



An Insect-Based Approach to the Dynamic Task Allocation Problem

Seminario per il corso di Sistemi Intelligenti

Prof. N. Alberto BORGHESE

Relatore: Roberto GHIZZOLI

22/10/2004



Scopo del lavoro e Tesi

Scopo del lavoro

sviluppare e implementare Ant Task Allocation (ATA), un **algoritmo** basato sul modello di **divisione del lavoro** osservato in **colonie di formiche** per risolvere un problema di **scheduling**.

Tesi

per diverse **classi di istanze** del problema e per una determinata **funzione obiettivo**, l'algoritmo sviluppato ottiene dei **risultati migliori** rispetto agli altri algoritmi considerati.



Il problema di Dynamic Task Allocation

Problema di **scheduling**, **non-deterministico**, con **macchine parallele** e parte dell'informazione disponibile solo durante la risoluzione del problema stesso.

Sistema industriale di verniciatura:

Scopo: assegnare camion a cabine verniciatura

- **m cabine che lavorano in parallelo** e con **coda** di lunghezza fissa
 - identiche – caso **omogeneo** del problema
 - differenti velocità processo per attività – caso **eterogeneo** del problema
- **costi/tempo di setup** (cambio colore)
- **n camion** da verniciare
- **date di rilascio** e **colori** non conosciuti all'inizio dell'esperimento

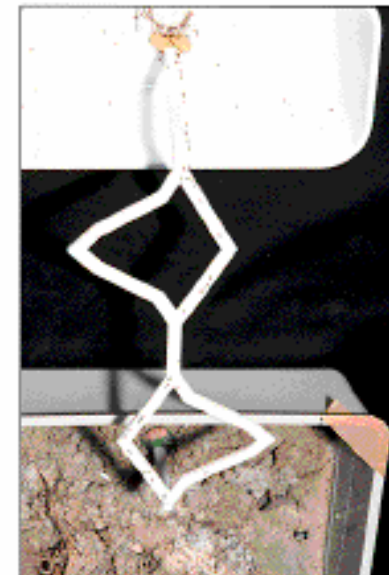
Obiettivo: minimizzazione makespan



Divisione del lavoro in colonie di insetti

Perché utilizzare l'intelligenza di colonie di insetti ?

- **Parallelizzazione** delle attività
- **Specializzazione** degli individui
- Divisione in **caste**
- Forma di **comunicazione indiretta**
- **Plasticità** della colonia



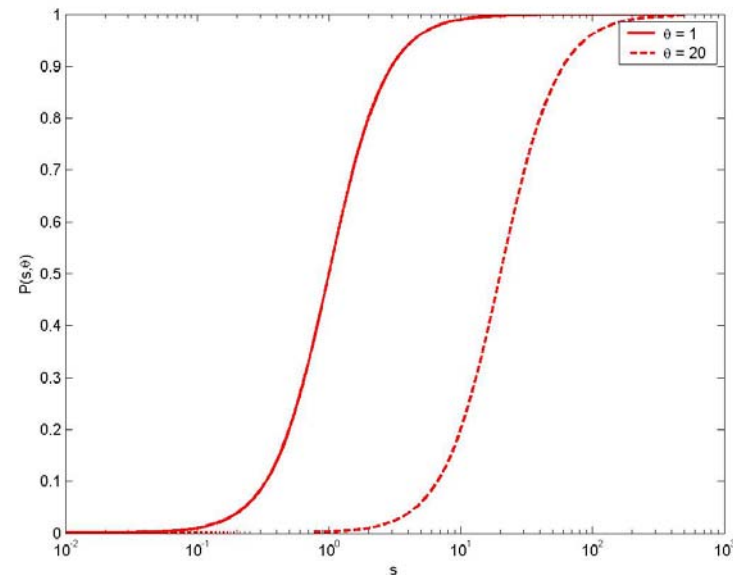
(Wilson *et al.*, 1984)
(Robinson *et al.*, 1987)
(Robinson *et al.*, 1992)
(Calderone *et al.*, 1996)



Il modello a soglie (Theraulaz *et al.*, 1998)

- **Stimolo s_j** : esigenza di una attività j (aumenta se l'attività non viene svolta)
- **Soglie di risposta $\theta_{i,j}$** : livello di specializzazione di un agente i per una attività j
- **Funzione di risposta $P(s_j, \theta_{i,j})$** : probabilità di un agente i ad essere ingaggiato in una attività j

$$P(s_j, \theta_{i,j}) = \frac{s_j^2}{s_j^2 + \theta_{i,j}^2}$$



Il modello a soglie (Theraulaz *et al.*, 1998) – cont.

