3.52 Esercizio. Una variante del problema della partizione prevede di ripartire i numeri $a_1, \ldots, a_n$ in due sottoinsiemi non solo di uguale somma ma anche di uguale cardinalità. Dimostrare che anche la variante è **NP**-completa. (suggerimento: nella variante si può sommare ai numeri una quantità costante senza alterare l'istanza...)

Soluzione. Sia data un'istanza $a_1, \ldots, a_n$ del problema della partizione e sia $b := \sum a_i/2$. Si immagini di aggiungere all'istanza valori $a_{n+j} = 0$, $j := 1, \ldots, n-2$. Il problema non viene alterato dai nuovi dati. Se esiste una soluzione ammissibile, ne esistono anche tutte quelle che si ottengono aggiungendo o togliendo i valori nulli. Essendo disponibili $(n-2)$ valori nulli c'è la garanzia che esista una soluzione con un numero bilanciato di termini. Quindi c'è una soluzione ammissibile al problema originario se e solo se nel problema variato c'è una soluzione ammissibile con insiemi di uguale cardinalità. Con i nuovi dati il problema non corrisponde alla definizione perché alcuni valori sono nulli. Tuttavia nella variante ogni valore può essere modificato di un valore costante senza alterare il problema. Quindi in definitiva, dati $a_1, \ldots, a_n$, e definiti $a'_i := a_i + 1$, $i := 1, \ldots, n$, e $a'_i := 1$, $i := n+1, \ldots, 2n-2$,

$$\exists J \subset \{1, \ldots, n\} : \sum_{i \in J} a_i = b \quad \Longleftrightarrow \quad \exists J \subset \{1, \ldots, 2n-2\} : |J| = n-1 \; , \; \sum_{i \in J} a'_i = b + n - 1$$

Ad esempio sia dato $a = \{5, 7, 3, 6, 3\}$ con soluzione $J = \{1, 2\}$. Si trasforma in $a' = \{6, 8, 4, 7, 4, 1, 1, 1\}$ con soluzione $J = \{1, 2, 6, 7\}$. ∎

Il problema opposto è quello di trasformare la variante nel problema originale in modo da poter risolvere la variante indirettamente usando un algoritmo noto per il problema originale della partizione. Si è detto che ogni valore dell'istanza può essere aumentato della stessa quantità nella variante. Se i valori aumentati sono tali per cui nel problema originale della partizione l'unica soluzione possibile è una con due insiemi di uguale cardinalità, la trasformazione è valida. Sia $J$ tale che $|J| < |N \setminus J|$. Allora

$$\sum_{j \in J} (a_j + c) = |J|\, c + \sum_{j \in J} a_j \leq |J|\, c + \sum_j a_j = |J|\, c + 2\, b$$

$$\sum_{j \notin J} (a_j + c) = |N \setminus J|\, c + \sum_{j \notin J} a_j > |N \setminus J|\, c \geq (|J| + 1)\, c$$

Basta fare in modo che $|J|\, c + 2\, b \leq (|J| + 1)\, c$ e sarà impossibile che $\sum_{j \in J} a_j = \sum_{j \notin J} a_j$. A tal fine è sufficiente scegliere $c \geq 2\, b$. Ad esempio dati i valori $\{2, 6, 1, 4, 2, 1\}$ vi sono due soluzioni per il problema normale della partizione ($J = \{1, 2\}$ e $J = \{1, 4, 5\}$) però se trasformiamo i dati in $\{18, 22, 17, 20, 18, 17\}$ l'unica soluzione è $J = \{1, 4, 5\}$. In questo modo un algoritmo di programmazione dinamica risolve la variante con complessità $O(n\,(b + n\, b)) = O(n^2\, b)$. Volendo risolvere la variante direttamente con la programmazione dinamica si può estendere il concetto di stato introducendo anche il numero di valori scelti (oltre che la loro somma) conducendo ad numero di nodi pari a $n\,(n/2)\, b$ e quindi anche in questo caso ad una complessità $O(n^2\, b)$.