IOSR Journal of Mathematics (IOSR-JM)
e-ISSN: 2278-5728.Volume 6, Issue 1 (Mar. - Apr. 2013), PP 38-52
www.iosrjournals.org

A Hypocoloring Model for Batch Scheduling Problem

Sangeetha Raghu
(Department of Mathematics, TheivanaiAmmal College for Women, India)

Abstract: *In this paper we define Scheduling, Batch Scheduling, Coloring, Subcoloring, Hypocoloring, Chromatic number, Subchromatic number, and Hypochromatic number for a given graph. A batch Scheduling problem has been obtained by using the above discussed concepts.An exponential algorithm has been developed for triangle free graphs. The solution is obtained by introducing COCA - 'Contract or Connect" Principle. The Subchromatic and Hypochromatic number for shell graphs has been illustrated. A Hypocoloring model for personnel assignment problem has been briefly discussed.*

Keywords: *Exponential algorithm, Hypochromatic number, Shell graphs, Subchromatic number*

I. Introduction

Vertex Coloring models to a number of scheduling problems. In the cleanest form, a given set of jobs need to be assigned to time slots, each job requires one such slot. Jobs can be scheduled in any order, but pairs of jobs may be in conflict in the sense that they may not be assigned to same time slot, for example because they both rely on a shared resource.

II. Definitions

1.1 Batch Scheduling

The scheduling of jobs or operations in batches is called *Batch Scheduling.*

1.2 K – vertex coloring

A *k – Vertex Coloring* of a graph G is an assignment of k- colors 1, 2,…,k to the vertices of G. The coloring is proper if no two distinctadjacent vertices have the same color. Thus a proper k- vertex coloring or k-coloring of a loop less graph G is a partition ($v_{1,}$ v_2,...,v_k) of V into k-independent sets.
G is k- colorable if and only if G has proper k- Coloring.

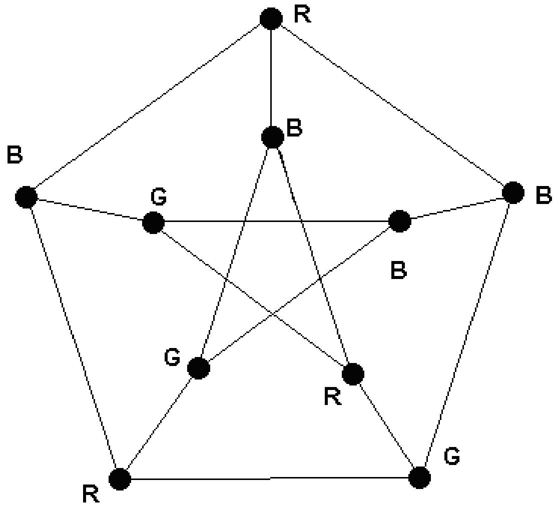


Figure 1.2.1 Coloring of Peterson Graph

1.3 Chromatic number

The *Chromatic number* χ (G) of a Graph G is the minimum k for which G is k – Colorable. If χ (G) =k, then G is said to be k – Chromatic.

In Figure 1.1, note that the chromatic number of the Peterson graph is 3.

1.4 Sub coloring

A *Subcoloring* is a vertex coloring of a graph G in which every color class induces a disjoint union of cliques.

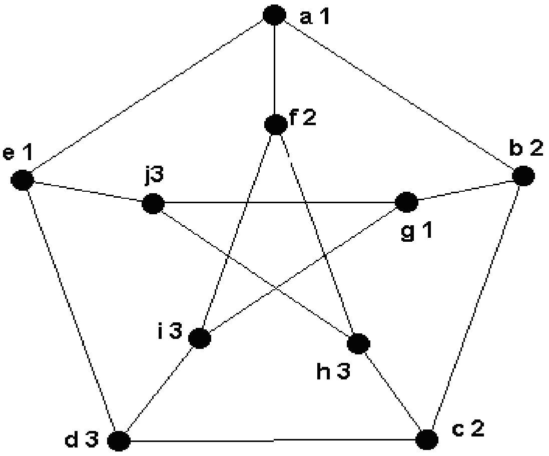


Figure 2.4.1ASubcoloring of Peterson Graph

For example, the Peterson graph can be subcolored with three colors as in figure 2.4.1

Let S_1= {(a, e), (g)} S_2 = {(b, c), (f)} S_3 = {(d, i), (h, j)}

Here $S_{1,}$ S_2, S_3 are the three Subcolor class each of which induces a union of disjoint cliques.

1.5 Subchromatic Number

The *Sub Chromatic Number* χ_s(G) of the graph G is the minimum k for which G is k – sub colorable. Note that, if χ_s(G) = k, then G is said to be k – chromatic. χ_s(G) = 3.

From Figure 1.1, the chromatic number of Peterson graph is 3 and from figure 1.2, the subchromatic number is also 3.

1.6 Hypocoloring

Given a graph G, and integral weights on vertices we want to partition the vertices of G into Subcolor class such that sum of the weights of the heaviest cliques in each Subcolor class is minimized. This partition of the vertices is known as the *Hypocoloring*of the graph G.

