

Dennis Fredriksen

MOSART
Teknisk dokumentasjon

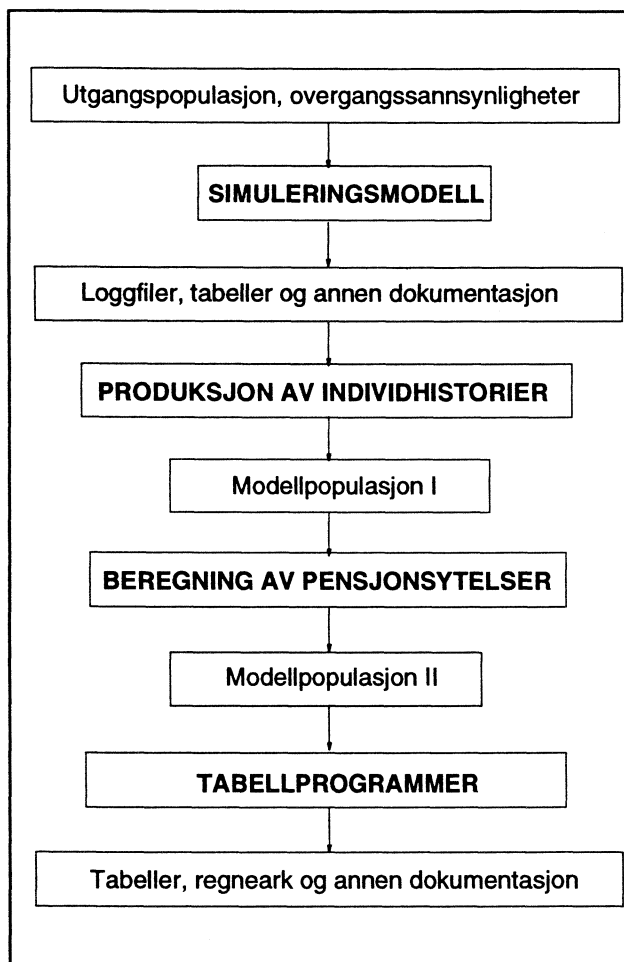
Notater

Innhold

1. Innledning	3
2. Katalogoversikt	4
3. Kjøring av modellen	4
4. Simuleringsprogram	8
5. Bearbeiding av loggfil	9
6. Beregning av trygdeytelser	11
7. Standard tabellprogram	13
Vedlegg	16
A. Kjørefil	16
B. Simuleringsparametre	17
C. Overgangsrater og annen input	18
D. Resultatfiler	20
E. Modellpopulasjon/filbeskrivelse	22
F. Dokumentasjon av MOSART	26

1. Innledning

MOSART er en demografisk basert mikrosimuleringsmodell for skolegang, trygd og arbeidstilbud. Idéen bak modellen er å trekke et utvalg av befolkningen, for deretter å simulere det videre livsløpet for hvert enkelt individ i dette utvalget. I denne versjonen av modellen simuleres begivenheter som inn- og utvandring, død, fødsler, bevegelser i ekteskapeleg status og valg av ektefelle, skolegang og innvirkning på utdanningsnivå, trygdestatus og arbeidstilbud med tilhørende arbeidsinntekter. Etterhvert som simuleringen går framover legges nye individer til utgangspopulasjonen i form av innvandrere og nye kull 16-åring. Resultatet av simuleringen blir en modellpopulasjon med individhistorier, som gitt forutsetningene bak simuleringen vil være representativ for befolkningen i simuleringsperioden. En tidligere modellversjon er dokumentert i Andreassen et al (1993), mens denne modellversjonen er dokumentert i Fredriksen og Spurkland (1993). Formålet med notatet er både å være en veiledning ved kjøring av modellen og en veiledning/ oppslagsverk for å kunne gjøre endringer i programmene. Bruk av notatet/modellen forutsetter visse grunnkunnskaper i bruk av arbeidsstasjon/UNIX, og helst også SIMULA og SAS. Nedenfor følger en oversikt over modellen og de programmene som inngår, og figur 1 gir en oversikt over MOSART.



Figur 1. Oversikt over MOSART

Forut for kjøringen av modellen ligger en mengde arbeid med tilrettelegging av utgangspopulasjonen (utvalget modellen starter med) og overgangssannsynligheter for de ulike begivenhetene (fødsler, giftemål, ...). I figur 1 tilsvarer dette alt ovenfor SIMULERINGSMODELL. Utgangspopulasjonen og overgangssannsynlighetene finnes dokumentert i Fredriksen (1992A), (1992B) og (1993) og Fredriksen og Spurkland (1993). Selve kjøringen av modellen omfatter alt det som kommer fra SIMULERINGSMODELL og nedover i figuren. Store bokstaver representerer her edb-programmer og små bokstaver representerer simuleringsresultater. Ved hjelp av en *kjørefil* kalles de ulike *programfilene* opp etter tur, og i tilknytning til disse finnes det *styrefiler* som angir verdier på parametre og navn på inputfiler (uten at man behøver å endre programmene). Kapittel 2 gir en oversikt over arkivsystemet (katalogtre) i modellen, mens kapittel 3 går nærmere inn på hvordan en simulering rent faktisk skal gjennomføres.

Første del av modellen omfatter *simuleringsmodellen*, og denne henter inn en utgangspopulasjon og overgangssannsynligheter, og simulerer det videre livsløpet for hvert enkelt individ i dette utvalget.

Simuleringsmodellen er skrevet i programmeringsspråket SIMULA, og simuleringsmodellen er nærmere omtalt i kapittel 4. I tilknytning til simuleringsmodellen finnes to styrefiler som bestemmer parameterverdier og filnavn på inputfiler. Simuleringsmodellen genererer to loggfiler med en record for hver begivenhet som har inntruffet i simuleringen. Resten av modellen går ut på å bearbeide disse loggfilene slik at de gir nyttig informasjon. Bearbeidingen av loggfilene er skrevet i programmeringsspråket SAS, og disse SAS-programmene har en felles styrefil som setter verdien på viktige parametre. I første omgang *produseres individhistorier*, og dette gjøres ved hjelp av et SAS-program (se kapittel 5) som omgjør loggfilen til et datasett med en observasjon for hvert individ for hvert år (med status for dette individet dette året). Denne kalles modellpopulasjon I i figur 1. Videre *beregnes pensjonsytelser* for personer i denne modellpopulasjonen, og dette gjøres også i et SAS-program (se kapittel 6), og resultatet blir modellpopulasjon II. Tilslutt lages det *tabeller* med aggregerte tall fra modellpopulasjonen, også ved hjelp av et SAS-program (se kapittel 7).

2. Katalogoversikt

Alle input-filer og EDB-programmene som inngår i driften av MOSART er samlet på katalogen /DATA1/MOSART tilknyttet arbeidsstasjonen Neumann. Når det refereres til kataloger i resten av notatet er dette kataloger under /DATA1/MOSART (med mindre annet sies).

