

# Podejście memetyczne do problemu VRP with Stochastic Demands

Adam Żychowski

Na podstawie pracy

X. S. Chen, L. Feng and Y. S. Ong

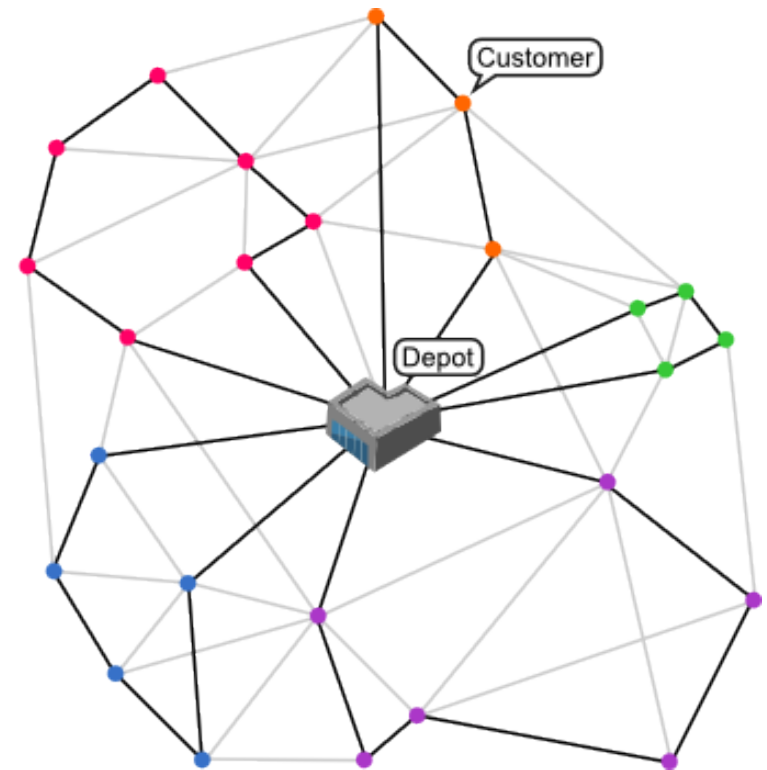
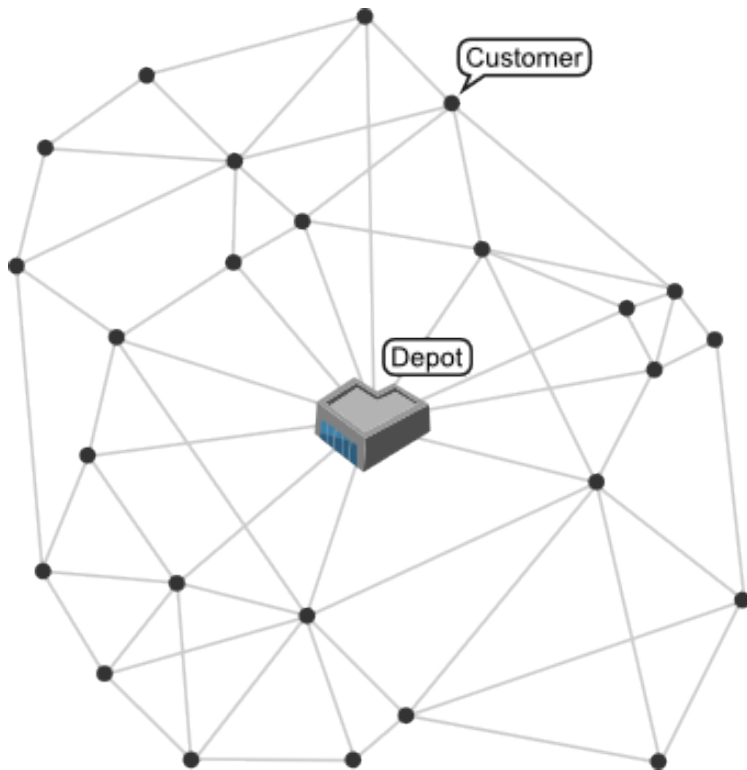
*A Self-Adaptive Memeplexes Robust Search Scheme for solving Stochastic Demands Vehicle Routing Problem*

