjent som *Kleene-Kreisels tetthetsteorem*. Det beviset vi skal gi er formet som et generelt lemma som vil tjene som induksjonsskrittet i beviset for tetthetsteoremet. Det var Ulrich Berger som analyserte det tradisjonelle beviset, og ga det en slik form at det ble enkelt å generalisere det. Derfor vil vi tilskrive det argumentet som nå følger Ulrich Berger. Vi sier at en mengde A i en ordning X er ubegrenset hvis det ikke finnes noen øvre grense for A i X.

Definisjon 7.6 La $(X, \sqsubseteq)$ være et Scott-domene, $T \subseteq X$ være ikketom slik at $a \in T \wedge a \sqsubseteq b \Rightarrow b \in T$.

a) Vi sier at T er *tett i X* hvis vi for alle kompakter $p \in X$ har en $a \in t$ slik at $p \sqsubseteq a$.

b) Vi sier at T er *ko-tett i* X hvis vi for alle ubegrensede endelige mengder $K = \{p_1, \ldots, p_k\}$ av kompakter i X har at det finnes en kontinuerlig funksjon $t : X \to \mathbb{B}_\perp$ slik at

   – $t(a) \in \mathbb{B}$ for $a \in T$.
   – $\{t(p) \mid p \in K\}$ er ubegrenset i $\mathbb{B}_\perp$, hvilket betyr at både $\mathbf{T}$ og $\mathbf{F}$ er med i mengden.

Lemma 7.7 *$T(\iota)$ er tett og ko-tett i $D(\iota)$ og $T(o)$ er tett og ko-tett i $D(o)$.*

Bevis
Vi viser dette for ι. Tetthet er trivielt. La K være en ubegrenset endelig mengde av kompakter. Siden K er ubegrenset må det finnes $n \neq m$ i K . La $t(\perp) = \perp$, $t(n) = \mathbf{T}$ og $t(i) = \mathbf{F}$ når $i \neq n$.

Teorem 7.8 (Ulrich Berger) *La X og Y være Scott-domener og anta at T er både tett og ko-tett i X og at S er både tett og ko-tett i Y.*
La $f \in X \to Y$ være i R hvis $f(a) \in S$ hver gang $a \in T$.
Da er R både tett og ko-tett i $X \to Y$.

Bevis
Vi viser først at R er tett:
La p være kompakt i $X \to Y$. Da finnes kompakter $q_1, \ldots, q_n$ i X og kompakter $r_1, \ldots, r_n$ i Y slik at for alle $a \in X$ vil

$$p(a) = \bigsqcup_Y \{r_i \mid q_i \sqsubseteq_X a\}.$$

La $K \subseteq \{1, \ldots, n\}$, la $K_X = \{q_i \mid i \in K\}$ og la $K_Y = \{r_i \mid i \in K\}$.
Hvis K_X er begrenset av a, vil K_Y være begrenset av $p(a)$, så, ved å snu på argumentet, vil K_Y ubegrenset $\Rightarrow K_X$ ubegrenset.
For hver $K \subseteq \{1, \ldots, n\}$ slik at K_Y er ubegrenset, la $t_K : X \to \mathbb{B}_\perp$ realisere ko-tetthetsegenskapen for K_X.
For hver $K \subseteq \{1, \ldots, n\}$ slik at Y_K er begrenset, la $b_K \in S$ være en utvidelse av $\bigsqcup_Y K_Y$.
Vi skal nå definere f_p og vise at f_p er konsistent med p. Den endelige realiseringen av tetthet vil da være $p \sqcup f_p$.
La $a \in X$. Vi deler konstruksjonen opp i to tilfeller:

1. Det finnes $K \subseteq \{1, \ldots, n\}$ slik at K_Y er ubegrenset, mens $t_K(a) = \perp$. Da lar vi $f_p(a) = \perp$.

2. Ellers. Vi lar $i \in I$ hvis $1 \leq i \leq n$ og for alle $K \subseteq \{1, \ldots, n\}$ slik at K_Y er ubegrenset og med $i \in K$ vil $t_K(q_i) \sqsubseteq_\sigma t_K(a)$.
Vi påstår at I_Y må være begrenset, og argumenterer kontrapositivt. Hvis I_Y er ubegrenset, vil vi ha at $t_I[I_X]$ på den ene siden er ubegrenset, mens på den andre siden er begrenset av $t_I(a)$. Dette er uforenlig.
Ettersom I_Y er begrenset, kan vi la $f_p(a) = b_I$

Det vil være tre egenskaper vi må vise før vi er ferdige:

i) f_p er kontinuerlig, det vil si at f_p er monoton og respekterer rettede grenser:
La $a \sqsubseteq_X b$. Monotonitet er bare en utfordring hvis vi er i tilfelle 2. når vi definerer både $f_p(a)$ og $f_p(b)$. La I_a og I_b være de to relevante mengdene. For alle $K \subseteq \{1, \ldots, n\}$ slik at Y_K er ubegrenset vil vi da ha at $t_K(a) = t_K(b)$, siden begge er i $\mathbb{B}$, og det betyr at vi må ha at $I_a = I_b$. Det følger at $f_p(a) = f_p(b)$, så monotoniteten holder.
Hvis $B \subseteq X$ er rettet, med $a = \bigsqcup_X B$ og vi er i tilfelle 2. når vi definerer $f_p(a)$, ser vi at det må finnes en $b \in B$ slik at $I_a = I_b$. Det følger at $f_p(a) = f_p(b)$ for en $b \in B$. Dette viser kontinuitet.

ii) $\{p, f_p\}$ er begrenset:
La $a \in X$. Vi må vise at $p(a) = \bigsqcup\{r_i \mid q_i \sqsubseteq a\}$ og $f_p(a)$ har en felles øvre begrensning. Hvis vi er i tilfelle 1. er det ikke noe å vise. Men hvis vi er i tilfelle 2. og $q_i \sqsubseteq a$, vil opplagt $i \in I$, og da er $r_i \sqsubseteq b_I$.

iii) $f_p \in R$: Vi ser at hvis $a \in T$ vil vi være i tilfelle 2. når vi angir verdien til $f_p(a)$. Da er $f_p(a) \in S$ pr. konstruksjon.

Vi viser så at R er ko-tett:
La $K = \{p_1, \ldots, p_n\}$ være en endelig mengde av kompakter i Z som er ubegrenset.
Hver p_i er den minste øvre grensen av en endelig mengde av kompakter på formen $f_{\langle q,r \rangle}$, og la $\{\langle q_j, r_j \rangle\}_{1 \leq j \leq m}$ være en opplisting av de aktuelle par $\langle q, r \rangle$.
Siden $p_1, \ldots, p_n$ ikke har noen øvre grense, vil det finnes en endelig mengde $J \subseteq \{1, \ldots, m\}$ slik at $\{q_j \mid j \in J\}$ er begrenset i X mens $\{r_j \mid j \in J\}$ ikke er begrenset i Y.
La $a \in T$ være en utvidelse av alle q_j for $j \in J$, og la $s : Y \to \mathbb{B}_\perp$ realisere ko-tetthetsegenskapen for $\{r_j \mid j \in J\}$.
For $f \in X \to Y$, la $t(f) = s(f(a))$.

Det er to ting som må bevises for å vise at t realiserer ko-tetthetsegenskapen for K:

1. $t(f) \in \mathbb{B}$ når $f \in R$:
   Hvis $f \in R$, vil $f(a) \in S$ siden $a \in T$. Siden s avbilder S på $\mathbb{B}$, vil $t(f) = s(f(a)) \in \mathbb{B}$

2. $\{t(p) \mid p \in K\}$ er ubegrenset i $\mathbb{B}_\perp$:
   Hvis $j \in J$, finnes $p \in K$ slik at $r_j \sqsubseteq p(a)$ og at derfor $s(r_j) \sqsubseteq t(p)$. Siden $\{s(r_j) \mid j \in J\}$ er ubegrenset i $\mathbb{B}_\perp$ vil $\{t(p) \mid p \in K\}$ også være det.

Dette avslutter beviset.
Legg merke til at vi bare brukte tetthet for Y og ko-tetthet for X da vi viste tetthet for $X \to Y$, mens vi bare brukte tetthet for X og ko-tetthet for Y da vi viste ko-tetthet.

Korollar 7.9 *For alle typer σ er $T(\sigma)$ tett og ko-tett i $D(\sigma)$.*

Bemerkning 7.10 Dette beviset har en svakhet, og det er at vitnene på tetthetsegenskapen og ko-tetthetsegenskapen i Scott-hierarkiet ikke alltid er PCF definerbare. Dette kan vi se ved å se på $\bigvee$, som både er kompakt og i $T(o, o \to o)$. Da vi generelt viste at hver kompakt p kan utvides til noe totalt, konstruerte vi en funksjon f_p som var forenlig med p, men ikke nødvendigvis utvidet p. For å få en utvidelse må vi slå f_p og p sammen. Dette illustreres ved å se på $\bigvee$, se Oppgave 7.2.

Dette eksemplet viser også at det finnes kompakter som ikke kan utvides til noe PCF-definerbart objekt.

En viktig konsekvens av Korollar 7.9 er

Korollar 7.11 *La σ være en type, og la a og b være i $T(\sigma)$. Da er følgende ekvivalente:*

1. $a \sim_\sigma b$

2. *a og b er* konsistente, *det vil si at $\{a, b\}$ er begrenset.*

Bevis
Vi viser dette ved induksjon på σ. For basistyper er dette trivielt, så anta at $\sigma = \delta \to \tau$ og at korollaret holder for δ og τ.
Hvis $\{a, b\}$ ikke er begrenset, må det finnes kompakt $p \in D(\delta)$ slik at $\{a(p), b(p)\}$ ikke er begrenset. La $x \in T(\delta)$ utvide p. Da er $\{a(x), b(x)\} \subseteq T(\tau)$ ikke begrenset, og i følge induksjonsantagelsen er ikke $a(x)$ og $b(x)$ ekvivalente. Da er ikke a og b ekvivalente.
Anta omvendt at $\{a, b\}$ er begrenset av c. La $x \sim_\delta y$ og la $z = x \sqcup y$. Da vil $c(z)$ være en øvre grense for $a(x)$ og $b(y)$, og vi er fremme.

Vi skal avslutte dette avsnittet med å gi to anvendelser av tetthet og kotetthet. Den første anvendelsen er Kreisels hovedmotivasjon for å bevise tetthetsteoremet, ved å bruke funksjonaler kan man skrive alle utsagn om til $\exists\forall$-form. Vi nøyer oss med å se på induksjonskrittet.

Definisjon 7.12 La $Ct(\sigma)$ være Kleene-Kreisel-objektene av type σ for hver type σ. La $\sigma_1, \ldots, \sigma_n$ være typer. Et *predikat* på $Ct(\sigma_1) \times \cdots \times Ct(\sigma_n)$ er en mengde $P \subseteq Ct(\sigma_1) \times \cdots \times Ct(\sigma_n)$ slik at det finnes en $p \in Ct(\sigma_1, \ldots, \sigma_n \to o)$ hvor $\vec{a} \in P \Leftrightarrow p(\vec{a}) = \mathbf{T}$.

Lemma 7.13 *La P være et predikat på $Ct(\sigma) \to Ct(\tau)$. Da er følgende ekvivalente:*

1. $\forall x \in Ct(\sigma) \exists y \in Ct(\tau) P(x, y)$
2. $\exists f \in Ct(\sigma \to \tau) \forall x \in Ct(\sigma) P(x, f(x))$

Bevis
Det er opplagt at 2. $\Rightarrow$ 1.
Anta 1. La $x \in Ct(\sigma)$ og la $\hat{x} \in T(\sigma)$ være slik at $x = \rho_\sigma(\hat{x})$ La $\{\hat{y}_n\}_{n\in\mathbb{N}}$ være en opptelling av elementer av $T(\tau)$ slik at alle kompakter i $D(\tau)$ har en utvidelse i form av en $\hat{y}_n$. La $p \in Ct(\sigma, \tau \to o)$ definere P og la $\hat{p} \in T(\sigma, \tau \to o)$ være en representant for p.
Da har vi at

$$\forall \hat{x} \in T(\sigma) \forall n \in \mathbb{N} (\hat{p}(\hat{x}, \hat{y}_n) \in \mathbb{B})$$

og

$$\forall \hat{x} \in T(\sigma) \exists n \in \mathbb{N} (p(\hat{x}, \hat{y}_n) = \mathbf{T}).$$

Det siste ser vi ved å velge en $\hat{y} \in T(\tau)$ slik at $\hat{p}(\hat{x}, \hat{y}) = \mathbf{T}$, så plukke ut en kompakt $q \sqsubseteq \hat{y}$ slik at $\hat{p}(\hat{x}, q) = \mathbf{T}$ og til sist en n slik at $q \sqsubseteq \hat{y}_n$. Da kan vi definere

$$f(x) = \rho_\tau(\hat{y}_{\mu n(p(x,\rho_\tau(\hat{y}_n))=\mathbf{T})}).$$

Dette avslutter beviset.