- BIN:** Denne katalogen inneholder filer som brukes for å gjennomføre en simulering, og dette gjelder spesielt filer som kaller opp sekvenser av programmer.
- DOK:** Ikke i bruk, omfatter dokumentasjon av simulaprogrammet skrevet i tex.
- FIL:** Denne katalogen inneholder programmer/filer som brukes til å bearbeide output fra simuleringsprogrammet.
- INPUT:** Denne katalogen omfatter all input til simuleringen, det vil si utgangspopulasjonen og alle overgangsrater.
- PROG:** Denne katalogen inneholder simulaprogrammene som brukes for å gjennomføre en simulering.
- RES:** Denne katalogen omfatter simuleringsresultater hvor hver simulering har sin egen underkatalog, og hvor grupper av simuleringer kan være samlet i egne mellomkataloger.
- RES\TMP:** Ved hver simulering legges alle resultater midlertidig ut på arbeidsområdet RES\TMP, før de flyttes over til en permanent underkatalog. Dette er en nyttig konstruksjon fordi man da slipper å endre katalognavn i de programmene som brukes.
- SORTTMP:**Arbeidsområde som brukes ved store sorteringer.

3. Kjøring av modellen

Kapittel 4 beskriver nærmere de viktigste grepene som er nødvendig for å kunne gjennomføre en simulering med MOSART. Modellen ligger på disken /DATA1 i tilknytning til arbeidsstasjonen NEUMANN. Dette notatet forutsetter at brukeren har en viss kjennskap til UNIX (operativsystemet på arbeidsstasjonen) og arbeidsstasjoner. Det er imidlertid lite som kreves for å kjøre modellen, og de viktigste resultatfilene kan lett flyttes over på PC. Modellen er videre skrevet i SIMULA og SAS, og det er også en fordel å ha en viss kjennskap til disse programspråkene. Kapitlet innledes med en forklaring av begrepet kjørefil (3.1), før beskrivelsen av hvordan modellen kjøres (3.2). Deretter følger en beskrivelse av kjørefilen for en full simulering (3.3) og de parametrene som er mest vanlige å endre (3.4). Tilslutt er det en kort omtale av andre måter å kjøre modellen på (3.6).

3.1. Kjørefil

Normalt vil en kjøring med MOSART kalle opp flere programmer på arbeidsstasjonen og således bestå av en serie UNIX-kommandoer. Normalt vil man ønske å kjøre disse programmene suksessivt ved hjelp av *en* kommando. Dette gjøres mest effektivt ved å samle jobbene i en såkalt *kjørefil*, som inneholder en linje for hver UNIX-kommando og som er eksekverbar¹ (tilsvarer bat-filer under DOS). De mest brukte kombinasjonene av programmer har ferdig tilrettelagte kjørefiler, og disse ligger samlet på katalogen BIN. Spesielt vil det finnes en kjørefil som genererer en full simulering med beregning av pensjonsytelser og uttak av tabeller, og p.t. heter denne 'standard'. Avsnittene 3.2-3.5 tar utgangspunkt i denne kjørefilen, og andre måter å kjøre modellen på kan avledes fra denne.

3.2. Standard kjøring av modellen

Etter at kjørefil og styrefiler for input er redigert (se 3.4), går man til arbeidsområdet RES/TMP og skriver navnet på kjørefilen man skal bruke, for eksempel 'standard', etterfulgt av en & (som gjør at jobben går i batch²) og RETURN:

```
</data1/mosart/res/tmp> standard & [RETURN]
```

Dette forutsetter at arbeidsstasjonen for din bruker er riktig satt opp, det vil si at maskinen er satt opp til å lete etter kjørefiler i katalogen BIN (overlates til noen med gode kunnskaper i UNIX). Er ikke dette tilfelle kan man istedet skrive fullt filnavn:

```
</data1/mosart/res/tmp> /data1/mosart/bin/standard & [RETURN]
```

Ved å stå i RES/TMP er man sikret at alle jobbrapporter med mere faktisk havner på ønsket sted (RES/TMP), og i neste omgang blir alle resultatfilene flyttet over til en permanent underkatalog som

¹ Enhver fil kan gjøres eksekverbar ved å endre riktig fil-attributt, og dette gjøres ved kommandoen *chmod u+x filnavn* (gitt at man eier kjørefilen).

² I motsetning til interaktivt som gjør at maskinen 'beholder' tastaturet til jobben er ferdig.

er spesifisert i kjørefilen 'standard'.

NB! Navnet på kjøringen er siste ledd i navnet på katalogen som simuleringen endelig skal lagres på (/DATA1/MOSART/RES/navn). Dette navnet spesifiseres øverst på kjørefilen, og det er viktig at dette navnet ikke er i bruk fra før. Dette er den eneste endringen man trenger å gjøre hvis man ønsker å gjenta simuleringen med de samme forutsetningene.

NB! Før en kjøring kan iverksettes må man kontrollere at det er nok plass på disken /DATA1, og en full simulering med beregning av pensjonsytelser og uttak av tabeller krever trolig 1000 Mb. Simuleringsmodellen alene krever bare 200-300 Mb.

NB! Når en kjøring er gjennomført bør man raskt ta stilling til om individhistoriefilen skal **overføres** til tape, slettes eller eventuelt lagres midlertidig på harddisken (det er begrenset hvor mange individhistoriefiler det er plass til). Plassforbruket kan reduseres ved å komprimere (compress) filene.

3.3. Standard kjørefil

Standard kjørefil som genererer en full simulering heter p.t. 'standard' og ligger på katalogen BIN, og en utskrift av filen finnes i vedlegg A. Forklaringen nedenfor tar utgangspunkt i denne.

Kjørefilen innledes med noe "grums" (!/bin/sh) som sikrer at UNIX-kommandoene blir riktig iverksatt. Deretter kommer navnet på underkatalogen som resultatene tilslutt skal legges ned på (res_dir=**navn**). Standardversjonen tester om denne katalogen allerede finnes før den oppretter en hjelpefil (echo...) som brukes som overskrift i alle tabeller. Hensiktsmessig er denne overskriften satt til katalognavnet, fordi man da lett kan finne tilbake til dokumentasjonen av kjøringen.

Så iverksettes simuleringprogrammet (/data1/mosart/prog/**filnavn** -m=16 > /dev/null), hvor filnavn angir modellversjon og (**viktig**) -m=16 sikrer tilgang på tilstrekkelig internminne på arbeidsstasjonen. Dette etterfølges av sammenslåing og sortering av loggfilen (sort -T ...). De to neste linjene fjerner de delene av loggfilen som ikke lenger er nødvendig (rm / ...).

Videre kopieres filen styrefilen for bearbeiding av output ('sas-opsjoner') over til arbeidskatalogen, da dette er den enkleste måten å dokumentere valget av disse opsjonene. Så følger bearbeidingen av loggfilen (sas /dat ... /loggfil) før loggfilen fjernes (rm /dat ... /loggsort). Etter dette følger beregningen av trygdeytelser (sas /dat ... /trygd) og uttak av standard tabeller (sas /dat ... /tabell). Tilslutt opprettes katalogen som resultatene skal lagres på (mkdir ...), og alle filer på arbeidsområdet flyttes over hit (mv ...). De to mellomliggende linjene sikrer at alle på MOSART-gruppa får skrevetillatelse til filene (chgrp ... og chmod ...) (Bør endres?).

NB! Kjørefilen redigeres lettest ved å kommentere ut de linjene som ikke skal brukes, og dette gjøres ved å skrive # på begynnelsen av linja.

