Web-based Supplementary Materials for
Optimal matching with minimal deviation from fine balance in a study of obesity and surgical outcomes

Dan Yang*, Dylan S. Small, Jeffrey H. Silber, and Paul R. Rosenbaum
Department of Statistics, The Wharton School, University of Pennsylvania,
400 Jon M. Huntsman Hall, 3730 Walnut Street, Philadelphia, PA 19104-6340 USA

*email: danyang@wharton.upenn.edu

Summary: This web-based supplement presents a minimum cost flow algorithm for near-fine matching. In currently available statistical software, this approach requires specialized programming, but it can be more efficient in its use of computer memory than the optimal assignment algorithm. The first author’s package finebalance in R uses this minimum cost flow algorithm.

In this section, an alternative but equivalent solution to problem in §3.3 is described in terms of minimum cost flow in a network. The description is brief because the method is a modification of the method in Rosenbaum (1989, §3.2) and is, in certain respects, equivalent to the assignment algorithm in §3.3. On the negative side, this network solution requires somewhat specialized terminology and software. On the positive side, the network solution uses the $T \times C$ matrix $\Delta$ directly, rather than the larger $K \times K$ matrix $\Upsilon$, so the network solution may require less computer memory. The first author has implemented the network solution called finebalance in R by adapting Bertsekas’ (1991) Fortran code as made available by way of Hansen’s (2007) optmatch package in R.

A network consists of a finite collection of nodes, $\mathcal{N}$, and a collection, $\mathcal{E}$, of directed edges comprised of ordered pairs of nodes, $e \in \mathcal{E}$ entails $e = (n, n')$ with $n, n' \in \mathcal{N}$; therefore, $\mathcal{E}$ is a subset of the direct product of $\mathcal{N}$ with itself, that is, $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$. Think for instance of a railroad network transporting boxes with stations $\mathcal{N}$ and tracks $\mathcal{E}$ connecting some pairs of stations. There are two special nodes, the source, $s \in \mathcal{N}$, and the sink, $\bar{s} \in \mathcal{N}$, such that no edge enters the source or leaves the sink, that is, $(n, s) \notin \mathcal{E}$ and $(\bar{s}, n) \notin \mathcal{E}$ for all $n \in \mathcal{N}$; metaphorically, the transport is from the source to the sink. An integral flow is a function $f : \mathcal{E} \to \{0, 1, 2, \ldots\}$ such that

$$\sum_{e = (n', n) \in \mathcal{E}} f(e) = \sum_{e = (n, n') \in \mathcal{E}} f(e) \text{ for each } n \in \mathcal{N} - \{s, \bar{s}\};$$

that is, aside from the source and the sink, the total flow into $n$ equals the total flow out from $n$ for each node $n$. Each edge $e \in \mathcal{E}$ has a nonnegative integer lower and an upper capacity, $0 \leq q_e \leq Q_e \leq \infty$. Each edge also has a cost-per-unit of flow, $h_e$. The minimum cost flow problem with specified value $F$ is to find an integral flow $f$ satisfying (1) that solves

$$\text{minimize } \sum_{e \in \mathcal{E}} h_e f(e)$$
subject to

\[ F = \sum_{e=(s,n) \in E} f(e), \]

\[ q_e \leq f(e) \leq Q_e \quad \text{for } e \in E. \]

Metaphorically, the problem is to minimize the cost of transporting \( F \) boxes intact from the source \( s \) to the sink \( \bar{s} \) respecting the constraints on the capacities of the edges; here, integer flows are required because the solution cannot cut a box in half.

To solve the problem in §3.3 using minimum cost flow in a network, set up the network as follows. The nodes are \( N = \{s, \bar{s}\} \cup T \cup C \cup \{1, 2, \ldots, J\} \). There is an edge from the source to each treated unit \( e = (s, \tau_t) \in E \) for each \( \tau_t \in T \) with capacity \( q_e = 0, Q_e = 1 \) and cost \( h_e = 0 \). There is an edge from each treated unit to each control \( e = (\tau_t, \gamma_c) \in E \) for each \( \tau_t \in T \) and each \( \gamma_c \in C \) with capacity \( q_e = 0, Q_e = 1 \) and cost \( h_e = \delta_{\tau_t, \gamma_c} \). There is an edge from each control \( \gamma_c \in C \) to the nominal category to which that control belongs, \( e = (\gamma_c, j) \in E \) if and only if \( \nu(\gamma_c) = j \) with capacity \( q_e = 0, Q_e = 1 \) and cost \( h_e = 0 \). Finally, there is edge \( e = (j, \bar{s}) \in E \) from each nominal category \( j \in \{1, 2, \ldots, J\} \) to the sink with capacity \( q_e = \kappa_j, Q_e = \bar{\kappa}_j \) and cost \( h_e = 0 \). Because each \( \gamma_c \in C \) has a single departing edge \( e = (\gamma_c, j) \in E \) with maximum capacity \( Q_e = 1 \), by (1), \( \gamma_c \) can receive at most one unit of flow from (i.e., can be matched to) a single treated unit \( \tau_t \in T \). A minimum cost integral flow in this network with value \( T = \sum_{e=(s,n) \in E} f(e) \) is (i) passes exactly one unit of flow from the source to each treated unit, \( f(s, \tau_t) = 1 \); (ii) passes either one or zero units of flow from each treated unit to each potential control, \( f(\tau_t, \gamma_c) = 0 \) or \( f(\tau_t, \gamma_c) = 1 \), and if the flow is 1 then \( \tau_t \) and \( \gamma_c \) are matched, \( \mu(\tau_t) = \gamma_c \); (iii) for each potential control \( \gamma_c \in C \), if there is a treated unit \( \tau_t \in T \) with \( f(\tau_t, \gamma_c) = 1 \) then, by (1), one unit of flow passes from \( \gamma_c \) to \( j = \nu(\gamma_c) \), that is \( f(\gamma_c, \nu(\gamma_c)) = 1 \), and otherwise \( f(\gamma_c, \nu(\gamma_c)) = 0 \); (iv) the integer flow from \( j \) to the sink satisfies \( \kappa_j \leq f(\gamma_c, j) \leq \bar{\kappa}_j \); (v) among all such integral flows, \( f(\cdot) \) minimizes the total cost \( \sum_{e \in E} h_e f(e) \) which equals the total distance within matched pairs,
\[ \sum_{\tau_t \in \mathcal{T}} \delta_{\tau_{\mu(\tau)}}. \] In brief, the minimum cost integral flow in this network solves the problem in \S3.3.

Minor adjustments permit matching each treated unit to the same number \( L \geq 1 \) of controls. In particular, set \( F = LT \) so \( L \) units of flow leave the source for each treated unit, set \( Q_e = L \) for \( e = (s, \tau_t) \in \mathcal{E} \) for each \( \tau_t \in \mathcal{T} \), and adjust \( q_e = \kappa_j, Q_e = \pi_j \) for each edge \( e = (j, s) \in \mathcal{E} \) to specify the permitted degree of departure from fine balance. For instance, exact fine balance has \( \kappa_j = \pi_j = Ln_j \) whereas a deviation of at most one has \((\kappa_j, \pi_j) = (Ln_j - 1, Ln_j + 1)\).