Det er en generell egenskaper for *cpo*-er at ethvert element har en maksimal utvidelse. I de mest generelle tilfellene trenger vi utvalgsaksiomet for å vise dette, men for Scott-domener klarer man seg med det som heter *svakt Königs lemma*. Eksistensen av den maksimale utvidelsen er gitt som oppgave 7.4. Det er også slik at i Scott-hierarkiet finnes det totale objekter som ikke er maksimale. Derfor er følgende begrep interessant:

Definisjon 7.14 La $(X, \sqsubseteq)$ være en *cpo* og la $x \in X$.
x er *premaksimal* hvis x har en entydig maksimal utvidelse.
For Scott-domener sier vi også at x er *Berger-total* når x er premaksimal.

Lemma 7.15 *La $(X, \sqsubseteq)$ være et Scott-domene.*

a) La $x \in X$. Da er følgende ekvivalente:

- *i) x er premaksimal.*
- *ii) Hvis p og q er kompakter slik at $\{p, x\}$ og $\{q, x\}$ begge er begrensede, vil $\{p, q\}$ være begrenset.*

b) La $T \subset X$ være lukket oppover og være ko-tett. Da er alle $x \in T$ premaksimale.

Bevis
Bevis for a):
Først la x være premaksimal, med y som den maksimale utvidelsen. Hvis p er kompakt slik at $\{p, x\}$ er begrenset, må $p \sqsubseteq y$. Det viser at mengden av slike kompakter er begrenset. Dette viser at i) $\Rightarrow$ ii)
Anta omvendt at x har to forskjellige maksimale utvidelser y og z, det vil si at i) ikke holder. Da er ikke $\{y, z\}$ begrenset i X, og vi vil finne kompakter $p \sqsubseteq y$ og $q \sqsubseteq z$ slik at $\{p, q\}$ ikke er begrenset. (Vi overlater til leseren å vise at det motsatte vil lede til en selvmotsigelse.) Men valget av p og q er slik at $\{p, x\}$ er begrenset av y og $\{q, x\}$ er begrenset av z, så da holder heller ikke ii). Dette viser omvendingen.

Bevis for b):
La T være ko-tett og la $x \in T$. La p og q være kompakter slik at $\{x, p\}$ og

$\{x, q\}$ begge er begrensede. Hvis $\{p, q\}$ ikke er begrenset finnes $t : X \to \mathbb{B}_\perp$ slik at $t(x) \in \mathbb{B}$ og slik at $t(p)$ og $t(q)$ er uforenlige. Da må begge verdiene være i $\mathbb{B}$ og forskjellige. Men det gir oss en motsigelse siden vi må ha

$$t(p) = t(p \sqcup x) = t(x) = t(q \sqcup x) = t(q)$$

ettersom alle verdier er i $\mathbb{B}$.

Hvis typen σ er av et rimelig høyt nivå, vil elementene av $Ct(\sigma)$ være objekter av rimelig høy kompleksitet i mengdeteoretisk forstand. Hvis vi representerer objektet som et ideal av kompakter, får vi i det minste en tellbar mengde av kompakter, men selv når vi er i den gunstige situasjonen at vi har en lukket term t av type σ slik at $[\![t]\!] \in T(\sigma)$ så trenger ikke $[\![t]\!]$ å svare til et beregnbart ideal av kompakter, i værste fall er idealet bare semiberegnbart.
Vi skal se at ko-tetthetsegenskapen for en mengde $T \subseteq X$ innebærer at alle $x \in T$ kan representeres som følger h_x av booleske verdier, slik at konsistente elementer i T får samme representasjon:

Definisjon 7.16 La $(X, \sqsubseteq)$ være et Scott-domene og la $T \subseteq X$ være ko-tett. La $\{t_n\}_{n\in\mathbb{N}}$ være en opptelling av ett vitne på ko-tetthetsegenskapen for hver endelige mengde K av kompakter som er ubegrenset.
La *bildet av* $x \in T$ være definert ved

$$h_x(n) = t_n(x).$$

Teorem 7.17 *La X, T og funksjonene h_x være som over. Hvis $x \in T$ vil $h_x(n)$ være definert for alle n, og hvis x og y er i T vil x og y være ekvivalente hvis og bare hvis $h_x = h_y$.*

Bevis
Den første påstanden er triviell.
Hvis x og y er ekvivalente, vil vi nødvendigvis ha

$$h_x = h_{x \sqcap y} = h_y.$$

Omvendt, hvis $h_x = h_y$, vil x og y være konsistente, hvilket betyr at de er ekvivalente, se Korollar 7.11.

7.3 Oppgaver til Kapittel 7

Oppgave 7.1 Vis at for alle typer σ vil $T(\sigma)$ være lukket oppover. Det kan lønne seg å bruke induksjon på oppbyggingen av σ

Oppgave 7.2 Vis at $f_\bigvee(b, c)$ er udefinert hvis $b = \perp$ eller $c = \perp$, hvor $f_\bigvee$ er som i beviset for tetthetsteoremet i forhold til kompakten $\bigvee$.

Oppgave 7.3 a) Vis hvordan vi på en naturlig måte kan tolke enhver term t i Gödels T av type σ og med fri variable av typer $\tau_1, \ldots, \tau_n$ som et element i

$$T(\tau_1, \ldots, \tau_n \to \sigma).$$

b) Vis at hvis t og s er to lukkede termer i Gödels T av samme type, slik at de er observabelt like, så vil de ha den samme tolkningen over de hereditært totale funksjonalene opp til ekvivalens.
Hint: Bruk at det finnes en tett mengde av T-definerbare objekter for hver type.

Oppgave 7.4 Vis at hvis $(X \sqsubseteq)$ er et Scott-domene og $x \in X$, så har x en maksimal utvidelse i X.
Hint: Ta utgangspunkt i en opptelling av kompaktene i X og utvid x med en og en kompakt slik at vi hele tiden holder oss i X.

Kapittel 8

En modell for System F

I dette kapitlet skal vi vise hvordan vi kan bruke Scott-domener til å gi en semantikk for System F.
Den tradisjonelle tilnærmingen til en denotasjonell semantikk for System F går via det som heter koherensrom. For å gi en anstendig behandling av koherensrom og modellen for System F basert på koherensrom, må vi innføre noen begreper fra kategoriteori, noe forfatteren etter litt ettertanke har funnet ut at han ikke vil gjøre.

8.1 Mettede Scott-domener

I dette avsnittet skal vi se på embeddingsbegrepet for Scott-domener og vise at det finnes et universalt Scott-domene $(S, \sqsubseteq_S)$ i den forstand at alle andre Scott-domener kan embeddes i S.

Lemma 8.1 *La $(X, \sqsubseteq_X)$ og $(Y, \sqsubseteq_Y)$ være to algebraiske domener og la X_0 og Y_0 være kompaktene i henholdsvis X og Y.*

a) La (η, π) være et ep-par fra X til Y. Da vil $\eta(a) \in Y_0$ når $a \in X_0$ og hvis $a, b \in X_0$ er $\{a, b\}$ begrenset i X hvis og bare hvis $\{\eta(a), \eta(b)\}$ er begrenset i Y.
Hvis i tillegg $\{a, b\}$ har en minste øvre grense c i X, vil $\eta(c)$ være den minste øvre grensen til $\{\eta(a), \eta(b)\}$ i Y.

b) Anta i tillegg at X og Y er Scott-domener. La $\eta_0 : X_0 \to Y_0$ være monoton og slik at om $A \subseteq X_0$ er endelig og bildet $\eta[A]$ er begrenset i Y, så er A begrenset i X. Da kan η_0 utvides til et ep-par (η, π) ved

- $\eta(x) = \bigsqcup_Y \{\eta_0(a) \mid a \in X_0 \wedge a \sqsubseteq_X x\}$.
- $\pi(y) = \bigsqcup_X \{a \in X_0 \mid \eta_0(a) \sqsubseteq_Y y\}$.

Bevis
a). La $a \in X_0$. La $B \subset Y$ være rettet slik at $\eta(a) \sqsubseteq_Y \bigsqcup_Y B$, og la $A = \{\pi(y) \mid y \in B\}$. Siden π er kontinuerlig vil A være rettet, og $\pi(\bigsqcup_Y B) = \bigsqcup_X A$.
Da er $a = \pi(\eta(a)) \sqsubseteq_X \bigsqcup_X A$, så det finnes $x \in A$ slik at $a \sqsubseteq_X x$. La $y \in B$ være slik at $x = \pi(y)$. Da har vi

$$\eta(a) \sqsubseteq_Y \eta(x) = \eta(\pi(y)) \sqsubseteq_Y y.$$

Dette viser at $\eta(a) \in Y_0$.
Alle monotone funksjoner vil avbilde en begrenset mengde på en begrenset mengde. Hvis $\{\eta(a), \eta(b)\}$ er begrenset av y, vil $\{a, b\}$ være begrenset av $\pi(y)$, så denne delen av påstanden holder også.
Anta nå at $c = a \sqcup_X b$, og videre at $\eta(a) \sqsubseteq_Y d$ og $\eta(b) \sqsubseteq_Y d$. Da vil

$$a = \pi(\eta(a)) \sqsubseteq_X \pi(d)$$

og

$$b = \pi(\eta(b)) \sqsubseteq_X \pi(d),$$

så $c \sqsubseteq_X \pi(d)$.
Da vil

$$\eta(c) \sqsubseteq_Y \eta(\pi(d)) \sqsubseteq_Y d.$$

Dette viser at $\eta(c) = \eta(a) \sqcup_Y \eta(b)$.
b). Dette punktet er så enkelt at leseren bør gjennomføre det selv, Se Oppgave 8.1.

Dette lemmaet viser at ep-par mellom Scott-domener kan karakteriseres som embeddinger på kompakt-nivå som er strukturbevarende i den forstand at de bevarer både "begrenset"- og "ubegrenset"-relasjonene. De vil selvfølgelig da også bevare $\sqcup$ mellom par av kompakter.
Vi vil bruke ordet *embedding* om kompaktdelen η_0 av et ep-par (η, π) slik at $\eta(\bot_X) = \bot_Y$.
Fra nu av vil vi betrakte $\bot$ som en del av strukturen, og $\bot$ skal bevares av alle ep-par.

Definisjon 8.2 La $(X, \sqsubseteq)$ være et Scott-domene med kompakter X_0. Et *subdomene* er et Scott-domene $(Y, \sqcup_Y)$ slik at $(Y_0, \sqsubseteq_Y, \sqcup_Y, \perp)$ er en delstruktur av $(X_0, \sqsubseteq_X, \sqcup_X, \perp)$ også med hensyn til "begrensethet" og "ubegrensethet" av par.

Lemma 8.3 *La $(X \sqsubseteq)$ være et Scott-domene og la $A \subseteq X_0$ være endelig. Da finnes det et endelig subdomene Y av X slik at $A \subseteq Y$.*

Bevis
Vi lukker A under $\sqcup_X$ og får en endelig mengde Y av kompakter. Denne mengden vil i seg selv være et domene, ettersom det ikke er behov for kompletteringer av endelige ordninger, de er komplette i seg selv.

Vi kan observere at alle endelige Scott-domener vil bestå av bare kompakter, og at idealkompletteringen av en endelig og begrenset komplett partiell ordning vil være isomorf med ordningen selv. Teorien for endelige Scott-domener er altså identisk med teorien for endelige begrenset komplette partielle ordninger. Derfor vil vi se på slike i fortsetningen.

