REALTRAN: AN OFF-LINE EMULATOR FOR ESTIMATING THE IMPACTS OF SCOOT

H. Rakha and M. Van Aerde
Department of Civil Engineering,
Queen's University, Kingston, Ontario,
Canada K7L 3N6

PHONE: (613) 545-2122
FAX: (613) 545-2128
E-MAIL: rakhah@jeff-lab.queensu.ca

ABSTRACT

This paper describes the development of an off-line emulation tool, entitled REALTRAN (REAL-time TRANsyt), that can emulate the SCOOT version 2.2 signal optimization logic. REALTRAN was derived from the TRANSYT-7F model (TRANSYT version 7F) by introducing various constraints into the optimization logic of TRANSYT-7F. These constraints allow the user to select optimization parameters that enable REALTRAN to operate in a similar fashion to the original SCOOT signal optimizer logic.

The REALTRAN model is currently intended to serve as an educational tool, but can, in the future, serve as a tool to fine-tune the operation of the real-time controls of the SCOOT system in a laboratory environment, where scientific and statistically valid testing and sensitivity analyses of the signal optimization algorithms can be performed. Alternatively, REALTRAN may be utilized to estimate off-line the expected benefits of SCOOT using location specific network and flow data.

(Keywords: Simulation, Real-Time signal control, ATMS, SCOOT, TRANSYT)

INTRODUCTION

Various field studies have indicated that various real-time Urban Traffic Control (UTC) systems, including the SCOOT system, are capable of attaining reductions in the range of 10% in the network travel time over conventional fixed-time signal control (*1*). However, it has been impossible to consistently achieve these reductions. It appears that the main reasons, for the lack of more extensive success of these real-time UTC systems, are related to the complexity of the problem, the variability of the link flows, and the inaccuracy of the vehicle detector measurements. To address each of the factors there is a need for tools to reliably fine-tune the operation of the real-time controls off-line in a laboratory environment. This environment would allow sensitivity analyses on the settings and signal optimization algorithms of such real-time UTC systems to be performed.

Paper Layout

Initially, the structure of the TRANSYT model is reviewed as this model forms the basis for both the SCOOT system and REALTRAN model. The SCOOT system is also described in the following section because the REALTRAN model can, depending on user specified inputs, attempt to replicate the SCOOT signal optimization logic. It must be noted at this point that any reference to TRANSYT in this paper refers to the general optimization logic of TRANSYT including that utilized in the TRANSYT-7F version.

Initially, the general concept of the REALTRAN model is presented. Subsequently, the details of the REALTRAN hillclimbing procedure and the associated use of high sensitivity parameters are presented together with the cycle length optimization process and how the cycle length, offset and phase split optimization procedures operate together.

In following section a simple example illustration is presented, due to the limited space available in the paper, in order to briefly illustrate how the REALTRAN model makes frequent but minor alterations to the signal settings such that the signal plan evolves incrementally towards the near optimum signal settings. A more detailed application of the REALTRAN model can be found in the literature (*2*). Finally, a summary and the conclusions of the paper are provided.

BACKGROUND

This section describes the more macroscopic concepts of the TRANSYT model followed by a microscopic description of the TRANSYT hillclimbing procedure. This latter detailed description is provided in order to have the reader appreciate the difference between the TRANSYT and SCOOT logic. In addition, this section provides a description of the SCOOT signal optimization logic prior to describing how REALTRAN can model the SCOOT logic.

Conceptual Description of TRANSYT

The TRANSYT model is perhaps the most widely used off-line signal optimization tool (*3*). The TRANSYT model was developed at the Transport Research Laboratory (TRL) and many versions have evolved. One of these TRANSYT versions is TRANSYT-7F, that was developed at the University of Florida (*4*).

TRANSYT is a macroscopic, deterministic, simulation and optimization model. The model requires the link flows and link turning proportions as inputs, and assumes them to be constant for the entire simulation period. The TRANSYT program simulates the traffic conditions for the duration of one complete cycle length and these conditions are assumed to be representative of all other cycles.

The TRANSYT model is macroscopic because the traffic module models the flow of vehicles as Cyclic Flow Profiles (CFP), rather than modeling individual vehicles. Specifically, the cycle length is divided into a number of short time steps, which are typically 1-5 s long. The CFP records platoons of vehicles as successive steps within the representative cycle and the shape of the CFP is calculated by the model, for each one-way flow in the study area.

