Algorithms for discovering Regular Plans in Wireless Mobile Environment

JOHN TSILIGARIDIS
Mathematics and Computer Science Department
Heritage University
Toppenish,WA
USA
tsiligaridis_j@heritage.edu

Abstract: - The emerging wireless communication technology gives mobile financial services, location-based services by providing users with the capability of accessing data at any time and place. The broadcast problem including the plan design is considered. The data are inserted and numbered at a predefined order into customized size relations. The server fetches the requests and broadcasts the data to the air. A set of algorithms can guarantee, the creation of a full Regular Broadcast Plan (RBP), for the server with equal spacing repeated instances of items using single or multiple channels. First, the Basic regular (BRA) and the Partition Value algorithm (PVA), can provide a static and dynamic RBP construction with multiple constraints solutions respectively. PVA can also build the grouping of strong relations with minimum number of channels. Second, the Grouping Dimensioning Algorithm (GD provides dynamic solutions and, creates groups with predefined average access time, considering economy of channels. The Cuchoo Search Algorithm (CS) is selected for optimization. CS can discover the RBPs with minimum number of channels. Theorems for discovery RBPs with their criteria are used for preparing the conditions for developing the CS algorithms. Servers with these advantages, in addition to their self-monitoring and self-organizing, will have the ability for channel availability and lower energy consumption by using smaller number of channels. Simulation results are provided.

Key-Words: -Broadcasting, Mobile Computing, Cuchoo Algorithm, Broadcast Plan, Relations, Grouping, Wireless Network.

1 Introduction

Mobile computing is based on the communication between clients and the large scale distributed database. An efficient broadcast schedule program minimizes the client expected delay, which is the average time spent by a client before receiving the requested items. The expected delay of data items is called waiting time. The broadcast channels are also known as “broadcast disks” [1]. There are three basic data broadcasting design methods: the flat, the skewed and the regular. The first two have attracted a great amount of attention [1],[2],[3],[4],[8]. The objective of broadcasting plan is to reduce the expected delay. In this context and based on data popularity, partition methods have been developed [2]. For the flat design the bigger the size of the data set to be transmitted by the server, the higher the expected delay. In large cycles of a flat design users have long wait before getting data that had preciously missed. A higher number of channels is used, for the long broadcast cycle, to reduce the waiting time. For the skewed design, the most frequently requested data are directed to fast channels, and the cold data to slow channels. The regular design is based on the attribute of equal spacing and offers channel availability and energy conservation. It outperforms the flat one by providing shorter average waiting time for both single and multiple channels. In our work it is considered that the users of the popular sets can find their data in the same channel while the users of the last set (most unpopular) have to use other channels as well. In previous work [5], a method used was based on the known size of data sets. The server works with a set of different message (service) queues and has to define the size of data of each set to be processed in order to create an RBP. This problem becomes complicated when there are many categories of data sets and the need for RBPs solution becomes more necessary. The correct selection of the size of data becomes key for an RBP creation. It is considered that there are queues for the various data sets and the scheduler gets packets of them in a cycle (starting from the frequently requested data to the cold data) and transfers them into the main queue. The main purpose of this work is to prepare a framework that will include the structure of the data, the server operations in order to provide an RBP. To this end, some theorems have been developed. Data can be sent by a single channel or a set of channels.Finding the number of channels that can send a group of data, providing also the equal spacing of repeated instances of items, could be a very interesting issue. GDA that belongs to the Compound Relations Algorithms (CRA) finds the minimum number of channels that produce an efficient RBP directly. The surplus of the available channels of both grouping algorithms may be used for other RBP.

The Cuckoo Search Algorithm (CS) based on the Levy flight behavior and brood parasitic behavior is introduced [7]. The CS has been proven to deliverexcellent performance, among others, in function optimization and neural networks training and engineering design.

For the server there is a need to discover Regular Plans for answering the users' mobile queries. The problem stated on the computation of the size of data of different services so that an RBP or RBPs will be feasible. An RBP is considered feasible if it follows the criteria, as will be developed next and with the smaller number of channels. For this purpose a framework with a set of service queues along with the main queue is considered. In addition, solutions for low and high capacity channels are given based on the grouping length. A Cuchoo Search for Regular Plans (CSRP) is developed which can find solutions for servers with a diversity of services and a big size of messages. Solutions with equal and different subrelations are provided.