International Journal of Systems Science,  
43:1347-1366, 2012

# Klasyczny problem marszrutyzacji

- Vehicle Routing Problem (VRP)
- $n$  klientów z zapotrzebowaniem + zajezdnia
- jednakowe ciężarówki o skończonej ładowności
- droga ciężarówki zaczyna i kończy się w zajezdni
- suma zapotrzebowań klientów, których obsługuje jedna ciężarówka nie może przekroczyć jej ładowności

Cel: rozplanowanie tras ciężarówek, aby sumaryczna przebyta odległość była minimalna



# VRP with Stochastic Demands

- niepewność związana z zapotrzebowaniem każdego klienta
- wielkość zapotrzebowania danego klienta poznajemy w momencie jego odwiedzenia
- znalezienie rozwiązania “odpornego” na zmiany wielkości zapotrzebowań przy jednoczesnej minimalizacji tras ciężarówek

# Safety stock

- bufor bezpieczeństwa

$$k \in (0, 1)$$

$$C_i = k \cdot C_v$$

- zmniejszenie prawdopodobieństwa konieczności powrotu do zajezdni z powodu braku dostępnego miejsca w ciężarówce

# Idea algorytmu

- geny reprezentują rozwiązania problemu
- lokalna optymalizacja sterowana memami
- memy przechowują fragment algorytmu lokalnej optymalizacji
- zarówno geny jak i memy podlegają mutacji i krzyżowaniu

**BEGIN**

*Create* an initial population of solutions  $\text{Pop}(\text{gen} = 0) = \{s_1, s_2, \dots, s_i, \dots, s_M\}$

**While**(stopping conditions are not satisfied)

$\text{gen} \leftarrow \text{gen} + 1$

**Perform** evolutionary operators to generate  $\text{Pop}(\text{gen})$  from  $\text{Pop}(\text{gen} - 1)$  based on  $f(s_i)$ :

**Evaluate**  $f(s_i), \forall s_i \in \text{Pop}(\text{gen})$

**Perform** Lifetime Learning

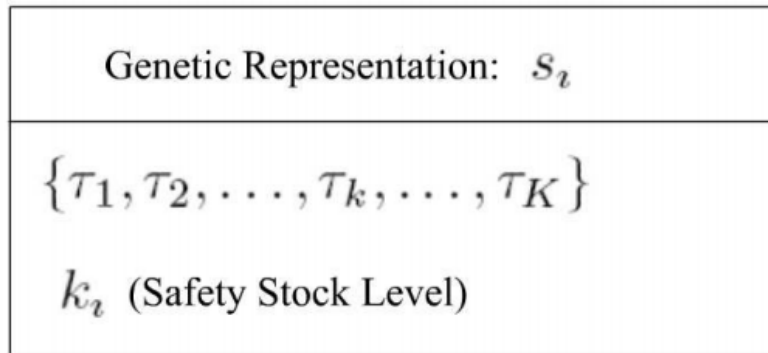
**Update** global best solution  $s_g$

**End While**

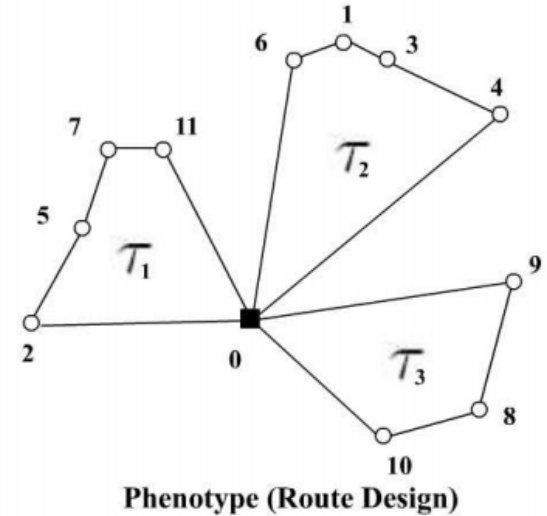
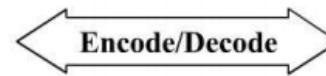
**END**