The TRANSYT Hillclimbing Procedure

The TRANSYT program carries out a sequence of iterations between the traffic simulation module and the signal setting optimization module. For the initial signal settings, the traffic module estimates the Performance Index (PI) by simulating the traffic as it reaches each intersection in the network. Subsequently, alterations are made to the signal settings by the optimization module. These signal setting changes are sent to the traffic module, which alters the CFP that leaves each signal, which in turn affects the arrival profile at any downstream signals.

The TRANSYT program searches for the optimum signal settings using a two stage procedure. In the first stage, the optimum cycle length is found through a search, at user specified intervals, within a user specified range of minimum and maximum cycle lengths. Subsequently, in the second stage, the cycle length producing the lowest PI in the former search is investigated in further detail to determine the optimum offsets and phase splits for this cycle length using a hillclimbing procedure.

As the shape of the PI objective function vs. offset, phase split and cycle length is not always convex, local minimas may exist. Thus most conventional derivative methods would fail to find the global minimum and could frequently be caught in local valleys. The offset and phase split searches verify that the global minimum PI is found by using a combination of small, medium and large step sizes that usually move the optimizer away from a local minimum. While there are no absolute guarantees that the search will find the global minimum, the TRANSYT heuristic has during the past 25 years been found to yield a very practical tradeoff between accuracy and efficiency. Considerable work has been conducted to test and evaluate other search methods, however, no major improvements have been made to the TRANSYT search heuristic (*5,6*).

An Overview of the SCOOT System

The SCOOT real-time UTC software (*7,8*) uses a similar traffic simulation model as that used by TRANSYT (*9*). This simulation model is used on-line, however, every cycle by the optimizer to evaluate alternative signal timings and thus find the best signal settings based on the prevailing dynamic traffic conditions. The objective of SCOOT, as in the TRANSYT model, is to minimize the PI. Traffic is also modeled as a CFP, in the SCOOT traffic model, however, the time interval is fixed at 4 s and each link's inflow CFP is measured directly from the street using detectors, as opposed to being inferred from the turning movements of the upstream intersection.

The SCOOT optimizer updates the traffic signal plan on a cycle-by-cycle basis. In doing so, the optimizer uses the previous cycle's signal settings as a seed in the search for new timings and makes minor, but very frequent, alterations to these seed signal settings. The changes to the signal settings are made based on a restricted search for a minimum PI in the immediate vicinity of the seed signal settings, rather than searching exhaustively for a global minimum PI, as in TRANSYT. The SCOOT signal optimizer effectively uses an elastic coordination plan that stretches and shrinks the coordination scheme to match the latest situation recorded by the real-time cyclic flow profiles. The changes made to the current plan while minor, are frequent such that over time the plan evolves considerably without causing major disruptions to traffic.

The three key principles of the SCOOT real-time UTC system that make it different from the TRANSYT model, are as follows:

- to measure the cyclic flow profile in real-time as opposed to deriving it from upstream turning movements,
- to update an on-line model of queues continuously as opposed to only updating once, and
- to make incremental as opposed to global optimizations to the signal settings.

AN OVERVIEW OF THE REALTRAN CONCEPT

The REALTRAN model was developed by adding to and altering the TRANSYT-7F optimization logic in order to perform the following: (a) to iteratively estimate the optimum signal timings of a network of traffic signals for a time series of link flows, (b) to evaluate these optimum signal timings using a second set of link flows, and (c) to allow the user to specify constraints to the standard TRANSYT signal optimization logic.

The REALTRAN model, as does the SCOOT logic, involves the application of the TRANSYT optimization module every minute to an externally specified data stream. Each TRANSYT application is seeded with the signal timings that were found during the previous minute. The search can be constrained, depending on the user specified parameters, to only look for those new signal timing solutions which are very similar to the previous minute's signal timings. The use of a good seed plus the constraints on the optimization, can therefore significantly reduce every minute's computational requirements, because the optimizer starts from signal settings which are already very close to the optimum signal settings. Furthermore, as only minor changes are made to the signal settings, this approach can also avoid disruptions to the traffic during signal plan changes in a similar fashion to the SCOOT logic.

Details of the Optimization Procedure

The REALTRAN model can perform standard TRANSYT signal optimization, a restrained SCOOT-like signal optimization or any user specified restrained signal optimization depending on the user specified input parameters. The REALTRAN model, in simulating the SCOOT logic, makes three main restrictions to the hillclimbing process of the TRANSYT model, namely: constraining the offset optimizer by only allowing changes of a few seconds each optimization, constraining the phase split optimizer to only few second changes and making cycle length optimizations at intervals not less than 3 min using a limited range of potential cycle length choices.

