Genetic Algorithms with Immigrants and Memory Schemes for Dynamic Shortest Path Routing Problems in Mobile Ad Hoc Networks

Hui Cheng
Dept. of Computer Science
University of Leicester
United Kingdom
09/06/2009

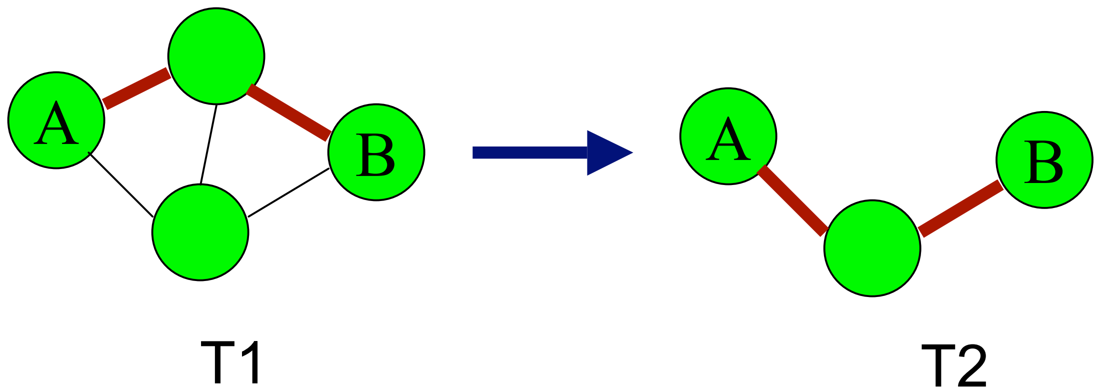
Outline

- Introduction
- Related Work
- The Dynamic Shortest Path Routing Problem
- Specialized GA for the Problem
- Investigated GAs for the Problem
  - Traditional GA Schemes
  - Dynamic GA Schemes
- Experimental Study
- Conclusions

Introduction

- With the advancement in wireless communications, more and more mobile wireless networks emerge, e.g., mobile ad hoc networks (MANETs), wireless mesh networks (WMNs), etc.
- An important characteristic in mobile wireless networks is the ***topology dynamics***, that is, the network topology changes over time due to energy conservation or node mobility.

- Mobile Ad Hoc Network
  - A MANET is a wireless network set up temporarily without a wired infrastructure (routers, switches, servers, cables, access points, etc.).
  - It is very suitable for disaster rescue and recovery, battlefield communication, etc.
  - Node mobility -> Topology change



- Therefore, in MANETs, the shortest path (SP) routing problem turns out to be a dynamic optimization problem (DOP).
- We propose to use GAs with immigrants and memory schemes to solve the dynamic shortest path routing problem in MANETs.

Related Work

- Quite a few research work have been done to solve the SP problems using artificial intelligence techniques:
  - artificial neural networks (Electron. Lett., 2001)
  - GAs (IEEE TEC 2002)
  - particle swarm optimization (Appl. Soft Comput., 2008)
- However, all these algorithms still address the static SP problem only.

Problem Model

- We consider a mobile ad hoc network operating within a fixed geographical region. We model it by a undirected and connected topology graph $G_0(V_0, E_0)$, where V_0 represents the set of wireless nodes (i.e., routers) and E_0 represents the set of communication links connecting two neighboring routers falling into the radio transmission range.
- Message transmission on a wireless communication link will incur remarkable delay and cost.

Notations

- $G_0(V_0, E_0)$, the initial MANET topology graph.
- $G_i(V_i, E_i)$, the MANET topology graph after the *i*th change.
- s, the source node.
- r, the destination node.
- $P_i(s, r)$, a path from s to r on the graph G_i.
- d_l, the transmission delay on the communication link l.
- c_l, the cost on the communication link l.
- $\Delta(P_i)$, the total transmission delay on the path P_i.
- $C(P_\mathrm{i})$, the total cost of the path P_i.

Problem Formulation

- More formally, consider a mobile ad hoc network $G(V, E)$ and a unicast communication request from the source node s to the destination node r with the delay upper bound Δ.
- The dynamic delay-constrained shortest path problem is to find a series of paths $\{P_i | i \in \{0, 1, \ldots\}\}$ over a series of graphs $\{G_i | i \in \{0, 1, \ldots\}\}$, which satisfy the delay constraint as shown in (1) and have the least path cost as shown in (2).

$$\Delta(P_i) = \sum_{l \in P_i(s,r)} d_l \leq \Delta \,. \qquad (1)$$

$$C(P_i) = \min_{P \in G_i} \left\{ \sum_{l \in P(s,r)} c_l \right\} . \qquad (2)$$

Specialized GA for The SP Problem

- Genetic Representation
  - A routing path is encoded by a string of positive integers that represent the IDs of nodes through which the path passes.
- Population Initialization: random initialization
- Fitness Function
  - the less the path cost, the better

$$F(Ch_i) = [\sum_{l \in P(s,r)} c_l]^{-1} \, . \qquad (3)$$

- Selection: Pair-wise Tournament
- Crossover
  - With the crossover probability, each time we select two chromosomes Ch_i and Ch_j for crossover.
  - Ch_i and Ch_j should possess at least one common node. Among all the common nodes, one node, denoted as *v*, is randomly selected. Then exchange the two subpaths from *v* to *r*.
- Mutation
  - With the mutation probability, each time we select one chromosome Ch_i on which one gene is randomly selected as the mutation point (i.e., mutation node), denoted as *v*. The mutation will replace the subpath from *v* to *r* by a new random subpath.