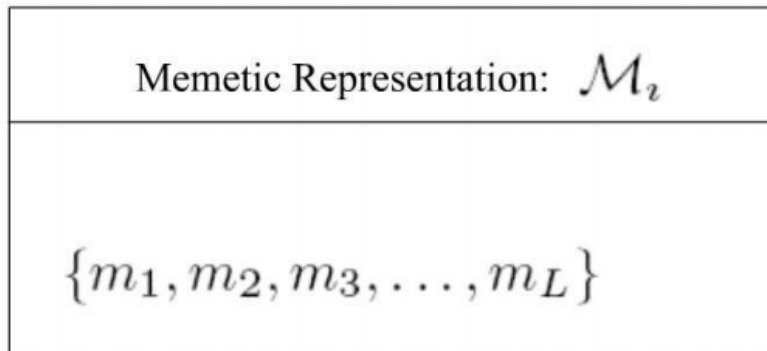
# Kodowanie genów i memów



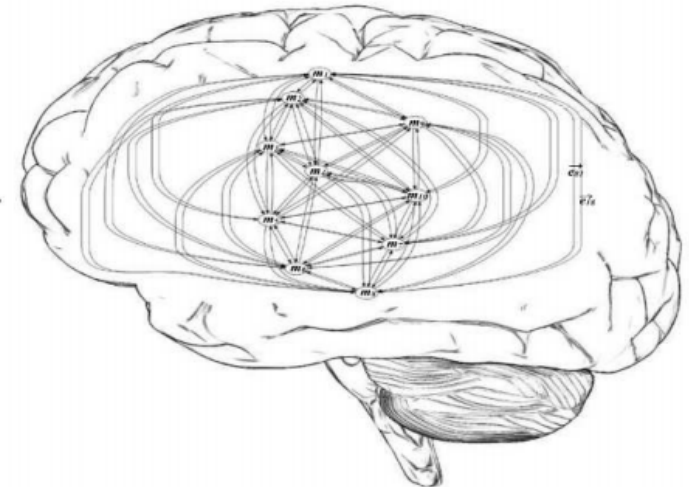
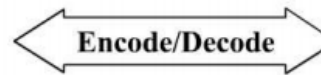
Genotype (Geneplex)



Phenotype (Route Design)



Memotype (Memeplex)



Sociotype (Search Instruction)

# Memy

Rodzaje memów:

$T_1$  - losowy klient zostaje wydzielony do osobnej trasy

$T_2$  - zamiana dwóch losowych klientów z różnych tras

$T_3$  - zamiana dwóch losowych fragmentów tras

$T_4$  - połączenie dwóch tras

$T_5$  - przeniesienie losowego klienta w losowe miejsce innej trasy

$T_6$  - odwrócenie kolejności odwiedzania klientów na fragmencie trasy

# Memy

3 składowe każdego memu:

- $T_i$  - rodzaj operacji
- strategia akceptacji (pierwsza poprawa lub najlepsza poprawa)
- głębokość przeszukiwań

## Macierz synerгии

$$W_s = \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1L} \\ w_{21} & w_{22} & \cdots & w_{2L} \\ \vdots & \vdots & \ddots & \vdots \\ w_{L1} & w_{L2} & \cdots & w_{LL} \end{pmatrix}$$