The paper is organized as follows. In Section 2, and 3 the model description with some analytical results is provided. Sections 4, and 5 the CS algorithm and the CSRP are developed, respectively. Finally, simulation results are provided in Section 6.

2 Model Description

This section contains the definitions, relations and the criteria for the Broadcast Plan.

The likelihood of having a full BP, is studied by using relations iteratively starting from the last set S_3. For this purpose three sets S_i (i=1,2,3) with their sizes S_{is} so that $S_{1s} \leq S_{2s} \leq S_{3s}$ are considered. A set of relations can be created using the S_1,S_2,S_3, considering different number of relations (n_rel) and subrelations in each set (i-subrelation, i=1,2,3). It is assumed that there are three or four subrelations per relation. For sets S_i (i=1,2) items will be sent at least twice , while for the last one S_3 at least one.

A set of *definitions* have been developed in regards to the construction of and RBP:

(1) the size of relation(s_rel) is the number of items that belong to the relation, a subrelation is a part of a relation.

(2) full BP (FBP): is the broadcast plan ,BP, without any empty slot

(3)regular BP (RBP)*:* the FBP with equal spacing property

(4) item multiplicity (it_mu): the number of items repeated in a subrelation

(5) different composition RBP : is an RBP that has the same number of channels when there is a swap between s_sum_i and ps_i. See example 4.

(6) grouping length (gl) it is a divisor $S_{ks}(1,..,k)$

(7) partition value (pv) : is the value with : $pv_i \mid S_{is}$ and $pv_i \mid gl$.

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

A set of *criteria* have also been developed:

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

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

(3)The criterion of PV (cpv or pvi): 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.

More details are in (5).

Example 1: If S_{3s} =80, gl=40, considering that s_sum_3=8 then pv_3 =10 (=80/8). Hence $pv_3 \mid S_{3s}$ and $pv_3 \mid gl$.

These definitions and criteria are used in the Tabu process for discovering RBPs.

A set of *grouping* operations for GDA:

Having ready the RBP then it is possible to combine the relations and create groups of relations depending on the needs of the system (number of channels, and the bounds for AWTi , i=3). For the grouping relations the following definitions are needed:

(1) *Homogenous grouping (hog):* it consists of a set of strong relations and the i_subrelations contain all their items of S_i (i=1,2) without common items. In other words: $\cup$(i_subrelation)=S_i and $\cap$(i_subrelation) ={$\varnothing$}, i=1,2,3. The same number of relations and all of them participate in the grouping process and symmetry load to the channels is provided. It provodes the perfect mapping.

(2) *Semi-homogenous grouping (shog):* is the case where the grouping is the same like the homogenous

but finally we have asymmetry load to the channels. More relations are mapping to some channels and less relations to some others.
(3) *Heterogeneous grouping (heg):* it consists of a set of weak relations where: $\cup$(i_subrelation) $\neq S_i$ and $\cap$(i_subrelation) $\neq\{\emptyset\}$, i=1,2,3. All the relations may participate in the grouping process.
(4) *Integrated relations(int_rel)* is a set of relations so that the union of items in the i_ sub relations (i=1,2) to give the S_1 and S_2. In other words the $\cup$(i_subrelation) = S_i (i=1,2).
(5) *Grouping Factor (gf):* is the ratio of the number of integrated relations divided by the number of channels. It is used only when gt is integer and there are more channels than the set of integrated relations. In that case we perform an homogenous grouping. If gf=1 then we have perfect matching (one channel for each ir) and this does not provide service discrimination, since all the items of the relation have the same multiplicity.
(6) *Homogenous grouping by gf (hgf):* Is when the gf >1 , gf $\in$ I. For example: when there are 10 ir and 5 channels, then the hgf give us the best matching, 2 ir for each channel.

3 BRA

The BRA is based on the conditions to find an RBP and basically works with three sets. It tries to find item multiplicity for the sets, and makes groups of relations using single or multiple channels. It starts with the discovery of an RBP with the smallest number of channels. Grouping methods for the composition of integrated relations with the case of the perfect or approximate matching are also examined. More details are in [5].

4 PVA

The PVA gives answers for cases of minimum number channels. It works with no grouping or RGA. For all the predefined number of integrated relations (d_4) we try to discover the values of $pv_i(i\leq n)$ so that the criterion of homogenous grouping is valid and the multiplicity constraint is satisfied.

