PERSONALIZED TRAINING PLAN OPTIMIZATION FOR CYCLISTS

NATTAPON KUMYAITO

A Thesis Submitted to the Graduate School of Naresuan University
in Partial Fulfillment of the Requirements
for the Doctor of Philosophy Degree in Information Technology
September 2018

Thesis entitled “Personalized Training Plan Optimization for Cyclist”

By Mr. Nattapon Kumyaito

has been approved by the Graduate School as partial fulfillment of the requirements for the Doctor of Philosophy in Information Technology of Naresuan University

Oral Defense Committee

.... Chair

(Associate Professor Punpiti Piamsa-nga, D.Sc.)

.... Advisor

(Assistant Professor Kreangsak Tamee, Ph.D.)

.... Internal Examiner

(Associate Professor Kraisak Kesorn, Ph.D.)

.... Internal Examiner

(Thanatorn Poeka, Ph.D.)

Approved

....

(Associate Professor Paisarn Muneesawang, **Ph.D.**)

Dean of the Graduate School

6 SEP 2018

ACKNOWLEDGEMENTS

I wish to gratefully thank Assistant Professor Thitirat Cheawsuwan, Ph.D. the Dean of the School of Information and Communication Technology, University of Phayao, for kindly permitting my leave to undertake this study.

My gratitude also to the staff at the Department of Computer Science and Information Technology, of the Faculty of Science at Naresuan University, for their time, interest and considerable assistance and advice on this program. Especially, I would like to thank my research supervisor, Assistant Professor Kreangsak Tamee, Ph.D., of the Department of CSIT for his invaluable guidance, supervision and constant inspiration throughout the study.

I also gratefully thank my family, my parents and brothers, for their sacrifices, encouragement, care, and support during this period. It would have been most difficult to complete this study and thesis without their great support.

I wish also to thank all my friends at CSIT, especially the members of the Intelligence and Information Science Research Group for their valuable suggestions, and the joyful moments that they provided: I will have precious memories of them and will keep them in my heart forever.

Lastly, I would like to thank the Graduate School for their support in publishing this thesis. Particularly, also, I thank Mr. Roy I. Morien, of the Graduate School, for his considerable assistance in editing the English expression and grammar in the thesis.

Nattapon Kumyaito

Title	PERSONALIZED TRAINING PLAN OPTIMIZATION FOR CYCLISTS
Author	Nattapon Kumyaito
Advisor	Assistant Professor Kreangsak Tamee, Ph.D.
Academic Paper	Thesis Ph.D. in Information Technology, Naresuan University, 2018
Keywords	Optimization, Personalization, Cycling training plan

ABSTRACT

Sports training is an essential element in the sporting career of any athlete. Without a correct and effective training program, no athlete can achieve the high level of success that most athletes aspire to. While training enhances both physical condition and mental condition, a poorly devised training program can not only fail to achieve the outcomes required but can cause actual injury to the athlete. A training program provides both fitness enhancement and fatigue. Thus, the training program must consider the physiological constraints of training monotony, chronic training load ramp rate, and daily TRIMP, to ensure that the cyclist does not become over-trained, with fatigue outweighing fitness gain. A training program based on Banister's Training Performance Interaction Model was adopted. Particle Swarm Optimization was proved to be the successful algorithm in devising an optimal training program. This was demonstrated in multiple simulations applying the PSO algorithm. Other techniques, including the Genetic Algorithm and heuristic search algorithms were included in our simulations but were found to be sub-optimal by failing to consider the physiological constraints. To ensure that the training plan algorithm that we devised can be used to create a personalized training program, we include the simulation of 20 cyclists who have different characteristic that reflect their starting level of fitness. This algorithm was shown to out-perform both the British Cycling's training plan and the Sufferfest™' s training plan. We consider, however, that further research opportunities are possible to further study and enhance sports training planning include improving parameters of Banister's model, an adaptive training plan, and mobile-specific optimization techniques also discussed.

LIST OF CONTENTS

Chapter **Page**

I INTRODUCTION 1

Background and motivation 1

The conceptual framework of the research 6

Research objectives 8

II THEORETICAL BACKGROUND 9

Principle of endurance sports training 9

Raising the athletic performance: Training-performance interaction model 17

The personalized training intensity zoning by heart rate 22

Physiological constraints 23

Computational intelligence: Scheduling techniques 25

III PROBLEM DESCRIPTION AND FORMULATION 37

Simulated cyclist 37

Training intensity zone and training session duration 39

Total periods of the training plan 40

Candidate solution formulation 40

Athletic performance evaluation 41

Physiological constraints 41

Problem's search space 41

IV HEURISTIC SEARCH ALGORITHM 44

Overview 44

Experiments 44

Results and discussion 50

Conclusion 54

LIST OF CONTENTS (CONT.)

Chapter **Page**

V GENETIC ALGORITHM 56
Overview 56
Experiments 56
Results and discussion 59
Conclusion 62

VI PARTICLE SWARM OPTIMIZATION 63
Overview 63
Experiments 63
Results and discussion 64
Conclusion 68

VII DISCUSSION 69
Personalization 69
Convergence 70
Computational time 74
Quality of solution 76

VIII CONCLUSION 80
Summary of research 80
Contribution of PhD research 81
Recommendations for future research 81

REFERENCES 83

GLOSSARY 92

BIOGRAPHY 96

LIST OF TABLES

Table		Page
1	Sample of cycling training plan	2
2	Summary of related works on scheduling a sports training plan	5
3	Coggan's training zones which depended on personal FTHR	23
4	Characteristics of the simulated cyclists	37
5	Simulated cyclist's heart rate zone	39
6	Training duration zone	40
7	Reduced number of sub-solution combination	49
8	Benchmarking between GA-based training plans and the UK training plan	61
9	All constraint violations in particular function evaluation times (FEs)	67
10	Computation time for different approaches	75
11	Quality of the best solution from different approaches	76
12	Comparison between British cycling's training plan and sufferfest training plan	78
13	Comparison between revised PSO training plan and sufferfest training plan	79

LIST OF FIGURES

FIGURES **Page**

1	Conceptual research framework	6
2	Super compensation	10
3	Training effect from different training effort	10
4	Sequencing training session and it consequences.	11
5	Volume vs Intensity vs Performance.	13
6	Physical response through time by a training session at 131.78 training load.	21
7	Cycle of processes in genetic algorithm	29
8	Example of search space in constrained optimization problem	34
9	Tri Surface plot of 100 samples of candidate solution	42
10	Athletic performance risen over time	46
11	Selecting training sessions for heuristic search algorithm	49
12	Rising trend of athletic performance by different sub-problem	52
13	TRIMP by different sub-problem	53
14	Chronic training load ramp rate for each cyclist's training plan by different sub-problem size	53
15	Monotony constraint for each cyclist's training plan by different sub-problem size	54
16	Athletic performance for each cyclist's training plan by different sub-problem size	54
17	Comparison of TRIMP value between the UK training plan and GA-based training plans	60
18	Comparison of athletic performance between the UK training plan and GA-based training plans	60
19	Comparison of PSO training plan against British cycling's training plan	65

LIST OF FIGURES (CONT.)

FIGURES **Page**

20 Performance corresponding to particular training plans constraints handling.......... 66

21 Convergence of summary constraints violation in early iterations...... 67

22 Daily training Load in each cyclist training plan.......... 69

23 Performance profile based-on average of best objective value by FEs 72

24 Comparison of population's error by FEs.......... 73

25 Computation time of different approaches.......... 75

26 Athletic performance from different approaches.......... 77

27 Result comparison between British cycling vs Sufferfest application.......... 78

28 Result comparison between customized PSO and Sufferfest application.......... 79

CHAPTER I

INTRODUCTION

Background and motivation

Sports training is one aspect of their sporting activity to which cyclists pay most attention. Training enhances both physical condition and mental condition, both of which are needed to ready themselves for competitions. Especially in endurance sports like road race cycling, race events test participants to the edge of their physical and mental capability. The Tour De France, for example, the preeminent world famous road cycling race, has 21 stages with total overall distance 3,360 km. In each stage, participants must race for 4-6 hours on average, which may include extreme weather conditions and tough terrain. The participants who race in this kind of event need training that emphasizes endurance, to raise their athletic performance to the highest level possible.

Banister et al. (1991) have proposed a mathematical model which describes training patterns for the enhancement of endurance capability. This model is based on the interaction between a program of daily training, and athletic performance. The daily training load is not without its difficulties, in that it can have two quite opposite effects to the physical condition of the athlete. The positive effect is the enhancement of physical fitness, whereas the negative effect is the inducement of fatigue.

These opposite effects are gained, or decay, over time, at different coefficients. The coefficient of the rate of both fitness gain and fitness decay is lower than the coefficient of the rate of fatigue gain and the rate of fatigue decay. The Banister model explains why the performance of cyclists who train too hard, but have insufficient rest, might drop, contrary to what might be expected. On the other hand, successful cyclists usually have a good training pattern which includes an appropriate daily training load but also sufficient rest periods.

For the best training outcomes, cyclists need to apply the training-performance interaction model for scheduling their training plan in advance. Having a well-constructed training plan brings several benefits, such as promoting training focus and, importantly, avoidance of overtraining. Cyclists who train according to such a training plan can focus on each specific training session because they know how and when to train, recognizing that their training pattern will result in a good progression in raising their athletic performance. In addition, cyclists who follow a disciplined training plan have less chance of training too hard, which leads to overtraining, damaging their progress. A sample training plan is illustrated in Table 1.

Table 1 Sample of cycling training plan

Week	Mon	Tue	Wed	Thu	Fri	Sat	Sun
Week 1 7 Hrs	1 Hr - warm up - FTHR testing	Rest	2 Hrs - Endurance training @HR70%max	Rest	2 Hrs - Endurance training @HR70%max	Rest	2 Hrs - Below FTHR @HR90%max - Interval training 2x15min
Week 2 8 Hrs	1 Hrs - Recovery Ride @HR50%max - Pedal technique	Rest	2.5 Hrs. - Below FTHR @HR90%max - Interval training 2x20min	Rest	2 Hrs - Below FTHR @HR90%max - Interval training 6x(6+4min)	Rest	3 Hrs - Endurance training @HR70%max
Week 3 9 Hrs	1 Hrs - Recovery Ride @HR50%max -Pedaling technique	Rest	2.5 Hrs - Below FTHR @HR90%max - Interval training 2x12 min	Rest	2.5 Hrs - Race pace	Rest	2.5 Hrs - Below FTHR @HR90%max - Interval training 3x20min
Week 4 7 Hrs	Rest	2 Hrs - FTHR @HR90%max - Interval training 1x15min	Rest	Rest	2 Hrs - Below FTHR @HR90%max - Interval training 2x15min	Rest	3 Hrs - Below FTHR @HR90%max - Interval training 2x15min

A good sports training plan normally is scheduled by experts, such as sports-specific coaches or sport scientists. Often, though, cyclists have limited access to this level of expertise and other expert services, either because it is unavailable, or its cost is too high for the athlete to afford; especially amateur athletes. An affordable sports training program for cyclists can be found, for free, published by some reliable sports organizations or software applications that are freely available on the Internet, or available from application stores. These training plans do not suit everyone, however. Cyclists need to personalise these training plans, which is difficult as ordinary cyclists usually have little knowledge of sports science, and making adjustments might lead the cyclist to overtrain and / or achieve only suboptimal athletic performance.