3.4. Endring av parametre

Endringer i simuleringer kan stort sett foretas ved endre kjørefilen, eller styrefilene med simuleringsparametre ('paramfil'), input ('tabellfil') og styrefil for bearbeiding av output ('sas-opsjoner'). Samtlige av disse filene ligger på katalogen BIN, og filnavnene er faste i den forstand at programmene leser fra samme sted hver gang. Ønsker man å lagre ulike versjoner av 'paramfil'/'tabellfil'/'sas-opsjoner', må dette gjøres under andre navn. Skal disse filene brukes i en simulering må de kopieres inn på de faste navnene 'paramfil'/'tabellfil'/'sas-opsjoner'. Avsnitt 3.4 omtaler de endringene som er mest vanlige, og en detaljert beskrivelse av kjørefilen/styrefilene finnes i vedlegg A-C.

Navn på simulering

Hver simulering tildeles et unikt navn som kun trenger å tilfredstille de krav som stilles til katalognavn under UNIX. Dette navnet skrives inn på toppen av kjørefilen (resdir=*navn*), og da skal all output fra simuleringen tilslutt havne på katalogen '/DATA1/MOSART/RES/*navn*'.

'Kjøreregler' for navn er ennå ikke vedtatt, og eneste restriksjon foreløpig er at kjøringen deles i grupper hvor to-fire første bokstaver angir gruppen, etterfulgt av en underscore (_), og deretter navn på varianten. Hensiktsmessig kan da en slik gruppe samles på en mellomkatalog '/DATA1/MOSART/RES/gruppe', hvor gruppe er de samme to-fire bokstavene i simuleringnavnene. Dato trenger ikke inngå i navnet, da dette likevel går igjen på all output.

Random-seed

Startverdien for rekken med tilfeldige tall i simuleringsmodellen er viktig, og ved normale kjøringer bør denne settes til 0. Dette gjøres i filen 'paramfil', se vedlegg B. Ved uttesting av modellen kan dette settes til en fast verdi, for å teste at modellen ikke endres uten at man er klar over det.

Utgangspopulasjon

Her må utvalgsstørrelsen endres på 'paramfil', utgangspopulasjonen på 'tabellfil' og oppblåsningsfaktoren og den historiske del av modellpopulasjonen på 'sas-opsjoner'.

Simuleringshorisont

Her må simuleringsslutt endres på både 'paramfil' og på 'sas-opsjoner'.

3.5. Forbruk av regnekraft

Forbruket av regnekraft i MOSART er såvidt stort at det bør vektlegges ved bruk av modellen, spesielt fordi det kan ta lang tid å få ut de resultatene man ønsker. Nedenfor følger en oversikt over hvor mye cpu-tid de ulike delene bruker når simuleringshorisonten er 2050 og utvalget er på 1 prosent av befolkningen. Er man alene om å bruke maskinen vil dette tilsvare reell tid. Eneste unntaket er sorteringer som normalt tar vesentlig lengre tid enn forbruket av cpu-tid selv når man er alene på

maskinen. Hvor lang tid sorteringen av loggfilene tar blir ikke dokumentert av maskinen, og anslaget på tiden det tar å tilrettelegge individhistoriene er derfor ekstra omtrentlig (sortering av loggfil+loggfil-program).

Simuleringsprogrammet - omtrent 1t55m.

Initiering av modell og overgangssannsynligheter - under 10 sekunder.

Initiering av utgangspopulasjon - omtrent 6-9 minutter.

Simulering av et år - omtrent 2 minutter.

Tilrettelegging av individhistorier - omtrent 40 minutter.

Beregning av trygdeytelser - omtrent 1 time.

Uttak av tabeller - omtrent 1 time.

En tabell med hele befolkningen - omtrent 5-10 minutter.

Total tid - omtrent 4,5 time.

Det er ikke lagt stor vekt på forbruket av regnekraft i programmeringen (relativt til det å gjøre programmene oversiktlige). Valg av andre programpakker enn SAS og mer komplisert programmering av simuleringsmodellen og beregningen av trygdeytelser kan redusere forbruket av regnekraft betydelig. Imidlertid går EDB-teknologien raskt framover, og dette aspektet bør tillegges liten vekt. Disponerer man maskinen alene, skal en simulering fint kunne gjennomføres på en natt, i det minste hvis det er god plass på diskene.

3.6. Andre måter å kjøre modellen på

Det vil være en rekke måter å kjøre modellen på, og hvorvidt man skal lage ferdige tilrettelagte kjøre-filer eller redigere standardfilen blir en smaksak.

En annen viktig måte å kjøre modellen på er å bare endre beregningsreglene for pensjon, og denne varianten har en egen kjøre-fil, 'trygd_prov'. Denne kjøringen innledes med at et SAS-program ('stripp' under FIL) først rensker en eksisterende modellpopulasjon for pensjonsytelser (denne må da ligge klar på '/RES/TMP'), før beregningsrutinene for pensjon og (en spesialvariant) av tabellprogrammet kan startes.

Ønsker man å kjøre uten beregning av pensjonsytelser (tar mye regnekraft), finnes det et tilhørende tabellprogram som hopper over tabellene med pensjoner ('tabell2'). Ønsker man å utelate de historiske delene av individhistoriene, finnes det også program for produksjon av individhistorier som utelater dette ('loggfil2').

4. Simuleringsprogram

Denne delen av modellen blir bare foreløpig og overfladisk dokumentert i dette notatet. Med god innsikt i modellen er SIMULA et relativt *lett*est programspråk, spesielt hvis man legger vekt på intuitive navn på prosedyre, variabler mm. Forhåpentligvis har vi klart dette i tilstrekkelig grad, slik at

dokumentasjonen av modellen og programmet i seg selv gir god nok dokumentasjon. Nedenfor følger noen føringer på hvordan det videre arbeidet bør legges opp.

Intuitive navn

Det bør legges STOR VEKT på å bruke intuitive navn i programmeringen, og dette gjelder både prosedyrenavn/globale variable og lokale hjelpevariable. Førstnevnte er viktig for å få oversikten over modellen (og dette ansvaret påligger den som 'spesifiserer' modellen), sistnevnte er viktig for at hver enkelt prosedyre skal være leselig (og dette er programmerens ansvar). Med unntak av simuleringsprosedyrer for demografi og skolegang, er dette p.t. i stor grad ryddet opp i.

Overskrifter

Det er lagt inn overskrifter i programmet og det gjør det lettere å søke seg fram til de delene av programmene man ønsker å se på.

Kommentarer

Lay-out bør tillegges en del vekt, spesielt fordi rotete kommentarer kan gjøre programmet mer uoversiktlig. Spesielt vil kommentarer inne i en prosedyre gjøre prosedyren LENGRE og dermed mer uoversiktlig. Man bør derfor tilstrebe at kommentarene samles i BOKSER FORUT for de relevante programlinjene/prosedyrene. I den grad det er nødvendig kan det legges inn hjelpekommentarer inne i prosedyren (i seg selv et hint om at prosedyren bør deles opp). Dette er det også ryddet opp i, med unntak av demografi&skolegang. Kommentarene kan med hell gjøres mer omfattende, og spesielt mangler mange kommentarer på hvor de enkelte prosedyrene blir brukt (den andre veien er mindre viktig, fordi dette gir seg selv).

5. Bearbeiding av loggfil

Hovedresultatet fra simuleringen er to loggfiler som omfatter alle begivenheter som inntreffer i simuleringen. For at denne informasjonen skal bli tilgjengelig må de bearbeides, og i første omgang må de slås sammen og sorteres på id.nummer, simuleringsår (og type melding). Dette gjøres med en enkel UNIX-kommando fra kjørefilen, og siden denne ikke må endres (id.nummer, sim.år og type melding har faste plasser), går vi ikke nærmere inn på den her. I neste omgang bearbeides den tilrettelagte loggfila av et SAS-program, og resultatet blir tre SAS-datasett som er relativt enkle å bruke til utskrift av livshistorier og produksjon av tabeller.