Definisjon 8.4 La $(S, \sqsubseteq_S)$ være et Scott-domene med kompakter S_0.
S er *mettet* hvis vi for alle endelige Scott-domener $(X, \sqsubseteq_X)$ og embeddinger $\eta_X : X \to S$ og alle endelige Scott-domener $(Y, \sqsubseteq_Y)$ med en embedding $\phi : X \to Y$ kan finne en embedding $\eta_Y : Y \to S$ slik at $\eta_X = \eta_Y \circ \phi$.

Lemma 8.5 *La S være et mettet Scott-domene.*
Ethvert Scott-domene X kan embeddes i S.

Bevis
Vi skriver X som grensen av en voksende følge av endelige Scott-domener:

$$X_0 = \{X_n \mid n \geq 1\}$$

hvor $X_1 = \{\perp_X\}$ og $\phi_n; X_n \to X_{n+1}$ er inklusjonsembeddingen. La $\eta_1(\perp_X) = \perp_S$. Siden S er mettet kan vi rekursivt finne embeddinger $\eta_{n+1} : X_{n+1} \to S$ slik at $\eta_n = \eta_{n+1} \circ \phi_n$. Da er unionen av alle η_n en embedding av X inn i S.

Bemerkning 8.6 Ved å bruke en standard metode, sikksakk-metoden fra modellteori, kan vi vise at to mettede Scott-domener vil være isomorfe. Dette får vi ikke bruk for, så vi overlater det til leseren som Oppgave 8.2.

Vi skal snart starte på beviset for at det finnes et mettet Scott-domene. Først skal vi imidlertid vise et generelt teorem om hvordan man konstruerer et begrenset komplett algebraisk domene fra et vilkårlig algebraisk domene.

Definisjon 8.7 La $(X, \sqsubseteq)$ være et algebraisk domene med kompakter X_0. La $C(X_0)$ bestå av alle endelige, begrensede delmengder C av X_0.
Vi preordner $C(X_0)$ ved

$$C \prec D \Leftrightarrow \text{alle øvre grenser for } D \text{ i } X_0 \text{ er også øvre grenser for } C \text{ i } X_0.$$

Da vil

$$C \equiv D \Leftrightarrow C \prec D \wedge D \prec C$$

være en ekvivalensrelasjon.
Vi lar $(C(X))_0$ være mengden av ekvivalensklasser $[C]$ med den avledede ordningen, og vi lar $C(X)$ være idealkompletteringen av denne ordningen.

Vi kan merke oss at dersom X_0 er tellbar vil $(C(X))_0$ være tellbar og dersom X er endelig vil $C(X)$ være endelig. Poenget er at denne konstruksjonen representerer en tillukning av et algebraisk domene til et begrenset komplett algebraisk domene. Vi samler denne påstanden opp i det neste teoremet, hvor vi ikke trenger punkt c) i fortsettelsen.

Teorem 8.8 *Vi bruker notasjonen fra Definisjon 8.7.*

a) $C(X)$ *er begrenset komplett.*

b) Det finnes et ep-par (η, π) *fra* X *til* $C(X)$.

c) Hvis Y *er begrenset komplett og det finnes et ep-par* (η_1, π_1) *fra* X *til* Y*, finnes det et ep-par* (η_2, π_2) *fra* $C(X)$ *til* Y *slik at*

$$\eta_1 = \eta_2 \circ \eta \wedge \pi_1 = \pi \circ \pi_2.$$

Bevis
a): La $C \prec E$ og $D \prec E$. Da vil $C \cup D \prec E$, og spesielt vil $C \cup D$ være begrenset siden E er det. Vi har at $[C \cup D]$ vil være den minste øvre grensen for $[C]$ og $[D]$.
b): La $\eta(x) = \{[C] \mid \forall a \in C(a \sqsubseteq_X x)\}$.
For å vise at definisjonen er uavhengig av $C \in [C]$ og at $\eta(x)$ er lukket nedover, er det nok å innse

$$D \prec C \wedge \forall a \in C(a \sqsubseteq_X x) \Rightarrow \forall d \in D(d \sqsubseteq x)$$

noe som følger av at C vil være begrenset under x. At $\eta(x)$ er lukket under $\sqcup$ følger av vår karakterisering av $\sqcup$ i $C(X)$.
La $\pi(y) = \bigsqcup_X \{a \in X_0 \mid [\{a\}] \in Y\}$.
Det er lett å se at (η, π) utgjør et ep-par.
c): La (η_1, π_1) være et ep-par fra X til Y hvor vi antar at Y er begrenset komplett.
Vi nøyer oss med å definere η_2 på $(C(X))_0$ og resten av η_2 og hele π_2 vil bli avledet.
La $\eta_2([C]) = \bigsqcup_Y \{\eta_1(c) \mid c \in C\}$.
Igjen bør vi påse at definisjonen er uavhengig av $C \in [C]$, at η_2 er monoton, at η_2 avbilder minste øvre grenser på minste øvre grenser, og at η_2 avbilder ubegrensede endelige mengder på ubegrensede mengder.
Alt dette er enkelt, og, ettersom vi ikke trenger resultatet videre, overlates beviset til den interesserte leser. Det er også lett å se at sammensetningspåstandene holder.

Lemma 8.9 [Amalgamering]
La X, Y og Z være Scott-domener og la η_Y og η_Z være embeddinger av X inn i Y og Z respektive.
Da finnes det et Scott-domene V og embeddinger ϕ_Y og ϕ_Z fra henholdsvis Y og Z inn i V slik at

$$\phi_Y \circ \eta_Y = \phi_X \circ \eta_X.$$

Bevis
Siden vi kan se på isomorfe kopier, kan vi uten tap av generalitet anta at $X \subseteq Y$, at $X \subseteq Z$, at $Y \setminus X$ og $Z \setminus X$ er disjunkte og at η_X og η_Y er inklusjonsavbildningene.
Vi vil la $U = Y \cup Z$ og vi vil la $\sqsubseteq_U$ være den transitive tillukningen av $\sqsubseteq_Y \cup \sqsubseteq_Z$.
I den transitive tillukningen kan vi hoppe mellom $\sqsubseteq_Y$ og $\sqsubseteq_Z$. Poenget er at hvis vi er i et punkt hvor vi skifter mellom disse to ordningene, må det punktet ligge i X.

Påstand 1: Hvis y og y' er i Y og $y \sqsubseteq_U y'$ vil $y \sqsubseteq_Y y'$.

Bevis: La $y = u_1 \sqsubseteq_{Y/Z} u_2 \sqsubseteq_{Y/Z} \cdots \sqsubseteq_{Y/Z} u_n = y'$. Vi bruker induksjon på antall i slik at $u_i \in Z \setminus X$. Notasjonen $\sqsubseteq_{Y/Z}$ antyder at vi ser på en voksende følge i $\sqsubseteq_Y \cup \sqsubseteq_Z$.
Hvis ingen u_i er i $Z \setminus X$ er alle sammen i Y, og vi kan bruke transitivitet av $\sqsubseteq_Y$. Hvis det fines $u_i \in Z \setminus X$ gir observasjonen vår over at det finnes en

største $j < i$ og en minste $k > i$ slik at $u_j \in X$ og $u_k \in X$. Da er hele følgen fra u_j opp til u_k en voksende følge i Z, så $u_j \sqsubseteq_Z u_j$, og siden begge elementene er i X vil vi ha $u_j \sqsubseteq_X u_k$. Det betyr at om vi fjerner alle ledd i mellom, får vi fortsatt et vitne på at $y \sqsubseteq_U y'$, men slik at induksjonsantagelsen da gir oss at vi også må ha at $y \sqsubseteq_Y y'$.

Påstand 2: Hvis y_1 og y_2 er begrenset i U, vil y_1 og y_2 være begrenset i Y, og $y_1 \sqcup_U y_2 = y_1 \sqcup_Y y_2$.
Bevis:
La $y_1 \sqsubseteq_U u$ og $y_2 \sqsubseteq_U u$.
Hvis $u \in Y$ vil u i følge Påstand 1 være en Y-øvre grense for y_1 og y_2 og vi har at $y_1 \cup_Y y_2 \sqsubseteq_U u$.
Hvis $u \in Z \setminus X$ ser vi fra beviset for Påstand 1 at det må finnes x_1 og x_2 i X slik at $y_1 \sqsubseteq_Y x_1 \sqsubseteq_Z u$ og $y_2 \sqsubseteq_Y x_2 \sqsubseteq_Z u$.
Siden $\{x_1, x_2\}$ er begrenset i Z har denne mengden en minste øvre grense x i X lik den minste øvre grensen i Z. Men da er $\{y_1, y_2\}$ begrenset av x i Y, og vi har

$$y_1 \sqcup_Y y_2 \sqsubseteq_U x \sqsubseteq_U .$$

Dette viser påstanden.

De symmetriske påstandene for Z holder selvfølgelig.
Det er mulig at det finnes $y \in Y$ og $z \in Z$ slik at $\{y, z\}$ er begrenset i U uten at det finnes noen minste øvre grense i U. Denne muligheten vil være der selv om X, Y og Z er endelige domener. Derfor lar vi $V = C(U)$. Da vil V være et Scott-domene, Y og Z vil være isomorfe til subdomener av V slik at X er isomorft til snittet av Y og Z.
Dette avslutter beviset.

Teorem 8.10 *Det finnes et mettet Scott-domene.*

Bevis
Vi vil beskrive kompaktene S_0 i et Scott-domene S som unionen av en voksende familie $\{S[n]\}_{n\in\mathbb{N}}$ av endelige Scott-domener slik at inklusjonsavbildningen fra $S[n]$ til $S[n+1]$ er en embedding. Vi skal bruke at i amalgameringslemmaet kan vi uten tap av generalitet kreve at om X er et subdomene av Y kan vi oppnå at Y er et subdomene av V. Vi kan alltid finne en isomorf V hvor dette er tilfellet.
Vi vil la $S[0] = \{\perp_S\}$.
Vi vil bruke koden $\langle m, k\rangle = 2^m \cdot (2k+1)$ og vi gjør bruk av ulikheten $n < \langle n, k\rangle$ for alle n og k, samt at $\langle \cdot, \cdot\rangle$ er en bijeksjon mellom $\mathbb{N}^2$ og $\mathbb{N}^+ = \{1, 2, \ldots\}$.

Når $S[n]$ er konstruert, lar vi $\{X_{\langle n,k\rangle}, \eta_{\langle n,k\rangle}, Z_{\langle n,k\rangle}\}_{k\in\mathbb{N}}$ være en opptelling av tripler (X, η, Z) slik at X er et subdomene av $S[n]$, Z er et endelig domene og $\eta : X \to Z$ er en embedding, og slik at for alle subdomener X av $S[n]$ har vi dekket alle mulige η og Z opp til isomorfi.
Da konstruerer vi $S[n+1]$ fra $S[n]$ ved bruk av amalgameringslemmaet hvor vi lar $n+1 = \langle m, k\rangle$, setter inn X_{n+1} for X, $S[n]$ for Y, Z_{n+1} for Z, inklusjonsavbildningen for η_X, η_{n+1} for η_Y og lar $S[n+1] = V$ med tilleggsforståelsen at vi kan la Y være et subdomene av V.
Hvis vi nå lar $S_0 = \bigcup_{n\in\mathbb{N}} S[n]$ med nedarvet ordning, ser vi at det tilhørende Scott-domenet S blir mettet.

8.2 Tolkning av System F

8.2.1 Domenet av alle domener

Fra nå av skal vi la S være et mettet Scott-domene, og vi lar det ligge fast i resten av kapitlet. Hvis E er et subdomene av S og D er et subdomene av E vil alle kompaktene i E være kompakter i D.

Lemma 8.11 *La D og E være subdomener av S slik at alle kompaktene i D ligger i E. Da er D et subdomene av E.*

Beviset er enkelt, og er gitt som Oppgave 8.3.

Definisjon 8.12 La DOM være mengden av subdomener av S ordnet ved subdomeneordningen $\sqsubseteq_{DOM}$.