Another issue that faces ordinary cyclists when attempting to design their personal sports training plan is their inability to correctly consider the physiological constraints necessary, and to take these into account. These constraints determine the appropriate training pattern for the cyclist, and should serve to remind the athlete on avoiding injuries or overtraining.

Physiological constraints such as monotony, chronic training load ramp rate (CTL ramp rate), and daily training load have been identified. In our study as the constraints necessary to support cyclists in their training, and avoiding injury caused by overtraining. All these physiological constraints are explained in detail in the 'Physiological Constraints' section below.

We refer to the techniques used by a computer program to formalise a training plan as the computational intelligence inherent in the program. This is what gives the computer the appearance of expertise. A training program is considered as a scheduling and planning problem, able to be produced by the computational intelligence of the software. It is the computational intelligence techniques, and its various components and aspects, particularly the physiological constraints imposed, that are the focus and subject of this thesis

Researchers have studied and developed computer systems intended to create personalized cycling training plans to emulate the decision making of sports science experts. From articles we have reviewed, illustrated in Table 2. We can categorize these into 2 groups; local search optimization techniques and machine learning techniques. Local search optimization techniques are discussed in Fister et al. (2015)

who proposed what they termed the Bat Algorithm for planning sport training sessions. They described this as an evolutionary algorithm, taking inspiration from the behavior of a species of micro-bat and its ability in orientation and prey finding. Their study used real training log data as the initial population data and searched for the optimal training plan with minimal errors. This plan was then compared against a training log selected by the coach. Elsewhere, Perl (Perl, 2001) applied the genetic algorithm (GA) to automatically create a training plan, and Brazostowski (2015) proposed a model to analyse the interaction between the performance of the athlete and the training proposed in a training plan. This interaction model was implemented as a dynamic programming system.

Machine learning techniques were the subject of work by Mezyk et al. (2011) who combined a fuzzy model with an immune algorithm, terming it the Improved IFRAIS (Alves et al., 2004). Rygula (2005) developed a computer system that uses artificial neural networks (ANN) to discover the influences that improve athletic performance.

By describing the factors that are important when creating a sports training plan, by using computational intelligence, the objective of our research was to analyze the sports training planning problem and to design a computer algorithm capable of generating a personalized sports training plan, taking into account the essential physiological constraints. We expected our system to be able to support a cyclist in training by providing a high quality training plan that minimizes the risk of overtraining and at the same time raises their athletic performance to a level necessary to enable them to compete successfully in their target racing event.

Table 2 Summary of related works on scheduling a sports training plan

Related works / Topic	Local Search Approaches			Machine Learning Approaches	
	Fister et al (2015)	Peri (2001)	Brzostowski (2015)	Me,zyk et al (2011)	Rygula(2005)
Techniques	Bat Algorithm	Genetic Algorithm	Dynamic Programming	fuzzy modeling + Immune algorithm (improved IFRAIS)	Artificial Neural Network
Outcome	Training Plan	Simulation Software Training Plan	Training Plan	"IF(fuzzy condition) THEN (class)" rule	Model of Training Loads
Constraint Handling	Coach-defined training session	Fined tuning model's parameters	Not presented	Not presented	Not presented
Pros	• Training plan with minimizing error	• Convenient tool to create and improve the training plan through UI • Automatic generates training plan	• Expertise is not required in the training planning.	• Increasing the quality of training units. • Support trainers in planning more optimal trainings	• Express the strongest influencer on the increase of score in 16-17yr girl runners • Express a model of optimal controls.
Cons	• Coach is needed • No mechanism for personalization • Physiological constrained optimization is not presented	• Coach is needed • No mechanism for personalization • Physiological constrained optimization is not presented	• No mechanism for personalization • Physiological constrained optimization is not presented	• Some marker in collected data not quite reliable, for example, a training stimulus. • Training plan is not presented in this work. Thus, users need to create their own plan regards to research findings.	• Training plan is not presented in this work. Thus, users need to create their own plan regards to research findings.

The conceptual framework of the research

We have identified the steps necessary to achieve our object: the design and creation of a computer algorithm capable of generating a personalized sports training plan. These steps are summarized in the conceptual research framework illustrated in Figure 1.

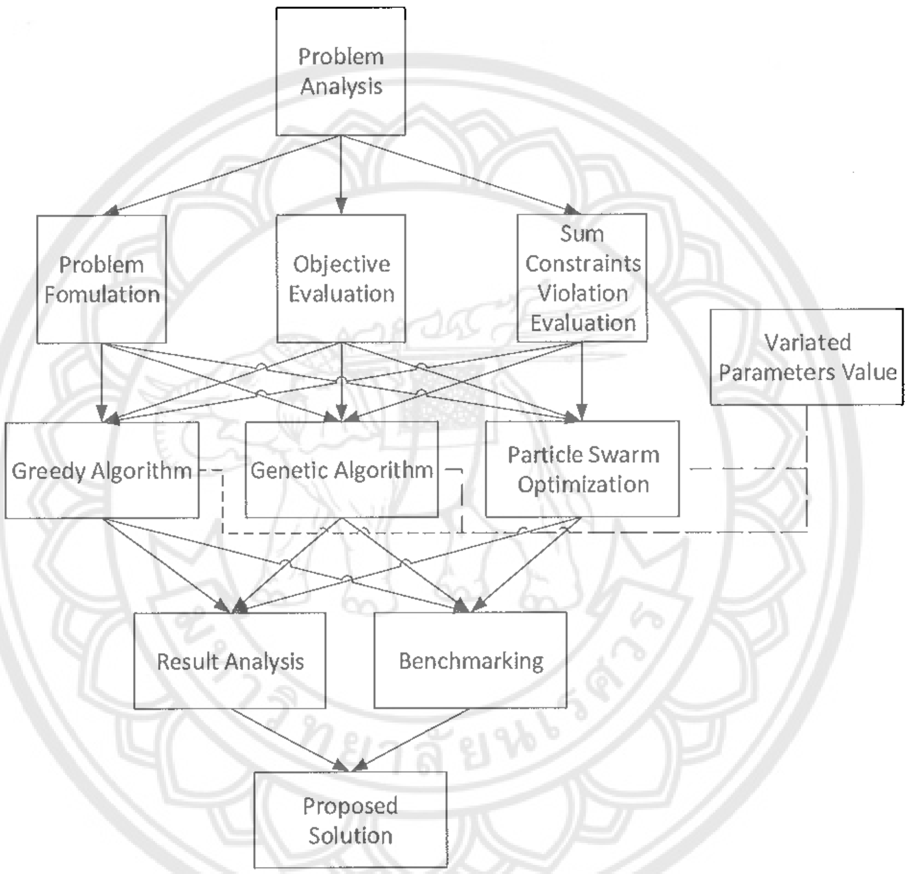

Figure 1 Conceptual research framework

We identified appropriate sport science articles and books with information relating to the problem of planning a sports training problem that ensures that the physiological constraints are acknowledged. We focused on the base training phase for cycling training. The objective of evaluation methods and the sum constraints violation functions were determined from the information found in the literature. The data set for our experiment was derived from simulated cyclist data, which included a simulated cyclist profile including gender, age, heart rate at resting state, functional threshold heart rate and maximum heart rate. This profile was then populated with the simulated data of 20 cyclists, from which were derived personalized training zones determined according to the characteristics of Coggan's training zones (Coggan, 2003). Training zones are training effort levels which are determined by the cyclist's heart rate.

For our study, we generated a training plan for an 8-week (56-day) training schedule. This is an appropriate period for the foundation training of any cyclist intending to participate in most endurance racing events (Seiler, & Tønnessen, 2009).

Having defined the evaluation function and the sum of constraints violation function, the problem was formulated according to each computational intelligence approach, and was based on deterministic optimization techniques and stochastic optimization techniques. We examined deterministic optimization techniques that applied a heuristic search algorithm. Heuristic search algorithms are an efficient technique that successfully solves many optimization problems with less computational cost. We also examined several stochastic optimization techniques, including the genetic algorithm and particle swarm optimization. These chosen techniques are well known by their efficiency and high quality solutions to many optimization problems.

All techniques studied are compared, later in our discussion, against each other, to select the most suitable technique for the sports training planning problem. In addition, the reported training plans produced when using these techniques are benchmarked against training plans from reliable organizations such as British Cycling in their publication (Britishcycling, n.d.), and a commercial version of the Sufferfest application (Henderson, & Cassin, n.d.).

We then chose the approach that produced the best practical training plan that enhanced the athletic performance to the highest extent. The chosen approach was analyzed, and is discussed below from several points of view.

Research objectives

1. Discover the optimization technique that best suit a cycling training plan problem.

2. Create a personalized cycling training plan that regards to cyclist current fitness.

3. Create a cycling training plan that recognized physiological constraints for minimize the risk of overtraining.

CHAPTER II

THEORETICAL BACKGROUND

Principle of endurance sports training

Improvement in athletic performance: super compensation

Our physical body is essentially a biomechanical device that reacts to upcoming phenomena that affects it. When we are bleeding, for example, the platelets that flow along blood vessels will close the wound and stop the bleeding.

Training is a phenomenon that places stress on the body, weakening the body. However, recovery occurs after a period of rest. In this period the physical body adapts to overcome the stress of the workload applied during the training effort. After rest, and recovery, the athlete feels stronger and is able to prepare for upcoming training sessions. However, if training is stopped or discontinued, the body loses the strength gained in the prior training, and reverts back to the basic fitness level.

This phenomenon is the super compensation effect demonstrated by Ackland (1999), shown in Figure 2.

When a cyclist exercises, his performance tends to drop from fatigue, which is the by-product of the effort expended in the training session. By resting after finishing the training session, the athlete's physical body recovers, at which point the athlete's physical body has adapted to the previous training workload resulting in a raise in their athletic performance, enabling them to proceed to further, possibly more strenuous, training sessions. If the cyclist, in this case, stops training for any significant period of time, their performance will drop to the basic level. The details of this concept of super compensation will be described in the Training-Performance Interaction Model section.

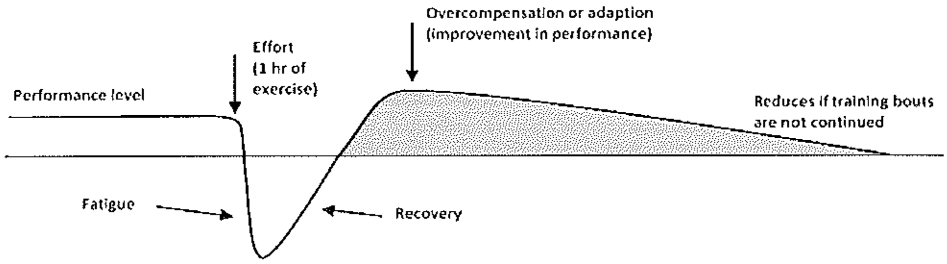

Figure 2 Super compensation

By the overcompensation that has been described in the previous section, we know that athletic performance can be raised by a proper program of continuous training with appropriate rest periods. Therefore, a cyclist whose objective is to raise his performance to a high standard may spend more effort in training with the hope and intention of achieving that goal. In Figure 3, different training efforts are graphed, showing the different supercompensation effects resulting from each.

