Algorithmic Operations Research

Organizing National Elections in India to Elect the 543 Members of the Lok Sabha

Bodhibrata Nag et Katta G. Murty

Volume 7, numéro 2, winter 2012

URI : https://id.erudit.org/iderudit/aor7_2art02

Aller au sommaire du numéro

Éditeur(s)

Preeminent Academic Facets Inc.

ISSN

1718-3235 (numérique)

Découvrir la revue

Citer cet article

Nag, B. & Murty, K. G. (2012). Organizing National Elections in India to Elect the 543 Members of the Lok Sabha. *Algorithmic Operations Research*, *7*(2), 55–70.

Résumé de l'article

There are 833 thousand polling stations in all of the 543 parliamentary constituencies spread over 35 states of India. On the day elections are being held in any one of these polling stations, a minimum of 4 Central Police Force(CPF) personnel must be deployed there, to maintain law and order and guarantee that voters can vote freely without being intimidated by anyone. As the number of CPF personnel available for this activity is limited, it is not possible to hold the Indian General elections on a single day over the whole country. So the set of 35 States of India is partitioned into a number of subsets, with elections in each subset of states being held on a single day. This partition is required to satisfy the constraints that the states in each subset are contiguous, and the subsets themselves must be contiguous. We present a method for organizing the Indian General Elections subject to these constraints, and minimizing the total number of election days required, and the total cost for the movement of CPF personnel involved. The method is based on the shortest Hamiltonian path problem, a tour segmentation problem defined in the paper, and the bipartite minimum cost flow problem.

Algorithmic Operations Research Vol.7 (2013) 55–70

Organizing National Elections in India to Elect the 543 Members of the Lok Sabha

Bodhibrata Nag [a]

[a]Indian Institute of Management Calcutta, Joka, Diamond Harbour Road, Kolkata, 700104, India

Katta G. Murty [b]

[b]Department of IOE, University of Michigan, Ann Arbor, MI 48109-2117, USA

Abstract

There are 833 thousand polling stations in all of the 543 parliamentary constituencies spread over 35 states of India. On the day elections are being held in any one of these polling stations, a minimum of 4 Central Police Force(CPF) personnel must be deployed there, to maintain law and order and guarantee that voters can vote freely without being intimidated by anyone. As the number of CPF personnel available for this activity is limited, it is not possible to hold the Indian General elections on a single day over the whole country. So the set of 35 States of India is partitioned into a number of subsets, with elections in each subset of states being held on a single day. This partition is required to satisfy the constraints that the states in each subset are contiguous, and the subsets themselves must be contiguous. We present a method for organizing the Indian General Elections subject to these constraints, and minimizing the total number of election days required, and the total cost for the movement of CPF personnel involved. The method is based on the shortest Hamiltonian path problem, a tour segmentation problem defined in the paper, and the bipartite minimum cost flow problem.

Key words: OR in government; Scheduling; Graph partitioning; Hamiltonian path problem; tour segmentation problem; minimum cost flow

1. Brief History of National Polls in India.

Three years after gaining independence from Britain, Bharat (India) became the Bharat Ganrajya (The Republic of India, or Indian Republic) by adopting the Constitution of India in 1950. Now the Indian Republic comprises 28 States and 7 Union Territories, which we will refer to as the 35 States of India in the sequel. The Indian parliamentary form of government is federal in structure with legislative powers distributed between the Parliament of India and State Legislatures. The Parliament of India comprises two legislative bodies – the Rajya Sabha (Upper house, corresponds to the "Senate" in the US, or the "House of Lords" in the UK), and the Lok Sabha (Lower House, corresponds to the "Congress" in the US, or the "House of Commons" in the UK). The 250 members of the Rajya Sabha are indirectly elected by legislators of States and Union Territories comprising the Union of India. The 543 members of the Lok Sabha are directly elected by universal adult franchise by the electorate of all the 35 States through the "National Elections", called **General Elections** in India. The term of office each Lok Sabha is five years from the date of its first meeting, unless dissolved earlier due to the ruling party losing a vote on a no-confidence motion in the Lok Sabha. These General Elections have been held for the first time in the history of India in 1951-52 after the adoption of the constitution of India; and regularly after that as depicted in Table 1. In this paper we will consider only the organization of general elections for electing the members of the Lok Sabha in India.

The total membership of the Lok Sabha is distributed amongst the 35 States in such a manner that the ratio of the population to the number of seats allotted to any State is nearly the same. The geographical area of the State is then demarcated into a number of territorial constituencies (with geographical boundaries), equal to the number of seats allotted, such that the population of all constituencies in that State is nearly the same.

Email: Bodhibrata Nag [bnag@iimcal.ac.in], Katta G. Murty [murty@umich.edu].

Table 1

Lok Sabha	General Elections	Date of first meeting	Date of dissolution
1	25 October 1951 to 21 February 1952	13 May 1952	4 April 1957
2	24 February to 14 March 1957	10 May 1957	31 March 1962
3	19 to 25 February 1962	16 April 1962	3 March 1967
4	17 to 21 February 1967	16 March 1967	27 December 1970
5	1 to 10 March 1971	19 March 1971	18 January 1977
6	16 to 20 March 1977	25 March 1977	22 August 1979
7	3 to 6 January 1980	21 January 1980	31 December 1984
8	24 to 28 December 1984	15 January 1985	27 November 1989
9	22 to 26 November 1989	18 December 1989	13 March 1991
10	20 May to 15 June 1991	9 July 1991	10 May 1996
11	27 April to 30 May 1996	22 May 1996	4 December 1997
12	16 to 23 February 1998	23 March 1998	26 April 1999
13	5 September to 6 October 1999	20 October 1999	6 February 2004
14	20 April to 10 May 2004	2 June 2004	18 May 2009
15	16 April to 13 May 2009	1 June 2009	-

General Elections held in India

Since there are large variations in population densities across States, constituencies vary largely in terms of geographical area- thus Ladakh (the constituency with largest area) covers 173266 sq.km in contrast to Delhi-Chandni Chowk (the constituency with smallest area) which covers only 11 sq.km. Each constituency has a large number of polling stations distributed across the constituency such that voters can reach the polling stations to cast their vote with minimum travel. The distribution of membership of the Lok Sabha and the total number of polling stations for each state is given in Table 2.

The General Elections of India are the world's biggest election exercise. During the 2009 General Elections, a 717 million strong electorate exercised their franchise through 1.3 million Electronic Voting Machines deployed in 834 thousand polling stations spread across the length and breadth of India to elect 543 Members of the Lok Sabha from amongst 8 thousand candidates contesting the elections. The only other comparable elections are the European Parliament elections with an electorate of 500 million and the US Congress elections with electorate of 312 million.

The responsibility for conducting the elections to the Lok Sabha is vested in the Election Commission of India according to the provisions of Article 324 of the Constitution of India. The Election Commission operates through its secretariat based at New Delhi manned by about 300 officials. It is assisted at the State level by the Chief Electoral Officer of the State, who is appointed by the Election Commission in consultation with the State government. The Chief Electoral Officer is assisted by District Election Officers, Electoral Registration Officers, and Returning Officers at the constituency level. In addition, the Election Commission co-opts a large number of officials from the Central (or federal) and State governments for about two months during each General Election, for conducting the elections. About 5 million officials were deployed during the 2009 General Election.