Teorem 8.13 *$(DOM, \sqsubseteq_{DOM})$ er et Scott-domene.*

Bevis
Når vi i dette beviset snakker om unioner av subdomener av S mener vi at vi tar unioner av mengder av kompakter og beholder ordningen $\sqsubseteq_S$.
Siden en rettet union av subdomener av S selv vil være et subdomene av S, er $(DOM, \sqsubseteq_{DOM})$ en *cpo*.
Kompaktene i denne *cpo*-en vil være de endelige subdomenene av S.
Vi har at ethvert subdomene av S er unionen av sine endelige subdomener. Dette viser at $(DOM, \sqsubseteq_{DOM})$ er et algebraisk domene.
Hvis D og E er to endelige subdomener av S finner vi det minste subdomenet C av S som omfatter både D og E ved å lukke kompaktene i $D \cup E$ under

$\sqcup_S$. Dette viser at DOM er begrenset komplett.
Siden S har tellbart mange kompakter, vil S ha tellbart mange endelige subdomener.
Tilsammen viser disse punktene at DOM er et Scott-domene.

Når vi skal tolke System F vil vi tolke typer som elementer i DOM og objekter som elementer i tolkningene av typene. På den annen side må vi ofte bevege oss opp i S for å gjennomføre konstruksjonene på en meningsfylt måte. Det gir oss et behov for en hendig notasjon for disse bevegelsene:

Definisjon 8.14 La $D \in DOM$ og la (η, π) være ep-paret fra D til S bestemt av inklusjonsavbildningen på kompaktene.
For $d \in D$ lar vi d^+ betegne $\eta(d)$.
For $d \in S$ lar vi d^D betegne $\pi(d)$.
Merk at d^+ faktisk kan bli definert for alle konsistente mengder av kompakter i S som den minste øvre grensen i S av d.

8.2.2 Funksjonsrom

Vi har tidligere sett at dersom D og E er Scott-domener, så er mengden $D \to E$ av kontinuerlige funksjoner fra D til E ordnet ved punktvis ordning også et Scott-domene. I beviset inngår det en karakterisering av kompaktene i $D \to E$ via mengder $\{\langle p_1, q_1\rangle, \ldots, \langle p_n, q_n\rangle\}$ av par av kompakter p_i i D og q_i i E. Hvis nå D utvides til et domene D' og E utvides til et domene E' ser vi at spørsmålet om en slik mengde av par av kompakter bestemmer en kompakt i funksjonsrommet eller ikke er absolutt for (D, E) og (D', E'). Den interne ordningen mellom kompaktene bestemt av to slike mengder av par vil også være uavhengig av om vi ser på den i lys av D og E eller i lys av D' og E'. Tilsist observerer vi at en kompakt i $D \to E$ vil kunne beskrives fra kompakter i endelige tilnærminger D_1 og E_1 til henholdsvis D og E. Alt dette bruker vi til å bevise

Lemma 8.15 *Det finnes en kontinuerlig funksjon $FUNC : DOM^2 \to DOM$ slik at hvis D og E er to subdomener av S, så er $FUNC(D, E)$ isomorft til funksjonsrommet $D \to E$.*

Bevis
La $F \in DOM$ være isomorft til $S \to S$ via embeddingen $\phi : (S \to S) \to S$. Det betyr at F er bildet til ϕ. La D og E være subdomener av S og la $D \to^* E$ være subdomenet av $S \to S$ isomorft til $D \to E$ som diskutert

over.
La $FUNC(D,E)$ være ϕ-bildet av $D \to^* E$. $FUNC(D,E)$ vil da være et subdomene av F.

Vi kommer til å referere til funksjonen ϕ når vi gir tolkningen av termer i System F.
Vi tar sjansen på å overbelaste notasjonen vi brukte for ep-paret mellom et subdomene $D \in DOM$ av S og S selv.
Hvis $f \in D \to E$ lar vi $f^+ \in D \to^* E$ være definert ved

$$f^+(d) = (f(d^D))^+$$

og hvis $g \in S \to S$, vil vi la $g^{D \to E} \in D \to E$ være definert ved

$$g^{D \to E}(d) = (g(d^+))^E.$$

Dette er standard løfting av ep-parene fra D og E til S til et ep-par fra $D \to E$ til $S \to S$.

8.2.3 Operatorer og uniforme objekter

Definisjon 8.16 a) En *domeneoperator*, eller i denne sammenhengen, bare *operator*, er en kontinuerlig funksjon $\Phi : DOM \to DOM$.

b) Hvis $\Phi : DOM \to DOM$ er en operator, vil et *uniformt objekt* for Φ være en funksjon ϕ slik at

i) ϕ er definert på DOM og for alle $D \in DOM$ er $\phi(D) \in \Phi(D)$.

ii) Det finnes en kontinuerlig $\hat{\phi} : DOM \to S$ slik at

$$\hat{\phi}(D) = \bigsqcup_S \{c \in (\Phi(D))_0 | \ c \sqsubseteq_{\Phi(D)} \phi(D)\}.$$

Vi kaller mengden av uniforme objekter for $\prod \Phi$

c) Hvis ψ og ϕ er uniforme objekter for Φ, lar vi $\psi \sqsubseteq_{\prod \Phi} \phi$ hvis $\psi(D) \sqsubseteq_{\Phi(D)} \phi(D)$ for alle $D \in DOM$.

Det vi ønsker å uttrykke med denne definisjonen er at ϕ er en funksjon som på en kontinuerlig måte plukker ut et element i $\Phi(D)$ for hver $D \in DOM$.

Lemma 8.17 *La $\Phi : DOM \to DOM$ være en operator. Da er $\prod \Phi$ et Scott-domene.*

Bevis
Vi vil etterhvert få at $\prod \Phi$ er et subdomene av $DOM \to S$. Vi kan oppfatte et uniformt objekt ϕ som en kontinuerlig funksjon (med hensyn til inklusjon) ϕ_0 som til $D \in DOM$ gir oss en mengde kompakter i $\Phi(D)$. Vi har da at $\phi_0(D) \subseteq \phi_0(S)$ ettersom S er det maksimale elementet i DOM. Det er opplagt at mengden av slike ϕ_0 utgjør en *cpo*, og da blir $\prod \Phi$ en *cpo*.
En kompakt ϕ vil være et uniformt objekt slik at $\phi(S)$ er en kompakt i $\Phi(S)$, hvor det finnes endelig mange muligheter $c_1, \ldots, c_n$ i S_0 knyttet til endelig mange endelige subdomener $D_1, \ldots, D_n$ av S slik at $c_i \in \Phi((D_i)_0)$ for alle i og slik at $\Phi(D) = \sqcup_S\{c_i \mid D_i \subseteq D\}$.
Det finnes opplagt bare tellbart mange slike kompakter, og alle uniforme objekter kan approksimeres av slike kompakter, så de utgjør et Scott-domene.

Teorem 8.18 *Det finnes en kontinuerlig funksjon*

$$PROD : (DOM \to DOM) \to DOM$$

slik at hvis $\Phi : DOM \to DOM$ vil $PROD(\Phi) \in DOM$ være isomorf til $\prod \Phi$.
Videre vil isomorfien iso_Φ fra $\prod \Phi$ til $PROD(\Phi)$ med invers osi_Φ være kontinuerlig i Φ.

Bevis
Vi så over at $\prod \Phi$ kan beskrives som et subdomene av $DOM \to S$. Det er lett å se fra beviset av lemmaet over at dette subdomenet avhenger kontinuerlig av Φ. Siden $DOM \to S$ er et Scott-domene, kan dette embeddes inn i S. Vi lar $PROD(\Phi)$ være bildet av $\prod \Phi$ under denne embeddingen. Kontinuitet av $\Phi \mapsto iso_\Phi$ og $\Phi \mapsto osi_\Phi$ følger av at alle konstruksjoner er uniforme.

8.2.4 Tolkning av typer og termer

Vi er nå klare til å tolke alle termene i System F. Målet vil være å tolke alle typene U med fri variable blant $X_1, \ldots X_n$ som kontinuerlige funksjoner

$$[\![U]\!]_F : DOM^n \to DOM$$

og alle objekttermer t som kontinuerlige funksjoner $[\![t]\!]_F$ definerte på passende domener som vi vil beskrive senere.

Definisjon 8.19 La U være en type med fri typevariable blant $\{X_i\}_{i\in I}$ hvor I er en endelig mengde indekser for typevariable. La $\{D_i\}_{i\in I}$ være et element i DOM^I.
Vi definerer $[\![U]\!]_F(\{D_i\}_{i\in I}) \in DOM$ ved rekursjon på oppbyggingen av U som følger:

1. Hvis $U = X_i$, la $[\![U]\!]_F(\{D_i\}_{i\in I}) = D_i$.
2. Hvis $U = V \to W$, la
$$[\![U]\!]_F(\{D_i\}_{i\in I}) = FUNC([\![V]\!]_F(\{D_i\}_{i\in I}), [\![W]\!]_F(\{D_i\}_{i\in I})).$$
3. Hvis $U = \prod X_j V$ la $\Phi(D) = [\![V]\!]_F([\{D_i\}_{i\in I}]_j^D)$, hvor vi kommer fra $\{D_i\}_{i\in I}$ til $[\{D_i\}_{i\in I}]_j^D$ ved å erstatte D_j med D om $j \in I$, og å utvide I med j til $J = I \cup \{j\}$ om $j \notin I$ og så la $D_j = D$.
Da lar vi $[\![U]\!]_F(\{D_i\}_{i\in I}) = PROD(\Phi)$.

Følgende observasjon er etterhvert så standard at vi ikke gir beviset:

Lemma 8.20 *For alle typer U og alle indeksmengder I er $[\![U]\!]_F : DOM^I \to DOM$ kontinuerlig.*

Vi kan nå se på problemene knyttet til å tolke objekttermer t. Hovedproblemet er å beskrive tolkningsområdet som et domene vi kan tolke t som et objekt i.

La t være en term av type U med fri typevariable $\{X_i\}_{i\in I}$ som over.
Objektvariablene til t vil være typede, $\{x_j^{V_j}\}_{j\in J}$.
En *tilordning* vil da være to sekvenser beskrevet som indekserte mengder $\{D_i\}_{i\in I}$ og $\{d_j\}_{j\in J}$ hvor hver $D_i \in DOM$ og $d_j \in [\![V_j]\!]_F(\{D_i\}_{i\in I})$. "verdien" til t skal da ligge i $[\![U]\!]_F(\{D_i\}_{i\in I})$
Ettersom tiden nå er moden for å droppe detaljer gir vi definisjonene og formulerer lemmaene uten å gi de omstendelige, men enkle, bevisene.

Definisjon 8.21 a) En *operator over indeksmengden I* er en kontinuerlig funksjon $\Phi : DOM^I \to DOM$.

b) Hvis Φ er en operator over indeksmengden I er ϕ definert på DOM^I et *uniformt objekt* hvis

   i) $\phi(\{D_i\}_{i\in I}) \in \Phi(\{D_i\}_{i\in I})$ for alle $\{D_i\}_{i\in I}$.

ii) Det finnes en kontinuerlig $\hat{\phi} : \{D_i\}_{i\in I} \to S$ slik at for alle $\{D_i\}_{i\in I}$ vil

$$\phi(\{D_i\}_{i\in I}) = \sqcup_{\Phi(\{D_i\}_{i\in I})}\{c \in (\Phi(\{D_i\}_{i\in I}))_0 \mid c \sqsubseteq_S \hat{\phi}(\{D_i\}_{i\in I})\}.$$

c) Vi generaliserer ordningen fra operatorer i en variabel på den kanoniske måten.