1.7 Hypochromatic number

The *Hypochromaticnumber* χ_h (G) of the graph G is the smallest k for which G has a *k–Hypocoloring.*

Let us color the weighted Peterson graph as in figure 2.7.1

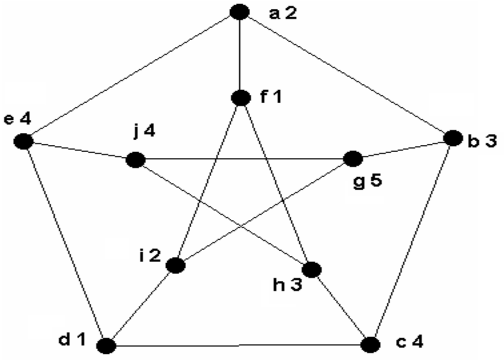


Figure 2.7.1 A Hypocoloring of weighted Peterson Graph

Then the Subcolor classes are given by,

S_1 = {(a, e), (g)} S_2 = {(b, c), (f)} S_3= {(d, i), (h, j)}

The weights of each Subcolor class is given by,

S_1 = {(a, e), (g)} = max {6, 5} = 6

S_2 = {(b, c), (f)} = max {7,1} = 7

S_3= {(d, i), (h, j)} = max {3, 7} = 7

20

The total weight of this Hypocoloring of the Peterson graph is 20.

Now let us color the Peterson graph with another Subcolor classes as in figure 1.4

Let $S_1 = \{(d, i), (c)\}$ $S_2 = \{(b, c), (g)\}$ $S_3 = \{(a, b), (e, j)\}$

The weights of each Subcolor class is given by,

$S_1 = \{(d, i), (c)\}$ = max {3, 4} = 4
$S_2 = \{(d, i), (g)\}$ = max {4, 5} = 5
$S_3 = \{(a, b), (e, j)\}$ = max {5, 8} = 8
17

By proceeding like this we can obtain the minimized weight Subcoloring of the weighted Peterson graph. Note that the minimum weight of the weighted Peterson sub graph as in figure 2.7.1 is 17. Hence the Hypochromatic number of the weighted Peterson graph is 3.

III. The Chromatic scheduling model

3.1 Introduction

Chromatic scheduling is the domain of scheduling problems which can be formulated in terms of graph coloring or more precisely of generalized graph coloring.

The applications of Chromatic scheduling

- Course Timetabling
- Satellite Communication
- Processor scheduling problems

Here the concept of weighted coloring has been introduced to generalize classical coloring models and to handle situations where operations occur with possible different processing times.

We shall generalize a weighted coloring model for studying same types of batch scheduling problems.

In this chapter we give a brief introduction to Chromatic scheduling and explain the Chromatic scheduling model for a graph.

3.2 A chromatic scheduling model

In order to describe our generalized weighted coloring model, we shall consider an instance of batch scheduling problem which can be stated as follows:

- Given a finite set V of operations v to be processed on some identical processors.
- Each operation v has a positive processing time w (v) which does not depend on the processor.
- No preemptions (superiority) will be allowed during the processing of an operation.
- Each processor will handle one operation at a time
- There are some incompatibilities between pair of operations (Incompatibility = not suitable, do not agree with each other)
- If u and v is incompatible then operations u and v cannot be processed simultaneously (or) different processors.
- We take each operations v as a vertices (nodes) of a graph G = (V, E) and the set E= the set of incompatible pair of operations.
- Each vertex v will have a weight w(v)
- S = Collection of pair wise compatible operations.
- The operations in S are assigned to different processors and they are processed simultaneously.
- All operations in S are completed when the operation v with the largest processing time w(v) is completed
- Weight $w(S) = \max \{w (v) / v \in S\}$.
- Assigning each operation to some batch corresponds them to partitioning the node set V(G) of the graph G into a number of
k stable sets.
- The problem is of finding of k-coloring $f = (S_1, S_{2\ldots\ldots\ldots}S_{k)}$ of G such that cost of f = C (f) = $w(S_1) + w(S_2) + \ldots\ldots\ldots + w(S_K)$ is minimum.
- All operations in a batch are assigned to different processors and processed simultaneously.
- In our batch scheduling model, we define the weight $w(k_n)$ of a clique k_n as $W(k_n) = \sum_{v \in K_n} w(v)$

Our batch scheduling problem now consists of in finding a k-Hypocoloring. $(S_{1\ldots}S_{k)}$ of the nodes of G, (i.e.) a particular of the node set into hypostable sets such that:

$$Ḱ(\delta) = \sum_{i=1}^{k} w(S_i) \text{ is minimum.}$$

Observe that k is generally not given; the value results from the minimization of Ḱ.

The above model may also be used for representing some machine scheduling problems we are for instance given a collection of jobs v with processing times w (v) in a flexible manufacturing system; we link the nodes representing two jobs if these share a certain number of tools; it will be interesting to assign these jobs to a same machine on which the appropriate tools will be installed. A batch will consist of an assignment of jobs to some machines in which we try to assign to a same machine jobs which share some tools. Since there is only a limited number of a tool of each type, we will try to assign to different machines jobs which do not need the same tools. Hence a batch will be represented by a hypostable set in the graph of compatibilities and the processing time of a batch will again due the maximum load of machine (maximum of the sums of processing times of jobs assigned to the same machine).

We shall concentrate on this model of weighted Hypocoloring which is motivated in a natural way by the batch scheduling context.

3.3 Conclusion

In this chapter, we give a model for chromatic scheduling and we obtained chromatic scheduling number for a cactus and cactus with a pendent edge

IV. Properties of optimal Hypocoloring

4.1 Introduction

In this chapter, we observe some of the properties of optimal Hypocoloring for any graph. That is, we obtain the minimum number of colors used in a optimal weighted coloring of a simple graph G does not exceed Δ (G) +1.

Lemma 4.1

In a weighted graph I = (G, w) such that $\forall$ vεV, w (v)>0, any k- hypo coloring δ with minimum cost $k^{\wedge}$ (δ) satisfies k$\leq \Delta$ (G) +1.

Proof

We shall show that any ℓ - hypo coloring $\delta' = (S_1', \ldots, S_\ell')$ with $\ell > \Delta$ (G) +1 can be transformed into a k-hypocoloring δ with K$\leq \Delta$ (G) +1 and $k^{\wedge}(\delta) < k^{\wedge}(\delta')$. This is done in polynomial time.