PVA : input: S1,S2,S3,S4, Sis (i≤4), n_ch: the # of channels, exist_v: the variable for chg
output: the homogenous grouping for multiple channels

```
find the divisors set D4of  S4 (d4∈D4 )
find the divisors of the S1,S2,S3 (in decreasing order)
 //D3 for S3, D2 for S2, D1 for S1
 //d3∈ D3, d2∈ D2,  d1∈ D1
find the n_ch =   D4/ d4
for each divisor (d4) of set   S4   (a)
   for all Si (i≤4)                              (b)
{  //define the s_sumi= di (i<4)
s_sumi = di (i<4)
pvi =Sis / s_sumi
it_mui= d4 / pvi
      //examine the chg
exist_v=”y”
      while (it_mui∉ I)  // (i<4)
{ j=i;  exist_v =”n”;
          //find new s_sumj ,
           go to (b) }
      if exist_v=”y”
      //examine the multiplicity constraint
it_mu_all=”y”
     if (it_mui+1<it_mui)
{ the  PVA provides solution for Si,Si+1
          homogenous grouping and for n_ch }
     else {//find another d4
it_mu_all=”n”;
            go to (a) }
     if it_mu_all = “y”
        {the PVA provides hg for all the sets with
n_ch}
```

Example 2: Consider the sets: S1s=10,S2s=20, S3s=40, S4s =120. The divisor of S4s are: D4={20,30,40}. For d4 =20 the number of channels, n_ch =120/20 =6.
The divisors of Si(i≤n), D3= {8,5}, D2=(5,4),D1={5,2}.
Taking: d3= 8,d2=5,d1=5 . Considering as n4=d4=20 , s_sum3=8(=d3), S_sum2=5(=d2), s_sum1=5(=d1) then we have:
pv3=40/8=5, and it_mu3= 20/5=4
pv2=20/5=4, and it_mu2= 20/4=5
pv1=10/5=2, and it_mu1= 20/2=10
So the chg and the cmc are valid ($it\_mu_3<it\_mu_2<it\_mu_1$)
If the divisors of S4 are at a decreasing order (e.i.,60,40,20) the n_cl will take the lower value. This is used when the design of RBP is only for a minimum number of channels.
The RBP for all the available channels can be achieved when the divisor of S4 are at an increasing order. This comes from the n_cl formula (D_4/ d_4).

5 GDA

The GDA works with creation of the groups using fewer channels. Economy of channels is very important factor for a large size of broadcast cycle. The grouping is made so that the AWT3 becomes less than a predefined average waiting time for S_3 data.

GDA:*input:*n_rel: # of relations from PVA
n_rel_per_s: is the integrated # relations for S_2
n_ch: #of channels that provide RBP,
pre_av_wt$_3$: is the predefined aver. waiting time for the S_3
n_int_rel: is # of integrated relations from a RBP
variables: AWT$_3$: the aver. waiting time for the concatenated relations for data of S_3
output: min_n_used_ch : the min # of channels that will be used with predefined AWT$_3$

```
for (i= 2: n_ch; i++)
{
n_int_rel = n_rel / n_rel_per_s
if   (n_int_rel = 2p,  p∈I  )     (A)
{ findk_i the integer divisors of  n_int_rel
    // k1>k2>k3>..>kn , K={k1,k2,k3,..,kn }
 for each k ∈ K    // # of channels
{ ma=  n_int_rel / k
grouping  by m  integrated relations and create the
                    k concatenated relations
        if    (AWT3≤ pre_av_wt3)
{  min_n_used_ch = k ;
send  k concatenated relations  to k   channels}
       } } }
if   (n_int_rel = 2p+1,  p∈I  )
{ we work with 2p integrated relations as in (A)
      and the last one (the 2p+1) is added to the last
channel }
 } // end for
```