Definisjon 8.22 La t være en term i System F av type U, la $\{X_i\}_{i\in I}$ omfatte alle de fri typetermene til t og la $\{x_j^{V_j}\}_{j\in J}$ omfatte alle fri objektvariable i t. Vi vil tolke t som et uniformt objekt $[\![t]\!]_F$ for operatoren

$$\Phi(\{D_i\}_{i\in I}) = \prod_{j\in J}[\![V_j]\!]_F(\{D_i\}_{i\in I}) \to [\![V]\!]_F(\{D_i\}_{i\in I})$$

på følgende måte:

1. Hvis $t = x_k^{V_k}$ lar vi $[\![t]\!]_F(\{D_i\}_{i\in I})(\{d_j\}_{j\in J}) = d_k$
2. Hvis $t = sr$, hvor s er av type $U \to V$ og r er av type U, lar vi

$$[\![t]\!]_F(\{D_i\}_{i\in I})(\{d_j\}_{j\in J}) =$$

$$(\phi^{-1}([[\![s]\!]_F(\{D_i\}_{i\in I})(\{d_j\}_{j\in J})]^+)([[\![r]\!]_F(\{D_i\}_{i\in I})(\{d_j\}_{j\in J})]^+))^{[\![V]\!]_F(\{D_i\}_{i\in I})},$$

hvor ϕ er som i beviset for Lemma 8.15 og ϕ^{-1} er den tilsvarende projeksjonen

$$\phi^{-1} : S \to (S \to S).$$

3. Hvis $t = \lambda x_k^{V_k}.r$ antar vi uten tap av generalitet at $k \notin J$. (Det motsatte vil bare kreve enda mer notasjon fra oss, og det er nok notasjon i denne definisjonen som det er.) Vi lar $K = J \cup \{k\}$ og vi lar $[\{d_j\}_{j\in J}]_k^a$ være utvidelsen av indekseringen til at d_k settes lik a.
Da lar vi

$$[\![t]\!]_F(\{D_i\}_{i\in I})(\{d_j\}_{j\in J}) = \phi(f^+_{\{D_i\}_{i\in I},\{d_j\}_{j\in J}})$$

hvor

$$f^+_{\{D_i\}_{i\in I},\{d_j\}_{j\in J}} : [\![U]\!]_F(\{D_i\}_{i\in I}) \to [\![V]\!]_F(\{D_i\}_{i\in I})$$

er definert ved

$$f^+_{\{D_i\}_{i\in I},\{d_j\}_{j\in J}}(a) = [\![r]\!]_F(\{D_i\}_{i\in I})([\{d_j\}_{j\in J}]_k^a).$$

4. Hvis $t = sU$ hvor s er av type $\prod X_k.V$ lar vi
$$[\![t]\!]_F(\{D_i\}_{i\in I})(\{d_j\}_{j\in J}) =$$
$$osi_\Phi([\![s]\!]_F(\{D_i\}_{i\in I})(\{d_j\}_{j\in J}))([\![U]\!]_F(\{D_i\}_{i\in I})(\{d_j\}_{j\in J}))$$
hvor
$$\Phi(D) = [\![V]\!]_F([\{D_i\}_{i\in I}]_k^D).$$
Se beviset for Teorem 8.18 for notasjon.

5. Hvis $t = \Lambda X_k.s$ hvor s har type V lar vi
$$[\![t]\!]_F(\{D_i\}_{i\in I})(\{d_j\}_{j\in J}) = iso_\Phi(\phi)$$
hvor
$$\Psi(D) = [\![V]\!]_F([\{D_i\}_{i\in I}]_k^D)$$
og
$$\phi(D) = [\![s]\!]_F([\{D_i\}_{i\in I}]_k^D)(\{d_j\}_{j\in J}).$$

Legg merke til i punktene 4. og 5. at X_k ikke kan forekomme fritt i noen V_j, det vil si typen til x_j. Dette er en syntaktisk begrensning på System F, og den er nødvendig for at tolkningsområdet til x_j er bestemt av $\{D_i\}_{i\in I}$. Ellers ville definisjonen av Φ i de to punktene blitt umuliggjort.

Teorem 8.23 *a) For alle t vil $[\![t]\!]_F$ være et uniformt objekt av den typen som er foreskrevet.*

b) Hvis t er en term av type U med en fri variabel x_k av type V_k og s er en term av type V_k vil vi ha
$$[\![t^s_{x_k}]\!]_F(\{D_i\}_{i\in I})(\{d_j\}_{j\in J}) = [\![t]\!]_F(\{D_i\}_{i\in I})([\{d_j\}_{j\in J}]_k^{[\![s]\!]_F(\{D_i\}_{i\in I})(\{d_j\}_{j\in J})}).$$

c) Hvis t er en term av type U med en fri typevariabel X_k og V er en type med fri typevariable blant $\{X_i\}_{i\in I}$ vil vi ha
$$[\![t^V_{X_k}]\!]_F(\{D_i\}_{i\in I})(\{d_j\}_{j\in J}) = [\![t]\!]_F((\{D_i\}_{i\in I})_k^{[\![V_k]\!]_F(\{D_i\}_{i\in I})})(\{d_j\}_{j\in J}).$$

d) Vår tolkning respekterer β-regelen for objekter.

e) Vår tolkning respekterer β-regelen for typer.

Bevis
a) bevises ved en omstendelig men for hvert skritt enkel induksjon over oppbyggingen av t.
b) og c) vises også ved induksjon på oppbyggingen av t, hvor det er den voldsomme notasjonen som skaper de eneste vanskene.
d) og e) vises fra b) og c) ved regning i tråd med verifikasjonen av β-regelen for modellene våre for utypet λ-kalkyle. Igjen er det den voldsomme notasjonen som skaper problemer. Som i tilfellet av utypet λ-kalkyle vil vi bruke ep-parene i rett rekkefølge slik at vi oppnår de ønskede identitetene.

8.3 Strukturelle egenskaper

En svakhet ved den modellen vi har konstruert så langt er at tolkningene våre av typer i System F kan være ganske rikholdige, og en analyse av hvordan typer og termer tolkes gir liten feedback til en forståelse av System F. La oss for eksempel se på tolkningen av $\prod X.X$. Det vil ikke finnes noen lukket term av denne typen, men det sier ikke modellen vår noe om. For hver $d \in S$, lar vi $\phi_d \in \prod id_{DOM}$ være definert ved

$$\phi_d(D) = d^D.$$

Da er ϕ_d et uniformt objekt for id_{DOM}. Hvis på den annen side ϕ er et uniformt objekt for id_{DOM} lar vi $d = \phi(S)$, og vi vil ha at $\phi(D) \sqsubseteq_D d^D$. så $\phi \sqsubseteq \phi_d$ for en d. Det er imidlertid ikke slik at ϕ må være på formen ϕ_d, selv om vi her ikke har bruk for å vise det ved et eksempel.

Vi skal nå se på noen strukturelle egenskaper ved tolkningene våre som kan brukes til en bedre analyse av System F. Den første slike egenskapen er *totalitet*.

8.3.1 Totaliteter

Definisjon 8.24 La D være et domene. En *totalitet* på D er en vilkårlig delmengde $\overline{D}$ av D.

Bemerkning 8.25 Vi følger Girard her og har et så generelt totalitetsbegrep som mulig. Det er ikke noe i veien til for eksempel å begrense seg til delmengder $\overline{D}$ av D som er lukket oppover, eller sågar til mengder som i tillegg oppfyller andre egenskaper av interesse i andre sammenhenger, men omkostningen vil da være at vi må vise at de totalitetene for lukkede typer som vi vil konstruere har disse ekstra egenskapene. En mulig gevinst kan være at vi vil ekskludere domeneelementer som umulig kan være totale i en slik forstand som potensielle tolkninger av lukkede termer. Dette går vi ikke nærmere inn på, men det kan være et aktuelt tema om noen vil følge opp disse trådene.

Vi skal nå utvide vår tolkning av typer som elementer i DOM til tolkning av typer som elementer i DOM med en utvalgt totalitet.

Definisjon 8.26 La U være en type med fri variable $\{X_i\}_{i\in I}$. La $D_i \in DOM$ for hver $i \in I$ og la $\overline{D}_i$ være en totalitet på D_i for hver $i \in I$.
Vi definerer den *avledede totaliteten* $\overline{[\![U]\!]_F(\{X_i\}_{i\in I})}$ ved rekursjon på oppbygningen av U som følger:

1. Hvis $U = X_i$ lar vi $\overline{[\![U]\!]_F(\{X_i\}_{i\in I})} = \overline{D}_i$.

2. Hvis $U = V \to W$ definerer vi en totalitet på $[\![V]\!]_F(\{X_i\}_{i\in I}) \to [\![W]\!]_F(\{X_i\}_{i\in I})$ ved at
   f er total hvis for alle $a \in \overline{[\![V]\!]_F(\{X_i\}_{i\in I})}$ så er $f(a) \in \overline{[\![W]\!]_F(\{X_i\}_{i\in I})}$.
   Vi lar så totaliteten følge embeddingen av funksjonsrommet inn i S.

3. Hvis $U = \prod X_k.V$, la $\Phi(D) = [\![V]\!]([\{D_i\}_{i\in I}]_k^D)$. Vi lar $\phi \in \prod \Phi$ være total hvis $\phi(D) \in \overline{[\![U]\!]_F([\{D_i\}_{i\in I}]_k^D)}$ for alle $D \in DOM$ og alle totaliteter $\overline{D}$ på D. Totaliteten til tolkningen av U følger nå embeddingen.

Bemerkning 8.27 Hvis vi ønsker å begrense oss til totaliteter $\overline{D}$ slik at $d \in \overline{D}$ og $d \sqsubseteq_D e$ medfører at $e \in \overline{D}$, så vil ikke det medføre noen problemer. Denne påstanden har vi samlet som en ganske omfattende Oppgave 8.4

Eksempel 8.28 $\prod X.X$:
Dette er en lukket type. Vi har følgelig en tolkning $[\![\prod X.X]\!]_F \in DOM$, og vi har ingen valgfrihet når vi definerer totaliteten på dette domenet.
Vi skal se at totaliteten her er den tomme delmengden. Argumentet blir mer oversiktlig hvis vi arbeider direkte med $\prod id_{DOM}$.
Anta at $\phi \in \prod id_{DOM}$. La $D \in DOM$ og $d = \phi(d) \in D$.
Enhver totalitet på D som ikke inneholder d vil være et moteksempel til at ϕ er total.

Eksempel 8.29 $\prod X.(X \to X)$: Igjen er det lettere å arbeide med $\prod D \in DOM.(D \to D)$ i stedet for den isomorfe kopien i DOM.
La $\phi(D) = id_D$. Hvis $\overline{D}$ er en totalitet på D vil id_D avbilde $\overline{D}$ på $\overline{D}$ uavhengig av hvordan den er valgt.
Hvis $\phi(D) \neq id_D$, la $d \in D$ være slik at $\phi(D)(d) = e \neq d$. La $\overline{D}$ være slik at $d \in \overline{D}$ mens $e \notin \overline{D}$.
Med denne tolkningen blir ikke $\phi(D)$ total.

Skal nå vise at alle tolkninger av termer er totale så langt dette gir mening:

Teorem 8.30 *La t være en term av type U med fri variable $\{x_j^{V_j}\}_{j\in J}$ og la $\{X_i\}_{i\in I}$ omfatte alle fri typevariable i t.*
La $(D_i, \overline{D_i})$ være et domene (i DOM) med totalitet for hver $i \in I$, og la $d_j \in \overline{[\![V_j]\!]_F(\{D_i\}_{i\in I})}$ for hver $j \in J$.
Da er

$$[\![t]\!]_F(\{D_i\}_{i\in I})(\{d_j\}_{j\in J}) \in \overline{[\![U]\!]_F(\{D_i\}_{i\in I})}.$$

Bevis
Beviset går ved induksjon på oppbyggingen av t. Hvis man insisterer på å holde seg til den formelle (og notasjonsmessig svært komplekse) definisjonen, kan man få bokholderiproblemer. Vi skisserer beviset basert på de reelle konstruksjonene vi har tatt isomorfe kopier av:

1. $t = x_j^{V_j}$. Da er påstanden triviell ut fra antagelsen om at $d_j \in \overline{[\![V_j]\!]_F(\{D_i\}_{i\in I})}$.

2. $t = sr$: Ettersom både s og r i henhold til induksjonsantagelsen tolkes som totale objekter, tolkes t som et totalt objekt.

3. $t = \lambda x_k^{V_k}$: Totalitet av $[\![t]\!]_F$ under en gitt instans betyr at hver gang x_j tolkes som et totalt objekt, er den tilsvarende tolkningen av s et totalt objekt. Dette er en reformulering av induksjonsantagelsen.

4. $t = sV$: s tolkes som et totalt, uniformt objekt ϕ i tolkningen Φ av U. Det betyr spesielt at $\phi([\![V]\!]_F(\{D_i\}_{i\in I}))$ er total i $\Phi([\![V]\!]_F(\{D_i\}_{i\in I}))$ med totaliteten $\overline{[\![V]\!]_F(\{D_i\}_{i\in I})}$. Det er dette som skal til.