We assume, $w(S_1') \geq w(S_2') \geq \ldots\ldots\ldots\ldots.. \geq w(S_\ell')$

Assume $S_\ell' \neq \Phi$ so there is a vertex (node) $x \in S_\ell'$ such that it has almost Δ (G) neighbors. (since Δ (G) is the maximum degree of G)

Since l>Δ (G) +1, there is at least one color say s, which satisfies s $\leq \Delta$ (G) +1 and which does not occur in the neighborhood N(x). So we can recolor 'x' with and setting Ss' = Ss' υ {x}, we have $w(S_s^*) = w(S_s')$ [(since w(x) $\leq w(s_l') \leq w(s_s')$)]

Setting Si* = Si' (i $\neq$ s, i $\neq$l), we get a hypo coloring δ^* with $|S_l^*| < |S_l'|$ and $k^{\wedge}(\delta^*) \leq k^{\wedge}(\delta')$

[Since S_l' is the last one such that $w(S_1') \geq w(S_2') \geq \ldots\ldots\ldots\ldots.. \geq w(S_\ell')$]

We repeat this process with for nodes in S_ℓ^* have been recolored with a smaller color, then we continue until there are no more nodes with colors s > Δ (G) +1.

Thus at the end, the cost of the resulting hypocoloring verifies $k^{\wedge}(\delta^*) \leq k^{\wedge}(\delta')$,since we have assumed $\forall$ vεV, w (v)>0.

Hence the proof.

Theorem 4.1

If I= (G, W) is a weighted graph with maximum degree Δ (G) then there exists a K-hypo coloring = ($S_1, S_2\ldots, S_k$) with minimum cost satisfying the following

(i) K $\leq$ Δ (G)

(ii) $\forall$i$\leq$ k, $\forall$v ε S_i, $d_{G_{i,v}}(v) \geq$ i-1 where $G_{i,v}$ is the sub graph of H induced by $S_1 \upsilon S_2 \upsilon \ldots S_{i-1} \upsilon \{v\}$

(iii) S_i Contains no $K_{\Delta(G)+3-i}$

Proof:

(i) Let us consider a k-hypocoloring δ .From the Lemma; we have k $\leq\Delta$ (G) + 1

If k $\leq \Delta$ (G), then it is proved.

Let us assume k= Δ (G) + 1.
We have a k – hypocoloring with a minimum number of nodes with colork.
$|S_k|$ is minimum.
Let v be a node in Sk. If there is some color S $\leq \Delta$ (G) missing in the neighborhood then we can recolor v with s and we obtain a k-hypo coloring δ' with $k^{\wedge}$ (δ')$\leq$ $k^{\wedge}$ (δ)
k-hypocoloring δ' with $k^{\wedge}$ (δ')$\leq$ $k^{\wedge}$ (δ)
This is a contradiction (since $|S_k|$ is minimum)
So we assume all the colors 1, 2,…, Δ (G) occurs in N (v).
Let u be a neighbor of v in $S_{\Delta (G).}$
u$\in$N (v) $\cap$ $S_{\Delta (G).}$
'u' has some color c $\in$ Δ (G) missing in N(u) .
Since v $\in$ $S_{\Delta (G) +1}$ (S_k, k = Δ (G) +1) is in N(u).(v is in neighborhood of u)

If c< Δ (G), then we can recolor u with 'c' and then we can be recolored with Δ (G), we still have a k-Hypocoloring and the cost has not increased.

But $|S_k'| < |s_k'|$ a contradiction.
So we must have color c = Δ (G) missing in N (u).
We recolor v with color c = Δ (G) and we get a new k-Hypocoloring.

Repeating this process until there are nodes (vertex) with color k= Δ (G) +1, we will finally have a Δ(G) hypo coloring.

δ' = (S_1', S_2'…., $S_{\Delta(G)}$') with $k^{\wedge}$ (δ')$\leq$ $k^{\wedge}$(δ) because w($S_{\Delta (G)}$')$\leq$w($S_{\Delta (G)}$)+w($S_{\Delta (G)+1}$)
This is a contradiction again.
Thus we have examined for all cases and we get a contradiction.
k $\neq \Delta$ (G) +1
k < Δ (G) +1 $\Rightarrow$ k$\leq\Delta$ (G)
This is the best possible bound.

(ii) The proof is exactly the same as previous. If suppose $d_{Gi,v}(v)$ <i-1 for some v $\in S_i$, then we can recolor v with some color missing in {1,2,…,i-1} without increasing the cost of the Hypocoloring.

We repeat thus as long as possible. Again we get a contradiction as
before.

Our assumption $d_{Gi,v}(v)$ < i-1 is wrong .
$d_{Gi,v}(v) \geq$ i-1 .

Hence the proof

(iii) If S_i contains no $K_{\Delta (G)+3-i}$
Assume that S_i contains a $K_{\Delta (G)+3-i}$ and let v be a node of this clique.
Since $d_G(v)\leq \Delta$ (G), we deduce that
$d_{Gi,v}(v) \leq d_G(v)$ –(Δ (G) +3 –i-1)
$\leq d_G(v)$ – (Δ (G) +2-i)
$d_{Gi,v}(v) \leq \Delta$(G) - Δ(G)-2 –i
$\leq$ i - 2
This is a contradiction.
Hence S_i does not contain $K_{\Delta (G) +3-i}$ Hence Proved.

Remark 4.1

The bound in theorem 3.1 is the best possible. For every integer p, there exists a tree G with Δ (G)-p and weights w (v) for the nodes of G such that all optimal k-Hypocoloring have k=p colors.