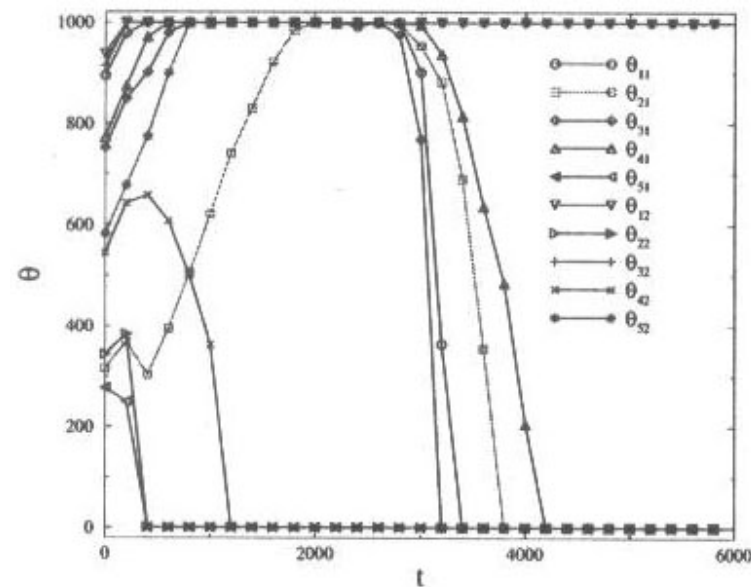
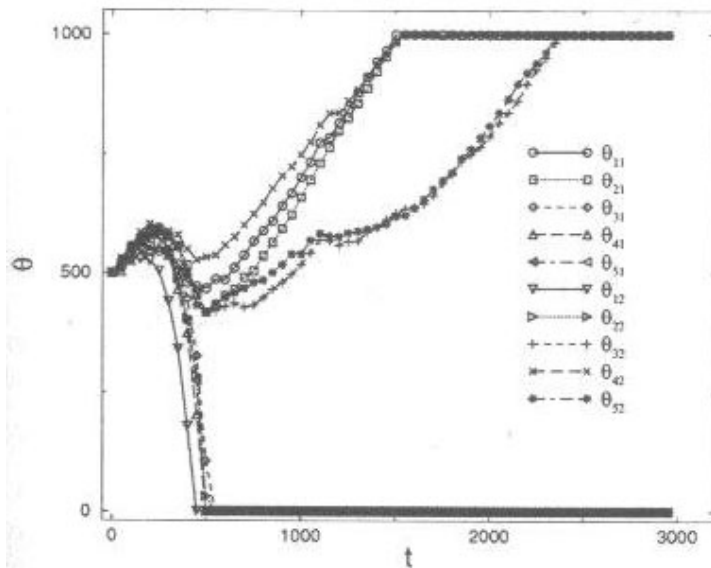
Come si specializza un agente?

Se un agente i sta eseguendo una attività j allora

$$\theta_{i,j}(t+1) \leftarrow \theta_{i,j}(t) - \alpha$$

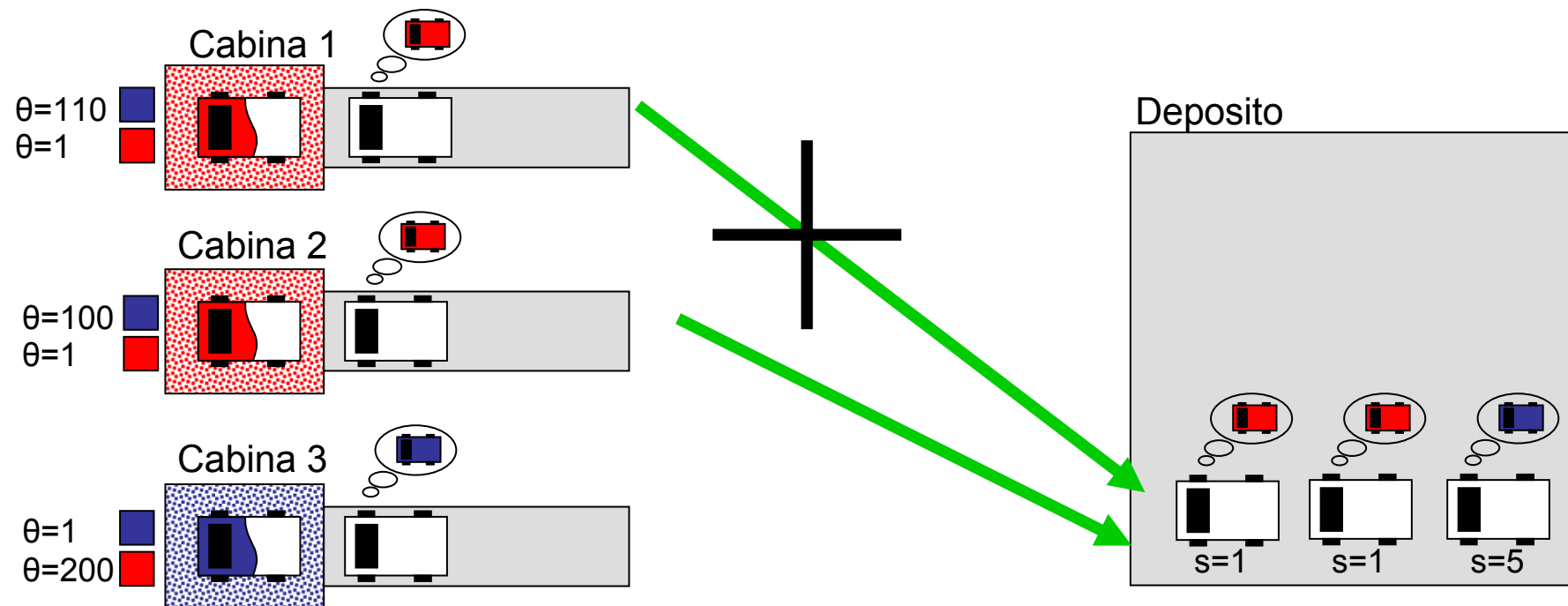
altrimenti:

$$\theta_{i,j}(t+1) \leftarrow \theta_{i,j}(t) + \beta$$



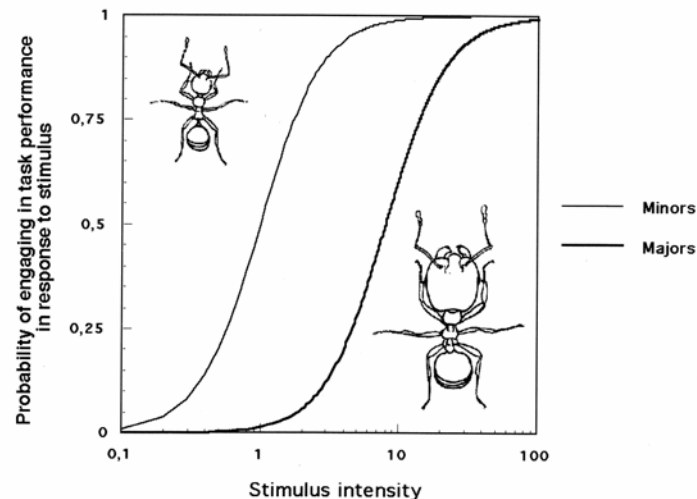
ATA: la soluzione proposta al caso omogeneo

- Basata sull'algoritmo R-WASP proposto da Cicirello *et al.* (2003) che utilizza il modello a soglie di Theraulaz *et al.* (1998).
- modifica tre regole dell'algoritmo base e ne aggiunge una (TUR, CFV, DOC, IMB)



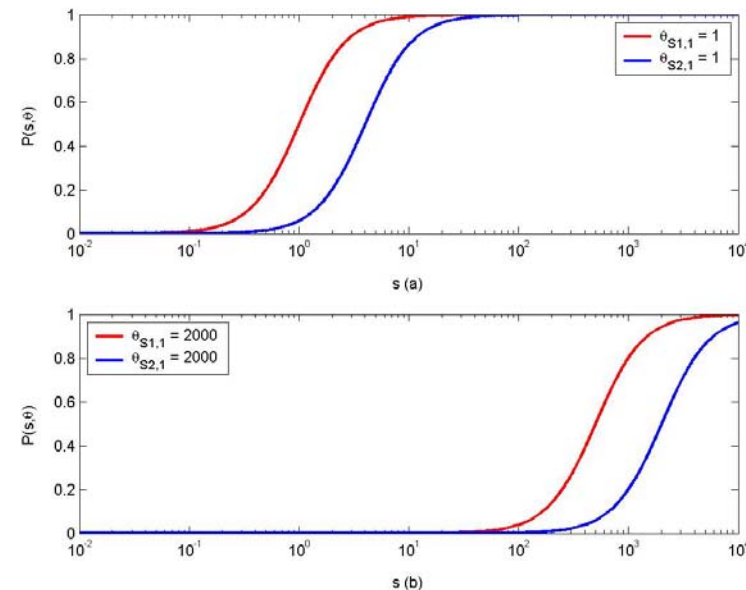
ATAc: la soluzione al caso eterogeneo

Wilson osservò che differenti caste di formiche hanno **differenti probabilità** di essere ingaggiate in un particolare lavoro in funzione delle loro caratteristiche



Regola Different Process Speed (DPS)

$$P(s_j, \theta_{i,j}) = \frac{s_j^2}{s_j^2 + \theta_{i,j}^2 (t_{proc,i,j} - tmin_{proc,j} + 1)}$$

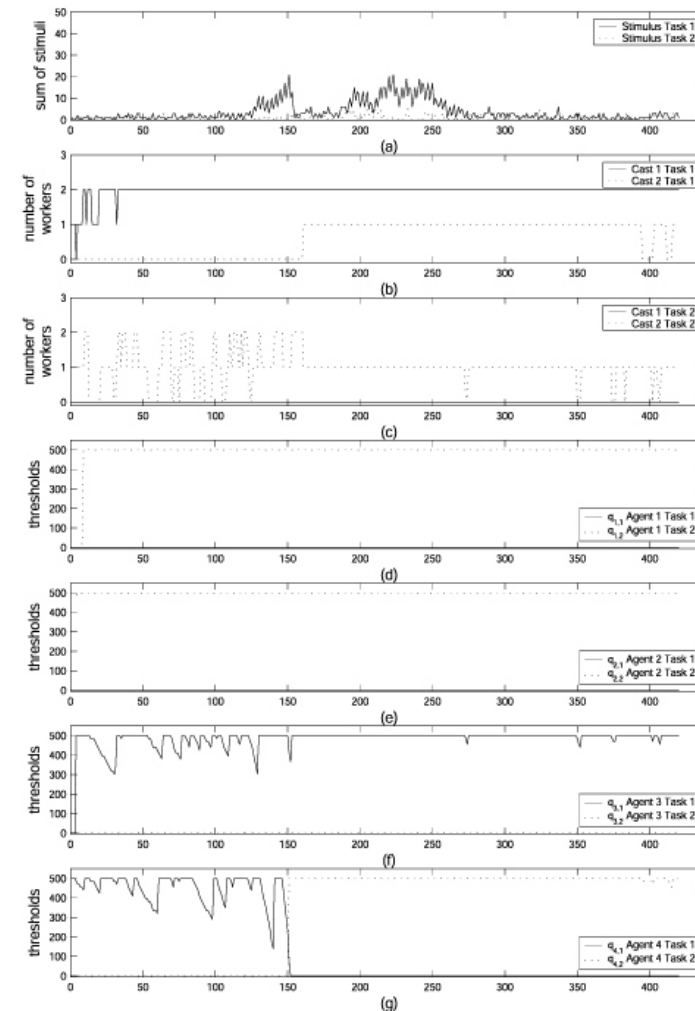


ATAc: esempio su una istanza semplice

Configurazione dell'esempio:

- 4 Macchine e 2 colori (bianco, nero)
 - 2 veloci a verniciare bianco
 - 2 veloci a verniciare nero

3/4 dei camion andranno verniciati di bianco!



Analisi empirica: le istanze considerate

- 2 classi di istanze del problema
- Simulazione su 7 ore lavorative
- Numero di camion = max. processabili in 420 min. senza setup e senza tempi di attesa
- Mix probabilità colori particolarmente dinamico

	<i>DTA_H</i> Class	<i>DTA</i> Class
Subsets of agents	1	2
Max. number of agents	24	12
Broken probability	0.02	0.02
Queue length	10	5
Process Time	5	3-9
Setup Time	10	10
Simulation Time	420	420
Number of Tasks	2016	840
Avg. types of tasks	12	10
Types mixes	1) $P(1..n/4) = 3P(n + 1/4..n)$ 2) like 1, switch after 210 steps	1) $P(1..n/2) = 3P(n + 1/2..n)$ 2) like 1, switch after 210 steps
Probability between mixes	0.5	0.5



Analisi empirica: gli algoritmi confrontati

MBA: algoritmo market-based applicato al sistema di verniciatura della GM (Morley, 1996)

ABA: primo algoritmo insect-based per risolvere lo stesso problema (Campos *et.al*, 2000)

R-WASP: la soluzione insect-based di base (Cicirello *et. al*, 2001)

ATA: la soluzione proposta (Ghizzioli *et al.*, 2004)

LOCUST: algoritmo non adattativo introdotto come punto di riferimento per il confronto delle prestazioni (Ghizzioli *et al.*, 2004)

I parametri di ogni algoritmo sono stati determinati utilizzando degli **algoritmi genetici** sulle rispettive classi del problema

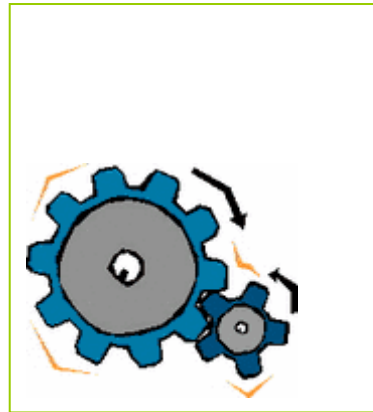


Analisi empirica: il processo sperimentale

Preferenze degli algoritmi:

- *parametri(soglie, costanti incr/decr)*
- *preferenze applicativo (logs ..)*

DTA solver



3 tabelle:

- *makespan*
- *numero setups*
- *dim. deposito*

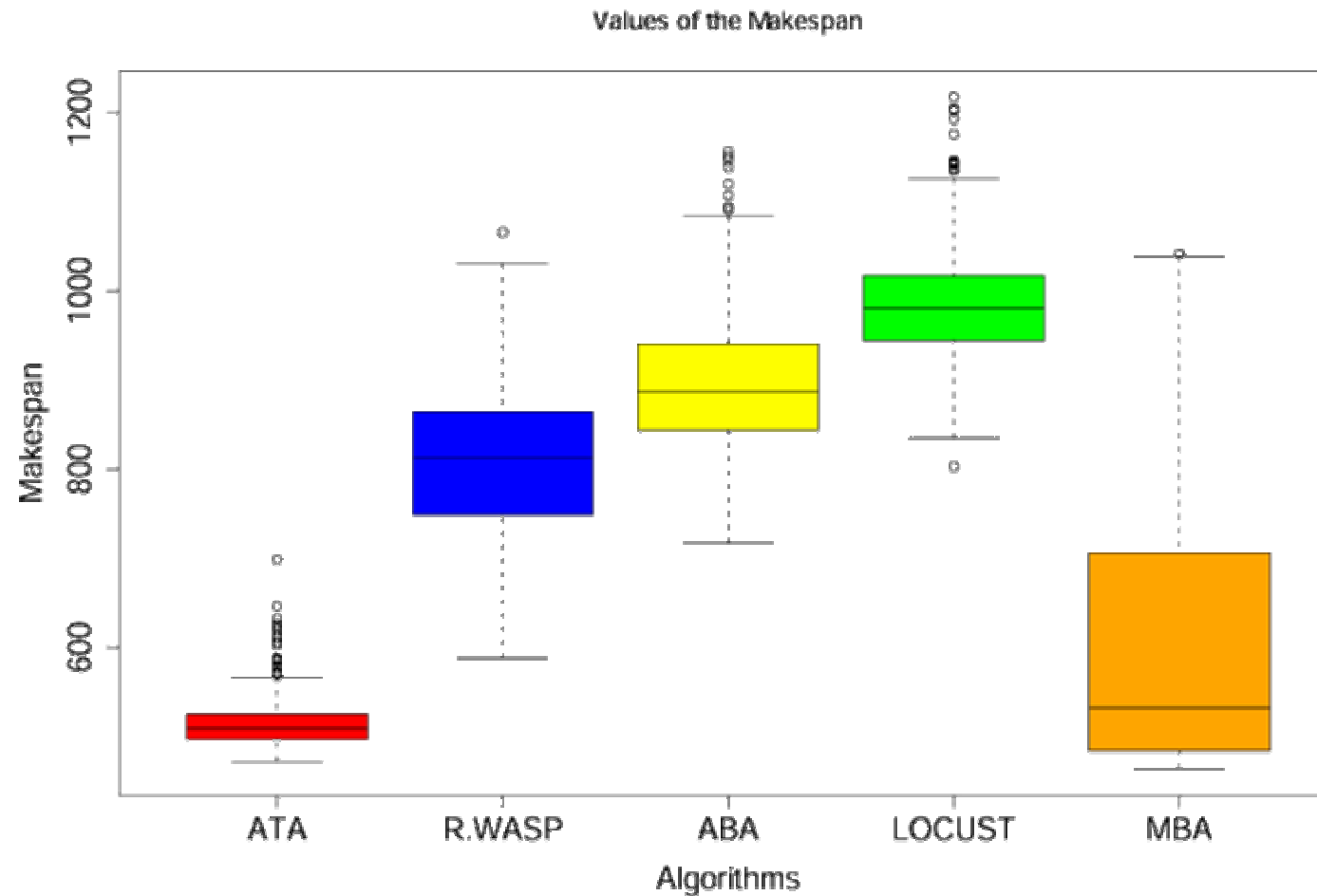
colonne: algoritmi
righe: risultati della singola istanza