5. $t = \Lambda X_k^{V_k}.s$: Vi kan bruke induksjonshypotesen direkte analogt med punkt 3.

Dette avslutter beviset.

Vi får som en direkte konsekvens av dette teoremet at det ikke finnes noen lukkede termer av type $\prod X.X$. Vi får også at alle lukkede termer av type $\prod X.(X \to X)$ har den samme denotasjonelle tolkningen som $\Lambda X.\lambda x^X.x$, men vi kan selvfølgelig ikke slutte at alle lukkede termer av denne typen har $\Lambda X.\lambda x^X.x$ som normalform.

Vi skal nå analysere hvordan totale objekter i $[\![BOOLE]\!]_F$ kan se ut. Vi vil se at vi nesten kan vise at et totalt element må være tolkningen av $\mathbf{T}$ eller tolkningen av $\mathbf{F}$, men vi vil ikke komme helt i land.

Vi bruker teoremet over til å se at $k_{\mathbf{T}} = \Lambda X.\lambda x^X \lambda y^X.y$ og $k_{\mathbf{F}} = \Lambda X.\lambda x^X \lambda y^X.x$ tolkes som totale objekter i $[\![BOOLE]\!]_F$.
La ϕ være et uniformt og totalt objekt for $\Phi(D) = (D, D \to D)$
Da må $\phi(D)(d, e) \in \{d, e\}$ for alle $D \in DOM$ og $d, e \in D$, ettersom totaliteten $\{d, e\}$ ellers vil være et moteksempel til at ϕ er total. Dette faktum vil vi bruke mange, mange ganger i resten av beviset uten å nevne det direkte.
Vi arbeider først med $\phi(S)$ og kompakter i S_0. Her vil vi se at $\phi(S)$ vil svare til tolkningen av en av de to sannhetsverdiene begrenset til S.
La a og b være inkonsistente kompakter slik at (uten tap av generalitet) $\phi(S)(a, b) = b$. Vi vil vise at vi for alle c og d i S har at $\phi(S)(c, d) = d$, og det er selvfølgelig tilstrekkelig å la c og d være kompakter.
Vi vil utnytte at S er mettet, og vi vil bevise påstanden over gjennom flere delpåstander. La $f = \phi(S)$. Hver gang vi påstår at "det finnes ..." bruker vi at S er mettet.

Påstand 1
Hvis c er inkonsistent med a er $f(a, c) = c$ og hvis c er inkonsistent med b er $f(c, b) = b$.

Bevis
I det første tilfellet vil det finnes en c' slik at c' er inkonsistent med a og under både c og b.
Da følger det at $f(a, c') = c'$ fordi $f(a, c') \sqsubseteq f(a, b) = b$ og at $f(a, c) = c$
I det andre tilfellet finnes det en c' som er inkonsistent med b og under a og c. Da er

$$b = f(a, b) = f(c', b) = f(c, b).$$

Påstand 2
Hvis c og d er inkonsistente, vil $f(c, d) = d$.

Bevis
Vi kan finne c' som er inkonsistent med alle a, b, c og d.
Da bruker vi Påstand 1 og får

$$f(a,b) = b \Rightarrow f(c',b) = b \Rightarrow f(c',d) = d \Rightarrow f(c,d) = d.$$

Påstand 3
For alle c og d vil $f(c,d) = d$.
Bevis
Hvis $c = d$ er dette riktig, og hvis c og d er inkonsistente er dette bevist. Anta derfor at c og d er konsistente, men forskjellige.
Hvis d ligger ekte under c, vil d ha en utvidelse d' som er inkonsistent med c, Da er $f(c,d') = d'$, og siden $f(c,d) \sqsubseteq f(c,d')$ må vi ha at $f(c,d) = d$.
Hvis d ikke ligger ekte under c kan vi utvide c til en c' inkonsistent med d. Da er $f(c,d') = d'$ og vi må ha et $f(c,d) = d$ av monotonitetsgrunner.

Hvis $D \in DOM$ følger det nå under forutsetningen at det finnes inkonsistente a og b i S slik at $\phi(S)(a,b) = b$ at

- $\phi(D)(d,e) \in \{d,e\}$ for alle $e \in D$ og $e \in D$.
- $\phi(D)(d,e) \sqsubseteq_D e$ for alle d og e i D.

Den eneste situasjonen vi står igjen med og som kan tjene som et moteksempel er

- d ligger ekte under e i D.
- $\phi(D)(d,e) = d$

Denne situasjonen vil vi ikke kunne unngå fullstendig, da kreves det litt mer struktur. Vi kan likevel begrense fraviket fra $[\![\mathbf{T}]\!]_F$ ytterligere. Så la D, d og e være et moteksempel som over.
Påstand 4
Hvis $a \sqsubseteq_D e$ vil $a \sqsubseteq d$ eller $d \sqsubseteq a$.
Bevis
Da vil $\phi(D)(d,a) \sqsubseteq \phi(D)(d,e) = d$. Hvis $\phi(D)(d,a) = a$ betyr det at $a \sqsubseteq_D d$ og hvis $\phi(D)(d,a) = d$ er vi i samme abnorme situasjon som i utgangspunktet, og vi må ha at $d \sqsubseteq a$.

Vi skal nå se på totale elementer i tolkningen av NAT.
Vi minner om at

$$NAT = \prod X.((X \to X), X \to X).$$

I resten av dette avsnittet vil vi la ϕ være et uniformt objekt for operatoren $\Phi(D) = ((D \to D), D \to D)$ (hvor vi for oversiktens skyld ikke lar $\Phi(D) \in DOM$ selv om $D \in DOM$) slik at $\phi(D)$ er total i den nedarvede totaliteten på $((D \to D) \to D)$ for enhver totalitet $\overline{D}$ på D.

Lemma 8.31

$$\forall D \in DOM \forall f \in D \to D \forall d \in D \exists n \in \mathbb{N}(\phi(D)(f, d) = f^n(d)).$$

Bevis
La D, f og d være gitt. La $\overline{D} = \{f^n(d) \mid n \in \mathbb{N}\}$ være en totalitet.
Både f og d er totale med hensyn til $\overline{D}$, så $\phi(D)(f, d) \in \overline{D}$.
Dette viser lemmaet.

Problemet blir å vise at vi, i det minste langt på vei, kan finne en uniform n som er uavhengig av D, f og d. Vi skal ta utgangspunkt i en naturlig kandidat.
Siden $\mathbb{N}_\perp$ kan embeddes i S lar vi $\{\perp, a_0, a_1, \ldots\}$ være en isomorf kopi av $\mathbb{N}_\perp$ i S. La $suc : S \to S$ være den minste kontinuerlige funksjonen slik at $suc(a_i) = a_{i+1}$ for alle $i \in \mathbb{N}$ og la $n = n_\phi$ være slik at $\phi(S)(suc, a_0) = a_n$.
Dette tallet finnes i henhold til lemmaet over. Vi skal holde ϕ og n fast i resten av dette avsnittet.

Vi minner om at hvis a og b er kompakter er $f_{\langle a,b\rangle}$ definert som den minste f slik at $f(a) = b$. Enhver kompakt kan skrives som den minste øvre grensen av endelig mange funksjoner på formen $f_{\langle a,b\rangle}$.

Definisjon 8.32 a) La $f \in S \to S$ og $a \in S$ være kompakte.
(f, a) er et *generatorpar* hvis det finnes parvis inkonsistente kompakter $a_0, \ldots, a_k$ i S slik at $a = a_0$ og

$$f = \bigsqcup \{f_{\langle a_i, a_{i+1}\rangle} \mid i < k\}.$$

Vi sier at (f, a) *genererer* $a_0, \ldots, a_k$.

b) La (f, a) og (g, b) være to generatorpar som genererer hhv. $a_0, \ldots, a_k$ og $b_0, \ldots, b_l$. Vi sier at (f, a) og (g, b) er *uavhengige* hvis a_i og b_j er inkonsistente for alle $i \le k$ og $j \le l$.

Vi ser at hvis (f, a) er et generatorpar, vil $a_i = f^i(a)$ for alle $i \leq k$.

Lemma 8.33 *La (f, a) og (g, b) være uavhengige generatorpar slik at $\phi(f, a) \neq \perp$ og $\phi(g, b) \neq \perp$. Da finnes en felles m slik at $\phi(S)(f, a) = f^m(a)$ og $\phi(S)(g, b) = g^m(b)$.*

Bevis
La $a_0, \ldots, a_k$ og $b_0, \ldots, b_l$ være som over. Vi kan anta at $k \leq l$.
Siden S er mettet kan vi vinne en uendelig følge $\{c_i\}_{i \in \mathbb{N}}$ av parvis inkonsistente kompakter slik at

- Hvis $i \leq k$ vil $c_i \sqsubseteq a_i$ og $c_i \sqsubseteq b_i$.
- Hvis $k < i \leq l$ vil $c_i = b_i$.
- Hvis $j \leq k$ og $j \neq i$ vil c_i og a_j være inkonsistente.
- Hvis $j \leq l$ og $j \neq i$ vil c_i og b_j være inkonsistente.

La

$$f^+ = \bigsqcup(\{f\} \cup \{f_{\langle c_i, a_{i+1} \rangle} \mid i < k\} \cup \{f_{\langle c_i, c_{i+1} \rangle} \mid k \leq i\})$$

og

$$g^+ = \bigsqcup(\{g\} \cup \{f_{\langle c_i, a_{i+1} \rangle} \mid i < l\} \cup \{f_{\langle c_i, c_{i+1} \rangle} \mid l \leq i\}).$$

Da er $f \sqsubseteq f^+$ og $g \sqsubseteq g^+$. Siden $\perp \neq \phi(S)(f, a) \sqsubseteq \phi(S)(f^+, a)$, de to sistnevnte er på formen $f^i(a)$ og $(f^+)^j(a)$ mens de er inkonsistente om $i \neq j$, ser vi at det må finnes $i \leq k$ slik at $\phi(S)(f^+, a) = a_i$. Tilsvarende må det finnes $j \leq l$ slik at $\phi(S)(g^+, b) = b_j$.
La nå $h = \bigsqcup\{f_{\langle c_i, c_{i+1} \rangle} \mid i \in \mathbb{N}\}$.
Da er $h \sqsubseteq f^+$ og $h \sqsubseteq g^+$. La $c = c_0$.
Da vil $\phi(S)(h, c) \sqsubseteq \phi(S)(f^+ .a)$ og $\phi(S)(h, c) \sqsubseteq \phi(S)(g^+, b)$.
Siden $\phi(S)(h, c)$ er på formen $h^m(c) = c_m$, må vi ha $i = m = j$, og vi er fremme.

Lemma 8.34 *La $n = n_\phi$. La (f, a) være et generatorpar slik at $\phi(S)(f, a) \neq \perp$.*
Da vil $\phi(S)(f, a) = f^n(a)$.

Bevis
La m være så stor at $\phi(S)(suc[m], 0) \neq \perp$. Ved mettethet kan vi finne et generatorpar (g, b) uavhengig av både $(suc[m], 0)$ og (f, a) slik at også

$\phi(S)(g, b) \neq \perp$.
Ved to gangers bruk av forrige lemma har vi at $\phi(S)(f, a) = f^n(a)$.

Vi er nå i stand til å vise det generelle resultatet om hvordan totale objekter i NAT kan se ut:

Teorem 8.35 *La* $\phi \in \prod X.((X \to X), X \to X)$ *være total, og la* $n = \phi(S)(suc, 0)$.
For alle $D \in DOM$, $f \in D \to D$ *og* $a \in D$ *vil da*

i) $\phi(D)(f, a) = f^m(a)$ *for en* $m \in \mathbb{N}$.

ii) $\phi(D)(f, a) \sqsubseteq f^n(a)$.

Bevis
i) har vi allerede vist. Grunnen til at vi nevner i) igjen er at i) sammen med ii) i mange tilfeller vil gi oss at $\phi(D)(f, a) = f^n(a)$.
For å vise ii) er det tilstrekkelig å se på tilfellet hvor f og a er kompakte. For alle i, la $a_i = f^i(a)$. Siden f er kompakt, vil denne følgen være syklisk etter en tid.
Vi beveger oss nå ut i S, bruker at S er mettet, og finner en følge $\{b_i\}_{i \in \mathbb{N}}$ av parvis inkonsistente kompakter slik at

- $a_i \sqsubseteq b_i$
- hvis $f_{\langle c,d \rangle} \sqsubseteq f$ er en av de endelig mange trinnfunksjonene som utgjør f, skal vi ha at $c \sqsubseteq a_i$ eller b_i er inkonsistent med c. Det tilsvarende vil vi kreve i forhold til d.

