

Ændring af beregningsmetode for momsbelastningsgraderne i foreløbige år

Resumé:

Den nuværende metode (A) for beregning af momsbelastningsgraderne i foreløbige år, resulterer ofte i, at nogle af størrelserne bliver større end 1. Dette papir beskriver, hvordan beregningsmetoden kan ændres, for at undgå disse tilfælde. Der undersøges to forskellige alternative metoder. Den første, metode (B), justerer momsbelastningsgraderne med afstanden til 1, i stedet for at justere alle momsbelastningsgraderne med den samme faktor. Derved justeres der mindst i momsbelastningsgrader tættest på 1, og mest i momsbelastningsgrader tættest på 0. I den anden metode, (C), beregnes først et fiktivt maksimalt provenu svarende til en momsbelastningsgrad på 1, og der justeres forskelligt alt efter om det beregnede momsprovenu bliver for højt eller for lavt. Metoden er altså asymmetrisk; hvis provenuet bliver for stort, justeres momsbelastningsgraden mod 0, dvs. metode (A). Hvis momsprovenuet bliver for lille, justeres mod en belastningsgrad på 1. Asymmetrien opstår fordi metode (A) benytter $k(btg-0)$ og metode (C) benytter $k*(1-btg)$. Det vurderes at metode (C) er bedst til beregningen af momsbelastningsgrader i foreløbige år.*

NNA250319

Nøgleord: Momsbelastningsgrad, *spg*, *btg*, metodeændring, data, foreløbige år

Modelgruppepapirer er interne arbejdsrapporter. De konklusioner, der drages i papirerne, er ikke endelige og kan være ændret inden opstillingen af nye modelversioner. Det henstilles derfor, at der kun citeres fra modelgruppepapirerne efter aftale med Danmarks Statistik.

1. Introduktion

Momsbelastningsgraden er udtryk for, hvor stor en del af anvendelsen, der i praksis påføres en moms. De samlede artsopdelte afgiftsprovenuer er tilgængelige for foreløbige år, mens statistik for de anvendelsesfordelte provenuer kun er tilgængelige for endelige år. I ADAM beregnes de anvendelsesfordelte momsprovenuer i foreløbige år derfor ud fra det seneste endelige års momsbelastningsgrad. Det gøres trinvist: Først antages at momsbelastningsgraden er uændret fra sidste endelige år, hvorefter momsprovenuerne proportionaljusteres, for til sidst at genberegne momsbelastningsgraderne.

Et problem med den metode der anvendes til beregningen af momsbelastningsgraderne er dog, at den for grupper med en momsbelastningsgrad tæt på 1, kan resultere i værdier større end 1. I afsnit 2 beskrives den metode der benyttes til justeringen og afsnit 3 giver forslag til to alternative metoder, som bedre kan tage højde for at momsbelastningsgraderne ikke bliver for store. De tre metoder (A, B, C) sammenlignes. Afsnit 4 konkluderer.

2. Beregning af moms og momsbelastningsgrader, *Spg* og *btg*

Udgangspunktet for at beregne de anvendelsesfordelte momsprovenuer i de foreløbige år, er det seneste endelige års momsbelastningsgrad, dvs. hvor stor en del af forbruget der rent faktisk påføres en moms. En momsbelastningsgrad på 90 pct., henviser eksempelvis til, at 10 pct. af forbruget er momsfritaget. Momsbelastningsgraderne beregnes for tre forskellige grupper i ADAM: Forbrug, investeringer og materialer (input i produktion). Typisk har belastningsgraden for forbrugsgrupperne en værdi tæt på 1, idet der pålægges moms på de fleste forbrugsgoder. Belastningsgraden for produktionsgrupperne har ofte en værdi tæt på 0, som det fremgår af tabellen nedenfor:

Tabel 1: Værdi af *btg*'erne i 2014

Forbrug	Værdi i 2014	Produktion	Værdi i 2014	Investeringer	Værdi i 2014
<i>btgcb</i>	0,7724	<i>btgxa</i>	0,0015	<i>btgib</i>	0,6127
<i>btgce</i>	0,9819	<i>btgxb</i>	0,0011	<i>btgibh</i>	0,9036
<i>btgcf</i>	0,9413	<i>btgxe</i>	0,0012	<i>btgibo1</i>	0,8088
<i>btgcg</i>	0,9816	<i>btgxb</i>	0,4196	<i>btgibp1</i> ¹	0,0713
<i>btgch</i>	0,0809	<i>btgxne</i>	0,0019	<i>btgikn</i>	0,3207
<i>btgco</i>	0,0161	<i>btgxnf</i>	0,0012	<i>btgil</i>	0,0000
<i>btgcs</i>	0,3437	<i>btgxng</i>	0,0002	<i>btgim</i>	0,1554
<i>btgcv</i>	0,9244	<i>btgxnz</i>	0,0050	<i>btgimo1</i>	0,3855
		<i>btgxo</i>	0,6859	<i>btgimp1</i>	0,1199
		<i>btgxqf</i>	0,3269		
		<i>btgxqs</i>	0,0127		
		<i>btgxqz</i>	0,0844		

¹ Det bemærkes at der for *btgibp1* sker en stor ændring fra 2011 til 2012, hvor momsbelastningsgraden skifter fra at ligge på et niveau omkring 0,20 til et niveau omkring 0,07. Det forklares umiddelbart ved metro-ringen og Køgebugt-udvidelsen.

Størstedelen af *btg*'erne relateret til forbruget har en værdi tæt på 1, bortset fra *btgch* og *btgco*. For produktionsgrupperne har størstedelen en værdi tæt på 0, hvis man ser bort fra gruppe *btgxo*, *btgxm* og *btgxf*. For investeringsgrupperne ligger momsbelastningsgraderne mere spredt i intervallet fra 0-1.