G is constructed as follows:
Start from tree T_2 for p = 2

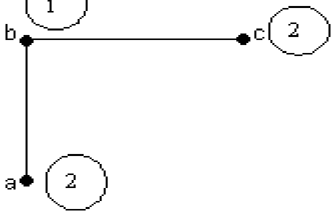


Figure 4.1 A Tree (T_2)

It consists of a chain a, b, c where nodes have labels ℓ (a) = ℓ (c)=2, ℓ (b) =1 and w (a) = w(c) =10 and w (b) = 10

Generally, having tree T_{i-1} (where nodes have labels1, 2 … i-1 and weights almost (10^{i-2}), we construct T_i by introducing at each node of T_{i-1} a chain of two additional nodes v', v'' and weights (½) (10^{i-1})

Now in T_i we take one of these chains v', v'' such that the node adjacent in T_{i-1} has a cost 10^{i-2} and we condense the edge [v', v''] into the node v_i.

[The weight of the new node is w (v_i) = w (v') + w (v'')= ½ 10 $^{i-1}$ + ½ 10 $^{i-1}$ =10 $^{i-1}$]

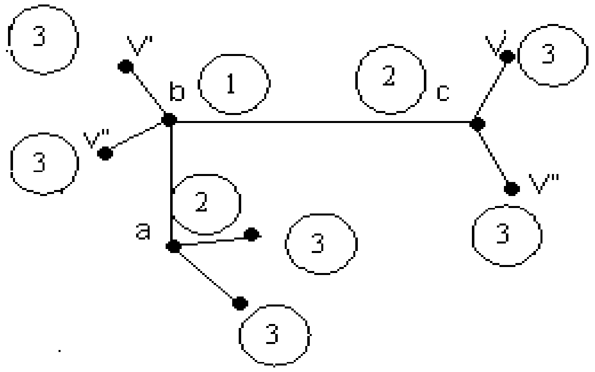


Figure 4.2 Tree G from T_2

One can verify that the graph T_i is a tree with maximum degree i, one also observes that the labels define p-Hypocoloring δ= (S_1, S_2,…, S_p) where S_i= {v/ ι(v) = p+1-i}. We can verify that I is the unique optimum Hypocoloring.

Theorem 4.2

Let I = (G, W) we a weighted Graph such that w (v) >0 for each v∈V then any Hypocoloring δ*= (S_1, S_2,...,S_k) with minimum cost $\acute{K}$ (δ) satisfies k = δ* □ 1+ |w| (□ (G)-1) | w| = the number of distinct values of the weights.

Proof:

Let I = (G, W) be a weighted Graph such that | w| ≤1 and

δ*= (S_1, S_2…, S_k) be an optimal hypocoloring of G with

w (S_1) ≤ w S_2) ≤ … ≤ w (S ℓ)

We show that ℓ = |δ*| ≤ 1+|w| (Δ(G) – 1)

We use induction hypothesis on | w |.

Let t = max {i: w (S_i) ≤$\max_{v□V}$ w (v)}

We have t ≤ Δ(G)

Otherwise, an optional coloring gives a better solution. Moreover if t= ℓ =|δ*|, then we have, $\acute{K}$ (δ*)≤ ℓ w_{max}

$$\acute{K}(\delta^*) = \sum_{i=1}^{\ell} w(S_i)$$

$$= w(S_1) + w(S_2) + \ldots + w(S_\ell)$$

$$\leq \ell\, w(s_l) \text{ [since } w(S_1) \leq w(S_2) \leq \ldots \leq w(S_\ell)]$$

$$\acute{K}(\delta^*) \leq \ell\, w_{max}$$

Also, an optimal coloring is a feasible solution with a cost almost w $_{max}$Δ(G)

$\acute{K}$ (δ*)≤ w_{max} Δ(G)

From, above equations, we have,

ℓ w_{max}≤ $\acute{K}$ (δ*) ≤w_{max} Δ(G) and since |w| ≥1,

We also have, Δ(G)≤1+|w|(Δ(G) -1)

Now assume t < |δ*|, we deduce t ≤χ (G)-1), since otherwise an optional coloring of G gives a better solution.

Observe that δ' = (S_{t+1} ,…., S_k) is an optional Hypocoloring on the sub-instance I'= (G',w') where G' is the sub graph of G induced by

$V' = S_{t+1} \cup \ldots \cup S_{\ell}$ and w' is the restriction of w to G'.
By construction I' $\neq 0$ (since $t < |\delta^*|$)
Moreover $|w'| \leq |w| - 1$
Thus, using an inductive hypothesis, we have,

$\ell - t = |\delta'| \leq 1 + |w'|\ (\chi(G') - 1)$
$\leq 1 + (|w| - 1)\ (\chi(G') - 1)$

$\ell - t \leq 1 + [|w|(\chi(G) - 1 - (\chi(G) - 1)]$

$\ell - t \leq (\chi G) - 1) - t \quad (\because t \leq \chi(G) - 1)$

$\ell \leq 1 + |w|\ (\chi(G) - 1)$
$\therefore$ Any hypo coloring $\delta^* = (S_1 \ldots\ldots S_k)$with minimum cost Ḱ (δ) satisfies
$k = |\delta^*| \leq 1 + |w|\ (\chi(G) - 1)$
Hence proved.

4.2 Conclusion

In this chapter, we have obtained some of the properties of optimal hypocoloring .we have obtained the best possible bound $k \leq \Delta$

V. An exponential algorithm for triangle free graphs

5.1Introduction

In this chapter, we define an exponential algorithm for triangular free graphs and hence we apply the algorithm for a triangular free graph.

