



Università degli studi di Padova

Dipartimento di Tecnica e Gestione dei Sistemi Industriali

Corso di Laurea Triennale in Ingegneria

Gestionale

**MODELLO PER LA SOLUZIONE DEL DYNAMIC FACILITY LAYOUT
PROBLEM CON REPARTI DI DIMENSIONI NON COSTANTI**

RELATORE: CH. MO PROF. GAMBERI MAURO

LAUREANDO: BARENGO MICHELE

ANNO ACCADEMICO: 2014/2015

INDICE

INTRODUZIONE.....	4
CAPITOLO 1	
Metodo euristico per il DFLP con reparti di dimensioni diverse.....	6
1. Introduzione.....	6
2. Tecnica di soluzione del DFLP.....	6
2.1. Boundary search heuristic (selection procedure e placement procedure).....	6
2.2. Tabu Search Heuristic (improvement procedure).....	8
CAPITOLO 2	
Modello matematico per la risoluzione del DFLP.....	9
CAPITOLO 3	
Presentazione dei problemi DFLP.....	11
1. P6.....	11
2. P12.....	11
CAPITOLO 4	
Risoluzione dei problemi.....	14
1. P6.....	14
2. P12.....	15
3. Commenti.....	18
BIBLIOGRAFIA.....	19

INTRODUZIONE

Il layout produttivo di un'azienda gioca un importante ruolo nel processo manifatturiero ed ha un forte impatto sul profitto della stessa. Il layout utilizzato definisce fisicamente le relazioni tra i vari reparti, come la movimentazione e la distribuzione dei materiali. Un layout efficiente permette dunque a un'azienda manifatturiera di ridurre i costi di produzione e aumentare la competitività.

L'SFLP (Static Facility Layout Problem) è il problema della disposizione dei reparti all'interno dell'area produttiva di un'azienda, in modo che questi non si sovrappongano e soddisfacendo una certa funzione obiettivo. Generalmente la funzione obiettivo da soddisfare è la minimizzazione del costo di movimentazione dei materiali. Tali costi possono essere approssimati con uno o più dei seguenti parametri: il flusso tra i reparti (f_{ij}), il costo di movimentazione di un'unità di carico per unità di distanza (c_{ij}), il grado di vicinanza tra i reparti (r_{ij}). Questi parametri sono utilizzati in due comuni funzioni di costo movimentazione dei materiali:

- $\max \sum_i \sum_j (r_{ij})x_{ij}$ dove $x = \begin{cases} 1 & \text{se il reparto } i \text{ è adiacente al reparto } j \\ 0 & \text{altrimenti} \end{cases}$
basata sul principio che il costo di movimentazione tra due reparti adiacenti sia ridotto sensibilmente.
- $\min \sum_i \sum_j (f_{ij}c_{ij})d_{ij}$ dove d_{ij} è la distanza tra il reparto i e il reparto j
basata sul principio che il costo di movimentazione aumenti all'aumentare della distanza che deve coprire l'unità di carico per portare il materiale dal reparto i al reparto j .

Questo problema è stato ampiamente studiato e sono stati proposti molti modelli matematici differenti e diversi metodi risolutivi sia esatti che euristici. Ma nel 1983 Nicol e Hollier sostennero la necessità di un trattamento dinamico di tale problema, infatti, essi intervistarono trentatré aziende manifatturiere di media dimensione e circa la metà di esse affermarono che mediamente la configurazione di un determinato layout permane per due anni o meno. Infatti, in un contesto reale nel 21° secolo le aziende lavorano in un ambiente dinamico e soggetto a variazioni, che richiedono la realizzazione di sistemi flessibili, quindi alla riorganizzazione del layout produttivo. Alcuni fattori che possono causare la variazione dei flussi di materiale tra i reparti sono:

- Cambiamento nel design di prodotti già esistenti;
- L'aggiunta o l'eliminazione di prodotti;
- La sostituzione di attrezzature di produzione esistenti;
- Cambiamenti nei volumi di produzione e i relativi piani di produzione.

Il DFLP (Dynamic Facility Layout Problem) è il problema di individuare la posizione dei reparti nel sistema produttivo per diversi periodi, in modo che tali reparti non si sovrappongano, e che sia minima la sommatoria dei costi di movimentazione materiali e dei costi di spostamento dei reparti. È definito dunque un layout plan per il DFLP, che consiste in una serie di layout associati ognuno a un certo periodo. Quindi il costo totale di un piano DFLP è dato dalla somma dei costi di movimentazione totali per ogni periodo con i costi di risistemazione totali. I costi di risistemazione sono dovuti allo spostamento di un reparto certa posizione a un'altra, e in esso sono contenuti il costo della manodopera e dell'attrezzatura per lo spostamento e un eventuale costo per la mancata produzione. Risolvere un DFLP non equivale semplicemente a risolvere una serie di SFLPs. Consideriamo i due casi estremi:

1. Se il costo di movimentazione è molto maggiore rispetto al costo di spostamento dei reparti, allora posso risolvere il DFLP risolvendo sequenzialmente i SFLPs per ogni periodo. In altri termini andando a risolvere l'SFLP per il primo periodo utilizzando i

dati solo del primo periodo, poi risolvendolo per il secondo periodo utilizzando i dati solo del secondo periodo e così via per tutti i periodi.

2. Se il costo di spostamento dei reparti è molto maggiore al costo di movimentazione, allora posso risolvere il DFLP come una serie di SFLPs. In questo caso si ottiene un layout per il primo periodo, che è assegnato a tutti i periodi, e se ne calcola il relativo costo totale di movimentazione. Poi si calcola il layout per il secondo e si assegna a tutti i periodi, calcolandone il relativo costo totale di movimentazione. Infine si selezionerà il layout che garantirà il minor costo totale di movimentazione.