Investigated GAs for The Problem

- Traditional Schemes
  - Standard GA: handling infeasible solutions caused by environmental changes by penalty.
  - Restart GA: once a change is detected, the population will be re-initialized based on the new environment.

- Immigrants Schemes
  - RIGA: every generation a small number of new randomly generated individuals (i.e., random immigrants) are introduced to the population to replace the worst ones.
  - EIGA: every generation a small number of new individuals (mutated from the elitism) are introduced to the population.
  - HIGA: every generation both of random immigrants and elitism-based immigrants are introduced.

Memory Related Schemes

- Memory Schemes
  - MEGA: memory-enhanced GA, to store useful information in the current environment for possible reuse in the new environment
- Memory and Immigrants Hybridized Schemes
  - MRIGA: memory and random immigrants
  - MIGA: memory-based immigrants
- By the design principle, memory related schemes are suitable for cyclic dynamic environments.

Experimental Study

- We implement the following eight algorithms
- Traditional GAs
  - Restart GA
  - SGA (Standard GA)
- Dynamic GAs
  - RIGA
  - EIGA
  - HIGA
  - MEGA
  - MIGA
  - MRIGA

Dynamic Test Environments

- Initial network topology
  - 100 network nodes are randomly distributed in 200*200 area. If the distance between two nodes falls into the radio transmission range, a link will be added to connect them.
- All the algorithms start from the initial topology. Then after a certain number (saying, *R*) of generations (i.e., the change interval), a certain number (saying, *M*) of nodes are scheduled to sleep or wake up depending on their current status.

Environmental Change Setting

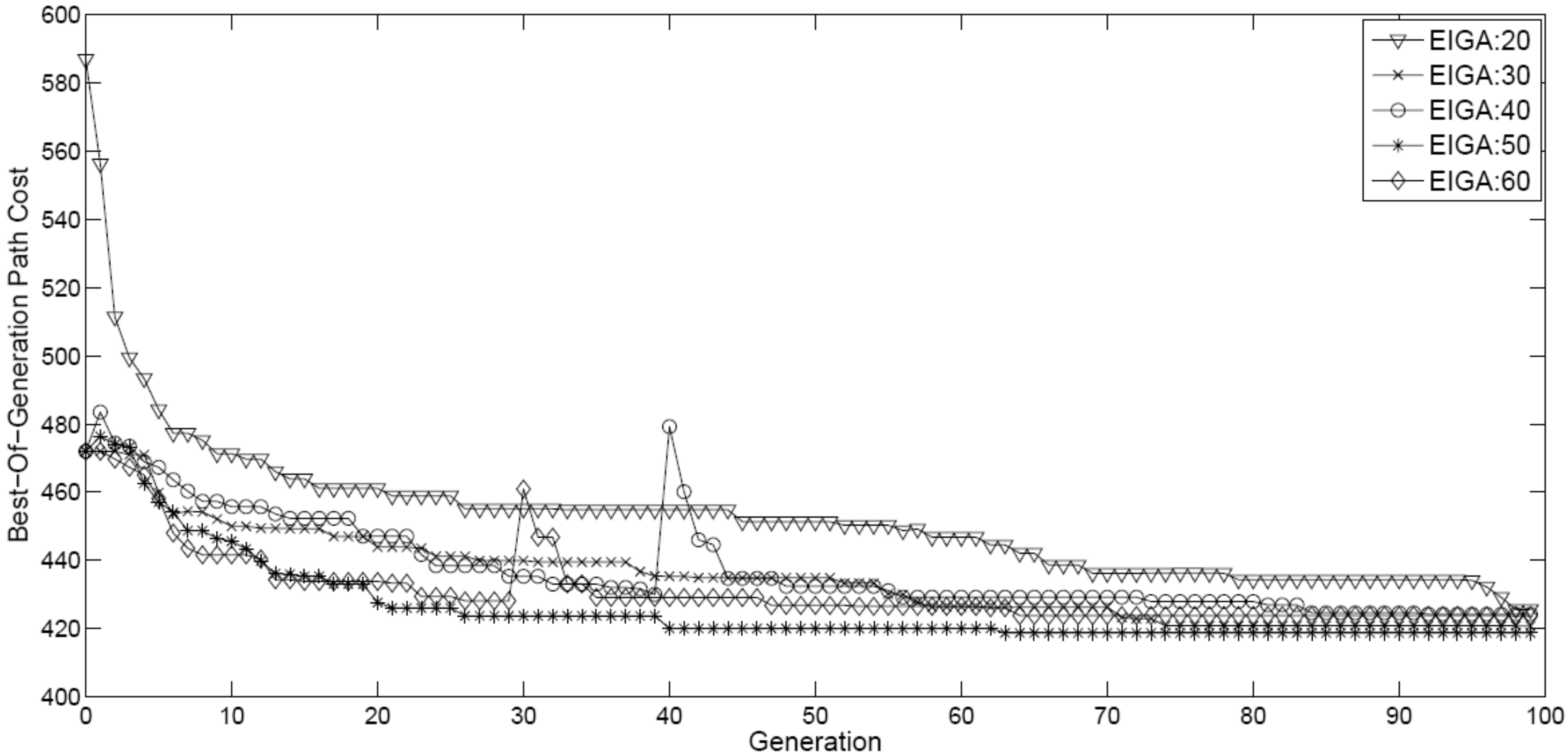
- Every R generations, randomly select M node to sleep or wake up
  - Therefore, every R generations, an updated network topology is read into the algorithms.
- $R = 5, 10, 15$; $M = 2, 3, 4$
- We generate 4 topology series
  - $M = 2, 3, 4$: topology series #2, #3, #4; acyclic environment, 21 topologies
  - $M = 1$: topology series #1; cyclic environment, 101 topologies (repeat 5 times)

Parameter Setting

- The ratio of random immigrants: 0.2
- The ratio of elitism-based immigrants: 0.2
- The ratios of hybrid immigrants: 0.1 for RI and 0.1 for EI
- Repeat 10 times.

