Compound Regular Plans and Delay Differentiation Services for Mobile Clients

JOHN TSILIGARIDIS
Math and Computer Science
Heritage University
3240 Fort Road, Toppenish, WA 98948
USA
tsiligaridis_j@heritage.edu

Abstract: - The broadcast problem including the plan design is considered. Data can be reached at any time and place. A larger number of users are asking for services. The Regular Broadcast Plan (RBP) can be created with single or multiple channels after some data transformations are examined. A server with the data-parallelism can administrate more than one plans and before broadcasting any of the plans it can consider the case of union ready or candidate RBPs. The case of union of several candidate or ready RBPs is examined and useful results are provided. A new RBP algorithm based on the group length is introduced (RBPG). A set of algorithms related to the creation of a compound regular broadcast plan (CO_RBP) and their possibilities are developed. CO_RBP is based on the "hot" and "near hot" sets. The conditions of preferring CO_RBP to separate RBPs is also presented. Two types of CO_RBP are examined; the additive RBPs (ARBPs) and the union RBPs (URBPs). This proposed framework gives service priority to the "hot or "near hot" sets. In addition, a Dimensioning algorithm (DA) based on the delay differentiation services and conditions that can guarantee the desired ratio for the CO_RBP are developed. In this way the server, has an increasing of broadcasting capability ,by deciding on the union of RBPs using single or multiple channels. This ability can enrich the server infrastructure for self-monitoring, self-organizing and channel availability. Simulation results are provided.

Key-Words: - Broadcasting, broadcast plan, mobile computing

1 Introduction

A lot of work has been done for the data dissemination with flat and skewed design using the broadcast disks [1], [2], [3], [4]. The problem of optimal broadcast program generation is examined in [9]. In [10] the study of the broadcast scheduling problem taking user impatience into account is developed. Techniques that handle concurrency control of transaction processing create a more realistic methodology for services in wireless information system in [11]. The dissemination of data to mobile clients considering the location dependent queries (LDQ) along with the D-Tree design is developed in [12]. The regular design with the equal spacing property [1], can provide broadcasting for single and multiple channels with average waiting time less than the one of the flat design. It also provides channel availability and less energy consumption. The minimum time broadcast problem has been addressed by computing the minimum degree spanning tree of directed acyclic graphs in [5]. The construction of compound regular plans must be under consideration since new plans are continuously created with short time interval between them, special attention is needed to reuse the circle being broadcasted and to avoid the process of creating new RBP again. Instead of sending data serially (of different candidate RBPs), the compound solution is more appropriate given that the waiting time is not over a predefined value. Server prepares plans and sends them over the air one after the other. Before an RBP is broadcasting the server has to decide for the creation of a compound candidate RBPs. When a server has many data to be served, instead of waiting for the end of the transmission of a circle in order to send the next circle, the case of including in the same circle more data from other next circles is examined. This provides the case of the "compound" regular plans. When the additional cost (waiting time) is not over a predefined limit the nested regular plans can provide a solution. Sending bigger size RBP, that comes from unification of other sub RBPs, the server can serve more users and with lower cost of preparation (i.e. queues etc). The serial regular plans increase the access time of the hot or near hot data of the RBPs. By following compound regular plans this can be diminished, since only a short increasing size is added to the initial RBP and data that belong to n RBP can be published with the data of the first RBP.

Some of long circle messages delay all the others and the service rate needs adjustment depending on the size of the message and the available amount of bandwidth the server can provide. Small size messages delay until an available server starts their service or to end the serve of the previous circle. In addition, since now the server can work fast simultaneously with several candidate plans (multiprocessing) it seems less time consuming to unify the candidate plans (independently of their size) into one RBP, (the CO_RBP), instead of creating several RBPs. The purpose of this work is to show that a server before sending an RBP can decide whether unification with next or ther RBPs is needed so that their hot and other type of data are included in the current one. The main focus is the unification step and the delay differentiation of services for RBPs and CO_RBPs. .

Servers with the ability of the data-parallelism can partition the work of RBP in multiple cores [8]. For this reason a reconsideration of the broadcasting plan problem is needed so that to decrease the delay of messages. Hot data that are to be served in the n cycle can be served in the n-1 cycle. This ability provides another dimension to the server. More users can be served in a circle. The compound of RBPs (CO_RBP) contains the two cases the additive (ARBP) and union (URBP) and can provide solution. It is assumed that there is availability of “cold data” so that, with some increment or decrement of the last set size (S_{3s}) the server can provide solution to the compound RBPs. The remainder of the “cold” data can be sent at the next cycle. There is also an RBP based on the group length (RBPG) which can be used for preparing a RBP. RBPG is used also for discovering the parameters of RBP for the CO_RBP.

The paper is organized as follows. In Section 2, the model description is given. In Sections 3, 4, 5 and 6 the BRA, the PVALD, and DA and are developed, respectively. Finally, simulation results are provided in Section 6.

