

Resident doctor allocation

William Pettersson

What are resident doctors, and why
do we allocate them?

Why did resident doctors strike and what are they paid?



14 July 2025

Updated 30 July 2025

Resident doctors in England are heading back to work after a five-day walkout over pay.

They say they haven't had a "credible pay deal" for 2025-2026, but Health Secretary Wes Streeting argues the strike is "unreasonable" after substantial pay rises in recent years.

Figure 1: From <https://www.bbc.co.uk/news/articles/cn0qyl4ljko>

'The NHS can't tell me where my job will be'



Jayne Evans is one of hundreds of "placeholder" resident doctors who do not know where they will be based

Charlie Haynes

BBC Investigations, East Midlands

7 April 2025 · 431 Comments

Jayne Evans has completed four years at medical school in London - but says she is still being left in the dark about where her first permanent NHS position will be.

Figure 2: From <https://www.bbc.co.uk/news/articles/c045l5r467ko>



A



B



C



D



W



X



Y



Z

$$d_a : h_w h_x h_y h_z$$

$$d_b : h_w h_z h_x h_y$$

$$d_c : h_w h_y h_x h_z$$

$$d_d : h_w h_x h_y h_z$$

$$h_w : d_b d_a d_c d_d$$

$$h_x : d_b d_c d_a d_d$$

$$h_y : d_a d_c d_b d_d$$

$$h_z : d_a d_d d_c d_b$$

$$d_a : h_w h_x h_y h_z$$

$$d_b : h_w h_z h_x h_y$$

$$d_c : h_w h_y h_x h_z$$

$$d_d : h_w h_x h_y h_z$$

$$h_w(1) : d_b d_a d_c d_d$$

$$h_x(2) : d_b d_c d_a d_d$$

$$h_y(1) : d_a d_c d_b d_d$$

$$h_z(2) : d_a d_d d_c d_b$$

$$d_a : \underline{h_w} h_x h_y h_z$$

$$d_b : h_w h_z h_x h_y$$

$$d_c : h_w h_y h_x h_z$$

$$d_d : h_w h_x h_y h_z$$

$$h_w(1) : d_b \underline{d_a} d_c d_d$$

$$h_x(2) : d_b d_c d_a d_d$$

$$h_y(1) : d_a d_c d_b d_d$$

$$h_z(2) : d_a d_d d_c d_b$$

$$d_a : \underline{h_w} h_x h_y h_z$$

$$d_b : h_w \underline{h_z} h_x h_y$$

$$d_c : h_w h_y h_x h_z$$

$$d_d : h_w h_x h_y h_z$$

$$h_w(1) : d_b \underline{d_a} d_c d_d$$

$$h_x(2) : d_b d_c d_a d_d$$

$$h_y(1) : d_a d_c d_b d_d$$

$$h_z(2) : d_a d_d d_c \underline{d_b}$$

$$d_a : h_{\underline{w}} h_x h_y h_z$$

$$d_b : h_w h_{\underline{z}} h_x h_y$$

$$d_c : h_w h_{\underline{y}} h_x h_z$$

$$d_d : h_w h_{\underline{x}} h_y h_z$$

$$h_w(1) : d_b d_{\underline{a}} d_c d_d$$

$$h_x(2) : d_b d_c d_a d_{\underline{d}}$$

$$h_y(1) : d_a d_{\underline{c}} d_b d_d$$

$$h_z(2) : d_a d_d d_c d_{\underline{b}}$$

$$d_a : \underline{h_w} h_x h_y h_z$$

$$d_b : \textcolor{red}{h_w} \underline{h_z} h_x h_y$$

$$d_c : h_w \underline{h_y} h_x h_z$$

$$d_d : h_w \underline{h_x} h_y h_z$$

$$h_w(1) : \textcolor{red}{d_b} \underline{d_a} d_c d_d$$

$$h_x(2) : d_b d_c d_a \underline{d_d}$$

$$h_y(1) : d_a \underline{d_c} d_b d_d$$

$$h_z(2) : d_a d_d d_c \underline{d_b}$$

Blocking pair

Given a matching problem, and a selected matching, a pair d, h is a blocking pair if:

- d is unmatched, or prefers h to their currently matched hospital, and
- h is under capacity, or prefers d to at least one of their currently matched doctors.

Stable matching

Given a matching problem, a matching is stable if there are no blocking pairs.