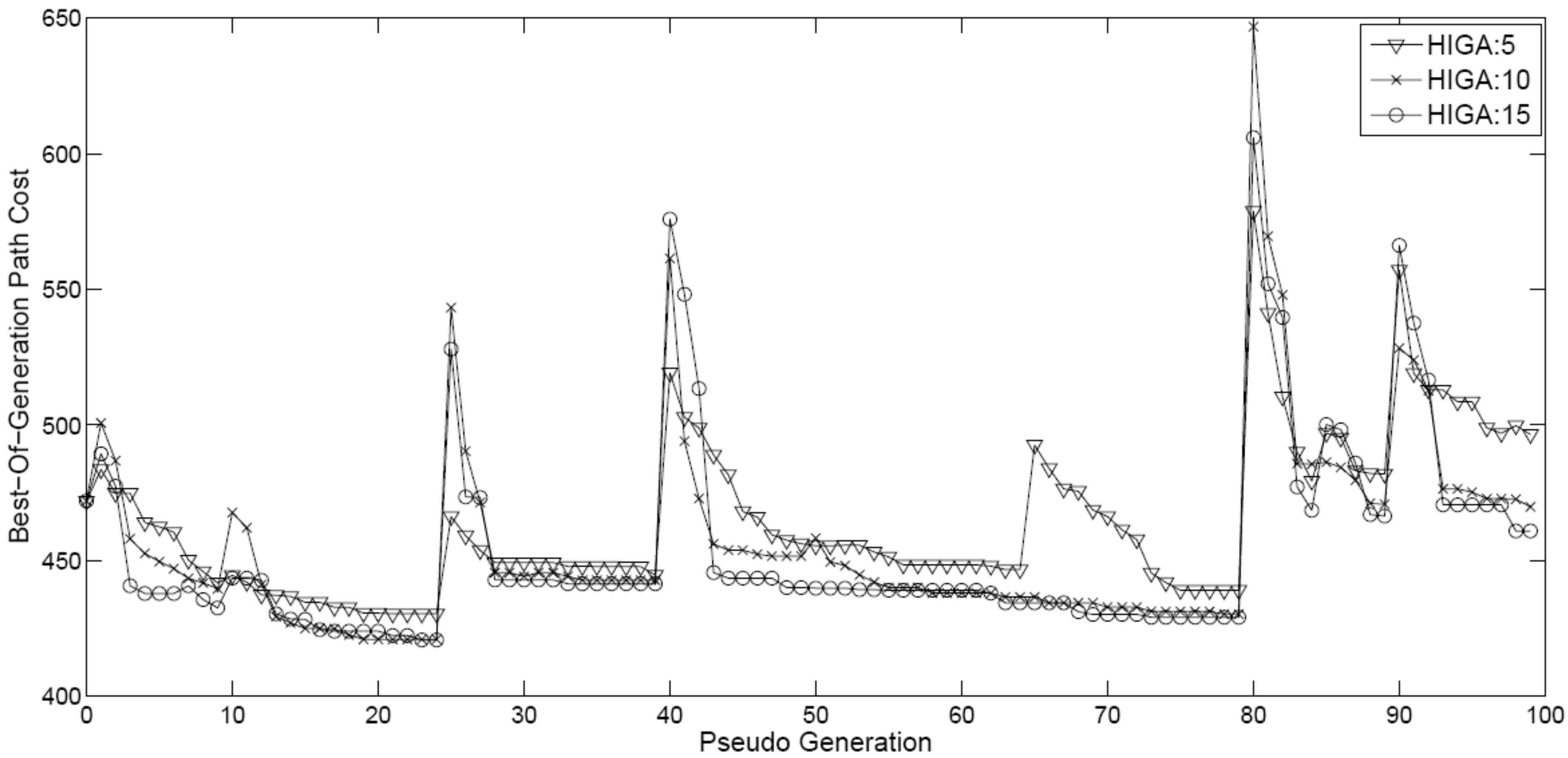
Basic Experimental Results

- Population Size: 50 is ideal



The Change Interval: 5, 10, 15

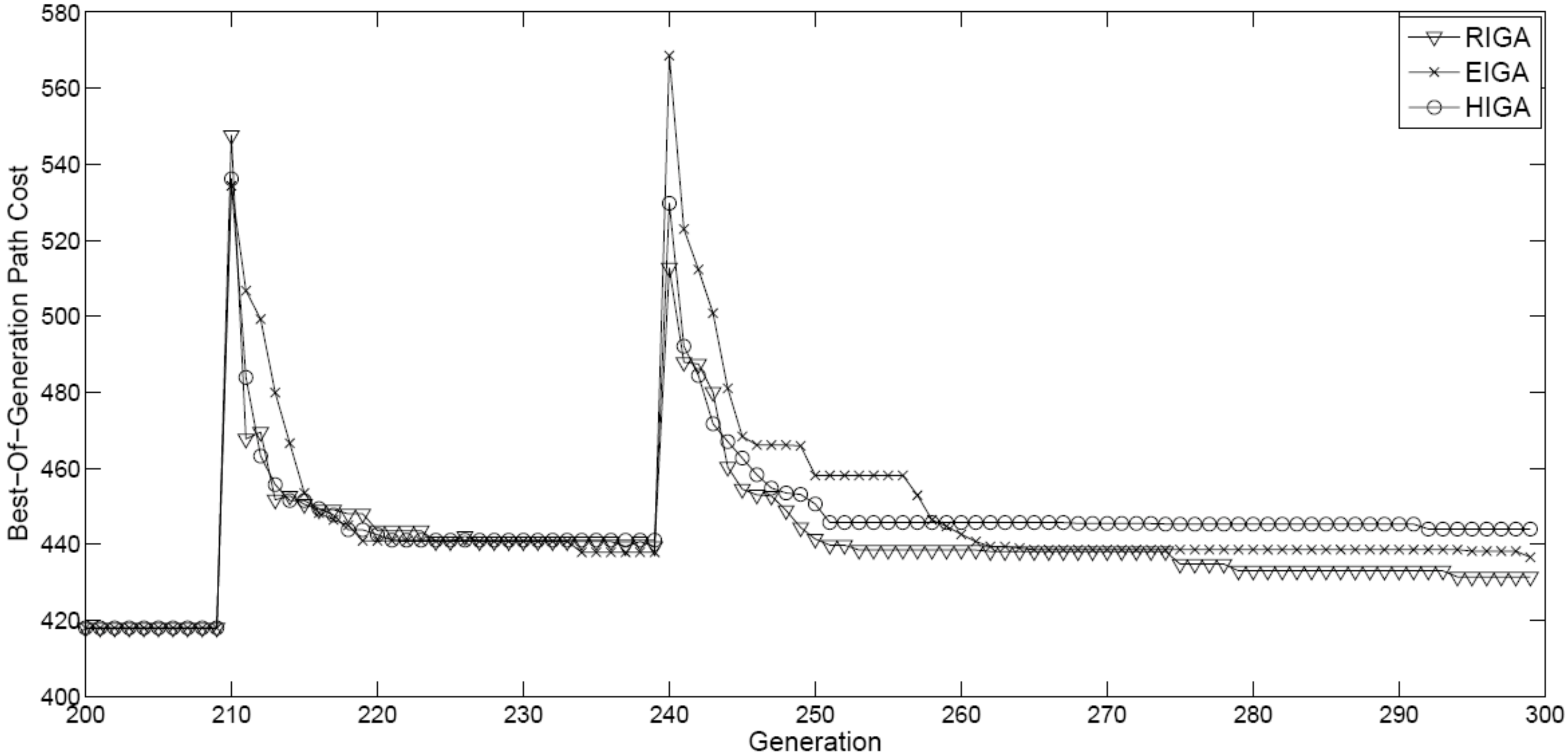
- To investigate the impact of the change interval on the algorithm performance.
- However, one problem is that the total generations are different for different intervals, i.e., 105, 210 and 315 versus the interval 5, 10, and 15.