Elections in the past have been marked by instances of voter intimidation through violence or harassment in various forms, as well as clashes between political opponents (Scharff [7]). These incidences have been largely arrested through deployment of additional police forces during the polling process in order to bring peace, restore confidence in candidates and voters, and thereby ensure fair and free elections.

The Constitution of India mandates that maintenance of law and order is the responsibility of the States. Thus while all States maintain police forces totaling about 1.5 million, the average police-population ratio for all the States is only 133 police per 100,000 (National Crime records Bureau, 2010 [6]) in comparison with average international ratio of 342 (Stefan Harrendorf, 2010 [8]). The Central Government therefore maintains **Central Police Forces** numbering about 800 thousand under 7 different divisions, to complement the State police, whenever and wherever required. (Bureau of Police Research & development [1]).

Since the State police are the arm of the State governments, allegations of partisan conduct of police in

Table 2

Sl.No	**State**	**Number of Members of the Lok Sabha**	**Total Number of polling stations**
1	Andhra Pradesh	42	66760
2	Arunachal Pradesh	2	2057
3	Assam	14	18828
4	Bihar	40	57020
5	Goa	2	1339
6	Gujarat	26	42568
7	Haryana	10	12894
8	Himachal Pradesh	4	7253
9	Jammu & Kashmir	6	9129
10	Karnataka	28	46576
11	Kerala	20	20510
12	Madhya Pradesh	29	47812
13	Maharashtra	48	82598
14	Manipur	2	2193
15	Meghalaya	2	2117
16	Mizoram	1	1028
17	Nagaland	1	1692
18	Orissa	21	31617
19	Punjab	13	18846
20	Rajasthan	25	42699
21	Sikkim	1	493
22	Tamil Nadu	39	52158
23	Tripura	2	3008
24	Uttar Pradesh	80	129446
25	West Bengal	42	66109
26	Chattisgarh	11	20984
27	Jharkhand	14	23696
28	Uttarakhand	5	9003
29	Andaman & Nicobar Islands	1	347
30	Chandigarh	1	422
31	Dadra & Nagar Haveli	1	161
32	Daman & Diu	1	94
33	NCT of Delhi	7	11348
34	Lakshadweep	1	40
35	Puducherry	1	856

Number of constituencies and polling stations in each State

enforcing law and order during the campaign closing phases and during the day of elections are likely. It has therefore become universal practice to deploy **Central Police Forces (CPF)**, in addition to State police at all polling stations during the General Elections. However, only about a quarter of the CPF can be spared for deployment during the elections, which amounts to only about 200,000 personnel. Thus General Elections are spread over different days with each day covering a few states only, such that the required number of CPF personnel can be deployed across all polling stations of all constituencies of those states. The days of elections are spread a few days apart to allow re-deployment of paramilitary personnel and allow them to be familiar with their constituencies. For example, the 2009 General Election was conducted in five phases on 16 April, 23 April, 30 April, 7 May and 13 May.

The movement of CPF personnel from their bases to the polling stations in the different phases and their subsequent return to the bases is a gigantic exercise, requiring coordination between different agencies such as CPF operations, Election Commission and State Chief

Electoral Officers, District Election Officers, Railways, airlines and the Indian Air Force. In the 2009 General Election, 119 special trains, 65 sorties by Indian Air Force transport aircraft, 600 sorties by Indian Air Force helicopters and Air India chartered flights were used for the cross-country movement of CPF personnel (Election Commission of India, 2009 [2]).

However the process of scheduling the elections and movement of police personnel is done manually by the Election Commission. This paper proposes an optimization methodology to find an optimum plan for organizing the General Elections for all the 543 parliamentary constituencies in the minimum possible time, with the available CPF personnel, while minimizing the total cost for the police movement involved. The paper is organized as follows: the problem statement and description of all the data is in Section 2; representation, methodology and algorithms for the problem are described in Section 3; followed by discussion of the solutions obtained in Section 4 and conclusions in Section 5.

2. The problem statement and description of all the data

The total number of polling stations of all the 543 parliamentary constituencies, spread over 35 states is 833,701. If 4 CPF (Central Police Forces) personnel are deployed at each polling station, the total requirement of police personnel is 3.3 million if elections are to be held on the same day all over the country. However, since only about a quarter of CPF amounting to 200,000 personnel can be spared for deployment during the elections, so additional reserve CPF and army personnel are also used to bring the number to 1.5 million for deployment during the elections. In the sequel, we will refer to this group of police with the responsibility of maintaining law and order at the polling booths, while elections are going on there, as "CPF personne". Even then it can be seen that it is not possible to conduct elections for all the 543 parliamentary constituencies on a single day. Thus elections will have to be conducted in phases, with CPF personnel movement between constituencies in the phase intervals. The proposed method assumes that the CPF personnel movement will be entirely by air, except the 'last mile' movement between the airports and the constituencies.

2.1. *The Constraints in the Problem and the Objective Function to be Optimized*

While conducting the elections in phases, the Election Commission requires that the following constraints should be satisfied as far as possible:

2.1.1. *Constraints to be Satisfied*

(a) In every State , the elections in all constituencies in it should be held on a single day
(b) As far as possible, States in which elections are held on a day must be contiguous
(c) Set of States in which elections are held on consecutive polling days should be contiguous
(d) General elections over the whole country should be completed using the smallest number of polling days
(e) At every polling station, 4 CPF personnel should be deployed on the election day
(f) The total number of CPF personnel available for deployment at polling stations on any polling day is at most 1,500,000 or 1.5 million. The proposed method

incorporates all these constraints, in the model used to solve the problem.

2.1.2. *Selection of the objective function to optimize*

In the original statement of the problem, the unit for measuring the objective function is stated in terms of people-miles. But the CFP personnel deployed move from a state to next by air (as far as possible), and from the airport to the polling stations in the constituency by road vehicles; and the costs per person-mile by air and road are very different. It is not logical or reasonable to add people-miles by different modes of travel directly to get the objective function to be minimized; particularly since in the end all travel has to be paid for in units of money, **Indian Rs (Indian Rupees or)** , in the total travel cost of all the CFP personnel involved.
So, we use the objective function as the total cost of travel of all the CFP personnel involved in the General Elections. Air travel cost should be the cost of air travel of all the moves made by air for all these personnel. Road travel can be in terms of cost incurred for it, the total cost of mileage of all the road vehicles used for all these personnel.

2.2. *The data for the problem*

All the data for the problem can be obtained by sending an e-mail to the first author.

2.2.1.

The first data set required for the problem is number of polling stations in each constituency in each state, the name of the nearest airport to that constituency (CPF personnel deployed to polling stations in this constituency will use this airport to arrive in this constituency and depart from it to their next assignment), and the road distance of this constituency to that airport. In the following Table 3 we show a portion of this information for one selected constituency in each State.

2.2.2.

