CHAPTER 2 LITERATURE REVIEW

This chapter firstly reviews the TPS fundamentals and related researches. The basic concepts of the methods, including FLC, GA, and ACO, adopted in this study are then briefly elaborated and reviewed. Finally, a short summary is followed.

2.1 TPS Fundamental

TPS is an operational strategy that facilitates the movement of in-service transit vehicles, such as trams or buses, through traffic-signal controlled intersections. Expected benefits of TPS include reduction of transit travel time, operating cost, and exhaust gas emissions and increase of transit schedule reliability and rider comfort. However, TPS can cause remarkable negative impacts on the traffic from competing approached. To properly design the TPS which can effectively curtail the transit delays with the minimal impact to the competing traffics, a variety of preemption signal strategies and two different approaches to provide priority to transit vehicles are discussed below.

2.1.1 Priority strategy for TPS

There are a variety of ways including passive priority, green extension, red truncation (early green), actuated transit phase, phase insertion, phase rotation, and adaptive/real-time control to provide priority strategies when designing a TPS system (ITS America, 2004). Excluding the passive priority, all other TPS strategies are active, where the real-time arrival information of transits and other traffics is required. Figure 2-1 indicates the various strategies of TPS and those strategies are detailed as follows.

Passive priority operates continuously regardless of whether transit is present or not, and does not require a transit detection system. In general, when transit operations are predictable (*e.g.* consistent dwell times), transit frequencies are high, and traffic volumes are low, passive priority strategies can be an efficient approach for TPS. One such passive priority strategy is establishing signal progression for transit. The coordination plan would account for the average dwell time at transit stops. Since the signals are coordinated for the flow of transit vehicles and not other traffic, other traffic may experience unnecessary delays, stops, and frustration (*i.e.* phone calls to the signal operators). Therefore, the volume of traffic parallel to the TPS movements should also be considered with a transit signal progression approach. It is important to note that other "passive" improvements may also be of benefit to transit. Operational improvements to signal timing plans, such as retiming or coordinating signals on a corridor, may improve traffic flow and reduce transit travel time as well.

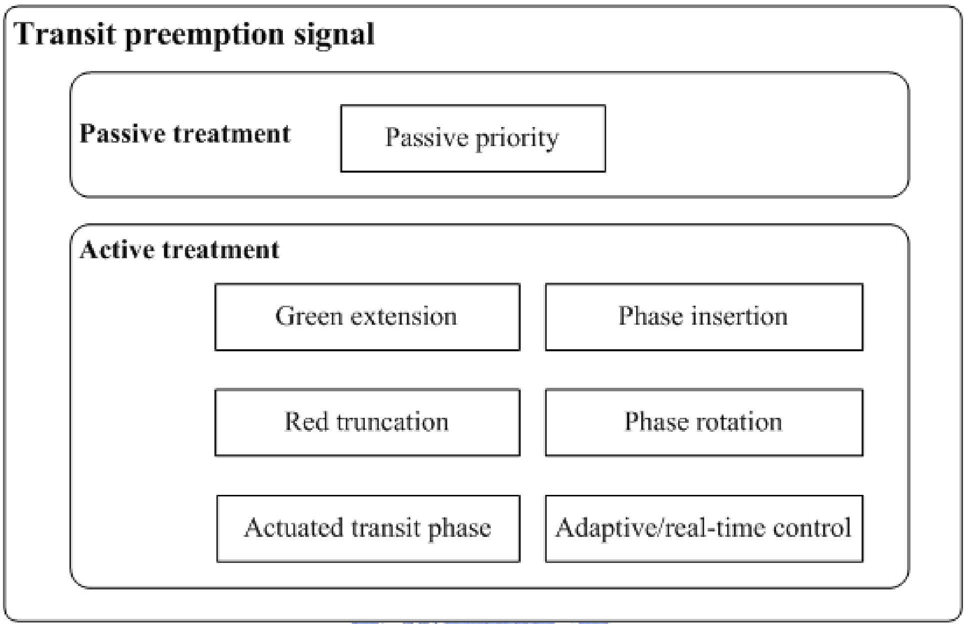

Figure 2-1 Various TPS strategies.

A **green extension** strategy extends the green time for the TPS movement when a transit vehicle is approaching. This strategy only applies when the signal is green for the approaching transit vehicle. Green extension is one of the most effective forms of TPS since a green extension does not require additional clearance intervals, yet allows a transit vehicle to be served and significantly reduces the delay to that vehicle relative to wait for a red truncation or special transit phase.

A **red truncation** strategy shortens the green time of preceding phases to expedite the return to green for the movement where a transit vehicle has been detected. This strategy only applies when the signal is red for the approaching transit vehicle. A red truncation and a green extension strategy may be applied together to maximize the time within the signal cycle in which transit would be eligible for priority.

Actuated transit phases are only displayed when a transit vehicle is detected at the intersection. An example would be an exclusive left turn lane for transit vehicles. The left turn phase is only displayed when a transit vehicle is detected in the lane. Another example would be the use of a queue jump phase that would allow a transit vehicle to enter the downstream link ahead of the normal traffic stream.

When a special priority phase is inserted within the normal signal sequence, it is referred to as **phase insertion**. The phase can only be inserted when a transit vehicle is detected and requests priority for this phase.

The order of signal phases can also be "rotated" (*i.e.* **phase rotation**) to provide TPS. For example, a northbound left turn phase could normally be a lagging phase, meaning it follows the opposing through signal phase. A northbound left turning bus-requesting priority that arrives before the start of the green phase for the through movement could request the left turn phase. With the phase rotation concept, the left turn phase could be served as a leading phase in order to expedite the passage of the transit vehicle.

Adaptive/real-time TPS strategies provide priority while simultaneously trying to optimize given performance criteria. The criteria may include person delay, transit delay, vehicle delay, and/or a combination of these criteria. These strategies continuously optimize the effective timing plan based on real-time, observed data. They typically require early detection of a transit vehicle in order to provide more time to adjust the signals to provide priority while minimizing traffic impacts. Adaptive systems also often require the ability to update the transit vehicle's arrival time, which can vary due to the number of stops and traffic conditions. The updated arrival time can then be fed back into the process of adjusting the signal timings.