Rosenblatt (1986) fu il pioniere della ricerca nel campo del DFLP, egli propose inizialmente un algoritmo esatto per la risoluzione, ma essendo un problema NP-difficile, propose successivamente degli algoritmi euristici basati sulla programmazione dinamica. Tali problemi, infatti, sono di tipo QAP (Quadratic Assignment Problem), per cui non è possibile risolverli in tempi polinomiali. Montreuil (1990) per primo propose una formulazione MIP (Mixed Integer Programming) per il FLP, che utilizza una funzione obiettivo basata sulle distanze, definite non su una intelaiatura discreta (come nei QAP), ma in una rappresentazione continua dello spazio. Urban (1993) propose un approccio euristico basato sull'interscambio di coppie steepest descent (o metodo del gradiente), evitando la complessità della risoluzione di un problema QAP con metodi ottimi. Conway e Venkataramanan (1994) usano un GA (Genetic Algorithm) per il CDFLP, ovvero il DFLP con un limite posto al costo totale di spostamento dei reparti; poi questo metodo viene migliorato da Balakrishnan e Cheng (2000). Un altro approccio meta-euristico proposto da Kaku e Mazzola (1997) è la TS (Tabu Search), che ricerca ottimi locali. Yang e Peters (1998) invece affrontano il problema considerando reparti non tutti della stessa dimensione, e i costi di spostamento dei reparti non dipendenti dalla distanza di cui sono spostati, ma come un costo fisso di spostamento. Baykasoglu e Gindy (2001) svilupparono un approccio SA (Simulated Annealing) per il DFLP con reparti di uguali dimensioni e forma, poi McKendall et al. (2006) applicarono la SA anche nel caso di reparti di dimensioni diverse. Balakrishnan et al. (2003) svilupparono un algoritmo ibrido basato sulla combinazione di GA con DP, per la risoluzione del DFLP con reparti di uguale dimensione e forma, poi Dunker et al. (2005) proposero un altro algoritmo ibrido basato su GA e DP che poteva risolvere il problema con reparti di dimensioni diverse. McKendall e Shang (2006) applicarono per primi HAS (Hybrid Ant System) al DFLP, ottenendo risultati migliori rispetto a tutti gli altri algoritmi ibridi fino ad allora proposti.

All'inizio del 21° secolo la ricerca si è focalizzata principalmente sulla risoluzione dell'SFLP (Stochastic Facility Layout Problem), una variante particolare di DFLP che invece di utilizzare le previsioni sull'andamento dei flussi nei periodi futuri, utilizza dati sul mix della domanda determinati stocasticamente. Infatti, risulta sempre più importante incorporare un certo grado di incertezza data la tendenza alla volatilità dell'economia e la tendenza dei cicli di vita dei prodotti a ridursi sempre più. Ma risulta una ricerca complessa che richiede anche la definizione di nuovi metodi di approssimazione della dinamica dei mercati, di nuove funzioni obiettivo e cambiamenti drastici nei layout preesistenti. Un altro ambito di ricerca è quello della risoluzione di problemi sempre più grandi, con un numero di reparti sempre maggiore. Inoltre ampia attenzione sta avendo il problema del FRLP (Facility Re-Layout Problem), caso particolare del DFLP, dove però si ha la configurazione del layout a partire da un layout preesistente, in quanto i casi di definizione del layout a partire da pianta vuota, sono molto meno diffusi rispetto alla richiesta di una risistemazione del layout. Un caso particolare dell'FRLP è il problema di espansione, che considera la risistemazione dei reparti in una azienda in funzione dell'aggiunta o dell'eliminazione di uno o più reparti. Inoltre un ruolo importante assume il concetto di layout dettagliato, cioè con la disposizione dei macchinari nei reparti, la posizione dei punti di Input e Output, i corridoi e la direzione dei flussi di materiale. Infatti, potrebbe essere che l'azienda sia in grado di rispondere al cambiamento della domanda, anche solo cambiando uno o più di questi fattori, senza ridefinire completamente il layout.

CAPITOLO 1

Metodo euristico per il DFLP con reparti di dimensioni diverse

1.Introduzione

Il paper di riferimento da cui è tratta la tesina è il lavoro di Alan R. McKendall e Artak Hakobyan del 2009. Qui è proposta una tecnica Boundary Search che piazza i reparti lungo i confini dei reparti già piazzati, fornendo così una soluzione per il DFLP, sotto le seguenti ipotesi:

- (1) I reparti potranno avere aree diverse. Cioè potranno essere sia rettangolari sia quadrati.
- (2) Le aree dei reparti sono fissate per ogni periodo, ma possono variare dall'uno all'altro.
- (3) I reparti potranno avere libero orientamento (verticale o orizzontale). Se il lato lungo è parallelo all'asse x, allora il reparto è orizzontale. Altrimenti, se il lato lungo è parallelo all'asse y, allora il reparto è verticale.
- (4) Il layout per ogni periodo è definito su una rappresentazione continua dello spazio.
- (5) L'obiettivo del DFLP è ottenere un layout per ciascun periodo dell'orizzonte di pianificazione tale che la somma del costo di movimentazione dei materiali con i costi di spostamento sia minima. Il costo di spostamento si avrà quando il reparto è effettivamente spostato (cioè quando i centri o le dimensioni cambiano in periodi consecutivi).

Una volta che una soluzione è costruita attraverso la Boundary Search, è usato un metodo euristico basato sulla Tabu Search per migliorare tale soluzione. È proposta una formulazione di tipo MILP (Mixed Integer Linear Programming) simile alla formulazione che si può trovare su Dunker et al. (2005), con la differenza che qui i punti di Input/Output sono ritenuti al centro del reparto.

2.Tecnica di soluzione del DFLP

La tecnica di soluzione del DFLP sotto le ipotesi (1)-(5) si articola in tre fasi:

- (1) Utilizza i dati sul flusso per determinare l'ordine in cui inserire i reparti sulla pianta (selection procedure).
- (2) Utilizza una procedura di piazzamento (placement procedure) per posizionare i reparti sulla pianta. Il risultato è il piano dei layout con il relativo costo.
- (3) Utilizza Tabu Search per migliorare il layout ottenuto alla fase 2 (improvement procedure).

2.1.Boundary search heuristic (selection procedure e placement procedure)

Anzitutto la BSH (Boundary Search Heuristic) consiste nelle procedure di selezione e piazzamento. Per prima cosa la selection procedure costruisce una soluzione. La soluzione è rappresentata mediante un vettore di coppie ordinate $\pi = \{(i_1, t_1), (i_2, t_2), \dots, (i_{NT}, t_{NT})\}$ e deve soddisfare le seguenti condizioni:

- (1)
$$\sum_{t=1}^T \sum_{d \neq i}^N F'_{tid} \geq \sum_{t=1}^T \sum_{d \neq j}^N F'_{tjd}$$
- (2)
$$\sum_{d \neq i}^N F'_{t1id} \geq \sum_{d \neq i}^N F'_{t2id}$$

In altre parole il reparto con il flusso totale massimo verso tutti i reparti in tutti i periodi, è messo come primo elemento secondo l'equazione (1), per tutti i periodi, solo una volta che i

è stato inserito in tutti i periodi si procede all'inserimento di un altro reparto. L'equazione (2) invece da un'indicazione sull'ordine dei periodi in cui inserire un determinato reparto, in particolare se il reparto i ha maggior flusso totale nel periodo 1 rispetto al periodo 2, allora (i, t_1) va inserito prima di (i, t_2) .