Der foreligger ikke data for de anvendelsesfordelte momsprovenuier i foreløbige år, og det antages derfor initialt at momsbelastningsgraden er uændret siden seneste endelige år. Ud fra momssatsen og værdien af forbruget kan momsprovenuet beregnes som:

$$Spg_{\{i\}t} = \frac{btg_{\{i\}t0} * tg_t * \{i\}_t}{1 + btg_{\{i\}t0} * tg_t} \quad (1a)$$

Og på følgende måde hvis anvendelsen indeholder registreringsafgift på motor-køretøjer:

$$Spg_{\{i\}t} = \frac{btg_{\{i\}t0} * tg_t * (\{i\}_t - Spr_{\{i\}})}{1 + btg_{\{i\}t0} * tg_t} \quad (1b)$$

Variablen $Spg_{\{i\}}$ er input-output leverancen fra tilgang Spg til anvendelse $\{i\}$ i løbende priser, tg er den generelle momssats, $btg_{\{i\}}$ er momsbelastningsgraden og $\{i\}$ er produktionen. Derudover er $\{i\} = cb, ce, cf, cg, ch, co, cs, ct, cv, ib, ibh, ibol, ibpl, ikn, il, im, imol, impl, xa, xc, xe, xm, xne, xnf, xng, xnz, xo, xqf, xqs, xqz$.

Efterfølgende proportionaljusteres momsprovenuierne på følgende måde:

$$Spg_{\{i\}} = Spg_{\{i\}} * \left(\frac{Spg}{Spg} \right) \quad (2)$$

Hvor Spg er det samlede momsprovenu for alle brancher og $\widehat{Spg} = \sum_i \widehat{Spg}_{\{i\}}$. Derved kan momsbelastningsgraderne for de foreløbige år beregnes som:

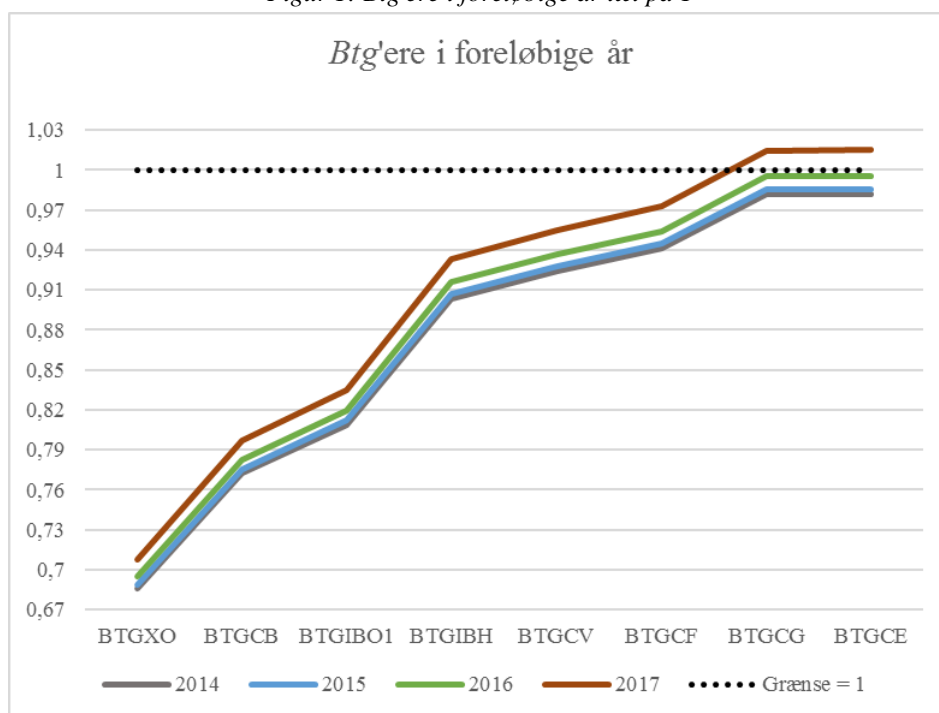
$$btg_{\{i\}} = \frac{Spg_{\{i\}}}{(\{i\} - Spg_{\{i\}}) * tg} \quad (3a)$$

Her udtrykker tælleren det faktiske momsprovenu fra anvendelsen, mens nævneren er det momsprovenu der ville være, hvis hele forbruget pålægges momsen. For anvendelser, der indeholder registreringsafgift på motor-køretøjer, tager beregningen højde for dette på følgende måde:

$$btg_{\{i\}} = \frac{Spg_{\{i\}}}{(\{i\} - Spr_{\{i\}} - Spg_{\{i\}}) * tg} \quad (3b)$$

Figur 1 nedenfor illustrerer et uddrag af *btg*'erne for det sidste endelige år (2014) og de beregnede *btg*'ere i foreløbige år (2015-2017). *Btg*'erne parallelforskydes i de foreløbige år. Det bemærkes, at kun de momsbelastningsgrader der er tæt på 1 fremgår af figuren, for bedre at illustrere parallelforskydningen.

Figur 1: *Btg*'ere i foreløbige år tæt på 1



Som det fremgår af figuren, er momsbelastningsgraderne for benzin og brændsel, *btgce* og *btgcg*, større end 1 i 2017, hvilket indikerer, at over 100 pct. af forbruget pålægges moms. Intuitivt giver det ikke mening, at der beregnes *btg*'ere der er større end 1. Det undersøges derfor i afsnit 3 om man kan beregne *btg*'erne således, at man kan undgå tilfælde, hvor belastningsgraden bliver større end 1.

