The flow network $\mathcal{N}_\lambda$ constructed in Section 4 can further be simplified, in an equivalent way. The new network is denoted by $\mathcal{N}'_\lambda$. Some notations are reused in the definition of the new network. All these reused notations corresponds to those with the same notations in $\mathcal{N}_\lambda$. $E, V, V^E, V^M, V^C, C_i, C, \lambda$ hold the same meanings as previous. $\mathcal{N}'_\lambda$ is constructed as follows.

$\mathcal{N}'_\lambda$ consists of 4 types of vertices s, t, $\{a_{C_i}|C_i \in C\}$ and $\{b'_i|v_i \in V^M\}$. The flow starts from s and ends at t. The edges consists of four types as follows.

1. For any $C_i \in C$, there is an edge from s to a_{C_i} with capacity $c'_\lambda(s, a_{C_i}) = |C_i| * \lambda$.
2. For any $C_i \in C$, there is an edge from a_{C_i} to t, with capacity $|C_i| - 1$.
3. For any $v_k v_j \in E$ satisfying $v_k \in C_i$ and $v_j \in V^M$, there is an edge from a_{C_i} to b'_j with capacity 1.
4. For any $v_j \in V^M$, there is an edge from b'_j to t with capacity 1.

Same as before, we conduct parametric flow on the new network. Then, the utility profile is decided. If $v_i \in V^E \cup V^M$, $u_i = 1$; otherwise, if $v_k \in C_i \subset C$, $u_k = f(s, a_{C_i})/|C_i|$. Here, $f(s, a_{C_i})$ denotes the flow from s to a_{C_i} after the parametric flow algorithm. Actually, u_k is the value of λ where the inflow of a_{C_i} stops increasing.

The utility profile yields by the construction above is the same as the utility profile of the original water-filling algorithm.

This theorem is proved by contradiction. First, we assume the utility profile U' generated by the simplified graph is not the same as the utility profile generated by the original graph U. In other words, we assume there exists a vertex $v_i \in C_x$ such that its utility is U (denoted by u_i) is different from its utility in U' (denoted by u'_i). In addition, without loss of generality, we assume that i satisfies $\forall u_j < u_i$, $u'_j = u_j$. In the following proof, we show that u'_i can be neither more nor less than u_i, which leads to a contradiction.

First, we consider the case $u'_i < u_i$. According to the construction of the $\mathcal{N}'_\lambda$, the inflow of a_{C_x} is $|C_x| * u'_i$, which is smaller than $|C_x| * u_i$.

This proof is conducted as follows. We prove that for any feasible final flow assignment[1] of $\mathcal{N}_\lambda$, we can find a flow assignment of ${\mathcal{N}_\lambda}'$ that yields the same utility profile. Conversely, any feasible final flow assignment of ${\mathcal{N}_\lambda}'$ also corresponds to a flow assignment of $\mathcal{N}_\lambda$ with the same utility profile. This theorem is proved by proving the following two statements: (1) the final flow assignment of $\mathcal{N}_\lambda$ corresponds to a state of ${\mathcal{N}_\lambda}'$ and (2) the final flow assignment of ${\mathcal{N}_\lambda}'$ corresponds to a state of $\mathcal{N}_\lambda$. (1) and (2) indicate that there is a one-on-one mapping between $\mathcal{N}_\lambda$ and ${\mathcal{N}_\lambda}'$. A state refers to the network with a flow assignment.

First, (1) is trivial,

1. For any $v_i \in V^E \cup V^M$, remove a_i, b_i and the edges connected to them. The utility of v_i is 1.
2. For each component $C_i \subset V^C$, in our original construction, we have A^{C_i} and B^{C_i} for it. However, we can replace these two parts with a single vertex

[1] The final flow assignment means the flow assignment when the algorithm terminates.

First, remove parts A^{EM} and B^{EM} and set final utility of each vertex in both sets to be 1. Second, for components that are not connected to M, calculate the utility of each vertex directly as follows: if the size of a component is s, then the utility of any vertex in this component is $\frac{s-1}{s}$. Third, s replace nodes of the same component by a single node, and add up all the capacities and flows between this component and outside nodes accordingly. One can show the simplified network is equivalent to the one constructed in Section 4.