2.1.2 Unconditional and conditional TPS

There are generally two different approaches to provide TPS. The first one, called unconditional TPS, is to provide priority to all transit vehicles sending a request signal without any premise. Most TPS applications currently utilize this simple approach due to its low cost advantage. After the priority request is transmitted to the traffic signal controller, in green extension or red truncation priority strategies, an extension of the green phase serving the transit vehicle is extended up to a maximum limit or a truncation of the corresponding cross street red phase is truncated. Such truncation should respect to the minimum green time required for pedestrians to safely cross the street.

An alternative approach, called conditional TPS, uses more sophisticated systems to determine if the transit vehicle is behind schedule or meets other pre-defined conditions and then makes a control decision after an approaching transit vehicle disseminates a priority request. This approach has been made feasible by recent technological developments such as automatic vehicle location (AVL) system (Chang *et al*., 1996). Conditional TPS means that not all transit vehicles would be provided with priority treatment, depending upon the bus schedule, the traffic conditions of competing approaches, etc.

2.2 TPS Related Research

Over the past several decades, a variety of studies related to transit priority strategies have been conducted, through either experimental studies with before and after analysis (*e.g.* Elias, 1976; Cottinet *et al*., 1979; Bishop, 1994; Hunter-Zaworski *et al*., 1995; Lewis, 1996; Duncan and Mirabdal, 1996; Boje and Nookala, 1996; Vahidi, 2000; Toone, 2003; Skehan, 2003) or analytical model and simulation (*e.g.* Jacobson and Sheffi, 1981; Khasnabis *et al*., 1993; Sunkari *et al*., 1995; Cisco and Khasnabis, 1995; Chang *et al*., 1996; Wu and Hounsell, 1998; Su and Lee, 1999; Hsu *et al.,* 2003; Dion *et al*., 2004).

In the experimental studies, Elias (1976) conducted a bus preemption study in Sacramento, California, demonstrating benefits derived from preemption and showing that the added delays to automobiles were negligible with low bus frequency. Cottinet (1977) investigated and compared three preemption strategies in an experiment in Nice, France. The first strategy allowed an incoming bus to change the signal to green whenever it hit the detector. The second strategy shortened the red period only and the third allowed only for an extension of the green period. The result reported that the first strategy was superior to the last two strategies.

Bishop (1994) implemented various TPS strategies for buses on five case study sites in Europe. The investigating results showed 6% ~ 42% reduction in transit travel time and 0.3% ~ 2.5% increase in auto travel times. Hunter-Zaworski *et al.* (1995) described the Powell Boulevard Pilot Project, conducted in Portland metropolitan area, which tested the effectiveness of two techniques, including green extension for far-side stop locations and queue jump for near-side stop locations, for determining traffic signal priority for buses. The survey results in this study indicated that bus travel time for the experimental bus line was reduced slightly in the peak direction with bus signal priority.

Lewis (1996) implemented green extension and red truncation strategies for buses in Tualatin Valley Highway, Portland, Oregon. The investigating results showed bus traveling time savings of 1.7% ~ 14.2% per trip. In term of the performance for all vehicles, there was 2 to 13 seconds reduction in per intersection delay and up to 3.4% reduction in travel time variability. Duncan and Mirabdal (1996) also implemented green extension and red truncation strategies. The TPS was equipped for the LRT and trolleys in San Francisco, California. The results showed 6% ~ 25% reduction in transit delay.

Boje and Nookala (1996) carried out green extension, red truncation, and actuated transit phase TPS strategies for buses running on the Louisiana Avenue; Minneapolis, Minnesota. The results revealed 0% ~ 38% reduction in bus travel time depending on the TPS strategies. However, the implementation of TPS also caused 23% increase in total traffic delay and skipping signal phase would cause some driver frustration. Vahidi (2000) implemented green extension and red truncation strategies for streetcars and buses in Toronto, Ontario. The results showed up to 46% reduction in transit delay and 10 streetcars and 4 buses removed from service due to the saving of traveling time. Moreover, the results also showed that the cross street traffic of TPS equipped intersection was not significantly affected.

Toone (2003) conducted green extension and red truncation strategies for buses in Rainier Avenue, Seattle, Washington. The results indicated 24% average reduction in stops for buses and 25% ~ 34% reduction in bus delay. For all vehicles passing the intersection, it had 5% ~ 8% average reduction in travel time. Skehan (2003) carried out green extension, red truncation, and actuated transit phase TPS strategies for buses running on Wilshire and Ventura Boulevards, Los Angeles, California. The TPS was introduced as part of BRT system in this project. The results showed 8% reduction in average bus running time and 33% ~ 39% decrease in bus delay at the TPS intersection. The results also revealed that the implementation of TPS has insignificant impact to cross street traffic: average of 1 second per vehicle per cycle increase in delay and did not change the traffic level of service. Those experimental studies mentioned above are summarized in Table 2.1.

In the analytical model and simulation studies, Jacobson and Sheffi (1981) developed an analytical model of the delays to bus passengers and automobile occupants at a signalized intersection under bus preemption. The preemption strategy included an extension of the duration of the green and a shortening of the duration of the red. The approximate analytical approach in this paper enabled the analyst to investigate the effects of several design parameters on the total intersection delay. The result of the experiments showed that bus preemption reduced the total delay, expressed in person-seconds, when both bus occupancy and the flow of buses were high. Moreover, it was also shown that, contrary to common engineering experience, bus preemption is beneficial even when the cross traffic was high.

Khasnabis *et al.* (1993) presented a computer simulation model (PREEMPT) to depict the operating cost and ridership consequence of signal preemption. This tool is for sketch planning purposes and is designed to provide the user with broad information on changes in fleet size, travel time, revenue, and operating cost as a consequence of changes in travel aped attributable to signal preemption. The output trends observed in three simulation cases presented appear reasonable, indicating that the PREEMPT model is applicable. However, no effort was made to validate the model through the actual deployment of the model.