Standardversjonen av programmet heter 'loggfil' og ligger på katalogen FIL, og det er denne som omtales her. I tillegg finnes en enklere variant ('loggfil2') som ikke tar med de historiske delene av individshistoriene. Resten av kapitlet kan med hell leses i sammenheng med en utskrift av programmet.

Viktige opsjoner/parametre i programmet:

- Historisk del av individhistoriene.
- Sluttår i simuleringen.

Begge disse parametrene hentes fra styrefilen 'sas-opsjoner', mens andre parametre som inngår i 'loggfil' stort sett er faste fra gang til gang (navn på loggfil, arbeidsområde, startår, ...).

Inndata/hjelpeprogrammer

- Sammenslått og sortert loggfil.
- Historisk del av individhistoriene.

Utdatasett (navn i parentes):

- (mp_ind) Individhistoriefil.
- (mp_ekt) Ekteskapsmeldinger.
- (mp_fods) Fødselsmeldinger.

Disse tre filene er beskrevet i vedlegg E, men merk at pensjonsrettigheter og -ytelser ikke inngår på den versjonen som genereres av 'loggfil' (dette gjøres senere). Spesielt vil individhistoriefilen inneholde en observasjon for hvert individ for hvert år dette individet er med i simuleringen. Ekteskapsmeldingene og fødselsmeldingene inneholder (p.t.) ikke de historiske opplysningene om disse forholdene.

Programsstruktur

Programmet har standard innledning hvor programmet henter inn opsjoner og filreferanser fra styrefilen 'sas-opsjoner', etterfulgt av filreferanse til loggfilen (fast) og følgende hjelpemakroer:

- Makro *null* nullstiller variable når nytt individ hentes inn.
- Makro *su_ekt* skriver ut ekteskapsmelding.
- Makro *u_n_aar* skriver ut status for ett år for ett individ til individhistoriefil.
- Makro *u_slutt* skriver ut siste år for hvert individ til individhistoriefilen.

Selve programmet er bygd opp i tre deler, hvor del 1 leser inn og bearbeider loggfilen, del 2 slår sammen de historiske og de simulerte delene av individhistoriene og del 3 tilrettelegger ekteskapsmeldingene.

Del 1. Bearbeiding av loggfil

Del 1 omfatter et datasteg som starter med standard innledning (inndata, utdata, variabeldefinisjoner). Utdatasett omfatter individhistoriefil, ekteskapsmeldinger og fødselsmeldinger. Første programbit leser inn id.nummer og sim.år for hver ny melding, og tester om meldingen gjelder et nytt individ eller nytt år (et individ kan ha flere begivenheter/meldinger i løpet av ett år). Har vi nytt individ/sim.år skriver programmet ut 'forrige' persons/års status til individhistoriefilen. Deretter fortsetter programmet med å lese inn den faste delen av meldingen, og bearbeider denne avhengig av hvilken type det er. Dette kan omfatte startverdimelding, dødsfall, utvandring, ekteskap, utdanning, trygd, AKU eller inntekt. Spesielt vil startverdimelding generere et nytt individ. Til slutt testes det om vi har siste observasjon, og siste person skrives da eventuelt ut til individhistoriefilen (normalt skjer dette når neste person dukker opp).

Del 2. Sammenslåing av historisk og simulert del

Del 2 omfatter et datasteg som *fletter* (merge) den historiske delen og den simulerte delen av individhistoriene etter person og årstall, til et datasett. Den simulerte delen kommer sist, slik at opplysningene fra den simulerte delen dominerer over den historiske delen (kun aktuelt i startåret, 1989). Noen data hentes like vel over fra den historiske delen også i 1989 (endringsvariable for ekteskap, utdanning og trygd).

Del 3. Bearbeiding av ekteskapsmeldinger

I del 3 sorteres menns og kvinners ekteskapsmeldinger på de to ektefellers id.nummer og dato for ekteskapsinngåelse, før den mannlige og kvinnlige delen slås sammen. Inkonsistenser gir opphav til feilmeldinger.

6. Beregning av trygdeytelser

Denne jobben beregner de pensjonsytelser (alders- uføre- og etterlattetrygd) som hvert individ har krav på i følge simuleringsresultatene (trygdestatus, pensjonsrettigheter) og de regler og satser som er lagt til grunn. Programmet henter inn en individhistoriefil, gjør de nødvendige beregningene og legger resultatet ut på den samme individhistoriefilen. Programmet omfatter flere store sorteringer, og er en vesentlig grunn til å begrense størrelsen på hver enkelt utgangspopulasjon. Resten av kapitlet kan med hell leses i sammenheng med en utskrift av programmet.

Viktige opsjoner/parametre i programmet:

- Sluttår.
- Startår for beregning av pensjonsytelser.
- Satser; grunnbeløp og særtilllegg.
- Antall år i besteårsregelen.
- Nedre aldersgrense for alderspensjon (67 år).
- Aldersgrense for alderspensjon (70 år).

Inndata/hjelpeprogrammer

- Individhistoriefil.
- Tidsserie med grunnbeløp.
- Tidsserie med særtilllegg.

Utdatasett (navn i parantes):

- (mp_ind) individhistoriefil med pensjonytelser.

Programsstruktur

Programmet har standard innledning hvor det henter inn opsjoner og filreferanser fra styrefilen 'sas-opsjoner', etterfulgt av noen andre opsjoner samt tre hjelpemakroer:

Makro *nedjust*(pensjonspoeng) nedjusterer pensjonspoeng ved overgang til nye beregningsregler i 1992 (gjelder spesielt BUP).

Makro *tillegg*(sluttpoeng, poengår, poengår før 1992, fødselsår, tilleggspensjon) beregner tilleggspensjon på grunnlag av de nevnte argumenter.

Makro *bytt*(x1,x2) bytter verdier på de to variablene x1 og x2.

Selve programmet er delt opp i fire deler, hvor første del/datasteg beregner pensjonspoeng og poengår, del 2 beregner sluttpoengtall, del 3 legger disse opplysningene tilbake på hvert individ og tilslutt beregnes pensjonsytelsene i del 4.

Del 1. Beregning av pensjonspoeng og poengår

Del 1 omfatter et datasteg som starter med standard innledning (inndata, utdata, variabeldefinisjoner), samt innhenting av G-serie. Utdata omfatter tre datasett med henholdsvis poengtall for beregning av sluttpoeng (POENG), tilbakelegging av beregnede pensjonspoeng (POENG2) og tilbakelegging av poengår (P_AAR). Den første programbiten nullstiller (hjelp)variable hvis vi har et nytt individ. Dernest begrenser datasteget seg til de som er i alderen for å opptjene pensjonspoeng (17-69 år).

De som er uføretrygdet eller har dødd samme år blir tildelt et beregnet pensjonspoeng (BUP; beregnet uførepoeng). For nye uføre i historisk del settes dette lik faktisk BUP. For nye uføre i simuleringsperioden og nye døde beregnes BUP som gjennomsnittet av de tre beste pensjonspoengene de siste fem årene *før* begivenheten inntraff (dette gjøres for å kompensere forat BUP ofte gis tilbakevirkende kraft i forhold til når uførepensjonen begynner å løpe). Nye uføre fra og med 1992 tildeles en egen indikator. BUP nedjusteres i 1992, og unge uføre sikres en minste-BUP (3.3, justeres ikke gjennom perioden). For eksisterende uføre settes BUP lik det faktiske i historisk del og lik det beregnede i simuleringsperioden. For alle uføre settes pensjonspoeng lik BUP (forenkling).