The first key, to the potential success of this effort, derives from forcing the TRANSYT optimization to start the search for signal timings for the subsequent minute at the signal timings that were found at the conclusion of the previous minute.

The second key to the practical success of this effort, derives from the implementation of a user specified constraint on the number and size of the optimization steps that can be taken each minute to find improvements on this previous minute's timings.

While the above two steps towards making a real-time version of TRANSYT satisfy the offset and phase duration considerations of REALTRAN, a further addition to the logic was required to mimic SCOOT's changes in cycle length. Specifically, at user-specified time intervals (typically every 2-5 min) the model examines if the overall network PI can be decreased by moving to either a longer or shorter cycle length. The maximum amount of permitted change in the cycle length is again user specified and cycle length dependent in order to replicate SCOOT's different cycle length increments.

Details of Input Requirements

The REALTRAN simulation program requires four input files, namely; a master file, the standard TRANSYT input file, a link flow file to be used for signal optimization, and a link flow file to be used for the evaluation of the signal settings.

In the master file, the names of the various input and output files are specified. In addition, the cycle length increment thresholds, cycle length increments and the maximum number of steps to be used by the hillclimbing procedure are specified. This provides the user with the flexibility of testing different optimization constraints on the potential traffic signal settings. Furthermore, the "optimization link flow file" identifies the flows to be used by the optimizer to select the new signal settings each minute, while the "evaluation link flow file" identifies the flows to be used in evaluating these new signal settings. In this fashion, the model can either simulate a time lag in the optimization procedure or the fact that the link flows input to REALTRAN may be filtered flows, and thus differ from the actual flows.

The above input data requirements imply that the present version of the REALTRAN program is intended to be a simulation model that can replicate SCOOT's real-time controls, and not a real-time control system that is to be a competitor for SCOOT. What is important, however, is that the user can specify the cycle length thresholds, increments and frequency of full optimization either as those used by the actual SCOOT system, making the REALTRAN model simulate control algorithms in a similar fashion to the SCOOT signal optimizer or vary these parameters to study the impact on the PI.

SPECIFICS OF THE REALTRAN OPTIMIZATION MODULE

The main reason for the success of the modified hillclimbing routine, that is described in this section, is the elimination of the arbitrariness of the seed solution that is utilized to initiate the search for the next minute's signal timings. This section discusses the modified hillclimbing module in further detail and also illustrates how the use of high sensitivity parameters in the standard TRANSYT input file can help speed up the optimization process and assist in modeling mini-areas to mimic the SCOOT logic. This section also briefly describes the cycle length search process and the combined cycle length, phase split and offset optimization process.

The Modified Hillclimb Module

The incremental nature of SCOOT was mimicked in REALTRAN by setting up the modified hillclimbing module so it uses the previous interval's signal timing settings as the seed to initiate the search for the optimum signal settings for the following time period. Figure 1 demonstrates some of the details of how the modified hillclimbing module operates using a typical two step constraint, for all possible combinations of the shape of the PI curve, as indicated below.

Case (1) illustrates an optimization scenario in which an initial step reduces the PI and therefore leads the algorithm to proceed with another second step. This second step leads to a further reduction of the PI. However, as the limit of a maximum of 2 steps in a given search direction prevents the algorithm from proceeding any further, the signal settings that are found following the first two steps are retained and are considered to be the new approximation of the global optimum.

In Case (2), of figure 1, the first optimization step is again shown to reduce the PI. However, when the second step is taken in this same direction, the PI is shown to start to increase again. Consequently, it appears that the algorithm has found a local minimum, and the algorithm returns to the signal timing settings that were found following the first step as the new found approximation to the global optimum signal settings.

Case (3) illustrates how an attempted shift in the signal timings to the right results in an increase in the PI. Consequently, the algorithm reverses its search direction and doubles its step size to make a shift in the signal timings to the left past the initial signal settings and returns the step size to its original value. This move is shown to lead to a reduction in the PI, when compared to the initial settings. The limit of a maximum of two steps in any direction prevents the algorithm from proceeding any further in this direction. Case (4) is similar to Case (3), but only makes one step to the left.

Finally, the example in case (5) illustrates that a shift in either direction leads to an increase in the PI. Consequently, the optimum signal settings are considered to be retained by simply keeping the initial signal settings that existed when the search was initiated.