2 Model Description

2.1 The relations in the RBP

The possibility of providing BP (full or not) is examined iteratively starting from the last level of hierarchy S_3. The *size of a set* stands for S_{is} (where i=1,2,3). It is considered that $S_{3s} \geq S_{2s} \geq S_{1s}$, and the number of S_3 items will be sent only *once* while for the other sets at least twice. We create a set of relations including their subrelations by considering items of different size from each set. This is achieved by finding the integer divisors of S_{3s} (k_1, k_2, k_3,..k_i…k_n) and put them at a decreasing order in an array (ar). Each relation has three *subrelations*. It is also assumed that S_{2s}, S_{3s} are not prime numbers. For the BP design in case that S_{2s} is a prime number, it is possible to add only one empty slot at the end of the last major cycle. The next integer number of a prime is a composite number. This idea helps to create the BP. The following definitions for single RBPs are essential:

Definition 1: The *size (or horizontal dimension) of a relation (s_rel)* is the number of items that belong to the relation and it is equal to the sum of the size of the three subrelations (s_rel= $\sum_{i=1}^{3} s\_sub_i$). The *number (or vertical dimension) of relations (n_rel)* with s_rel define the *area of the relations (area_rel).*

Example 1: The relation A=(a, b, c, d, f) has the following three subrelations starting from the end one; the 3-subrelation (f) with $s\_sub_3 = 1$, the 2-subrelation (b,c,d) with $s\_sub_2 = 3$, and the 1-subrelation (a) with $s\_sub_1 =1$. The s_rel=5

Definition 2: The *area of the i-subrelation (area_i_sub)* is defined from its size (s_sub_i) and the number of the relations (n_rel) that are selected. It is given by (s_sub_i) x (n_rel).

Example 2: From a relation with s_rel=5 and if n_rel=5 then the area of this relation is 5x 5 . Hence there are 25 locations that have to be completed.

Example 3: If two relations are: (1,2,3,5,6,7), (1,3,4,8,9,10) with s_sub_3=3, s_sub_2=2, then : 2-$subrelation_1$ =(2,3) and 2-$subrelation_2$=(3,4). The last two subrelations ((2,3),(3,4)) comes from S_2 ={2,3,4} having 3 as repeated item.

Definition 3: An BP is *full* if it provides at least 2 repetitions of items and it does not include empty slots in the area_rel

Definition 4: The number of items that can be repeated in a subrelation is called *item multiplicity (it_mu) or number of repetitions (n-rep).*

Definition 5: Integrated relations (or integrated grouping) are when after the grouping, each group contains relations with all the data of S_2 and S_1. This happens when: ($\cup$ (2_subrelation) = S_2) $\wedge$ ($\cup$ (1_subrelation) = S_1). See example 7 for details.

Definition 6. The SD_4 is the set of divisors of the size of the last set.(S_{4s})

Example 4: If S_{4s} =120 the SD_4 = {10,20,30,40,60}. Moreover, gl $\in SD_4$. The symbol d_4 represents any divisor of S_{4s} ($d_4 \in SD_4$) while gl is the final value of d_4 that makes the candidate RBP feasible. For an RBP, the gl term is used.

Grouping length(gl): The gl is a divisor of S_{ks} (1,..,k). It is the n_rel that can provide homogenous grouping.

Partition value (pv): It is the common divisor of S_{is} (i=1,.., k) and gl for a given size of s_sum_i. Hence: $pv_i \mid S_{is}$ and $pv_i \mid gl$. Each set must have its own pv.

Example 4: If S_{3s} =40, gl=20, considering that s_sum_3=8 then pv_3 =5 (=40/8) . Hence $pv_3 \mid S_{3s}$ and $pv_3 \mid gl$

The criterion of homogenous grouping(chg): when $pv_i \mid gl$.

The criterion of multiplicity constraint(cmc) or differential multiplicity: This happens if: $it\_mu_{i+1} < it\_mu_i$ (i= 1,..,n-1).

The criterion of PV (cpv): when: $pv_i < pv_j$ (for i<j).

The chg along with cpv can guarantee the cmc for different multiplicity (Theorem 1) and because of that the cmc is not necessary to be examined.

The pv criterion can guarantee differential multiplicity service. For having an RBP the criterion of chg along with pv have to be held.

The number of channels (nc): S_k / gl (where S_k is the last set)

It is considered that a|b (a divides b) only when b mod a =0 (f.e. 14 mod 2=0). The relation with the maximum value of n_rel provides the opportunity of *maximum multiplicity* for all items of S_2 and S_1 and finally creates the *minor cycle* of a full BP. The *major cycle* is obtained by placing the minor cycles on line.

The criterion of homogenous grouping(chg): when $pv_i \mid gl$.

The criterion of multiplicity constraint(cmc): This happens when: $it\_mu_{i+1} < it\_mu_i$ (i= 1,..,n-1).

The PV criterion: when $PV_i > PV_{i+1}$

The number of channels (nc or n_ch): S_k / gl (where S_k is the last set)

It is considered that a|b (a divides b) only when b mod a =0 (f.e. 14 mod 2=0). The relation with the maximum value of n_rel provides the opportunity of *maximum multiplicity* for all items of S_2 and S_1 and finally creates the *minor cycle* of a full BP. The *major cycle* is obtained by placing the minor cycles on line.

For more than one RBPs new definitions are needed:

Definition 7: The sets for RBPs stands for $S_{i,j}$ (i: is the number of set, j: is the number of RBP) For example: the two "hot" sets for two RBPs are : $S_{1,1}$, $S_{1,2}$, $S_{2,1}$, $S_{2,2}$.