Example 3: Let us consider $S_1 = 1$, $S_2 =\{2,3,4\}$, $S_3 =\{5,\ldots,76\}$with: $S_{1s} = 1$, $S_{2s} = 3$, $S_{3s} = 72$, pre_av_wt$_3$ = 40. Here $S_{3s} >> S_{2s} >> S_{1s}$ .Using PVA we have k= 24 (72/3), it_mu$_2$= 8(24/3). Hence we have a RBP with it_mu$_2$= 8,and 24 lines. The 24 relations(n_rel) are (1,2,5,6,7),(1,3,8,9,10), (1,4,11,12,13), (1,2,14,15,16),…,(1,4,74,75,76). The 8 integrated relations (with n_rel_per_s =3) can be created from the 24 relations are: ((1,2,5,6,7),(1,3,8,9,10), (1,4,11,12,13)),.. ,((1,2,68,69,70), (1,3,71,72,73),(1,4,74,75,76)). If a single channel is used, AWT$_3$= 72 and 72>25 and grouping is needed with multiple channels in order to have less AWT$_3$.

We have n_int_rel = 8 (24/3). The int. divisors of 8: 4,2.

For k=4, ma =2 (8/4) we have the grouping integrated relations for 2 channels as follows: *channel 1:* ((1,2,5,6,7), (1,3,8,9,10),(1,4,11,12,13), (1,2,14,15,16), (1,3,17,18,19), (1,4,20,21,22), (1,2,23,24,25), (1,3,26,27,28), (1,4,21,30,31), (1,2,32,33,34),(1,3,35,36,37),(1,4,38,39,40)).
channel 2: ((1,2,41,42,43), (1,3,44,45,46), (1,4,47,48,49), (1,2,50,51,52),(1,3,53,54,55), (1,4,56,57,58), (1,2,59,60,61),(1,3,62,63,64), (1,4,65,66,67), (1,2,68,69,70), (1,3,71,72,73), (1,4,74,75,76)).

The AWT$_3$ is: 58. Since 58>40 a new loop for k=4 is needed.

For k=4, m=4 (8/2) we have the four integr. relations
for four channels: *channel 1* : ((1,2,5,6,7),.., (1,4,20,21,22)),
channel 2: ((1,2,23,24,25),.., ,(1,4,38,39,40)),
channel 3: ((1,2,41,42,43),.., (1,4,56,57,58),
channel 4: ((1,2,59,60,61),.., (1,4,74,75,76)).

The AWT$_3$ = 28< 40. Hence, the minimum number of channels is: 4 and this can guarantee the existence of RBP (keep the service discrimination for all the sets).

6 Analytical Results

The previous analysis for RBP was based on a certain size of various sets given also the gl. A basic theorem dealing with the criteria (cmc, and pvi) was developed. These theorems are fundamentals for the characteristics of an RBP. To this end, they are referred in this work.

Theorem1 : Let us consider the case of multiple channel allocation with different multiplicity of sets (such as: S_1, S_2, S_3). Then, if pvi|d4, the validity of multiplicity constraint (it_mu$_{i+1}$<it_mu$_i$ (i=1,..,k-1) can be achieved from the pv criterion ($pv_i<pv_{i+1}$, i<k, k=#sets). Similarly the pv criterion can guarantee the multiplicity constraint criterion.

Proof: Lets prove that if $pv_i< pv_{i+1}$ (1) then it_mu$_i$> it_mu$_{i+1}$. (2). From (1) =>$1/ pv_i> 1/ pv_{i+1}$ => $d_4/ pv_i> d_4/ pv_{i+1}$. If (d_4/ pv_i) ∈ I, =>it_mu$_i$> it_mu$_{i+1}$. Following the reverse order we can get from (2) to (1). So it is not necessary to examine the multiplicity criterion and the pv criterion can provide the multiplicity.

Example 4: Four sets are considered: S1,S2,S3,S4 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_sum1 = 5, s_sum2=5,s_sum3=8 then pv1 = 10/5=2, pv2= 20/5=4, pv3 =40/8=5. We have pv1<pv2<pv3 (pv criterion) and since pv1|20 ,pv2|20,pv3|20 (or d_4 | pv_i) ∈ I) then the chg is valid and an RBP can be constructed. From this process it is evident that it is not necessary to test the cmc.

