

2. ANALISI BIBLIOGRAFICA DEI SISTEMI DI INTEGRAZIONE

2.1 LA FASE DI PROCESS PLANNING E SCHEDULING IN UN JOB SHOP FLESSIBILE

L'integrazione tra la fase di process planning e la fase di scheduling è importante per ottenere un'efficiente utilizzazione delle risorse produttive. Sebbene la fase di process planning e il job shop scheduling sono relazionati l'uno con l'altro, a lungo la comunità scientifica li ha considerati separatamente o in modo sequenziale.

Come abbiamo già analizzato, la fase di process planning stabilisce le richieste tecnologiche necessarie a convertire un particolare da una forma iniziale a una forma finita. Invece lo scheduling è un'altra funzione produttiva che tenta di assegnare le risorse di produzione alle operazioni indicate nel process plan con lo scopo di ottenere alcuni obiettivi come minimizzare il makespan. Se durante la fase di process planning, le decisioni sono prese senza tener conto dell'obiettivo di scheduling, potrebbero sorgere dei problemi, come colli di bottiglia dovuti a fattori quali carichi di lavoro non bilanciati.

L'approccio tradizionale al job shop scheduling supponeva che ci fosse un singolo process plan fattibile per ogni job. Quindi nessuna flessibilità era considerata nel process plan. Per ottenere una flessibilità nella produzione è possibile generare più process plans ammissibili per ogni job. Questo aspetto è stato preso in considerazione da Nasr e Elsayed [7], da Kim e Egbelu [8] e da Shafaei e Brunn [9].

Altri ricercatori hanno trattato la necessità di integrare la fase di process planning con la fase di scheduling. Iwata et al. [10], Nasr e Elsayed [7], Hutchison et al. [11] e Brandimarte [12] hanno considerato macchine alternative per ogni operazione, ma ogni job ha solo una singola sequenza di operazioni (routing flexibility).

Brandimarte e Calderini [13] hanno considerato differenti opzioni per ogni operazione tecnologica (operation flexibility), invece di considerare macchine alternative, ognuna delle quali consiste di un set di operazioni elementari. Loro suppongono che entrambi le macro e le micro-operazioni hanno relazioni di precedenza. Un'*operazione elementare* è una lavorazione tecnologica caratterizzata da un prefissato insieme di parametri quali la qualità della lavorazione (esempio sgrossatura o finitura), l'utensile utilizzato, la superficie interessata, la macchina utilizzata e il sistema di afferraggio. Al cambiare di uno solo di questi attributi si ha un'operazione elementare diversa.

Anche Hankins, Wysk e Fox [14], hanno discusso i vantaggi di usare macchine utensili alternative per migliorare la produttività. Loro hanno dimostrato che usando macchine alternative si riducono i lead-times e si migliora l'utilizzazione delle stesse.

L'integrazione tra la fase di planning e la fase di scheduling tramite la generazione di process plans che forniscono macchine alternative per eseguire i processi è stata inoltre analizzata da Dong e Parsaei [15]. Lo scheduling da loro analizzato usa come criterio le date di consegna di ogni job, calcolando il tempo per eseguire l'assegnamento delle macchine in real-time. Nello scheduling il sistema non prende in considerazione i tempi di setup e i tempi di material handling.

Chryssolouris et al.[16], propone di integrare process planning e scheduling focalizzando l'attenzione sul loro elemento comune: la necessità di assegnare le risorse per produrre i job. La fase di process planning ha il compito di generare i cicli di lavorazione laddove sono disponibili risorse alternative per ogni operazione con tecniche di tipo Closed Loop Process Planning (CLPP). Allora l'assegnamento delle risorse a ogni job è visto come un problema decisionale multiplo ed un approccio euristico è utilizzato per giungere alla selezione.

Zhao e Kops [17], considerano un sistema di process planning accoppiato con un sistema di scheduling generando un sistema a tre fasi principali. La prima fase genera un singolo process plan per un particolare, ma offre celle di lavorazione alternative. Nella seconda fase, un algoritmo di scheduling genera uno schedule per ogni cella. Il sistema di scheduling è flessibile in quanto usa macchine alternative specifiche in ogni plan. Dopo che lo scheduler ha selezionato quali risorse usare per ogni particolare, il terzo step riguarda una continuazione delle attività di process planning tramite le quali il sistema esegue le operazioni finali di planning per il particolare e ottiene il plan finale.