Definition 8: Unified sets: (usets) are $US_{1,s} = S_{1s,1} + S_{2s,1}$

Definition 9: The $s\_sub_{i,k}$ stands for the i sub_relation (s_sub_i) for the k RBP.

Definition 10: The npv_i : is the new pvi after the addition or union of two or more RBPs

Definition 11: The candidate regular plan (RBP) is a broadcast plan which will be examined whether it can become a RBP. The ready RBP has already be computed and the parameters are known.

Definition 12: a normal RBP is a the RBP that is only one RBP while the nested RBP (NRBP) contains >1 RBPs

Definition 13: apv_i : stands for the pvi for the addition of RBPs and upvi:for the union of RBPs.

The next example below uses the various parameters for RBP construction and illustrates pv_i.

Example 5: (the pvi) Let's consider four sets S_1,S_2,S_3,S_4 with S_{1s}=10, S_{2s}=20,S_{3s}= 40, S_{4s} =120. If gl =20 (20 is a divisor of 120) then S_{1s} / gl, S_{2s} / gl, gl / $S_{3s.}$ The chg exists. The number of channels is: nc=120/20= 6. Considering s_sum_1 = 5, s_sum_2=5, s_sum_3=8 then pv_1 = 10/5=2, pv_2= 20/5=4, pv_3 =40/8=5. We have $pv_1<pv_2<pv_3$ (pv criterion) and since $pv_1|20$, $pv_2|20$, $pv_3|20$ (or $d_4 \mid pv_i \in I$) then the chg is valid and an RBP can be constructed. From this process it is evident that there is no need to test the cmc .

The condition for an ARBP is when the two (or more) RBPs have the same i_sub relations ($s\_sub_{i,k} = s\_sub_{i,m}$, where: i: set , k, m: the two RBPs.)

By contrast with it, the URBP works, with any size of sub-relations.

It is supposed that the chg, cmc, cpv criteria are valid for both RBPs and the CO_RBP.

There are two algorithms for finding the URBP. The URBP1 based on pvi (partial rearrangement) and sum and the URBP2 based only on sum of the set's data (total rearengemant). For URBP2 it is not necessary to create first the two RBPs but it can works directly having the total sets of the two candidate RBPs. This is an advantage of URBP2 over the URBP1. The URBP2 works according to RBPG. Before broadcasting an RBP, the server has to detect the possibility of union with RBPs of various sizes.

2.2 The RBPG

The RBPG,which is the basis of the CO_RBP, can create an RBP and is developed in about the same manner as BRA (another algorithm for

RBP creation) [6] but it has a different way of finding the group size. It has three steps: (a) discovering the pvi so that the cpv is valid and (b) discovering the gl so that the chg is valid and (c) the n_ch will be available. More details on Partition Value Algorithm are also in [6].

```
RBPG: input: S1,S2,S3,  Sis (i ≤4), n_ch: the # of channels, ,av_ch:#avail.chan.  for an RBP
output: the RBP homogenous grouping for multiple channels
//find the common divisors of the S1s,S2s,S3s (in increasing  order)
 // D2 for S2, D1 for S1, D3 for S3
 //d2∈ D2,  d1∈ D1, d3∈ D3
for each s_sum (s_sumi,s_sumi = di (i<3)) (a)
     for all Si (i≤4)
   {  //define the s_sumi = di  (i<4)
      s_sumi = di (i<4)
      pvi =Sis / s_sumi
      //pv criterion
      if (pvi< pvi+1)
         {"the pv  criterion is valid "
      else {go to (a)
      //finding gl
      m=max(pvi); ind = argmaxi (pvi)
      k=2; n=k*m; gl=0;   //find gl            (b)
      bool divisible = false;
      while (divisible)
        {
          if (n | S3s)
             {gl=n; print gl;
               divisible=true; }
          else
            { k=k++; n=k*m;}
      //finding n_ch
      if  (gl | S3s)
          n_ch= S3s / gl;
      else {go to (a) , new s_sum}
      if (n_ch <= av_ch)
           n_ch=nn
       else {go to (b), new gl}

}
```

Example 6: For RBP with $S_{1s}=16$, $S_{2s}=32$, $S_{3s}= 120$ with $s\_sum_1 =8$, $s\_sum_2=8$ and $pv_1=2$ and $pv_2=4$ and av_ch = 11. Applying RBPG we find (from n=k*m, k=2) for n=max(pvi)=8 , and n_ch =15 (120/8). But n_ch<av_ch. Applying again for k=3 then n=12 and n_ch = 10 (120/12). Since n_ch <av_ch the n_ch=10. The criterion of chg is valid (pvi|gl), the cmc is valid ($it\_mu_{i+1}< it\_mu_i$). the cpv is also valid. The case of cumulative RBPs will be examined next.

ARBP is not feasible since $s\_sum_{i,1} \neq s\_sum_{i,2}$.

2.3 Some Analytical Results

Let us consider, for the compound RBP that npv_i is the new pv_i that comes from two or more RBPs, and $pv_{i,m}$ is the pv for i sets and m RBPs.