After this short introduction and for framework construction purposes new analytical results are needed.
Theorem 2: The grouping length (gl or d4) comes from : d4=k* max (PAVi). Where k $\in$ I and plays a role to the capacity and the number of channels.
If k =1 then the S4 (the last or cold set) does not repeated in the broadcasting cycle. (low capacity channel) and larger number of channels are needed to send all the S4 data. Increasing the k then less number of cannel with more capacity is needed.
Example 3: From Example 2, the max (pvi) = 5. If k =1 ,considering that a unit is served for each relation (or s_sum), it will need n_ch =120/5 = 24 channels (low capacity channel). But if gl =20(20 is a divisor of 120) then n_ch =120/20 =6 (fast channels)
Theorem 3: The size of data for a RBP, can be multiple of the size data of S1. It is considered that S4 has unit service.
Proof: the data with sizes: S1s , S2s =k S1s, S3s = m S1s provide the needed structure for RBP directly. Assuming s_sum1 = S1s, s_sum2= S1s and S3s =S1s , the PAV1= 1, PAV2= k, PAV3= m. If k/d4 and m/d4 and k<m , the criteria will be valid and an RBP exists.
Theorem 3 is the necessary condition in order to have RBP.
Theorem 4: (for equal size of subrelations) The size of data can have some equal factors after their factorization. This property can provide an RBP if the non common factors are multiple. (for the same size of subrelations).
Proof: let's consider the size of the data: S1s(=non1*com1), S2s(non2*com2),S3s(non3*com3), S4s. If com1=com2=com3 then the s_sum1= s_sum2 = s_sum3 = com1. Moreover if non2| non1, non3| non2 then an RBP is visible since the two criteria are valid.
Example 5: Let's S1s=10, S2s =30, S3s= 60 , S4s=120. The factorization gives: S1s= 2*5, S2s=3*2*5, S3s= 3*2*2*5. The common factor is 2*5 (=s_sumi)and the non common factors are: 1, 3, 6. These are divisible numbers. Hence an RBP exists with PAV1=1, PAV2=3,PAV3=6. For the d4 if k=1 then d4= 6 (low capacity channel).
Theorem 5: (for non equalsubrelations) After the factorization an RBP is feasible when the PAV criterion is valid and a choice for gf (or d4) can be made.
Example 6: When subrelations of different size are used. Let's : S1s=5, S2s=6, S3s=6 , and PAV1=2, PAV2=5, PAV3=10. For d4 if k=1 then d4=10.(low capacity). In addition, the size of the data sets is not necessary to be multiple of each other and they can use different subrelations.
Example 7: Let's S1s=20, S2s =35, S3s= 48 , S4s=120. The factorization gives: S1s= 4*5, S2s=5*7, S3s= 6*8 and PAV1=4, PAV2=5, PAV3=6. The subrelations are: s_sub1= 5, s_sub2=7, s_sub3= 8. If k=1 then d4=6 and n_ch = 120/6 =20.(low capacity channels). If k=20 then n_ch =120/20 =6 (fast channels).
Theorem 5 provides a new criterion for searching with CS. This means that for the case of different subrelations after the factorization it needs: (a) the non common factors has to follow the pavi criterion (pavi<pavj, for i<j), (b) the value of gf can be discovered so that gf | S4s , and pavi | gf or max(pavi) = gf (c) the value of k depends on the available channels. This new criterion is called as *compound criterion* and it is used from CS.

7 CS

The CS can be directly applied to virtually any kind of optimization problem. We can state most of these problems in the following form, where "optimize" means to minimize or maximize:: Here is how to optimize (minimize the number of channels) for the construction of the RBPs:

optimize $f(x) = (f_1(x), f_2(x), .., f_n(x))$

subject to $x \in X$

where X represents the feasible space, that is the set of all valid solutions, the solutions that fulfill every constraint of the problem.

The CS is a higher-level metaheuristic procedure for solving optimization problems, designed to guide the search operator in avoiding the traps of local optimality. Cuckoo birds attract attention because of their unique aggressive reproduction strategy. Cuckoos engage brood parasitism. It is a type of parasitism in which a bird (brood parasite) lays and abandons its eggs in the nest of another species [6]. Cuckoo Search algorithm is population based stochastic global search metaheuristics. It is based on the general random walk system which will be briefly described in this chapter. In Cuckoo Search algorithm, potential solutions corresponds to Cuckoo eggs. One approach is to simplify novel Cuckoo Search algorithm through three below presented approximation rules: (1) Cuckoos chose random location (nest) for laying their eggs. Artificial cuckoo can lay only one egg at the time. (2) Elitist selection process is applied, so only the eggs with highest quality are passed to the next generation (3) Host nests number is not adjustable. Host bird discovers cuckoo egg with probability *pd* ∈ [0,1]. If cuckoo egg is disclosed by the host, it may be thrown away, or the host may abandon its own nest and commit it to the cuckoo intruder.