A stable matching can be found in polynomial time using e.g. the Gale-Shapley algorithm.

$$d_a : h_w \underline{h_x} h_y h_z$$

$$d_b : \underline{h_w} h_z h_x h_y$$

$$d_c : h_w \underline{h_y} h_x h_z$$

$$d_d : h_w \underline{h_x} h_y h_z$$

$$h_w(1) : \underline{d_b} d_a d_c d_d$$

$$h_x(2) : d_b d_c \underline{d_a} d_d$$

$$h_y(1) : d_a \underline{d_c} d_b d_d$$

$$h_z(2) : d_a d_d d_c d_b$$

There can be different stable matchings, however:

Rural hospitals theorem

- The set of assigned doctors is always the same (i.e., if a doctor is not assigned in one stable matching, they're not assigned in any stable matching)
- If a hospital has unassigned space, then in any stable matching said hospital is always assigned the exact same set of doctors

- Some doctors are “equally good”, and doctors may rank hospitals as “equally good” (also known as ties)
- Doctors also aren’t required to rank every single hospital

$$d_a : h_w[h_x h_y]h_z$$

In this preference list, h_w is the rank-1 choice of d_a , h_x and h_y are both rank-2 choices of d_a , and h_z is the rank-3 choice of d_1 .

Note that the rural hospitals theorem doesn’t hold if ties are present.

As a result of having ties and “incomplete lists”, different stable matchings may have different sizes, and people generally want a largest stable matching.

Finding such a matching is NP-hard.

Integer linear programming

$$\begin{aligned}
& \max \sum_{i=1}^{n_d} \sum_{j \in H(i)} x_{ij} \\
& \text{s.t.} \quad \sum_{j \in H(i)} x_{ij} \leq 1, & i \in \{1, \dots, n_d\}, \\
& \quad \sum_{i \in D(j)} x_{ij} \leq c_j, & j \in \{1, \dots, n_h\}, \\
& \quad c_j \left(1 - \sum_{q \in H_j^{\leq}(i)} x_{iq} \right) \leq \sum_{p \in D_i^{\leq}(j)} x_{pj}, & i \in \{1, \dots, n_d\}, j \in H(i), \\
& \quad x_{ij} \in \{0, 1\} & i \in \{1, \dots, n_d\}, j \in H(i)
\end{aligned}$$

$$\begin{aligned}
& \max \sum_{i=1}^{\mathbf{n}_d} \sum_{j \in H(i)} x_{ij} \\
& \text{s.t.} \quad \sum_{j \in H(i)} x_{ij} \leq 1, & i \in \{1, \dots, \mathbf{n}_d\}, \\
& \quad \sum_{i \in D(j)} x_{ij} \leq c_j, & j \in \{1, \dots, \mathbf{n}_h\}, \\
& \quad c_j \left(1 - \sum_{q \in H_j^{\leq}(i)} x_{iq} \right) \leq \sum_{p \in D_i^{\leq}(j)} x_{pj}, & i \in \{1, \dots, \mathbf{n}_d\}, j \in H(i), \\
& \quad x_{ij} \in \{0, 1\} & i \in \{1, \dots, \mathbf{n}_d\}, j \in H(i)
\end{aligned}$$

$$\begin{aligned}
& \max \sum_{i=1}^{n_d} \sum_{j \in \textcolor{red}{H}(i)} x_{ij} \\
& \text{s.t.} \quad \sum_{j \in \textcolor{red}{H}(i)} x_{ij} \leq 1, & i \in \{1, \dots, n_d\}, \\
& \quad \sum_{i \in \textcolor{red}{D}(j)} x_{ij} \leq \textcolor{red}{c}_j, & j \in \{1, \dots, n_h\}, \\
& \quad c_j \left(1 - \sum_{q \in H_j^{\leq}(i)} x_{iq} \right) \leq \sum_{p \in D_i^{\leq}(j)} x_{pj}, & i \in \{1, \dots, n_d\}, j \in \textcolor{red}{H}(i), \\
& \quad x_{ij} \in \{0, 1\} & i \in \{1, \dots, n_d\}, j \in \textcolor{red}{H}(i)
\end{aligned}$$

$$\begin{aligned}
& \max \sum_{i=1}^{n_d} \sum_{j \in H(i)} x_{ij} \\
& \text{s.t.} \quad \sum_{j \in H(i)} x_{ij} \leq 1, & i \in \{1, \dots, n_d\}, \\
& \quad \sum_{i \in D(j)} x_{ij} \leq c_j, & j \in \{1, \dots, n_h\}, \\
& \quad c_j \left(1 - \sum_{q \in \mathbf{H}_{\mathbf{j}}^{\leq}(i)} x_{iq} \right) \leq \sum_{p \in \mathbf{D}_{\mathbf{i}}^{\leq}(j)} x_{pj}, & i \in \{1, \dots, n_d\}, j \in H(i), \\
& \quad x_{ij} \in \{0, 1\} & i \in \{1, \dots, n_d\}, j \in H(i)
\end{aligned}$$

