Laborationsrapport 1
Optimering av Returpacks återvinningssystem
TAOP61

Adnan Avdagic, Carl-Martin Johansson, Joel Runesson

27 november 2017

Innehåll

Figurer

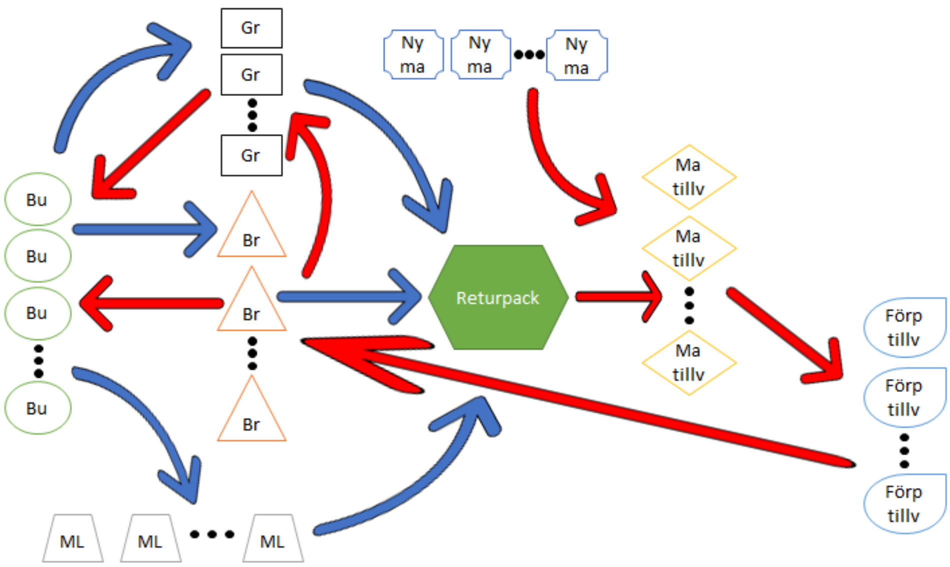
Tabeller

1 Inledning

1.1 Bakgrund

I Sverige ansvarar Returpack AB för hanteringen av återvinningssystemet av aluminiumburkar och PET-flaskor. Bolaget är inte vinstdrivande utan har som målsättning att hantera driften av och informera om återvinningssystemet, så att återvinningsmålen uppnås. Alla yrkesmässiga dryckesproducenter och importörer av dryckesför-packningar ansvarar för att flaskorna och burkarna som säljs i Sverige ingår i ett retursystem för återvinning. Aktörerna som ingår i systemet utgörs av bryggerier, grossister, butiker, materialtillverkare, förpackningstillverkare samt Returpack. Eventuellt bör även mellanlager ingå i återvinningssystemet, om dessa ger en lägre systemkostnad. I den här rapporten ska det undersökas huruvida byggnationen av mellanlager är lönsam eller inte. Rapporten är ett delmoment i kursen TAOP61 som ges av matematiska institutionen på Linköpings Tekniska Högskola.

I butikerna uppstår en efterfrågan då kunderna vill köpa dryck (i detta fall endast i burkar, inte andra sorters behållare). Kunden lämnar sedan tillbaka burken (pantar) till butiken. Alla burkar lämnas dock inte tillbaka utan det skapas ett spill. De burkar som lämnas in till butiken igen kan skickas till antingen grossister, bryggerier eller till lager om detta är möjligt i uppgiften. Detta kan ses i bilden nedan med de blå pilarna från butikerna. Grossisterna, bryggerierna och lagren skickar sedan vidare burkarna till Returpack vilket kan ses i samma bild nedan med blå pilar mellan dem. Returpack komprimerar dem sedan och skickar dem vidare till materialtillverkare vilket illustreras med en röd pil i bilden nedan mellan dem. Nytt material måste här tillföras för att uppfylla behovet av burkar då det försvinner lite material vid spill och omvandlingsfaktorer i kedjan. Detta visas med en röd pil mellan nytt material och materialtillverkare. Flödet går sedan vidare till förpackningstillverkare (röd pil dem emellan), där nya burkar produceras. Dessa är de burkar som sedan skickas till bryggerierna, visas med en röd pil mellan förpackningstillverkare och bryggerier, och där fylls med dryck. Bryggerierna kan sedan skicka fyllda burkar vidare till grossister som skickar till butiker eller skicka till butiker direkt. I bilden nedan representeras dessa flöden av röda pilar mellan bryggeri-grossist-butik och bryggeri-butik. Sedan säljs de fyllda burkarna igen i butikerna och det blir en cirkulerande kedja. I bilden är röda pilar burkflöden efter Returpack tills dess att burkarna lämnas tillbaka tomma till butikerna och ska till Returpack. Blå pilar är alltså burkflöden från tomma burkar i butikerna till Returpack.