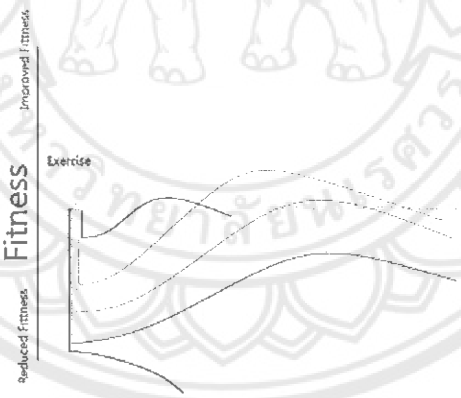

Figure 3 Training effect from different training effort

The green line is considered as the optimal training effort because it raises the performance level the most as compared to the others. The blue line shows the performance trend from an easy training effort, and the orange line shows the performance trend from a heavy training effort. Both the easy and heavy training efforts raise the cyclist's performance above his basic level, but, ultimately, the outcome is a much poorer performance than is achieved from the optimal training effort. Negative outcomes arise from the red black training efforts, which are examples of bad training efforts that result in a consequential drop in athletic performance to below the basic performance level.

This clearly indicates that it is necessary to create a sports training plan that correctly takes into account, and utilizes, the supercompensation effect, to raise athletic performance in a prolonged training program, which should be planned in advance. Proper sequencing of training sessions that alternate between heavy sessions, easy sessions, and rest sessions, will influence the consequential athletic performance and achieve the objective of high performance (Figure 4 (a),(b),(c),(d) & (e)).

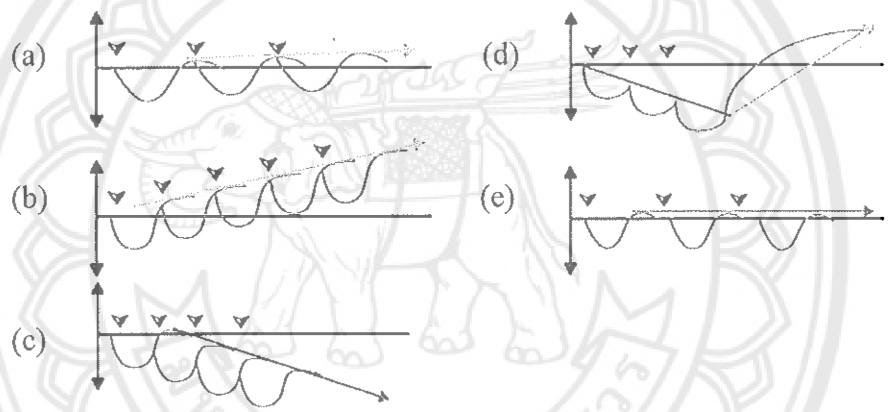

Figure 4 Sequencing training session and it consequences.

(a) relative long rest session between training sessions;

(b) optimal sequence;

(c) very short resting session between training sessions;

(d) very short resting session between training sessions followed by long resting period;

(e) long resting period between training sessions.

Training sessions sequenced as illustrated in Figure 4(a) raise performance only slightly above the basic level because the training sessions are being undertaken at a time at which the cyclist has lost their original fitness level, perhaps having stopped training for a period. This training sequence is considered as a suboptimal training sequence that cannot achieve a significant rise in performance. It is, however, appropriate for a cyclist who intends to maintain their current, satisfactory, fitness level or as preparation for subsequently undertaking heavier training sessions.

Figure 4(b) illustrates a well prepared training sequence in which the cyclist ramps up their training intensity with the objective of gaining a higher level of physical fitness. Each training session takes place at the time that the cyclist has almost fully recovered from previous training sessions. This kind of training sequence make cyclist feel for the accumulated tiredness from incomplete recovery. This period is very importance and need a well monitoring to avoid the risk of become overtraining. However, cyclist will significantly become stronger after resting just for a couple sessions.

If the cyclist undertakes a heavier training session when they are still suffering fatigue from the previous training session, the cyclist runs the risk of overtraining, which is potentially damaging to their progress. This training session sequence is shown in Figure 4(c).

Figure 4(d) illustrates the better training sequence pattern where the cyclist follows a strenuous training session with a good quality rest period. In this training sequence, athletic performance is accumulatively raised, which is the desirable outcome required.

Figure 4(e) shows a training sequence that includes easy training sessions, each followed by a sufficient resting period. This sequence is suitable for a cyclist whose objective is to completely clear muscle soreness, that is, fatigue, from a previous heavy training session while still maintaining their current fitness level. This training pattern is normally undertaken in the off-season of competition, as a fitness maintenance program.

Designing an annual training plan

In an annual training program, that is a plan for a full year ahead, the scheduled processes should allow the cyclist to continually increase the training load to their body constantly over the year. This intention to constantly increase their fitness level will be accompanied by training fatigue, and these must be balanced out to ensure that the cyclist does not overtrain or detrain. The two main factors that the cyclist should monitor in their training are training intensity and training volume, both of which influence the training load of any particular training session which can affect the upward trend of athletic performance.

Training intensity is the effort expended in a specific training session while training volume is the amount or extent of training in a specific training session. Ackland (1999) suggested that a cyclist should have a training pattern as illustrated in Figure 5.

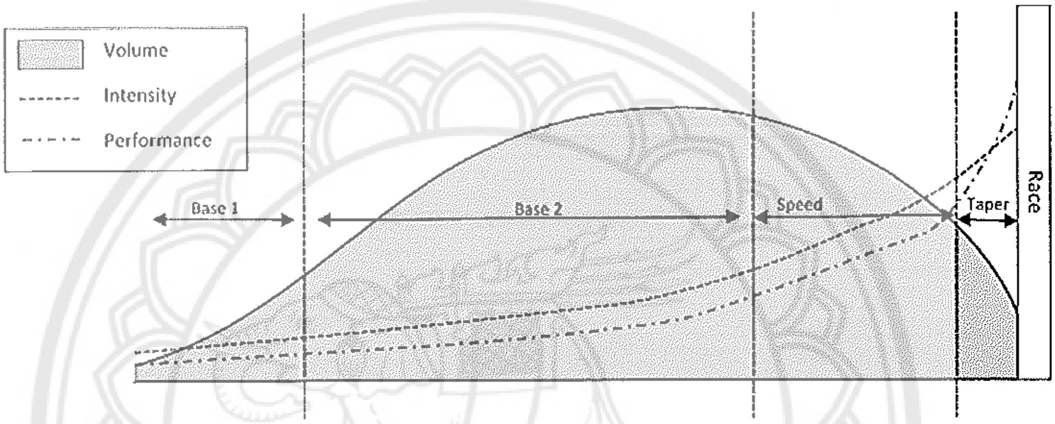

Figure 5 Volume vs Intensity vs Performance.

Cyclists should undertake an easy training program at the beginning of the season. This will have low volume and low intensity, and is the more appropriate program for initially developing a solid physical condition in the body, which would include aerobic capacity and tendon, joint, and muscle strength that will be needed in subsequent training. The training volume and training intensity will increase over time during those initial period, and, once a strong base level of fitness is achieved, the next training period will impose higher levels of training intensity. In these more intense training periods, greater stress on the body will be experienced, so each training session should be limited in volume and more resting time must be introduced to avoid overtraining. Training by this pattern raises neuromuscular strength which is converted to power and speed. The last period before race event should be the shortest, in which training volume is tapered off, but intensity is maintained. By this means, accumulated fatigue is reduced or overcome, while maintaining the high level of fitness previously achieved. The conclusion is that, if the cyclist follows this training pattern accurately, they should be at their strongest and highest state of readiness for their target racing event.

Three phases in a training plan

Bompa, & Buzzichelli (2015) recommended that an annual training plan should be divided into three phases: the preparation phase, the competition phase, and the transition phase.

1. Preparation phase

For endurance-dominant sports, athletes normally rely on anatomical adaptation with some sub maximum strength training. Cyclists particularly, are advised to train mainly at low-intensity and at a steady pace, but also include some speed work such as interval, time trial and sprints. Suring this phase the cyclist should aim for anatomical adaptations which will be needed for training in the subsequent phases. The intensity and volume should be gradually increased to strengthen the physical body.

Anatomical adaptation is the objective of strength training that is intended to elicit a progressive adaption of the cyclist's anatomy rather to achieve an immediate overload. Anatomical adaptation emphasizes strengthening the body and good coordination of the skeleton and the muscle so as will prevent the need for rehabilitation. The main physiological objectives of this training are:

1. To strengthen the tendons, ligaments, and joints, which can be achieved through a higher volume of training than in the remainder of the year (we are discussing a year-long program here).

2. Increase bone strength and propagation of the connective tissue.

As well, in this initial phase, the objective is to improve cardiovascular fitness, strengthening of muscle, and to practice neuromuscular coordination for strength movement pattern, or muscle memory. Even though this training does not focus on increasing muscle size, that result still may happened.

The strength of tendons is increased by implementing a time under tension per set that falls between 30 and 70 seconds (the time under tension that sees the anaerobic lactic system as the main energy system). The lactic acid releases the hydrogen ions that stimulate the release of growth hormone and collagen synthesis. These circumstances are also stimulated by eccentric load (Babraj et al., 2005;

Crameri et al., 2004; Doessing, & Kjaer, 2005; Kjaer et al., 2005; Kjær et al., 2006; Langberg et al., 2007; Miller et al., 2005). For this reason, the majority of the time under tension is spent in the eccentric phase of the exercise (3 to 5 seconds per repetition). Muscular balance is achieved both by using an equal training volume between agonist and antagonist muscles around a joint and by making greater use of unilateral exercises than of bilateral ones.

Maximum strength depends on many factors including the diameter of the cross-sectional area of the muscles, the ability to recruit fast-twitch muscle fibers, their frequency of activation, and the capacity to concurrently call into action all the related muscles involved in a given movement (Howard et al, 1985). These factors involve both structural and neural flow changes that happen as a function of training with moderate weights lifted explosively, as well as heavy loads that close to 90 percent of 1RM or more. These adaptive responses can also be triggered by eccentric training with loads greater than 100% of 1RM, although its practical application is limited to very few conditions.

A cyclist could benefit from traditional maximum strength training methods. For example, performing heavy weight with long period resting (3 to 5 minutes) between sets. However, to increase the lifting weight in a training bout over the time, the key factor is intermuscular coordination training. With time, as the nervous system learns the movement pattern, fewer motor units get activated by the same weight, therefore, leaving more motor units available for any activation by higher weights. In addition, the concentric action should be done with explosive movement in order to activate the fast-twitch muscle fibers and to achieve the highest specific hypertrophy.

Thus, intermuscular coordination training is the preferred method for general strength. It provides the basic level of physical strength for the training in later phases in which intramuscular coordination is trained by using higher loads and longer resting period intervals. In addition, periodization of strength continually stresses and engages the nervous system by altering several factors includes training loads, sets, and training methods.