The above hillclimbing decision logic is used in both the offset and the phase split optimization process within REALTRAN using the minimum resolution if not specified otherwise in card type 4 of the standard TRANSYT input file. Utilizing card 4, one can investigate the effect on SCOOT of different step sizes. In addition, the maximum number of steps utilized by the REALTRAN model in each direction can be set by the user. This value is typically set to two in order to mimic SCOOT's minimum signal changes while maintaining the ability to escape local minimums.

Use of High Sensitivity Parameters

The sensitivity parameter sets a limit below which any downstream changes to the CFP are neglected. The base TRANSYT model permits the use of a Sensitivity Parameter Card (card type 6) in order to limit the extent to which the downstream effects, at other intersections, of a change in signal timings at a given intersection will be examined. The setting of these parameters affects the time required to execute the model. Because the SCOOT optimization logic only considers the flows arriving at the traffic signal in generating the cycle length duration and phase splits, and only considers the surrounding signals in estimating the optimum offsets, the use of high sensitivity parameters of 20% were utilized as the default in the REALTRAN model.

Search for the Optimum Cycle Length

In order to minimize any disruptions to traffic within SCOOT, the changes in cycle length are restricted to be very small. This objective is achieved within REALTRAN in three ways: by controlling the number of cycle lengths to be evaluated, by controlling the cycle length increment, and by controlling the frequency of cycle length optimization runs.

During each cycle optimization, the REALTRAN program uses the previous interval's cycle length as a seed. The minimum cycle length to be considered is then selected as the initial seed cycle length minus the user specified cycle length increment. Similarly, the maximum cycle length to be considered is chosen as the seed cycle length plus the cycle length increment. These optimizations are performed at a user specified interval and thus can be performed, for example, every 3 min, as is the case with the SCOOT system or at more/less frequent intervals.

The REALTRAN model can use up to 3 different cycle length increments depending on the cycle length. The model can therefore again replicate the SCOOT system's cycle optimization logic to a large extent.

EXAMPLE ILLUSTRATION

In order to briefly illustrate how the REALTRAN model can examine the impacts of a constrained SCOOT-like optimization for different traffic flow patterns, an extract of the results from a nine traffic signal grid network is presented in this section. Due to the limited space in the paper only a very brief summary of the results is presented, however, the details of the network and results can be found in the literature (*1,10*).

The network was simulated for a hypothetical sequence of five hours within which the flows experienced a peak in the east bound direction, followed by a peak in the west bound direction. Each minute, the REALTRAN optimizer optimized the signal settings using two optimization scenarios. In the first scenario, the REALTRAN model utilized the standard unconstrained TRANSYT full optimization every minute. Subsequently, in the second scenario the REALTRAN optimizer was constrained to emulate the SCOOT signal optimization logic. In order to simplify the illustration of the traffic flow pattern, only a sample of the link flows for 10 typical minutes during the simulation period, for the four approaches to the traffic signal located at intersection 5, is provided in table 1. For these sample arrival link flows, table 2 illustrates the signal settings that were selected by the REALTRAN optimizer for the two scenarios studied, namely, the TRANSYT and SCOOT emulations.

It can be noted in table 1 that, during this time period the flow on the west, south and north bound approaches to signal 5 remained constant while the flow on the east bound approach increased from 525 vph to 750 vph. This change represents approximately a 50% increase in flow in 10 min. While such an increase within 10 min may not be very likely to occur in practice, the intent was to investigate, by means of such a rapid hypothetical change in flows, the robustness and responsiveness of the SCOOT emulator.

It can be noted in table 2 that the TRANSYT emulator responded to these drastic changes in flows with major changes in the signal settings. Specifically, the cycle length changed from 52 to 72 s, which is equivalent to a 35% change, at the onset of the 40th min. In contrast, the SCOOT emulator altered the cycle length by only 4 s from 44 to 48 s following which the SCOOT emulator was restricted from performing another cycle length optimization for 3 min. Also, the TRANSYT emulator made a drastic change in the offset, from an offset of 0 s to an offset of 20 s, at the start of the 36th min, for a change in cycle length from 48 to 44 s, while the SCOOT emulator was restricted to minor alterations. The difference between the maximum and minimum offsets selected by the TRANSYT emulator was 20 s (0 to 20 s) as opposed to a 4 s difference for the SCOOT emulator (13 to 9 s). The TRANSYT emulator also varied the phase 1 duration from 23 to 47 s at the onset of the 40th min, which is equivalent to a 14 s variation, while the SCOOT emulator only varied the phase 1 duration from 24 to 26 s.