Sunkari *et al.* (1995) developed a model to evaluate the impacts of implementing a priority strategy at signalized intersections. Priority was provided by phase extension and early start of the priority phase at regular intervals. The model uses the delay equations for signalized intersections in the 1985 Highway Capacity Manual. To compare the model delay and investigated delay, a priority strategy was developed and implement in the field and delay was measured. The comparison results indicated that the model seems to be predicting delay reasonably and accurately. In some phases, however, the model was overestimating delay.

Table 2.1 Summary of experimental TPS related studies

Author (year)	Location	Transit type	TPS strategy	Benefit/Impact
Elias (1976)	Sacramento, California	Bus	Various	● Added delays to automobiles were negligible with low bus frequency
Cottinet *et al.* (1979)	Nice, France	Bus	Green extension, red truncation, actuated transit phase	● Performance of actuated transit phase was superior to the other two strategies
Bishop (1994)	Europe	Bus	Various	● 6% ~ 42% reduction in transit travel time ● 0.3% ~ 2.5 increase in auto travel times
Hunter-Zaworski *et al.* (1995)	Powell Boulevard, Portland, Oregon	Bus	Green extension, queue jump	● Bus travel time for the experimental bus line was reduced slightly in the peak direction
Lewis (1996)	Tualatin Volley Highway, Portland, Oregon	Bus	Green extension, red truncation	● Bus traveling time savings of 1.7% ~ 14.2% per trip ● 2 to 13 seconds reduction in vehicle delay per intersection ● Up to 3.4% reduction in travel time variability
Duncan and Mirabdal (1996)	San Francisco, California	LRT and Trolleys	Green extension, red truncation	● 6% ~ 25% reduction in transit delay
Bojr and Nookala (1996)	Louisiana Avenue, Minneapolis, Minnesota	Bus	Green extension, red truncation, actuated transit phase	● 0% ~ 38% reduction in bus travel time depending on the TPS strategies ● 23% increase in total traffic delay ● Skipping signal phase caused some driver frustration
Vahidi (2000)	Toronto, Ontario	Streetcar and bus	Green extension, red truncation	● Up to 46% reduction in transit delay ● 10 streetcars and 4 buses removed from service ● Cross street traffic was not significantly affected
Toone (2003)	Rainier Avenue, Seattle, Washington	Bus	Green extension, red truncation	● 24% average reduction in stops for buses ● 25% ~ 34% reduction in bus delay ● 5% ~ 8% average reduction in travel time for all vehicles
Skehan (2003)	Wilshire and Ventura Boulevards, Los Angeles, California	Bus	Green extension, red truncation, actuated transit phase	● Introduced as part of BRT system ● 8% reduction in average bus running time ● 33% ~ 39% decrease in bus delay ● Average of 1 second per vehicle per cycle increase in delay of cross street traffic ● Did not change the traffic level of service

Cisco and Khasnabis (1995) presented two deterministic methods for assessing delay and queue length consequences of bus preemption at signalized intersection. The procedures were adapted from queuing theory. Three types of preemption strategies including green extension, red truncation, and red interruption were tested. The two deterministic methods macroscopically simulated groups of vehicles and microscopically treated each individual vehicle at the intersection using regular signal timing and timing under preempted conditions. The case studies indicated some variation between three strategies tested, between the two methods used, and between the different traffic levels. Macroscopic method is preferred for higher traffic level while microscopic method should be used for lighter traffic level.

Chang *et al.* (1996) presented two integrated models for adaptive bus preemption control in the absence and presence of Automatic Vehicle Location (AVL) systems. Instead of using prespecified strategies, such as phase extension and/or phase early start, the proposed models make a preemption decision based on a performance index which includes vehicle delay, bus schedule delay, and passenger delay. Real-time traffic variables from the output of TRAF-NETSIM were made use of to test the performance of the algorithms. The proposed models with a preemption function yielded favorable results, both in the absence and presence of AVL technology, over the strategies without preemption, for all traffic conditions.

Wu and Hounsell (1998) developed analytical procedures which allow pre-implementation evaluation of specific categories of pre-signal. The pre-signal aimed to give buses priority access into a bus advance area of the main junction stop line so as to avoid the traffic queue and reduce bus delay at the signal controlled intersection. This paper analyzed two categories of pre-signal, which have different operating characteristics, different requirements for signaling and different impacts on capacity and delay. Equations were developed to enable delays to priority and non-priority traffic, with and without pre-signals, to be estimated. The example analyses had shown that category A pre-signal (where buses were unsignalized at the pre-signal) could save bus delays without significant disbenefit to non-priority traffic. Delay savings to bus were highest where there was a long red period at the non-priority traffic pre-signal, which was possible when the proportion of green time at the main signal was low. However, category B pre-signal (where buses were signalized at the pre-signal) showed to generally cause disbenefit to buses unless bus detectors were installed to give signal priority to buses.

Su and Lee (1999) proposed a fully activated bus preemption control model comprising the strategies of green extension, red truncation, and red interruption. To prevent the adversely control effects, a fuzzy model designed according to the used green duration was proposed to properly choose red truncation or red interruption strategies. This paper employed the microscopic simulation software named CORSIM to generate extensive traffic information for system control efficiency evaluation. The results showed that the proposed model could outperform the other models and effectively reduce the total passenger delay under various traffic conditions.

Hsu *et al.* (2003) developed a simulation system named MISSBUS (Microscopic Simulation System for Bus Operation) with microscopic aspect for investigating the performance of bus preemption signal. This system could simulate the various bus preemption signal control logic (including green extension, red truncation, and red interruption) under different bus traffic volumes and different layouts of bus stops. The example simulation results indicated the proposed system might be a useful tool for investigating bus preemption signal design.

Dion et al. (2004) evaluated the potential benefits of implementing TPS by the INTEGRATION microscopic traffic simulation software. The priority strategy provided in this paper was green extension and red truncation within a fixed-time traffic signal control environment. The simulation results indicated that the buses would typically benefit from transit priority, but that these benefits might be obtained at the expense of the overall traffic, particularly when traffic demand was high. However, it was also found that in periods of lesser traffic demand, the overall negative impacts could be negligible due to the availability of spare capacity at the signalized intersection. The TPS related analytical model and simulation studies are summarized in Table 2.2.