Figur 1: Flödesschema för burkar i systemet.

2 Syfte

Syftet med projektet är att formulera en optimeringsmodell som optimerar olika problem som Returpack står inför. Dels ska modellen i ett normalt fall räkna ut hur burkar ska skickas mellan olika aktörer inom systemet beroende på kostnader och utsläpp, dels hur mycket nytt material som köps in. Sedan modifieras denna modell något, för att undersöka om det är lönsamt att bygga mellanlager om alla burkar skulle pantas, d.v.s. om inget spill finns, samt om det blir någon skillnad om råvarupriset multipliceras med hundra i fallet med noll spill. Detta körs sedan för alla varianter av problemet (totalt sju varianter). Syftet är alltså att lösa alla dessa optimeringsproblem och ge feedback kring hur Returpack bör tänka i de olika fallen för de olika varianterna av problemet, för att hitta optimalitet.

3 Metod

Valet av optimeringsprogram beror på strukturen på problemet som ska lösas. I detta fall är både målfunktionen och bivillkoren linjära, varför till exempel GLPK (GNU Linear Programming Kit) är lämpligt att använda. För den här laborationsuppgiften var det till och med bestämt att GLPK skulle användas. Utöver detta användes modelleringsspråket GMPL och lösaren glpsol.

3.1 Målfunktion

$$\text{min cost} : v\_cost + f\_cost + m\_cost \tag{3.1}$$

Målfunktionen ska minimera den sammanlagda kostnaden av återvinningssystemet, genom att minimera rörliga kostnader, fasta kostnader och materialkostnader.

3.2 Bivillkor

$$\sum_{j=1}^{\text{ngross}} nburk\_GR[i,j] + \sum_{k=1}^{\text{nbrygg}} nburk\_BR[i,k] + \sum_{l=1}^{\text{nlager}} nburk\_LR[i,l] = pant[i], \quad \forall i \in [1 \ldots \text{nbutik}] \tag{3.2}$$

De av kunderna pantade tomförpackningarna som varje butik i erhåller, ska också skickas från respektive butik till antingen bryggerierna, grossisterna eller mellanlagren. Panten som varje butik erhåller beräknas som butikens behov av förpackningar minus det antal förpackningar som inte pantades för just den butiken.

$$bin\_GR[j] * gkaptom[j] - \sum_{i=1}^{\text{nbutik}} nburk\_GR[i,j] \geq 0, \quad \forall j \in [1 \ldots \text{ngross}] \tag{3.3}$$

Antalet tomförpackningar som varje grossist får från alla butiker måste vara mindre än eller lika med den maximala kapaciteten av tomförpackningar som den specifika grossisten kan hantera.

$$bin\_BR[k] * bkaptom[k] - \sum_{i=1}^{\text{nbutik}} nburk\_BR[i,k] \geq 0, \quad \forall k \in [1 \ldots \text{nbrygg}] \tag{3.4}$$

Antalet tomförpackningar som varje bryggeri får från alla butiker måste vara mindre än eller lika med den maximala kapaciteten av tomförpackningar som det specifika bryggeriet kan hantera.

$$bin\_LR[l] * lkaptom[l] - \sum_{i=1}^{\text{nbutik}} nburk\_LR[i,l] \geq 0, \quad \forall l \in [1 \ldots \text{nlager}] \tag{3.5}$$