Costruita la soluzione iniziale π , la placement procedure determina la disposizione dei reparti sulla pianta. Innanzitutto si piazza il reparto i_1 nel periodo t_1 , e si pone al centro della pianta. Dopodiché seguendo l'ordine delle coppie in π , si piazza il secondo reparto, se $t_1 \neq t_2$, allora il reparto viene piazzato sempre al centro della pianta. Se $t_1 = t_2$ allora il reparto è piazzato lungo i bordi del reparto i_1 precedentemente piazzato, minimizzando la somma dei costi di movimentazione. Man mano che più reparti sono inseriti il reparto i_k nel periodo t_k viene inserito:

- nella posizione che minimizza la funzione di costo lungo i margini dei reparti già piazzati, se il costo di spostamento dei reparti risulta basso rispetto a quelli di movimentazione;
- nella posizione del medesimo reparto nel periodo t_{k-1} o t_{k+1} , se i costi di spostamento risultano particolarmente alti.

In ogni caso la posizione migliore è determinata in modo tale che siano minimi i costi di movimentazione e di riposizionamento. Il procedimento continua fino a che tutti i reparti non sono stati piazzati. Per identificare lo spazio ammissibile dove piazzare un nuovo reparto, BSH utilizza quattro vettori ordinati che definiscono i confini dei reparti già piazzati. In questi vettori sono contenuti tutti i bordi (orizzontale - superiore e inferiore, verticale - sinistro e destro) dei reparti piazzati. Ogni volta che un reparto è inserito, sono aggiornati i vettori. Per la ricerca delle aree ammissibili lungo i bordi efficientemente è utilizzata una ricerca binaria incorporata dentro BSH a partire dai vettori limite. Prima di porre un altro reparto è calcolato il centro di gravità, per il reparto che andrà posto successivamente, che rappresenta la posizione ottimale dove andrebbe inserito.

$$cg_x = \sum_{\forall j \in \pi_{per_{t_{curr}}}} (F'_{t_{curr}, i_{curr}, j} x_{t_{curr}, j}) / \sum_{\forall j \in \pi_{per_{t_{curr}}}} F'_{t_{curr}, i_{curr}, j}$$

$$cg_y = \sum_{\forall j \in \pi_{per_{t_{curr}}}} (F'_{t_{curr}, i_{curr}, j} y_{t_{curr}, j}) / \sum_{\forall j \in \pi_{per_{t_{curr}}}} F'_{t_{curr}, i_{curr}, j}$$

Individuata la posizione del centro di gravità, questo potrebbe trovarsi in un punto dove è già posizionato un altro reparto, per cui attraverso una ricerca binaria si vanno ad identificare delle aree ammissibili, lungo i bordi dei reparti già esistenti. Le aree individuate sono denominate candidate locations (posizioni candidate), a cui corrisponde un centro (x_{cand}, y_{cand}) . Per la scelta di quale tra questi punti candidati scegliere si guarda innanzitutto, se la scelta di quel punto originerà un layout ammissibile o meno (cioè se resta entro i confini della pianta), dopodiché se i punti sono entrambi ammissibili o entrambi inammissibili, allora si valuterà il costo totale generato dal posizionamento del reparto in quella determinata posizione ($TC_{i_{curr}, t_{curr}}$). Tale costo è dato dalla sommatoria dei costi di movimentazione con i reparti già piazzati con i costi di spostamento del reparto, che sarà pari a zero se né in $t_{curr}+1$ né in $t_{curr}-1$ il reparto i_{curr} è mai stato piazzato.

Il BSH per DFLP presentato è una variazione del Cluster Boundary Algorithm (CBA) presentato da Imam e Mir (1998) per il SFLP. Le differenze principali sono:

- CBA si può utilizzare solo per risolvere il SFLP e non per DFLP, mentre BSH si può utilizzare per risolvere entrambi i problemi;
- CBA piazza i reparti esclusivamente lungo i bordi di quelli già piazzati, mentre con BSH possono essere posizionati anche dove si trovano nel periodo precedente o successivo;

- CBA risulta molto lento nella ricerca delle aree ammissibili, mentre BSH tramite l'uso di centro di gravità, speciali vettori ordinati per definire i confini e un algoritmo di ricerca binaria, risulta più rapido.

2.2.Tabu Search Heuristic (improvement procedure)

Una volta ottenuto un iniziale layout plan utilizzando BSH si utilizza TS per migliorarlo. TS considera il vettore π con i relativi layout plan e costi totali, definendoli come correnti. Dopodiché considera tutti i possibili scambi reciproci tra i reparti in ogni periodo, $N(\pi)$, e considera le prime M mosse, ordinate in ordine crescente in base al costo stimato che si verificherebbe effettuando lo scambio. Considera poi la prima mossa dell'elenco, che genera la soluzione $\bar{\pi}$ e tramite un BSH modificato si ottengono il layout e il costo totale. Se quest'ultimo risulta minore di quello corrente, allora questo diventerà la nuova soluzione corrente, altrimenti si passa mossa successiva. Esaurite le M mosse, otterrò il layout migliore. La mossa utilizzata per ottenere la soluzione corrente è definita come TABU per un certo numero d'iterazioni. E può essere riutilizzata solo se il costo stimato per l'effettuazione della mossa è minore del costo minore scelto come corrente, o se è passato un certo numero di iterazioni da quando è stata definita tabu. In questa parte si utilizza una forma modificata di BSH per ottenere velocemente il layout e il costo totale stimato. Per far ciò BSH varia solo i centri dei reparti che sono scambiati nella mossa (senza considerare variazioni di lunghezza e larghezza) nel layout plan per la soluzione corrente. Dopodiché il costo totale è aggiornato determinando un costo stimato. Si tratta di un costo stimato poiché in realtà considerando anche le dimensioni dei reparti, potrebbero variare le posizioni effettive dei centri, anche di quelli che non sono stati scambiati nella mossa. Infatti, considerando il vettore ordinato π si ha che tutte le coppie (reparto, periodo) che si trovano prima della prima coppia scambiata, non subiscono modificazioni, cioè i reparti si trovano esattamente dove stanno nella soluzione corrente. Mentre le coppie che si trovano dopo subiranno delle variazioni nella posizione dei centri, il reparto si ritroverà nella vicinanza di dove era prima, infatti BSH cerca di minimizzare il più possibile tali spostamenti.