Due to the simplicity of implementing green extension and red truncation strategies, most of the studies reviewed above conducted these two strategies and the effectiveness have been proven. Therefore, this study also adopts these two strategies to develop the novel TPS control mechanisms.

Table 2.2 Summary of analytical model and simulation studies

Author (year)	Model development	TPS strategy	Findings
Jacobson and Sheffi (1981)	Develop an analytical model of the delays to bus passengers and automobile occupants at a signalized intersection under bus preemption.	Green extension, red truncation	● Bus preemption reduced the total person delays when both bus occupancy and the flow of buses were high. ● Bus preemption was beneficial even when the cross traffic was high.
Khasnabis *et al.* (1993)	Propose a computer simulation model to depict the operating cost and ridership consequence of signal preemption.	Not mentioned	● The reasonable output trends indicated that the PREEMPT model was functional.
Sunkari *et al.* (1995)	Develop a model to evaluate the impacts of implementing a priority strategy at signalized intersections.	Green extension, red truncation	● The model could predict delay reasonably and accurately. ● The model overestimated delay in some phases.
Cisco and Khasnabis (1995)	Develop two deterministic methods for assessing delay and queue length consequences of bus preemption at signalized intersection.	Green extension, red truncation, red interruption	● Macroscopic method is preferred for higher traffic level while microscopic method should be used for lighter traffic level.
Chang *et al.* (1996)	Develop two integrated models for adaptive bus preemption control in the absence and presence of Automatic Vehicle Location (AVL) systems.	Adaptive bus preemption control	● The proposed models performed better than the strategies without preemption both in the absence and presence of AVL technology for all traffic conditions.
Wu and Hounsell (1998)	Develop analytical procedures which allow pre-implementation evaluation of specific categories of pre-signal.	Pre-signal	● Category A pre-signal could save bus delays without significant disbenefit to non-priority traffic. ● Category B pre-signal showed to generally cause disbenefit to buses unless bus detectors were installed to give signal priority to buses.
Su and Lee (1999)	Propose a fully activated bus preemption control model.	Green extension, red truncation, red interruption	● The proposed model could outperform the other models and effectively reduce the total passenger delay under various traffic conditions.
Hsu *et al.* (2003)	Develop a simulation system named with microscopic aspect for investigating the performance of bus preemption signal.	Green extension, red truncation, red interruption	● The proposed system might be a useful tool for investigating bus preemption signal design.
Dion et al. (2004)	Evaluate the potential benefits of implementing TPS by a microscopic traffic simulation software.	Green extension, red truncation	● Buses would typically benefit from transit priority. ● The benefits were at the expense of the overall traffic, particularly when traffic demand was high. ● The overall negative impacts could be negligible in lesser traffic demand.

2.3 Methods

The proposed GFLC and AGFLC comprise three methods: Fuzzy Logic Controller (FLC), Genetic Algorithm (GA), and Ant Colony Optimization (ACO). Brief introductions of these methods are given below.

2.3.1 Fuzzy Logic Controller

The underline theory for the FLC system, first proposed by Zadeh (1973), is to use fuzzy logic rules to form a control mechanism to approximate expert perception or judgment under given conditions. This system is also termed as fuzzy control system, or fuzzy inference system, or approximate reasoning, or expert system. The FLC is a rule-based system that uses fuzzy linguistic variables to model human rule-of-thumb approaches for problem solving, and thus overcome the limitation that classical expert systems may meet because of their inflexible representation of human decision making. The major strength of a FLC also lies in the way a non-linear output mapping of a number of inputs can be specified easily using fuzzy linguistic variables and fuzzy rules (Chin and Qi, 1998). The framework of FLC is depicted in Figure 2-2. A typical FLC system composes of four major components including rule base, data base, inference engine, and defuzzification. They are briefly explained in the following.

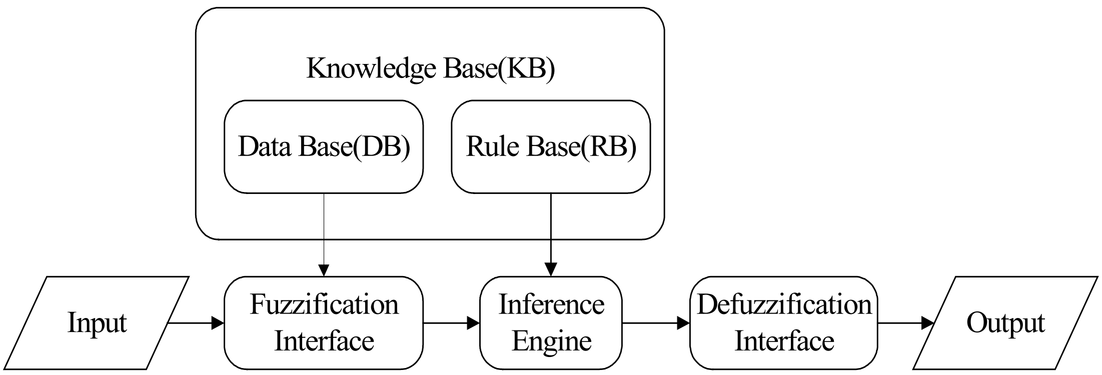

Figure 2-2 Framework of the FLC.

(1) Rule base (RB). The RB is composed of finite IF-THEN rules, from which an inference mechanism is formed. A standard form of RB with M fuzzy rules is represented as:

Rule 1 ： IF $x_1 = A_{11}$ AND $x_2 = A_{12}$ AND … AND $x_N = A_{1N}$ THEN $y = B_1$

Rule 2 ： IF $x_1 = A_{21}$ AND $x_2 = A_{22}$ AND … AND $x_N = A_{2N}$ THEN $y = B_2$

.

.

.