3. Sammenligning af beregningsmetoder

Overordnet set, 'beregnes' momsbelastningsgraderne i to omgange. Først fremskrives de uændret for at *spg_{i}*'erne kan beregnes, derefter genberegnes de i hovedprogrammet. Der præsenteres nu to alternative beregningsmetoder for foreløbige år, der er med til at sikre, at momsbelastningsgraderne ikke bliver større end 1. Den nuværende metode benævnes *metode A* og de to alternative *metode B* og *metode C*.

De tre metoder beregner momsbelastningsgraderne som beskrevet nedenfor:

Metode A
<p>(1) Momsbelastningsgraderne fremskrives uændret i foreløbige år.</p> <p>(2) Det undersøges hvor meget for stort/for lille det samlede provenu, spg, bliver i foreløbige år. Alle provenuer øges med denne procentandel, $k = \frac{Sp_g}{\widehat{Sp_g}}$ (ligning 2).</p> <p>(3) Momsbelastningsgraden beregnes igen i hovedprogrammet (ligning 3a og 3b).</p>
Metode B
<p>(1) Momsbelastningsgraderne fremskrives uændret i foreløbige år.</p> <p>(2) Det undersøges hvor meget for stort/for lille det samlede provenu, spg, bliver i foreløbige år. De enkelte provenuer ændres med denne absolute forskel, ganget med afstanden til 1 for den enkelte gruppe, som andel af den samlede afstand til 1 fra alle grupper. Det vil sige at ligning (2) ændres til:</p> $Sp_g\{i\} = \widehat{Sp_g}\{i\} + \left((Sp_g - \widehat{Sp_g}) * \left(\frac{1 - btg\{i\}}{\sum_i (1 - btg\{i\})} \right) \right) \quad (2^*)$ <p>(3) Momsbelastningsgraden beregnes igen i hovedprogrammet (ligning 3a og 3b).</p>
Metode C
<p>I metode (C) vil beregningen fortsat være som i metode (A) hvis det beregnede momsprovenu bliver for stort (dvs. $k < 1$), og altså skal nedjusteres. Hvis det beregnede momsprovenu i stedet bliver for stort (dvs. $k > 1$), benyttes metode (C). Her beregnes først et fiktivt provenu, svarende til en momsbelastningsgrad på 1 (den maksimale moms):</p> $Sp_g\widehat{max}\{i\} = \frac{1 * tg * \{i\}}{1 + 1 * tg} = \frac{tg}{1 + tg} * \{i\}$ <p>Hvor $\{i\}$ er anvendelsen. Der beregnes en k-faktor, som afspejler afstanden til maksimal moms:</p> $k' = \frac{Sp_g - \widehat{Sp_g}}{Sp_g\widehat{max} - \widehat{Sp_g}}$ <p>Hvor $\widehat{Sp_g} = \sum_i \widehat{Sp_g}\{i\}$ og $Sp_g\widehat{max} = \sum_i Sp_g\widehat{max}\{i\}$. Når Sp_g er større end $\widehat{Sp_g}$, vil k' få en værdi der er mindre end 1 (og større end 0).</p> <p>Momsprovenuet beregnes i stedet som:</p> $\widehat{Sp_g}\{i\} = Sp_g\widehat{max}\{i\} + (Sp_g\widehat{max}\{i\} - \widehat{Sp_g}\{i\}) * k' \quad (2^{**})$ <p>Hvor $Sp_g\widehat{max}\{i\} - \widehat{Sp_g}\{i\}$ er værdien som svarer til at afstanden til momsbelastningsgrad = 1. Herefter beregnes btg'erne som i ligning (3a) og (3b).</p>

Gekkokoden for metode (A), (B) og (C) fremgår af bilag 1 og bilag 2. Alle tre metoder benytter proportionaljustering. I metode (A) er ankeret værdien 0 og der er dermed risiko for at få værdier som er større end 1. Der justeres med samme faktor i alle brancher, hvorved momsbelastningsgraderne parallelforskydes. I metode (B) ganges en faktor på, som tager højde for hvor langt de enkelte grupper er fra 1. Derved justeres der mere i de grupper der har en lav momsbelastningsgrad og mindre i de grupper der har en høj momsbelastningsgrad. Risikoen for at få en momsbelastningsgrad der er større end 1 bliver derved væsentligt formindsket. I metode (B) parallelforskydes momsbelastningsgraderne for de enkelte forbrugsgrupper ikke med ændringen i det beregnede momsprovenu som i metode (A). I metode (C) beregnes først et maksimalt (fiktivt) momsprovenu, og i tilfælde hvor momsprovenuet bliver for lille og skal opjusteres, benyttes det maksimale provenu til justeringen.

Tabellen nedenfor illustrerer forskellen i beregningen af *btg*'ere i foreløbige år med metode (A) og (B), for momsbelastningsgrader tæt på 0 og tæt på 1:

Tabel 2: Sammenligning af beregnede *btg*'er tæt på 1 og 0 i foreløbige år ved metode A og B

	2014	2015	2016	2017
Metode A				
<i>btgce</i>	0,981912	0,985770	0,995324	1,014924
<i>btgcg</i>	0,981598	0,985454	0,995005	1,014598
<i>btgxnf</i>	0,001160	0,001163	0,001172	0,001191
<i>btgxng</i>	0,000196	0,000197	0,000198	0,000202
Metode B				
<i>btgce</i>	0,981912	0,981982	0,982171	0,982559
<i>btgcg</i>	0,981598	0,981744	0,982158	0,982950
<i>btgxnf</i>	0,001160	0,002351	0,005545	0,012043
<i>btgxng</i>	0,000196	0,006329	0,026815	0,047454
Metode C				
<i>btgce</i>	0,981912	0,981932	0,981983	0,982085
<i>btgcg</i>	0,981598	0,981618	0,98167	0,981774
<i>btgxnf</i>	0,001160	0,002058	0,004327	0,008860
<i>btgxng</i>	0,000196	0,001095	0,003366	0,007903