Istanze:

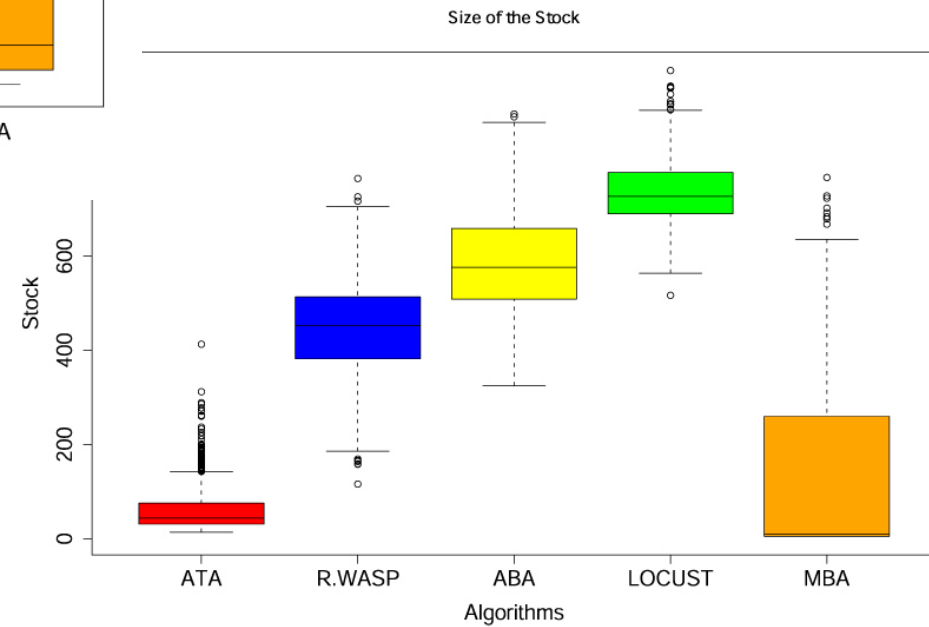
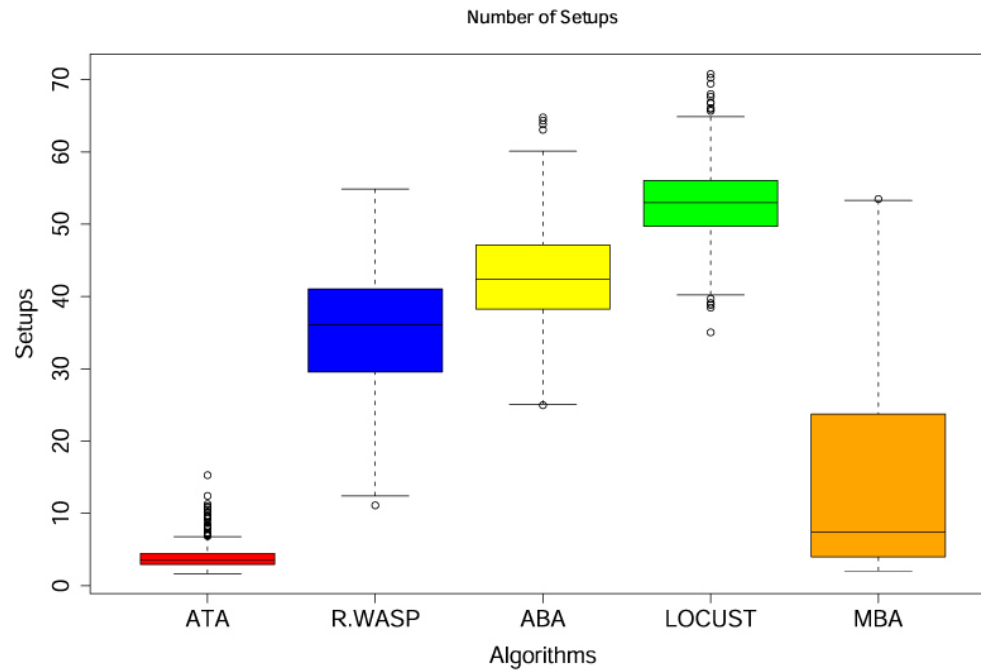
- *configurazione sistema (vedi tabella precedente)*
- *elenco dei camion*