For the URBP there are two cases: (a) when there are two (or more) candidate RBPs and the unification is examined (URBP1) using their pvi , (b) when there are two (or more) candidate RBPs that need to be unified (URBP2) using factorization. Theorem 1 provides a URBP1 by searching (brute force) of the correct values of pv. On the contrary Theorem 2 works with factorization of S1, S2 (direct solution). For ARBP, Theorem 3 provides one approach.

Theorem 1 : (using the pv-for URPB1-) In a URBP1 the upvi can be discovered from two RBPs , given the $pv_{i,j}$ (i= #set, j= #RBPs) when the upv_i is equal to the minimum number that can be divided by pv_{ij} ($pv_{i,j} \mid upv_i$). Finally the upv_i can divide the us_sum (= $S_{1s,1} + S_{1s,2}$) .

Proof: Let us consider that there are two subsets $S_{1,1}$, $S_{1,2}$ with their $pv_{i,j}$. The $pv_{1,1}= a_1*a_2*a_3..*a_k$, (product factorization) and $pv_{1,2} = b_1*b_2*…*b_m$. The product factorization provides the divisors [7]. If $a_i=b_i$ and the common factors number is c then there is a multiplier m and the number: c*m so that both $pv_{1,1}$ and $pv_{1,2} \mid c*m$ (1). If (1) is valid then there is a upv_i for the subsets, otherwise the pv1,1 and pv1,2 can not produce a upv_1.

Finally the total # of data for two sets is us_sum_i= sum ($S_{is,1}$ + $S_{is,2}$). After finding the upv_1 the us_sum_i = sum ($S_{is,1}$ + $S_{is,2}$) | upv_i. This is the method of URBP based on sum and the pv_i. •

Example 7:(using the common factor) Lets consider the two sets: $S_{1s,1}$ = 40, $S_{1s,2}$ = 80. (us_sum_1=120), with their: $pv_{1,1}$ =8 and $pv_{1,2}$=10 respectively. The $pv_{1,1}$= 2*2*2 , $pv_{1,2}$= 2*2*5. The common factor is c=2*2=4. The multipliers (m) of 4: (a) 4*2 = 8, 8| $pv_{1,1}$ but 8 □ 16, (b) 4*3=12 , 8,10 □ 12, (c) 4*4=16 , 8|16,10□16, …(d) 4*10=40, and 8,10 | 40. Hence there is a upv_1= 40 and upv_1| us_sum. (=3). That means that there is a valid upv_1 and ns_sum_1= 40 (120/3).

Theorem 2: (direct use of pv –for UPRB1-) The minimum common number after the factorization of pv_{ij} can be used as upvi

Proof. The minimum common number of pv_{ij} can divide not only each of the S_{ijs} but also the summation of them •

Example 8: (using part of the common factor) Lets consider again the same sets as the previous example: $S_{1s,1}$ = 40, $S_{1s,2}$ = 80. (us_sum_1=120), with their: $pv_{1,1}$ =8 and $pv_{1,2}$=10 respectively. The $pv_{1,1}$= 2*2*2 , $pv_{1,2}$= 2*2*5. Considering only 2 as the upv1 then 2| 40,80 and 120. the us_sum1 = 60 (120/2).

Theorem 3: (direct factorization –for UPRB2-) For the creation of URBP from two RBPs , the upvi , can be computed by using the number factorization and defining first the s_sum_i (from the non common factors) and then the upvi (from the common factors).

Proof: the URBP2 can be created after the creation of the usets and getting the common and non common factors. •

Example 9: For the URBP2 usets: US_{1s} =50(18+32), US_{2s}= 28(12+16) with S_{3s} =120 (is the same for both RBPs. Analyze the numbers into primes (50= 2*5*5 and 28=2*2*7) with common factor. From URBP2 then s_sum_1 =25, s_sum_2= 7 (from non common factors) with pv_1=2, pv_2=4 (from the common factors. The gl can be found from the RBPG as gl=8, n_ch= 15(120/8). All the criteria (chg, cmc, cpv) are valid.

Theorem 4: For the creation of ARBP from two ready RBPs, if the twoRBPs have the same pvi then the URBP can have the same pvi and s_sumi the sum of them

Proof: For the RBP1 for set S1: pv1,1 * s_sum1,1 = Sis,1. (2) For the RBP2 for set S2: $pv_{1,2}$ * s_sum1,2 = $S_{is,2}$.(3), and $pv_{1,1}$=$pv_{1,2}$.

By adding (2)+(3): US_1 = $S_{1s,1}$ +$S_{2s,2}$ = $pv_{1,1}$ ($s\_sum_{1,1}$ + $s\_sum_{1,2}$) •

Example 10: For the sets of the two RBPs: $S_{1,1}$=18, $S_{1,2}$=12, $S_{2,1}$=12, $S_{2,2}$=16 with $pv_{1,1}$=2, $pv_{2,1}$ = 4, $pv_{1,2}$=2, $pv_{2,2}$=4 and $s\_sum_{1,1}$= 9, $s\_sum_{1,2}$=3, $s\_sum_{2,1}$=16, $s\_sum_{2,2}$=4. The URBP2 for the two RBPs has the upv_1 = 2 and s_sum_1 = 9+16=25, (2*25=50= 18+32)=and upv_2 = 4 and s_sum_1 = 3+4=7. (4*7=28=12+16)

Similarly, in case that the two RBPs have the same s_sum_i , then the ARBP has the common s_sum_i and as upv_i the sum of partial pvi.