TS/BSH termina raggiunto un certo valore massimo di tempo di calcolo (Max_Time).

TS/BSH è migliore di TS per i seguenti motivi:

- TS/BSH risulta più rapido di un tradizionale TS poiché a differenza di questo che calcola il reale costo totale dovuto allo spostamento dei reparti, ne calcola un valore stimato, come precedentemente spiegato;
- TS/BSH usa una strategia di tipo a discesa per l'individuazione delle mosse da fare a differenza di un tradizionale TS che usa una strategia a discesa ripida, che individua un numero di soluzioni minore.

CAPITOLO 2

Modello matematico per la risoluzione del DFLP

In questa sezione è presentata una formulazione di tipo MILP per il DFLP con reparti di dimensioni diverse. Anzitutto è presentata la notazione che sarà utilizzata in seguito.

Indici

$i, j = 1, \dots, N$ dove N è il numero dei reparti
 $t = 1, \dots, T$ dove T è il numero di periodi

Parametri

F_{ijt} costo di trasporto dei materiali per unità di distanza dal reparto i al reparto j nel periodo t
 $F'_{ijt} = F_{ijt} + F_{jit}$ costo totale di trasporto dei materiali per unità di distanza dal reparto i al reparto j , nel periodo t (matrice triangolare superiore)

R_{it} costo di spostamento del reparto i all'inizio del periodo t

Sh_{it} lunghezza del lato più corto del reparto i nel periodo t

Lng_{it} lunghezza del lato più lungo del reparto i nel periodo t

$Orient_{it} = \begin{cases} 0 & \text{se il reparto } i \text{ nel periodo } t \text{ può avere orientamento qualsiasi} \\ 1 & \text{se il reparto } i \text{ nel periodo } t \text{ è vincolato ad essere orizzontale} \\ 2 & \text{se il reparto } i \text{ nel periodo } t \text{ è vincolato ad essere verticale} \end{cases}$

L lunghezza della pianta

W larghezza della pianta

M numero molto grande

Variabili

(x_{it}, y_{it}) centro del reparto i nel periodo t

l_{it}, w_{it} lunghezza e larghezza del reparto i nel periodo t

$x_{p_{ijt}}, y_{p_{ijt}}$ distanza orizzontale e verticale tra i centri dei reparti i e j nel periodo t

$h_{it} = \begin{cases} 1 & \text{se il reparto } i \text{ è orizzontale nel periodo } t \\ 0 & \text{altrimenti} \end{cases}$

$lef_{ijt} = \begin{cases} 1 & \text{se il reparto } i \text{ è a sinistra del reparto } j \text{ nel periodo } t \\ 0 & \text{altrimenti} \end{cases}$

$below_{ijt} = \begin{cases} 1 & \text{se il reparto } i \text{ è sotto il reparto } j \text{ nel periodo } t \\ 0 & \text{altrimenti} \end{cases}$

$r_{it} = \begin{cases} 1 & \text{se il reparto } i \text{ è riposizionato all'inizio del periodo } t \\ 0 & \text{altrimenti} \end{cases}$

La formulazione MILP del problema è la seguente:

$$(1) \quad \text{Minimize totalcost} = \sum_{t=1}^T \sum_{i=1}^N \sum_{j>i}^N F'_{ijt} (x_{p_{ijt}} + y_{p_{ijt}}) + \sum_{t=2}^T \sum_{i=1}^N Rr_{it}$$

Soggetta ai seguenti vincoli:

$$(2) \quad (x_{it} + 0.5l_{it}) - (x_{jt} - 0.5l_{jt}) \leq M(1 - left_{ijt}) \quad \forall i, j, t;$$

$$(3) \quad (y_{it} + 0.5w_{it}) - (y_{jt} - 0.5w_{jt}) \leq M(1 - below_{ijt}) \quad \forall i, j, t;$$

$$(4) \quad left_{ijt} + left_{jit} + below_{ijt} + below_{jit} = 1 \quad \forall i, j, t;$$

$$(5) \quad x_{it} + 0.5l_{it} \leq L \quad \forall i, t;$$

$$(6) \quad x_{it} - 0.5l_{it} \geq 0 \quad \forall i, t;$$

- (7) $y_{it} + 0.5w_{it} \leq W \quad \forall i, t;$
(8) $y_{it} - 0.5w_{it} \geq 0 \quad \forall i, t;$
(9) $x_{-p_{ijt}} \geq x_{it} - x_{jt} \quad \forall i, j, t;$
(10) $x_{-p_{ijt}} \geq x_{jt} - x_{it} \quad \forall i, j, t;$
(11) $y_{-p_{ijt}} \geq y_{it} - y_{jt} \quad \forall i, j, t;$
(12) $y_{-p_{ijt}} \geq y_{jt} - y_{it} \quad \forall i, j, t;$
(13) $l_{it} = Lng_{it}h_{it} + Sh_{it}(1 - h_{it}) \quad \forall i, t;$
(14) $w_{it} = Lng_{it}(1 - h_{it}) + Sh_{it}h_{it} \quad \forall i, t;$
(15) $h_{it} = 1 \quad \forall i, t, \quad Orient_{it} = 1;$
(16) $h_{it} = 0 \quad \forall i, t, \quad Orient_{it} = 2;$
(17) $x_{it} - x_{i,t-1} \leq Mr_{it} \quad \forall i, t > 1;$
(18) $-x_{it} + x_{i,t-1} \leq Mr_{it} \quad \forall i, t > 1;$
(19) $y_{it} - y_{i,t-1} \leq Mr_{it} \quad \forall i, t > 1;$
(20) $-y_{it} + y_{i,t-1} \leq Mr_{it} \quad \forall i, t > 1;$
(21) $w_{it} - w_{i,t-1} \leq Mr_{it} \quad \forall i, t > 1;$
(22) $-w_{it} + w_{i,t-1} \leq Mr_{it} \quad \forall i, t > 1;$
(23) $x_{it}, y_{it}, l_{it}, w_{it}, x_{-p_{ijt}}, y_{-p_{ijt}} \geq 0 \quad \forall i, j, t;$
 $h_{it} = 0 \text{ o } 1, r_{it} = 0 \text{ o } 1, left_{ijt} = 0 \text{ o } 1, below_{ijt} = 0 \text{ o } 1, \quad \forall i, j.$