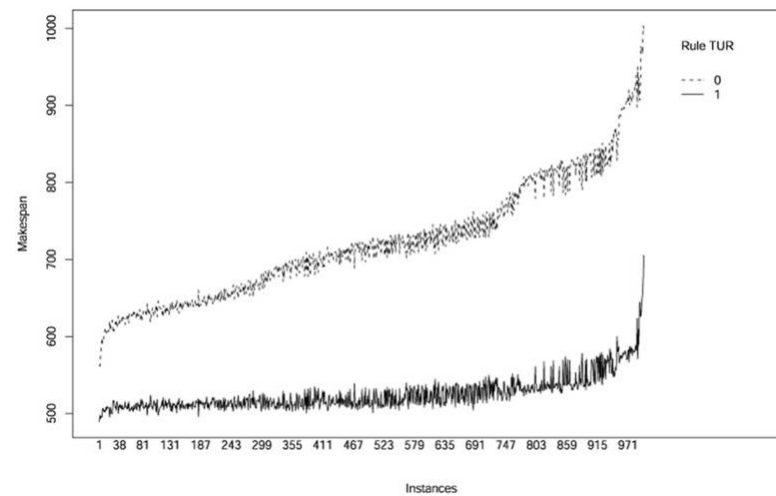
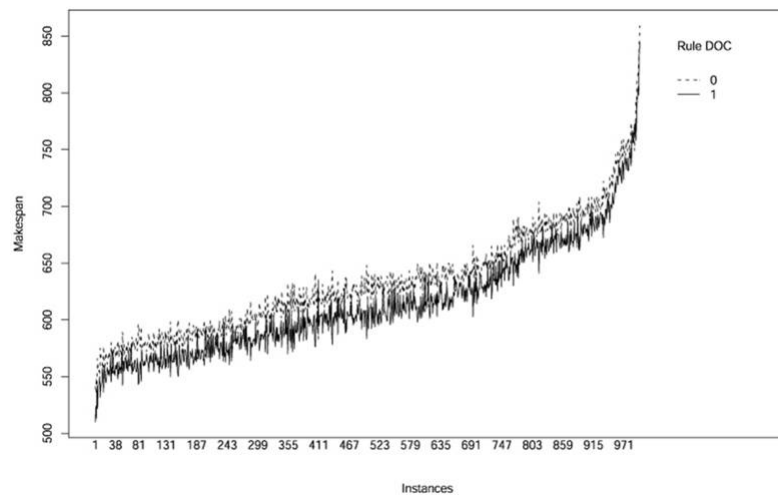
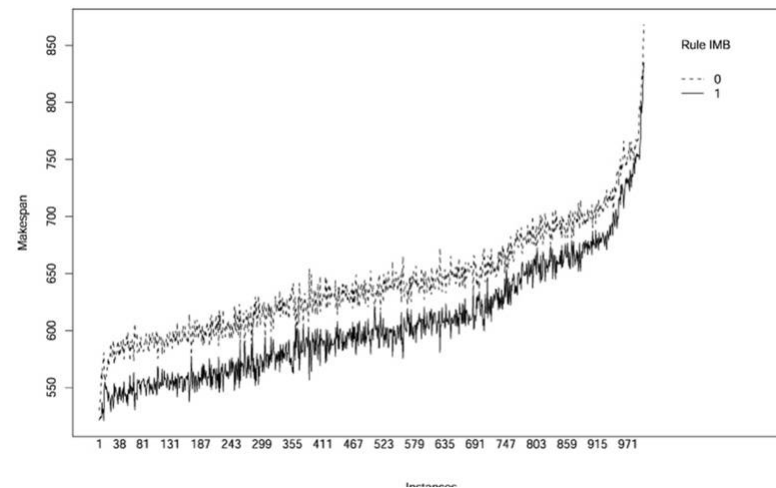
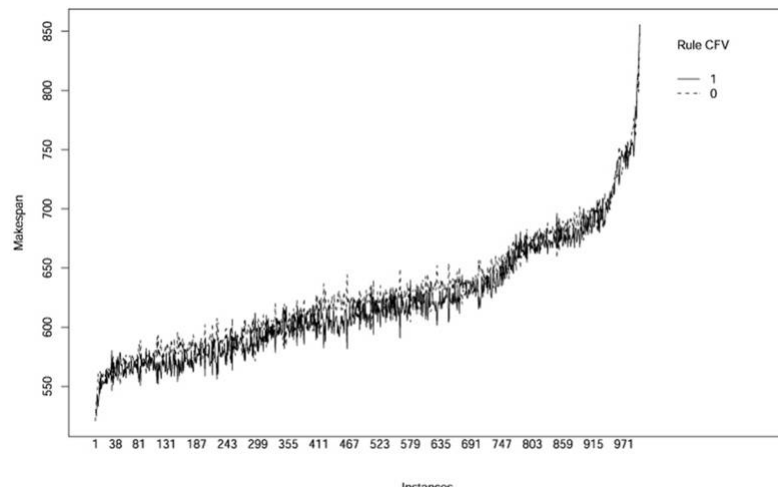
Risultati caso omogeneo