Da er $\{f, f_{\langle b_i, b_{i+1} \rangle} \mid i \in \mathbb{N}\}$ en begrenset mengde. La f^+ være den minste øvre grensen. Det følger fra i) at $\phi(S)(f^+, b) = b_m$ for en m.
La nå $g = \bigsqcup\{f_{\langle b_i, b_{i+1} \rangle} \mid i \in \mathbb{N}\}$, og la $b = b_0$.
g genererer en parvis inkonsistent følge av kompakter fra b, og ved lemmaet over vil $\phi(S)(g, b) = b_n$. Siden $g \sqsubseteq f^+$ følger det at $\phi(S)(f^+, b) = b_n$, siden ellers ville verdiene vært inkonsistente.
Hvis vi så ser på $\phi(S)(f, a)$, må det finnes en m slik at $\phi(S)(f, a) = a_m \sqsubseteq b_n$.
Valget av b_n gir oss da at $a_m \sqsubseteq a_n = f^n(a)$.
Siden $\phi(S)(f, a) \sqsubseteq f^n(a)$ vil vi også ha at $\phi(D)(f, a) \sqsubseteq f^n(a)$. Dette avslutter beviset.

8.3.2 Funktorielle aspekter

I dette avsnittet vil vi beskrive et lite forskningsprosjekt. Hvis vi lar U være en type med fri variable $X_1, \ldots, X_n$, vi lar $D_1, \ldots, D_n$ og $E_1, \ldots, E_n$ være mulige tolkninger i DOM av $X_1, \ldots, X_n$ og vi antar at vi har ep-par (η_i, π_i) fra D_i til E_i for hver i, vil vi kunne beskrive et ep-par (η, π) fra $[\![U]\!]_F(D_1, \ldots, D_n)$ til $[\![U]\!]_F(E_1, \ldots, E_n)$.

Det er en rimelig hypotese at tolkningen av termer respekteres av disse epparene i den forstand at π delene sender tolkning av term på tolkning av term under forutsetning av at dette gjelder for tolkningene av de fri objektvariablene.
Dette lar seg imidlertid ikke vise ved første forsøk, en direkte induksjon på oppbyggingen av termen. Prosjektet går derfor ut på å avgjøre om hypotesen er sann, og i tilfelle den er det, å finne flere strukturelle egenskaper ved tolkninger av termer som gjør det mulig å bevise hypotesen.

8.4 Oppgaver til Kapittel 8

Oppgave 8.1 Vis Lemma 8.1b).

Oppgave 8.2 La S og T være to mettede Scott-domener. Vis at S og T er isomorfe.
Hint: Bruk beviset for at alle Scott-domener kan embeddes i både S og T til å konstruere en sekvens av partielle isomorfier, ved at vi i annethvert skritt konstruerer en embedding av S inn i T og i annethvert skritt sørger for at den inverse er en embedding av T inn i S.

Oppgave 8.3 Vis Lemma 8.11.

Oppgave 8.4 I denne oppgaven skal vi la en totalitet på et domene D være en mengde $\overline{D} \subseteq D$ slik at

$$d \in \overline{D} \wedge d \sqsubseteq e \Rightarrow e \in \overline{D}.$$

a) Vis at om vi bruker denne definisjonen som grunnlag for å tolke de totale elementene i domenetolkningen av en type, så er fortsatt tolkningene totaliteter i denne forstanden.

b) Vis at alle termer vil tolkes som totale objekter i denne nye betydningen samme forstand som vi hadde det i den opprinnelige betydningen.

c) [Litt krevende] Vis at vi fortsatt ikke har noen totale objekter i $[\![\prod X.X]\!]_F$.

d) [Langt mer krevende] Vis at den uniforme identitetsfunksjonen fortsatt er det eneste totale objektet i $[\![\prod X.X \to X]\!]_F$.
Advarsel: Tilsynelatende vil alle utvidelser av den uniforme identitetsfunksjonen være totale i denne nye forstanden. Du bør derfor vise at den uniforme identitetsfunksjonen er maksimal og derfor ikke kan utvides. Her må du, sålangt oppgavestilleren forstår, benytte deg av at S er mettet.

Kapittel 9

Tillegg

9.1 Den naive modellen for T er ikke fullt abstrakt

I dette avsnittet skal vi vise at den opprinnelige modellen vår for Gödels T ikke er fullt abstrakt. Det er gode grunner til at dette stoffet er utenfor pensum. For det første er ikke resultatet en naturlig del av pensum. For det andre bygger beviset på kontinuitetsbetraktninger som ikke alle kursdeltagene kan antas å være fortrolige med.
Når vi likevel tar det med, er det fordi beviset illustrerer styrken til Gödels T, vi er faktisk i stand til, innenfor T, til å avgjøre om visse følger konvergerer eller ikke, selv om konvergens er definerbart bare ved hjelp av et par tallkvantorer. Denne delen av beviset bruker en metode utviklet av Tom Grilliot tidlig på 70-tallet, men som trolig benyttes for første gang innenfor rammen av T i dette kompendiet.

Definisjon 9.1

1. La b være en av basistypene ι eller o. La $a \in ||b||$ og la $a_n \in ||b||$ for alle $n \in \mathbb{N}$.
   Vi lar $a = \lim_{n\to\infty} a_n$ bety at $a = a_n$ for alle untatt endelig mange n.

2. La nå $\sigma = \delta \to \tau$, la $F \in ||\sigma||$ og la $F_n \in ||\sigma||$ for alle $n \in \mathbb{N}$.
   Vi lar $F = \lim_{n\to\infty} F_n$ bety at $F(a) = \lim_{n\to\infty} F_n(a_n)$ ut fra definisjonen for δ hver gang $a = \lim_{n\to\infty} a_n$ ut fra definisjonen for τ

Det er på plass med en liten advarsel her. For basistypene har vi brukt et naturlig konvergensbegrep, og for alle typer på nivå mindre enn 2 vil definisjonen vår svare til intuisjonen. Det finnes imidlertid funksjoner F på nivå 2 slik at konstantfølgen $F_n = F$ ikke har F som grense i den forstanden vi har gitt her. Dette er altså en teknisk definisjon som ikke helt samsvarer ved vanlig intuisjon. Vi skal likevel se at termene i T er "kontinuerlige" med hensyn til dette grensebegrepet.

Bemerkning 9.2 Vi har faktisk at

$$\{F \in ||\sigma|| \; ; \; F = \lim_{n \to \infty} F\}$$

er mengden av Kleenes opprinnelige tellbare funksjonaler. Det vil føre oss helt på viddene å gå nærmere inn på hvorfor.

Definisjon 9.3 La ν være en tilordning, og la ν_n være tilordninger for hver $n \in \mathbb{N}$.
Vi sier at $\nu = \lim_{n\to\infty} \nu_n$ dersom $\nu(i) = \lim_{n\to\infty} \nu_n(i)$ for alle $i \in \mathbb{N}$.

Lemma 9.4 *La t være en term i T og la ν og $\{\nu_n\}_{n\in\mathbb{N}}$ være tilordninger slik at $\nu = \lim_{n\to\infty} \nu_n$.*
Da vil $[\![t]\!](\nu) = \lim_{n\to\infty}[\![t]\!](\nu_n)$.

Bevis
Beviset er ved induksjon på oppbygningen av t og detaljene overlates leseren.
Hvis t er en av konstantene på nivå 0 eller 1, er $[\![t]\!](\nu)$ uavhengig av ν, så man må sjekke at $[\![t]\!] = \lim_{n\to\infty}[\![t]\!]$ i disse tilfellene. Det er lett. Hvis $t = Rec_\tau$ vil tolkningen av t være en funksjon F som tar tre argumenter, $a \in [\![\tau]\!]$, $f : \mathbb{N} \times [\![\tau]\!] \to [\![\tau]\!]$ og $m \in \mathbb{N}$, og vi må vise at om $a = \lim_{n\to\infty} a_n$, $f = \lim_{n\to\infty} f_n$ og $m = \lim_{n\to\infty} m_n$, så er

$$[\![t]\!](a, f, m) = \lim_{n\to\infty}[\![t]\!](a_n, f_n, m_n).$$

Det finnes en k slik at $m_n = m$ for alle $n \geq k$. Det betyr at for $k \geq n$ vil

$$[\![t]\!](a_k, f_k, m_k) = f_k(m, f_k(m-1, \cdots, f_k(1, a_k)\cdots))$$

og den tilsvarende likningen holder om vi fjerner indeksen k.
Vårt konvergensbegrep er designet slik at applikasjon skal respektere grenseoverganger, så m iterasjoner av applikasjon vil respektere grensene.
Hvis $t = x_i$ følger påstanden direkte fra antagelsen om at $\nu = \lim_{n\to\infty} \nu_n$.
Tilfellene med sammensetning og abstraksjon er enkle, og overlates leseren.

I resten av dette avsnittet skal vi la $\alpha : \mathbb{N} \to \mathbb{B}$ og $\alpha_n : \mathbb{N} \to \mathbb{B}$ være definert ved

- $\alpha(k) = \mathbf{T}$ for alle k.
- $\alpha_n(k) = \mathbf{T}$ hvis $k \leq n$.
- $\alpha_n(k) = \mathbf{F}$ hvis $n < k$.

Da er $\alpha = \lim_{n\to\infty} \alpha_n$.

Lemma 9.5 *Det finnes en lukket term t av type $((\iota \to o) \to o) \to o$ slik at for alle $F \in ||(\iota \to o) \to o||$ vil $[\![t]\!](F) = \mathbf{T}$ om $F(\alpha) = \lim_{n\to\infty} F(\alpha_n)$ og $[\![t]\!](F) = \mathbf{F}$ ellers.*

Bevis
Det vil forkludre forståelsen av argumentet om vi skal tvinge alle konstruksjoner inn i formalismen til T. Vi kan merke oss at ingen av de konstruksjonene vi gjør trenger noen avanserte deler av T, og de kunne vært gjennomført innen enhver endelig typeteori som utvider kjerneeksemplet med den karakteristiske funksjonen til $<$ og med begrensede kvantorer.
La $m \in \mathbb{N}$ være gitt, og definer β_m ved

$$\beta_m(k) = \begin{cases} \mathbf{T} & \text{om} \quad k < m \\ \mathbf{T} & \text{om} \quad m \leq k \wedge \forall n(m \leq n \leq k \Rightarrow F(\alpha_n) = F(\alpha)) \\ \mathbf{F} & \text{om} \quad \exists n(m \leq n \leq k \wedge F(\alpha_n) \neq F(\alpha)) \end{cases}$$

Da har vi at $\beta_m = \alpha$ hvis $\forall n \geq m(F(\alpha_n) = F(\alpha))$, mens $\beta_m = \alpha_n$ for den minste $n \geq m$ slik at $F(\alpha_n) \neq F(\alpha)$ ellers.
Det gir oss

$$\exists n \geq m(F(\alpha_n) \neq F(\alpha) \Leftrightarrow F(\beta_m) \neq F(\alpha)).$$

Det siste er avgjørbart i T, så dermed er det første avgjørbart.
Hvis $\forall n \geq 0(F(\alpha_n) = F(\alpha))$ setter vi $t(F) = \mathbf{T}$.
Hvis ikke definerer vi

$$\gamma(k) = \begin{cases} \mathbf{T} & \text{om} \quad \exists n > k(F(\alpha_n) \neq F(\alpha)) \\ \mathbf{F} & \text{om} \quad \forall n > k(F(\alpha_n) = F(\alpha)) \end{cases}$$

Hvis $F(\alpha) = \lim_{n\to\infty} F(\alpha_n)$ vil $\gamma = \alpha_n$ for den største n slik at $F(\alpha) \neq F(\alpha_n)$, mens $\gamma = \alpha$ ellers.
Da kan vi sette $t(F) = \mathbf{T}$ om $F(\gamma) \neq F(\alpha)$ og $t(F) = \mathbf{F}$ ellers.
Dette avslutter beviset.