Per quanto riguarda la flessibilità, Kim, Park e Ko [18], propongono un sistema di integrazione standard nel senso che esso contempla routing ed operation flexibility, ma non un sistema di vincoli non-lineari. Particolare enfasi è data alla sequencing flexibility che corrisponde alla possibilità di intercambiare la sequenza secondo la quale le operazioni richieste sono eseguite. Il tipo di flessibilità considerato consiste in macchine alternative e sequenze alternative, ma le operazioni da eseguire sono fissate. I suddetti autori fanno anche notare che la flessibilità nel poter utilizzare cicli di lavorazione alternativi per una data operazione tecnologica consente di ottenere migliori performance in termini di flusso di tempo e utilizzazione delle macchine.

Kim e Egbelu [19], affrontano il problema dello scheduling in cui ad ogni particolare meccanico si associano process plans multipli in un ambiente di tipo job shop. Cioè è possibile generare differenti cicli di lavorazione per la realizzazione del prodotto finito. Ogni sequenza ammissibile di operazioni genera un process plan. Molti process plans aumentano la flessibilità di produzione perché considerano differenti cicli di lavorazione per una singola operazione tecnologica ma allo stesso tempo accrescono le difficoltà nella fase di planning e scheduling. Tale problema viene risolto tramite un approccio matematico. Due algoritmi sono sviluppati per trovare una soluzione al

modello. Il primo algoritmo, chiamato algoritmo di preprocessing ricerca uno spazio di soluzioni e risolve esplicitamente un sottoproblema soltanto quando è identificato un miglioramento per la soluzione. Il secondo algoritmo, chiamato algoritmo iterativo, è un metodo di ottimizzazione. Esso inizia con un set di process plans, S_1 , e risolve il problema di scheduling, associato con S_1 . Successivamente uno dei process plan in S_1 è rimosso ed è sostituito da un altro process plan appartenente allo stesso job. Questo genera un nuovo set S_2 , e viene risolto il problema di scheduling associato a tale set. Se c'è una soluzione migliore, la soluzione attuale è aggiornata e un nuovo set di process plan è generato. Il processo continua fino a che non è identificato nessun miglioramento nella qualità della soluzione.

Anche Nars e Elsayed [20], presentano due approcci iterativi per determinare uno schedule efficiente per un problema con n jobs e m macchine, con macchine utensili alternative per ogni operazione. L'obiettivo è minimizzare il makespan. Presentano un algoritmo efficiente per risolvere il problema, suddividendolo in sottoproblemi, più facili da risolvere. Tale algoritmo risolve i problemi in un tempo relativamente breve. Propongono inoltre un secondo algoritmo chiamato SFT (Shortest Finish Time). Questo è un'estensione del primo basato sulle regole del SPT (Shortest Processing Time), che è noto per trovare l'ottimo flusso temporale nel caso di macchina singola.

Usher e Fernandes [21], utilizzano un approccio a due fasi. Nella prima fase, static planning, vengono generati plans alternativi ed inclusi la selezione, l'assegnamento e la sequenza dei processi e delle macchine. L'output di questa fase è un set di plans alternativi di macro livello, che il sistema archivia. La fase di planning dinamico avviene con un approccio DPP (Distributed Process Planning) quando il job è pronto per essere prodotto. In tale fase i plans di macro livello vengono recuperati e si completa la fase di planning prendendo in considerazione la disponibilità di risorse e gli obiettivi specificati dallo scheduler.

2.2 MODELLI DI RAPPRESENTAZIONE DEL PROBLEMA DI INTEGRAZIONE

Per realizzare l'integrazione tra la fase di process planning e la fase di scheduling, nasce il concetto di non-linear routing. Tali process plans contengono routings alternativi, i quali offrono un grado alto di flessibilità allo scheduling.

Al contrario dei process plans tradizionali (lineari), un process plan non-lineare non contiene una sequenza fissata di operazioni, ma un set di routings con macchine alternative in un grafo AND/OR. In tale grafo i nodi AND rappresentano una sequenza di operazioni mentre i nodi OR rappresentano una scelta tra le varie operazioni alternative (Kempenaers, Pinte, Detand e Kruth [22]).