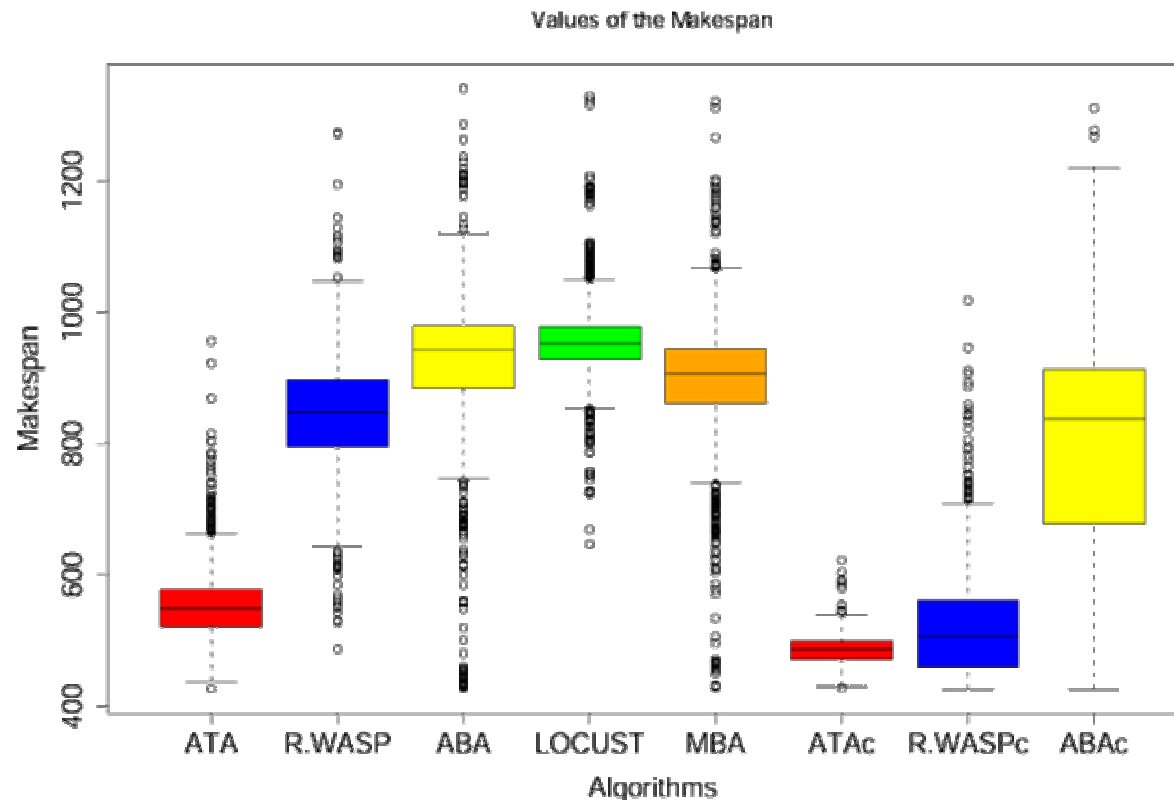
Risultati caso omogeneo



Risultati caso omogeneo



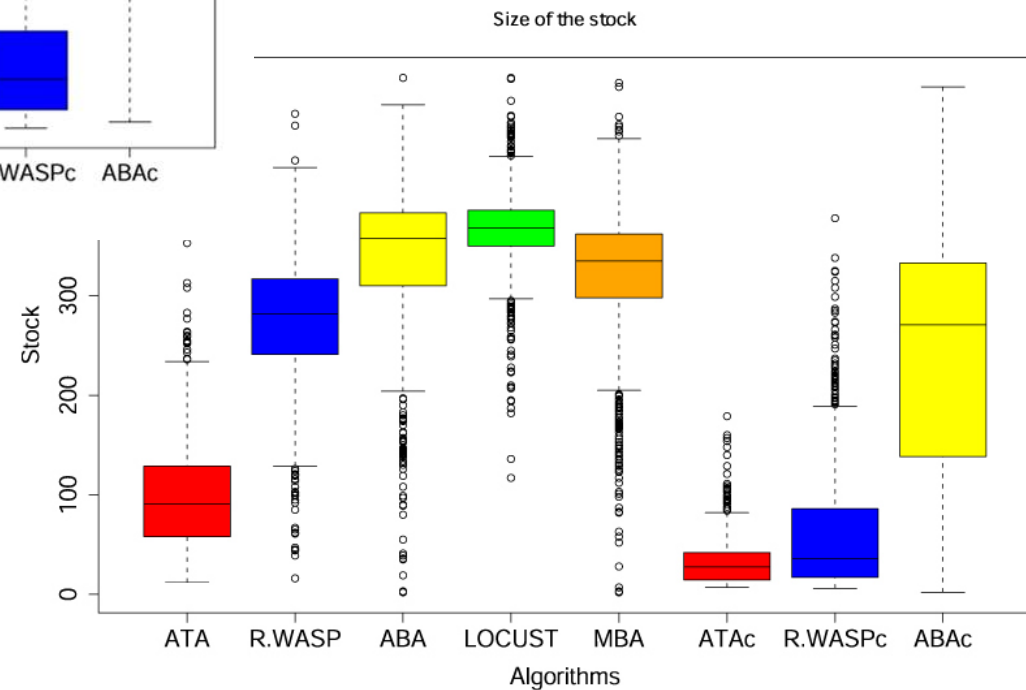
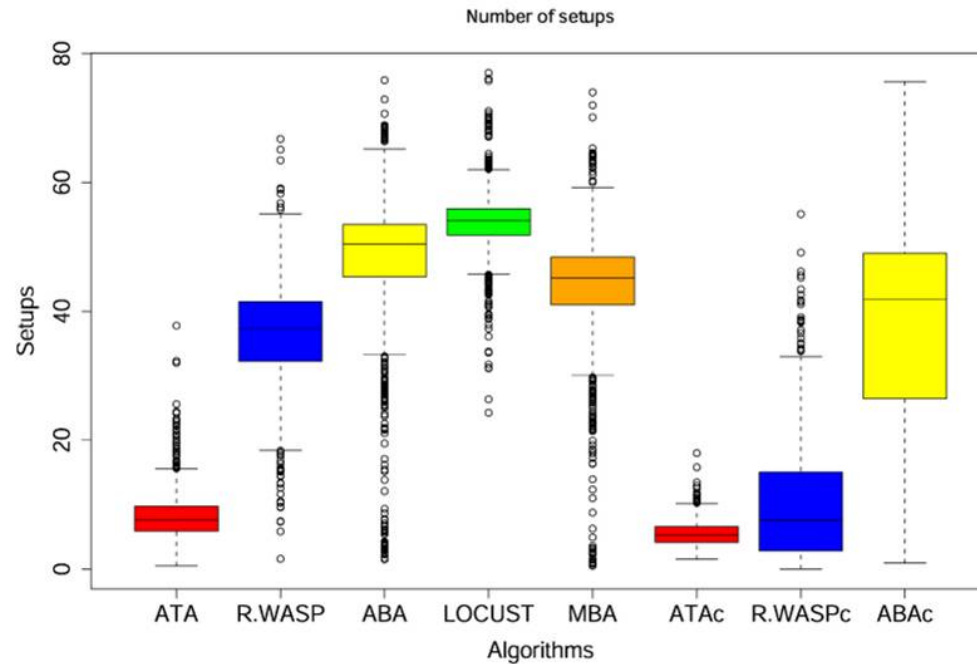
Risultati caso eterogeneo



La regola che tiene conto della velocità di processo delle macchine ed ispirata alla suddivisione del lavoro in caste tra insetti migliora considerevolmente le prestazioni degli algoritmi insect-based nella risoluzione del problema eterogeneo.



Risultati caso eterogeneo



Conclusioni & Lavoro futuro

CONCLUSIONI

- Sistema plastico ai cambiamenti sperimentali
- L'algoritmo proposto ottiene i migliori risultati rispetto agli altri confrontati
- Il contributo delle regole introdotte al makespan è positivo

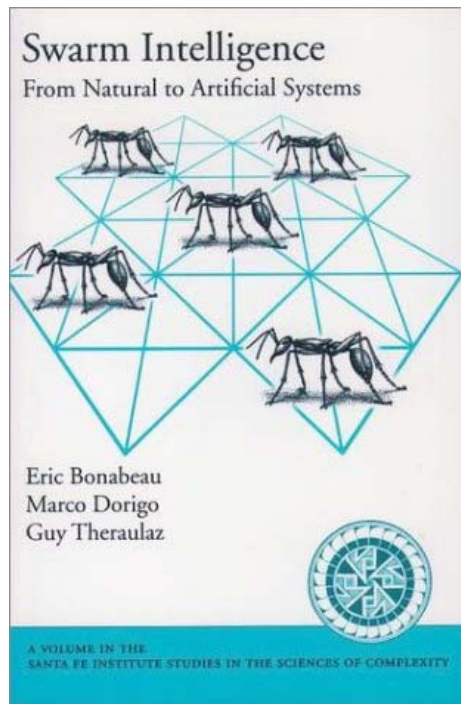
LAVORO FUTURO

- Migliorare le prestazioni dell'algoritmo (contesto dominante...)
- Estendere il problema (preemption, tempi di setup diversi, priorità delle attività...)
- Confrontarlo con altri approcci



Riferimenti

Bibliografia:



Bonabeau, Dorigo and Theraulaz
Oxford University Press, 1999

Table of contents

1. Introduction
2. Foraging and optimization
3. Division of labor and task allocation
4. Clustering, sorting and applications
5. Template mechanisms and self-organisation
6. From collective building to self-assembly
7. Collective transport
8. Conclusions

Internet:

DTA Web Site:

<http://www.robertoghizzioli.it>

<http://iridia.ulb.ac.be/~rghizzioli/dta>

mail: rghizzioli@iridia.ulb.ac.be, mbiro@ulb.ac.be, mdorigo@ulb.ac.be