For ikke uføre i simuleringsperioden beregnes pensjonspoeng og eventuelt omsorgspoeng deles ut. Videre telles opp poengår og poengår før 1992. Hvert individ skrives ut med en observasjon til henholdsvis datasettet for beregning av sluttpoeng (POENG) og tilbakelegging av pensjonspoeng (POENG2). I tillegg blir nye døde og personer som er uføre i sluttåret skrevet ut til datasettet POENG med pensjonspoeng lik BUP for hvert år fram til pensjonsalder (67 år). Ved siste observasjon for hvert individ blir individet skrevet ut til datasettet for tilbakelegging av poengår (P_AAR).

Del 2. Beregning av sluttpoeng

Del 2 starter med å begrense datasettet (POENG) til de observasjoner hvor pensjonspoengene er større enn null. Deretter sorteres datasettet på person og fallende pensjonspoeng. Bestårsregelen kan da i neste datasteg lett anvendes ved å bruke de tyve første observasjonene på hvert individ til beregning av sluttpoeng. Dette er trolig en effektiv måte å plukke ut de tyve beste poengårene, da sorterings-

programmet utnytter at datasettet allerede er sortert på person, og fordi det å plukke ut de 'n' beste årene i realiteten vil tilsvare en sortering og vi da kan utnytte 'profesjonelle' sorteringsprogrammer. Datasettet POENG reduseres her til en observasjon pr person med opplysninger om sluttpoengtallet.

Del 3. Tilbakelegging av sluttpoeng og poengår

Del 3 starter med at POENG (sluttpoeng) og P_AAR (poengår) flettes sammen etter person (en obs./person), før de flettes tilbake på individhistoriene ('forflettingen' gir en betydelig besparelse av cpu-tid). Spesielt får de som ikke har fått beregnet sluttpoengtall (ingen pensjonspoeng) satt dette sluttpoenget til null (unngår plagsomme missing-verdier). Deretter flettes POENG2 (beregnete pensjonspoeng) tilbake på individhistoriene etter person og sim.år, og samtidig opprettes et datasett EKT med id.nummer for personen og ektefellen for hvert enkelt sim.år. Resten av del 3 omfatter innhenting av ektefelles sluttpoeng, poengår, fødselsår og trygdestatus. Denne biten står for halvparten av forbruket av cpu-tid ved beregning av pensjonsytelser. Det er visse muligheter for å redusere forbruket av cpu-tid i denne biten, tildels ved å nøye oss med tilnærminger i disse beregningene (ikke gjort).

Først sorteres EKT på ektefelles id.nummer og sim.år, og dette brukes senere til å hente ut utvalget av ektefeller. Dernest hentes et datasett POENG2 med alle personer med de variablene vi trenger for ektefellene, og disse variablene navnes om. Videre brukes EKT til å hente ut ektefellene, og dette sorteres tilbake på personen før det slås sammen med (merge) med individhistoriene.

Del 4. Beregning av pensjonsytelser

Del 4 omfatter et datasteg med standard innledning (utdata, inndata og variabeldefinisjoner), samt innhenting av G-serie og serie med særtillegg. Utdata er individhistoriefilen med pensjonsytelser. Beregningene er begrenset til årene fra og med 1985, det første året vi kjenner trygdestatus skikkelig. Beregningene gjennomføres separat for ulike trygdegrupper (alders-, uføre- og etterlattertrygd, samt ikke trygdemottakere (nullstilles)). De ulike enkeltytelsene beregnes separat (grunnpensjon, særtillegg, tilleggspensjon, etterlattepensjon), før samlet pensjon beregnes. Særtillegget og tilleggspensjonen avkortes mot hverandre, og overlappet gis et eget navn.

7. Standard tabellprogram

Kapittel 8 beskriver produksjonen av tabeller på grunnlag av den tilrettelagte individhistoriefilen. Tabeller som genereres direkte fra simuleringsprogrammet omtales ikke her, men oversikt over alle tabellene/regnearkene som genereres av simuleringsmodellen/tabellprogrammet finnes i vedlegg D om resultatfiler. Rutinene for uttak av tabeller er skrevet i SAS, da dette er et programsspråk som er rimelig lett tilgjengelig for de fleste som skal jobbe med modellen. SAS er derimot lite effektiv med hensyn til regnekraft, og omfanget av tabeller og programspråk bør derfor vurderes etterhvert. Nedenfor følger en beskrivelse av de rutinene som benyttes i tabelluttaket, og denne leses best ved å ha en utskrift av programmene tilgjengelig.

Hovedprogram

Programmet iverksettes som et SAS-program, og programmet ligger på fila 'tabell' på katalogen FIL. I tillegg finnes spesialvarianter av tabellprogrammet som ikke tabellerer pensjonsytelser ('tabell2') og som bare tabellerer pensjonsytelser ('tabell3'). Fila 'tabell' henter inn hjelpe- (.mcr/sup) og aggregeringsrutiner (.agg/ag2/ag3) fra underkatalogen FIL/TABELL_SUP. Disse rutinene kalles etterhvert opp av programmet, først i et datasteg som tilrettelegger dataene og tilslutt i rutiner som aggregerer og skriver resultatene ut til regnearksfiler (.prn) og tabeller. Ønsker man å redusere omfanget av tabeller gjøres det enklest ved å stjerne ut de rutinene man ikke ønsker. Vil man utvide omfanget av tabeller, må nye aggregeringsrutiner skrives først. Nedenfor følger først en beskrivelse av hjelperutinene, deretter en viss generell beskrivelse av aggregeringsrutinene, før hver enkelt aggregeringsrutine får en kort beskrivelse.

Systemet med et felles datasteg i begynnelsen av tabellprogrammet, gjør at det blir mange og store midlertidige filer. Med omfanget av tabeller som brukes nå, kreves det anslagsvis 500 Mb diskplass, og alle datasettene legges derfor ned på arbeidsområdet RESVTMP. En annen organisering av tabellproduksjonene bør derfor vurderes, spesielt hvis man ønsker å utvide antallet tabeller.

Tabell inneholder overbygningen til tabelluttaket, og henter inn og iverksetter prosedyrer i de andre filene. Det finnes varianter av denne fila, og spesielt vil 'tabell2' hoppe over tabellene med pensjonsytelser og 'tabell3' bare ta ut tabeller med pensjoner.

Tabell.sup inneholder (felles) formater for å gjøre tabellene mere leservennlige.

Opprett.mcr inneholder en hjelperutine som tildeler inntektsstatus og arbeidsinntekt for personer hvor dette er tvetydig/mangelfullt i den historiske del av livshistoriene (i stor grad personer i alderen 16 år og 70 til 74 år). I tillegg omfatter rutinen tilrettelegging av en del viktige variable.

Akuvekt.mcr inneholder en hjelperutine som beregner sannsynligheten for at en person med gitt inntektsstatus (og kjønn, alder, trygdestatus og skolegang), skal være i arbeidsstyrken i AKU.