Anche Ho e Monodie [23], utilizzano la teoria dei grafi per rappresentare process plans flessibili. Ci sono tre tipi di nodi: il nodo iniziale, il nodo intermedio e il nodo finale. I nodi intermedi rappresentano le operazioni elementari mentre il nodo iniziale e quello finale sono nodi aggiuntivi ed indicano l'inizio e la fine di un processo produttivo di un job. I nodi intermedi contengono le macchine alternative che possono eseguire l'operazione e i tempi di lavorazione richiesti per l'operazione su ogni macchina.

Le frecce che collegano i nodi rappresentano le precedenze tra i nodi. Il simbolo OR è usato per descrivere una flessibilità di lavorazione, secondo la quale una stessa features può essere completata con differenti procedure di lavorazione, rappresentando quindi la precedentemente accennata operation flexibility. Se tra i due nodi c'è il simbolo OR allora bisogna scegliere solo una via tra le alternative possibili (figura 9).

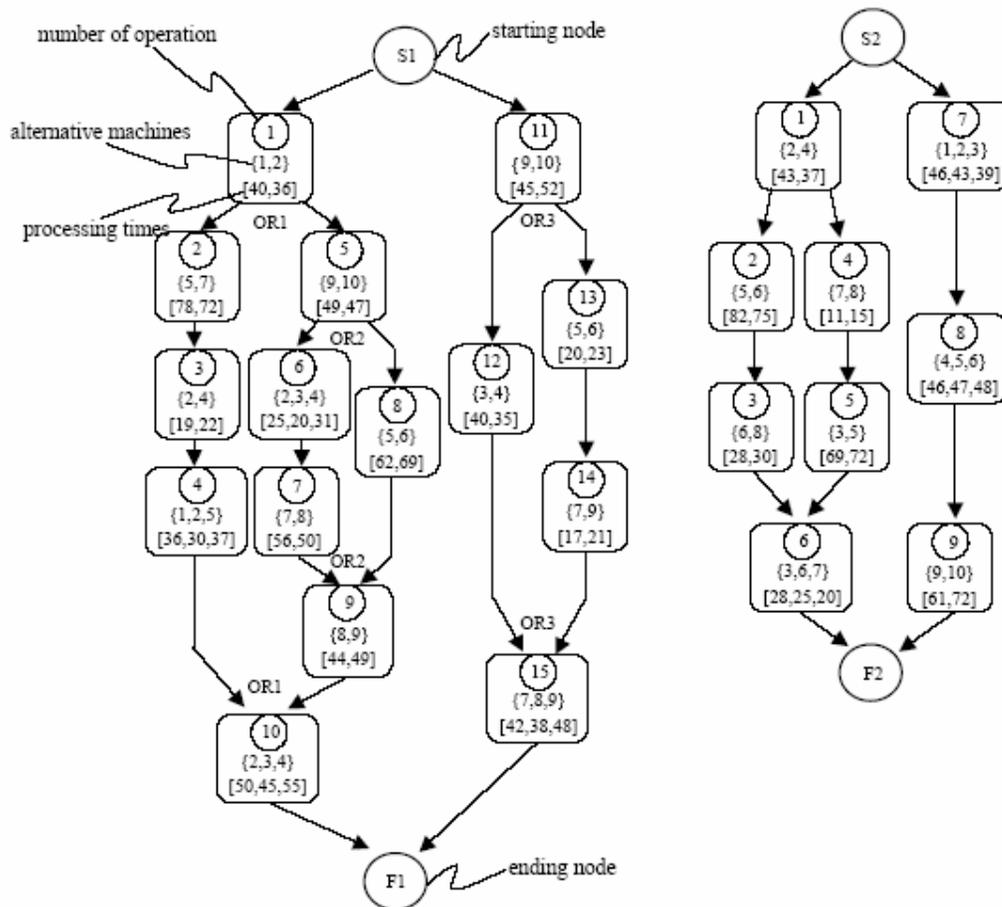


Figura 9. Grafo rappresentante process plans flessibili(Ho e Monodie) [23]

Lin e Solberg [24], hanno introdotto un grafo rappresentato anche esso da nodi AND e OR. Tutte le possibili operazioni di un particolare sono rappresentate da nodi e tutte le relazioni di precedenza tra le operazioni sono rappresentate da archi diretti. Ogni nodo porta le informazioni delle corrispondenti lavorazioni. I nodi sono classificati come nodi AND, nodi OR e nodi virtuali AND/OR. Un nodo AND connette un nodo padre a più nodi figli con il significato che tutte le operazioni “figlio” devono essere eseguite. Un nodo OR impone che una e solo una delle operazioni “figlio” deve essere eseguita. Un nodo virtuale AND/OR ha tempo di lavorazione nullo ed è usato quando si hanno casi in cui entrambi gli archi AND e gli archi OR sono necessari per un nodo (figura 10).