L'equazione (1) è la funzione obiettivo, quella da minimizzare. La prima parte serve per ottenere i costi di movimentazione, mentre la seconda per i costi di spostamento dei reparti. Le equazioni (2)-(8) assicurano che i reparti non si sovrappongano ((2)-(4)) e non sconfinino ((5)-(8)). Le equazioni (9)-(12) servono per ottenere le distanze rettilinee tra i centri dei reparti. Equazioni simili a queste sono state presentate anche da Serali et al. (2003). Le equazioni (13)-(16) sono utilizzate per controllare l'orientamento dei reparti. Le equazioni (17)-(22) garantiscono che il reparto mantenga gli stessi valori di lunghezza, di larghezza e di coordinate del centro tra due periodi consecutivi in cui il reparto non è spostato. Queste equazioni da (13)-(22) sono simili a quelle definite da Dunker et al. (2005), con la differenza che in Dunker et al. (2005) il mantenimento dei valori, imponendo che siano le coordinate del centro e l'orientamento a rimanere gli stessi, mentre qui le coordinate del centro e le dimensioni. Inoltre in Dunker et al. (2005) s'impone che rimangano costanti le coordinate dei punti di Input/Output, che qui sono considerati posti in un unico punto al centro del reparto. Nella (23) invece sono fornite le restrizioni sulle variabili.

CAPITOLO 3

Presentazione dei problemi DFLP

Il modello è testato su due problemi presentati da Yang e Peters (1998) nella versione modificata da Dunker et al. (2005). In quest'ultima versione infatti si utilizzano dei costi di spostamento dei reparti più bassi per permettere di avere più possibilità di cambiamento di layout nella soluzione. I problemi si chiamano P6 e P12 e fanno riferimento rispettivamente a un problema per 6 e 12 reparti, partendo da un layout iniziale.

1.P6

$N=6$ $T=6$

$W=L=30$

$R=19$

Valori iniziali:

Reparti (i)	1	2	3	4	5	6
x_i	12.0	0.0	4.5	6.5	0.0	8.5
y_i	6.0	0.0	7.0	0.5	6.0	6.0
orient _i	1	0	0	1	0	1
Lng _i	5	9	6	6	4	5
Sh _i	4	8	5	4	4	3

Matrici di flusso aggregate:

Periodo 1

	2	3	4	5	6
1	6	10	2	10	16
2	-	17	15	15	8
3	-	-	6	4	18
4	-	-	-	11	19
5	-	-	-	-	12

Periodo 4

	2	3	4	5	6
1	3	6	4	19	11
2	-	7	0	15	12
3	-	-	8	6	7
4	-	-	-	2	14
5	-	-	-	-	9

Periodo 2

	2	3	4	5	6
1	0	12	1	3	14
2	-	19	1	10	1
3	-	-	14	9	8
4	-	-	-	11	4
5	-	-	-	-	12

Periodo 5

	2	3	4	5	6
1	3	9	3	17	6
2	-	6	18	15	11
3	-	-	7	14	11
4	-	-	-	5	2
5	-	-	-	-	6

Periodo 3

	2	3	4	5	6
1	1	11	15	13	14
2	-	10	19	15	1
3	-	-	12	9	8
4	-	-	-	6	4
5	-	-	-	-	12

Periodo 6

	2	3	4	5	6
1	4	5	14	2	11
2	-	7	14	13	1
3	-	-	16	8	8
4	-	-	-	5	3
5	-	-	-	-	7

2.P12

$N=12$ $T=4$

$L=W=60$

$R=50$

Valori iniziali:

i	1	2	3	4	5	6	7	8	9	10	11	12
x_i	0.0	5.5	11.5	16.0	5.5	16.0	5.5	17.0	11.0	11.5	11.5	0.5
y_i	2.5	13.5	2.0	4.0	8.0	13.5	0.0	8.5	8.0	18.5	13.5	8.0
orient _i	1	0	1	0	0	1	1	0	1	0	0	1
Lng _i	5	7	6	4	6	5	10	7	6	5	5	6
Sh _i	4	5	5	4	6	4	7	5	5	5	5	4

Matrici di flusso aggregate:

Periodo 1

	2	3	4	5	6	7	8	9	10	11	12
1	9	3	6	1	10	18	5	19	10	13	13
2	-	9	5	2	13	17	16	8	0	18	16
3	-	-	0	2	2	7	15	16	18	16	19
4	-	-	-	8	19	1	17	15	1	15	0
5	-	-	-	-	10	1	17	15	3	7	12
6	-	-	-	-	-	11	13	13	14	13	6
7	-	-	-	-	-	-	18	6	10	4	14
8	-	-	-	-	-	-	-	11	19	6	10
9	-	-	-	-	-	-	-	-	4	11	12
10	-	-	-	-	-	-	-	-	-	15	3
11	-	-	-	-	-	-	-	-	-	-	6

Periodo 2

	2	3	4	5	6	7	8	9	10	11	12
1	4	0	8	4	19	8	5	6	15	0	3
2	-	0	18	16	7	4	8	0	13	5	9
3	-	-	3	0	12	5	11	19	2	4	11
4	-	-	-	0	10	1	16	3	5	2	12
5	-	-	-	-	13	18	12	1	1	1	12
6	-	-	-	-	-	2	2	14	18	5	9
7	-	-	-	-	-	-	3	12	6	5	5
8	-	-	-	-	-	-	-	19	3	2	6
9	-	-	-	-	-	-	-	-	9	13	19
10	-	-	-	-	-	-	-	-	-	7	10
11	-	-	-	-	-	-	-	-	-	-	6

Periodo 3

	2	3	4	5	6	7	8	9	10	11	12
1	3	12	8	17	13	6	15	7	14	5	18
2	-	3	13	14	19	11	11	1	19	11	3
3	-	-	18	14	4	16	13	7	7	12	12
4	-	-	-	6	2	5	13	5	8	10	7
5	-	-	-	-	4	16	18	4	6	16	14
6	-	-	-	-	-	1	9	3	6	2	0
7	-	-	-	-	-	-	15	0	8	15	12
8	-	-	-	-	-	-	-	17	11	6	4
9	-	-	-	-	-	-	-	-	8	11	18
10	-	-	-	-	-	-	-	-	-	1	11
11	-	-	-	-	-	-	-	-	-	-	3

Periodo 4

	2	3	4	5	6	7	8	9	10	11	12
1	16	11	2	9	15	18	10	15	2	15	16
2	-	2	6	14	15	19	16	16	3	3	9
3	-	-	9	2	9	1	0	5	9	10	1
4	-	-	-	11	6	3	11	0	3	0	4
5	-	-	-	-	10	18	7	1	16	6	19
6	-	-	-	-	-	6	5	19	11	5	1
7	-	-	-	-	-	-	2	6	14	13	18
8	-	-	-	-	-	-	-	5	19	14	1
9	-	-	-	-	-	-	-	-	2	12	5
10	-	-	-	-	-	-	-	-	-	8	0
11	-	-	-	-	-	-	-	-	-	-	13

CAPITOLO 4

Risoluzione dei problemi

Il modello è stato implementato utilizzando GAMS, per entrambi i problemi. Si è scelto di adottare delle semplificazioni, quali considerare che i reparti conservino le dimensioni tra i periodi e di ignorarne l'orientamento. Queste semplificazioni sono pensate anche nell'ottica che sia poi il progettista a disegnare il layout definitivo, ricevendo dall'elaboratore le posizioni dei centri e il costo complessivo di realizzazione del layout plan.