2. Competition phase

In this phase, the objective is to maintain the same level of training load but minimize the training volume for clearing the fatigue and maintain high performance. The training regime to achieve this goal is replacing some long and slow steady stage training sessions with some maximal or sub maximal strength training sessions. The cyclist might need to focus on converting the strength and power that was gained from the preparation phase into strength and power which is specific to target sports movement.

Based on particular sport, three fundamental options can be achieved after a maximum strength phase of the training. these fundamental options are: conversion to power, power endurance, or muscular endurance. Conversion to power or power endurance can be accomplished through moderate to heavy loads (8% to 40% of 1RM). Keep in mind that the lifting movement should be done as quickly as possible, with altering the duration of the sets. For engaging the nervous system, speed training and upper-body or lower-body plyometric training can improve cyclists' ability to recruit and engage the high-powered fast-twitch motor units. the solid foundation of maximum strength is needed for maximizing the force production rate. As the matter of fact, even training with high loads that moved at low speed has been shown to transfer to power gained if the cyclist attempts the movement as fast as possible (Behm, & Sale, 1993).

Depending on the demands of the sport, muscular endurance can be trained for different period of durations. Short duration muscle endurance is the main energy system is anaerobic lactic, whereas medium and long muscle endurance are predominately aerobic. Development of muscular endurance requires training more than 15 to 20 reps per set; indeed, it can require as many as 400 reps per set, implemented along with metabolic training. In fact, metabolic training and muscular endurance training pursue similar physiological training objectives.

The muscular contractions consume energy from the combined efforts of three energy systems: the anaerobic alactic, the anaerobic lactic, and the aerobic. Training for muscular endurance requires expanded adaptation of the aerobic and the anaerobic alactic systems. The objectives of aerobic training include improvement in several biological parameters, such as metabolic parameters, which result in greater use of fat as energy and an increased rate of removal and reuse of lactic acid, biochemical parameters, such as increased mitochondria and capillary density which result in greater diffusion and use of oxygen, and heart efficiency. Raising the neuromuscular and cardiovascular systems physiologically, biochemically, and metabolically provides invaluable benefit to athletes in many endurance sports. For athletes in muscular endurance sports, maximum strength training must be followed by a combination of specific metabolic training and specific strength training in order to prepare the body for the demands of the sport.

3. Transition phase

The intensity and volume drop in this phase to allow full recovery in physical and mental conditions. Cyclists may take a long vacation during which they perform easy workouts that enhance the recovery process and still maintain some athletic performance.

Once the neuromuscular system has been adapted to maximize athletic performance, it is time to test the gains. Unfortunately, most cyclists and coaches work hard strategically as the competitive season approaches but discontinue strength training when the season begins. As the matter of facts, maintaining a strong and stable basic formed during precompetitive phase requires the cyclist to continue strength training during the competitive season. Lack of success to plan at least once-a-week session that dedicated to strength training may results in declined performance or early onset of fatigue as the season goes on.

Falling down and then attempting to recover is always harder than staying up. Periodization of strength includes planning phases to optimize biological adaptation and planning to maintain the benefits for the going on season. When the season is over, fatigued cyclists can take 2 to 4 weeks off to regenerate their physical condition and mental condition.

Raising the athletic performance: Training-performance interaction model

In order to train precisely according to training plan, accurate measurement of training loads is need. Cyclists need some techniques to quantify their training load and to subsequently estimate their performance achieved from their training plan. Two data elements that are widely used to quantify the training load are power data and heart rate data. Power data is mostly used by professional cyclists who demand very high precision training while other cyclists more often use heart rate data that is more affordable to acquire and is reliable. Since the cycling power sensors are very high cost instrument which is not widely adopted in our country, our study apply Banister's methods which rely on more affordable sensors, heart rate monitor sensors.

Training load quantification by power data: Training stress balance (TSB) model

The Training Stress Balance model (Coggan, 2008) is an extended modification of the Banister's model which emphasize only on the changes in athletic performance. The Training Stress Balance (TSB) model uses the terms Chronic Training Load (CTL), Acute Training Load (ATL), and Training Stress Balance (TSB) for "fitness", "fatigue", and "performance", respectively. Both CTL and ATL are based on Banisters Training Impulse (TRIMP) model, which shows the effect to athletic performance is greater on CTL than on ATL. In addition, the TSB model assumes that the effect on a given workout reducing over time, but the effect lasts longer on CTL than on ATL. The TSB model estimates athletic performance by following equation:

$$Athletic\ Performance = CLT - ATL \tag{1}$$

where the CTL and ATL that result from a certain training load have different decay rates. The CTL and ATL from a series of training sessions are calculated by following equations:

$$ATL_i = ATL_{i-1} + \frac{(TSS_i - ATL_{i-1})}{\lambda_a} \tag{2}$$

$$CTL_i = CTL_{i-1} + \frac{(TSS_i - CTL_{i-1})}{\lambda_f} \tag{3}$$

where λ_a and λ_f are calculated by:

$$\lambda_f = 2/(N_f + 1) \tag{4}$$

$$\lambda_a = 2/(N_a + 1) \tag{5}$$

where N_a is the time decay constant for ATL and N_f is the time decay constant for CTL. Normally N_a is 7 days and N_f is 42 days.

The Training Stress Score (TSS) (Friel, 2009) in equation 2 and equation 3 represents a workload from a training session. It is a product of the workout's intensity and duration. As either of these increases, TSS also increases. The equation for TSS is:

$$TSS = (duration \times NP \times IF) / (\mathrm{FTP} \times 3600) \times 100 \qquad (6)$$

where duration is the training duration in seconds, the intensity factor (IF) is the percentage of the cyclist's functional threshold power (FTP) where FTP is the best average power that the cyclist can maintain for a one-hour race or test, 3600 is the number of seconds in an hour. Normalized power (NP) can be calculated by these following steps (Allen, & Coggan, 2012):

1. Calculate a 30-second rolling average of the power data
2. Raise these values to the fourth power
3. Average the resulting values
4. Take the fourth root of the result

Professional cyclists mostly rely on a power sensor which is considered the most precise sensor for estimating the training load. The power sensor measures the force that the cyclist applies directly to the bicycle. The power sensor output can be used to estimate the training load, which is used to estimate athletic performance later.

However, the benefit in precise measurement comes at a very high cost. Reliable power sensors are normally priced around US$100 to US$500. Many cyclists cannot afford this very high cost hardware. The low cost but reliable alternative data that represents the intensity of training is the heart rate data collected by a heart rate monitor. These are the measurement and measuring device considered and tested in our research. A heart rate monitor has a cost of between US$10 and US$30, well within the budget of most cyclists.

How the training load is quantified, and how the athletic performance is evaluated, when using heart rate data, is described in the following sections.

Training load quantification by heart rate data: Training IMPulse (TRIMP)

In our research, our emphasis was on cyclists who would be unable to afford a power meter which, as indicate already, is a high cost piece of equipment. Because we use a heart rate monitor in estimating the training load, we were able to use TRIMP to quantify the training load based on the heart rate data measured as an output of a specific training session.

Applying TRIMP, the training load for a specific training session can be calculated. We simulated a training session occurring once a day, and calculated the arbitrary TRIMP value of the training day i in Equation 7,

$$TRIMP_i(d_i, \overline{hr_i}) = d_i \cdot {}^{norm}HR(\overline{hr_i}) \cdot e^{y \cdot {}^{norm}HR(\overline{hr_i})} \quad (7)$$

where d_i is the duration of the training session in day i (minute), y is the model constant (1.92 for males and 1.67 for females (Banister, 1991)), $\overline{hr_i}$ is the average heart rate throughout a training session in day i, and ${}^{norm}HR()$ is the normalized value of $\overline{hr_i}$ throughout a training session in day i, which is determined by Equation 8.

$$^{norm}HR(\overline{hr_i}) = \frac{\overline{hr_i} - {}^{resting}HR}{^{max}HR - {}^{resting}HR} \quad (8)$$

where $\overline{hr_i}$ is the average heart rate during a training session in day i, ${}^{resting}HR$ is the cyclist's heart rate at resting state, and ${}^{max}HR$ is the cyclist's maximum heart rate.

Banister's training-performance interaction model

In our study, we use Banister's model of elite athletic performance (1991) to evaluate a training plan. Banisters' model can be stated as Equation 9.

$$p_t = p_0 + (k_1 \sum_{i=0}^{t-1} w_i e^{-(t-i)/r_1}) - (k_2 \sum_{i=0}^{t-1} w_i e^{-(t-i)/r_2}) \quad (9)$$

where p_t is the cyclist's performance at day t of a training plan, p_0 is the basic level performance of the cyclist, w_i is the training load at day i of the training plan, k_1 is the coefficient of fitness gain, k_2 is the coefficient of fatigue gain, r_1 is the decay rate of fitness, r_2 is the decay rate of fatigue, and t is the total days of the training plan.

The cyclist's performance at day t (p_t) is calculated by including all of the following aspects: the cyclist's basic level of performance (p_0) which is a positive term, or physical fitness gain from training for t days $\left(k_1 \sum_{i=0}^{t-1} w_i e^{-(t-i)/r_1}\right)$ which is a negative term, or fatigue gained by training for t days $\left(k_2 \sum_{i=0}^{t-1} w_i e^{-(t-i)/r_2}\right)$. The coefficient of fitness is k_1 and k_2 is the coefficient of fatigue. The decay rates of fitness and fatigue are r_1 and r_2. In this study, all model parameters are defined by the results of model fitting from Busso et al. (1997).

To sum up, a cyclist physically gains 2 antagonist products after training, simultaneously; fitness and fatigue. Since $k_1 < k_2$ and $r_1 < r_2$, the cyclist gains and loses their fitness and fatigue in different ratios. The explanation of this is that cyclists gain fatigue more than fitness after training at a certain amount of training load. When cyclists have a rest, fatigue decays faster than fitness. The simulation that explains this training-performance interaction is illustrated in Figure 6.

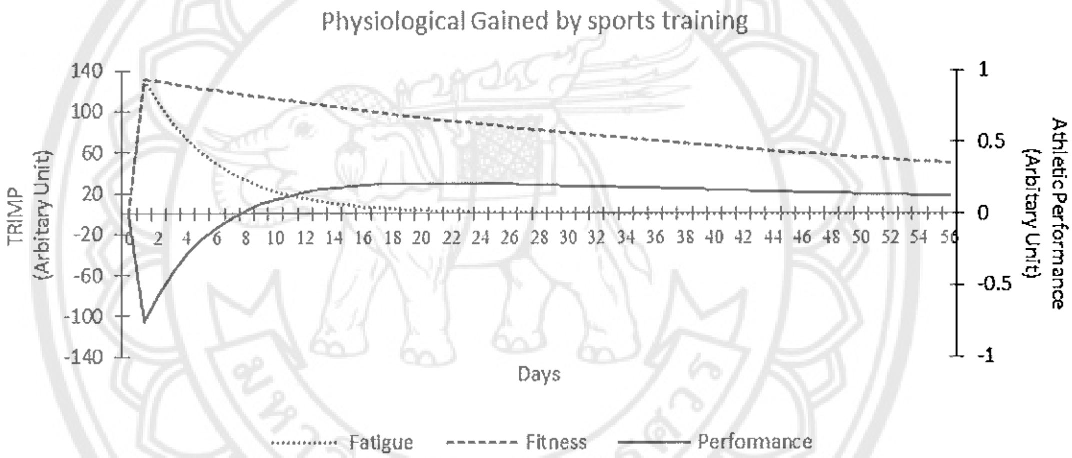

Figure 6 Physical response through time by a training session at 131.78 training load.