Rule M ： IF $x_1 = A_{M1}$ AND $x_2 = A_{M2}$ AND … AND $x_N = A_{MN}$ THEN $y = B_M$

where $x_1,\ldots, x_N$ are N state variables and y is a control variable. A_{i1} ,…, A_{iN} and B_i (i=1,…,M) are respectively the linguistic variables for $x_1,\ldots, x_N$ and y in the universe of discourse of $U_1,\ldots,U_n$ and V. Taking the driving speed as an example, the linguistic degrees can be very fast, fast, normal, slow and very slow. The more general form of the fuzzy rules listed above is: IF premise THEN consequent. The left-hand-side of the rules, the premise or so-called the antecedent, is associated with the fuzzy controller inputs (or called state variables). The right-hand-side of the rules, the consequent, is associated with the fuzzy controller outputs (or called control variables). Each antecedent can be composed of the conjunction of several state variables; however, each consequent is usually formed by one control variable.

(2) Data base (DB). The DB is formed by the specific membership functions of linguistic variables A_{i1} ,…, A_{iN} and B_i that transform crisp inputs into fuzzy ones. Triangle, trapezoid and bell-shaped membership functions are commonly used.

(3) Inference engine. The operators within the fuzzy rules form the inference engine. Generally, fuzzy rules use AND (taking minimum value) or OR (taking maximum value) as connecting operators between state variables.

(4) Defuzzification. For making a decision, defuzzification is the synthesis of inference results of all activated fuzzy rules into crisp outputs. Mean of maximum method, center of gravity method, Tsukamoto's method, and weighted average method are commonly used. The diagrammatic representations of these defuzzification methods are illustrated in Figure 2-3.

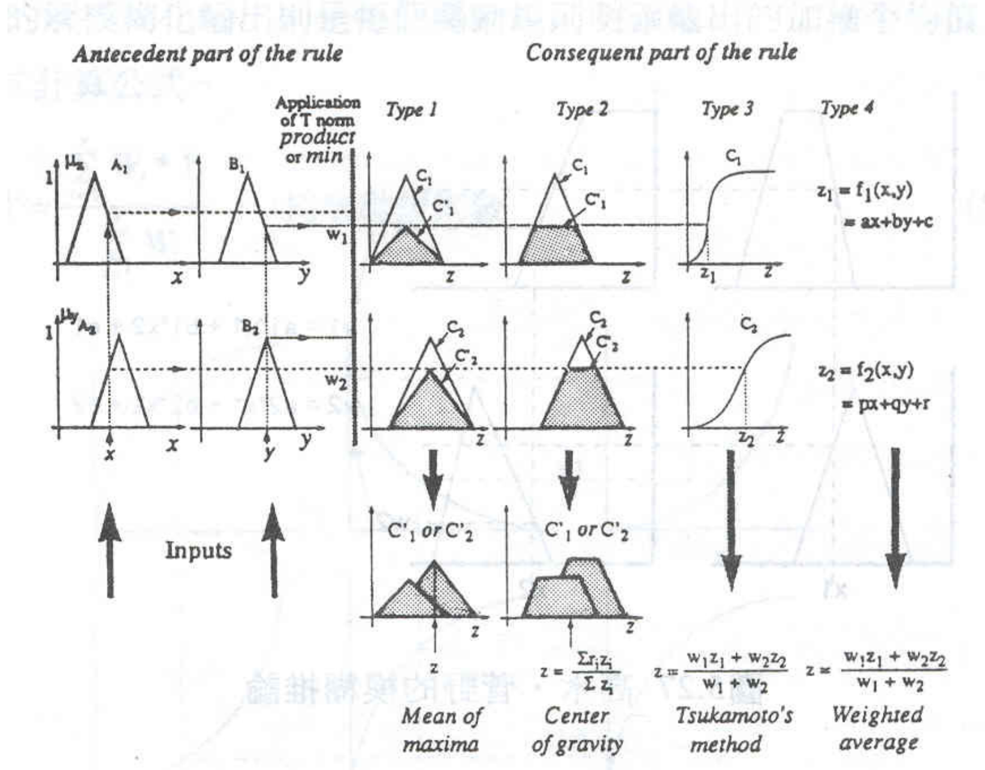

Figure 2-3 Diagrammatic representations of defuzzification methods (Passino and Yurkovich, 1997).

Conventional control theory is well suited for applications where the process can be reasonably well characterized in advance and where the number of parameters that must be considered is small. However, there are many processes that are not well characterized or are subjected to a large number of uncontrolled, changeable or immeasurable parameters. The FLC appears to offer a new method to produce high-performance control rules for those control processes without having good models of the processes being controlled (Li and Zhang, 1996). For example, the conventional adaptive signal controllers, such as SCOOT, SCATS, and OPAC, employ mathematical equations or models to determine crisp threshold values as the cores of control mechanism. Its control performance might be negatively influenced by the uncertainty of traffic conditions. Since a fuzzy control system has excellent performance in data mapping as well as in treating ambiguous and vague aspects of human perception or judgment, many recent researches have applied FLC to traffic signal control.

The applications of FLC to signal control are to determine the signal phasing and timing plans, including priority of phases, cycle length and split, by utilizing the real-time traffic data, such as vehicle arrival or arrival rate, occupancy, queue length, and speed, collected by detectors. Pappis and Mamdani (1977) first apply FLC to signal control by using 25 fuzzy rules with three states variables: elapsed time, vehicle arrivals, and queue length to determine the extension of green time. Their simulation results show that the FLC signal control has total vehicle delays 10 to 21% less than an actuated signal control. Favilla *et al.* (1993) employ 11 fuzzy rules with two state variables, vehicle arrivals in the green phase and queue length in the red phase, to control the extension of green time. Mohamed *et al.* (1999) establish a two-stage FLC model. The first stage is to evaluate the traffic intensity in the competing directions by 16 fuzzy rules with traffic flows or queue lengths as state variables. The second stage is to decide the extension or termination of current phase by 16 fuzzy rules with traffic intensities in green and red phases as state variables. Niittymäki (2001) also develops a two-stage FLC model. The first stage is to evaluate the traffic conditions by three fuzzy rules with traffic flow and occupancy as state variables. The second stage is to determine the green time extension by 20 fuzzy rules with vehicle arrival in green phase and queue length in red as state variables. The results from both simulation and field test reveal that the FLC model has outperformed over the actuated signal control. Except the application to traffic signal control, in transportation related researches, FLC has also been applied to transportation planning (including trip generation, trip distribution, modal spilt, and route choice), selection of transportation investment projects, accident analysis and prevention, level of service evaluation, aircraft control, and ship loading/unloading control (Teodorovic, 1999).