- Since the number of change points is the same, we take the data at each change point and its left two and right two generations.



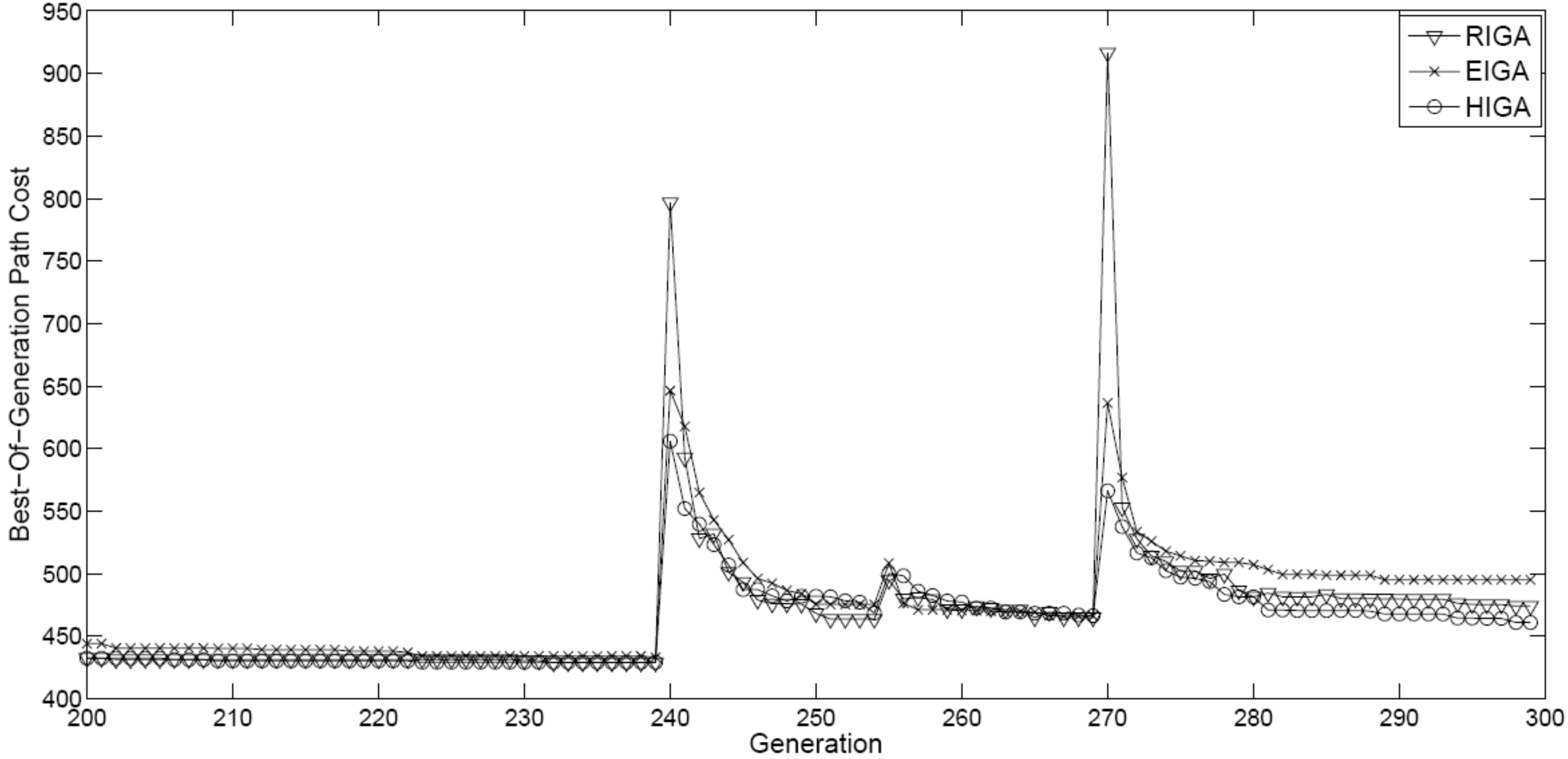
HIGA:5
HIGA:10
HIGA:15
650
600
550
500
450
400
0
10
20
30
40
50
60
70
80
90
100
Best-Of-Generation Path Cost
Pseudo Generation