The next theorem is related to the relation between the pvi and the delay differentiation of services (a_1,a_2,a_3..,a_{k-1}, for k services). This is useful for the delay differentiation for an RBP and for the CO_RBP.

Theorem 5: If pv_i (i<k, k =#sets) are analogous to a_i the AWT_i are also analogous to the a_i and pv1 / AWT_1 = pv_2/AWT_2 =…= pv_{k-1} / AWT_{k-1} (3).

Proof: Let us consider n=4 (the number of sets) and $pv_1/a_1 = pv_2/a_2 = pv_3/a_3$.

Finding AWT_1 (if pv_1=2) AWT_1= s_sum*pv1 =((s_sum_1-1+s_sum_2 + s_sum_3 + s_sum_4) + (s_sum_1 + s_sum_2+ s_sum_3 +s_sum_4) + 1)). In analogous way AWT_2 = s_sum*pv_2. and AWT3= s_sum*pv_3 . For n=k , the hypothesis is : $pv_1/a_1 = pv_2/a_2 = pv_3/a_3 =..= pv_{k-1}/a_{k-1}$ (4) and AWT_{k-1} =s_sum*pv_{k-1}. The equivalence is: pv1/AWT1 = $pv2/AWT_2$ = pv_3/AWT_3 =…= pv_{k-1} / AWT_{k-1} (5). Moreover , dividing the ratios (4),(5) : $AWT_1 /a_1 = AWT_2 /a_2$ … AWT_{k-1}/a_{k-1} . •

3 The Compound Framework (CO_RBP)

The CO_RBP is the framework that uses the URBP1, URBP2, URBP3. CO_RBP is the framework from which any unification of RBP starts. For the implementation of any of URBP1, URBP2, ARBP all or part of CBRP is needed. URBP1 is based on Theorem1 and Theorem2 and URBP2 is based on Theorem3.

```
CO_RBP: input: S1,S2,
output: the URBP1
if  (s_sum 1.1 = s_sum1,2  and s_sum1.1. =
     s_sum2,2) or
    (pv1,1 =pv1,2 and pv2,1 =pv2,2)
    then apply ARBP (Theorem 3)
else
   URBP1 (using pvi,j)  (Theorem 1)
   URBP2 (using factorial)  (Theorem 2)
```

```
URBP1: input: S1,S2, pvi,j
output: the URBP1
for the pvi.j                  (A)
find the candidate upvi
//test for the final upvi
if  upvi |  sum (Sis,1 + Sis,2) //theorem1
    upvi is valid
else
    go to (A) //for finding new pvij
```

```
URBP2
input: S1,S2,
output: the URBP2
find US1s, US2s
factorization of  US1s, US2s
 find  npvi
 //work same as RBPG
```

Example 11: Let us consider the S_{1s}=18, S_{2s}=12,S_{3s}= 120 , (candidate RBP1) and the S_{1s}=32, S_{2s}=16,S_{3s}= 120 (candidate RBP2). Applying the RPBG for RBP1 we find: s_sum_1 =9, s_sum_2=4 with pv_1=2 and pv_2=3. Considering as gl=6 the n_ch =20 (120/6).
Apply again RBPG for RBP2 we find s_sum_1 =16, s_sum_2=4 with pv_1=2 and pv_2=4. Considering as gl=12 the n_ch =10 (120/12). The criterion of chg is valid (pvi|gl), the cmc is valid (it_mu_{i+1}< it_mu_i) and the cpv is also valid. The case of cumulative RBPs is examined. For ARBP the solution is not feasible since $s\_sum_{i,1} \neq s\_sum_{i,2}$.
For the URBP2 usets: US_{1s} =50(18+32), US_{2s}= 28(12+16). Analyze the numbers into primes (50= 2*5*5 and 28=2*2*7) with common factor (cf). From URBP2 then s_sum_1 =25, s_sum_2= 7 with pv_1=2, pv_2=4 . the gl=pv_1*pv_2 =8, n_ch= 15(120/8).
All the criteria (chg, cmc, cpv) are valid.

4 The ARBP

For ARBP (from Theorem 3) if for two RBPs that pv_{ij} are the same , then an ARBP with s_sum_i is given by the addition of all the same set subdivisions as: $s\_sum_i = s\_sum_{i,1} + s\_sum_{i,2}$.
Example 12: Let us consider the S_{1s}=6, S_{2s}=24,S_{3s}= 120 , (candidate RBP1) and the S_{1s}=8, S_{2s}=32,S_{3s}= 120 (candidate RBP2).Applying RBPG for RBP1 s_sum_1= 2, s_sum_2=4 with pv_i=3 and pv_2=6. The gl= 12 and n_ch= 10 (120/12). The gl= 12 and n_ch= 10 (120/12). Applying RBPG for RBP2 s_sum_1= 2, s_sum_2=4 with pv_1=4 and pv_2=8.
Applying the ARBP for both RBPs we find: s_sum_1 =2, s_sum_2=4 with pv_1=7 (3+4)and pv_2=14 (6+8). The gl =28 and n_ch= 4.28 ($\notin$I). For the increasing size of S_{3s} an additional amount of S_3 data (20) is needed so that n_ch =5(120+20=140/28). On the contrary for the decreasing size of S_{3s} (8) n_ch=4(120-8=112/28)

5 The Dimensioning Algorithm (DA)

The DA is very useful for finding the AWTi by applying the Theorem 2. In addition any change to the integrated relation (s_sum) or any subrelation (s_sub_i) can easily be translated into delay. This is very important for the server making decision process and for having successful differentiation of services.

