Poniamo $B(T) - B(T') = \sum_{e \in C} f(e)\, d_T(e) - \sum_{e \in C} f(e)\, d_{T'}(e) =$

$= f(x)\, d_T(x) + f(y)\, d_T(y) - f(z)\, d_{T'}(z)$

$= (f(x) + f(y))\, d_T(x) - (f(x) + f(y))\, d_{T'}(z))$

$= (f(x) + f(y))\, d_T(x) - (f(x) + f(y))(d_T(x) - 1)$

$= f(x) + f(y)$

Per alberi T e T' vale che $B(T) = B(T') + f(x) + f(y)$

Per induzione sul numero n dei caratteri $c_1, c_2 \dots c_n$.

Per $n = 2$ l'alg Huffman(C, f) produce un albero ottimo.

Supponiamo che $|C| = n$, e consideriamo l'esecuzione su Huffman (C', f), dove $C' = \{c_1, c_2, \dots c_n, c_{n+1}\}$.

Siano x e y due caratteri con frequenze minime in C', la prima cosa che fa l'algoritmo è di estrarre x e y come fratelli e poi richiamare se stesso sull'insieme di n caratteri $C = (C - \{x, y\}) \cup \{z\}$. Per ipotesi induttiva C produce un albero ottimo O. ~~Allora~~ provare che l'albero N ritornato con Huffman su C' è ottimo. ~~Per~~ Per l'albero N ritornato su C' vale $B(N) = B(O) + f(x) + f(y)$.

Se l'albero N non fosse ottimo, allora esisterebbe un altro albero M sugli n+1 caratteri $c_1, c_2, \dots c_n, c_{n+1}$ per cui $B(M) < B(N) = B(O) + f(x) + f(y)$, come già detto esiste sicuramente un albero ottimo T in cui i due caratteri x, y di frequenze minime appaiono nell'albero T in due foglie alla profondità massima, e sono fratelli. Eliminando tali due foglie dall'albero T e rendendo il loro padre z foglia, otteniamo un albero T' per cui vale $B(T) = B(T') + f(x) + f(y)$. Abbiamo che,

$B(T') + f(x) + f(y) = B(T) = B(M) < B(N) = B(O) + f(x) + f(y) \Rightarrow B(T') < B(O)$

assurdo perché per ipotesi l'albero O era ottimo su n caratteri. Ne segue che anche N è ottimo e pertanto Huffman produce sempre alberi ottimi per qualsiasi insieme di caratteri.