

ALGORITMI DI OTTIMIZZAZIONE DELLE CONNESSIONI DI SISTEMI FOTOVOLTAICI

P. L. Buccheri, R. Candela, P. Romano,

Dipartimento di Ingegneria Elettrica, Elettronica e delle Telecomunicazioni
Università degli Studi di Palermo
Viale delle Scienze, 90128 Palermo

In un campo fotovoltaico, per ottimizzazione delle connessioni tra pannelli solari si intende la connessione nel modo opportuno dei pannelli ombreggiati e non, in maniera tale da avere in uscita la massima potenza ottenibile. Per fare ciò si può utilizzare una rappresentazione matriciale delle interconnessioni le cui righe corrispondono alla connessione in parallelo di pannelli e tutte le righe sono connesse in serie tra loro.

Considerando condizioni d'irraggiamento non uniformi (caso di parziale oscuramento del campo fotovoltaico), se i moduli possono essere connessi in modo tale da ottenere righe con irraggiamento simile, la potenza massima in uscita potrebbe essere ottimale. Questa capacità di riordinamento si può associare alla matrice, che prende il nome di matrice di spostamento. La matrice di spostamento alla fine darà un'unica stringa – serie di pannelli connessi in parallelo, stabilendo una connessione in parallelo tra tutti i pannelli della stessa riga. Al fine di ottenere la soluzione ottimizzata si è implementato un algoritmo di controllo della matrice di spostamento basato sulla selezione della riga, in cui ogni pannello è connesso in parallelo agli altri della stessa riga, sulla base del valore di irraggiamento, in maniera tale da equalizzare gli irraggiamenti delle righe facenti parte della stringa serie finale.

L'algoritmo di equalizzazione dell'irraggiamento è in grado di generare tutte le possibili combinazioni di connessioni tra i pannelli, noti i parametri d'ingresso dei singoli pannelli (coeff. d'irraggiamento). Per fare ciò si considerano delle maschere (*mask*) costituite da simboli binari che rappresentano le righe. Tutte le maschere vengono costruite tenendo conto che se un pannello compare in una riga non può figurare in un'altra. Per esempio, per un sistema a 4 pannelli, se il pannello 4 è l'unico ad essere nella prima riga, allora si devono effettuare tutte le possibili combinazioni tra i pannelli 1 – 2 – 3 nelle tre seguenti righe. Risulta evidente che per l'ultima riga viene adottata una tecnica di completamento, perché se un pannello non figura in nessuna delle precedenti righe, sicuramente farà parte dell'ultima. Tutte le combinazioni calcolate sono conservate all'interno di un *array* di celle per poi essere analizzate e confrontate.

A differenza di quanto fatto in precedenti applicazioni [1] in cui si trattano solo matrici quadrate e simmetriche, si è implementato un algoritmo capace di considerare matrice $m \times n$ con un numero di pannelli arbitrario. Le tecniche utilizzate per la ricerca della soluzione ottimizzata si sono basate in un primo momento sullo sviluppo di un *algoritmo esaustivo*, capace di considerare tutte le possibili combinazioni tra i pannelli, utilizzabile però con tempi di calcolo accettabili solo per un numero di pannelli limitato. Si è quindi optato per l'impiego di algoritmi più efficienti dal punto di vista computazionale quali quello basato sulla tecnica *random search*, ed infine su un algoritmo di tipo *tabu search*.

L'algoritmo esaustivo ha il vantaggio di considerare il problema per intero senza tralasciare alcuna configurazione, infatti, genera tutte le combinazioni possibili e le analizza una per una confrontandole tra loro. L'unica neo è rappresentato dal fatto che per un numero elevato di pannelli questo opera un numero eccessivo di calcoli.

L'algoritmo *Random – Search* è un algoritmo che procede per tentativi operando una ricerca casuale della soluzione ottima, tramite un processo che non è mirato, e che talora può non convergere. Il principale vantaggio è che può trovare una soluzione buona in un numero limitato di passi. Lo svantaggio principale è che non ha la stessa efficienza dell'algoritmo esaustivo, poiché

non si ha la certezza che consideri tutte le configurazioni possibili. Il suo impiego si basa sul compromesso tra numero di soluzioni desiderate e tempo di calcolo ritenuto accettabile.