DA: *input*: s_sum_i (i<k, k:#sets), PA_i/a_i : the desired ratio
output: RBP with desired AWTi ratio
for each s_sum (s_sum_i , $s\_sum_i = d_i$ (i<4)) (a)
  for each divisor (d_4) of set S_4 (b)
 //find PV_i
 $PV_i = S_{is}$ / s_sumi
 if $pv_i \mid d_4$
   {chg criterion is valid}
 else {go to (b) }
 if ($pv_1 < pv_2 < \ldots < pv_{k-1}$)
  {the pv criterion is valid}
 else {go to (b) }
 if ($pv_1/a_1 = pv_2/a_2 = \ldots = pv_{k-1}/a_{k-1}$)
  {$AWT_1 /a_1 = AWT_2 /a_2 = . . = AWT_{k-1}/a_{k-1}$
  there is a RBP with the predefined ratio}
 else {go to (b) }
 if (there is not an RBP for all d_4 –b-)
   {go to (a) , new s_sum}

Example13: Let's consider: S_{1s}=10,S_{2s}=20, S_{3s}=40, S_{4s} =120. The divisor of S4s are: SD_4={10,20,30,40}. The purpose is to see for: AWT1/ 2 = AWT2/4 = AWT3/8. For d_4 =10 and s_sum_1=5, s_sum_2=5,s_sum_3=5, the pv1 = 10/5 = 2, it_mu1=10/2=5, pv2= 20/5 = 4, it_mu2= 10/4 $\notin$ I, pv3=40/5=8 , and it_mu3= 10/8 $\notin$ I (pv criterion no t valid). For d_4 =20 and s_sum_1=5, s_sum_2=5,s_sum_3=5, the pv1 = 10/5 = 2 , it_mu_1=20/2=5, pv_2= 20/5 = 4, it_mu_2= 20/4 =5, pv3=40/5=8 , and it_mu_3= 40/8=5 (pv criterion is valid). Also $pv_1 < pv_2 < pv_3$ (2<4<8) so the pv criterion is valid.
The pv ratio is: $pv_1/2 = pv_2/4 = pv_3/8$ give the $AWT_1/2 = AWT_2/4 = AWT_3/8$.

6 The CO_RBP Delay Differentiation

For the ARBP, if two RBPs have the same values of pvij then the unified RBP will have the delay differentiation of the composed RBPs.
Example 14: For RBP1: S_{1s}=18,S_{2s}=12, S_{3s}=120, and s_sum_1=7, s_sum_2=4, pv_1=2,pv_2=4, gl=12
For RBP2: S_{1s}=32,S_{2s}=16, S_{3s}=120, and s_sum1=16, s_sum_2=4, pv_1=2,pv_2=4, gl=12
For ARBP: S_{1s}=34 (18+16), S_{2s}=28(12+16), and s_sum_1=25(9+16), s_sum_2=7(3+4), pv_1=2,pv_2=4, gl=12.
Following the DA and Theorem 5 there are differentiated services: $AWT_{1,1}/2 = AWT_{1,2}/4$ (from RBP1) , $AWT_{2,1}/2 = AWT_{2,2}/4$ (from RBP1) and the same ratio $UAWT_1/2 = UAWT_2/4$ (from ARBP)

For URBP1, Theorem 2 provides the details for the desired delay ratio of services. Scenario 4 of the simulation is referred to that. The Delay Ratio (DR) of a RBP is the ratio: AWT2 /AWT1.

6 Simulation

For our simulation, Poisson arrivals are considered for the mobile users' requests. The items are separated into four categories according to their popularity using Zipf distribution. Four scenarios have been developed:

Scenario 1: In this scenario the use of URBP for two RBPs is examined. The use of URBP. Considering two RBPs, RBP1: $S_{1s,1}$ = 18, $S_{2s,1}$=32,$S_{3s,1}$=120. and RBP2: $S_{1s,2}$ = 12, $S_{2s,2}$=16, and the same $S_{3s,1}$=120. Using RBPG for RBP1 : $s\_sum_{1,1}$= 9, $s\_sum_{2,1}$= 8, $pv_{1,1}$ =2, $pv_{1,2}$=4 the AWT1 = 36 ((8+8+1)+ (9+8+1)+1).

Using RBPG for RBP2 : $s\_sum_{1,2}$= 4, $s\_sum_{2,2}$= 4, $pv_{1,2}$ =3, $pv_{2,2}$=4 the AWT1 = 27((3+4+1)+ (4+4+1)+(4+4+1)+1). For the URBP2 (Theorem 3) the URBP2 usets: US_{1s}= 18+12 =30 (2*3*5), US_{2s}=16+32=48 (2*3*4*2).Analyze into primes and get the upv_1=2, upv_2=4 , us_sum_1= 15, us_sum_2=12. The UAWT1 = 56((14+12+1)+(15+12+1) +1)) In Fig. 1 the UAWT1 is greater than the ones of the other two RBPs . This shows that Theorem 3 can provide UAWT1 for URBP2, and it can be less than the summation of the two AWT1s (56 < 36 + 27).