$$\begin{aligned}
& \max \sum_{i=1}^{n_d} \sum_{j \in H(i)} x_{ij} \\
& \text{s.t.} \quad \sum_{j \in H(i)} x_{ij} \leq 1, & i \in \{1, \dots, n_d\}, \\
& \quad \sum_{i \in D(j)} x_{ij} \leq c_j, & j \in \{1, \dots, n_h\}, \\
& \quad c_j \left(1 - \sum_{q \in H_j^{\leq}(i)} x_{iq} \right) \leq \sum_{p \in D_i^{\leq}(j)} x_{pj}, & i \in \{1, \dots, n_d\}, j \in H(i), \\
& \quad x_{ij} \in \{0, 1\} & i \in \{1, \dots, n_d\}, j \in H(i)
\end{aligned}$$

Assessing, and improving, ILP models

We now talk about ways of assessing and improving ILP models. Improving is specifically aimed at “finding a solution faster” — the model shown already finds an optimal solution, there cannot be a “more optimal” solution.

Recall that, if every hospital ranks every doctor, and vice-versa, we will have $n_d \times n_h$ constraints for stability.

$$d_a : h_w[h_x h_y] h_z$$

1. Let w_{ir}^d be 1 if and only if doctor i is assigned to a hospital of rank at most r .
2. Let w_{jr}^h be 1 if and only if hospital j is assigned to a doctor of rank at most r .

Then we can model stability as

$$c_j \left(1 - w_{i, r_j^d(i)}^d \right) \leq w_{j, r_i^h(j)}^h.$$

Now if every doctor has R_d ranks and each hospital has R_h ranks, we have $R_d \times R_h$ stability constraints. If the doctors and hospitals have many ties, and thus few ranks, then this will mean fewer stability constraints. This can result in faster solve times.

The real world is hard

Some resident doctors are in relationships such that two different resident doctors are considered a couple.

The couple cannot be assigned to two arbitrary hospitals, instead the couple (d_1, d_2) submits a list of preference pairs $(h_1, h_1), (h_1, h_2), (h_1, \emptyset)$, where this last entry (h_1, \emptyset) means d_1 is assigned to h_1 and d_2 is unassigned.

With couples, it is possible that there will be no stable matchings at all.

Stability is based on blocking pairs, but do hospitals know about couples?

$$h_1(2) : \underline{d_1}, d_2, \underline{d_3}$$

Would h_1, d_2 be a blocking pair?

Stability is based on blocking pairs, but do hospitals know about couples?

$$h_1(2) : \underline{d_1}, d_2, \underline{d_3}$$

Would h_1, d_2 be a blocking pair? What if (d_1, d_3) are a couple?

Stability is based on blocking pairs, but do hospitals know about couples?

$$h_1(2) : \underline{d_1}, d_2, \underline{d_3}$$

Would h_1, d_2 be a blocking pair? What if (d_1, d_3) are a couple?

$$h_1(2) : d_1, \underline{\underline{d_2}}, d_3$$

Would h_1, d_1 be a blocking pair?

Stability is based on blocking pairs, but do hospitals know about couples?

$$h_1(2) : \underline{d_1}, d_2, \underline{d_3}$$

Would h_1, d_2 be a blocking pair? What if (d_1, d_3) are a couple?

$$h_1(2) : d_1, \underline{\underline{d_2}}, d_3$$

Would h_1, d_1 be a blocking pair? What if (d_1, d_3) are a couple?

$$h_1(2) : d_1, \underline{d_2}, \underline{d_3}, d_4$$

Now (d_1, d_4) are a couple and (d_2, d_3) are a couple. Are there any blocking pairs? What if (d_1, d_4) are assigned instead of (d_2, d_3) ?

- What if there are different capacities for domestic and international resident doctors?

- What if there are different capacities for domestic and international resident doctors?
- What if a hospital must be closed if there are less than k assigned resident doctors?

- What if there are different capacities for domestic and international resident doctors?
- What if a hospital must be closed if there are less than k assigned resident doctors?
- What if we consider a pair to be blocking if the doctor gets a better placement, and the hospital does not get a worse doctor?

- Resident doctors do need to be allocated to hospital courses for further training
- The doctors and hospital courses each have preferences over each other
- This problem is “easy” if we assume a simplified model
- The problem becomes hard if we allow the complexities of the real world to get involved