We shall consider graphs containing no induced triangles; these are precisely the Graphs G for which the largest size w (G) of a clique is two.

In such graphs, hypo stable sets consist of nodes and of edges (Cliques of size 2). Such Graphs can have arbitrarily large chromatic numbers, it follows that they can also have arbitrarily large hypochromatic number, and indeed a triangle-free graph G with $\Delta(G) \leq 2k$ has $\chi_h(G) \leq k$

If we had χ_h (G) <k, then we could take a minimum Hypocoloring $\delta = (S_1 \ldots S_r)$ of G with r<k. Each S_i could be decomposed into two stable sets S_i', S_i'' (Since S_i could be decomposed of nodes of edges) and we would get a 2r coloring ($\delta = (S_1', \ldots, S_r', S_1'', \ldots, S_r'')$) of G with 2r<2k, which is a contradiction.

We shall now show that, based on the separation principle (link two nodes or merge them) described for usual colorings of graphs, we can develop a "light version" procedure for determining a Hypocoloring δ with minimum cost Ḱ (δ) in a weighted triangle free graph. This procedure will in addition exhibit in a striking way the symmetry between usual colorings andHypocoloring. Forusual colorings, one separates the possible colorings of a graph G into two classes by repeatedly choosing a pair of non adjacent nodes x and y. There is a one to onecorrespondence between the colorings of G where
x and y have the same color and the colorings of the graph G, obtained from
G by merging nodes x, y into a single nodes x'.

In the same way there is a one-one correspondence between the colorings of G where x and y have different colors and the colorings of the graph G_2 obtained from G by introducing an edge <x, y>

So, G can be replaced by G_1 and G_2. We repeat this operation for each one of G_1 and G_2 as long as possible (i.e.) until nodes x, y in xy we set w (xy) = max {w(x), w(y)}. The above algorithm can be applied as before. The clique with minimum weight will give the optimal cost of a weighted coloring.

5.2 Contract or connect principle (COCA)

Our purpose is to give an algorithm for finding a Hypocoloring
δ with minimum cost Ḱ (δ) in a weighted triangle free graphs by using an additional separation principle. Since we are handling hypostable sets we will have to introduce some blocking mechanism which will prevent us from introducing some edges into hypostable sets. If e is in some S_i, then no adjacent edge can be introduced into the same hypostable set S_i.

The algorithm will be based on a 'Contract or Connect" Principle; we call it the Contract or Connect Algorithm or shortly COCA. It is described in Table 5.1.

It enumerates in an implicit way all Hypocoloring of G and finds the minimum value of the cost Ḱ (δ) of such Hypocoloring δ.

Edges which are no longer allowed to be introduced into a hypostable set are blocked.

While II (H) separates the usual colorings of H as described above, procedure II (H) separates the Hypocoloring into two classes; the Hypocoloring where x and y (linked) are in the same hypostable set and the Hypocoloring where x and y in different sets.

Table 5.2.1 The COCA Algorithm

We now describe the two separation procedures:	
Separation Procedure I (H)	
(1)	Choose a free edge <x,y> in H;
(2)	Let H_1 be obtained from H by condensing <x,y> into a node xy with w(xy) = w(x) + w(y) all edges adjacent to xy are blocked;
	Let H_2 be obtained from H by blocking <x,y>;
(3)	
	Choose two non-adjacent nodes x,y in H;
Separation Procedure II (H)	Let H_1 be obtained from H by condensing <x,y> into a node xy with w(x,y) = max {w(x), w(y)}; all edges adjacent to xy remain blocked.
(1)	
(2)	
(3)	Let H_2 be obtained from by introducing the edge <x,y>, is blocked

At the final stage, a graph H will be clique with all edges blocked; the corresponding colorings can be we constructed by looking at the name of each node which is obtained by concatenation of the names of the nodes

which have been sequentially merged. The light version of the COCA algorithm is just the enumeration algorithm for usual (unweighted) colorings.

Even if G contains no triangles by assumption, the auxiliary graphs constructed by the separation procedures I and II may contain triangles.

Data: Weighted triangle free graphs G.

Output: A Hypocoloring δ with minimum cost $\hat{K}(\delta\ )$

```
(1)   Initialization: All edges are free; L (G)
      S(G) = best solution so far, $w^* = \infty$
(2)   While L (G)$\neq\phi$ do
      (a) Choose a graph H in L(G);
      (b) If H has at least one free edge then apply procedure I (H)
      else (all edges are blocked)
    (i)       If H is not a clique then apply procedure II (H)
      else (H is a clique with all edges blocked)

       (A)     $w = \sum_{v \in V(H)} w(v)$ ;
       (B)     If w < w* then S (G)  = H and w* = w;
(3)   Remove H from L and introduce the graphs $H_1$ and $H_2$ (Procedure by separation procedure) if they exist into L.
```