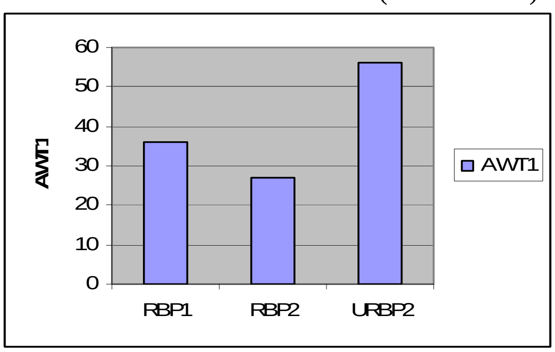

Fig. 1 URBP from two RBPs

Scenario 2: In this scenario the ARBP for two RBPs is examined. Let us consider the same size of sets as Scenario 1. RBP1: $S_{1s,1}$ = 18, $S_{2s,1}$=32,$S_{3s,1}$=120. and RBP2: $S_{1s,2}$ = 12, $S_{2s,2}$=16, and the same $S_{3s,1}$=120. For RBP1 the $s\_sum_{1,1}$= 6, $s\_sum_{2,1}$= 8, $pv_{1,1}$ =3, $pv_{1,2}$=4 the AWT1 = 45. For RBP2 the $s\_sum_{1,1}$= 4, $s\_sum_{2,1}$= 4, $pv_{1,1}$ =3, $pv_{1,2}$=4 the AWT1 = 27. If the ARBP is used (since the two RBPs have the same pvi) then : $s\_sum_{1,1}$= 10(6+4), $s\_sum_{2,1}$= 12(8+4) , $pv_{1,1}$ =3, $pv_{1,2}$=4 the AWT1 = 69. It is evident that the ARBP has greater value of AWT1 and it is not preferable comparing with the URBP2. In Fig. 2 the AWT of ARBP is greater than the ones of the other two RBPs. URBP2 is more preferable than ARBP since from Fig.1 UAWT1= 56 and ARBP from Fig. 2 is 69. This means that with the factorization method the URBP2, can provide better AWTs than the ones based on the addition of s_sub_i of the ARBP.

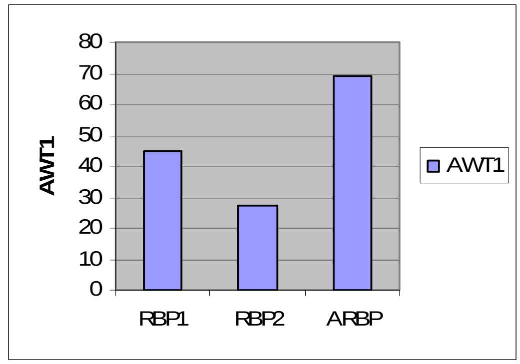

Fig. 2 ARBP from two RBPs

Scenario 3: In this scenario the impact of the ARBP on the DR is examined. Considering the same sets as previous Scenario. The ARBP since the two RBPs have the same pv_i then according to DA and Theorem 5 the delay ratio (DR=4/3) remain the same after the unification of the two RBPs. From Fig. 3 it is evident that the ARBP has the ability to provide the same DR as each of the two compound RBPs do.

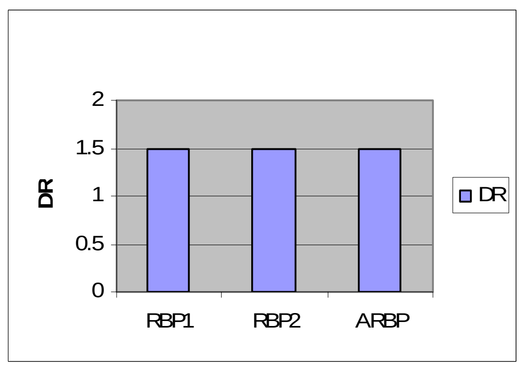

Fig. 3 Delay differentiation with ARBP

Scenario 4: In this scenario the impact of the minimum common factor for URBP1 (Theorem 2) is examined. Considering two RBPs, RBP1: $S_{1s,1}$ = 6, $S_{2s,1}$=24,$S_{3s,1}$=120 and RBP2: $S_{1s,2}$ = 60, $S_{2s,2}$=480, and the same $S_{3s,1}$=120. Using RBPG for RBP1 : $s\_sum_{1,1}$= 3, $s\_sum_{2,1}$= 15, $pv_{1,1}$ =2, $pv_{1,2}$=4.

Using RBPG for RBP2 : $s\_sum_{1,2}$= 15, $s\_sum_{2,2}$= 60, $pv_{1,2}$ =4, $pv_{2,2}$=8. Using 2 as the minimum common factor of pv_{11}, pv_{12} and 4 as the minimum common factor of pv_{21},pv_{22} then the (UAWT1 /2) = (UAWT2 /4). With Theorem 2 it is possible to have minimum values for delay ratio for the unified sets. Fig. 4 shows that by using the URBP1 it is possible to have the same DR for both RBPs and the compound one.