$w_{ij}$  reprezentuje pewną relację między memem  $m_i$  oraz  $m_j$

# Algorytm optymalizacji lokalnej

**BEGIN**

$\Delta C(\mathcal{M}_i) \leftarrow 0$

$\Delta T(\mathcal{M}_i) \leftarrow 0$

$m_{prev} \leftarrow \emptyset$

**While TRUE**

**Select** meme  $m_j \in \mathcal{M}_i$  with activation probability  $P_{activate}(m_j, m_{prev})$

$(\Delta C_j, \Delta T_j) \leftarrow$  **Improve**  $s_i$  by  $m_i$  in the context of  $P_i$

**Assign Credit** to meme pair  $(m_i, m_{prev})$  based on feedback  $(\Delta C_j, \Delta T_j)$

$\Delta C(\mathcal{M}_i)+ = \Delta C_j$

$\Delta T(\mathcal{M}_i)+ = \Delta T_j$

**If**  $\Delta C_j == 0$  **then**

**If**  $\text{rand}(0, 1) > \exp(-\frac{\Delta T(\mathcal{M}_i)}{\Delta C(\mathcal{M}_i)})$  **then**

**Break**

**End If**

**End If**

$m_{prev} \leftarrow m_j$

**End While**

**END**

$$P_{activate}(m_j, m_i) = \begin{cases} \frac{w_{ij}}{\sum_{\mu=1}^L w_{i\mu}} & m_i \neq \emptyset \\ \frac{w_{jj}}{\sum_{\mu=1}^L w_{\mu\mu}} & \text{otherwise} \end{cases}$$

Aktualizacja macierzy po realizacji memu  $m_j$  według wzoru:

$$\begin{aligned}w_{ij} &= \gamma w_{ij} + r_j \\ &= \gamma w_{ij} + \frac{\Delta C_j}{\Delta T_j}\end{aligned}$$

$0 < \gamma < 1$  - współczynnik znaczenia wydajności memu w czasie

# Funkcja przystosowania

- założenie dotyczące rozkładu gęstości prawdopodobieństwa - rozkład logarytmiczno-normalny

$$PDF_X(x, \mu_i, \sigma_i^2) = \frac{1}{x\sigma_i\sqrt{2\pi}} e^{-\frac{(\ln x - \mu_i)^2}{2\sigma_i^2}}, X = \{x | x = demand(i), 1 \leq i \leq N\}$$

$$\mu_i = \ln(E[demand(i)]) - \frac{1}{2} \ln\left(1 + \frac{\text{Var}[demand(i)]}{E[demand(i)]^2}\right),$$