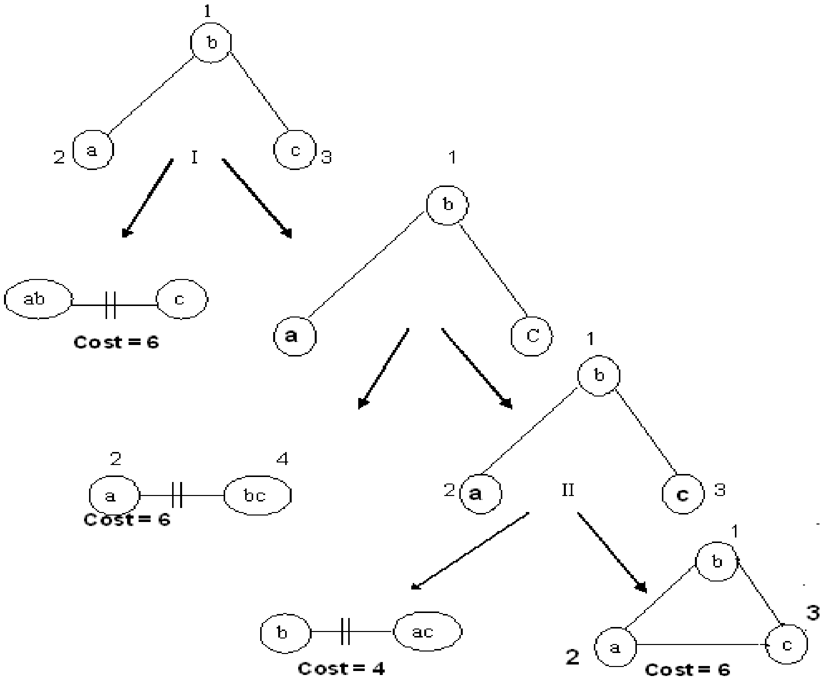


Figure 4.1 An Illustration of the COCA Algorithm

5.3 Conclusion

In this chapter, we describe the properties of optimal coloring and the COCA algorithm were described and illustrated for weighted triangular free graphs.

VI. Chromatic number, Subchromatic number and Hypochromatic of a shell graph.

6.1 Introduction

In this chapter we define shell graphs C (n, n-3) and we find the chromatic number, Subchromatic number and the Hypochromatic number of the shell graphs with examples.

6.2 Shell Graph

A graph obtained from the cycle Cn: <v_0,v_1,v_2,...,v_{n-1}> of order n by adding n-3 consecutive chords incident with a common vertex say v_0, is called *Shell Graph.* The shell graph of order n is denoted by C (n, n-3). The common vertex v_0 is called apex of the *shell.*

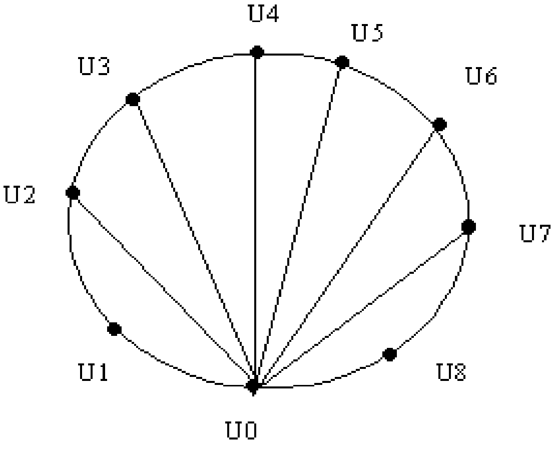


Figure 6.1 A Shell Graph C (9, 6)

6.3 The chromatic number of a shell graph

Result 6.3.1

χC (n, n-3)) = 3 for n≥3

6.4 TheSubchromatic number of a shell graph

Result 6.4.1

$$\chi_s(C\ (n,\ n\text{-}3)) = \begin{cases} 1, \text{ when } n = 3 \\ 2, \text{ when } n = 4,5,6,7 \\ 3, \text{ when } n \geq 8 \end{cases}$$

Consider the shell graph C (4, 1)

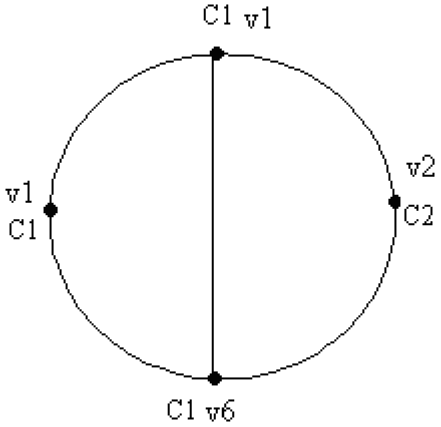


Figure 6.4.1 A Shell Graph C (4, 1)

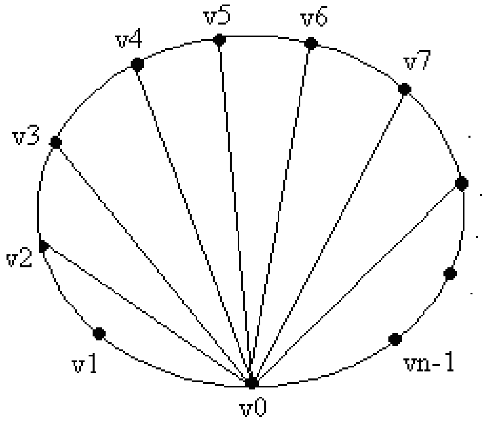


Figure 6.4.2 A Shell Graph C (n, n-3)

In particular, when n=8, we have,

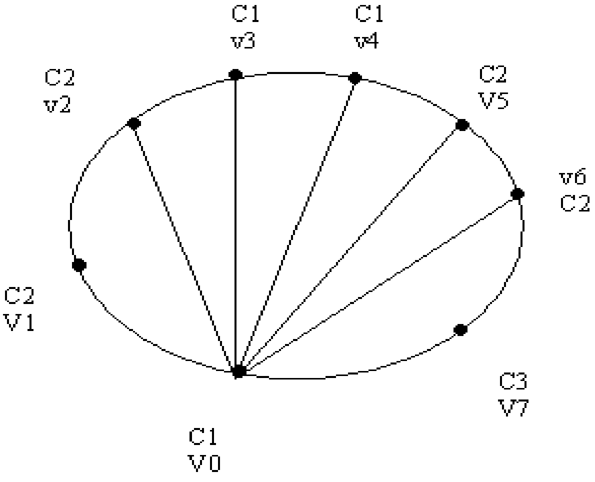


Figure 6.4.3 A Subcoloring of C (8, 5)

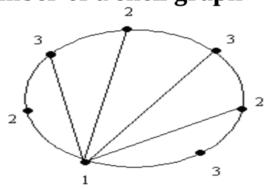
Result 6.4.2

If G is a connected graph with χ_s(G) = r and weight of each edge is 1, then (G) = r.