The Change Severity

- To investigate the impact of the change severity on the algorithm performance, i.e., the respond speed.
- The change severity is reflected by the number of nodes involved per change.
- We choose topology series #2 and #4 as the two environments with different change severity.
- We pick up RIGA, EIGA, and HIGA together as the examples.



(a) Over topology series #2

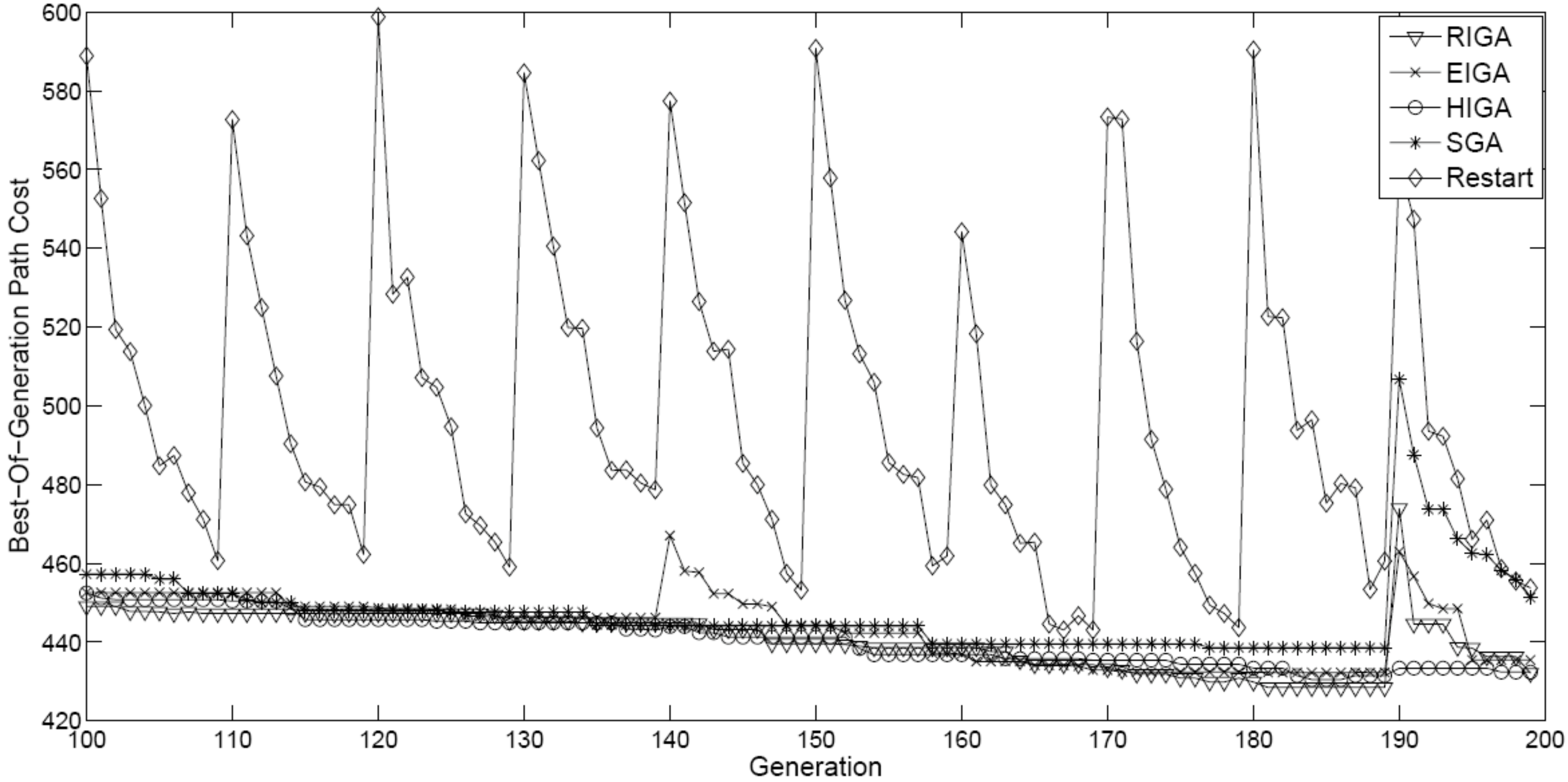


(b) Over topology series #4

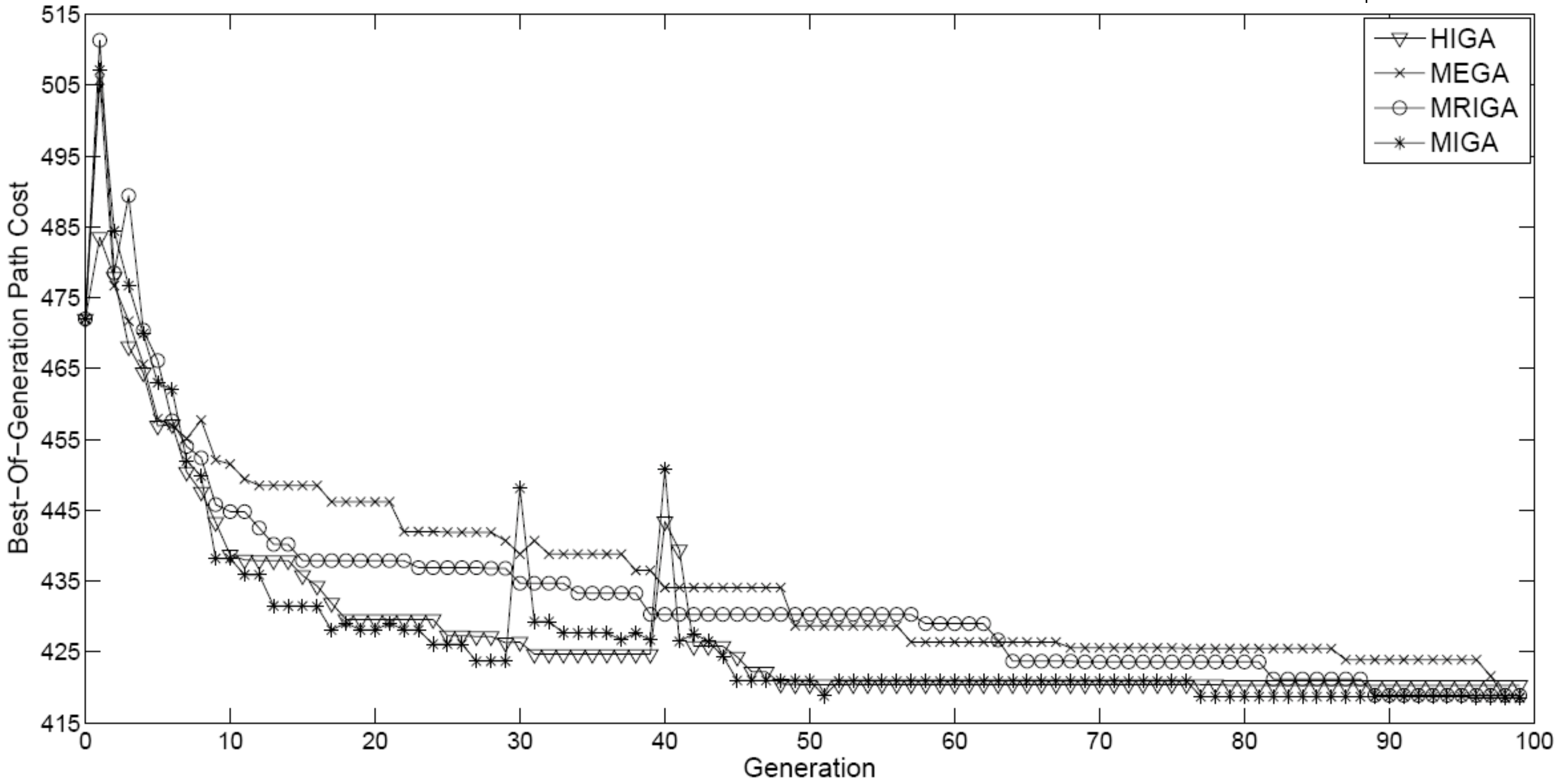
- In (a), there are two drastic change points. In (b), there are three.
- We count the number of the generations that the population spends to find the best solution before next change occurs.
  - In (a), the average value: 10.67
  - In (b), the average value: 12.67
  - In average, two more generations are spent since the change severity is higher in (b) than in (a).

Compare Dynamic GAs with Traditional GAs

- We compare RIGA, EIGA, and HIGA with SGA and Restart GA.