Hvis de røde tal i tabellen sammenlignes, fremgår det, at momsbelastningsgraderne, *btgce* og *btgcg* er større end 1 i det sidste foreløbige år, 2017, med metode (A). De tilsvarende røde tal ved metode (B) og metode (C) viser, at momsbelastningsgraderne, *btgce* og *btgcg*, ikke bliver større end 1 i 2017. Tilsvarende sammenlignes to momsbelastningsgrader, *btgxng* og *btgxnf*, der er meget tæt på 0. De er markeret med blå i tabellen ovenfor, og det fremgår, at de stiger meget lidt i de foreløbige år med metode (A), mens de stiger betydeligt mere med metode (C) og endnu mere med metode (B). Det store ændring i metode (B) skyldes, at der justeres mest i de momsbelastningsgrader der er tættest på 0.

Tabel 3: Procentvis ændring fra 2014-2017 ved de tre forskellige metoder
Procentvis ændring fra 2014-2017, angivet i pct.

Variabel	Metode (A)	Metode (B)	Metode (C)
<i>btgce</i>	3,36	0,07	0,02
<i>btgcg</i>	3,36	0,14	0,02
<i>btgxnf</i>	2,67	938,19	663,79
<i>btgxng</i>	3,06	24111,22	3932,14

Ved metode (A) er der en procentvis ændring fra 2014-2017 på ca. 3 pct. Det er tydeligt, at der er større forskel på metode (B), hvor der justeres meget lidt på de høje momsbelastningsgrader og meget på de små momsbelastningsgrader. Metode (C) har lidt af de samme udfordringer som metode (B), som det fremgår af tabel 3 ovenfor. Her justeres også mindst på momsbelastningsgrader der er tæt på 1, og mest i momsbelastningsgrader der er tæt på 0, dog ikke helt i samme grad som ved metode (B).

Det bemærkes, at metode (C) er asymmetrisk, hvis provenuet bliver for stort, justeres momsbelastningsgraden mod 0, dvs. metode (A). Hvis momsens bliver for lille, justeres mod en belastningsgrad på 1. Asymmetrien opstår fordi metode (A) benytter $k*(btg-0)$ og metode (C) benytter $k*(1-btg)$. Hvis man i stedet ser på den procentvise ændring i $(1-btg)$ fra 2014-2017, vil metode (A) altså blive justeret meget i de høje momsbelastningsgrader og meget lidt i de lave momsbelastningsgrader, mens metode (C) justeres nogenlunde ens i grupperne. Der ses altså en spejlvendt udvikling i forhold til tabel 3.

Tabel 4: Procentvis ændring fra 2014-2017 for $(1-btg)$

Variabel	Metode (A)	Metode (C)
<i>btgce</i>	182,53	0,96
<i>btgcg</i>	179,35	0,96
<i>btgxnf</i>	0,003	0,77
<i>btgxng</i>	0,001	0,77

Ved at benytte metode (B) eller (C) undgås tilfælde hvor momsbelastningsgraden bliver større end 1, til gengæld justeres der betydeligt mere i de varegrupper der har en meget lille momsbelastning. Derved kan de alternative metoder også medføre store forskydninger i momsprovenuerne i de foreløbige år. Omkostningerne er størst ved metode (B)². Metode (C) virker derfor som et bedre alternativ,

² Det skal bemærkes, at metode (B) som her beskrevet, ikke tager højde for fald i *spg* i de foreløbige år. Her skal afstanden til 0 i stedet for 1 benyttes, for derved at justere mest i grupper med høj momsbelastningsgrad og mindst i grupper med lav momsbelastningsgrad, og derved undgå negative momsbelastningsgrader i de foreløbige år.

men det skal her bemærkes, at der i så fald vil blive benyttet metode (A) i tilfælde hvor momsprovenu bliver for stort, og skal nedjusteres i stedet for opjusteres.

Det vurderes, at den bedste løsning er at vælge metode (C) til tilfælde hvor k-faktoren er større end 1, det vil sige når det beregnede momsprovenu er for lavt i forhold til det samlede, og der skal opjusteres. Når k-faktoren i stedet er mindre end 1, og det beregnede momsprovenu bliver for stort og skal nedjusteres, benyttes fortsat metode (A). Derved sikres, at de beregnede momsbelastningsgrader for foreløbige år vil være i intervallet $[0, 1]$.

4. Konklusion

Papiret har vist, hvordan beregningsmetoden for *btg*'erne i foreløbige år kan ændres så belastningsgraden undgår at blive større end 1. Ved at benytte den alternative metode (B), parallelforskydes momsbelastningsgraderne for de enkelte forbrugsgrupper ikke med ændringen i det beregnede momsprovenu som i den nuværende metode (A). I stedet ganges en faktor på, som tager højde for hvor stor afstanden til 1 er for de enkelte grupper, således at momsbelastningsgrader tæt på 1 kun vil ændres ganske lidt, mens momsbelastningsgrader der er tæt på 0 vil øges tilsvarende mere. Hvis det samlede momsprovenu falder i stedet for at stige i de foreløbige år, skal metode (B) ændres til at fordele justeringen i forhold til de enkelte gruppers afstand til 0 i stedet for afstand til 1. Alternativt kan man benytte metode (C), som også vil sikre at momsbelastningsgraderne i foreløbige år er mindre end 1. Det vurderes, at der er færrest udfordringer med metode (C), hvor der justeres lidt mindre i de mindste grupper, og derfor vælges det at benytte denne til beregning af momsbelastningsgrader i foreløbige år.