2.3.2 Genetic Algorithm

The GA, first proposed by Holland (1975), is a searching process based on the mechanics of natural selections and natural genetics. GA is a global optimization technique that avoids many shortcomings exhibited in conventional search techniques on a large and complicated search space. Generally, a simple GA contains three basic operators: selection, crossover, and mutation. GA starts with a population of randomly generated solutions (also called chromosomes) determined by genes that are in code term, and advance toward better solutions by applying genetic operators, modeled on the genetic processes occurring in nature. During the iterative procedures, a constant size of population of candidate solution is maintained, and this population undergoes evolution in a form of natural selection (Herrera *et al.*, 1998).

In each generation, the selection is a process by which the chromosomes, coded strings, with larger fitness values can produce accordingly with higher probabilities large number of their copies in the new generation. The crossover is a process by which the systematic information exchange between two coded strings is implemented using probabilistic decisions. In a crossover process, two coded strings are chosen from the matching pool and arranged to exchange their corresponding positions of binary strings at a randomly selected partitioning position along them. This process can combine better qualities among the preferred good strings. And then the mutation is a process by which the chance for the GA to reach the optimal point is reinforced through just an occasional alteration of a value at a randomly selected bit position. The mutation process may quickly generate those strings which might not be conveniently produced by the previous selection and crossover process to avoid the trap of local solutions. The GA runs iteratively repeating the above process until it arrives at a predetermined ending condition. The process of going from the current population to the next population constitutes one generation in the execution of a GA. A typical GA cycle is depicted as Figure 2-4.

GA is theoretically and empirically proven to provide robust search in complex spaces, giving a valid approach to problems requiring efficient and effective searching. GA methods have been applied to many different problems like function optimization, routing problem, scheduling, design of neural networks, system identification, digital signal processing, computer vision, control and machine learning (Goldberg, 1989).

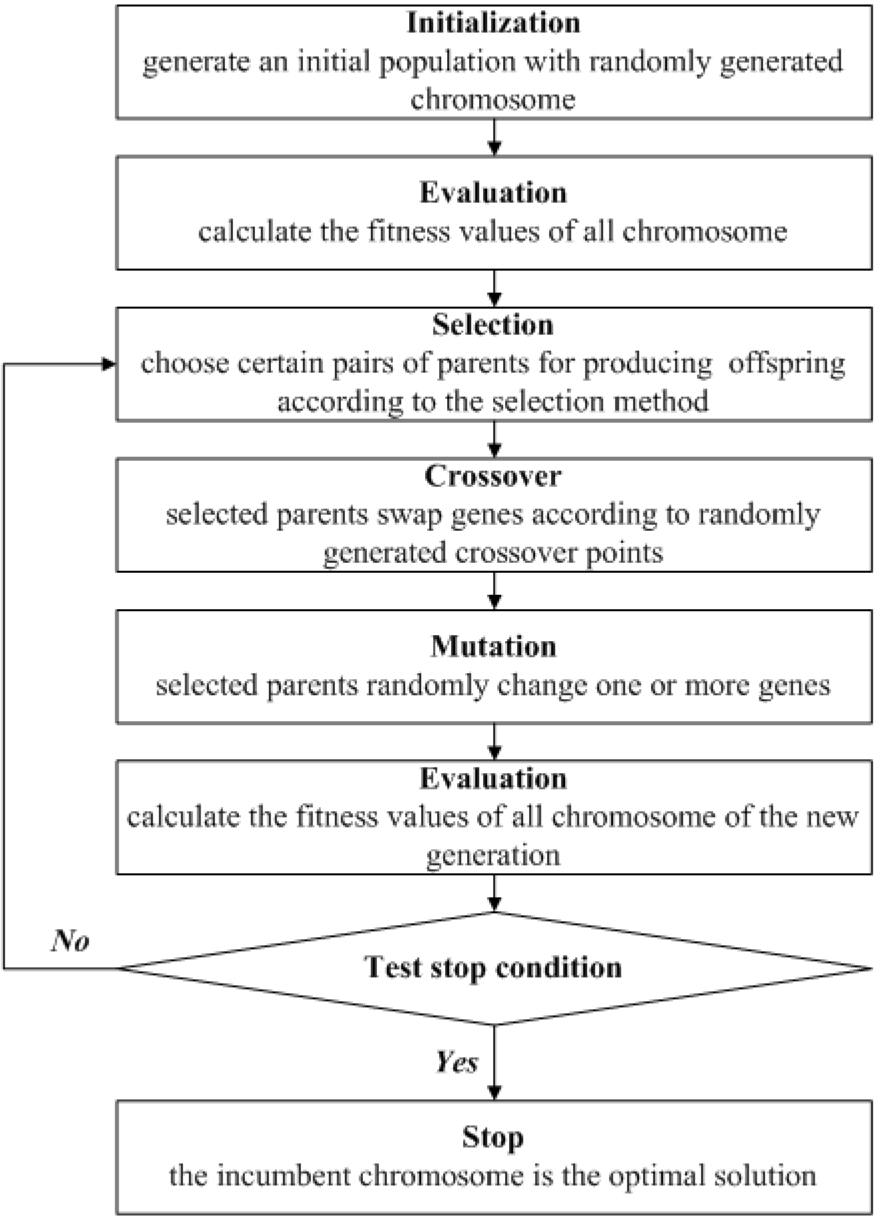

Figure 2-4 Typical operation of a GA.

2.3.3 Ant Colony Optimization

Ant algorithm draws inspiration from the social behavior of ants to provide food to the colony. In the food search process including the food finding and the return to the nest, ants deposit a substance called a pheromone. Ants have the ability to smell the pheromone and pheromone trails guide the ants during the search. When an ant reaches a branch, it decides to take the path according to a probability defined by the amount of pheromone existing in the links. In this way, the depositions of the pheromone dominate the construction of a path between the nest and the food that can be followed by new ants. The progressive action of the ants makes the length of the path reduced step by step. The shortest path is finally the more frequently visited one and has the highest pheromone on it. Conversely, the longer paths are less visited and the pheromone on them is evaporated with time passing.