Aggregeringsrutiner

Disse er bygd opp av tre deler, hvor første del inneholder et/flere utdata-sett fra bearbeidingen av individhistoriene (%d·), et sett statements som bearbeider dataene (%s·), og tilslutt et sett rutiner som aggregerer dataene (%a·). Sistnevnte vil normalt først omfatte en aggregeringsrutine og deretter rutiner som genererer regneark. Sistnevnte er litt komplisert fordi det må legges inn rutiner som sikrer at alle elementer blir satt til en tallverdi (SAS lar manglende celler være urepresentert i sine datasett - rimeligvis).

Felles variable (i tillegg til de som er definert på modellpopulasjonen) er alder, ald_gr, aku, aku2, arb_st, hfu og barn. Felles format er \$hfu, \$kjoenn og \$dummy.

Alder.agg aggregerer opp antall personer i ulike grupper etter kjønn og detaljert alder. P.t. omfatter dette befolkningen og arbeidsstyrken, men kan lett utvides. Definerer formatet ald_gr.

Tm_sos.agg aggregerer opp befolkningen og arbeidsstyrken etter kjønn, alder og sosialstatus

(under utdanning, pensjonist). Ekstra variable er sos_stat og format er ald_gs. Dette datasettet danner også utgangspunktet for den tabellen som tas ut med befolkningen og yrkesprosjenter etter kjønn, alder og sosialstatus.

Td_aku.agg aggregerer opp befolkningen 16-74 år etter kjønn og detaljerte kjennetegn yrkesdeltaking. Ekstra variable er aku_tj; j=1,2,3,4,4b,5 og 6, samt aku_p4b.

Td_try.agg aggregerer opp trygdemottakere etter trygdestatus og kjønn med pensjonsytelser fordelt etter type (grunnpensjon, særtillegg, tilleggspensjon).

Td_utd.agg aggregerer opp befolkningen og arbeidsstyrken etter kjønn og detaljerte opplysninger om utdanningsnivå. Format er \$hfu_d.

Td_hush.agg aggregerer opp menn og kvinner etter ekteskapeleg status og barnetall.

Kohort.agg aggregerer opp personer etter kjønn, fødeår og utvalgte variable (p.t. trygd, barnetall og utdanningsnivå). Ekstra variabel er tp.

Synt_koh.agg aggregerer opp fødselskullene født 1974-1983 (rent syntetiske) etter kjønn, alder og utvalgte variable (p.t. ekteskapeleg status, barnetall, utdanningsnivå, trygd og yrkesdeltaking). Ekstra variabler er barnj, ektj, hfuj, trygdj; j=1, ..., samt stud og arb_ti.

Vedlegg

A. Kjørefil

```
#!/bin/sh
```

```
res_dir=test
```

```
if test -s /data1/mosart/res/$res_dir ; then  
  echo "Omraadet eksisterer allerede !!"  
  exit  
fi
```

```
echo "%LET versjon='$res_dir';" > /data1/mosart/res/tmp/overskrift
```

```
/data1/mosart/prog/mosart2_1 -m=16 > /dev/null
```

```
sort -T /data1/mosart/sorttmp -o /data1/mosart/res/tmp/loggsort +0.3 -0.13 /data1/mosart/res/tmp/utfily  
/data1/mosart/res/tmp/utfilstart
```

```
rm /data1/mosart/res/tmp/utfily  
rm /data1/mosart/res/tmp/utfilstart
```

```
cp /data1/mosart/bin/sas-opsjoner /data1/mosart/res/tmp/sas-opsjoner
```

```
sas /data1/mosart/fil/loggfil  
rm /data1/mosart/res/tmp/loggsort  
sas /data1/mosart/fil/trygd  
sas /data1/mosart/fil/tabell
```

```
mkdir /data1/mosart/res/$res_dir  
chgrp -R mosart /data1/mosart/res/$res_dir  
chmod g+ws /data1/mosart/res/$res_dir  
mv /data1/mosart/res/tmp/. /data1/mosart/res/$res_dir
```


B. Simuleringsparametre

Her følger en oversikt over de parametrene som inngår i filen 'paramfil' på katalogen BIN. Alle tall er angitt som heltall og må under innlesningen eventuelt omgjøres til desimaltall.

sim_start angir startåret (med fire sifre) for simuleringen.

sim_slutt angir sluttåret (med fire sifre) for simuleringen.

andel_av_befolkning angir hvor stor andel av befolkningen som utvalget utgjør, angitt i 10 000-deler.

antall_innvandrere_per_år angir antall nettoinnvandrere pr år. Dette tallet må være ikke-negativt (nettoutvandring krever en annen modellspekifisering), og ved eksogen rekruttering er det viktig at tilvekstfilen med nye 16-åringer endres samtidig.

u angir startverdi for random-generator, $u=0$ gjør at systemet velger en tilfeldig startverdi.

opsjon_dod angir om dødeligheten skal økes for uførepensjonister, 0:ikke økes, 1:økes.

D_mu angir hvor mye dødeligheten skal økes for uføre menn, angitt i prosent.

D_ku angir hvor mye dødeligheten skal økes for uføre kvinner, angitt i prosent.

endogen_rek angir måten nye 16-åringer skal rekrutteres på, 0:eksogent, 1:endogent.

pensj_alder_nedre angir nedre aldersgrense for alderspensjon, det vil si 67 år, ved endringer må også tabellen med overgangsrater til alderstrygd endres.

pensj_alder_ovre angir aldersgrensen for alderspensjon, det vil si 70 år, ved endringer må også tabellen med overgangsrater til alderstrygd endres.

inntekts_grense angir hvor grensen for "null" inntekt gå, angitt i hele 1989-kroner.

delta angir startverdien for parameteren som styrer nivået på sannsynligheten for å ha inntekt, må deles på 100.

ant_iter angir maksimalt antall iterasjoner i algoritmen som skal justere nivået på sannsynligheten for å ha inntekt.

konv_krit_justalg angir konvergeringskriteriumet i den samme algoritmen, må deles på 100.

g_z_vekt angir i prosent hvor stor betydning restleddet i simuleringen av nivået på inntekt skal få, 0:intet restledd.

g_z_bytt angir i prosent sannsynligheten for at en person skal bytte restledd, 0:bytter hver gang, 100:bytter aldri.

C. Overgangsruater og annen input

Her følger en oversikt over inputfilene i simuleringsmodellen, nærmere beskrivelse av filene finnes i Rapport 93/7, IN 92/8 og filene selv (de fleste har en kort forklaring på toppen). Katalognavn i overskrift angir hvilken katalog filene er lagret på, og hver linje angir en inputfil hvor første navn angir navnet i simuleringsmodellen, mens det andre navnet angir filnavnet på standardutgaven av inputfilen.