Bilag 1: Metode (A)

```

***** Beregning af spg'erne i datarevisionen, metode (A)*****
CLOSE * ;clear ;
run g:\datopgek\periode ;

list erh = a,b,e,h,o,ne,nf,ng,nz,qf,qs,qz ;
list forb = cb,ce,cf,cg,ch,co,cs,cv ; // ,ct
list forxb = ce,cf,cg,ch, cs,cv ; // cb,co,ct ;

time %per0 %per2;
open<> spp ;
copy <from=spp to=work > spr_imp ;
copy <from=spp to=work > spr_cb ;
copy <from=spp to=work > spr_co ;
close spp ;

open %obk ;
list lverh = #erh prefix=v ;
copy <from=obk to=work> #lverh, #forb ;
copy <from=obk to=work> spg,tg,ibh,ibol,ibpl,impl,imol,ib,im,il,ikn;
close obk ;

time %per0 %per0 ;

if ('%per0' == '%per1' );
open './histdata/histdata' as a;
index a:btg* lbtg ;
copy <from=a to=work respect> #lbtg ;
else
open g:\datopgek\obk as a;
index a:btg* lbtg ;
copy <from=a to=work respect> #lbtg ;
end;
close a;
time %per0 %per2;

for i = #lbtg ;
series {i} = {i}[%per0] ;
end ;

// moms fra varekøb
for i = #erh ;
series jspg_x{%i} = btgx{%i}*tg*v{%i}/(1+btgx{%i}*tg) ;
end ;
list lspg_x = #erh prefix=jspg_x ;

// moms fra privat forbrug
for i = #forxb ;
series jspg_{%i} = btg{%i}*tg*{%i}/(1+btg{%i}*tg) ;
end ;

series jspg_cb = btgcb*tg*(cb-spr_cb)/(1+btgcb*tg) ;
series jspg_co = btgco*tg*(co-spr_co)/(1+btgco*tg) ;

list lspg_c = #forb prefix=jspg_ ;

// moms fra investeringer
list lspg_i = null ;
for i = ibh,ibol,ibpl,imol,il,ikn;
series jspg_{%i} = btg{%i}*tg*{%i}/(1+btg{%i}*tg) ;
list lspg_i = #lspg_i &+ 'jspg_{%i}' ;
end ;

series jspg_imp = btgimpl*tg*(impl-spr_imp)/(1+btgimpl*tg) ;
list lspg_i = #lspg_i &+ 'jspg_imp' ;

```

```

// til proportionaljustering
series spgsum = sum(#lspg_x) + sum(#lspg_c) + sum(#lspg_i) ;

series kspg = spg/spgsum ;

for i = #erh ;
  series spg_x{%i} = btgx{%i}*tg*v{%i}/(1+btgx{%i}*tg)*kspg ;
end ;

for i = #forbxb ;
  series spg_{%i} = btg{%i}*tg*{%i}/(1+btg{%i}*tg)*kspg ;
end ;

series spg_cb = btgcb*tg*(cb-spr_cb)/(1+btgcb*tg)*kspg ;
series spg_co = btgco*tg*(co-spr_co)/(1+btgco*tg)*kspg ;

series spg_il = btgil*tg*il/(1+btgil*tg)*kspg ;
series spg_ikn = btgikn*tg*ikn/(1+btgikn*tg)*kspg ;
series spg_ibh = btgibh*tg*ibh/(1+btgibh*tg)*kspg ;
series spg_imo = btgimol*tg*imol/(1+btgimol*tg)*kspg ;
series spg_ibp = btgibpl*tg*ibpl/(1+btgibpl*tg)*kspg ;
series spg_ibo = btgibol*tg*ibol/(1+btgibol*tg)*kspg ;
series spg_imp = btgimpl*tg*(impl-spr_imp)/(1+btgimpl*tg)*kspg ;
series spg_im = spg_imp + spg_imo ;
series spg_ib = spg_ibp + spg_ibo + spg_ibh ;
series spg_c = spg_cf+spg_ce+spg_cg+spg_cv+spg_ch+spg_cb+spg_cs+spg_co;
series spg_il = spg_il+spg_ib+spg_im;
series spg_x =
spg_xa+spg_xh+spg_xb+spg_xe+spg_xo+spg_xne+spg_xng+spg_xnf+spg_xnz+spg_xqf+spg_
xqs+spg_xqz;

// nul provenuer
for i=e01,e2,e3,e59,e7y,es,et ;
  series spg_{%i} = 0 ;
end;

// Dannelse af spg bank
index spg_x*,spg_c*,spg_i* lspga ;
index spg_*1 lspgx ;

// fjerner alle variabler med efterstillet 1
list lspg = #lspga &- #lspgx ;

index btg* lbtg ;
open <edit> spg ; clear spg ;
copy <from=work to=spg> #lspg, #lbtg ;

close * ;clear ;

***** Efterberegning af btg'erne i hovedprogrammet, metode (A) *****
list ilist=f,v,e,g,h,s ; // minus b,t,o
for i=#ilist;
  series TPC%i = SPP_C%i/FC%i ;
  series BTGC%i = SPG_C%i/((C%i-SPG_C%i)*TG) ;
end;
series tpcb = 0 rep *;
series spr_cb=spr-spr_impl-spr_co;
series tpcb = (Spp_cb-Spr_Cb )/fCb;
series tpcO = (Spp_co-Spr_Co )/fCo;

list ilist=ibp,ibo,imo;
for i=#ilist;
  series TP%i|1 = SPP_%i/F%i|1 ;
  series BTG%i|1 = SPG_%i/((%i|1 -SPG_%i)*TG) ;
end;