The second data set required for the problem is the air travel cost in (Rupees) between various pairs of airports. Actual air travel cost of CPF polling personnel is hard to get exactly since some of it is on Indian Air Force Transport Aircraft, helicopters, and some on Air India chartered flights. So this data is based on estimated average cost in 10/mile (based on 2012 prices) on these different types of flights used, and the air distance data between pairs of airports. In the following Table 4 we show a portion of this information for a few pairs of airports.

2.2.3.

The third data set required for the problem is the estimated cost in (Rupees) of the road or train travel of CPF polling personnel from airport used to the polling stations in the constituency and back for each constituency. This data is based on estimated average cost in .3/mile by these modes of travel(road or rail). In the following Table 5 we show a portion of this information for a few selected constituencies.

2.2.4.

The fourth data set required for the problem is the list of adjacent states for each state. This data is shown in full in the following Table 6. Two states are defined to be **adjacent** if they share a common boundary line.

2.2.5.

The fifth data set required for the problem is the cost in of travel/person from the various bases from where CPF polling personnel come from, to each constituency. In the following Table 7 we show a portion of this information for a few selected constituencies and bases.

2.2.6.

The sixth data set required for the problem is the number of personnel that come from each base. We assume 15 bases at Agartala, Mumbai, Kolkata, Chandigarh, Delhi, Guwahati, Hyderabad, Imphal, Jaipur, Jammu, Jorhat, Lucknow, Patna, Shillong and Raipur.

2.2.7.

The seventh data set required for the problem is the cost of travel/person from each constituency to each constituency across the country. In the following Table 8 we show a portion of this information for a few pair of selected constituencies.

3. A graph representation of the problem

A **graph** (also known as an **undirected network**) is a pair of sets *G = (N, A)* where *N* is a set of **nodes** (also called **vertices**, these are numbered serially and referenced by their numbers), and *A* is a set of lines or **edges** (also called arcs in some books), each edge joining a pair of nodes. If an edge joins the pair of nodes *i, j* it is represented by the pair *(i, j)* [in some books the symbol *(i; j)* is used instead]; this edge is said to be incident at nodes *i* and *j*. Nodes *i, j* are said to be **adjacent** if there is an edge joining them. See Murty[1992] for a discussion of networks and network algorithms. The network *G* is said to be **connected** if for every pair of nodes *p, q* in it, there is a path of edges in *G* connecting them; otherwise it is not connected. See Figures 6, 7. Nodes are circles with its number entered inside the circle; each edge is a straight line joining the pair of nodes on it.

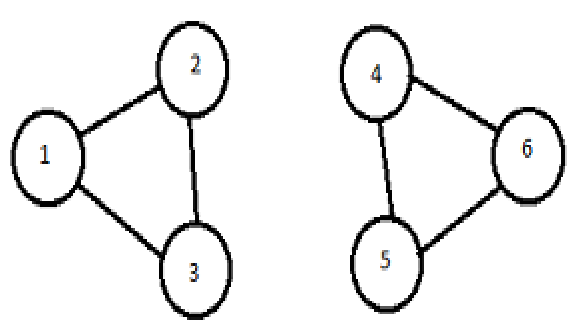

Figure 6: A 6 node, 6 edge network which is not a connected network (for example there is no path consisting of edges between nodes 1, 4 in this network).

Table 3

Sl.No	State	Constituency	No.of Polling Stations	Nearest Airport	Distance to Airport (miles)
1	Andhra Pradesh	Adilabad	1464	Ramagundam	85
2	Arunachal Pradesh	Arunachal East	851	Pasighat	52
3	Assam	Karimganj	1229	Silchar	27
4	Bihar	Purvi Champaran	1193	Muzaffarpur	46
5	Goa	South Goa	660	Dabolimgoa	0
6	Gujarat	Anand	1510	Vadodara	22
7	Haryana	Kurukshetra	1263	Chandigarh	52
8	Himachal Pradesh	Mandi	1921	Kulu	20
9	Jammu & Kashmir	Anantnag	1502	Srinagar	31
10	Karnataka	Dharwad	1455	Hubli	12
11	Kerala	Wayanad	988	Kozhikode	40
12	Madhya Pradesh	Bhind	1659	Gwalior	44
13	Maharashtra	Dhule	1624	Aurangabad	79
14	Manipur	Inner Manipur	970	Imphal	0
15	Meghalaya	Shillong	1326	Shillong	0
16	Mizoram	Mizoram	1028	Aizawl	0
17	Nagaland	Nagaland	1692	Dimapur	29
18	Orissa	Cuttack	1319	Bhubaneswar	13
19	Punjab	Jalandhar	1764	Ludhiana	34
20	Rajasthan	Sikar	1574	Jaipur	63
21	Sikkim	Sikkim	493	Darjeeling	30
22	Tamil Nadu	Viluppuram	1376	Pondicherry	19
23	Tripura	Tripura West	1558	Agartala	0
24	Uttar Pradesh	Rae Bareli	1576	Lucknow	46
25	West Bengal	Jalpaiguri	1560	Bagdogra	27
26	Chattisgarh	Raigarh	2166	Bilaspur	81
27	Jharkhand	Kodarma	2097	Gaya	43
28	Uttarakhand	Garhwal	1876	Dehradun	46
29	Andaman & Nicobar Islands	Andaman & Nicobar Islands	347	Portblair	0
30	Chandigarh	Chandigarh	422	Chandigarh	0
31	Dadra & Nagar Haveli	Dadar & Nagar Haveli	161	Daman	15
32	Daman & Diu	Daman & Diu	94	Daman	0
33	NCT OF Delhi	New Delhi	1540	Delhi	9
34	Lakshadweep	Lakshadweep	40	Agatti	0
35	Puducherry	Puducherry	856	Pondicherry	0

Number of polling stations and nearest airport of select constituencies

Table 4

	AGARTALA	AGATTI	AGRA	AHMEDABAD	AIZAWL	AKOLA
AGARTALA	-	15438	8597	11878	914	9389
AGATTI	15438	-	11907	8429	16183	7520
AGRA	8597	11907	-	4450	9485	4527
AHMEDABAD	11878	8429	4450	-	12791	3272
AIZAWL	914	16183	9485	12791	-	10276
AKOLA	9389	7520	4527	3272	10276	-

Air travel cost between select pair of airports

Table 5

State/Union-Territory	**Parliament Constituency Name**	**Number of Polling Stations**	**Nearest Airport**	**Cost of Road Travel from Parliament Constituency to nearest Airport (in Rupee/person)**
Andhra Pradesh	Medak	1571	HYDERABAD	289
Assam	Barpeta	1396	GUWAHATI	281
Bihar	Madhubani	1354	MUZAFFARPUR	274
Gujarat	Surendranagar	1768	AHMEDABAD	353
Kerala	Kannur	983	KOZHIKODE	389

Cost of road/train travel in Rupee/person between constituency and nearest airport, for selected constituencies.