Utgangspopulasjonen (INPUT/UTG_POP)

hfil	i885e1a4.v10	Individhistorier
efil	ektfil0	Ekteskapsmeldinger
ffil	i885e1a3.v00	Fødselsmeldinger

Trygd og arbeidstilbud (INPUT/TABELLER)

ekso	ekso-001.o4s	Eksogen periodeeffekter
ufo_til	ufo-til.o4p	Tilgang av uføre
ufo_fra	ufo-fra.o4s	Avgang av uføre ("friskmeldinger")
ufo_grad	ufo-grad.o4s	Uføregrad
etterl_p	etterl-p.o4s	Overgang til etterlattepensjon
alders_p	alders-p.o4s	Overgang til alderspensjon
akutall	akutall.o4p	Yrkesprosjenter med mere
innt_ssh	innt-ssh.o4p	Sannsynligheten for å ha arbeidsinntekt
innt_niv	innt-niv.o4p	Nivået på arbeidsinntektene
aku_innt	aku-innt.o4s	Kobling av AKU og antall inntektstakere
kaliber	kaliber.o4s	Kalibrering av AKU-tall med mere

Skolegang (INPUT/STUDERE+INPUT/TAB_FAGFELT+INPUT/TAB_FORMAT)

utdikst	b87-ikst.p36	Bli student
utdstus	b87-stus.p36	Studenter/elever med uobserverbar fullføring
utdstuu	b87-stuu.p36	Studenter/elever med observerbar fullføring
utdfort	UTDFORT	Valg av nytt klassetrinn
utdnytt	UTDNYTT	Valg av nytt fag
utdyrke	UTDYRKE	Lærlinger med mere
utdrekr	UTDREKR	Utdanning for nye 16-åringer
formffn	FORMFFN	Fullføringsnivåer
formtni	FORMTNI	Fullføringsnivåer for lærlinger
alddtab	ALDDTAB	Standard aldersformat for utdanning

Tilvekstfiler (INPUT/TAB_REKRUT)

dor	DOR	Dødelighet
tilv	TILV	Antall nye 16-åringer
innvan	INNVAN	Innvandring
utvan	UTVAN	Utvandring

Fødsler (INPUT/FODSLER)

barn1	a89-barn.1st	1.fødsel
barn2	a89-barn.2nd	2.fødsel
barn3	a89-barn.3de	3.fødsel
barn4	a89-barn.4de	4.fødsel og høyere

Ekteskap (INPUT/TAB_EKTSK)

marry1	MARRY1	Endringer i ekteskabelig status
marry2	MARRY2	Ektefelles alder

D. Resultatfiler

Her følger en oversikt over de filer som normalt genereres fra en fullstendig simulering.

Filer som legges inn manuelt

navn.dok I den grad den automatiske dokumentasjonen er utilstrekkelig, bør det manuelt legges inn en tekstfil som dokumenterer kjøringen.

Simuleringen

Kjore_info Dokumentasjon av input med mere
Testfil Fil hvor testmeldinger kan skrives ut
Feil Fil med feilmeldinger

Tabeller fra simuleringen

Tab_01.prn Bevegelser i folkemengden
Tab_02.prn Bevegelser i sosio-økonomisk status (uføre, utdanning)
Tab_03.prn Viktige arbeidsstyrketall
Tab_04.prn Viktige arbeidsstyrketall etter kjønn
Tab_05.prn Befolkningen etter kjønn og alder
Tab_06.prn Arbeidsstyrken etter kjønn og alder
Tab_07.prn Alderspensionister etter kjønn og alder
Tab_08.prn Uførepensionister etter kjønn og alder
Tab_09.prn Etterlattepensionister etter kjønn og alder
Tab_10.prn - reservert AFP-pensionister -
Tab_11.prn Befolkningen etter utdanningsnivå og kjønn

Bearbeiding av output

Overskrift Tekstbit med overskrift i tabell-output med mere
Sas-opsjoner Kopi av styrefil for bearbeiding av output
Utfilstart Startverdimeldinger for utgangspopulasjonen
Utfily Alle andre loggmeldinger
Loggfil Utfilstart og utfily slått sammen og sortert
Loggfil.log Jobbrapport (SAS) fra bearbeiding av loggfil
Mp_ind.ssd01 Individhistorier på SAS-format
Mp_ekt.ssd01 Ekteskapsmeldinger på SAS-format
Mp_fods.ssd01 Fødselsmeldinger på SAS-format
Trygd.log Jobbrapport (SAS) fra beregning av pensjonsytelser
Tabell.log Jobbrapport (SAS) fra tabelluttak
Tabell.lst Utskrift av av noen viktige regneark til tabeller

Regneark med detaljert aldersinndeling

Aldersgrupper: Ialt, 0-6, 7-15, 16-19, 20-24, 25-29, 30-34, 35-39, 40-44, 45-49, 50-54, 55-59, 60-64, 65-66, 67-69, 70-74, 75-79, 80-84, 85+.

Ra_bef.prn Befolkningen etter kjønn og alder
Ra_arbst.prn Arbeidsstyrken etter kjønn og alder

Regneark med moderat aldersinndeling

Aldersgrupper: Ialt, 16-24, 25-55, 55-66, 67-74, 75+.

Rm_sos_b.prn Befolkningen etter kjønn, alder og sosialstatus.
Rm_sos_a.prn Arbeidsstyrken etter kjønn, alder og sosialstatus.
Sosialstatus: Ialt, øvrige (=aktiv), under utdanning, pensjonist.

Regneark uten aldersinndeling

Rd_aku.prn Noen viktige arbeidsstyrketall.
Rd_utd_b.prn Befolkningen etter kjønn og utdanning.
Rd_utd_a.prn Arbeidsstyrken etter kjønn og utdanning.
Rd_try.prn Pensjonister i alt etter kjønn, antall og ytelser.
Rd_ufo.prn Uførepensjonister i alt etter kjønn, antall og ytelser.
Rd_ett.prn Etterlattepensjonister i alt etter kjønn, antall og ytelser.
Rd_ald.prn Alderspensjonister i alt etter kjønn, antall og ytelser.
Rd_hush.prn Menn etter ekteskapelig status og
 kvinner etter ekteskapelig status og antall barn under 16 år.

Fødselskohorter etter fødeår

Rk_try.prn Personer ved alder 70 år, kjønn og opptjente pensjonsrettigheter.
Rk_utd.prn Personer ved alder 60 år, kjønn og utdanningsnivå.
Rk_barn.prn Kvinner ved alder 50 år og antall barn.

Syntetisk kohorter født 1974-1983

Rk_dem.prn Personer etter alder, kjønn og ekt.status+barnetall.
Rk_utd.prn Personer etter alder, kjønn og utdanningsnivå.
Rs_try.prn Personer etter alder, kjønn og trygdestatus+ytelser.
Rs_arb.prn Personer etter alder, kjønn og yrkesdeltaking.

E. Modellpopulasjon/filbeskrivelse

Her følger en beskrivelse av individhistoriefilen, ekteskapsmeldingene, fødselsmeldingene og noen av de viktigste variablene (merket med stjerne).

E.1. Individhistoriefil

Individ/record

DNUMMER	4	Identifikasjonsnummer.
DUBLETT	Å1	· Tilhørighet til utvalget.
VITAL	Å1	· Registreringsstatus.
KJONN	Å1	Kjønn.
FODS_AAR	3	Fødselsår.
SIM_AAR	3	Årstall for gjeldende opplysninger.

Utdanning

IGUART	Å2	Fagfelt for igangværende utdanning.
IGUKLT	Å2	Klassetrinn for igangværende utdanning.
HFUART	Å2	Fagfelt for høyeste fullførte utdanning.
HFUKLT	Å2	Klassetrinn for høyeste fullførte utdanning.
HFU_AAR	3	Årstall for når HFU ble fullført.
E_UTD	Å1	· Endringsindikator for utdanning.

