

# Application of Harmony Search to Vehicle Routing

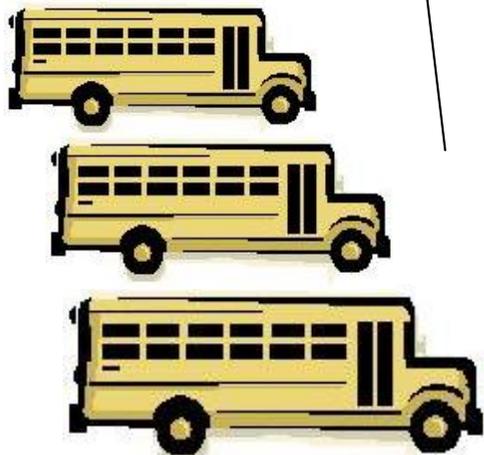
Z.W. Geem, K.S. Lee and Y. Park

American Journal of Applied Sciences (2005)

# Percorso

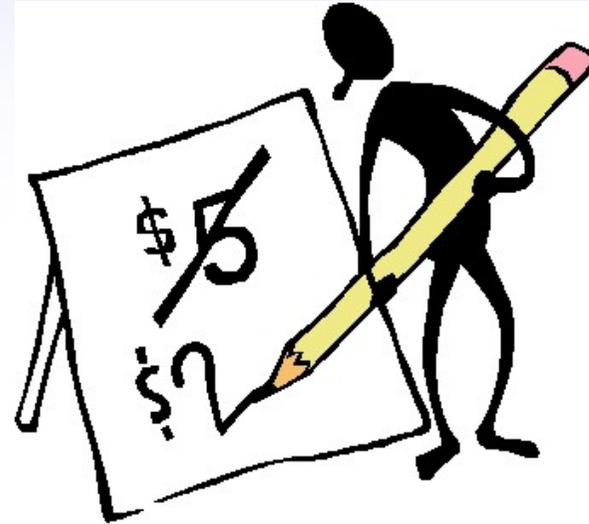
- Problema dello scuolabus
- Nuova tecnica di soft computing
- Applicazione della tecnica al problema
- Confronto dei risultati ottenuti utilizzando tecniche diverse
- Conclusioni

# School bus routing problem



# Quanto costa alla scuola?

- Costo fisso per scuolabus



- Costo del viaggio (carburante, autista, etc.) che dipende dal tempo impiegato

# OBIETTIVO

## MINIMIZZARE

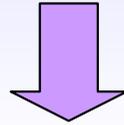
- numero di scuolabus utilizzati
- tempo totale di viaggio di tutti gli scuolabus

## VINCOLI

- capacità degli scuolabus
- tempo massimo di viaggio



I metodi esatti possono impiegare molto tempo  
per arrivare alla soluzione esatta



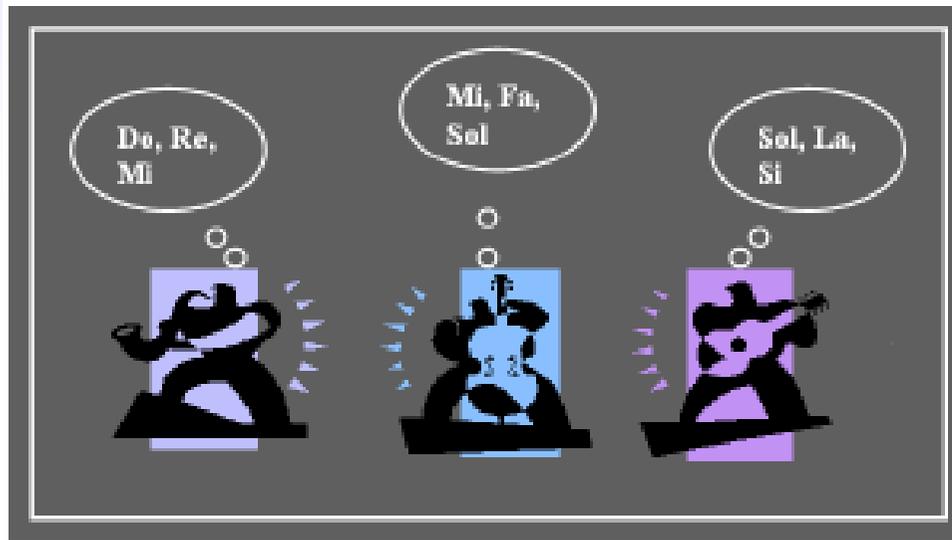
Necessità di utilizzare tecniche di soft computing