Figure 6 illustrates the data attained from a 30-year-old cyclist whose heart rate was 175 bpm, having trained only once for 90 minutes at a certain level of effort. This heart rate averaged around 90-95% of FTHR. This training session had a training load of 137.78. After the cyclist finished this training session, the gain in both fitness and fatigue was relatively equal. The cyclist's athletic performance dropped to a negative value by a slightly higher magnitude than the level of fatigue. On the 8th day, the performance rose to 0 because fatigue had decayed faster than fitness. After day 8, athletic performance began to rise to be slightly higher than the basic level, but as the cyclist did not continue his training, no additional training outcomes were achieved, and his fitness and fatigue decayed to his basic level, and therefore the athletic performance also gradually dropped to the basic level.

This rise in performance is called super compensation. Cyclists can exploit super compensation to raise their athletic performance by optimal scheduling of their training plan.

The personalized training intensity zoning by heart rate

In our research, we used data that had been collected, by others, from cyclists in the field, in actual events, to create a sample personalized sports training plan. In sports cycling, field tests are frequently used to define the level of a cyclist's fitness, using the parameters of Maximum Heart Rate, Resting Heart Rate, and FTHR.

1. Maximum Heart Rate normally collected by letting the cyclist start training as usual .Normally start with proper warm up and gradually increase the exercise intensity until cyclist cannot progress to more intensities .The maximum heart rate of this session is considered as his maximum heart rate.

2. Resting heart rate is collected by the time cyclist wake up .The more reliable value should be collected when cyclist is fully recovered.

3. Function threshold heart rate (FTHR) is the heart rate that reflect the effort intensity which particular cyclist can sustain for long period of time .In our study, the FTHR for 1 hour is selected.

Since training at specific heart rate is not convenience to train, especially for cycling that cyclist also have to pay attention at traffic, not only at heart rate monitor screen. The range of heart rate can be more safety and efficiency option to train. Coggan (2003) had propose the training zone which is a proportion to FTHR. Coggan's training zone considered as proper approach to defined personalized training zone because its FTHR dependent. Coggan's training zone is illustrated in Table 3.

Table 3 Coggan's training zones which depended on personal FTHR

Intensity	$\overline{hr_i}$	Description
Level 1	<68%	Active recuperation
Level 2	69-83%	Endurance
Level 3	84-94%	Tempo
Level 4	95-105%	Lactate threshold
Level 5	>106%	Maximal aerobic power

In our research we applied Coggan's training zones as the personalized training zones that reflect the intensity of a training session in the training plan.

Physiological constraints

To minimize the risk of overtraining and injuries, the sports training plan needs to include related physiological constraints. Three physiological constraints of the cycling training domain are identified as training monotony (Foster, 1998), CTL ramp rate (Coggan, 2008) and daily TRIMP.

Monotony

Training monotony (Foster, 1998) is a factor arising from training with a monotonous pattern which may have the consequence of the cyclist overtraining. Training monotony can be estimated by Equation 10.

$$Training\_Monotony = \frac{\sum_{i=1}^{n} \frac{\overline{TRIMP_i}}{std(TRIMP_i)}}{n} \tag{10}$$

In our simulations, training monotony was analyzed throughout an 8-week training plan. The training monotony of the i-th week is defined as the mean of *TRIMP* (Equation 8) of the i-th week, divided by the standard deviation of *TRIMP* for the corresponding week, so the overall training monotony is the summation of the training monotony calculated for each of the 8 weeks (Equation 10). Therefore, the training plan with a very monotonic training pattern will consequently have a high training monotony value. Due to the many possible variations in different training plans, it is impossible to find the appropriate upper bound of the training monotony value, so we were unable to scale these values for ease of use. We limited the training monotony value in conformity with Foster's suggestion that a good training plan should restrict training monotony to a value less than 1.5 (Foster, 1998).

CTL ramp rate

Coggan (2008) identified *CTL_RampRate* as an essential constraint, acknowledging that cyclists may risk illness or symptoms of overtraining if they attempt to increase their Chronic Training Load (CTL) too rapidly. Coggan stated that increasing the CTL score at a rate of 5 to 7 weekly for 4 weeks is the recommended maximum. In our current study, the CTL ramp rate was used to restrict the progressive increase of training load so that cyclists would not overtrain. In equation 11, the *CTL_RampRate* (Coggan, 2008) is simply an average of the increment in CTL values from the i-th day back over the preceding 7 days. A good training plan should not present any sequence of 7 days where the CTL ramp rate is higher than 5 to 7.

$$CTL\_RampRate = \frac{\sum_{i=7}^{n}(CTL_i - CTL_{i-6})}{n-6} \tag{11}$$

The CTL of the i-th day is defined as the summation of the chronic training loads from the previous 7 training sessions together with the Training Stress Score (TSS) from the current training session. With CTL_i calculated from equation 11, TSSTSS is then estimated by Equation 12.

The TSS estimation method was adopted from (Anonymous, n.d.). This equation replaces the original power data with heart rate data (^{HR}TSS). This is a pragmatic decision based on the exorbitant cost of a power meter, which has previously been discussed as being out of the reach of most ordinary cyclists. The heart rate data from a heart rate monitor sufficed for our study. ^{HR}TSS is determined by Equation 12:

$$^{HR}TSS_i = \frac{t_i \cdot (\overline{hr_i} - {}^{resting}hr) \cdot \left(\frac{(\overline{hr_i} - {}^{resting}hr)}{(FTHR - {}^{resting}hr)} \right)}{(FTHR - {}^{resting}hr) \cdot 3600} \cdot 100 \tag{12}$$

where t is training the duration in seconds, ${}^{resting}HR$ is the cyclist's heart rate at resting state, and *FTHR* is the cyclist's functional threshold heart rate. The duration data is measured in seconds.

Limitation of daily training load

In consideration of the possibility that a GA-based sports training plan can limit its effectiveness if training sessions exceed appropriate physiological constraints, we added another physiological constraint for limiting the TRIMP values of each daily training load so that they did not exceed 450 to 600, which was derived from a UK cycling training plan (Britishcycling, n.d.)

Computational intelligence: Scheduling techniques

In overview, planning a sports training is about sequencing training sessions with different effort levels. The objective of the sequencing is to raise athletic performance as high as possible while maintaining all physiological constraints. Therefore, planning a sports training plan may be considered as a scheduling problem with important constraints. Therefore, computational intelligence techniques for scheduling are appropriate. These techniques are discussed in the following sections.

Deterministic techniques

Deterministic optimization is a global optimization technique with the objective of solving the numerical problem while providing the theoretical guarantee that the reported solution is the global one. The term 'deterministic' typically refers to the complete optimization techniques which converge to the global optima in a finite time. Deterministic techniques take advantage of the analytic properties of the problem to generate sequences of searching points. These sequences converge to the global optima. The disadvantage of these techniques is in the task of the problem analysis itself. The researcher needs to analyse the problem in-depth in order to generate the proper sequences of search points that will converge to the global optimal solution. Because the deterministic optimization techniques normally parse the sequence of search points in one direction with no backward stepping, which means that poorly analysed and constructed sequences will not result in the correct global solution. In addition, the problem size, which must be considered in the formulation of the sequence, will affect the performance of this approach: the time necessary for the optimization algorithm to process a long sequence and the possibly large number of search points, may be excessive and therefore impractical.

Heuristic search algorithm

Heuristic search algorithm is the technique that applies the heuristic on its moves with hope to find the global optimal solution. Heuristic employed in algorithm as search strategies to tackle the exponential nature of the most problems. This technique reduces the search space from the exponential number to polynomial number.

We applied the greedy strategy in our heuristic search algorithm as the appropriate algorithm for our optimization problem. Using this algorithm, we iterated through a set of choices of sub-problems in each step. As indicated in the definition cited above, greedy methods always select the best choice at the current step, subsequently leading to the globally optimal solution. The greedy strategy is a powerful method for wide range of problems, and is the basis of the minimum-spanning-tree algorithm, and Dijkstra's algorithm, for finding shortest paths form a single source.

The greedy strategy has been applied to many complex problems. For logistic scheduling problem, Chang et al. (2014) used the greedy strategy to solve emergency logistic scheduling problems, they demonstrated that it outperformed existing algorithms in 'time to delivery' by between 46.15% and 63.57%. In a renewable energy problem, Song et al. (2015, 2014) studied the performance of the lazy greedy algorithm for the optimization of wind turbines positioned above complex terrain. Their results showed that the lazy greedy algorithm combined with the virtual particle wake model produce a better solution to the problem, faster than previous bionic methods. Neil et al. (2014) proposed a greedy strategy to determine how to distribute tidal energy devices over the northwest European shelf seas. The last example of greedy strategy application in a sparse unmixing problem. Tang et al. (2014) proposed the regularized simultaneous forward-backward greedy algorithm (RSFoBa) for sparse unmixing of hyperspectral data. This algorithm resulted in a more stable result and was less likely to be trapped into the local optimum, with improve the accuracy and time efficiency.

In order to apply a greedy strategy to arrive at a complete solution to a problem, the overall problem needs to be analyzed to clarify the possible sub-problems that will be encountered before arriving at the complete solution. A sub-problem is considered as a substructure of the complete solution, which will be used as choices that may be made in each algorithm step. The choice is always made on the best sub-problem which is evaluated by the problem's objective function.

Once the sub-problems are defined, the greedy strategy is defined according to the approach of Cormen et al. (2009) applying the following steps:

1. Determine the optimal substructure of the problem.
2. Develop a recursive solution.
3. Show that if we make the greedy choice, then only one sub-problem remains.
4. Prove that it is always safe to make the greedy choice. (Steps 3 and 4 can occur in either order)
5. Develop a recursive algorithm that implements the greedy strategy.
6. Convert the recursive algorithm to an iterative algorithm.

By using the greedy strategies above, our heuristic search algorithm will check that its result is derived from locally optimal solution to globally optimal solution.

Stochastic techniques

Stochastic techniques randomly initiate candidate solutions as search points corresponding to the search space in the hope of locating the global optima. The generated search points are evaluated by the objective function. High quality search points will be continuously exploited for locating the global optima in the next iterations.

This technique has been found to be more flexible than deterministic techniques even though its reported solutions cannot be guaranteed to be the global optima. However, researchers have successfully adopted stochastic techniques to solve such complex problems and have reported it to be an efficient method in many ways.