A simple representation where one egg in a nest represents a solution and a cuckoo egg represents a new solution is used here. The aim is to use the new and potentially better solutions (cuckoos) to replace worse solutions that are in the nests. The random walk is called Lévy flight and it describes foraging patterns in natural systems,

Only two parameters are needed by the algorithm, the discovery rate pd ∈[0,1] and the size of population n. When n is fixed, pd controls the elitism and the balance of randomization and local search. Most notably, an increase in the local search is needed and this is referred in the CSRP

8 CS for regular plan, PVA, GDA

The theorems apply to CSRP for discovering RBPs. It starts with randomly generated initial population. Each individual is evaluated by the fitness function. The fitness function finds the divisors of the S_{is} (except for the last set size) and test the applicability of the pvi and cmc criterion. Individuals are eliminated when they do not follow the fitness function. The next generations are created with the application of the GA operators. The step size for the Levy flight is set to the $upperbound_i$ /100 . Where of $upperbound_i$ is the size of each of the set queues. The size of population n ,is the sum of number of available messages in the service queues.

CSRP : input : initial population of n host nests (the size of data in the service queues).
output : discover an RBP
Generate initial population of n host nests xi
while {(*t*<*MaxGenerations*) and (! termin.condit.)
  get a cuckoo randomly via Lévy flights
    //in order to find the next sizes of the sets
  evaluate its fitness *Fi* //if the numbers follows
      the compound criterion
  randomly choose nest among *n* available
      nests (for example *j*)
if(*Fi* > *Fj*) {replace *j* by the new
    solution;}
  fraction pd of worse nests are abandoned
    and new nests are being built;
  //search around pd (n*pd) for best nests (A)
  // and replace if a better solution exists(A)
  for the rest of n keep the best solutions or
    nests with quality solutions;
  rank the solutions and find the current best
  }//end while
  //Post process and visualize results

In the CSRP pseudocode, two lines (A) are added to provide local search for best nests in the area around the pd.
For the PVA the CSPVA is created in analogous way as the CSRP. It takes the RBP from the CSRP and then it works according to PVA. The multiplicity constraint is the condition that is examined when discovering the pvi values. Finally, for the GDA the CSGDA has a constraint for the AWT3 and the minimum number of channels.

9 Simulation

Our simulation is concentrated on the effectiveness of CSRP,CSPVA,CSGDA and BRA for the discovery of RBPs. The items are separated into at most five categories according to their popularity using Zipf distribution. The scenarios are the following:
Scenario 1: CSRP for various sets. Let's consider 3 groups of sets with their corresponding queue sizes (SQi). Group 1: SQ3= 150, SQ2=85, SQ1 = 40, Group 2: SQ4=350, SQ3=80, SQ2=62, SQ1= 25, and Group 3: SQ5=330, SQ4=280, SQ3=260, SQ2=140, SQ1= 70. Group 4: SQ5=350, SQ4=300, SQ3=250, SQ2=130, SQ1= 80. The CSRP finds the RBPs in shorter time for the Group 1 then Group 2 and finally for Group 3. This happens because there is an increasing amount of services for the various groups. Group 3 and Group 4 have approximately the same values (Fig. 1).

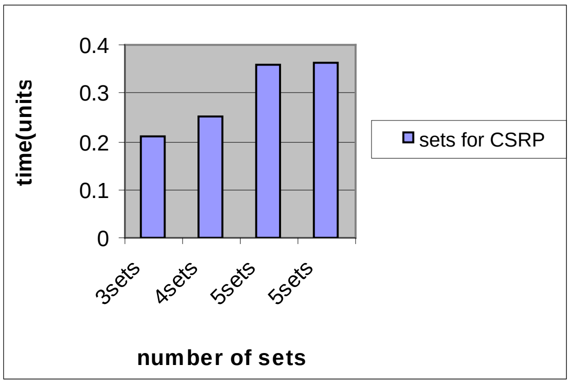

Fig. 1 CSRP for various RBPs

Scenario 2: CSRP vs BRA. Let us consider a group of services with 5 sets and the CSRP running against BRA. The CSRP outperforms BRA. BRA has random non local search. CSRP discovers the RBP earlier since it works also with the Levy flights in the next generations and the additional ability for local search (Fig. 2).