Result 6.4.3

χ_s(C(n,n-3)) = 1 when n=3
2when n= 4,5,6,7
3 when n≥ 8

6.5.1 Chromatic number of a shell graph



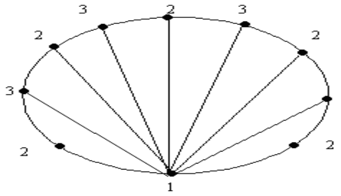
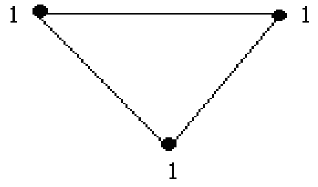


Figure 5.5 A shell Graph C (7, 4) Figure 5.6 A Shell Graph C (10, 7)

6.5.2 Sub chromatic number of a shell graph



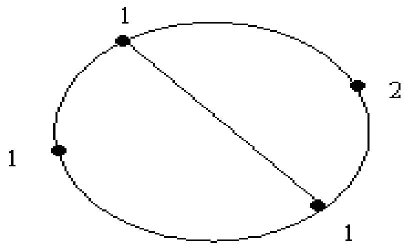


Figure 5.7 C (3, 0) having χ_s=1, Figure 5.8 C (4, 1) having χ_s=(C (4,1)) = 2

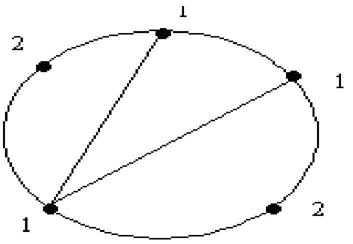


Figure 5.9 C (5, 2) with χ_s=2

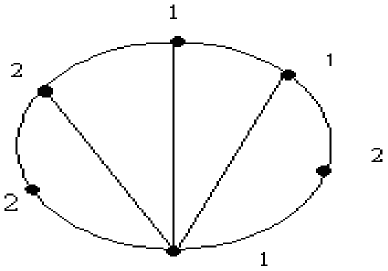


Figure 5.10 C (16, 3) with χ_s=2

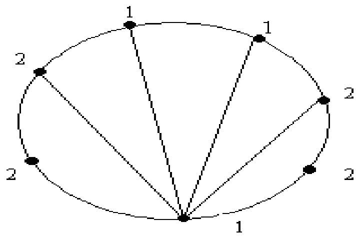


Figure 5.11 C (7, 4) with χ_s=2

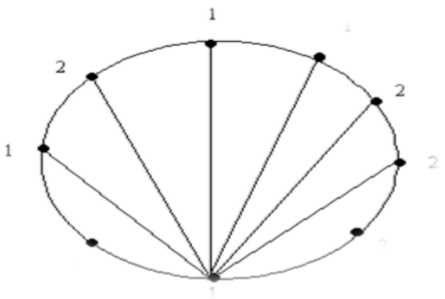
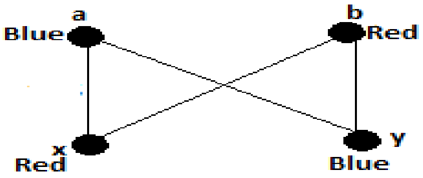


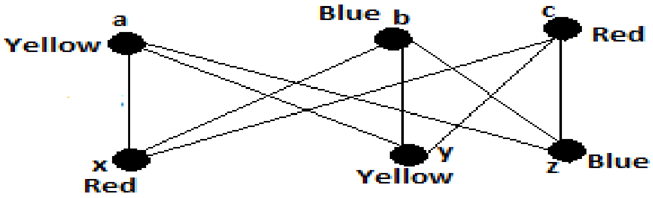
Figure 5.12 C (9, 6) with χ_s=3

VII. Subcoloring of Bipartite graphs

7.1 Sub chromatic number of complete bipartite graphs



7.1.1Sub chromatic number for the above graph is 2



7.1.2 Sub chromatic number for the above graph is 3

Hence from the above examples we can generalize the result of Subchromatic number of complete bipartite graph as,

$$\chi_s(G) = \begin{cases} n+1, \text{ when} m > n \\ \text{m+1 , when} m < n \\ \text{m or n ,when m = n} \end{cases}$$

7.2 Personnel Assignment Problem

7.2.1 Introduction

Consider a fuzzy graph where each vertex bears a weight which denotes the necessity/ efficiency needed/ available according to their meaning of representation. The edge weight represents the reliability between the two vertices.

Now the objective is to find an optimal assignment in such a way that the required efficiency is almost satisfiedand applicants are assigned to the jobs so that the reliability is maximized.

7.2.2 Definitions

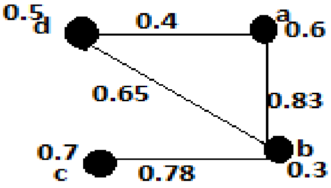
7.2.2.1 Fuzzy graphs

A fuzzy graph is defined to have a set of vertices and edges where each vertex bears a weight which varies depending on their attributes. The edges hold weights with respect to the relation between the vertices connecting them.

Example 7.2.2.1

There are four hills named a, b, c and d whose heights are 0.6, 0.3, 0.7, and 0.5 respectively. The distance between two hills are given as ab = 0.83, ad = 0.4, bd = 0.65 and bc = 0.78.