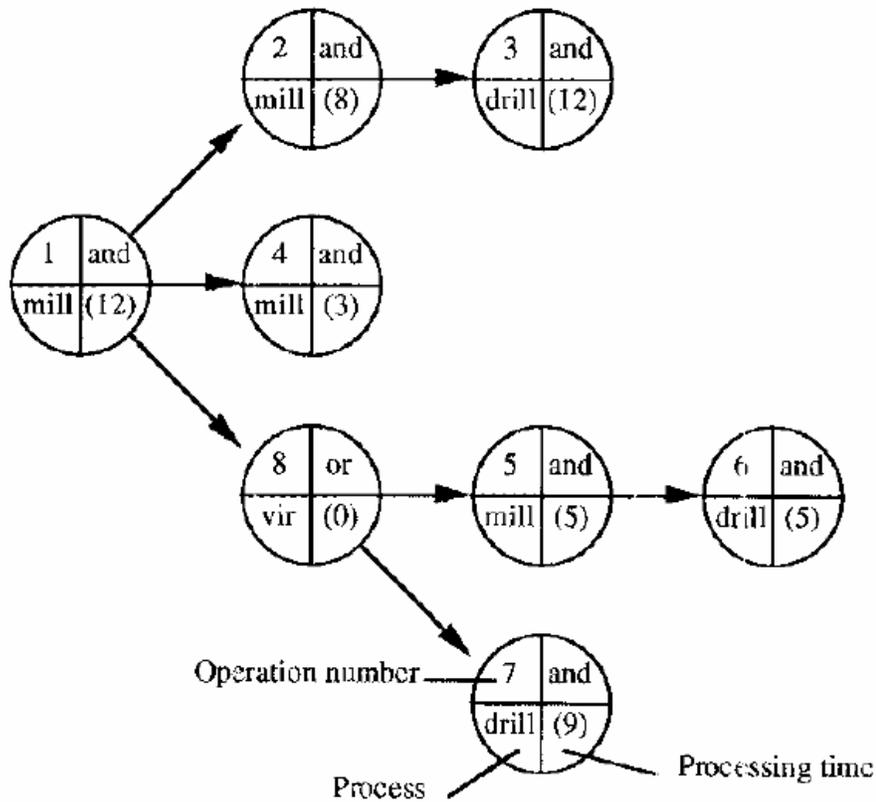


Figura 10. Rappresentazione di un grafo secondo Lin e Solberg [24]

Anche Usher e Fernandes [21] utilizzano la teoria dei grafi per ottenere i process plans. Questi sono ottenuti percorrendo il grafo da sinistra verso destra. Un nodo OR indica che bisogna seguire solo una via tra le alternative possibili. Mentre il nodo “&” rappresenta la necessità per il sistema di

percorrere tutti i cammini paralleli tra il nodo “&” e il nodo precedente. Questo implicherà di tornare indietro al precedente nodo e scegliere una via non ancora percorsa. La presenza di un nodo vuoto indica che nessuna lavorazione è necessaria in tale nodo e che quindi si può continuare a percorrere il grafo (figura 11)

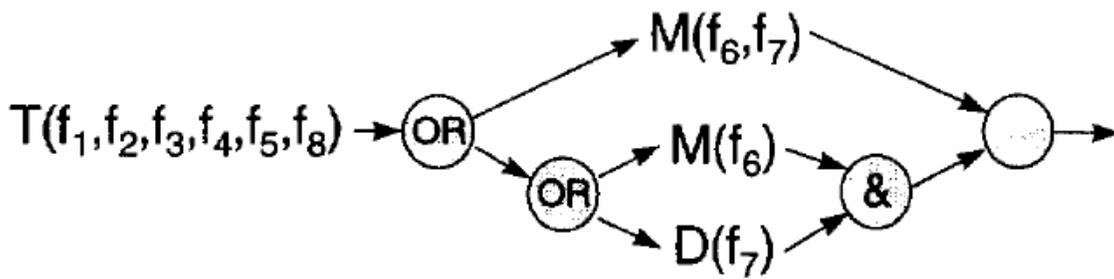


Figura 11. Rappresentazione di un grafo secondo Usher e Fernandes [21]