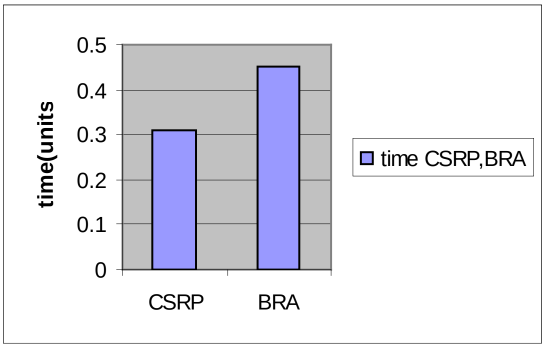

Fig. 2 CSRP vs BRA

Scenario 3: Considering S_{4s} = 1200, S_{3s}=600,S_{2s}=400,S_{1s}= 100. The AWT for the S1,S2,S3 remain the same (Fig. 3) because PAV could find the same values of pv_i for all the number of channels(6,3,2). The AWT_4 has increasing trend, and depends on the number of channels the PAV discovers. The lower the number of channels is the greater the AWT_4. It is considered that for each relation there is one element (no repetitions).

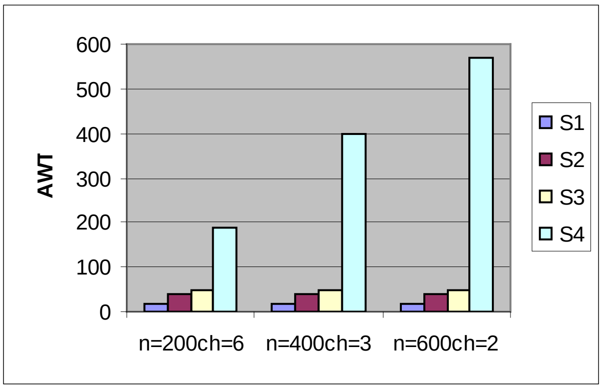

Fig. 3 The AWT for the PAV

Scenario 4: Considering the same as the previous scenario with CSPAV running against PAV. SCPAV discover the RBP earlier due to the additional ability for search (Fig. 4)

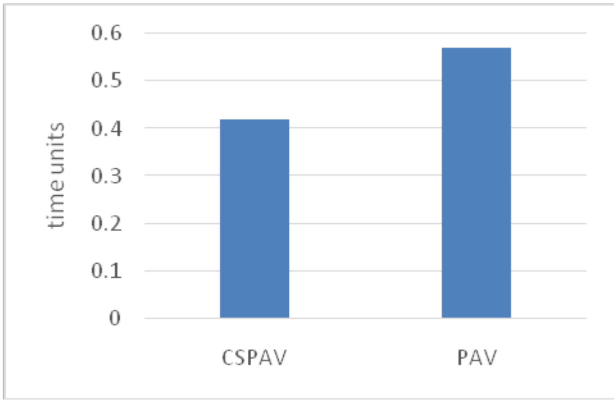

Fig. 4 CSPAV vs PAV

Scenario 5: Three set of data are used and three cases (each one for each set) are developed starting from left to right in Fig. 5. All of them have the same S_1 data. The second set has more data (relations) of S_3 and the same size of S_2 data (relations). Because of this, in the second case four channels are used instead of three in order to provide the same AWT_3. The number of channels are selected according to GDA considering $pre\_av\_wt_3$ = 40sec. The third set has more data on S3 and less data on S2 comparing with the data of the second set. Because of this there is an increase of AWT_3 (18 sec comparing with 16sec) and a decrease of AWT_2 (from 8sec to 6sec).