```

```

series TPIMP1 = (Spp_imp-Spr_imp1 )/fImp1 ;
series BTGIMP1 = Spg_imp/((Impl-Spg_imp-Spr_imp1)*TG) ;
series TRIMP1 = Spr_imp1/(Impl-Spr_imp1) ;

list ilist=ibh,il,ikn;
for i=#ilist;
  series TP%i = SPP_%i/F%i ;
  series BTG%i = SPG_%i/((%i-SPG_%i)*TG) ;
end;

list minuse = a,b,ne,ng,nf,nz,qf,qs,qz,o,h; // e,
for i = #minuse ;
  series BTGX%i = SPG_X%i/((V%i-SPG_X%i)*TG) ;
end;

series BTGXe = Spg_xe/((Ve-Spg_xe)*TG) ;
series btgibh = Spg_ibh/(tg*pibh*fIbh-Spg_ibh*tg) ;
series btgibp1 = Spg_ibp/(tg*pibp1*fIbp1-Spg_ibp*tg) ;
series tpil = Spp_il/fil;

```

Bilag 2: Metode (B)

***** Efterberegning af btg'erne i hovedprogrammet, metode (B) *****

```

close * ;clear ;
run p:\nna\afgift\ny_btg\periode ;

list erh = a,b,e,h,o,n,e,nf,ng,nz,qf,qs,qz ;
list forb = cb,ce,cf,cg,ch,co,cs,cv ; // ,ct
list forbx = ce,cf,cg,ch, cs,cv ; // cb,co,ct ;

time %per0 %per2;
open<> spp ;
copy <from=spp to=work > spr_imp ;
copy <from=spp to=work > spr_cb ;
copy <from=spp to=work > spr_co ;
close spp ;

open p:\nna\afgift\ny_btg\obk;

list lverh = #erh prefix=v ;
copy <from=obk to=work> #lverh, #forb ;
copy <from=obk to=work> spg,tg,ibh,ibol,ibpl,impl,imol,ib,im,il,ikn, spr,
spr_impl, pibh, fibh, pibpl, fibpl,
pcb,pcb,pibh,fibh,pibol,fibol,pimpl,fimpl,pimol,fimol,pil,fil,pikn,fikn,
spp_cf, fcf, spp_cv, fcv, spp_ce, fce, spp_cg, fcg, spp_ibh, spp_ikn, fikn,
spp_il, fil, fibh, spp_ch, spp_imp, fimp, spp_imo, fimo, spp_ibo, fibo, fch,
spp_cs, fcs, spp_ibp, fibp, spp_cb, spp_co, fco, fcb;
close obk ;

time %per0 %per0 ;

if ('%per0' == '%per1' );
open p:\nna\afgift\histdata\histdata as a;
index a:btg* lbtg ;
copy <from=a to=work respect> #lbtg ;
else
open p:\nna\afgift\ny_btg\obk as a;
index a:btg* lbtg ;
copy <from=a to=work respect> #lbtg ;
end;
close a;

time %per0 %per2;

for i = #lbtg ;
series {i} = {i}[%per0] ;
end ;

// moms fra varekøb
for i = #erh ;
series jspg_x{%i} = btgx{%i}*tg*v{%i}/(1+btgx{%i}*tg) ;
end ;
list lspg_x = #erh prefix=jspg_x ;

// moms fra privat forbrug
for i = #forbx ;
series jspg_{%i} = btg{%i}*tg*{%i}/(1+btg{%i}*tg) ;
end ;

series jspg_cb = btgcb*tg*(cb-spr_cb)/(1+btgcb*tg) ;
series jspg_co = btgco*tg*(co-spr_co)/(1+btgco*tg) ;

list lspg_c = #forb prefix=jspg_ ;

// moms fra investeringer
list lspg_i = null ;
for i = ibh,ibol,ibpl,imol,il,ikn;

```

```

series jspg_{%i} = btg{%i}*tg{%i}/(1+btg{%i}*tg) ;
list lspg_i = #lspg_i &+ 'jspg_{%i}' ;
end ;

series jspg_imp = btgimpl*tg*(impl-spr_imp)/(1+btgimpl*tg) ;

list lspg_i = #lspg_i &+ 'jspg_imp' ;

series spgsum = sum(#lspg_x) +
                sum(#lspg_c) +
                sum(#lspg_i) ;

series delta= spg-spgsum;

list erh1 =
xa,xb,xe,xh,xo,xne,xf,xng,xnz,xqf,xqs,xqz,ce,cf,cg,ch,cs,cv,cb,co,ikn,ibh,imol
,ibpl,ibol;
for i=#erh1;
    SERIES dbtg_%i = 1-btg%i; /* Her skal ændres, hvis ændringen i
spg er negativ. Ændres til btg%i i stedet for 1-btg%i */
end;

SERIES dbtg_imp = 1-btgimpl;

list erh2 = #erh1 &+ 'imp';
list dbtg = #erh2 prefix =dbtg;
for i=#erh1;
    SERIES sum_dbtg = sum(#dbtg);
end;

for i=#erh2;
    SERIES delta_dbtg_%i = delta*dbtg_%i/sum_dbtg;
end;

for i = #erh ;
    series spg_x{%i} = btgx{%i}*tg*v{%i}/(1+btgx{%i}*tg)+delta_dbtg_x%i;
end ;

for i = #forbxb ;
    series spg_{%i} = btg{%i}*tg*{%i}/(1+btg{%i}*tg)+delta_dbtg_%i;
end ;

series spg_cb = btgcb*tg*(cb-spr_cb)/(1+btgcb*tg)+delta_dbtg_cb;
series spg_co = btgco*tg*(co-spr_co)/(1+btgco*tg)+delta_dbtg_co;

series spg_il = btgil*tg*il/(1+btgil*tg); // +delta_dbtg_il;
series spg_ikn = btgikn*tg*ikn/(1+btgikn*tg)+delta_dbtg_ikn;
series spg_ibh = btgibh*tg*ibh/(1+btgibh*tg)+delta_dbtg_ibh;
series spg_imo = btgimol*tg*imol/(1+btgimol*tg)+delta_dbtg_imo;
series spg_ibp = btgibpl*tg*ibpl/(1+btgibpl*tg)+delta_dbtg_ibpl;
series spg_ibo = btgibol*tg*ibol/(1+btgibol*tg)+delta_dbtg_ibol;
series spg_imp = btgimpl*tg*(impl-spr_imp)/(1+btgimpl*tg)+delta_dbtg_imp;

series spg_im = spg_imp + spg_imo ;
series spg_ib = spg_ibp + spg_ibo + spg_ibh ;

series spg_c = spg_cf+spg_ce+spg_cg+spg_cv+spg_ch+spg_cb+spg_cs+spg_co;
series spg_il = spg_il+spg_ib+spg_im;
series spg_x =
spg_xa+spg_xh+spg_xb+spg_xe+spg_xo+spg_xne+spg_xng+spg_xnf+spg_xnz+spg_xqf+spg_
xqs+spg_xqz;

***** Efterberegning af btg'erne i hovedprogrammet, metode (B) *****

for i=#forb;
    series BTG%i = SPG_%i/((%i-SPG_%i)*TG) ;
end;