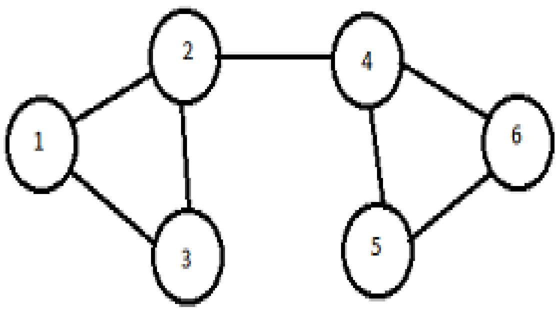

Figure 7: A 6 node, 7 edge network that is a connected network

A **partial network** of N is a network $G_1 = (N_1, A_1)$ where the set of nodes N_1 is a subset of N, and the set of edges N_1 is the set of all edges of G that have both their incident nodes in N_1. G_1 is said to be the **partial network of** G **induced by** the subset of nodes N_1, it is connected if it forms a connected network. For example for the network G in Figure 5, $G_1 = (\{1, 2, 3\}, \{(1, 2), (2, 3), (3, 1)\})$ is a connected partial network; and $G_2 = (\{1, 2, 3, 4\}, \{(1, 2), (2, 3), (3, 1)\})$ is a partial network which is not connected because there is no path connecting nodes 1 and 4 in it.

Let $\{N_1, N_2, \ldots, N_k\}$ be a partition node set N in G (i.e., their union is G, and every pair of subsets in this partition are mutually disjoint). For $t = 1$ to k, let $G_t = (N_t, A_t)$ be the partial network of G induced by N_t. Then $\{G_1, \ldots, G_k\}$ is said to be a **partition** of G. The **graph partitioning problem** is the general problem of finding a partition of a given graph G subject to specified constraints that minimizes a specified objective function.

Our polling problem can be viewed as an instance of the graph partitioning problem. To see this let $NI = \{1, \ldots, 35\}$ where node i represents the state of India with serial number i in Table 2. Let $AI = \{(i, j)$: nodes i, j correspond to adjacent states as given in Table 4$\}$. Then $(NI, AI) = GI$ is known as the **adjacency network** for states in India.

With 1.5 million CPF personnel available for deployment for the elections, and the required 4 CPF personnel at every polling station, on a single day elections can be held at 1500000/4 = 375000 polling stations at most. Since there are 833701 polling stations over the whole country, and 833701/375000 is strictly between 2 and 3,

we need at least 3 polling days to complete holding the elections over the whole country. (1)

So, on any single day, elections are held in a subset of states in the country. Suppose elections are held on k different days. We know that $k >= 3$. Let NI_t be the set of states in which elections are held on the tth day for $t = 1$ to k, and let $GI_t = (NI_t, AI_t)$ be the partial network of GI induced by the subset of nodes NI_t. So, our problem boils down to finding the partition $(GI_1, \ldots, GI_k)$ of GI satisfying constraints (a) to (f) listed above in Section 2.1.1, and **sequence** these partial networks generated corresponding to the election days 1, 2, ..., k; while minimizing the total cost of moving the CPF personnel across the country to monitor the polling stations as required. So our problem actually involves a graph partitioning problem, and a sequencing problem.

In constraint (b) of Section 2.1.1, the meaning of the word "contiguous" is left somewhat vague. We will interpret it to mean that the partial network GI_t of GI induced by the set of states in which elections are held on the tth day should be a connected network for each $t = 1$ to k. Also, we will interpret constraint (c) of Section 2.1.1 that the set of states in which elections are held on consecutive days should be "contiguous" to mean that there should be at least one edge in GI joining a node in NI_t to a node in NI_{t+1} for each $t = 1$ to k-1.

Table 6

State	Adjacent States
Andhra Pradesh	Orissa,Chattisgarh, Maharashtra, Karnataka, Tamil Nadu
Arunachal Pradesh	Assam
Assam	West Bengal, Arunachal Pradesh, Nagaland, Manipur, Tripura, Meghalaya, Mizoram
Bihar	Uttar Pradesh, Jharkhand, West Bengal
Goa	Karnataka, Maharashtra
Gujarat	Maharashtra, Madhya Pradesh, Dadra & Nagar Haveli, Daman & Diu, Rajasthan
Haryana	Punjab, NCT-Delhi, Chandigarh, Himachal Pradesh, Uttar Pradesh, Rajasthan
Himachal Pradesh	Punjab, Jammu & Kashmir, Uttar Khand, Haryana, Chandigarh
Jammu & Kashmir	Punjab, Himachal Pradesh
Karnataka	Goa, Maharashtra, Andhra Pradesh, Tamil Nadu, Kerala, Lakshadweep
Kerala	Karnataka, Tamil Nadu, Lakshadweep
Madhya Pradesh	Maharashtra, Gujarat, Rajasthan, Uttar Pradesh, Chattisgarh
Maharashtra	Goa, Karnataka, Andhra Pradesh, Chattisgarh, Madhya Pradesh, Gujarat, Daman & Diu, Dadra & Nagar Haveli
Manipur	Nagaland, Mizoram, Assam
Meghalaya	Assam
Mizoram	Assam, Tripura, Manipur
Nagaland	Manipur, Assam
Orissa	Andhra Pradesh, Chattisgarh, Jharkhand, West Bengal
Punjab	Rajasthan, Haryana, Himachal Pradesh, Jammu & Kashmir, Chandigarh
Rajasthan	Gujarat, Madhya Pradesh, Uttar Pradesh, Haryana, Punjab
Sikkim	West Bengal
Tamil Nadu	Kerala, Karnataka, Andhra Pradesh, Puducherry
Tripura	Assam, Mizoram
Uttar Pradesh	Uttarakhand, Himachal Pradesh, Haryana, Rajasthan, NCT-Delhi, Madhya Pradesh, Chattisgarh, Jharkhand, Bihar
West Bengal	Orissa, Jharkhand, Bihar, Sikkim, Assam
Chattisgarh	Orissa, Andhra Pradesh, Maharashtra, Madhya Pradesh, Uttar Pradesh, Jharkhand
Jharkhand	Orissa, Chattisgarh, Uttar Pradesh, Bihar, West Bengal
Uttarakhand	Himachal Pradesh, Uttar Pradesh
Andaman & Nicobar Islands	Tamil Nadu, Andhra Pradesh
Chandigarh	Punjab, Haryana, Himachal Pradesh
Dadra & Nagar Haveli	Gujarat, Maharashtra
Daman & Diu	Gujarat, Maharashtra
NCT of Delhi	Haryana, Uttar Pradesh
Lakshadweep	Kerala, Goa, Karnataka
Puducherry	Tamil Nadu

Adjacent states of each state

Table 7

State/Union-Territory	ParliamentConstituency	Central Police Force Bases			
		Agartala	Mumbai	Chandigarh	Kolkata
Andhra Pradesh	Peddapalle	8474	4353	6425	8505
Jammu & Kashmir	Baramulla	12377	10652	11615	2756
Karnataka	Bijapur	11164	2579	9120	9409
Madhya Pradesh	Tikamgarh	7605	6471	5983	4917
Punjab	Amritsar	11385	8812	10390	1290