Antalet tomförpackningar som varje mellanlager får från alla butiker måste vara mindre än eller lika med den maximala kapaciteten av tomförpackningar som den specifika kan hantera.

$$0 \leq \sum_{l=1}^{\text{nlager}} bin\_LR[l] \leq maxlager \tag{3.6}$$

Antalet byggda mellanlager får inte vara större än det maximala antalet mellanlager som kan byggas, givetvis ej heller negativt.

$$0 \leq \sum_{m=1}^{\text{nmatr}} nnytt\_matr[m] \leq tot\_nytt\_matr \tag{3.7}$$

Mängden inköpt material för varje materialtillverkare får inte vara större än det totala behovet av inköpt material, givetvis ej heller negativt.

$$\sum_{m=1}^{\text{nmatr}} nburk\_RM[m] - faktR * tot\_pant = 0 \tag{3.8}$$

Alla tomförpackningar som skickas till Returpacks anläggning ska komprimeras så att summan av utflödet till alla materialtillverkare är lika med inflödet till Returpack gånger omvandlingsfaktorn faktR.

$$\left(\sum_{n=1}^{\text{nfpack}} nburk\_MT[m,n]\right) - nburk\_RM[m] * faktM - nnytt\_matr[m] = 0, \quad \forall m \in [1 \ldots \text{nmatr}] \tag{3.9}$$

Mängden material som skickas till alla tomförpackningstillverkare från varje materialtillverkare är lika med antalet komprimerade burkar som skickas från Returpack till alla materialtillverkare gånger omvandlingsfaktorn faktM plus mängden inköpt nytt material för varje materialtillverkare.

$$\left(\sum_{k=1}^{\text{nbrygg}} nburk\_TB[n,k]\right) - \sum_{m=1}^{\text{nmatr}} nburk\_MT[m,n] * faktF = 0, \quad \forall n \in [1 \ldots \text{nfpack}] \tag{3.10}$$

På samma sätt är antalet nytillverkade tomförpackningar som skickas till alla bryggerier från varje tomförpackningstillverkare lika med mängden material som skickas från alla materialtillverkare till varje tomförpackningstillverkare gånger omvandlingsfaktorn faktF.

$$\sum_{n=1}^{\text{nfpack}} nburk\_TB[n,k] - \sum_{j=1}^{\text{ngross}} nburk\_BG[k,j] \geq 0, \quad \forall k \in [1 \ldots \text{nbrygg}] \tag{3.11}$$

Bivillkor (3.11) anger att antalet nya tomförpackningar som varje bryggeri tar emot, ska vara större eller lika med det antal burkar som skickas ut från varje bryggeri till alla grossister. Detta möjliggör för programmet att välja den billigaste väg att skicka ut burkar från bryggerierna med slutdestination butikerna.

$$\sum_{k=1}^{\text{nbrygg}} nburk\_BG[k,j] - \sum_{i=1}^{\text{nbutik}} nburk\_GBu[j,i] = 0, \quad \forall j \in [1 \ldots \text{ngross}] \tag{3.12}$$

Bivillkoret ovan anger att antalet burkar som varje grossist tar emot från alla bryggerier ska vara lika med antalet burkar som skickas ut från varje grossist till alla butiker.

$$\sum_{n=1}^{\text{nfpack}} nburk\_TB[n,k] - \sum_{i=1}^{\text{nbutik}} nburk\_BBu[k,i] \geq 0, \quad \forall k \in [1 \ldots \text{nbrygg}] \tag{3.13}$$

Likaså säger bivillkor (3.13) att antalet burkar som varje bryggeri tar emot från alla tomförpackningstillverkare ska vara större än eller lika med antalet burkar som skickas ut från varje bryggeri till alla butiker.

$$\sum_{k=1}^{\text{nbrygg}} \left(\sum_{j=1}^{\text{ngross}} nburk\_BG[k,j] + \sum_{i=1}^{\text{nbutik}} nburk\_BBu[k,i] \right) = \sum_{n=1}^{\text{nfpack}} \sum_{k=1}^{\text{nbrygg}} nburk\_TB[n,k] \tag{3.14}$$