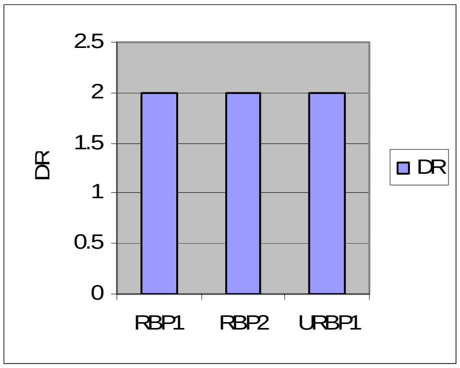

Fig. 4 Delay Differentiation for URBP1

Moreover, if the minimum common factors of two sets are defined as pv_1, pv_2 values then the DR is the ratio of pv_1/pv_2. This can provide an opportunity for achieving a predefined DR.
From all the above some crucial results can be obtained for the Compound Regular Plans and their DR: a. the UAWT1 has less AWT than the sum of the ones of the RBPs. b. The URBP2 can have superiority over ARBP, c. The ARBP can provide the same DR as the RBPs, d. the URBP1 using the minimum common factors can also provide the same DR as the RBPs.
The factorization methods for URBP2 and URBP1 are more promising, for better performance, than the ARBP and for having a predefined DR of two RBPs. The restricting factor for applying the ARBP is that it requires the same values of s_sum_i or of pv_i for the two RBPs. This restriction diminishes the effectiveness of the operational behaviour of the server to create the CO_RBPs and to provide a desired RBP.

8 Conclusion

A framework with a set of algorithms for the compound regular plans is presented. Data-parallelism offers solution to the unification of RBPs in the multiprocessing servers. The URBPs and the ARBPs give different solutions. URBPs have better performance than the ARBP. The DA can provide delay differentiated services for the RBPs , the ARBP and UPRB1. Predefined DR for two services can be achieved based on the minimum common factors of two RBPs. The factorization method can successfully be applied to achieve both goals: better performance and predefine DR. The server can apply any of these solutions for servicing more "hot" data and will be more self-sufficient, self-monitoring and addressing quality of service, among other issues with minimal human intervention. Future work could focus more on decision making for servers on regular data using computational approaches. Broadcast strategies for wireless networks would also be another option for future work.

References:

[1] Acharya, S., Zdonik, F., Alonso, R., "Broadcast Disks: Data Management for asymmetric communications environments", *Proc. of the ACM SIGMOD Int. Conf. on Management of Data,* San Jose, May 1995 , 199-210

[2] Yee, W., Navathe, S., Omiecinski, E., Jemaine, C., " Efficient Data Allocation over Multiple Channels of Broadcast Servers", *IEEE Trans. on Computers*, vol. 51, No.10,Oct 2002.

[3] Ardizzoni, E., Bertossi. A., Pinotti, M., Ramaprasad, S., Rizzi, R., Shashanka, M., " Optimal Skewed Data Allocation on Multiple Channels with Flat per Channel", *IEEE Trans. on Computers* , Vol. 54,No. 5, May 2005

[4] Bertossi. A., Pinotti, M., Ramaprasad, S., Rizzi, R., Shashanka, M., "Optimal multi-channel data allocation with flat broadcast per channel", *Proceedings of IPDS'04,* 2004, 18-27

[5] Yao, G., Zhu, D., Li, H.,Ma. S., " A polynomial algorithm to compute the minimum degree spanning trees of directed acyclic graphs with applications to the broadcast problem", *Discrete Mathematics*, 308(17), 2008, pp.3951-3959.

[6] J. Tsiligaridis, "Consistency in Regular Plans for Mobile Clients" , *Int. Conf. in Wireless Communications, Networking, and Mobile Computing , WiCom2010*, September 23-25, Chengdu, China

[7] R. Grimaldi , *Discrete and Combinatorial Mathematics: An Applied Introduction*, 5e, Addison Wesley,2003.

[8] P.Pacheco, *An Introduction to Parallel Programming*, Morgan Kaufmann, 2011.

[9] J. Huang, S. Cho, C. Huang, "Optimized Scheduling on Broadcast Disks", *Proceedings of second Int. Conf. on Mobile Data Management*

(MDM01), Hong Kong, China, January 8-10, 2001, pp.91-104.

[10] S.Jiang, N. Viadya,"Scheduling Data Broadcast to Impatient Users", *Proceeding of ACM Int. WorkShop Data Engineering Wireless and Mobile Access*, 1999,pp. 52-69.

[11]E. Pitoura, P. Chrysantis, "Multiversion Data Broadcasting", *IEEE Trans. on Computers*, 51(10), pp. 1224-1230.

[12] B.Zheng, L.Lee,"Information dissemination via Wireless Broadcast", *Communications of the ACM*, *(CACM)*, 48(3), 2005, pp. 105-110.

[13] R.Rani, M.Papayee, "An efficient data dissemination method for wireless devices", *Int. Conf. Emerging Trends in Electrical and Computer Technology , (ICETECT)*, 2011, Tamil Nadu, India, pp. 897-900

[14] K. Tabassum, A. Sultana, A. Damodaram," Flexible Data Dissemination Strategy for effective cache consistency in Mobile Wireless Communication Networks, *Int. Journal of Distributed and Parallel Systems (IJDPS)*, Vol.3 ,No.3, May 2012