1 The model

Two states of the world: policies x and y. Party A and B prefer x and y respectively. We can make this more explicit with

$$U_A(x) > U_A(y)$$
$$U_B(y) > U_B(x)$$

For voters, $v_i > 0$ means voter i prefers x to y All $i \in N$ vote for x or y. Simple majority rule determines which policy gets implemented. For each voter, each party sets a bribe schedule

$$a \in (a_1, ..., a_n) \in \mathbb{R}^n_+$$
$$b \in (b_1, ..., b_n) \in \mathbb{R}^n_+$$

Solving through backward induction, given bribe schedules (a, b), voter i prefers to vote for x if $a_i + v_i > b_i$ and for y otherwise. Since indifferent voters choose y, party B needs to only match bribes from A, adjusting for individual voters' preferences: $b_i = a_i + v_i$. Therefore, B solves

$$\min_C \left\{ \sum_{i \in C} \max\{0, a_i + v_i\} : |C| > \frac{n}{2} \right\}$$

As long as this sum is strictly less than W_B; otherwise party B chooses to set $b_i = 0 : \forall i \in N$.

Following Banks (2000), we restrict our analysis to the set of equilibria in which party A wins, i.e. W_A is sufficiently large relative to $\mathbf{v}$ and W_B so that policy x prevails over y. In other words, the folloing inequality must hold:

$$\sum_{i \in C} \max\{0, a_i + v_i\} \geq W_B$$

Let $U(v, W_b) \subseteq \mathbb{R}^n_+$ denote the set of unbeatable bribe schedules. Additionally, let $S(a) = \sum_i^n a_i$ denote the bribe schedule for party A. The above assumptions on W_A, W_B and v guarantee that there is an

$$\tilde{a} \in U(\mathbf{v}, W_B) : S(\tilde{a}) \leq W_A$$

For party A, the solution is

$$\min_a \{S(a) : a \in U(\mathbf{v}, W_B)\} \tag{1}$$

To fully describe the solution to equation 1, we note the following: for any $a \in \mathbb{R}^n_+$, let $C(a) : i \in N : a_i > 0$ denote the set of individuals who receive a bribe from A. One can show that there is a bribe schedule a' such that for any $i, j \in C(a)$, $a'_i + v_i = a'_j + v_j$. The intuition is that A has no incentive to make voters differentially bribed, because B will simply ignore the more expensive voters and target the weakest rings in the chain. Following Groseclose and Snyder (1996) we refer to this as a leveling schedule.

Let $U^l(\mathbf{v}, W_B) \subseteq U(\mathbf{v}, W_B)$ denote the set of unbeatable leveling schedules. These are bribe schedules such that $a_i + v_i = a_j + v_j \equiv t(a)$. The bribe $a_i = t(a) - v_i$ is the sum of two terms. The first is the common "transfer" among all voters in $C(a)$, the second $(-v_i)$ is individual specific. The latter term makes voters indifferent between x and y absent any bribe from B; the former represents the per capita amount necessary to make $C(a)$, together with any unbribed voters, unaffordable for B.

To further simplify the analysis, Banks introduces the folowing sets of assumption:

$$A_1 : v_{(n+1)/2} < 0$$
$$A_2 : v_1 < 2W_B/(n+1)$$

A_1 implies that absent any bribes by A, y will defeat x. Therefore A must bribe at least one voter. A_2 further implies that A must bribe at least a majority of voters, otherwise B will have sufficient resources to bribe $(n+1)/2$ voters and win.

Banks then proceeds to show that there are monotonic bribing schedules contained within the solution for equation 1. For any $a \in \mathbb{R}^n_+$ let $k(a) = |C(a)|$. Suppose that $a \in U^l(\mathbf{v}, W_B)$ is such that $v_i \geq v_j$ and $j \in C(a)$ but $i \notin C(a)$. Then, under A_2, there exists $a' \in U^l(\mathbf{v}, W_B)$ with $S(a') \leq S(a)$, $k(a') = k(a)$ and $i \in C(a')$ but $j \notin C(a')$ by simply swapping i for j.[1]

Generalizing, and recalling that $v_1 \geq ... \geq v_n$, we see that for all $a \in U^l(\mathbf{v}, W_b)$ there exists a bribe schedule $a' \in U^l(\mathbf{v}, W_b)$ such that $S(a') \leq S(a)$ and $C(a') = \{1, ..., k(a)\}$. Therefore, we can without loss of generality restrict attention to schedules a by A which bribe the first $k(a)$ voters. Call these monotonic leveling schedules and let $U^l_m \subseteq U(\mathbf{v}, W_B)$.