Cost of travel in Rs./person from a few selected bases to some constituencies

Table 8

	Hassan (Karnataka)	Vidisha (Madhya Pradesh)	Mayurbhanj (Orissa)	Sangrur (Punjab)	Tura (Meghalaya)
Medak (Andhra Pradesh)	4390	4595	6751	10048	10598
Nagpur (Maharashtra)	6703	2000	4920	7305	7953
Uluberia (West Bengal)	10966	7286	1704	10000	3046
Fatehpur (Uttar Pradesh)	10672	3307	4885	4676	6421
Nowgong (Assam)	14938	10235	5270	11064	2231

Cost of travel in Rs./person between selected pairs of constituencies

3.1. *A Hamiltonian Path Heuristic For the Problem*

A **Hamiltonian path** in an undirected network *G* is a path consisting of edges in the network that contains all the nodes in the network. Given the lengths of all the edges in *G*, the **Hamiltonian path problem** in *G* is that of finding a shortest Hamiltonian path in *G*. A Hamiltonian path in G is a path in G that contains all the nodes in G; and a shortest Hamiltonian path is one whose length (the length of a Hamiltonian path is the sum of the lengths of the edges in it) is the shortest among all Hamiltonian paths. In some applications, we may require Hamiltonian paths beginning with a specified node as the initial node. This can easily be transformed into the well known **traveling salesman problem**.

Since our CPF polling personnel team has to cover each of the states in India satisfying the constraints listed above, the shortest Hamiltonian path in the network *GI* using the cost of traveling/person between states *i, j* as the length of edge *(i, j)* , may provide useful information to develop a good solution to our problem. But there are important differences. No member of the CPF polling team visits all the states, everyone visits only a subset of the states, so our problem is really one of partitioning *GI* into connected partial networks $GI_1, \dots, GI_k$ of *GI*, corresponding to election days 1 to *k* such that there is an edge joining a node in GI_t to a node in GI_{t+1} for all *t* = 1 to *k-1*; and each member of the CPF polling team visits one node in each of GI_t for *t* = 1 to *k*.

Define a **segment of a Hamiltonian path** as the portion of this path between a pair of nodes on it. One way of generating partial networks of *GI* satisfying the contiguity requirements in our problem is to obtain a minimum cost Hamiltonian path in *GI*, and then divide it into segments of required size, and take the partial networks of *GI* to be the partial networks induced by the sets of nodes of the various segments. This is the basis for this method, which we will describe next. The method has two stages. Stage 1 determines the election day for each state with day 1 as the starting day of the elections.

Selecting the sequence of partial networks of *GI* to cover the various polling days as the sequence of segments along the shortest Hamiltonian path helps keep the cost of movements of the NPF polling team between consecutive polling days small, thus achieving our objective of minimizing the total cost of travel of this team in conducting the elections, while satisfying the contiguity requirements (b), (c) among the specified constraints of Section 2.1.1.

3.1.1. *Stage 1: Determining the Election Day for Each State*

There may be up to 13 different airports that the CPF polling team will use to enter and leave a state. In reality then, each individual of this team has a separate problem to be solved to estimate correctly the cost incurred for his participation; but all the data for that individual is not known until the solution of this stage is determined, making it impractical to get into such precise detail.

So, as an approximation, we will assume that all members of the CPF team visiting a state will travel to and from this state use one airport, the one that majority of this team will use (this is a reasonable and good approximation).

So,

make the cost coefficient of edge *(i, j)* in *GI* to be c_{ij} = cost of air travel/person between the airports (determined as above) corresponding to states *i, j*.

For each node *i* in *GI* define its weight w_i = number of polling stations in state *i*.

As the polling team moves from one state to the other, our original problem is actually many Hamiltonian path problems, one for each individual in the CPF polling team, depending on the airports they will use for each move. By using a single airport for each state, one that majority will use, we will adopt a single countrywide shortest Hamiltonian path as an approximation to the collection of all of them. This Stage 1 involves two Steps, we will describe them now.

3.1.1.1 Step 1: Finding the Shortest Hamiltonian Path in *GI*

We assume that the national elections in India always begin in the Nation's Capital, "NCT of Delhi" (State number 33). So, find a shortest Hamiltonian path in *GI* beginning with State 33 as the starting state, and covering all the airports selected above corresponding to the various states. Let this Hamiltonian path be *H*. The order of states on *H*, is the order in which elections will be held in the solution determined by this method.

Figure 8: Shortest Hamiltonian Path covering the airports used by the CPF team for traveling to and from the states.

Re-numbering of the states: Re-number the states in the order of appearance on *H* beginning with number 1 for the state "NCT of Delhi", the starting state. **In the sequel, we will refer to the states by these numbers.**

3.1.1.2 Step 2: Determining the Election Day for each state:

Define

Weight of a segment of *H* = sum of the weights w_i of nodes on it

$$w_o = \textbf{maximum number of polling stations that the CPF polling team can handle in a day} = 1500000/4 = 375000 \quad (2)$$

In this step we will break up H into the smallest number of segments subject to the constraint that the weight of each segment is $<= w_0$; each segment corresponds to an election day in the solution developed by this method. We discuss two algorithms for breaking up *H* into segments. This problem is known as the **tour segmentation problem**.

Algorithm 1: A Simple Heuristic Algorithm for Breaking Up *H* Into Segments

In this algorithm segments are formed one by one. Starting at node 1, move along *H* until the sum of the weights of states included so far is $<= w_0$, and exceeds w_0 if the next state is included. At this time complete the current segment, remove it from *H*, and continue the same way with the remaining part of *H*.

Let *k* be the number of segments obtained by this heuristic. Number these segments 1 to *k* in the order of appearance on *H*. The solution obtained by this heuristic is to hold elections in all the states in segment *p* on day *p*, for each *p* = 1 to *k*.

We applied this algorithm and found the solution to the problem, *k* in it is = 3. From (1), we conclude that this solution is optimal to the problem of breaking up H into the smallest number of segments subject to the constraint that the weight of each segment should be $<= w_0$. According to this solution, the elections will be held according to the following schedule:

- Day 1: NCT-Delhi, Haryana, Punjab, Jammu & Kashmir, Himachal Pradesh, Chandigarh, Uttarakhand, Uttar Pradesh, Bihar, Sikkim, Jharkhand, West Bengal, Meghalaya, Assam, Arunachal Pradesh,