## **HARMONY SEARCH**

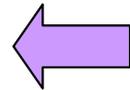
*“A new heuristic algorithm, mimicking the improvisation of music players”*

Z.W. Geem, J.H. Kim and G.V. Loganathan. *A New Heuristic Optimization Algorithm: Harmony Search*. *Simulation*, 76: 60-68 (2001)

# Improvvisazione musicale



si fanno altre prove  
finché si arriva  
all'armonia perfetta



ogni strumento ha un certo  
range di note



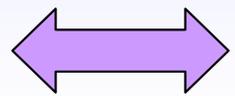
ogni musicista, partendo da  
alcune note che ha in mente,  
ne sceglie una da suonare



si valuta l'armonia ottenuta:  
se è migliore di una delle  
precedenti, la si tiene in conto;  
se no, si tralascia

# Analogia tra improvvisazione musicale e ottimizzazione

ricerca della “fantastic  
harmony” giudicata  
tramite standard estetico



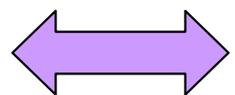
ricerca dell’ottimo  
globale determinato da  
una funzione obiettivo

l’armonia è data  
dall’insieme dei suoni di  
tutti gli strumenti



la funzione obiettivo  
dipende dai valori di  
tutte le variabili

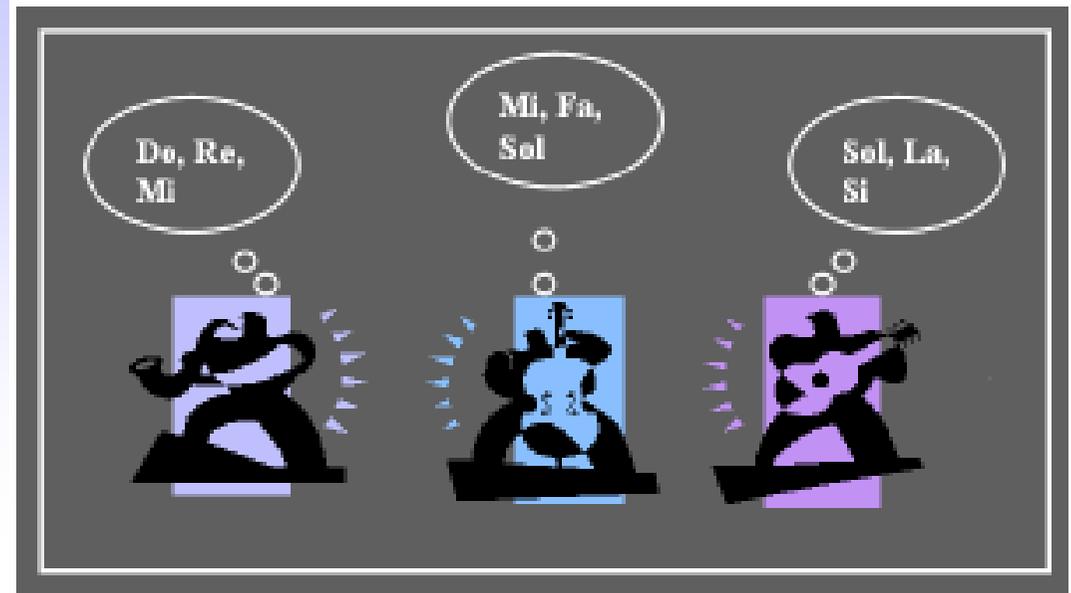
la qualità estetica del suono  
può essere migliorata con  
tanta pratica