Bivillkor (3.14) knyter ihop bivillkor (3.11) och (3.13) genom att ange att alla burkar som bryggerierna erhåller från tomförpackningstillverkarna, måste skickas till butikerna antingen via grossisterna eller direkt från bryggerierna. Detta åstadkommes genom att deklarera att summan av bryggeriernas erhållna burkar ska vara lika med summan av de burkar som lämnar bryggerierna.

$$\sum_{j=1}^{\text{ngross}} \sum_{i=1}^{\text{nbutik}} nburk\_GBu[j,i] = \sum_{k=1}^{\text{nbrygg}} \sum_{j=1}^{\text{ngross}} nburk\_BG[k,j] \tag{3.15}$$

Ytterligare ett bivillkor (3.15) krävs för att reglera att alla burkar som grossisterna tar emot, också skickas vidare till butikerna.

$$\sum_{j=1} nburk\_GBu[j,i] + \sum_{k=1}^{\text{nbrygg}} nburk\_BBu[k,i] = behov[i], \quad \forall i \in [1 \ldots \text{nbutik}] \tag{3.16}$$

Slutligen krävs bivillkor (3.16) för att butikernas behov av fyllda burkar ska tillgodoses. Detta sker genom att ange att varje butiks efterfrågan ska vara lika med alla burkar som skickas till butikerna direkt från bryggerierna eller via lagren.

Målfunktionen är uppdelad i tre olika kostnader; rörliga kostnader, fasta kostnader samt materialkostnader, varför dessa kostnader regleras av tre separata bivillkor.

$$
\begin{aligned}
& v\_cost = \\
tomkost \quad & * \Bigg(\sum_{i=1}^{\text{nbutik}} \sum_{j=1}^{\text{ngross}} \Big(cost\_G[i,j] * nburk\_GR[i,j] \Big) + \\
& + \sum_{i=1}^{\text{nbutik}} \sum_{k=1}^{\text{nbrygg}} \Big(cost\_B[i,k] * nburk\_BR[i,k] \Big) \Bigg) + \\
+lagkost \quad & * \sum_{i=1}^{\text{nbutik}} \sum_{k=l}^{\text{nlager}} \Big(cost\_L[i,l] * nburk\_LR[i,l] \Big) + \\
+fullkost \quad & * \Bigg(\sum_{m=1}^{\text{nmatr}} \Big(distRM[m] * nburk\_RM[m] \Big) + \\
& + \sum_{m=1}^{\text{nmatr}} \sum_{n=1}^{\text{nfpack}} \Big(distMT[m,n] * nburk\_MT[m,n] \Big) + \\
& + \sum_{n=1}^{\text{nfpack}} \sum_{k=1}^{\text{nbrygg}} \Big(distTB[n,k] * nburk\_TB[n,k] \Big) + \\
& + \sum_{j=1}^{\text{ngross}} \sum_{k=1}^{\text{nbrygg}} \Big(distBG[k,j] * nburk\_Bg[k,j] \Big) + \\
& + \sum_{i=1}^{\text{nbutik}} \sum_{j=1}^{\text{ngross}} \Big(distGBu[j,i] * nburk\_GBu[j,i] \Big) + \\
& + \sum_{i=1}^{\text{nbutik}} \sum_{k=1}^{\text{nbrygg}} \Big(distBBu[k,i] * nburk\_BBu[k,i] \Big) \Bigg)
\end{aligned}
\tag{3.17}
$$

I (3.17) anges samtliga rörliga kostnader. Tomkost, som representerar kostnaden för att transportera en tomförpackning, multipliceras med burkar från alla flöden skickade mellan butiker till grossister respektive bryggerier. Produkten adderas med lagkost (kostnad att transportera en tomförpackning från butiker till lager och till Returpack) multiplicerad med burkar skickade från butiker till lager. Slutligen adderas kostnader för att skicka alla fyllda burkar eller alla enheter material. Detta utförs genom att summera burkflöden mellan Returpack och materialtillverkarna, materialtillverkarna och förpackningstillverkarna, förpackningstillverkarna och bryggerierna, bryggeriernas till butikerna och grossisterna samt grossisternas fyllda burkar till butikerna. Alla dessa multipliceras med faktor fullkost.