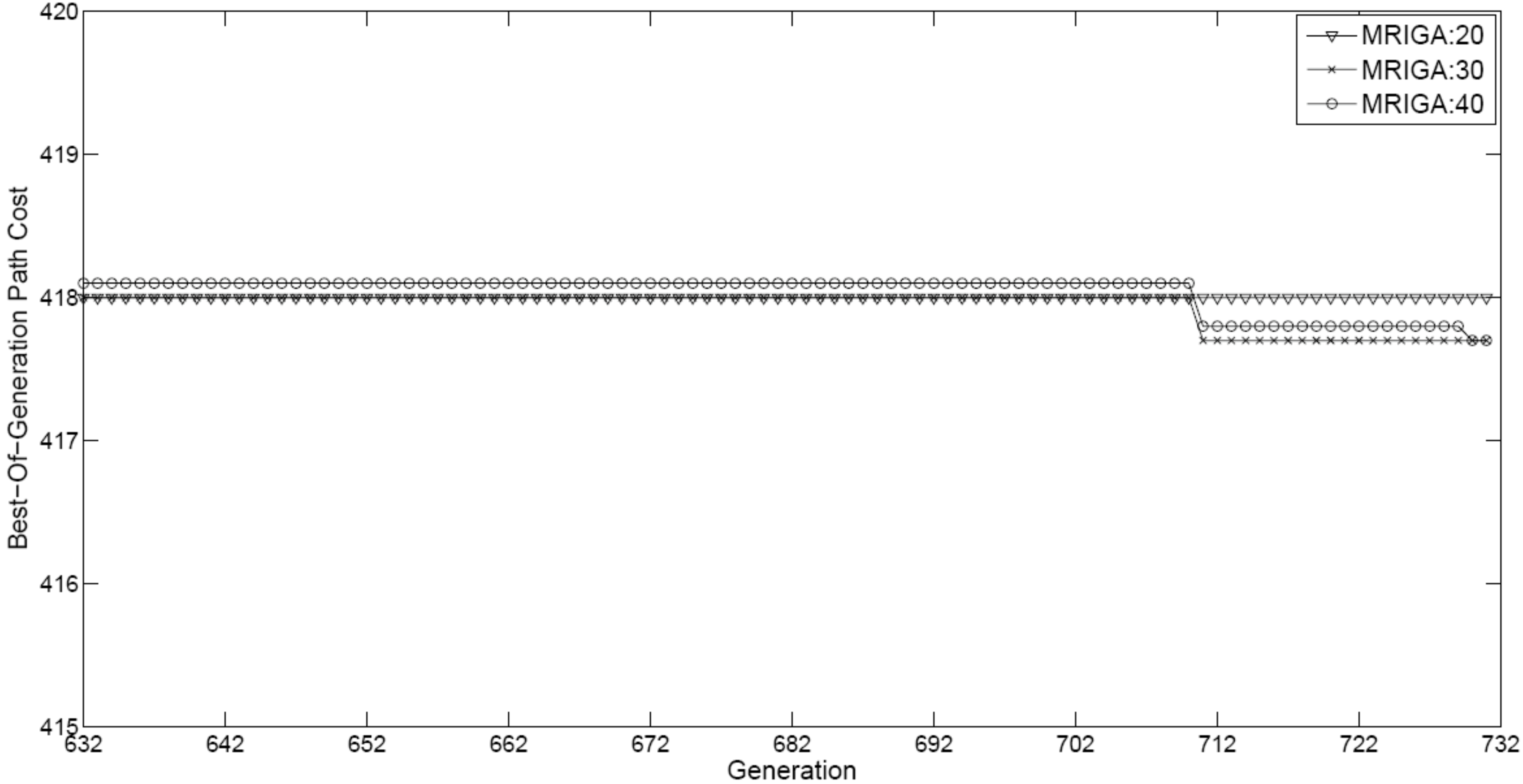


Compare Immigrants GAs with Memory related GAs in Acyclic Environments

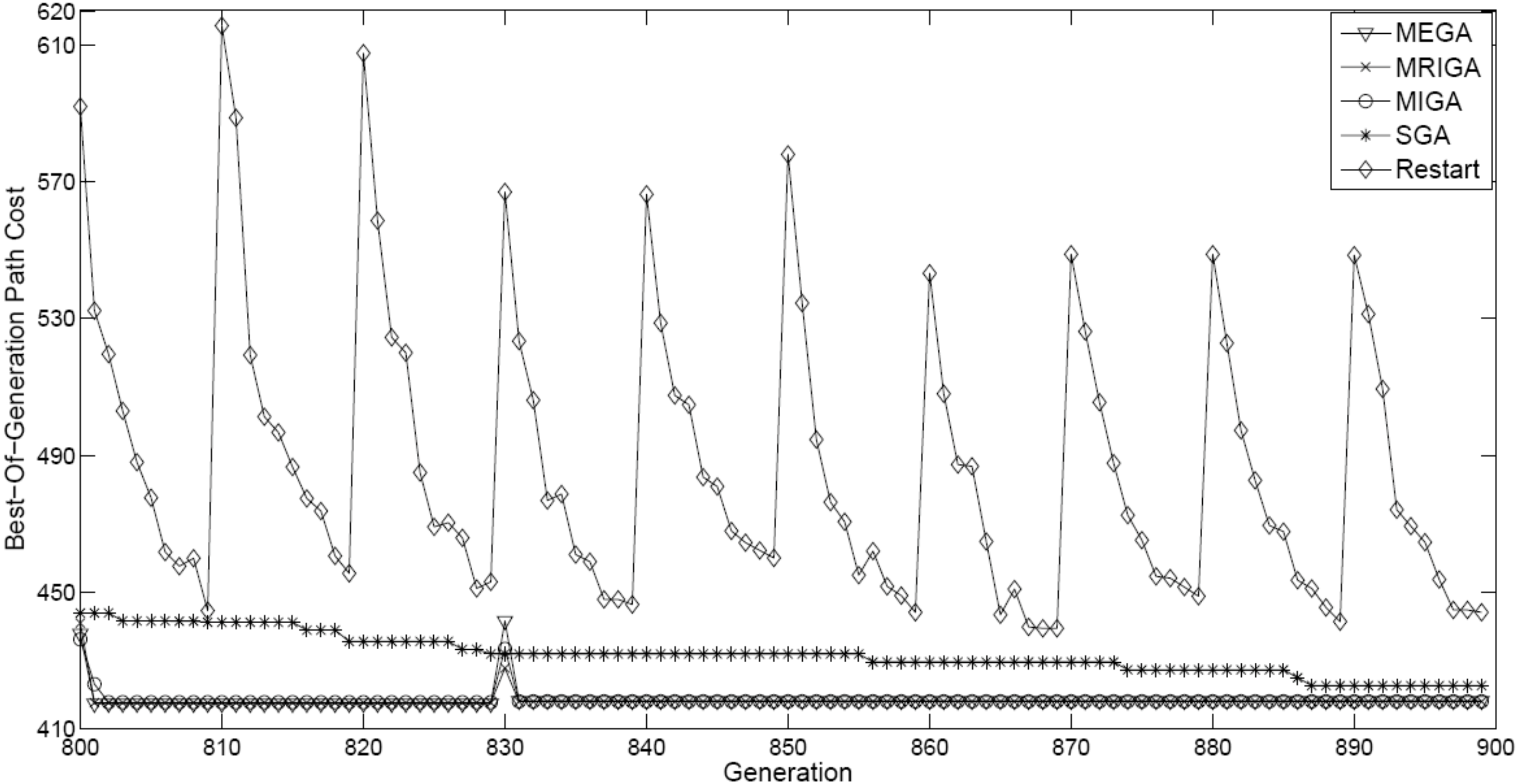


Experimental Results in Cyclic Dynamic Environments

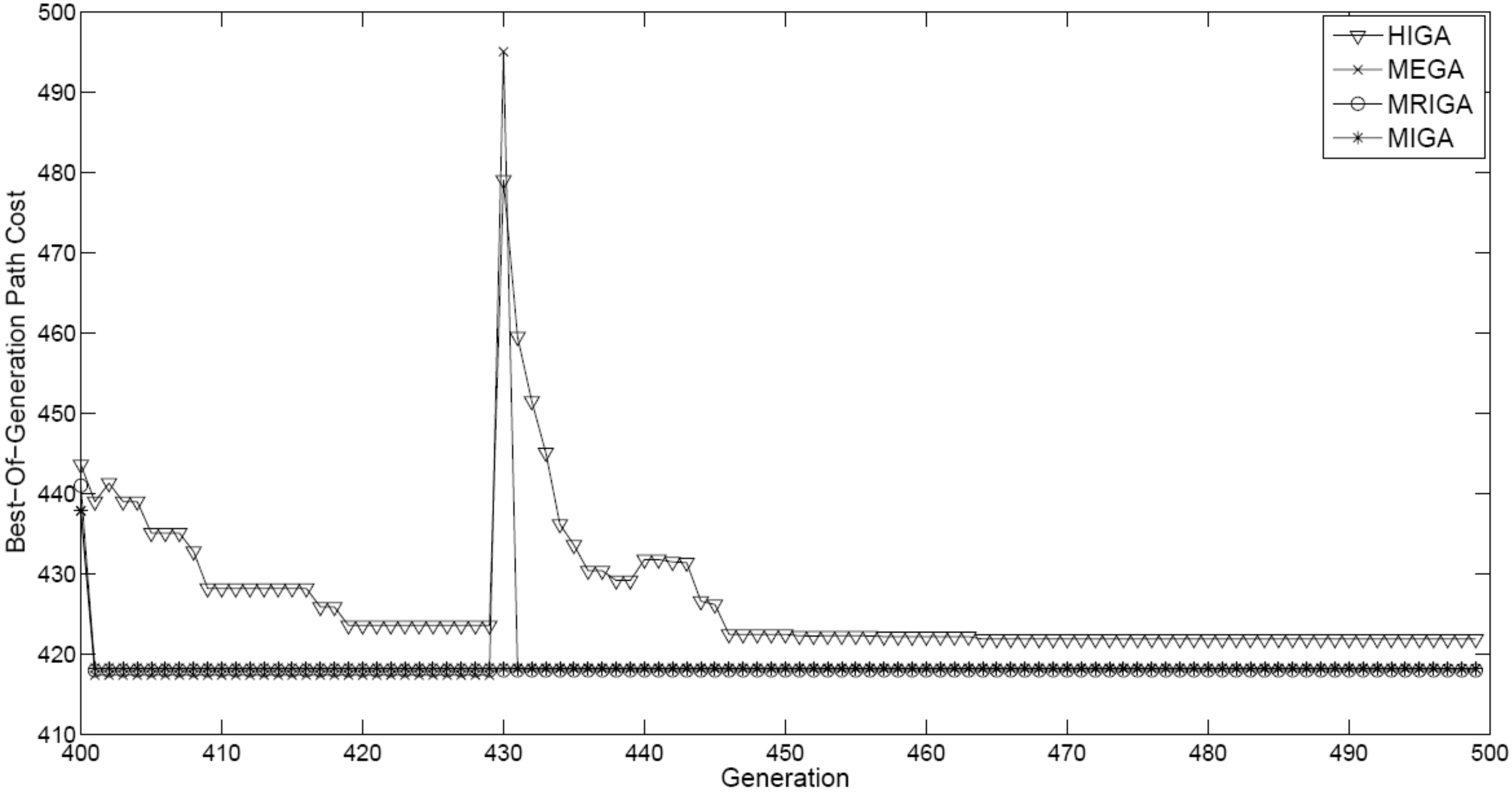
- The ideal memory size: 20 is good enough



Compare Memory related GAs with Traditional GAs in Cyclic Environments



Compare Memory related GAs with Immigrants GA in Cyclic Environments



Conclusions

- Create acyclic and cyclic dynamic wireless network environments.
- Test the two traditional GAs and six dynamic GAs (immigrants, memory, immigrants&memory) over the dynamic shortest path routing problem.
- The six dynamic GAs show decent performance, i.e., quickly adapting to the environmental changes (the network topology changes) and producing new good solutions.
- Immigrants GAs work better than memory related GAs in acyclic environment, vice versa in cyclic environment.

Thank you!

Q&A