```

```

list ilist=ibp1,ibo1,imol;
for i=#ilist;
  series BTG%i = SPG_%i/((%i -SPG_%i)*TG) ;
end;

series BTGIMP1 = Spg_imp/((Imp1-Spg_imp-Spr_imp1)*TG) ;

list ilist=ibh,il,ikn;
for i=#ilist;
  series BTG%i = SPG_%i/((%i-SPG_%i)*TG) ;
end;

for i = #erh &- 'e' ;
  series BTGX%i = SPG_X%i/((V%i-SPG_X%i)*TG) ;
end;

series BTGXe = Spg_xe/((Ve-Spg_xe)*TG) ;
series btgibh = Spg_ibh/(tg*pi bh*fIbh-Spg_ibh*tg) ;
series btgibp1 = Spg_ibp1/(tg*pi bpl*fIbpl-Spg_ibp1*tg) ;

prt [btg*], [spg*];

```

Bilag 3: Metode (C)

```

//*****
//
//          Beregning af spg'erne
//*****
// Undersøger hvor meget for lille det samlede provenu bliver i foreløbige år
//*****
close * ;clear ;
run p:\nna\afgift\ny_btg\periode ;

list erh = a,b,e,h,o,ne,nf,ng,nz,qf,qs,qz ;
list forb = cb,ce,cf,cg,ch,co,cs,cv ; // ,ct
list forbxb = ce,cf,cg,ch, cs,cv ; // cb,co,ct ;

time %per0 %per2;
open<> spp ;
  copy <from=spp to=work > spr_imp ;
  copy <from=spp to=work > spr_cb ;
  copy <from=spp to=work > spr_co ;
close spp ;

open p:\nna\afgift\ny_btg\obk;

list lverh = #erh prefix=v ;
copy <from=obk to=work> #lverh, #forb ;
copy <from=obk to=work> spg,tg,ibh,ibol,ibpl,impl,imol,ib,im,il,ikn, spr,
spr_impl, pibh, fibh, pibpl, fibpl,
pcb,pcb, pibh, fibh, pibol, fibol, pimol, fimpol, pimol, fimpol, pil, fil, pikn, fikn,
spp_cf, fcf, spp_cv, fcv, spp_ce, fce, spp_cg, fcg, spp_ibh, spp_ikn, fikn,
spp_il, fil, fibh, spp_ch, spp_imp, fimp, spp_imo, fimo, spp_ibo, fibo, fch,
spp_cs, fcs, spp_ibp, fibp, spp_cb, spp_co, fco, fcb;
close obk ;

time %per0 %per0 ;

if ('%per0' == '%per1' );
  open p:\nna\afgift\histdata\histdata as a;
  index a:btg* lbtg ;
  copy <from=a to=work respect> #lbtg ;
else
  open p:\nna\afgift\ny_btg\obk as a;
  index a:btg* lbtg ;
  copy <from=a to=work respect> #lbtg ;
end;
close a;

time %per0 %per2;

for i = #lbtg ;
  series {i} = {i}[%per0] ;
end ;

// moms fra varekøb
for i = #erh ;
  series jspg_x{%i} = btgx{%i}*tg*v{%i}/(1+btgx{%i}*tg) ;
end ;
list lspg_x = #erh prefix=jspg_x ;

// moms fra privat forbrug
for i = #forbxb ;
  series jspg_{%i} = btg{%i}*tg*{%i}/(1+btg{%i}*tg) ;
end ;

series jspg_cb      = btgcb*tg*(cb-spr_cb)/(1+btgcb*tg) ;
series jspg_co      = btgco*tg*(co-spr_co)/(1+btgco*tg) ;