Therefore, when A_2 holds,

$$\min\{S(a) : a \in U(\mathbf{v}, W_B)\} = \min\{S(a) : a \in U^l_m(\mathbf{v}, W_B)\}$$

We can now further simplify the total expenditure $S(a)$

$$S(a) = \sum_{i \in C(a)} a_i = k(a) \cdot t(a) - \sum_{i \leq k(a)} v_i$$

[1]Note that since $v_i \geq v_j$, we have that $t(a) - v_i \leq t(a) - v_j$, i.e. $a'_i \leq a_j$. A_2 guarantees that a'_i and a_j are non-negative.

Note that the choice of $k(a)$ and $t(a)$ fully characterize any schedule $a \in U^l_m(\mathbf{v}, W_B)$. We can thus fully characterize the optimization problem of A as

$$\min_{k,t} k \cdot t - \sum_{i \leq k} v_i$$

subject to the constraint that the induced schedule $a \in U^l_m$. Banks then reformulates this as an unconstrained problem by noting the following. First, if $a(k, t, \mathbf{v})$ is unbeatable, it must be that $k \geq (n+1)/2$, so by A_1 it must be that if $a_i(k, t, \mathbf{v}) = 0$, then $v_i < 0$. Therefore, B receives all non-bribed voters for free. For $a(k, t, \mathbf{v})$ to be unbeatable, then, it must be that B cannot afford the remaining $(n+1)/2 - (n-k) = k - (n-1)/2$ voters, or

$$t * (k - (n-1)/2) \geq W_B$$

Solving this for equality yields the optimal transfer from A to members of $C(A) = \{1, ..., k\}$, conditional on k:

$$t(k, W_B) = \frac{W_B}{k - (n-1)/2} \tag{2}$$

Defining minimal winning expenditures as

$$E(k, \mathbf{v}, W_B) = k \cdot t(k, W_B) - \sum_{i \leq k} v_i \tag{3}$$

we can state A's problem as

$$\min_k \{E(k, \mathbf{v}, W_B) : k \in (n + 1/2), ..., n\} \tag{4}$$

Denote the solution to expression 4 as $k^*(\mathbf{v}, W_B)$. This solution genenerates a solution to expression 1

2 Results

First, characterize a solution for k^*. Because k is finite, calculus cannot be employed. Instead, we deploy a discrete version of these techniques. First let's define $\Delta(k) = E(k+1) - E(k)$ as the difference in expenditure from adding another coalition member. Note that if $\Delta(k) \geq 0$ then A does not want to add another member to the coalition.

Conversely, if $\Delta(k) < 0$, then A is strictly better off by adding the $k+1$th member of the coalition.

Next, suppose that $\Delta(k)$ is increasing in k. The following algorithm can then be used to identify k^*: if $\Delta((n+1)/2) \geq 0$, then we know from]$\Delta(k)$ increasing that A is better off by setting k^* to $(n+1)/2$. If $\Delta((n+1)/2) < 0$, then we know that k^* must be greater than $(n+1)/2$, so we next solve for $\Delta((n+3)/2)$, and so on.

We can therefore search for the optimal k^* with the following algorithm:

$$k^* = \begin{cases} (n+1)/2 & \text{if } \Delta((n+1)/2) \geq 0 \\ \max\{k : \Delta(k-1) < 0\} & \text{otherwise} \end{cases} \tag{5}$$

Banks also shows that $\Delta(k)$ is indeed nondecreasing.

$$\Delta(k) = \left[\frac{(k+1)W_B}{k+1-(n-1)/2} - \sum_{i \leq k+1} v_i\right] \tag{6}$$

$$= \frac{-W_B(n-1)}{2(k+1-(n-1)/2)(k-(n-1)/2)} - v_{k+1} \tag{7}$$

$$\equiv T(k, W_B) - v_{k+1} \tag{8}$$