L'algoritmo di *Tabu Search* è una metodologia di calcolo che serve per superare i limiti tipici di una ricerca locale. Infatti, un algoritmo di Ricerca Locale prosegue la sua ricerca sino a quando trova soluzioni miglioranti; una volta individuato un ottimo locale, la ricerca si ferma a prescindere dalla qualità della soluzione trovata. Per garantire una migliore qualità della soluzione trovata occorrerebbe esplorare il maggior numero di ottimi locali. Negli algoritmi di tipo *Tabu Search* si rilascia il vincolo di trovare ad ogni iterazione una soluzione migliorante: ad ogni iterazione cerchiamo la migliore soluzione dell'intorno anche se questa è peggiore della precedente, così facendo la ricerca, in linea di principio, non termina mai e lo spazio delle soluzioni viene esplorato in modo più esteso. Il rischio di un approccio di questo tipo è quello di incorrere in una sequenza di soluzioni che determina un ciclo. Per evitare di incorrere in cicli si considera una *tabu list*: mantenendo una lista delle ultime soluzioni trovate possiamo evitare un ciclo definendo *tabu* una soluzione, dell'intorno corrente, che appartiene a questa lista; quindi le soluzioni *tabu* vengono scartate per evitare cicli. Una soluzione *tabu* non viene scartata se il valore della soluzione è migliore dell'ottimo trovato. In particolare, dopo aver creato la configurazione di partenza con tutti i pannelli connessi in serie tra loro, si procede allo spostamento degli stessi, presi ad uno ad uno, nelle righe così da stabilire una connessione in parallelo tra i pannelli della stessa riga. Quindi si calcolano i coefficienti d'irraggiamento delle righe e si applica il *principio di equalizzazione dell'irraggiamento*; questo procedimento si effettua per ogni pannello. Se lo spostamento di un pannello porta ad una configurazione migliore della configurazione considerata migliore prima dello spostamento, allora questa configurazione viene memorizzata come *Migliore Locale*, quindi si confronta con la migliore configurazione del sistema, *Migliore Globale* e si aggiorna la *tabu list* che può contenere tante mosse quanto è il suo *tabu tenure* che rappresenta la lunghezza della *tabu list*. La configurazione *migliore locale* è considerata come mossa di partenza per il successivo ciclo. Questi cicli si ripetono fin quando il *tabu tenure* si mantiene maggiore di 1. Il *tabu tenure* verrà decrementato ogni talvolta che la soluzione sia rimasta costante per un numero di passi prestabilito. Decrementare il *tabu tenure* significa ridurre la dimensione della *tabu list* quindi liberare alcune mosse *tabu*.

Il vantaggio di utilizzare un algoritmo di questo tipo è legato alla diminuzione dei tempi di esecuzione che risultano essere inferiori a quelli di un algoritmo esaustivo, perché non si vanno a considerare tutte le possibili combinazioni, ma soltanto quelle più favorevoli.

Dalle analisi effettuate si è osservato che l'*algoritmo esaustivo* è in grado di trovare la soluzione ottima, ma ha un tempo di esecuzione particolarmente alto. Impiega, infatti, oltre un'ora per completare le combinazioni possibili di un sistema a 8 pannelli, tempo da considerarsi eccessivo se si pensa che le condizioni meteorologiche possono mutare completamente in 5 minuti circa. Quindi se il numero di pannelli è minore di 8 il tempo di esecuzione potrebbe essere accettabile, ma per un numero elevato di pannelli è chiaro che i tempi di calcolo sono notevolmente alti. Per questo motivo si è preferito provare algoritmi euristici tra cui il migliore riscontrato è stato il *Tabu – Search*. L'algoritmo *Tabu – Search* non ha la certezza di considerare tutte le possibili combinazioni, ma adotta una tecnica di miglioramento dei risultati ottenuti precedentemente. Dato che non deve considerare tutte le possibili combinazioni, quindi deve compiere meno cicli, impiegherà un tempo minore dell'algoritmo esaustivo per generare la soluzione ottimizzata. Infatti, nel caso di sistema costituito da 8 pannelli, la soluzione ottimizzata è calcolata in modo istantaneo, mentre per un sistema a 50 pannelli impiega circa 20 millisecondi.

BIBLIOGRAFIA:

1. G. Velasco, J.J. Negroni, F. Guinjoan and R. Piqué: *Energy generation in PV grid-connected systems: power extraction optimization for plant oriented PV generators*, Industrial Electronics, 2005. ISIE 2005. Proceedings of the IEEE International Symposium on Volume 3, 20-23 June 2005 Page(s):1025 - 1030 vol. 3.
2. R. Candela, V. Di Dio, E. Riva Sanseverino, P. Romano: "Reconfiguration Techniques of Partial Shaded PV Systems for the Maximization of Electrical Energy Production". IEEE International Conference on Clean Electrical Power. 21-23 May, Capri, Italy, 2007, pp. 716-719.