$$
\begin{aligned}
&f\_cost = \\
&\sum_{j=1}^{\text{ngross}} \Big(fkostG[j] * bin\_GR[j] \Big) + \\
&+ \sum_{k=1}^{\text{nbrygg}} \Big(fkostB[k] * bin\_BR[k] \Big) + \\
&+ \sum_{l=1}^{\text{nlager}} \Big(fkostL[l] * bin\_LR[l] * 10 \Big)
\end{aligned}
\tag{3.18}
$$

De fasta kostnaderna (3.18) i räknas ut genom att att summera de fasta kostnaderna för alla grossister, bryggerier och lager som används/byggs. För att inte räkna med fasta kostnader för de som inte byggs har en binär variabel använts som anger om den specifika grossisten, bryggeriet eller lagret används/byggs. Binärvariabeln sätts automatiskt till 1 om det byggs/används, annars tilldelas den värdet 0.

$$
m\_cost = \sum_{m=1}^{\text{nmatr}} \Big(nnytt\_matr[m] * matrkop \Big) \tag{3.19}
$$

Materialkostnaderna i bivillkor (3.19) i räknas enkelt ut genom att multiplicera allt nytt material som köps med kostnaden för att köpa en enhet råvarumaterial.

4 Resultat

I följande tabeller presenteras data från samtliga instanser av optimeringsproblemet. Nedan förklaras några av de förkortningar som används i tabellerna.

N är antalet trädsökningsnoder som gplsol använder för att lösa problemet.
I-LP är antalet simplexiterationer för första LP-relaxationen.
I-tot är antalet simplexiterationer för hela problemet.
nbrygg, ngross och nlager är antalet bryggerier, grossister och mellanlager som används i optimallösningen.
Kostnad A innefattar den totala kostnaden (rörliga och fasta kostnader samt materialkostnader) för återvinningssystemet.
Kostnad B exkluderar de fasta kostnaderna från den totala kostnaden.

Tabell 1: Utan mellanlager

Problem	N	I-LP	I-tot	Tid [s]	nbrygg	ngross	nlager	Kostnad A	Kostnad B
Ex.1	5	28	32	0.0	0	2	0	111 736.6467	111 536.6467
Ex.3	9	35	40	0.0	2	2	0	125 136.9633	125 111.9633
Ex.5	11	152	157	0.0	2	4	0	421 631.2633	421 592.2633
Ex.6	5	804	837	0.0	4	10	0	2 424 744.4489	242 4628.4488
Ex.7	15	3980	4351	2.8	5	30	0	10 743 320.0400	10 743 072.0400
Ex.9	5	7695	8004	16.5	5	30	0	22 278 745.1667	22 278 550.1666
Ex.10	9	3594	6128	17.3	4	26	0	21 233 277.1667	21 233 106.1666

Tabell 2: Med mellanlager

Problem	N	I-LP	I-tot	Tid [s]	nbrygg	ngross	nlager	Kostnad A	Kostnad B	nyttmatr
Ex.1	3	35	39	0.0	0	0	1	104 850.6467	103 850.6467	18.4467
Ex.3	1	33	33	0.0	0	0	1	113 311.96	111 601.96	17.0633
Ex.5	11	132	197	0.0	0	0	2	353 522.2633	338 972.2633	74.4633
Ex.6	15	738	945	0.1	1	0	7	2 049 412.4489	1 995 615.4489	401.027
Ex.7	249	4619	10598	10.4	4	25	10	10 166 749.04	10 084 111.04	1 860.84
Ex.9	231	10537	23621	60.5	5	24	10	20 630 262.1667	20 547 829.1667	3 679.97
Ex.10	3	6950	15034	45.9	1	0	10	17 614 522.1667	17 519 759.166	3 679.97