Table 9

Arc no. *i* on *H*	**Start node on arc *i***		**End node on arc *i***		**Length of arc *i* on *H* (miles)**
	State/Union Territory no.	**Airport**	**State/Union Territory no.**	**Airport**	
i = 1	1 =NCT Delhi	Delhi	2 = Haryana	Delhi	0
2	2 = Haryana	Delhi	3 = Punjab	Ludhiana	178
3	3 = Punjab	Ludhiana	4 = Jammu & Kashmir	Srinagar	228
4	4 = Jammu & Kashmir	Srinagar	5 = Himachal Pradesh	Kulu	199
5	5 = Himachal Pradesh	Kulu	6 = Chandigarh	Chandigarh	86
6	6 = Chandigarh	Chandigarh	7 = Uttarkhand	Dehradun	70
7	7 = Uttarkhand	Dehradun	8 = Uttar Pradesh	Lucknow	298
8	8 = Uttar Pradesh	Lucknow	9 = Bihar	Muzaffarpur	279
9	9 = Bihar	Muzaffarpur	10 = Sikkim	Darjeeling	189
10	10 = Sikkim	Darjeeling	11 = Jharkhand	Ranchi	316
11	11 = Jharkhand	Ranchi	12 = West Bengal	Kolkata	201
12	12 = West Bengal	Kolkata	13 = Meghalaya	Shillong	305
13	13 = Meghalaya	Shillong	14 = Assam	Guwahati	46
14	14 = Assam	Guwahati	15 = Arunachal Pradesh	Zero	160
15	15 = Arunachal Pradesh	Zero	16 = Nagaland	Dimapur	116
16	16 = Nagaland	Dimapur	17 = Manipur	Imphal	77
17	17 = Manipur	Imphal	18 = Mizoram	Aizawl	108
18	18 = Mizoram	Aizawl	19 = Tripura	Agartala	91
19	19 = Tripura	Agartala	20 = Andaman & Nicobar Islands	Port Blair	847
20	20 = Andaman & Nicobar Islands	Port Blair	21 = Orissa	Bhubaneswar	751
21	21 = Orissa	Bhubaneswar	22 = Chattisgarh	Raipur	281
22	22 = Chattisgarh	Raipur	23 = Andhra Pradesh	Hyderabad	337
23	23 = Andhra Pradesh	Hyderabad	24 = Tamil Nadu	Chennai	322
24	24 = Tamil Nadu	Chennai	25 = Puducherry	Pondicherry	84
25	25 = Puducherry	Pondicherry	26 = Karnataka	Bangalore	165
26	26 = Karnataka	Bangalore	27 = Kerala	Kochi	229
27	27 = Kerala	Kochi	28 = Lakshwadeep	Agatti	285
28	28 = Lakshwadeep	Agatti	29 = Goa	Dabolim Goa	332
29	29 = Goa	Dabolim Goa	30 = Maharashtra	Mumbai	264
30	30 = Maharashtra	Mumbai	31 = Dadra & Nagar Haveli	Daman	93
31	31 = Dadra & Nagar Haveli	Daman	32 = Daman & Diu	Daman	0
32	32 = Daman & Diu	Daman	33 = Gujarat	Ahmedabad	182
33	33 = Gujarat	Ahmedabad	34 = Madhya Pradesh	Indore	211
34	34 = Madhya Pradesh	Indore	35 = Rajasthan	Jaipur	291

Arcs in the Shortest Hamiltonian Path H in Figure 7 in order of appearance on H

Nagaland, Manipur, Mizoram- total 18 States, 245 constituencies, 373,574 polling stations

- Day 2: Tripura, Andaman & Nicobar Islands, Orissa, Chattisgarh, Andhra Pradesh, Tamil Nadu, Puducherry, Karnataka, Kerala, Lakshwadeep, Goa, Maharashtra, Dadra & Nagar Haveli, Daman & Diu, Gujarat- total 15 States, 244 constituencies,369,616 polling stations
- Day 3: Madhya Pradesh, Rajasthan- total 2 States, 54 constituencies, 90,511 polling stations

Algorithm 2: A 0-1 model for breaking up *H* into segments: For constructing this model we need an upper bound on the minimum number of segments into which H can be partitioned subject to the upper bound w_0 on the weight of each segment. From the results obtained from the solution given by Algorithm 1, we know that $k = 3$ is an upper bound that we need. We will describe this model denoting this upper bound by k (in our problem $k = 3$). Define variables

$y_{i,j}$= 1 if State i belongs to segment j , for $i = 1$ to 35, $j = 1$ to k;

0, otherwise.

For each *j*, we need to form constraints to guarantee that the set of all states *i* corresponding to $y_{i,j}$ = 1 form a segment. For this, for each *j* and each *i* = 1 to 33 we must guarantee that:

$y_{i,j}$ = $y_{i+2,j}$ = 1 implies that $y_{i+1,j}$ = 1 also.

This requires that if $y_{i,j}$ + $y_{i+2,j}$ = 2 then $y_{i+1,j}$ = 1;

and if $y_{i,j}$ + $y_{i+2,j}$ = 1 then $y_{i+1,j}$ =0 or 1.

So, for the pair ($y_{i,j}$ + $y_{i+2,j}$; $y_{i+1,j}$) all the values in the set {(0, 0), (0, 1), (1, 0), (1, 1), (2, 1)} are possible, but (2, 0) is not.

Plotting these points on the 2-dimensional Cartesian plane with $y_{i,j}$ + $y_{i+2,j}$ on the horizontal axis and $y_{i+1,j}$ on the vertical axis; we will now see see that $y_{i,j}$ + $y_{i+2,j}$, and $y_{i+1,j}$ must satisfy the constraint $y_{i,j}$ + $y_{i+2,j}$ - $y_{i+1,j}$ = 1. To see this clearly, denote:

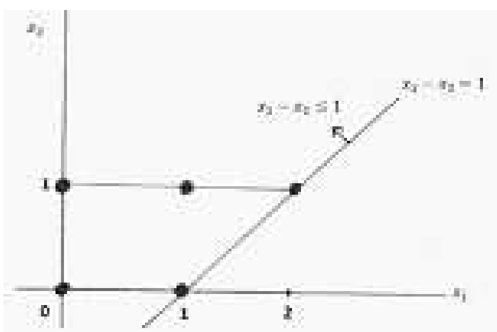

Figure 9: Plot of feasible values for the integer variables x_1 (= $y_{i,j}$ + $y_{i+2,j}$) and x_2 (= $y_{i+1,j}$). All feasible values {(x_1, x_2): (x_1, x_2) = (0,0), (0,1), (1, 0), (1,1), (2,1)} are plotted with a "•", and all of them satisfy x_1 - x_2 <= 1.

$y_{i,j}$ + $y_{i+2,j}$ by x_1 , and
$y_{i+1,j}$ by x_2

and by plotting feasible values of these variables on the x_1, x_2-Cartesian plane we get the following Figure 9.

From this we see that the system of constraints that the variables $y_{i,j}$ have to satisfy are:

$$\sum_{j=1}^{k} y_{i,j} \leq 1 \text{ for each } i = 1 \text{ to } 35,$$

$$\sum_{i=1}^{35} w_i y_{i,j} \leq w_0 \text{ for each } j = 1 \text{ to } k,$$

$$y_{i,j} + y_{i+2,j} - y_{i+1,j} \leq 1 \text{ for all } i = 1 \text{ to } 33, \text{ and } j = 1 \text{ to } k,$$

$$y_{1,1} = 1,$$

$$\sum_{j=1}^{k} j y_{i-1,j} \;\leq\; \sum_{j=1}^{k} j y_{i,j} \;\leq 1 + \sum_{j=1}^{k} j y_{i+1,j} \quad \text{for all } i = 2 \text{ to } 35.$$

$$\text{all } y_{i,j} = 0 \text{ or } 1 \text{ for all } i = 1 \text{ to } 35, \text{ and } j = 1 \text{ to } k.$$