1.P6

```
SETS i "reparti" /i1*i6/, t "periodi" /t1*t6/;
ALIAS (i,j);
SCALAR r "costi di riassetto" /19/, length "lunghezza layout" /30/, width "larghezza layout" /30/,
m "numero scalare grande" /1000/;
PARAMETERS w(i) "lato corto" /i1 4, i2 8, i3 5, i4 4, i5 4, i6 3/,
l(i) "lato lungo" /i1 5, i2 9, i3 6, i4 6, i5 4, i6 5/,
x1(i) "coordinata x del centro in 1" /i1 12.0, i3 4.5, i4 6.5, i6 8.5/,
y1(i) "coordinata y del centro in 1" /i1 6.0, i3 7.0, i4 0.5, i5 6.0, i6 6.0/;
```

TABLE f1(i,j)

	i2	i3	i4	i5	i6
i1	6	10	2	10	6
i2		17	15	15	11
i3			6	4	11
i4				11	2
i5					6

TABLE f(i,j,t)

	i2.t2	i3.t2	i4.t2	i5.t2	i6.t2
i1		12	1	3	14
i2		19	1	10	1
i3			14	9	8
i4				11	4
i5					12

+	i2.t3	i3.t3	i4.t3	i5.t3	i6.t3
i1	1	11	15	13	14
i2		10	19	15	1
i3			12	9	8
i4				6	4
i5					12

+	i2.t4	i3.t4	i4.t4	i5.t4	i6.t4
i1	3	6	4	19	11
i2		7		15	12
i3			8	6	7
i4				2	14
i5					9

	+	i2.t5	i3.t5	i4.t5	i5.t5	i6.t5
i1		3	9	3	17	6
i2			6	18	15	11
i3				7	14	11
i4					5	2
i5						6

	+	i2.t6	i3.t6	i4.t6	i5.t6	i6.t6
i1		4	5	14	2	11
i2			7	14	13	1
i3				16	8	8
i4					5	3
i5						7;

VARIABLES

x(i,t), y(i,t), dx(i,j,t), dy(i,j,t), spost(i,t), costotot
left(i,j,t), below(i,j,t), dx1(i,j), dy1(i,j), a(i,j,t), b(i,j,t), s1(i,t),s2(i,t), s3(i,t), s4(i,t);
POSITIVE VARIABLES x,y,dx,dy,dx1,dy1;
BINARY VARIABLES spost,left,below;
FREE VARIABLES a,b;

EQUATIONS

ob "funzione obiettivo"
posx(i,j,t), posy(i,j,t) "distanza relativa tra i bordi dei reparti"
sx(i,j,t), sotto(i,j,t) "posizione reciproca tra i reparti"
vinc1(i,j,t), vinc2(i,j,t), vinc3(i,j,t), vinc4(i,t), vinc5(i,t), vinc6(i,t), vinc7(i,t) "overlap e
sconfinamento"
distx1(i,j),distx2(i,j),disty1(i,j),disty2(i,j) "distanze rettilinee nel periodo iniziale"
vinc8(i,j,t), vinc9(i,j,t), vinc10(i,j,t), vinc11(i,j,t) "distanze rettilinee tra i centri"
sp1(i,t),sp2(i,t),sp3(i,t),sp4(i,t),spostamento(i,t) "spostamento reparto"
vinc12(i,t), vinc13(i,t), vinc14(i,t), vinc15(i,t) "conservazione dei parametri se non
spostato";
ob..costotot=e=sum((i,j),f1(i,j)*(dx1(i,j)+dy1(i,j)))+sum((i,j,t),f(i,j,t)*(dx(i,j,t)+dy(i,j,t)))+sum((i,t),
r*spost(i,t));
posx(i,j,t)..a(i,j,t)=e=(x(i,t)+0.5*l(i)-x(j,t)-0.5*l(j));
posy(i,j,t)..b(i,j,t)=e=(y(i,t)+0.5*w(i)-y(j,t)-0.5*w(j));
sx(i,j,t)..left(i,j,t)=e=1\$(a(i,j,t)<0);
sotto(i,j,t)..below(i,j,t)=e=1\$(b(i,j,t)<0);
vinc1(i,j,t)..((x(i,t)+0.5*l(i))-(x(j,t)-0.5*l(j)))=l= m*(1-left(i,j,t));
vinc2(i,j,t)..((y(i,t)+0.5*w(i))-(y(j,t)-0.5*w(j)))=l= m*(1-below(i,j,t));
vinc3(i,j,t)..left(i,j,t)+left(j,i,t)+below(i,j,t)+below(j,i,t)=e=1;
vinc4(i,t)..x(i,t)+0.5*l(i)=l=length;
vinc5(i,t)..x(i,t)-0.5*l(i)=g=0;
vinc6(i,t)..y(i,t)+0.5*w(i)=l=width;
vinc7(i,t)..y(i,t)-0.5*w(i)=g=0;
distx1(i,j)..dx1(i,j)=g=x1(i)-x1(j);
distx2(i,j)..dx1(i,j)=g=x1(j)-x1(i);
disty1(i,j)..dy1(i,j)=g=y1(i)-y1(j);
disty2(i,j)..dy1(i,j)=g=y1(j)-y1(i);
vinc8(i,j,t)..dx(i,j,t)=g=x(i,t)-x(j,t);
vinc9(i,j,t)..dx(i,j,t)=g=x(j,t)-x(i,t);
vinc10(i,j,t)..dy(i,j,t)=g=y(i,t)-y(j,t);
vinc11(i,j,t)..dy(i,j,t)=g=y(j,t)-y(i,t);
sp1(i,t)..s1(i,t)=e=x(i,t)-x(i,t-1);
sp2(i,t)..s2(i,t)=e=x(i,t)-x(i,t+1);
sp3(i,t)..s3(i,t)=e=y(i,t)-y(i,t-1);
sp4(i,t)..s4(i,t)=e=y(i,t)-y(i,t+1);


```

spostamento(i,t)..spost(i,t)=e=1$((s1(i,t) or s2(i,t) or s3(i,t) or s4(i,t))<>0);
vinc12(i,t)..x(i,t)-x(i,t-1)=l=m*spost(i,t);
vinc13(i,t)..-x(i,t)+x(i,t-1)=l=m*spost(i,t);
vinc14(i,t)..y(i,t)-y(i,t-1)=l=m*spost(i,t);
vinc15(i,t)..-y(i,t)+y(i,t-1)=l=m*spost(i,t);
MODEL p6/all/;
OPTIONS mip=cplex, optcr=0.1;
SOLVE p6 USING mip MINIMIZING costotot;
DISPLAY costotot.l, x.l, y.l;