Tabell 3: Mellanlager som byggs

Problem	Byggda lager
Ex.1	2
Ex.3	3
Ex.5	1 & 3
Ex.6	2, 3, 4, 5, 6, 9 & 10
Ex.7	1, 2, 6, 10, 11, 12, 13, 16, 19 & 20
Ex.9	1, 7, 18, 29, 31, 33, 39, 57, 59 & 75
Ex.10	1, 12, 13, 31, 59, 64, 68, 71, 83 & 87

Tabell 4: Inget spill med mellanlager

Problem	N	I-LP	I-tot	Tid [s]	nbrygg	ngross	nlager	Kostnad A	Kostnad B	nyttmatr
Ex.1	3	35	39	0.0	0	0	1	105 334.67	104 334.67	14.6667
Ex.3	1	33	33	0.0	0	0	1	114 918.0733	113 208.0733	8.87333
Ex.5	19	135	220	0.0	0	0	2	362 777.0433	348 227.0433	40.4433
Ex.6	27	734	1154	0.1	2	1	7	2 101 862.8189	2 048 051.8189	223.997
Ex.7	87	4620	7119	6.3	4	25	10	10 489 570.39	10 406 932.39	1070.19
Ex.9	195	10188	28082	64.8	5	27	10	21 347 417.9667	21 268 109.9667	2152.85
Ex.10	7	7510	14867	46.5	1	2	10	18 095 542.9667	18 004 791.9667	2152.85

Tabell 5: Inget spill - 100 gånger dyrare råvarumaterial

Problem	N	I-LP	I-tot	Tid [s]	nbrygg	ngross	nlager	Kostnad A	Kostnad B	nyttmatr
Ex.1	3	35	39	0.0	0	0	1	106 786.6667	105 786.6667	14.6667
Ex.3	1	33	34	0.0	0	0	1	115 796.5333	114 086.5333	8.87333
Ex.5	19	135	220	0.0	0	0	2	366 780.9333	352 230.9333	40.4433
Ex.6	27	734	1155	0.1	2	0	7	2 101 862.8189	2 048 051.8189	223.997
Ex.7	87	4620	7112	6.4	4	25	10	10 595 519.2	10 512 881.2	1070.19
Ex.9	195	10188	28080	65.3	5	27	10	21 560 549.7867	21 481 241.7867	2152.85
Ex.10	6	7510	19242	54.3	0	0	10	18 283 298.7867	18 194 798.7867	2152.85

Tabell 6: Skillnad mellan kostnader om man bygger mellanlager

Problem	Δ Totala kostnader	Δ Rörliga kostnader samt material	Δ Fasta kostnader
Ex.1	6886	7686	- 800
Ex.3	11 825	13 510	- 1685
Ex.5	68 109	82 620	- 14 511
Ex.6	375 332	429 013	- 53 681
Ex.7	576 571	658 961	- 82 390
Ex.9	1 648 483	1 730 721	- 82 238
Ex.10	3 618 755	3 713 347	- 94 592

Tabell 7: Skillnad i lönsamhet av att minska spillet till 0, för ordinarie material-kostnad

	Kostnad då spill förekommer		Kostnad vid inget spill		Δ Totalt	Δ Rörligt
Problem	Totalt	Rörligt	Totalt	Rörligt		
Ex.1	104 850.6467	103 850.6467	105 334.6667	104 334.6667	-484.02	-484.02
Ex.3	113 311.96	111 601.96	114 918.0733	113 208.0733	-1606.1133	-1606.1133
Ex.5	353 522.2633	338 972.2633	362 777.0433	348 277.0433	-9254.78	-9304.78
Ex.6	2 049 412.4489	1 995 615.4489	2 101 862.8189	2 048 051.8189	-52 450.37	-52 436.37
Ex.7	10 166 749.04	10 084 111.04	10 489 570.39	10 406 932.39	-322 821.35	-322 821.35
Ex.9	20 630 262.1667	20 547 829.1667	21 347 417.9667	21 268 109.9667	-717 155.8	-720 280.8
Ex.10	17 614 522.1667	17 519 759.1667	18 095 542.9667	18 004 791.9667	-481 020.8	-485 032.8