Here is an explanation for these constraints. From the definition of the binary variables y_{ij} given above, the first constraint here guarantees that each of the states i = 1 to 35 is contained in exactly one segment. The second constraint guarantees that the weight of each segment obtained is = the maximum possible weight w_0 determined above in (2). It is discussed above that the third constraint guarantees that for each *j* the set of all the states *i* corresponding to y_{ij} = 1 form a segment. The fourth constraint says that Segment 1 begins with State 1 where the elections begin in the country on the first day of polling. The fifth constraint says that for each *i* = 2 to 35, either State *i* belongs to the same segment as State *i* - 1, or the next segment following it.The last constraint says that all the variables are binary variables. Clearly these are all the constraints in the problem of breaking up H into segments.
From the numbering of the states (defined above in Table 9), this guarantees that each segment is a portion of the Hamiltonian path H between two nodes, and that segments are numbered serially in the order in which they appear along H.
Now the actual number of segments into which H is partitioned in the solution obtained, is the serial number of the segment containing the last State 35 on H; so this is $\sum_{j=1}^{k} j y_{35,j}$, which we have to minimize. So the binary integer programming model for this tour segmentation problem is to minimize $\sum_{j=1}^{k} j y_{35,j}$ subject to the above constraints.
We solved this 0-1 integer programming problem and obtained an optimum solution, consisting of 3 segments. According to it, the elections will be held by the following schedule:

- Day 1: NCT-Delhi, Haryana, Punjab, Jammu &

Kashmir, Himachal Pradesh, Chandigarh, Uttarakhand, Uttar Pradesh, Bihar, Sikkim, Jharkhand, West Bengal, Meghalaya, - total 13 States, 225 constituencies, 347,776 polling stations

- Day 2: Assam, Arunachal Pradesh, Nagaland, Manipur, Mizoram, Tripura, Andaman & Nicobar Islands, Orissa, Chattisgarh, Andhra Pradesh, Tamil Nadu - total 11 States, 136 constituencies, 200,672 polling stations
- Day 3: Puducherry, Karnataka, Kerala, Lakshwadeep, Goa, Maharashtra, Dadra & Nagar Haveli, Daman & Diu, Gujarat, Madhya Pradesh, Rajasthan- total 11 States, 182 constituencies, 285,253 polling stations.

Like the Heuristic Algorithm 1, the solution obtained by Algorithm 2 also partitions the shortest Hamiltonian path H into 3 segments, which is the minimum possible. Now we will take the best among the solutions obtained from both the methods as the one to implement. Even though both correspond to the same value 3 for the number of segments into which H is partitioned; the one obtained by Algorithm2 is better than that obtained by Algorithm1, in other respects. For example, in the solution obtained by Algorithm 2, the number of constituencies and polling stations are more evenly distributed across the segments. Hence we will implement the solution obtained by Algorithm2.

In the sequel k denotes the number of segments selected, which is 3.

3.1.2. *Stage 2: Moving polling personnel from one state to the next during the elections*

After completing the elections in the states in segment j on day j, the batches of polling personnel have to be moved to the states in segment $j+1$ for $j = 1$ to $k-1$.

Construct a bipartite network F with states in segment j as the source nodes, and states in segment $j+1$ as sink nodes. Join each source node p to each sink node q by a directed arc (p, q) with cost coefficient $c^2{}_{p,q}$ = plane-fare between states p, q using the appropriate airports in each state to minimize the cost for this move (at this stage with information available we could use different airports than those used in Stage 1 to make the total cost for this stage small. Also at this stage, notice that the airports used by different batches of polling personnel may be different in each state. Make **availability** (quantity to be shipped to the set of sink nodes) at each source node = the number of batches of polling personnel in the corresponding state on day j, and the **requirement** at each sink node = the number of batches of polling personnel required to hold elections at the corresponding state on day $j+1$. An optimum solution of this bipartite minimum cost flow problem (a transportation problem) gives the moves to be made for this transitions of the polling personnel from day j to day $j+1$ for each $j = 1$ to $k-1$.

Moving CPF polling personnel from their bases to states in segment 1 on Day 1, and from states in the last segment back to their bases at the end of elections: This problem can also be modeled as a bipartite minimum cost flow problem exactly as above, and solved.

4. Results

Solving the model, we obtain a solution with a cost of about 25 billion (Rupees). The model takes about 21 seconds for processing and solution using IBM ILOG CPLEX 12.1.0 on a 1.6 GHz computer.

The optimal movement of Central Police Force personnel from Agartala base is given in Table 10 as an illustration. Here “number moved” is the number of police moving from a base/constituency to another constituency in the next segment.

“Number required” is the number of police personnel required at that constituency (given by 4 times the number of polling stations in that constituency).

Hence there are 2 situations occurring:

- number moved exactly equals the number required which means that this set of movements is sufficient to meet its needs. For example, in base to segment 1, movement from Agartala Base to West Bengal-Dum Dum (5992/5992 which means that 5992 personnel are moved and positioned at that constituency while 5992 personnel are required at that constituency as per norms).

- number moved is less than number required, which means that there are policemen coming in from other places to meet the requirement in this constituency. For example, in segment 2 to segment 3 movements, Karnataka- Chikkballapur requires 7292 police which is met by 1788 from Tamil Nadu-Arani and 5504 from Tamil Nadu-Viluppuram.

5. Discussion

The method demonstrated in Section 4, enables (a) scheduling of elections within the minimum number of segments (b) sequencing the segments, such that the

Table 10

To West Bengal State Constituencies: Number moved/ Total Number required	1.Barrackpore (3164/5308) 2.Dum Dum(5992/5992) 3.Barasat(6128/6128) 4.Joynagar(5700/5700) 5.Jadavpur(6516/6516) 6.Kolkata Dakshin(7452/7452) 7.Kolkata Uttar(6584/6584) 8.Howrah(6600/6600) 9.Jhargram(7016/7016) 10.Purulia(6336/6336) 11.Bankura(6752/6752) 12.Bishnupur(6488/6488) 13.Burdwan Durgapur(6752/6752) 14.Asansol(6252/6252)
To Jharkhand State Constituencies: Number moved/ Total Number required	1.Dumka(396/6572) 2.Jamshedpur(6504/6504) 3.Singhbhum(5368/5368)

Movement of Agartala based Central Police Force personnel from base to Segment 1

movement of Central Police Forces (measured in men-miles) is minimized and (c) sourcing the appropriate number personnel from the most convenient bases.

The method assumes that there is only one set of movements from the bases, which is from the bases to constituencies where the first segments of elections are being held. Similarly there is only one set of movements from the segment where the final phases of elections are being held to the bases. There are no movements between the bases and any segment, where intermediate phases of elections are in progress.

The method can be modified to incorporate ground realities. For example, the requirement of polling personnel may vary across constituencies, depending on the perceptions of threat to maintenance of law and order. In that case, suitable data will have to be incorporated in both the models.

6. Conclusion

In this paper, a methodology is proposed and demonstrated for obtaining the optimal scheduling and logistics planning of the Indian General Elections. The method can be utilized for scheduling and planning any nation-wide event requiring scarce resources.

Here "n.d." means "no date".