There are many different heuristics named based on the general ant algorithms structure. The first class of ant algorithms called the Ant System (AS) was proposed in 1991 and then three types of ant algorithms called ant-density, ant-quantity, and ant-cycle are introduced (Dorigo et al., 1991; Colorni et al., 1992; Dorigo, 1992). The typical problem, which was researched, was the well-known Traveling Salesman Problem (TSP). In the ant-density and ant-quantity algorithms, the ants update the pheromone directly after a move from one city to an adjacent city. In the ant-cycle algorithm, the pheromone update is only done after all the ants have constructed the tours and the amount of pheromone deposited by each ant is set to be a function of the tour quality. Although AS is useful for discovering good or optimal solutions for small TSP, the time required is unbearable for large size TSP problems. Therefore, a substantial amount of research on ACO has focused on how to improve the AS (Lee et al., 2001).

ACO was first proposed by Dorigo et al. (1996) and differs from AS in three main points (Dorigo and Gambardella, 1997a,b). First, it exploits the search experience accumulated by the ants more strongly than AS does through the use of a more aggressive action choice rule. Second, pheromone evaporation and pheromone deposit take place only on the arcs belonging to the best-so-far tour. Third, each time an ant use an arc (i, j) to move from city i to city j, it removes some pheromone from the arc to increase the exploration of alternative paths. The typical operation of ACO for TSP is briefly narrated as follows (Dorigo and Stützle, 2004):

Step 0: Initialization. K ants are placed on randomly chosen cities. The initial pheromone trail (τ^0) between any two cities is set to be a small positive constant. Set the values of all parameters including the parameter of transition rule (q_0), pheromone decay parameter for local update rule (ξ), pheromone decay parameter for global update rule (ρ), number of ants (K), and maximal iteration (t_{max}).

Step 1: Tour construction. To construct a complete solution, an ant successively goes over each city it has not visited yet with a probability that depends on the heuristic information and pheromone trail. The probability of the k^{th} ant moves from city r to city s can be computed as follows:

$$s = \arg\max_{j \in J_r^k} \{[\eta_{rj}]^\alpha [\tau_{rj}]^\beta\} \quad if \;\; q \le q_0 \;\; \text{(exploitation)}, \tag{2.1}$$

or visit s with P_{rs}^k, if $q > q_0$ (exploration), where

$$P_{rs}^k = \begin{cases} \dfrac{[\eta_{rs}]^\alpha [\tau_{rs}]^\beta}{\sum_{j \in J_r^k} [\eta_{rj}]^\alpha [\tau_{rj}]^\beta}, & if \;\; s \in J_r^k \\ 0 & , otherwise \end{cases} \tag{2.2}$$

where η_{rj} is a heuristic value representing for the clossness, which are the inverse of distance between city r and city j. τ_{rj} represents the amount of pheromone trail on edge linking city r to city j. J_r^k represents the set of cities that remain to be visited by the ant k positioned on city r. The symbols α and β are two parameters which determine the relative importance of closeness and pheromone trail between two cities. The q is a random number chosen randomly with uniform probability in [0,1] and q_0 $(0 \le q_0 \le 1)$ is a parameter representing the threshold to implement exploitation or exploration. P_{rs}^k represents the probability with which ant k chooses to move from city r to city s when implementing exploration.

Step 2: Local updating. The local pheromone update rule is applied immediately after one ant has crossed an arc (i, j) during the tour construction. It can be represented by:

$$\tau_{ij} \leftarrow (1-\xi)\tau_{ij} + \xi\tau^0 \tag{2.3}$$

where $\xi \in (0,1)$ is a pheromone decay parameter of local update rule making the pheromone not going too far beyond τ^0. Experimentally, a good value for τ^0 is found to be $\tau^0 = (NL_{nn})^{-1}$. N is the number of cities in the TPS instance and L_{nn} is the total distance solved by greedy heuristic. The effect of the local updating rule is that each time an ant uses an arc (i, j) its pheromone trail τ_{ij} is reduced, so that the arc becomes less desirable for the following ants. In other word, this allows an increase in the exploration of arcs that have not been visited yet and, in practice, has the effect that the algorithm does not show a stagnation behavior.

Step 3: Global updating. After all ants have completed their tours, the global updating rule is to deposit a certain amount of pheromone ($\Delta\tau_{ij}$) on the arcs belonging to the best-so-far tour ($T^*(t)$) constructed by the best-so-far performed ant. The pheromones on the other links remain unchanged. The amount of pheromone $\Delta\tau_{ij}$ deposited is inversely proportional to the length of the tour. That is, the shorter the best-so-far tour, the greater the amount of pheromone deposited on links. The pheromone updates for the t^{th} iteration are as follow:

$$\tau_{ij}(t+1) \leftarrow (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}(t) \quad if \ arc(i,j) \in T^*(t) \tag{2.4}$$

where $\tau_{ij}(t)$ and $\tau_{ij}(t+1)$ are the pheromone level of the incumbent iteration and next iteration on arc (i, j), respectively. $\rho \in (0,1]$ is a pheromone decay parameter of global update rule governing the evaporation of pheromone trail. $T^*(t)$ is the best-so-far tour constructed by the best-so-far ant till the t^{th} iteration and $\Delta\tau_{ij}(t) = 1/L^*(t)$, $L^*(t)$ is the tour length of $T^*(t)$.

It is important to note that the global pheromone update only applied to the arcs of best-so-far tour, not to all the arcs as in AS. In this way the computational complexity of ACO is reduced at each iteration. Besides, the pheromone is discounted by the evaporation factor ρ, this results in the new pheromone being a weighted average between the old pheromone value and the amount of pheromone deposited.

Step4: Incumbent tour updating. After an iteration (global updating) has been completed, the incumbent solution is tested and updated as: If $\min_k\{L_k(t)\} = L^+(t) < L^*(t)$, then let $L^*(t)$=$L^+(t)$ and $T^*(t)$=$T^+(t)$; otherwise $L^*(t)$

and $T^*(t)$ remain unchanged, where $L_k(t)$ is the constructed tour length of ant k, $L^+(t)$ is the shortest path of iteration t.