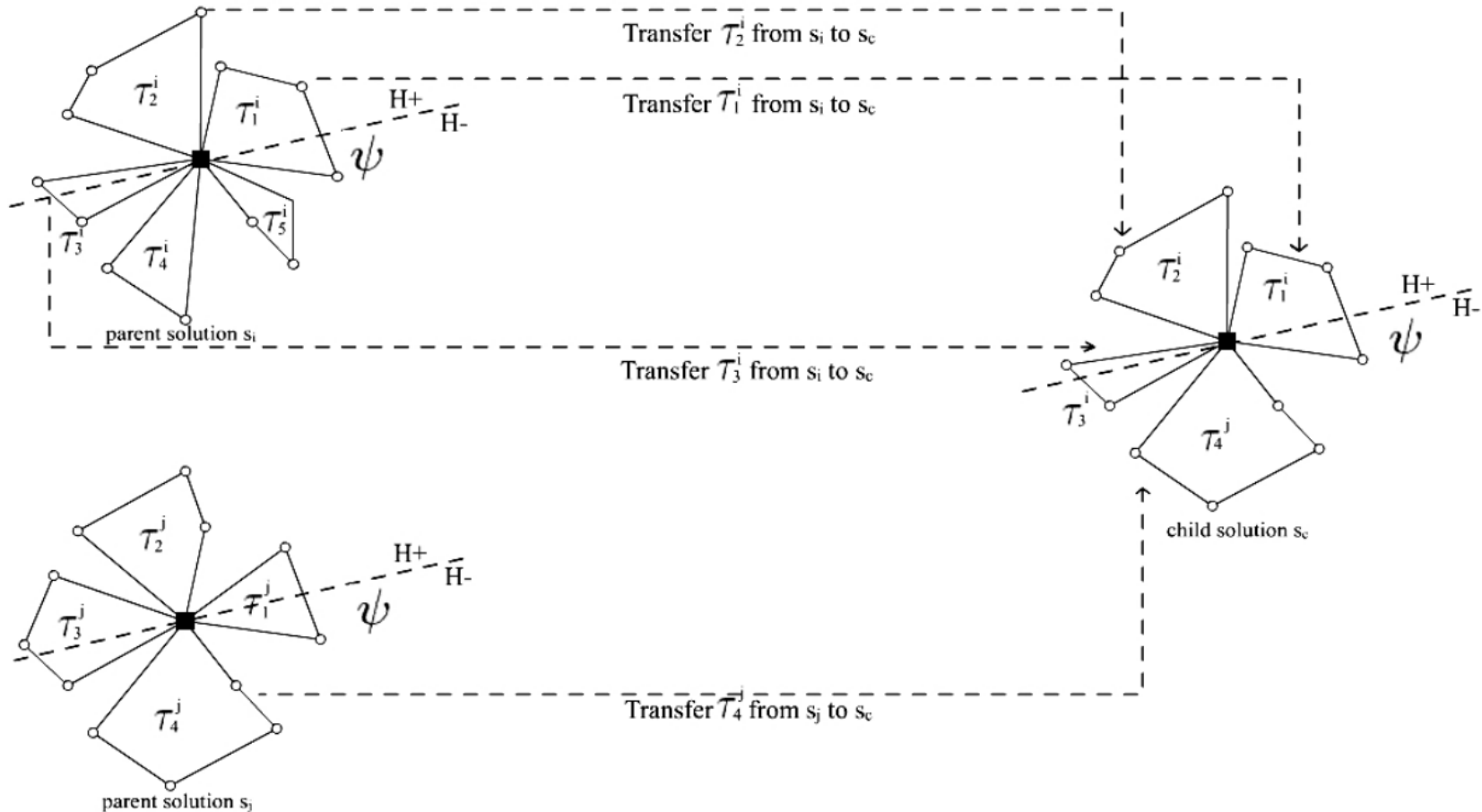
$$\sigma_i^2 = \ln\left(1 + \frac{\text{Var}[demand(i)]}{E[demand(i)]^2}\right).$$

# Funkcja przystosowania

- wartość funkcji przystosowania = średnia dystansów z kilku instancji problemów
- symulowanie jazdy ciężarówek z uwzględnieniem przypadków niewystarczającej ładowności i konieczności powrotu do zajezdni
- “uodpornienie” rozwiązania na różne “zachowania” klientów



# Krzyżowanie



$$k_c = k_i + t \cdot (k_j - k_i), t \in (0, 1)$$

# Krzyżowanie memów

- w macierzy relacji memów dziecka w każdym miejscu miejscu losowo wartość z macierzy jednego lub drugiego rodzica

# Mutacja

geny: wybierany losowo rodzaj mutacji zamiana dwóch klientów miejscami, odwrócenie kolejności odwiedzania klientów, przemieszczenie klientów między trasami

memy: do każdej liczby w macierzy relacji memów dodawana jest losowa niewielka wartość wylosowana z rozkładem normalnym

# Parametry algorytmu

wielkość populacji: 30

liczba iteracji: 50 000

prawdopodobieństwo mutacji: 0,5

prawdopodobieństwo krzyżowania: 0,5

selekcja ruletkowa

elita - 2 osobniki

Dziękuję za uwagę