Tabell 8: Skillnad i lönsamhet av att minska spillet till 0, för 100 ggr dyrare materialkostnad

	Kostnad då spill förekommer		Kostnad vid inget spill		Δ Totalt	Δ Rörligt
Problem	Totalt	Rörligt	Totalt	Rörligt		
Ex.1	106 676.8667	105 676.8667	106 786.6667	105 786.6667	-109.80	-109.80
Ex.3	115 001.2333	113 291.2333	115 796.5333	114 086.5333	-795.30	-795.30
Ex.5	360 894.1333	346 344.1333	366 780.9333	352 230.9333	-5 886.80	-5 886.80
Ex.6	2 0891 14.0889	2 035 317.0889	2 101 862.8189	2 048 051.8189	-12 748.73	-12 734.73
Ex.7	10 350 972.2	10 268 334.2	10 595c519.2	10 512 881.2	-244 547.00	-244 547.00
Ex.9	20 994 578.8667	20 912 145.8667	21 560 549.7876	21 481 241.7867	-565 970.92	-569 095.92
Ex.10	17 978 720.8667	17 884 467.8667	18 283 298.7867	18 194 798.7867	-304 577.92	-310 330.92

5 Diskussion

5.1 Effekten av byggnation av mellanlager

I tabell 6 kan vi utläsa att vid byggnation av mellanlager ges ett negativt Δ i fasta kostnader samt att den huvudsakliga kostnadsskillnaden ligger i de rörliga kostnaderna. Detta tyder på att byggnationen av mellanlager leder till högre fasta kostnader eftersom lager är dyrare att bygga än det är att använda grossister och/eller bryggerier. Den totala kostnadssänkningen tillskrivs de sänkta rörliga kostnaderna som uppkommer då tomburkar skickas till mellanlager, detta då distanserna blir kortare vilket i sin tur sänker miljöpåverkan.

5.2 Att panta eller inte panta...

Som framgår av tabell 7 ökar både de totala och rörliga kostnaderna när alla burkar pantas (spill noll). Att de rörliga kostnaderna ökar är inte förvånande, då fler pantade burkar leder till bland annat ökade transportkostnader. Men det är inte lika intuitivt att de totala kostnaderna ökar eftersom fler pantade burkar leder till att mindre mängd nytt material behöver köpas in. Tolkningen av detta är att materialkostnaden för det nya materialet är så pass liten i förhållande till ökningen av de rörliga kostnaderna. Därför modifierades modellfilen genom att öka inköpskostnaden för nytt material med en faktor 100. Som tabell 8 visar, innebär den väsentligt högre materialkostnaden fortfarande att kostnadsminskningen är mycket mindre i förhållande till kostnadsökningen av de rörliga kostnaderna. Den slutsats som möjligtvis kan dras av detta är att indata över priser och kostnader inte är särdeles verklighetsförankrade.

Till sist kan nämnas att utifrån ett ekonomiskt perspektiv innebär detta resultat att återvinningssystemet borde avvecklas eftersom varje pantad burk leder till högre kostnader. Men målsättningen med återvinningssystemet är inte monetärt utan fokus är på resurseffektivitet och miljöaspekter.

5.3 Osäkerhetsanalys

I de fall då data från problem 10 användes tog problemen för lång tid att lösa (över 20 minuter). Därför lades kommandot mipgap till vid anropandet av modellfilen. Dock medför detta att osäkerheter i modellen införs då approximativa kapningar i trädsökningen tillåts. Av nyfikenhet gjorde vi ett testförsök på problem 10 utan att använda approximering vilket gav en felprocent på under 1 %. Detta anser vi vara en försumbar skillnad av resultatet och osäkerhetsfaktorn kan därmed ses som liten. Däremot fås inte optimum vilket hade kunnat erhållas om datorn tillåtits arbeta en längre tid. Allt handlar om vilken tid beställare har att lägga på datorns uträkningar, i vissa fall behövs snabba resultat och då kan en felprocent vara acceptabel medan de i andra fall inte kräver snabba svar utan kan låta datorn optimera klart utan approximeringar.