Genetic algorithms

Genetic algorithms (GA) are classified as metaheuristic optimization algorithms for solving complex problems by iterative searching among possible solutions in the search space and to evaluate the quality of each solution found. When compared against deterministic techniques, solutions presented by a GA might not the optimal solution, but they usually require less computational time: this is the tradeoff. GA's are considered to be practical techniques for very complex problems. Gas are well-known techniques that have been applied to many diverse combinatorial problems. One complex problem to which GA's have been applied is the scheduling of examinations in a university. Exam schedules must consider the variety of subjects in which a student must sit an exam, the scheduling of rooms, avoiding time, room and subject clashes for all students. In other words, a complex set of inputs to the scheduling problem. Traditional techniques would usually demand very high computational power and time to solve this kind of problem. A better approach is to search for potential solutions instead of seeking the one best solution. Other problems of the same potential complexity to which GA's have been applied include job shop scheduling (Chen et al., 2012; Wang, 2012; Xu et al., 2014), metaheuristic applications in structures and infrastructures (Faghihi, Reinschmidt, & Kang, 2014; Hejazi et al., 2013; Rashedi, & Hegazy, 2015), image enhancement and segmentation (Hoseini, & Shayesteh, 2013; Kanan, & Nazeri, 2014; Xie, & Bovik, 2013), management applications (Arif, Javed, & Arshad, 2014; Biethahn, & Nissen, 2012; Vidal et al., 2013), classification (Devos, Downey, & Duponchel, 2014; Martis et al., 2014; Welikala et al., 2015).

Studies of intelligent computing by means of scheduling for a sports training plan are limited in number. Fister et al. (2015) proposed what they termed the Bat Algorithm for planning sport training sessions, which they described as an evolutionary algorithm that was inspired by the behavior of a micro-bats particular its ability to orientate itself, and for locating prey, in the 3-dimensional space. Their study used real training log data as the initial population sample and searched for the optimal training plan with minimal errors, as a comparison to a training log selected by the coach involved in that sample training program. Huang et al. (2015) proposed an intelligent computing system for scheduling which uses fuzzy logic inference to generate diagnostics and prescriptions for a customized physical fitness schedule and healthcare. These diagnostic and prescription data were intelligent and scheduled by their genetic algorithm.

For other related problems in the sports domain, many studies adopted intelligent computing in many aspects in respect to the sports domain. Novatchkov and Baca (2013) proposed an implementation of artificial intelligence (AI) techniques such as machine learning algorithms based on neural networks (NNs) for the automated classification of sensor information gathered from weight training equipment equipped with weight and force sensors, using fuzzy logic techniques for the evaluation of exercises performed on those exercise machines. Meng et al.(2014) used GA to optimally assign referees to their preferred time in a volleyball tournament schedule. Their results show that the proposed algorithm is reliable. Atan, & Hüseyinoğlu (2017), addressed a scheduling problem relating to football players and referees simultaneously by applying a mixed integer linear program formulation to games with specific rules, in the Turkish league. Atan, & Hüseyinoğlu (2017) applied GA methods concerning referee-related workload constraints. Their results show good performance in terms of computation time and objective function values.

So many studies have been published that proposed an efficient method to create a good training plan. However, these required some effort in collecting the initial.

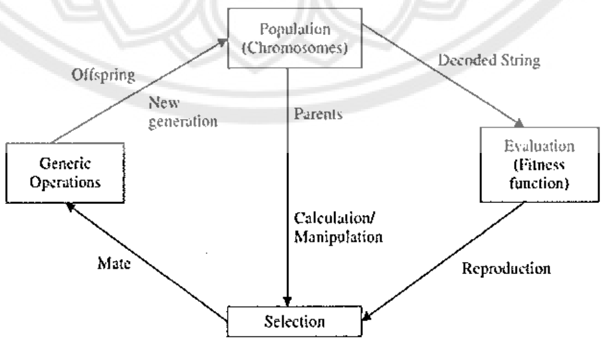

Figure 7 Cycle of processes in genetic algorithm

Source: Introduction to Genetic Algorithm (Sivanandam, & Deepa, 2007) data, an effort that might have affected the extent, and therefore diversity, of the collected results, which resulted in a struggle to find global maxima. In addition, methods for constraint handling were not present in these studies.

The overview of genetic algorithm is illustrated in Figure 7.

The algorithm begins with the formulation of candidate solutions for a string of binary, integer or continuous numbers, which are called chromosomes. The GA randomly initiates the number of chromosomes, which is called a population. All the chromosomes in the population will be assigned a fitness or objective value by evaluating with a fitness function. Better chromosomes will be selected to be a parent for producing offspring in the next generation. Reproduction methods are mate and mutate. Mating will potentially produce an offspring with a large difference to the parent chromosome (exploration) while mutation will produce an offspring with a smaller difference (exploitation). Each offspring will replace its parent and the GA process continues until the exit criteria is meet.

The pseudo code of GA is:

```
population = initPopulation(popSize)
while not exit criteria {
  evaluate (population)
  population = select(population)
  offspring = crossover (population, CXPB)
  offspring = mutation (population, MUTPB)
  population = offspring
}
return best(population)
```

In summary, a GA search produces an optimal solution by moving and evaluating candidate solutions in the search space. Mating is considered as an exploration process that emphasizes the discovering of better candidate solutions that lie in the unknown region. Mutation is considered as an exploitation process that seeks a better candidate solution that lies nearby the best-known solutions. Successfully balancing the tradeoff between these processes will increase the opportunity to locate the global optima.

Particle swarm optimization

The activity of creating a training plan has become simpler and more convenient with many sports organizations and companies having published apps for training planning and training plan creation British Cycling (Britishcycling, n.d.) and Trainingpeaks (n.d.) amongst others). These apps are often available on the Internet, but may have a significant cost. It is our sense, based on our simulations and mathematical modelling of one of the major training planning apps, that, while the training plans created according to these apps may include sophisticated training strategies, they do not achieve a substantial raise in athletic performance. Our simulation models are described and justified below.

Other researchers have developed computer systems that generate sports training plans simply, quickly and efficiently. Brzostowski et al. (2015) use Banister's model (1991) as the objective function of their dynamic programming algorithm. The outcome of this algorithm was an optimal training plan which indicated the same training load per training day from the beginning of the training plan until tapering off shortly before the end of the planned training program. While this training plan was effective in raising athletic performance, it did not consider physiological constraints. For example, by having no training load variation over most of the training duration, the plan did not consider the matter of training monotony which has been reported as a cause of overtraining which is detrimental to training effectiveness and athletic performance (Foster, 1998). Kumyaito et al. (2017; 2016a) used a genetic algorithm to generate the optimal training plan using Banister's model as the objective function. While the training outcome from the training plan was improved athletic performance, the system was somewhat impractical with no mechanisms to handle or manipulate necessary constraints. As well, the computations were complex, and the cost in terms of required computational time was high.

Considering the drawbacks identified in existing work, a need for an optimization technique to enhance the quality of the solution (that is, ensuring that physiological constraints are taken into account), and to reduce computational cost, was considered essential. Kennedy, & Eberhart (1995) discussed particle swarm theory, and developed a Particle Swarm Optimization (PSO) algorithm that was adopted as the main optimization technique: this reportedly gave outstanding performance. The PSO algorithm is a population-based heuristic search algorithm which tries to iteratively improve the candidate solution by a given measure of quality. Each candidate solution, or each particle in the swarm, in the PSO sense, is moving within the search space toward a better position by a movement formula that is influenced by the particle's known best position and the swarm's known best position. At the end of the processing, all particles would be located at the swarm's best position. These particles will be decoded to an optimal result by the algorithm. The PSO algorithm has been successfully applied to solving discrete combinatorial optimization problems (Jarboui, Ibrahim, Siarry, & Rebai, 2008; Pan, Tasgetiren, & Liang, 2008; Sha & Hsu, 2008). In a comparison between PSO and GA, in Hassan et al. (2005), several functions were tested. The results showed that PSO, being a less complex algorithm than GA, is faster than GA in complex situations while both of them exhibited the same quality of solution. Although PSO is fast and can find high quality solutions efficiently, in some cases the velocity of particles may become too high which may make particles jump out of the feasible region of search space. One solution to this problem is by controlling the maximum velocity of particles (Takahama, & Sakai, 2006), but it is very difficult to choose a proper maximum velocity in advance because the proper maximum velocity depends on the problem at hand. Thus, the velocity of particles should be adaptively controlled at runtime.

The Particle Swarm Optimization (PSO) algorithm is a population-based heuristic search algorithm using optimization techniques based on swarm behavior such as can be observed in the natural behavior of bird flocks and schools of fish. The PSO algorithm searches the space of the objective function by adjusting the velocity of individual agents, called particles, which are formed by positional vectors. The movement of particles in the swarm consists of two components: a stochastic component and a deterministic component. Each particle moves towards the position of the current global best p_{gd} and its own best location p_{id} in their history, while the particle moves at random velocity v_{id}. When a particle i finds a location that is better than any previously found location, this location is set as the new best location of that particle.

Most implementations evaluate a particle i in the neighborhood consisting of itself and its immediate neighbors, particle i-1 and particle i+1, within an N-length array. The variable g is assigned the index value of the particle with the best performance so far in the neighborhood. That particle's velocity is changed by the following formula:

$$v_{id} = v_{id} + \varphi(p_{id} - x_{id}) + \varphi(p_{gd} - x_{id}) \tag{13}$$

where φ is a random positive number generated for each individual (*id*), whose upper limit is a parameter of the system, and the particle's position is changed by the following formula:

$$x_{id} = x_{id} + v_{id} \tag{14}$$

The particle swarm algorithm is robust in solving problems featuring nonlinearity and nondifferentiability, multiple optima, and high dimensionality through adaptation which is derived from social-psychological theory (Eberhart, & Kennedy, 1995). The original pseudocode is:

```
Initialize the population randomly
While (population size){
    Loop
        Calculate fitness
        If fitness value is better from the best fitness value (pbest)in history
            then Update pbest with the new pbest
    End loop
    Select the particle with the best fitness value from all particles as gbest
    While maximum iterations or minimum error criteria is not attained {
        For each particle
            Calculate particle velocity by equation (1)
            Update particle position according to equation (2)
        Next
    }
}
```

Constraints handling

In the real world, many optimization problems do not rely on an objective value alone. The practical solution should emphasize the problem's constraints. For example, in project management problem, the solution with the fastest finish time might not be a feasible solution because it demands a high workload which may overwhelm the workers and staff on the project with undesirable quality outcomes. In sports training, an efficient sports training plan should consider any physiological constraints that allow the cyclist to avoid overtraining.

For ease of understanding, let candidate solutions a, b, c, d, e, f, and x lie in the problem's search space, as illustrated in Figure 8.

The gray area is the feasible region which is a subset of the entire problem's search space. The white space is the infeasible region which includes any solutions that violate the problem's constraints. Let f be the solution that has the best objective value among a set of other candidate solutions. In constrained optimization, the candidate solution f cannot be considered as the optimal solution because it lies in the infeasible region which mean that it violates the problem's constraints. For this reason, the optimization techniques must handle this issue and select the optimal solution that lies in the feasible region, c or x in this case.