Step5: Testing of stop condition. If the maximal iterations t_{max} has been reached, then terminate. $T^*(t_{max})$ is the best tour and $L^*(t_{max})$ is its tour length. Otherwise, go back to Step 1.

Since the first application to the TSP, a lot of research has tried to apply ACO to TSP more efficiently (Stützle and Hoos, 1997, 2000; Bullnheimer et al., 1999c; Cordón et al., 2000). Except TSP, ACO has also been proven to be more efficient and effective in solving many problems, such as vehicle routing problem (Bullnheimer et al., 1999a, b; Gambardella et al., 1999; Reimann et al., 2002; Bella and McMullenb, 2004), quadratic assignment problem (Stützle, 1997; Maniezzo and Colorni, 1999; Maniezzo, 1999; Stützle and Hoos, 2000), scheduling problem (Colorni et al. 1994; Pfahringer, 1996; Stützle, 1998; Bauer et al., 2000; Jayaraman et al. 2000), and clustering problem (Parpinelli *et al.*, 2002a; Chiou, 2005).

2.3.4 FLC with GA and ACO

Since FLCs are highly non-linear systems which have high-dimensional, multi-model, and discontinuous response surface, the choice of optimization technique may not be obvious and easy (Li and Zhang, 1996). When designing a fuzzy logic controller, design parameters such as structure of fuzzy rules, choice of membership functions, etc. need to be determined. Traditionally, the establishment of fuzzy rules and membership functions has been mainly based on the experts' control experience and actions. However, converting experts' knowledge into IF-THEN rules or fuzzy sets is difficult because the investigation result is often incomplete and conflicting. Therefore, the task of automatically defining the fuzzy rules and membership functions for a concrete application is considered as a hard problem and a large number of methods have been proposed to generate the involved algorithms from numerical data, making use of different techniques such as ad hoc data-driven methods (Bárdossy and Duckstein, 1995), neural networks (Gupta and Gorsalcany, 1992; Esobgue and Murrell, 1993; Nauck and Kruse, 1993; Du and Wolfe, 1995; Nauck et al., 1997), fuzzy clustering (Babuška, 1998), GA (Wang and Mendel, 1992; Linkens and Nie, 1993; Bonissone et al., 1996; Hwang, 1998; Cordón et al., 2001), and ACO (Casillas et al., 2000, 2005; Parpinelli et al. 2002b). Due to the powerful ability of GA and ACO for solving hard combinational optimization problems, this study is interested in these two bio-inspired algorithms.

GA was first applied to the FLC by Karr (1991) to learn the fuzzy rules and determine the membership functions. In general, the integration of GA and FLC, called as GFLC, can be divided into four categories. (1) use of GAs to tune membership functions under a given set of fuzzy rules (e.g. Herrera, *et al*., 1995, 1998; Karr, 1991), (2) use of GAs to select fuzzy rules with known membership functions (e.g. Lekova, *et al.,* 1998; Chin and Qi, 1998; Thrift, 1991), (3) use of GAs to learn both fuzzy rules and membership functions simultaneously (e.g. Tarng, *et al*., 1996; Herrera, *et al*., 1998; Wang and Yen, 1999), (4) use of GAs to learn both fuzzy rules and membership functions in sequence (e.g. Karr, 1991; Kinzel, *et al*., 1994; Chiou and Lan, 2002, 2004, 2005; Chiou et al. 2003, 2005). In the first two categories, only one of the fuzzy rule and membership function components is learned and the other component is set subjectively; thus, the applicability of that GFLC is very likely reduced. In the third category, both components are learned simultaneously, thus the efficiency and effectiveness of that GFLC could be declined due to a very long chromosome needed. Therefore, this study attempts to employ the fourth category of GFLC with iterative evolutions to develop the FLC for transit preemption.

Integrating ACO into the FLC is still a novel thinking so far. Casillas et al. (2000) made a first attempt and Casillas et al. (2005) extends the original works done in 2000. In their papers, the rules selection problem is formulated into a combinational optimization problem with the capability of being represented on a graph. In this way, the problem is graphed as an assignment problem where a fixed number of rules are assigned one of the consequents with a probability that depends on the pheromone trail and the heuristic information of the ACO algorithm. Unlike the previous paper just assigning rule consequents, Parpinelli et al. (2002b) used ACO to generating crisp IF-THEN rule antecedent. In the graph of this problem, each node represents a condition that may be selected as part of the crisp rule antecedent being built by an ant. An ant goes round the graph selecting nodes according to a constraint satisfaction method to build its rule antecedent. The rule consequent is assigned afterwards by a deterministic method. Although the research mentioned above applied ACO to select the fuzzy rules of a FLC, the membership functions were still preset subjectively.

Based on this, this study aims to propose an Ant-Genetic based Fuzzy Logic Controller (AGFLC) which selects the fuzzy rules by ACO and tunes the membership functions by GA sequentially and iteratively.

2.4 Summary

No matter the results of experimental studies or analytical model and simulation studies, the TPS has been proven to be beneficial to the transit vehicles and passengers onboard. However, there are still few evidences that the implementation of transit priority strategies has little or no impact on the travel delays of other motorists on the competing approaches. The lack of explicit awareness of the cost-benefit has obstructed the widespread installation of TPS. Therefore, it is essential to well define the impacts of the implementing of priority strategies and develop a compromising control logic considering both the benefits and impacts to all vehicles involved. Since a fuzzy control system has excellent performance in dealing with the non-linear and complicated systems, it would be suitable to apply the FLC to establish a TPS control model. Furthermore, to equip the FLC an automatic learning mechanism for fuzzy rule selection and membership function tuning, this study integrates GA and ACO methods to develop the GFLC and AGFLC models. However, if only rule selection or membership function tuning is learned in constructing a FLC, the other one must be set subjectively and the generality of the FLC would be greatly declined. Therefore, this study develops an iterative evolution algorithm that could iteratively learn the rule selection and membership function tuning in sequence. The details of these two proposed models are described in the following chapter.