```

2.P12

```

SETS i "reparti" /i1*i12/, t "periodi" /t1*t4/;
ALIAS (i,j);
SCALAR r "costi di riassetto" /50/, length "lunghezza layout" /60/, width "larghezza layout"
/60/, m "numero scalare grande" /2000/;
PARAMETERS
  w(i) "lato corto" /i1 4, i2 5, i3 5, i4 4, i5 6, i6 4, i7 7, i8 5, i9 5, i10 5, i11 5, i12 4/,
  l(i) "lato lungo" /i1 5, i2 7, i3 6, i4 4, i5 6, i6 5, i7 10, i8 7, i9 6, i10 5, i11 5, i12 6/,
  x1(i) "coordinata x del centro in 1" /i1 0.0, i2 5.5, i3 11.5, i4 16.0, i5 5.5, i6 16.0, i7 5.5,
  i8 17.0, i9 11.0, i10 11.5, i11 11.5, i12 0.5/,
  y1(i) "coordinata y del centro in 1" /i1 2.5, i2 13.5, i3 2.0, i4 4.0, i5 8.0, i6 13.5, i7 0.0,
  i8 8.5, i9 8.0, i10 18.5, i11 13.5, i12 8.0/;

```

TABLE f1(i,j)

	i2	i3	i4	i5	i6	i7	i8	i9	i10	i11	i12
i1	9	3	6	1	10	18	5	19	10	13	13
i2		9	5	2	13	17	16	8		18	16
i3				2	2	7	15	16	18	16	19
i4				8	19	1	17	15	1	15	
i5					10	1	17	15	3	7	12
i6						11	13	13	14	13	6
i7							18	6	10	4	14
i8								11	19	6	10
i9									4	11	12
i10										15	3
i11											6 ;

TABLE f(i,j,t)

	i2.t2	i3.t2	i4.t2	i5.t2	i6.t2	i7.t2	i8.t2	i9.t2	i10.t2	i11.t2	i12.t2	
i1	4			8	4	19	8	5	6	15	0	3
i2				18	16	7	4	8		13	5	9
i3				3		12	5	11	19	2	4	11
i4						10	1	16	3	5	2	12
i5						13	18	12	1	1	1	12
i6							2	2	14	18	5	9
i7								3	12	16	5	5
i8									19	3	2	6
i9										9	13	19
i10											7	10
i11												6

	i2.t3	i3.t3	i4.t3	i5.t3	i6.t3	i7.t3	i8.t3	i9.t3	i10.t3	i11.t3	i12.t3
i1	3	12	8	17	13	6	15	7	14	5	18
i2		3	13	14	19	11	11	1	19	11	3
i3			18	14	4	16	13	7	7	12	12
i4				6	2	5	13	5	8	10	7
i5					4	16	18	4	6	16	14
i6						1	9	3	6	2	
i7							15		8	15	12
i8								17	11	6	4
i9									8	11	18
i10										1	11
i11											3

	i2.t4	i3.t4	i4.t4	i5.t4	i6.t4	i7.t4	i8.t4	i9.t4	i10.t4	i11.t4	i12.t4
i1	16	11	2	9	15	18	10	15	2	15	16
i2		2	6	14	15	19	16	16	3	3	9
i3			9	2	9	1		5	9	10	1
i4				11	6	3	11		3		4
i5					10	18	7	1	16	6	19
i6						6	5	19	11	5	1
i7							2	6	14	13	18
i8								5	19	14	1
i9									2	12	5
i10										8	
i11											13;

VARIABLES

x(i,t), y(i,t), dx(i,j,t), dy(i,j,t), spost(i,t), costotot
left(i,j,t), below(i,j,t), dx1(i,j), dy1(i,j), a(i,j,t), b(i,j,t), s1(i,t),s2(i,t), s3(i,t), s4(i,t);
POSITIVE VARIABLES x,y,dx,dy,dx1,dy1;
BINARY VARIABLE spost,left,below;
FREE VARIABLES a,b;