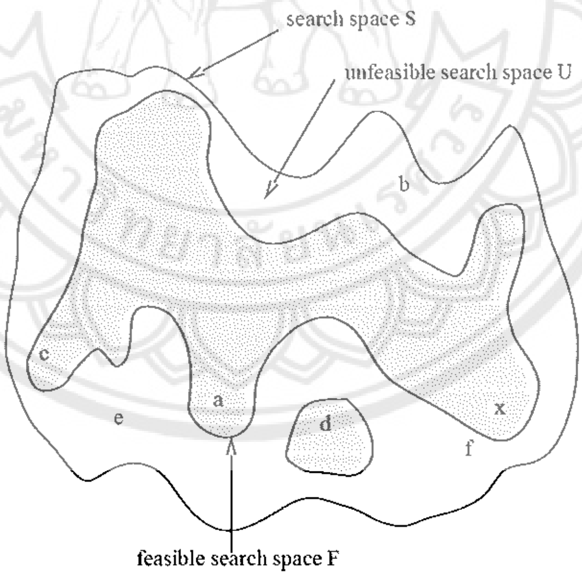

Figure 8 Example of search space in constrained optimization problem

The optimization techniques mentioned above must be customized to handle the problem constraints. According to Takahama, & Sakai (2010), methods for handling constraints in optimization techniques can be classified by the way the constraints are treated:

1. Death penalty method.

This approach uses constraints to check whether a search point is feasible or not.The methods in this category usually do not initiate the search points belonging to the feasible region, and demand high computational time when the feasible region is very small.

2. The constraint violation

The sum constraint violation is the summation of violations from all constraints functions and the objective function. The main difficulty is the selection of appropriate values for the penalty coefficient that adjust the strength of the penalty.To solve this difficulty, some methods adaptively control the penalty coefficient.

3. The constraints violation and the objective function"

Are used separately, and in this category, the optimization is based on the lexicographic order in which the constraint violation precedes the objective function. These methods have been successfully applied in various problems.

4. Optimizing the constraints and the objective function together.

Methods in this category take both the constraints function and the objective function as objective functions which are optimized by the multiobjective optimization methods.But in many cases, solving the multiobjective optimization instead of the single optimization is more difficult and is an expensive task.

5. Hybridization methods.

Methods in this category combine some of the above methods to solve the optimization problem.

It has been shown that methods in category 3 have better performance than methods in the other categories. Especially, the ε-constrained method can convert an unconstrained optimization algorithm into a constrained optimization algorithm. The modification is achieved simply by replacing the ordinal comparison with the ε-level comparison in direct search methods.

ε-constraint methods

The main idea of the ε-level comparison is the lexicographic ordering in which constraint violations precede the objective value. The overview of the ε-level comparison can be expressed by following equation.

$$(f_1,\phi_1) <_\varepsilon (f_2,\phi_2) \Leftrightarrow \begin{cases} f_1 < f_2, if \phi_1,\phi_2 \le \varepsilon \\ f_1 < f_2, if \phi_1 = \phi_2 \\ \phi_1 < \phi_2, otherwise \end{cases} \tag{15}$$

$$(f_1,\phi_1) \le_\varepsilon (f_2,\phi_2) \Leftrightarrow \begin{cases} f_1 \le f_2, if \phi_1,\phi_2 \le \varepsilon \\ f_1 \le f_2, if \phi_1 = \phi_2 \\ \phi_1 \le \phi_2, otherwise \end{cases} \tag{16}$$

The strength of the affection of the constraint violation influences the lexicographic ordering which is controlled by the ε-level value. Where the ε-level value is close to zero, the order is virtually arranged by the sum of constraints violation (ϕ). As the ε-level approaches infinity, the order is arranged by the objective functions (f).

These comparison methods can be easily modified in existing selection methods of unconstrained optimization algorithms. In the early step of initializing the search points, the optimization algorithm does not usually generate the feasible search points. Adaptively adjusting the ε-value can promote the use of the summed value of the constraints' violations in the early steps of the optimization process. This strategy can enhance the optimizer by allowing it to exploit the search points that may be close to feasible region.

CHAPTER III

PROBLEM DESCRIPTION AND FORMULATION

In our study, we used several optimization techniques to create a sports training plan, and then evaluate the results achieved, to discover the pros and cons of particular optimizing algorithms, such as heuristic searching, all discussed above, in producing solutions to the sports training planning problem.

In all of these techniques, the dataset derived from the problem domain needs to be analyzed and formulated with regard to each particular optimization technique. This chapter describes the dataset characteristics and candidate solution formulations that were used by all the optimization techniques used in our study.

Simulated cyclist

A personalized training plan needs the cyclist's profile in order to adjust the daily training load. For our purpose, the profile of the simulated cyclist as illustrated in Table 4.

Table 4 Characteristics of the simulated cyclists

Cyclist No.	Gender	Resting HR	FTHR	Maximum HR
1	Male	45	174	192
2	Female	64	133	202
3	Male	64	132	200
4	Female	44	174	198
5	Male	40	147	196
6	Female	50	168	192
7	Male	73	169	190

Table 4 (cont.)

Cyclist No.	Gender	Resting HR	FTHR	Maximum HR
8	Female	77	135	188
9	Male	55	139	186
10	Female	48	147	184
11	Male	45	154	182
12	Female	60	149	180
13	Male	75	156	178
14	Female	67	165	176
15	Male	81	156	174
16	Female	45	144	172
17	Male	77	127	170
18	Female	51	151	168
19	Male	55	146	166
20	Female	84	113	162

The gender of the cyclists has an effect on the training load applied in each training session. Banister's TRIMP model (see Equation 12), which we used to quantified training load, has a gender-dependent constant (1.92 for males and 1.67 for females (Banister, 1991)).

Maximum heart rate and resting heart rate data used to determine ${}^{norm}HR()$, the normalization of $\overline{hr_i}$ throughout a particular training session (see Equation 8). ${}^{norm}HR()$ is use to quantify the daily training load in the TRIMP model (see Equation 12).

The Function Threshold Heart Rate (FTHR) is used to classify the personal exercise program's intensities zone (see "The personalized level of training intensity by heart rate" section in Chapter II).

Training intensity zone and training session duration

For effective training, cyclists need to achieve a certain training load that depends on the training intensity and training volume appropriate to that cyclist. The training intensity was classified according to the personalized training intensities stated in Coggan's training zones. These training zones can be used to formulate the candidate solution in the optimization search space. For the training volume, we observed the cyclist's behavior and determined the appropriate lower and upper bounds of the training duration for endurance-dominant sports.

The intensity data and duration data in this problem included heart rate (HR), which has measured as beats per minute, and duration (D), which has measured in minutes. We applied Coggan's training zones to the simulated cyclist's HR data to encode the cyclist's actual HR data into a value from 0 to 9, as shown in Table 5.

The training duration data encodes the observed data in the range from 0 to 9 (Table 6).

Table 5 Simulated cyclist's heart rate zone

HR Zone	HR (bpm)	HR (% of FTHR)
0	45 - 82	25.86 - 47.13
1	83 - 118	47.70 - 67.82
2	119 - 132	68.39 - 75.86
3	133 - 144	75.86 - 82.76
4	145 - 155	83.33 - 89.08
5	156 - 164	89.66 - 94.25
6	165 - 174	94.83 - 100.00
7	175 - 183	100.57 - 105.17
8	184 - 188	105.75 - 108.05
9	189 - 192	108.62 - 110.34

Table 6 Training duration zone

Duration Zone	Duration (mins)
0	30
1	60
2	90
3	120
4	150
5	180
6	210
7	240
8	270
9	300

Total periods of the training plan

The most significant training phase is the preparation phase which builds a solid base of fitness for endurance-dominant sports. This phase promotes the efficient use of fat as energy, improves the cardiovascular system, and strengthens bones, tendons, and ligaments in preparation for the completion phase. Our study focused on the base training phase for 8 weeks which was considered long enough to substantially raise the athletic performance of the cyclist (Seiler, & Tønnessen, 2009).

Candidate solution formulation

Prior to estimating an optimal solution, the sports training plans consisted of 56 training sessions which were encoded into a string. Since a training session consists of two data items, each pair of consecutive characters were encoded from these two data items, the target training HR, and the training duration. Thus, the complete structure of the candidate solution for M training sessions, can be expressed as

$$HR_1\, D_1\, HR_2\, D_2\, HR_3\, D_3\, \cdots HR_M\, D_M$$

where HR_1 is the heart rate zone and D_1 is the duration zone corresponding to the first training session. As shown in

Table 5 and Table 6, both characters corresponding to the HR and D share the same upper bound and lower bound values of 0 and 9, respectively. By this encoding method, the problem domain data was successfully encoded into a chromosome. The target exercise values of each training session were normalized and transformed into a pair of genes that was suitable for processing in various local search algorithms.

Athletic performance evaluation

In a cyclist's performance assessment, the training load of each training session needed to be quantified. Then, all training loads were used as input into the athletic performance assessment.

We used Banister's TRIMP model (Banister, 1991) for quantify the training loads of daily training sessions. The TRIMP-based training loads from the daily training sessions were used as input for Banister's training-performance interaction model. The details of this corresponding models were described in Chapter II.

Physiological constraints

To minimize the risk of overtraining and injuries, the sports training plan needs to include the related physiological constraints. Three physiological constraints of the cycling training domain were training monotony (Foster, 1998), CTL ramp rate (A. Coggan, 2008) and daily TRIMP. The details of the corresponding physiological constraints were described in Chapter II.

Problem's search space

From the structure of the candidate solution, we can analyze the search space of this problem. Let a sample 56-days training plan that has a complete structure be as follows

$$TrainingPlan_i = HR_1\, D_1\, HR_2\, D_2\, HR_3\, D_3\, \cdots HR_{56} D_{56}\,,$$

where HR is the heart rate zone and D is the duration zone for each particular training session. Applying data from

Table 5 and Table 6, the number of all possible data for HR and D is 10 because they have encoded their continuous value to the discrete zones numbered from 0 to 9. The number of all possible combinations in a 56-day training plan is therefore 9^{56}. It is essential to use efficient searching methods to seek the global optimal solution in this very large search space.

The search space can be visualized as the overview of how the variations in a 56-day training plan interacts with the consequential athletic performance. Five-thousand training plans were randomly initiated and the final athletic performance of each was evaluated. We used principal component analysis (PCA) (Jolliffe, 1986) to project all 56 training days into 2 components as predictor variables. The response variable is the athletic performance value that results from 2 predictor variables.

Figure 9 illustrates 2 predictor variables on the x- and y-axes and a 3D surface that represents the response value on the z-axis. The peaks and valleys correspond with combinations of x and y that produce local minima or maxima.