la funzione obiettivo può  
essere migliorata iterazione  
dopo iterazione

# ALGORITMO

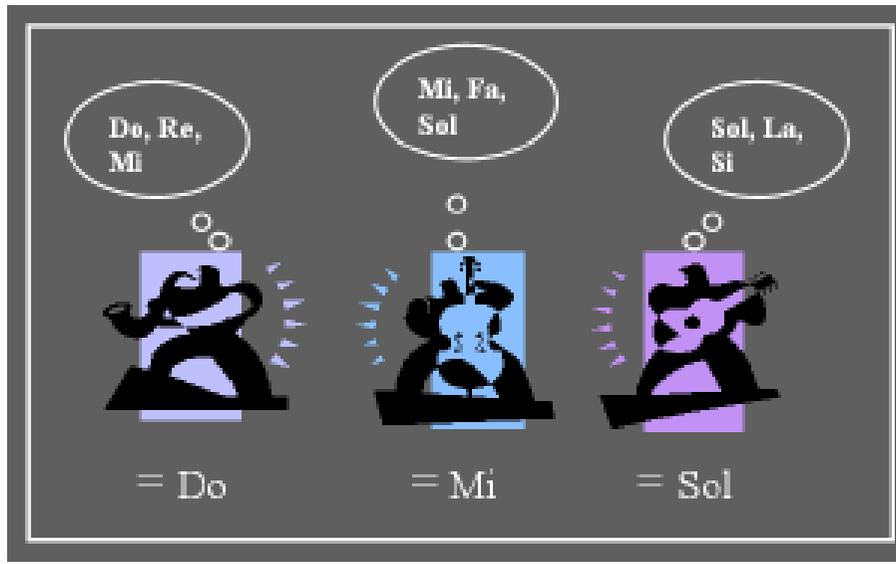
1. Inizializzare la Harmony Memory (HM) con le note che ciascun musicista ha in mente



	sassofono	contrabbasso	chitarra	
armonia 1	DO	MI	SOL	eccellente
armonia 2	RE	FA	LA	buono
armonia 3	MI	SOL	SI	sufficiente

## 2. Improvvisazione

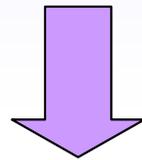
- ogni musicista suona una nota presente nell'HM
- ogni nota della HM ha la stessa probabilità di essere scelta (in questo caso è del 33%)



- otteniamo una nuova armonia (DO-MI-SOL)

### 3. Valutazione estetica

se la nuova armonia è migliore di una delle armonie presenti nell'Harmony Memory (per es. dell'armonia 3)