Lemma 9.6 *La t være termen konstruert i beviset for Lemma 9.5 og la r være en lukket term av type $(\iota \to o) \to o$.*
Da vil $tr \to tt$.

Bevis
Ved Lemma 9.4 vil $[\![r]\!](\alpha) = \lim_{n\to\infty}[\![r]\!](\alpha_n)$, så $[\![t]\!]([\![r]\!]) = \mathbf{T}$. Lemmaet følger da av at modellen er adekvat.

Teorem 9.7 *Modellen for T hvor vi bruker det fulle typehierarkiet er ikke fullt abstrakt.*

Bevis
La t være termen konstruert i beviset for Lemma 9.5.
La $s = \lambda x.tt$ hvor s har samme type som t.
Hvis vi lar F være en funksjon slik at $F(\alpha)$ ikke er grensen til $F(\alpha_n)$ når n vokser, eksempelvis

$$F(f) = {}^2E(f) = \begin{cases} \mathbf{T} & \text{om} \quad \exists n(f(n) = \mathbf{F}) \\ \mathbf{F} & \text{om} \quad \forall n(f(n) = \mathbf{T}) \end{cases}$$

ser vi at $[\![t]\!](F) \neq [\![s]\!](F)$.
Det er derfor tilstrekkelig å vise at s og t er observabelt like.
Så la p være en term av basistype, med høyst en variabel x fri, hvor x har den samme typen som s og t.
Vi viser ved induksjon på lengden av maksimal normalisering av p^t_x at $p^t_x \equiv p^s_x$.
Hvis p ikke er på normalform, følger påstanden trivielt fra induksjonsantagelsen.
Hvis p er på normalform, er p på formen

$$p = Kp_1 \cdots p_k$$

hvor K er en konstant eller K er variabelen x. Muligheten at K er en abstraksjonsterm er uforenlig med antagelsen om at p både er av basistype og på normalform.
Vi nøyer oss med å se på tilfellet hvor K er x og på det mest komplekse tilfellet hvor K er en konstant, nemlig rekursjon :

Tilfelle $p = xp_1$:
Ved Lemma 9.6 vil $t(p_1)^t_x \to tt$ og vi har opplagt at $s(p_1)^s_x \to tt$, så påstanden holder.

Tilfelle $p = Rec_\tau p_1 \dots p_k$:
Da er p_3 av type ι, og ved induksjonsantagelsen finnes det et tall n slik at både $(p_3)^t_x$ og $(p_3)^s_x$ normaliseres til e_n.
Hvis $n = 0$ vil p^t_x kunne skrives om til til $(p_1 p_4 \cdots p_k)^t_x$ og p^s_x vil kunne normaliseres til $(p_1 p_4 \cdots p_k)^s_x$. Da følger påstanden ved induksjonsantagelsen for $(p_1 p_4 \cdots p_k)^t_x$.
Hvis $n = m + 1$ kan p^t_x reduseres til $(p_2 e_m (rec_\tau p_1 p_2 e_m) p_4 \cdots p_k)^t_x$ og p^s_x kan reduseres til $(p_2 e_m (rec_\tau p_1 p_2 e_m) p_4 \cdots p_k)^t_x$. Påstanden følger derfor fra induksjonsantagelsen for $(p_2 e_m (rec_\tau p_1 p_2 e_m) p_4 \cdots p_k)^t_x$.
Dette avslutter beviset for teoremet.

9.2 Induktivt definerte typer i System F

Vi har sett at vi i en viss forstand kan lage kartesiske produkter og funksjonsrom i System F. Det finnes mange andre typer som det kunne vært aktuelt å representere, som for eksempel typer av lister, typer av endelige trær etc. Det er slike konstruksjoner vi skal se på i dette avsnittet. La oss først se på at vi kan representere den tomme typen i System F:

Definisjon 9.8 La $T_\emptyset$ være typen $\prod X.X$

Lemma 9.9 *La t være en term på normalform, X en typevariabel og U en type.*
Da er t^U_X også på normalform.

Bevis
En redeks i t^U_X må være på formen $(\lambda x.s^U_X)s^U_X$ eller på formen $(\Lambda Y.s^U_X)V^U_X$ og vil komme fra en redeks i t. Siden t er på normalform, er dette umulig.

Bemerkning 9.10 En viktig forskjell mellom objekttermer og typetermer er at en typeterm aldri kan stå først i en applikasjonsterm. Det betyr at om vi substituerer en type inn for en variabel, kan vi ikke skape nye redekser, slik som vi kan når vi erstatter visse objektvariable med termer.

Lemma 9.11 *Det finnes ingen lukket term t av type $T_\emptyset$.*

Bevis
La t være lukket, på normalform og av type $T_\emptyset$. Vi skal vise umuligheten ved

induksjon på antall forekomster av Λ i t.
t vil enten 1. være en sammensetning eller 2. på formen $t = \Lambda X.s$, hvor s er på normalform.

1. t kan skrives entydig som en sammensetning $t = t_1 t_2 \cdots t_n$ hvor t_1 ikke er en sammensetning og hvor $n \geq 2$. Siden t er lukket, er ikke t_1 en variabel. Da er t_1 en applikasjonsterm, og $t_1 t_2$ er en redeks. Dette er i konflikt med antagelsen om at t er på normalform.

2. Ved forrige lemma er $s' = s_X^{T_\emptyset}$ en term på normalform. s' er av type $T_\emptyset$, lukket og med færre forekomster av Λ enn t, så induksjonsantagelsen sier at det er umulig.

Intuitivt sett er den tomme avbildningen en imbedding av den tomme mengden inn i enhver annen mengde. Dette kan vi uttrykke ved

Definisjon 9.12 For hver type U definerer

$$em_U = \lambda x^{T_\emptyset}.xU$$

den tomme avbildningen fra $T_\emptyset$ inn i U.
$\Lambda X.em_X$ av type $\prod X.(T_\emptyset \to X)$ representerer da den tomme avbildningen.

Før vi forklarer hensikten med å representere den tomme mengden og den tomme avbildningen i F, la oss gi en lyninnføring i induktive definisjoner.
La X være en mengde, og la $\mathcal{P}(X)$ være *potensmengden* til X, det vil si mengden av alle delmengder av X.
La $\Gamma : \mathcal{P}(X) \to \mathcal{P}(X)$. Vi sier at Γ er *monoton* hvis $A \subseteq B \subseteq X \Rightarrow \Gamma(A) \subseteq \Gamma(B)$.
Vi har at $\Gamma(X) \subseteq X$.

Teorem 9.13 *Hvis $\Gamma : \mathcal{P}(X) \to \mathcal{P}(X)$ er monoton som over, finnes det en minste mengde $\Gamma_\infty \subseteq X$ slik at $\Gamma(\Gamma_\infty) = \Gamma_\infty$, det vil si at Γ har et minste fikspunkt.*

Bevis
La

$$B = \bigcap \{A \subseteq X \mid \Gamma(A) \subseteq A\}.$$

Siden $\Gamma(X) \subseteq X$ tar vi snittet over en ikketom mengde, så B er velfundert.
La A være vilkårlig slik at $\Gamma(A) \subseteq A$. Da er $B \subset A$, så ved monotoniteten

er $\Gamma(B) \subseteq \Gamma(A) \subseteq A$. Siden A var valgt vilkårlig, følger det at $\Gamma(B) \subseteq B$. Det følger videre ved monotonitet at $\Gamma(\Gamma(B)) \subseteq \Gamma(B)$, så $\Gamma(B)$ var en av de mengdene A vi tok snittet av da vi definerte B. Det betyr at $B \subseteq \Gamma(B)$ og dermed at $B = \Gamma(B)$. Vi setter derfor $\Gamma_\infty = B$.
Det er opplagt fra definisjonen at dette blir det minste fikspunktet.

System F har langt på vei uttrykskraft nok til å formalisere denne konstruksjonen. Det er imidlertid slik at når det gjelder System F snakker vi ikke om inklusjon mellom mengder, men om embeddinger av en type inn i en annen. Videre kan vi ikke snakke om monotone avbildninger. Alternativet vil være positive type-konstruksjoner. Vi skal se på noen eksempler, med de såkalte velfunderte trærne med forgreninger over $\mathbb{N}$ på en mengde som hovedeksempel:

Definisjon 9.14 La A være en mengde. Vi definerer mengden av *velfunderte trær* med forgrening over $\mathbb{N}$ som følger:

1. Hvis $a \in A$ er $a*$ et velfundert tre. Et slikt tre kalles et *blad*. * er et syntaktisk grep vi bruker for å sikre at bladet ikke er en funksjon.
2. Hvis f er en funksjon med definisjonsområde $\mathbb{N}$ og verdiområde blant de velfunderte trærne, er f er et velfundert tre.
3. De velfunderte trærne er den minste klassen som oppfyller 1. og 2.

Hvis vi lar $\oplus$ betegne disjunkt union, kan vi definere de velkfunderte trærne over A som den minste mengden X slik at $X = A \oplus (\mathbb{N} \to X)$ og vi ser at definisjonen kan tilpasses formatet til induktive definisjoner.
Dette er ett av mange tilfeller av induktivt definerte strukturer det kan være aktuelt å se på.
$\mathbb{N}$ kan oppfattes som den minste X slik at $X = \{*\} \oplus X$, mengden av binære trær kan oppfattes som den minste X slik at $X = \{*\} \oplus (X \times X)$, mengden av lister over en mengde A kan oppfattes som den minste X slik at $X = \{*\} \oplus X \times A$. Mengden av binære tall kan enten oppfattes som mengden av endelige lister fra $\{0.1\}$ eller som minste X slik at

$$X = \{*\} \oplus X \oplus X.$$

I alle tilfellene har vi et eller flere *grunnobjekter*, bladene, 0, den tomme sekvensen, det tomme treet eller det tomme binære tallet, og en eller flere *generatorer*, tellbar forgrening, etterfølger, paring, tilføying av element, utvidelse med 0 eller 1.

Vi skal se på tolkningen av de tellbare trærne i System F, og drøfte hvorfor dette gir en forstandig tolkning. For en mer systematisk forståelse av sammenhengen mellom induktive definisjoner og System F henvises leseren til litteraturen.

La U være en type. La

$$Tre(U) = \prod X.((U \to X), (NAT \to X) \to X) \to X).$$

Her skal vi legge merke til at i matriks er det plass til to argumenter, en for en funksjon som avbilder U inn i X og en som til hver følge av elementer i X gir oss et element i X. Dette svarer til de to (i generalisert forstand) generatorene for de velfunderte trærne, den første genererer bladene fra U og den andre genererer et nytt tre fra en følge av gamle trær.
Hvis t er en term av type U kan vi finne en term $tre(t)$ av type $Tre(U)$ avhengig av t som følger:

$$tre(t) = \Lambda X.\lambda x^{U \to X} \lambda y^{NAT \to X}.xt$$

Hvis s er en term av type $NAT \to Tre(U)$ finner vi en term $tre(s)$ av type $Tre(U)$ som følger:

$$tre(s) = \Lambda X.\lambda x^{U \to X} \lambda y^{NAT \to X}.ys$$

Ettersom vi nå beveger oss mot en bratt bakke i sidekant av den retningen det er meningen at dette kompendiet skal ta, stopper vi her, og overlater det til den interesserte leseren å studere representasjon av induktivetyper på egen hånd.

9.3 Oppgaver

Oppgave 9.1 Dette er mer å betrakte som en liten prosjektoppgave enn som en øvelsesoppgave:
Vis hvordan de andre induktivt definerte mengdene vi så på i avsnitt 9.2 kan realiseres i System F.

Bibliografi

[1] J.-Y. Girard, Y. Lafont og P. Taylor: *Proofs and Types*, Cambridge University Press 1989.

[2] D. Normann: *λ-kalkyle på 90 minutter*, Elektronisk sirkulert forelesningsmanuskript 1998

[3] D. Scott, *A type-theoretical alternative to ISWIM, CUCH, OWHY*, Unpublished notes, Oxford (1969).

[4] D. Scott, *A type-theoretical alternative to ISWIM, CUCH, OWHY*, Theoretical Computer Science 121 (1993), 411 - 440.

[5] V. Stoltenberg-Hansen, I. Lindström og E.R.Griffor: *Mathematical Theory of Domains*, Cambridge University Press 1994.

Register