The above data can represented using fuzzy graphs where the vertex weight denote the altitudes of the hills and the edge weights relate the distance between the two hills.



Here V= {a, b, c, d} and E= {ab, bc, bd, da}

Here each vertex and edge holds weights. The weights are represented between the range 0-1.Hence the above graph is a fuzzy graph.

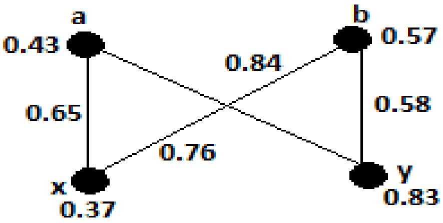
Two hills a and b each of height 0.6 units and 0.3 units are at a distance 0.4 units

7.2.2.2 Fuzzy bipartite graphs

A bipartite graph in which vertex and edge holds weight is called as the fuzzy bipartite graph.

Example 7.2.2.2

Here the vertex set denotes two different category- ingredients and food product.



The weights of the vertices (a, b) denotes the costs of the ingredients and the weights of the vertices (x, y) denotes the costs of the products.
The weights of the edges indicate the cost of the product made from the corresponding ingredients.

7.2.2.3 A Hypocoloring model for Personnel Assignment Problem

- Given a fuzzy graphwith the vertex denoting the efficiency of each category and the edge weight denotes the reliability between the two vertices(category)
- We obtain the Subcolor classes for the graph considering only the weights of the vertices.
- The Subcolor classes are partitioned into disjoint union of cliques in such a way that the difference of the weights of the two vertices in each clique is to be minimum.
- The Hypocoloring classes are obtained by considering the weights of the edges which yields the maximum value.

Consider the below graph.

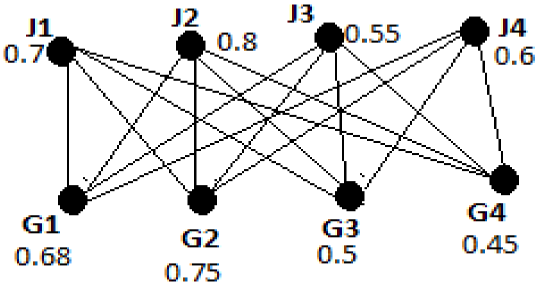


Fig 7.2.2.3.1A Personnel Assignment Problem (CBG)

There are four jobs and four group applicants with varying efficiencies.
The reliability (edge weights)are given by,
(J1,G1)= 0.8 (J2, G1) = 0.9 (J3, G1) = 0.7 (J4, G1) = 0.6
(J1,G2)= 0.7(J2,G2)= 0.4(J3, G2) = 0.5(J4, G2) = 0.7
(J1, G3) = 0.2(J2, G3) = 0.3 (J3,G3)= 0.4 (J4, G3) = 0.6
(J1, G4) = 0.4 (J2, G4) = 0.5 (J3, G4) = 0.8 (J4, G4) = 0.4

For example, (J3, G1) = 0.7 represents the efficiency of group 1 for the job 3.

Now finding the possible sub coloring for the above graph, we have,
$S_1 = \{(J1, G1), (J2, G2)(J3, G4), (J4, G3)\}$with each class having the efficiency deficient 0.02, 0.05, 0.1 and 0.1 respectively. The total efficiency deficient = 0.27
The Subchromatic number $\chi_s(CBG)$= 4.

$S_2 = \{(J1, G3), (J2, G2)(J3, G4), (J4, G4)\}$with each class having the efficiency deficient 0.2, 0.05, 0.1 and 0.08 respectively. The total efficiency deficient = 0.43
Here again the Subchromatic number $\chi_s(CBG)$= 4.

Now by considering the weights of the edges and finding the Hypocoloring we have,

ForS_1, the total reliability is 2.6 whereas forS_2, the total reliability is 2.0 and also the total efficiency deficient is low for S_1 is low when compared with S_2
Thus we choose the first assignment (S_1) as the required solution.

$$\begin{array}{ll} J1 & \to G1 \\ J2 & \to G2 \\ J3 & \to G4 \\ J4 & \to G3 \end{array}$$

withtotal reliability 2.6.

VIII. Conclusion

In this project we developed hypocoloring model for batch scheduling problems. We also found the sub chromatic number for shell graphs. We obtain some properties of optimal hypocoloring for any graph. Hence we developed an exponential algorithm for hypocoloring technique and it is illustrated. In this direction, we can find the subchromatic number and hypochromatic number for many family graphs.
Finally finding the solution of personnel assignment problem using hypo coloring model given the applicant's efficiencies as weights has been briefly discussed.

IX. Further research

Further research can be focused on considering our graph as fuzzy where each vertex and edge would bear a weight. We can proceed by relating the fuzzy graphs with the real time problem, and finding the solution for it using the Hypocoloring model.

References

Journal Papers:

[1] De Werra D., Demange M.., Monnot J. and Paschos V.T., *Discrete Applied Mathematics'*, *146*, 2005, 3-26.

[2] Bodlaender H.L., Jansen K. and Woe ginger G.J., 'Scheduling with incompatible jobs',*Discrete Applied Mathematics 55*, 1994, 219-232.

[3] Albertson M.O., Jamison R.E., Hedetniemi S.C. and Locke S.C. , 'The Subchromaticnumber of a graph', *Discrete Mathematics*, *74*, 1989 33-49.

[4] Sethuraman G. and Dhavamani R., 'Graceful Numbering of an Edge -gluing of Shell graphs' *Discrete Mathematics 218*,2000,283-287.

[5] Brown J.L. and Corneil D.G., 'On generalized graph colorings', *Journal Graph theory*, *11*, 1987, 87-99.