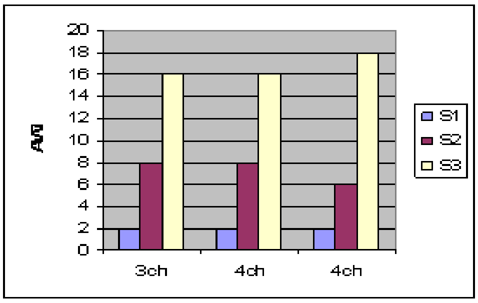

Fig. 5 AWT with GDA grouping

Scenario 6: Considering the same as in the previous scenario; the CSGDA supersedes GDA because of the nature of the CSGDA (Fig. 6)

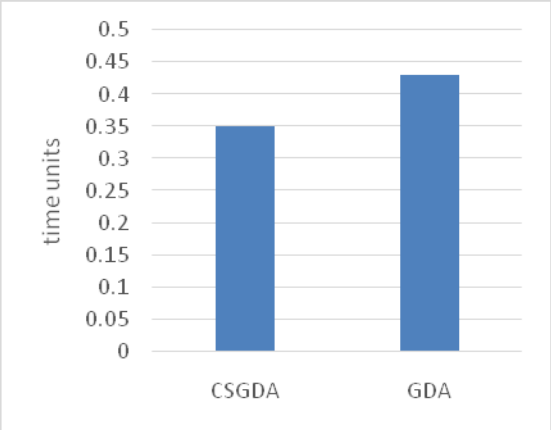

Fig. 6 CSGDA vs GDA

10 Conclusion

A set of algorithms that can discover RBP for mobile users using the CS algorithm has been developed. Theorems can discover the sizes of the data sets considering equal and non equal subrelations. The CSRP can discover the sizes of the datasets using the compound criterion and provide an integrated solution to the regular broadcast schedule program minimizing also the client expected delay. It is very effective and flexible for solving hard constraint satisfaction problems and more generally constrained combinatorial search problems. It can provide a new dimension in the design of RBP. The next generation servers, by applying the CSRP, CSPVA, CSGDA will enhance their ability to self-sufficiency, self-monitoring, and they may address quality of service with minimal human intervention. Future work could include the use of other optimization methods for the RP problem

References:

[1] S. Acharia, M.Franklin, S. Zdonik, R.Alonso, "Broadcast disks: Data management for asymmetric communications environments", *in ACM SIGMOD- 1995,* ACM SIGMOD Conference Proceedings, pp. 199-210.

[2] W. Wee, S. Navathe, E. Omiecinski, C. Jermaine, "Efficient Data Allocation over Multiple Channels of Broadcast Servers", *in IEEE Trans. On Computers*, vol. 51, No.10,Oct 2002, pp. 1231-1236.

[3] E. Ardizzoni, A. Bertossi, M. Pinotti, S. Ramaprasad, R. Rizzi, M. Shashanka, " Optimal Skewed Data Allocation on Multiple Channels with Flat per Channel", in *IEEE Trans. on Computers*, Vol. 54, No. 5, May 2005, pp. 558-572.

[4] A. Bertossi, M.Pinotti, S.Ramaprasad, R.Rizzi, M. Shashanka, "Optimal multi-channel data allocation with flat broadcast per channel", in *International Parallel and Distributed Processing Symposium (IPDS)-2014 ,* IPDS Proceedings, pp.18-27

[5] J. Tsiligaridis, "Static and Dynamic Algorithm for Regular Data Broadcasting Plans in Wireless Mobile Environment", *International Journal of Advanced Computer Science, IJACS,* Vol.2 No.2, Feb. 2012, pp. 42-48.

[6] Yang, X. S., and Deb, S." Engineering Optimization by Cuckoo Search", in *Int. Journal of Mathematical Modelling and Numerical Optimization*, Vol.1, No 4, 2010, pp. 330-343

[7] E. Valian, S. Mohanna, S. Takavoli, "Improved Cuckoo Search Algorithm for Global Optimization", in *Int. Journal of Communications and Information Tech.*, IJCIT, Vol.1, No.1, 2011,pp. 31-62.

[8] N. Vaidya, S. Hameed, "Scheduling Data Broadcast in Asymmetric Communication Environment", in *ACM/Baltzer Wireless Networks*, vol.5, 1999, pp.171-182.

[9] Ljubomir Lazic, Nikos Mastorakis "Cost effective software test metrics " in *WSEAS Transactions on Computers*, Volume 7, Issue 6, pp 599-619

[10] Ljubomir Lazic, Nikos Mastorakis, "Orthogonal Array application for optimal combination of software defect detection techniques choices" *in WSEAS Transactions on Computers*, Volume 7, Issue 8, pp 1319-1336