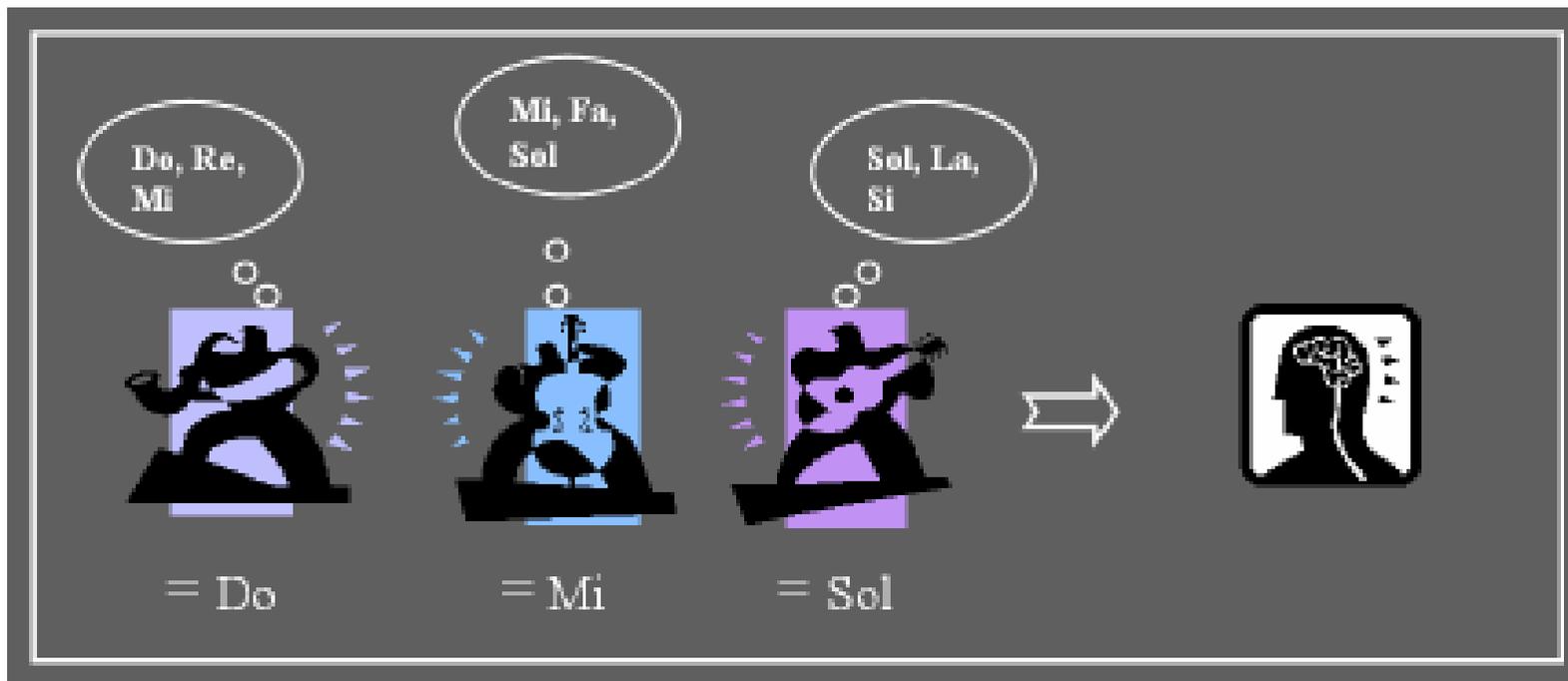


la nuova armonia (DO-MI-SOL) viene inclusa nella HM e viene eliminata l'armonia peggiore

	sassofono	contrabbasso	chitarra	
armonia 1	DO	MI	SOL	eccellente
armonia 2	RE	FA	LA	buono
armonia 3	DO	MI	SOL	sufficiente

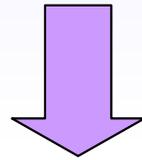
# 4. Iterazione

- il procedimento viene ripetuto finché si raggiunge un risultato soddisfacente (vicino all'ottimo)



## OSSERVAZIONE 1:

con questo procedimento, l'ottimo globale può essere raggiunto solo se le sue componenti sono tutte nella HM

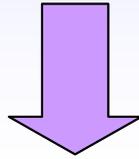


### Harmony Memory Considering Rate (HMCR)

- Il parametro HMCR in  $[0,1]$  dice con quale probabilità scegliere le note nell'HM.
- Ad ogni iterazione viene generato un valore  $T$  in  $[0,1]$  con distribuzione uniforme.
- Se  $T > \text{HMCR}$ , allora si potranno scegliere le note tra tutte quelle possibili, senza tener conto dell'HM.