```

```

list lspg_c = #forb prefix=jspg_ ;

// moms fra investeringer
list lspg_i = null ;
for i = ibh,ibol,ibpl,imol,il,ikn;
  series jspg_{%i} = btg{%i}*tg{%i}/(1+btg{%i}*tg) ;
  list lspg_i = #lspg_i &+ 'jspg_{%i}' ;
end ;

series jspg_imp = btgimpl*tg*(impl-spr_imp)/(1+btgimpl*tg) ;

list lspg_i = #lspg_i &+ 'jspg_imp' ;

series spgsum = sum(#lspg_x) +
               sum(#lspg_c) +
               sum(#lspg_i) ;

series kspg1 = spg/spgsum;

// btg = 1
for i = #erh ;
  series bspg_x{%i} = 1*tg*v{%i}/(1+1*tg) ;
end ;
list bspg_x = #erh prefix=bspg_x ;

// moms fra privat forbrug
for i = #forbxb ;
  series bspg_{%i} = 1*tg*{%i}/(1+1*tg) ;
end ;

series bspg_cb = 1*tg*(cb-spr_cb)/(1+1*tg) ;
series bspg_co = 1*tg*(co-spr_co)/(1+1*tg) ;

list bspg_c = #forb prefix=bspg_ ;

// moms fra investeringer
list bspg_i = null ;
for i = ibh,ibol,ibpl,imol,ikn;
  series bspg_{%i} = 1*tg*{%i}/(1+1*tg) ;
  list bspg_i = #bspg_i &+ 'bspg_{%i}' ;
end ;

series bspg_imp = 1*tg*(impl-spr_imp)/(1+1*tg) ;
series bspg_il = 0;

list bspg_i = #bspg_i &+ 'bspg_imp' ;

series bspgsum = sum(#bspg_x) +
                sum(#bspg_c) +
                sum(#bspg_i) ;

series kspg2 = 1+((spg-spgsum)/(bspgsum-spgsum));
prt kspg1, kspg2;
//*****
//          spg'erne beregnes
//*****

for date yy = %per0 to %per2 ;
  time %yy %yy ;
  if(kspg1[%yy] < 1 ) ;
    for i = #erh ;
      series spg_x{%i} = btgx{%i}*tg*v{%i}/(1+btgx{%i}*tg)*kspg1 ;
    end ;

    for i = #forbxb ;
      series spg_{%i} = btg{%i}*tg*{%i}/(1+btg{%i}*tg)*kspg1 ;
    end ;
  end ;
end ;

```



```

end ;

series spg_cb = btgcb*tg*(cb-spr_cb)/(1+btgcb*tg)*kspg1 ;
series spg_co = btgco*tg*(co-spr_co)/(1+btgco*tg)*kspg1 ;
series spg_ikn = btgikn*tg*ikn/(1+btgikn*tg)*kspg1 ;
series spg_ibh = btgibh*tg*ibh/(1+btgibh*tg)*kspg1 ;
series spg_imo = btgimol*tg*imol/(1+btgimol*tg)*kspg1 ;
series spg_ibp = btgibp1*tg*ibp1/(1+btgibp1*tg)*kspg1 ;
series spg_ibo = btgibo1*tg*ibo1/(1+btgibo1*tg)*kspg1 ;
series spg_imp = btgimp1*tg*(imp1-spr_imp)/(1+btgimp1*tg)*kspg1 ;
series spg_il = 0 ;

else if(kspg2[%yy] > 1 ) ;

for i = #erh ;
series spg_x{%i} = jspg_x{%i}+(bspg_x{%i}-jspg_x{%i})*(kspg2-1) ;
end ;

for i = #forbxb ;
series spg_{%i} = jspg_{%i}+(bspg_{%i}-jspg_{%i})*(kspg2-1) ;
end ;

series spg_cb = jspg_cb+(bspg_cb-jspg_cb)*(kspg2-1) ;
series spg_co = jspg_co+(bspg_co-jspg_co)*(kspg2-1) ;
series spg_ikn = jspg_ikn+(bspg_ikn-jspg_ikn)*(kspg2-1) ;
series spg_ibh = jspg_ibh+(bspg_ibh-jspg_ibh)*(kspg2-1) ;
series spg_imo = jspg_imo+(bspg_imo-jspg_imo)*(kspg2-1) ;
series spg_ibp = jspg_ibp+(bspg_ibp-jspg_ibp)*(kspg2-1) ;
series spg_ibo = jspg_ibo+(bspg_ibo-jspg_ibo)*(kspg2-1) ;
series spg_imp = jspg_imp+(bspg_imp-jspg_imp)*(kspg2-1) ;
series spg_il = 0 ;

end;
end ;
end ;

//*****
//          HOVED
//*****
time %per3 %per2;

for i=#forb &- 'cb' &- 'co';
series BTG%i = SPG_%i/((%i-SPG_%i)*TG) ;
end;

series BTGcb = SPG_cb/((cb-spr_cb-SPG_cb)*TG) ;

series BTGco = SPG_co/((co-spr_co-SPG_co)*TG) ;

list ilist=ibp,ibo,imo;
for i=#ilist;
series BTG%i|1 = SPG_%i/((%i|1 -SPG_%i)*TG) ;
end;

series BTGIMP1 = Spg_imp/((Impl-Spg_imp-Spr_imp1)*TG) ;

list ilist=ibh,ikn;
for i=#ilist;
series BTG%i = SPG_%i/((%i-SPG_%i)*TG) ;
end;

for i = #erh &- 'e' ;
series BTGX%i = SPG_X%i/((V%i-SPG_X%i)*TG) ;
end;

series btgil = 0;
series BTGXe = Spg_xe/((Ve-Spg_xe)*TG) ;

```

```
series btgibh    = Spg_ibh/(tg*plib*fIbh-Spg_ibh*tg)  ;
series btgibp1  = Spg_ibp/(tg*plibp1*fIbp1-Spg_ibp*tg) ;

prt <2014 2017> [btg*], [spg*];
```

```
//*****
//          Tjekker ændring i spg
//*****

// Tjekker resultat
time %per0 %per2;
index spg_* lspg;
prt spgsum, spg, sum(#lspg);
prt <dec=6> spg/spgsum, sum(#lspg)/spgsum;
```