In comparing the PI at each minute, columns 6 and 11 of table 2, it is evident from, that initially the PI for the SCOOT emulator was lower. However, as the TRANSYT emulator was not restricted, this trend changed as the link flows varied. However, because the unconstrained optimizer made larger variations to the signal settings, it would cause major disruptions to the traffic and thus would add inefficiencies which are not accounted for in table 2. It must be noted based on these limited results, also, that the SCOOT emulator succeeded, albeit with a lag, in following the trend in the variation of cycle length, offset and phase split, thus allowing the signal timings to evolve over time in a similar fashion to the SCOOT signal optimizer.

SUMMARY AND CONCLUSIONS

This paper made a significant step towards addressing the need for a simulation tool that is capable of emulating the SCOOT optimization logic. In this paper the structure of such a model, entitled REALTRAN, was presented. The REALTRAN model is based on the well-known TRANSYT program, specifically TRANSYT-7F, by constraining the hillclimbing procedure within the TRANSYT model. The user can specify the maximum number of steps allowed in each direction and thus allow REALTRAN to model various constraining conditions that are different from the SCOOT default. The REALTRAN model permits the specification of a separate link flow file that is used to select the optimum signal settings, and an optional link flow file that is used to evaluate these signal settings. Furthermore, the user can specify, through the master file, both the cycle length increments to be evaluated and the cycle length thresholds separating various cycle length increment zones.

It must be noted that the REALTRAN program, like the SCOOT, SCAT and PRODYN traffic models, is built on the vertical queue model and thus cannot consider in detail the effect of downstream link congestion on the signal output. These models operate well as long as the network is not overly congested. However, they fail to model the effect of downstream congestion on the capacity of upstream intersections during queue spillback. In the case of SCOOT the queuing model is updated by queue measurements from the field. In addition, the REALTRAN model cannot model the re-routing of traffic in response to changes in signal timings.

REFERENCES

1. Boillot F., J.M. Blosseville, J.B. Lesort, V. Motyka, M. Papageorgiou and S. Sellam (1992), "Optimal Signal Control of Urban Traffic Networks," IEE International Conference on 'Traffic Road Monitoring and Control', pp. 75-79.
2. Rakha H., M. Van Aerde and E.R. Case (1991), "Experiments in Incremental Real-Time Optimization of Phase, Cycle, and Offset Times Using an On-Line Adaptation of TRANSYT-7F," Presented at the Engineering Foundation Conference on 'Traffic Management: Issues and Techniques,' Palm Coast, Florida, April.
3. Robertson D.I. (1969), "TRANSYT: A Traffic Network Study Tool", RRL Report No. 253, Crowthorne, U.K.
4. Courage K. and C. Wallace (1991), "TRANSYT-7F Users Guide,", University of Florida.

5. Timmermans W.F., P.P. Van Den Bosch, and J.J. Klijnhout (1979), “Improved Network Control by Using TRANSYT”, *Traffic Engineering and Control*, 21, pp. 353-356.

6. Foulds L.R. (1986), “TRANSYT Traffic Engineering Program Efficiency Improvement via Fibonacci Search,” *Transportation Research*, Vol. 20A, No. 4, pp. 331-335.

7. Hunt P.B., D.I. Robertson, R.D. Bretherton and R.I. Winton (1981), “SCOOT: A Traffic Responsive Method for Coordinating Signals,” TRRL Report LR 1014.

8. Hunt P.B., D.I. Robertson, R.D. Bretherton and M.C. Royle (1982), "The SCOOT On-line Traffic Signal Optimization Technique," IEE International Conference on 'Road Traffic Signalling', Publication No. 207, pp. 59-62.

9. Robertson D.I. and R.D. Bretherton (1991), "Optimizing Networks of Traffic Signals in Real-Time --The SCOOT Method," *IEEE Transactions on Vehicular Technology*, Vol. 40, No. 1, February, pp. 11-15.

10. Rakha H. (1993), "A Simulation Approach for Modeling Real-Time Traffic Signal Controls," Ph.D. Thesis, Queen's University.