## OSSERVAZIONE 2:

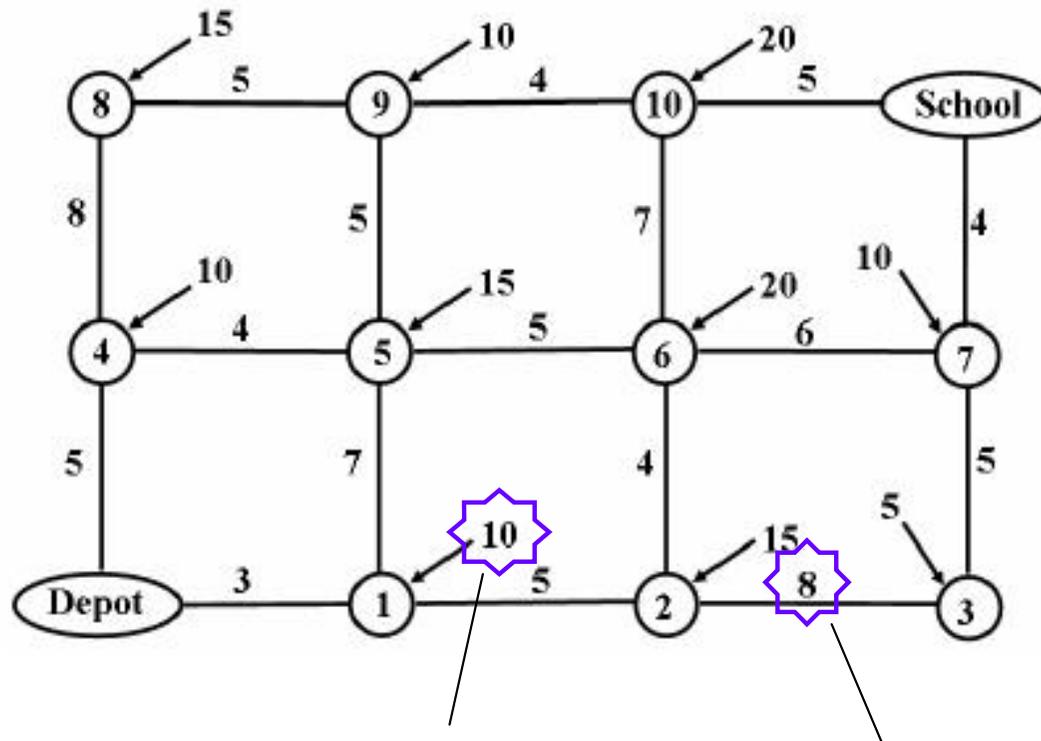
c'è il rischio di restare intrappolati in minimi locali



### Pitch Adjusting Rate (PAR)

- Il parametro PAR in  $[0,1]$  indica con quale probabilità si deve passare dalla nota precedente ad una ad essa contigua.
- C'è uguale probabilità di passare alla nota più bassa o a quella più alta. Questa probabilità è pari alla metà di PAR.

# Problema dello scuolabus



- un deposito degli autobus
- una scuola
- 10 fermate dell'autobus

ogni fermata è richiesta da un certo numero di studenti

il tempo necessario per spostarsi da una fermata all'altra è indicato in minuti

# Applicazione dell'Harmony Search

- ogni strumento è una fermata ( $x_1 - x_{10}$ )
- le note a disposizione di ogni strumento sono il numero di scuolabus a disposizione (1 - 4)
- ogni fermata “sceglie” uno scuolabus

	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$	$f(x)$
soluz 1	1	1	3	2	4	3	3	2	2	1	?
soluz 2	2	3	3	1	3	3	3	2	1	2	?
soluz 3	3	1	2	2	1	3	4	4	2	1	?

# parametri di costo di $\mathbf{x} = (x_1, \dots, x_{10})$

- $cf$  = costo fisso di un bus
- $nbus(\mathbf{x})$  = numero di bus operativi
- $cv$  = costo di viaggio per unità di tempo
- $T_{bus}^k = \sum_i \sum_j sp_{ij} \delta_{ij} =$  tempo impiegato dal bus  $k$

con  $sp_{ij}$  = cammino più breve (in minuti) tra le fermate  $i$  e  $j$

con  $\delta_{ij} = \begin{cases} 1 & \text{se il bus } k \text{ percorre il tratto fermata } i - \text{fermata } j \\ 0 & \text{altrimenti} \end{cases}$