EQUATIONS

ob "funzione obiettivo"
posx(i,j,t), posy(i,j,t) "distanza relativa tra i bordi dei reparti"
sx(i,j,t), sotto(i,j,t) "posizione reciproca tra i reparti"
vinc1(i,j,t), vinc2(i,j,t), vinc3(i,j,t), vinc4(i,t), vinc5(i,t), vinc6(i,t), vinc7(i,t) "overlap e sconfinamento"
distx1(i,j),distx2(i,j),disty1(i,j),disty2(i,j) "distanze rettilinee nel periodo iniziale"
vinc8(i,j,t), vinc9(i,j,t), vinc10(i,j,t), vinc11(i,j,t) "distanze rettilinee tra i centri"
sp1(i,t),sp2(i,t),sp3(i,t),sp4(i,t),spostamento(i,t) "spostamento reparto"
vinc12(i,t), vinc13(i,t), vinc14(i,t), vinc15(i,t) "conservazione dei parametri se non spostato";
ob..costotot=e=sum((i,j),f1(i,j)*(dx1(i,j)+dy1(i,j)))+sum((i,j,t),f(i,j,t)*(dx(i,j,t)+dy(i,j,t)))+sum((i,t), r*spost(i,t));
posx(i,j,t)..a(i,j,t)=e=(x(i,t)+0.5*l(i)-x(j,t)-0.5*l(j));
posy(i,j,t)..b(i,j,t)=e=(y(i,t)+0.5*w(i)-y(j,t)-0.5*w(j));
sx(i,j,t)..left(i,j,t)=e=1\$(a(i,j,t)<0);
sotto(i,j,t)..below(i,j,t)=e=1\$(b(i,j,t)<0);
vinc1(i,j,t)..((x(i,t)+0.5*l(i))-x(j,t)-0.5*l(j))=l= m*(1-left(i,j,t));
vinc2(i,j,t)..((y(i,t)+0.5*w(i))-y(j,t)-0.5*w(j))=l= m*(1-below(i,j,t));
vinc3(i,j,t)..left(i,j,t)+left(j,i,t)+below(i,j,t)+below(j,i,t)=e=1;
vinc4(i,t)..x(i,t)+0.5*l(i)=l=length;
vinc5(i,t)..x(i,t)-0.5*l(i)=g=0;
vinc6(i,t)..y(i,t)+0.5*w(i)=l=width;
vinc7(i,t)..y(i,t)-0.5*w(i)=g=0;

```

distx1(i,j)..dx1(i,j)=g=x1(i)-x1(j);
distx2(i,j)..dx1(i,j)=g=x1(j)-x1(i);
disty1(i,j)..dy1(i,j)=g=y1(i)-y1(j);
disty2(i,j)..dy1(i,j)=g=y1(j)-y1(i);
vinc8(i,j,t)..dx(i,j,t)=g=x(i,t)-x(j,t);
vinc9(i,j,t)..dx(i,j,t)=g=x(j,t)-x(i,t);
vinc10(i,j,t)..dy(i,j,t)=g=y(i,t)-y(j,t);
vinc11(i,j,t)..dy(i,j,t)=g=y(j,t)-y(i,t);
sp1(i,t)..s1(i,t)=e=x(i,t)-x(i,t-1);
sp2(i,t)..s2(i,t)=e=x(i,t)-x(i,t+1);
sp3(i,t)..s3(i,t)=e=y(i,t)-y(i,t-1);
sp4(i,t)..s4(i,t)=e=y(i,t)-y(i,t+1);
spostamento(i,t)..spost(i,t)=e=1$(s1(i,t) or s2(i,t) or s3(i,t) or s4(i,t))<>0);
vinc12(i,t)..x(i,t)-x(i,t-1)=l=m*spost(i,t);
vinc13(i,t)..-x(i,t)+x(i,t-1)=l=m*spost(i,t);
vinc14(i,t)..y(i,t)-y(i,t-1)=l=m*spost(i,t);
vinc15(i,t)..-y(i,t)+y(i,t-1)=l=m*spost(i,t);
MODEL p12/all;
OPTIONS mip=cplex, optcr=0.1;
SOLVE p12 USING mip MINIMIZING costotot;
DISPLAY costotot.l, x.l, y.l;

```

3. Commenti

Le equazioni presenti nella codifica sono quelle presentate nella formulazione MIP del problema, cui sono state aggiunte le equazioni $posx(i,j,t)$, $posy(i,j,t)$, $sx(i,j,t)$, $sotto(i,j,t)$, $distx1(i,j)$, $distx2(i,j)$, $disty1(i,j)$, $disty2(i,j)$, $sp1(i,t)$, $sp2(i,t)$, $sp3(i,t)$, $sp4(i,t)$, $spostamento(i,t)$. Mentre sono state tolte quelle relative all'orientamento dei reparti. Le equazioni $sx(i,j,t)$ e $sotto(i,j,t)$ servono per definire la posizione relativa tra i reparti, nell'equazione sx la variabile $left$ assume valore pari a 1 se il reparto i si trova a sinistra del reparto j , analogamente nell'equazione $sotto$, la variabile $below$ assume valore pari a 1 se il reparto i si trova sotto al reparto j . La posizione dei reparti è determinata nelle equazioni $posx(i,j,t)$ e $posy(i,j,t)$, mediante il calcolo di $a(i,j,t)$ e $b(i,j,t)$. $a(i,j,t)$ corrisponde alla differenza tra le coordinate x del bordo destro del reparto i e il bordo sinistro del reparto j , se tale differenza risulta negativa, allora significa che il reparto i si trova a sinistra del reparto j . Analogamente la variabile $b(i,j,t)$ corrisponde alla differenza tra le coordinate y del bordo superiore del reparto i e il bordo inferiore del reparto j . Le equazioni $distx1(i,j)$, $distx2(i,j)$, $disty1(i,j)$ e $disty2(i,j)$ invece sono inserite per calcolare le distanze rettilinee lungo x e lungo y tra i centri dei reparti nel periodo iniziale $t=1$. Infatti nei problemi proposti si parte da un layout iniziale già esistente, quindi si sono inserite le condizioni iniziali come parametri, anche se facenti riferimento a grandezze considerate variabili nel problema. Le equazioni $sp1(i,t)$, $sp2(i,t)$, $sp3(i,t)$, $sp4(i,t)$ e $spostamento(i,t)$, sono utilizzate per determinare se il reparto è spostato da un periodo all'altro. Nelle equazioni $sp1(i,t)$, $sp2(i,t)$, $sp3(i,t)$ e $sp4(i,t)$, sono calcolate le differenze ($s1$, $s2$, $s3$ e $s4$) tra le coordinate x e y , dello stesso reparto tra il periodo t , $t-1$ e $t+1$. Nell'equazione $spostamento(i,t)$ è assegnato alla variabile $spost(i,t)$ valore pari a 1 se almeno una delle differenze precedentemente calcolate risultano diverse da zero.

Bibliografia

Alan R. Mckendall Jr., Artak Hakobyan – Heuristics for the dynamic facility layout problem with unequal-area departments, *European Journal of Operational Research* 201 (2010) 171-182

Ehsan Shahbazi – Mathematical Modeling and Solution Development for “Dynamic Facility Layout Problem” In Job Shop Process Using Meta-heuristic Algorithms, Department of Industrial Engineering, Islamic AZAD University, Ilam Branch, Iran

Gary Chen, Jamie Rogers – Managing Dynamic Facility Layout with Multiple Objectives, *PICMET 2009 Proceedings*, August 2-6, Portland, Oregon USA (1175-1176)

Ramazan Sahin, Orhan Türkbey – A new hybrid tabu-simulated annealing heuristic for the dynamic facility layout problem, *International Journal of Production Research*, Vol. 47, No. 24, 15 December 2009 (6855-6858)

Russell D. Meller, Kai-Yin Gau – The Facility Layout Problem: Recent and Emerging Trends and Perspectives, *Journal of Manufacturing Systems* Vol. 15/No. 5 1996 (351-358)

Sadan Kulturel-Konak – Approaches to uncertainties in facility layout problems: Perspectives at the beginning of the 21st Century, *J Intell Manuf* (2007) 18: 273-284

T. Yang, B.A. Peters - Flexible machine layout design for dynamic and uncertain production environments, *European Journal of Operational Research* 108 (1998) (49-64)