Table 1: Link flows arriving at signal 5

Time into simulation (min)	Approach flows to signal 5			
	Eastbound (vph)	Westbound (vph)	Southbound (vph)	Northbound (vph)
35	525	350	500	250
36	550	350	500	250
37	575	350	500	250
38	600	350	500	250
39	625	350	500	250
40	650	350	500	250
41	675	350	500	250
42	700	350	500	250
43	725	350	500	250
44	750	350	500	250

Table 2: Signal settings chosen by the TRANSYT and SCOOT emulations (Signal 5)

Time	TRANSYT emulation					SCOOT emulation				
(min)	cycle	offset	phase 1	phase 2	PI	cycle	offset	phase 1	phase 2	PI
35	48	0	24	24	80.2	40	12	20	20	78.3
36	44	20	23	21	83.7	40	12	21	19	82.9
37	52	0	27	25	87.7	44	13	23	21	82.0
38	52	3	28	24	91.8	44	11	24	20	92.5
39	52	3	28	24	97.8	44	12	24	20	99.3
40	72	9	40	32	100.1	48	13	26	22	103.8
41	64	6	36	28	106.8	48	9	27	21	115.3
42	68	5	39	29	112.3	48	8	27	21	133.3
43	76	9	44	32	119.9	52	11	29	23	156.9
44	80	4	47	33	129.8	52	9	30	22	173.4

List of Tables

List of Figures

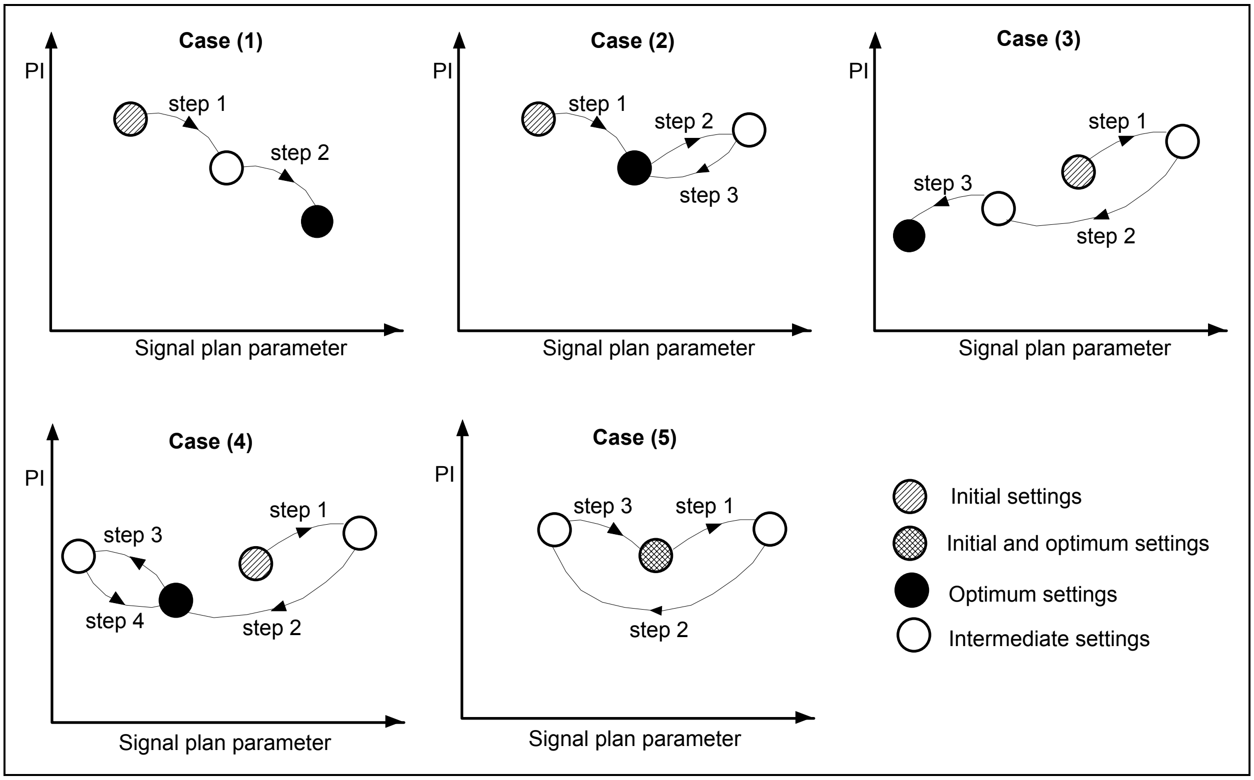

Figure 1: The modified hillclimb mechanism for a two step constraint