# funzione costo

costo fisso dei bus

costo per tempo necessario

$$f(x) = fc \times nbus(x) + rc \times \sum_k Tbus^k +$$
$$+ pc1 \times \sum_k vcp^k + pc2 \times \sum_k vtm^k$$

costi aggiuntivi per penalità

## penalità 1: violazione capienza del bus

$$pc1 \times \sum_k vcp^k$$

$$vcp^k = \begin{cases} 1 & \text{violata capienza bus } k \\ 0 & \text{altrimenti} \end{cases}$$

pc1 = costo di penalità



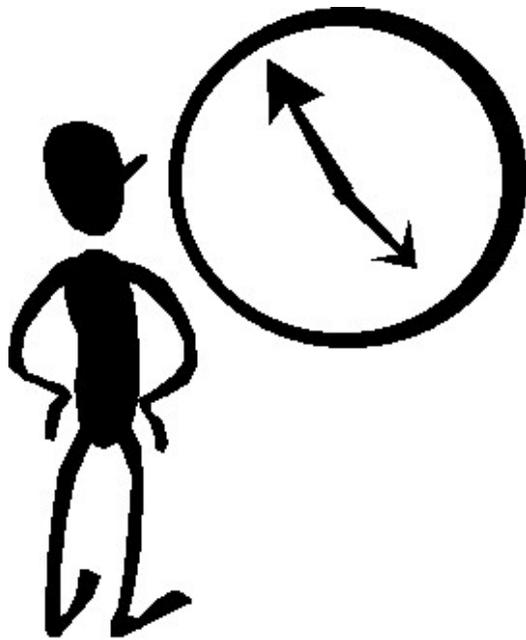
la capacità del bus  $k$  viene violata se lungo il percorso sale a bordo un numero di persone maggiore della capienza massima

## penalità 2: violazione tempo massimo

$$pc2 \times \sum_k vtm^k$$

$$vtm^k = \begin{cases} 1 & \text{violati tempi bus k} \\ 0 & \text{altrimenti} \end{cases}$$

pc2 = costo di penalità



se il tempo di viaggio + il tempo di salita è maggiore del tempo massimo concesso allo scuolabus, si incorre in una penalità

# Valori dei parametri

- costo fisso = \$ 100'000/bus
- routing cost = \$ 105/min
- penalità per la capienza = \$ 100'000/bus se salgono più di 45 studenti
- penalità per tempo = \$ 100'000 se un bus impiega più di 32 minuti per portare gli studenti a scuola
- tempo di salita per studente = 6 sec
- num iterazioni = 1'000                      HMS = 10 ~ 100
- HMCR = 0.3 ~ 0.95                      PAR = 0.01 ~ 0.05

# Soluzione ottimale

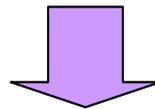
- si utilizzano 3 scuolabus con un costo totale di 307.980 \$
- sono sufficienti 1.000 iterazioni per il raggiungimento della soluzione ottima
- 1.000 iterazioni rappresentano lo 0.1% dello spazio delle soluzioni ( $4^{10} \approx 1.05 \times 10^6$ )
- occorrono 6.6 secondi su Intel 233 MHz per arrivare alla soluzione

# Soluzioni a confronto

costo servizio \$	bus	tragitto	n. studenti	tempo
HS 307'980	1	dep→8→9→10→scuola	45	31.5
	2	dep→4→5→6→scuola	45	28.5
	3	dep→1→2→3→7→scuola	40	29.0
	4	non serve	-	-
GA 410'185	1	dep→2→6→scuola	35	25.5
	2	dep→1→3→7→scuola	25	27.5
	3	dep→5→9→10→scuola	45	27.5
	4	dep→4→8→scuola	25	29.5

# Conclusioni

- l'Harmony Search è in grado di trovare l'ottimo globale con un numero di iterazioni relativamente piccolo rispetto al numero di soluzioni possibili
- il modello che utilizza HS trova soluzioni migliori di GA in termini di numero di tentativi andati a buon fine, di costo medio del servizio e di tempo di calcolo



metodo utile nei problemi di calcolo combinatorio