Ekteskap/barn

EKT_STAT	Å1	· Ekteskapeleg status.
E_IDNR	4	Ektefelles identifikasjonsnummer.
E_EKT	Å1	· Endring i ekteskapeleg status.
ANT_BARN	3	Antall barn (for kvinner).
YNGST1-3	3	Alder på yngste, nestyngste og tredje yngste barn.
ALDER_F1	3	Alder ved første fødsel.

Arbeid/inntekt

AKU_P1	3	Forventet ssh for å være i arbeid.
AKU_P2	3	Forventet ssh for å være midl.frav., syk.
AKU_P3	3	Forventet ssh for å være midl.frav., ferie.
AKU_P4	3	Forventet ssh for å være midl.frav., annet.
AKU_P5	3	Forventet ssh for å være i arbeidssøker uten arb.inntekt.
AKU_P6	3	Forventet ssh for å være utenfor arbeidsstyrken.
AKU_TI	3	Forventet gj.sn. arbeidstid/uke, gitt arbeid.
I_STAT	Å1	· Yrkesstatus v/inntekt.
SENS	Å1	· Hvor opplysninger om i_stat er ufullstendig.

INNTEKT	4	Inntekt i 1989-kroner.
PP	3	Pensjonspoeng.
BUP	3	Beregnet uførepoeng.

Trygdestatus

TRY_STAT	Å1	· Trygdestatus.
TRY_GRAD	3	Uføregrad/trygdegrad.
E_TRY	Å1	· Endring i trygdestatus.

Pensjonsrettigheter

SP	3	Sluttpoengtall
P_AAR	3	Poengår.
P_AAR_F	3	Poengår før 92'-reformen.
IND_UF92	Å1	Ufør før 92'-reformen (0:ja, 1:nei).

Ektefelles trygdestatus og -rettigheter

E_TRY_ST	Å1	'Siste' ektefelles trygdestatus.
E_SP	3	'Siste' ektefelles sluttpoengtall.
E_F_AAR	3	'Siste' ektefelles fødselsår.
E_P_AAR	3	'Siste' ektefelles poengår.
E_P_AF	3	'Siste' ektefelles poengår før 92'-reformen.

Pensjonbeløp

PENSJON	4	Samlet pensjon.
P_GRUNN	4	Grunnpensjon.
P_KOMP	4	Kompensasjonstillegg.
P_ST	4	Rent sært tillegg.
P_ST2	4	Sært tillegg avkortet mot tilleggspensjon.
P_TIL	4	Tilleggspensjon utover sært tillegg.
P_ET	4	Etterlattepensjon utover sært tillegg og tilleggspensjon.

E.2. Ekteskapsmeldinger

K_IDNR	4	Kvinnens id.nummer.
M_IDNR	4	Mannens id.nummer.
E_DATO	D9	Dato for inngåelse av ekteskap.
OPP_DATO	D9	Dato for oppløsning av ekteskap.
OPP_GRUNN	Å1	Årsak til oppløsning.

E.3. Fødselsmeldinger

MOR_ID	4	Morens identifikasjonsnummer.
B_FAAR	3	Barnets fødeår.
B_FLAND	Å1	Barnets fødeland.

E.4. Variabelliste

Dublett

- 1: Med i utvalget.
- 2: Supplerende person.

Vital

- 1: Bosatt i Norge.
- 2: Død.
- 3: Utvandret.
- 4: Innvandrer neste år.
- 5: Antatt bosatt i utlandet (ikke i bruk her).
- 6: Antatt bosatt i Norge.
- U: Ny 15-åring fra historisk del.
- B: Ny 15-åring fra simuleringen.

E_utd

Person som ikke er under utdanning, og:

- 0: Ingen endring.
- 1: Begynner å studere.
- 7: Fullfører en (yrkes)utdanning.
Elev eller student, med IGU som leder fram til ny HFU, som:
 - 4: Fortsetter IGU.
 - 5: Fullfører IGU og fortsetter å studere.
 - 6: Fullfører IGU og slutter å studere.
- 2: Avbryter IGU, begynner på nytt fag.
- 3: Avbryter IGU, slutter å studere.
Elev eller student, med IGU som ikke leder fram til ny HFU, som:
 - C: Fortsetter IGU.
 - A: Forlater IGU, begynner på nytt fag.
 - B: Forlater IGU, slutter å studere.

Ekt_stat

- 1: Ugift.
- 2: Gift.
- 3: Enke/enkemann.

4: Skilt.

E_ekt

0: Ingen endring.
Ellers ny ekteskapelig status.

I_stat

I arbeid:

J: Inntekt større eller lik ett tusen 1989-kroner i løpet av året.
N: Inntekt mindre enn ett tusen 1989-kroner i løpet av året.
Klassifisering tar utgangspunkt i fjorårets, årets og neste års inntekt.
1: Er i arbeid (JJJ).
2: Begynner å arbeide i løpet av året (NJJ).
3: Slutter å arbeide i løpet av året (JJN).
4: Utenfor arbeid (.N.).
5: Begynner og slutter arbeide i løpet av året (NJN).

Try_stat

1: Ikke trygdemottaker (av 3-5).
3: Uføretrygdet.
4: Alderstrygdet.
5: Etterlattetrygdet.

E_try

Blank; ingen endring.

A: Blir ufør.
B: Blir alderstrygdet.
C: Blir etterlattetrygdet.
D: Går fra uførhet til alderstrygd.
E: Forlater uføretrygd.
F: Dør som uføretrygdet.
G: Går fra etterlattetrygd til alderstrygd.
H: Går fra etterlattetrygd til uføretrygd.
I: Forlater etterlattetrygd.
J: Dør som etterlattetrygdet.
K: Forlater alderstrygd.
L: Dør som alderstrygdet.

F. Dokumentasjon av MOSART

NB! MOSART-T var et tidligere arbeidsnavn på MOSART 2.

NB! Vi har p.t. i stor grad gått over til å kalle MOSART 2 for MOSART.

Leif Andreassen, Truls Andreassen, Dennis Fredriksen, Gina Spurkland og Yngve Vogt (1993): "Framskrivning av arbeidsstyrke og utdanning", *Rapport 93/6*, Statistisk sentralbyrå.

Dennis Fredriksen (1991): "Forprosjektrapport" og "Behov for bruttostrømsstatistikk i modellen MOSART", *Interne notater 91/7*, Statistisk sentralbyrå.

Dennis Fredriksen (1992A): "Datagrunnlaget for trygdemodellen MOSART-T", *Interne notater 92/7*, Statistisk sentralbyrå.

Dennis Fredriksen (1992A): "Analyser av overganger til uførhet og opptjening av pensjonsgivende inntekt, for trygdemodellen MOSART-T", *Interne notater 92/8*, Statistisk sentralbyrå.

Dennis Fredriksen (1993): "Dokumentasjon av input til MOSART", *Notater 93/42*, Statistisk sentralbyrå.

Dennis Fredriksen og Gina Spurkland (1993): "Framskrivning av alders- og uføretrygd ved hjelp av mikrosimuleringsmodellen MOSART", *Rapport 93/7*, Statistisk sentralbyrå.

Det kan være hensiktsmessig å ha en utlisting av simuleringsprogrammet og de SAS-programmene som bearbeider output.

Statistisk sentralbyrå

Oslo
Postboks 8131 Dep.
0033 Oslo

Tlf.: 22 86 45 00
Fax: 22 86 49 73

Kongsvinger
Postboks 1260
2201 Kongsvinger

Tlf.: 62 88 50 00
Fax. 62 88 50 30



Statistisk sentralbyrå
Statistics Norway