References

[1] Bureau of Police Research & Development. (n.d.). *Data on Police Organizations* Retrieved November 6, 2011, from BPRD(Ministry of Home Affairs, Government of India): `http://www.bprd.nic.in/index3.asp?sslid=547\&\\subsublinkid=204\&lang=1`, 2009.

[2] Election Commission of India. (2009). *Lok Sabha Election 2009-Reinforcing Indian Democracy.* New Delhi: Election Commission of India.

[3] Google. (n.d.). *Google Maps*. Retrieved October 2, 2011, from `http://maps.google.co.in`.

[4] iTouchMap.com. (n.d.). *Latitude and Longitude of a point*. Retrieved October 2, 2011, from `http://itouchmap.com/latlong.html`.

[5] Murty, Katta G., (1992), *Network Pringogramming*, Prentice-Hall, Englewood Cliffs, NJ, `http://ioe.engin.umich.edu/people/fac/books/murty/network\_programming/` also can be seen at this website.

[6] National Crime Records Bureau. (2010). *Crime in India 2009 Statistics.* New Delhi: NCRB (Ministry of Home Affairs, Government of India).

[7] Scharff, M. (n.d.). *Policing Election Day: Vulnerability Mapping in India 2006-2009.* Retrieved November 6, 2011, from `http://www.princeton.edu/successfulsocieties/content/data/policy\_note/PN\_id173/Policy\_Note\_ID173.txt`.

[8] Stefan Harrendorf, M. H. (2010). *International Statistics on Crime and Justice.* Helsinki: European Institute for

Table 11

Movement from West Bengal State Constituencies to Tamil Nadu State Constituencies (Number moved/ Total Number required)	1.Barrackpore to Chennai North(1288/4968) 2.Barrackpore to Chennai South(1876/5576) 3.Dum Dum to Mayiladuthurai(5292/5292) 4.Barasat to Chennai North(1536/4968) 5.Barasat to Chennai Central(4592/4592) 6.Jadavpur to Chidambaram(2944/5612) 7.Kolkata Dakshin to Arani(5548/5548) 8.Kolkata Uttar to Arakkonam(5388/5388) 9.Howrah to Viluppuram (5504/5504) 10.Howrah to Chidambaram(1096/5612)
Movement from West Bengal State Constituencies to Orissa State Constituencies (Number moved/ Total Number required)	1.Dum Dum to Bhadrak(700/6168) 2.Joynagar to Kandhamal(56/5152) 3.Joynagar to Kendrapara(336/6656) 4.Joynagar to Jagatsinghpur(5308/6844) 5.Jadavpur to Dhenkanal(3572/5500) 6.Kolkata Dakshin to Aska(1904/5560) 7.Kolkata Uttar to Jajpur (1196/5596) 8.Jhargram to Balasore(5856/5856) 9.Purulia to Sambalpur (5604/5604) 10.Bankura to Nabarangpur(5828/5828) 11.Bishnupur to Mayurbhanj(6488/6488) 12.Burdwan Durgapur to Keonjhar(332/6584) 13.Burdwan Durgapur to Koraput(1612/6064) 14.Asansol to Keonjhar(6252/6584)
Movement from West Bengal State Constituencies to Andhra Pradesh State Constituencies (Number moved/ Total Number required)	1.Jhargram to Srikakulam(1160/7160) 2.Purulia to Aruku (732/6424) 3.Bankura to Aruku (924/6424) 4.Burdwan Durgapur to Kakinada(4808/5780)
Movement from Jharkhand State Constituencies to Andhra Pradesh State Constituencies (Number moved/ Total Number required)	1.Dumka to Aruku(396/6424) 2.Jamshedpur to Srikakulam(692/7160) 3.Jamshedpur to Kakinada(420/5780) 4.Jamshedpur to Narsapuram(5392/5392) 5.Singhbhum to Anakapalli(5368/6220)

Movement of Agartala based Central Police Force personnel from Segment 1 to Segment 2

Crime Prevention & Control (HEUNI).

[9] The Airport Authority. (n.d.). *Every airport, everywhere*. Retrieved November 8, 2011, from `http://airport-authority.com/`

Received 4-1-2013; accepted 18-2-2013

Table 12

Movement from Andhra Pradesh State Constituencies to Maharashtra State Constituencies (Number moved/ Total Number required)	1.Aruku to Akola(2052/6740) 2.Srikakulam to Buldhana(1852/6680) 3.Kakinada to Palghar(420/7696) 4.Narsapuram to Bhiwandi(5392/7304) 5.Kakinada to Palghar(3868/7696) 6.Kakinada to Bhiwandi(940/7304)
Movement from Andhra Pradesh State Constituencies to Gujarat State Constituencies (Number moved/ Total Number required)	1.Anakapalli to Bhavnagar(4272/6176) 2.Anakapalli to Rajkot(1096/6324)
Movement from Orissa State Constituencies to Maharashtra State Constituencies (Number moved/ Total Number required)	1.Nabarangpur to Yavatmal Washim(5828/7524) 2.Koraput to Amravati(1612/7180) 3.Bhadrak to Ramtek(700/8180)
Movement from Orissa State Constituencies to Madhya Pradesh State Constituencies (Number moved/ Total Number required)	1.Balasore to Rewa(5856/5808) 2.Sambalpur to Rewa (5604/5808) 3.Mayurbhanj to Shahdol(6488/6664) 4.Keonjhar to Rewa (6584/5808) 5.Kandhamal to Rewa (56/5808) 6.Kendrapara to Rewa (336/5808) 7.Jagatsinghpur to Rewa (5308/5808) 8.Dhenkanal to Rewa (3572/5808) 9.Aska to Rewa (1904/5808) 10.Jajpur to Balaghat(1196/7016)
Movement from Tamil Nadu State Constituencies to Karnataka State Constituencies (Number moved/ Total Number required)	1.Chennai North to Bangalore Central(1744/7068) 2.Chennai North to Bangalore South (956/6996) 3.Chennai North to Kolar(124/7560) 4.Chennai South to Bangalore North(1876/7804) 5.Mayiladuthurai to Hassan(4368/6000) 6.Mayiladuthurai to Bangalore North(736/7804) 7.Mayiladuthurai to Shimoga(188/6820) 8.Chennai Central to Davangere(3204/6620) 9.Chennai Central to Tumkur(1232/6844) 10.Chennai Central to Bangalore Rural(156/8644) 11.Chidambaram to Tumkur (4040/6844) 12.Arani to Bangalore Rural(336/8644) 13.Arani to Chikkballapur(1788/7292) 14.Arakkonam to Bagalkot(32/6012) 15.Viluppuram to Chikkballapur (5504/7292)
Movement from Tamil Nadu State Constituencies to Puducherry State Constituencies (Number moved/ Total Number required)	1.Arani to Puducherry(3424/3424)
Movement from Tamil Nadu State Constituencies to Goa State Constituencies (Number moved/ Total Number required)	1.Arakkonam to North Goa(2716/2716) 2.Arakkonam to South Goa(2640/2640)

Movement of Agartala based Central Police Force personnel from Segment 2 to Segment 3