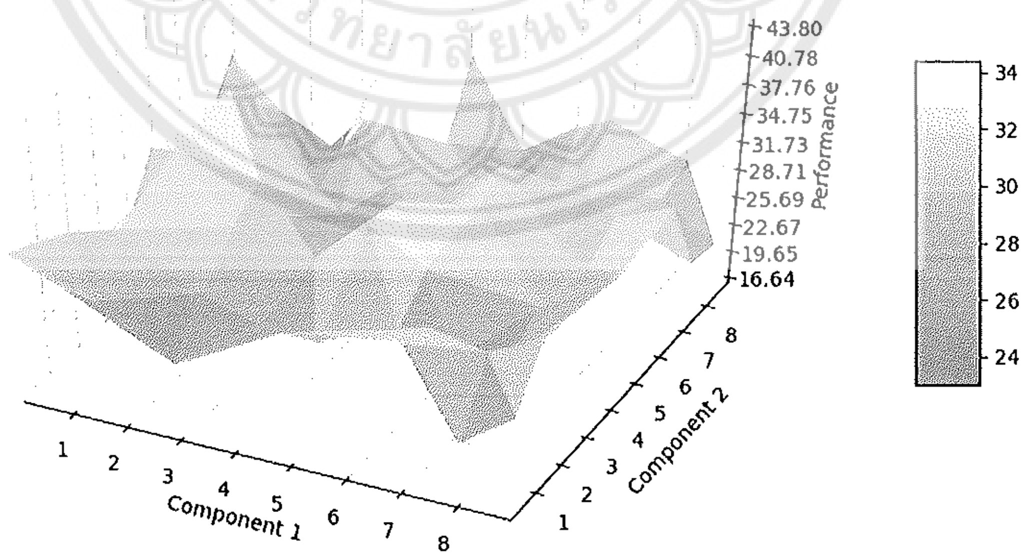

Figure 9 Tri surface plot of 100 samples of candidate solution

Figure 9 shows that the search space includes many peaks and valleys. Therefore, traditional techniques, such as simulated annealing and gradient descents, may not suit this problem. Therefore, examining different approaches for searching for the global optimal is needed to discover the best approach for the problem of scheduling a sports training plan.

CHAPTER IV

HEURISTIC SEARCH ALGORITHM

Overview

Our heuristic search algorithm used the greedy strategy which is the technique that has been applied in the past for solving scheduling problems. The advantages of this algorithm are the ease of implementation and overall performance. The type of problem to which the greedy strategy is best applied is one in which the solution's structure can be broken down into sub-structures. The greedy strategy always makes a choice based on the local optimal sub-solution and then iterates in the hope of finding the globally optimal complete solution.

In this chapter, the problem of scheduling a sports training plan is analyzed. As explained in Chapter III, the overall situation is analyzed, and then the sub-problem structure is clarified. Greedy methods make the best choice by using the objective value which is evaluated by Banister's training-performance interaction model. The constraint handling method is also involved in the process of analyzing choices. The result that we achieved showed that our heuristic search algorithm is capable of creating a sports training plan that can successfully raise the cyclist's athletic performance to a high level.

Experiments

Prior to scheduling a sport training plan with the heuristic search algorithm, the structures of the candidate solutions must be extracted. The problem analysis processes are performed in order to design the substructures of the problem candidate solutions, a complete structure of the problem candidate solutions, and the related variables. After all components are defined, the heuristic search algorithm iterates through the sub-solutions to find and select the best sub-solution. The iterations continue until the complete solution is found. Once all components are defined, the heuristic search algorithm will process a set of pre-defined variable parameters. The results of each variable set is then evaluated to find the best solution found by the heuristic search algorithm.

Problem analysis

In applying the heuristic search algorithm to scheduling a sports training plan problem, the initial processes are the problem analysis and the determination of the sub-solution structures that will subsequently be merged into the complete solution structure. In order to do this, the objective function and the physiological constraint violation functions that were used for selecting the best sub-structure should be analyzed.

1. Objective function

This function was used to evaluate the quality of the sub-structures, or the complete structure, in the solution. Banister's training-performance interaction model was used to analyze the objective value, that is the athletic performance, of the structure.

In our study, a sport training plan consisted of a daily training session, which was, therefore, the smallest sub-problem. For the i^{th} day in a training plan, athletic performance, as calculated in Equation 12 may have the simple form:

$$gained\,athletic\,performance_i = gained\,fitness_i - gained\,fatigue_i$$

Since the magnitude of the fatigue gained is greater than the magnitude of the fitness gained, the performance of the i^{th} day only will always be a negative value. But fatigue decays faster than fitness, so the athletic performance will begin to rise when the remaining fitness is greater than now-decayed fatigue. For ease of understanding, the 1st simulated cyclist's athletic performance level over a 14-day, 100-TRIMP training sessions followed by rest, is illustrated in Figure 10.

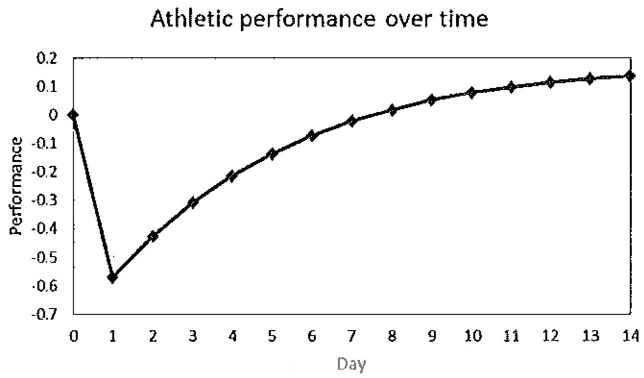

Figure 10 Athletic performance risen over time

By including the given parameters of training-performance interaction in the Objective Function section, the athletic performance will drop immediately after the training finishes. Following a 7 or 8 days rest, the remaining fatigue is less than the fitness gained, resulting in raised athletic performance.

This situation supports the notion that having different sized sub-solutions will affect the way that the heuristic search algorithm makes choices. Experimenting with different sub-solution sizes was therefore an appropriate research approach. In our study, the experiments on the sub-solution size were done by varying the sub-solution as 1 day, 4 days, and 7 days, which allowed the distribution of these tests across the 56-day training plan, divided into equal sized sub-solutions. Another reason that we had to limit the sub-solution sizes in this way was the problem of the exponential growth of the number of sub-solution combinations, which could become too large to process in a timely manner. In the heuristic search algorithm, all possible daily sessions were generated for all combinations of training session values of HR and Duration, as demonstrated in

Table 5 and Table 6. Since both HR and Duration range between 0 and 9, the number of all possible daily sessions is $10^2 = 100$. According to the daily TRIMP Constraint (described in the physiological constraints function), we could eliminate all daily sessions that had TRIMP greater than 450, leaving the remaining 61 daily sessions in the plan, as shown in Table 7.

2. Physiological constraint violation function.

The physiological constraints included monotony, the CTL Ramp Rate, and the Daily TRIMP are functions that were used to evaluate the feasibility of the sub-solution structure with regard to the domain constraints. Since the objective of the heuristic search algorithm is to select the local optimal and feasible sub-solution that will lead to the globally optimal and feasible complete solution, the sum constraints violation is evaluated against all sub-solutions to determine how feasible each sub-solution is. Hence, the sub-solution size is considered to be of importance for evaluating each physiological constraint that is needed on a specific day, to estimate how this sub-solution violates the physiological constraints. Therefore, the smallest sub-solution needs to be ignored in the overall feasible solution.

In this experiment, to be able to evaluate the feasibility of the CTL ramp rate constraint and the monotony constraint, at least 7 training sessions were needed.

3. Constraint handling: ε-constraint methods

The efficient sports training plan should include training sessions that take the physiological constrains in to account. In the heuristic search algorithm, we first computed the constraint violation corresponding to each physiological constraint, using the functions defined in Chapter III. All choices made in each iteration of the heuristic search algorithm are made by the ε-constraint methods which compare both the objective value and sum of constraints violation value against the ε-level. The detail of ε-level comparison was described in Chapter II.

Algorithm's parameters and configurations

Since the most appropriate parameter values can support the heuristic search algorithm in searching for both the local optimal sub-solution and the globally optimal complete solution, this experiment should be done to find the most promising parameter values that will result in a feasible training plan that raises the cyclist's athletic performance as high as possible.

The first variable we considered was the size of the sub-problem that was constructed from possible daily sessions. Since the heuristic search algorithm solves the problem by iterated appending of the current best sub-solution, until the complete solution is reached, the sub-solution quality directly affects the complete solution.

When making a choice on the most current optimal sub-solution, all sub-solutions need to be evaluated in both the objective value and the sum of constraints violation.

In the evaluation of the sum of constraints violation value, all possible current sub-solutions have to be evaluated against all constraint functions. To test all constraints in this experiment, the CTL ramp rate and the monotony need sub-solutions for at least 7 training sessions. This limitation makes any sub-solutions that are smaller than 7 training sessions unable to be included in the feasibility evaluation. As well, the other physiological constraint, the daily TRIMP score, should be under 450 for any feasible training session.

In order to handle all constraints in this experiment, all feasible training sessions were limited by the TRIMP score of 450. This limitation reduced the number of feasible daily sessions from 81 to 61, and also reduced the number of combinations for each set of sub-solution sizes, as demonstrated in Table 7.

The exponential growth in the number of sub-problem combinations reflects on the algorithm's performance. Therefore, in this experiment, another method for reducing the number of sub-problem was needed. The first step was to generate all combination of training according to a specific sub-solution size, then sort all sub-problems by the TRIMP score in descending order. The second step was to divide the 1st step result into 3 equal partitions and then select the sub-problems whose position met the following criteria: for the maximum TRIMP session, the session at the boundary between the 1st and 2nd partitions, the session at the boundary between the 2nd and 3rd partitions, and the minimum TRIMP session. As illustrated in Figure 11, the choice of training sessions to combine in the specific length of the sub-solution was reduced to 4 or less choices. The choice selected was on the equally distributed partitioning. Finally, as shown in Table 7, the reduced number of available choices of training sessions made for faster computation times overall.

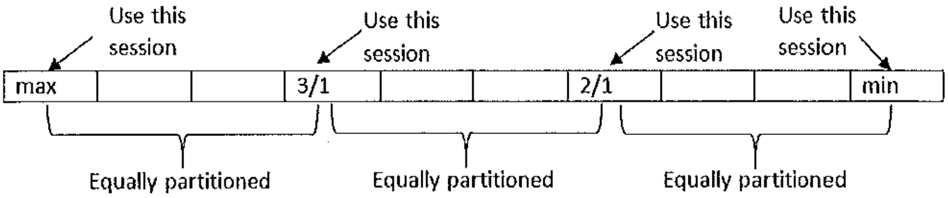

Figure 11 Selecting training sessions for heuristic search algorithm

Table 7 Reduced number of sub-solution combination

Sub-solution size	Number of sub-solution combination	Reduced number of sub-solution combination
1 day	$61^1 = 61$	$4^1 = 4$
4 days	$61^4 = 13{,}845{,}841$	$4^4 = 256$
7 days	$61^7 = 3{,}142{,}742{,}836{,}021$	$4^7 = 16{,}384$

Pseudocode

By using the structures of candidate solutions and described variables, the pseudocode of the heuristic search algorithm is:

1 Let *totalDays* be the total number of days in the training plan.

2 Let *n* be the size of the sub-problem.

3 Let *TP* be a list of the initial training plan (initial as empty list).

4 Let *dailySessions* be a list containing possible daily sessions restricted by the daily TRIMP constraint.

5 Let *subSolutionSessions* be a list of all possible combinations of training sessions in the form of the *n*-day sub-problem.

6 Let *subSolutionEvalData* be a 2-dimension list that contains{

A sub-problem's objective value in 1st dimension.

A sub-problem's sum of constraints violation in 2nd dimension.

}

7 generate subSolutionSession []

```
While TP's sessions < =totalDay{
        Empty subSolutionEvalData.
        for each member in subSolutionSession{
            Store the objective value of TP +current member of
                subSolutionSession in 1st dimension of the current
                member of subSolutionEvalData.
            Store the sum of constraint violation of TP +current member of
                subSolutionSessio in the 2nd dimension of the current
                member of subSolutionEvalData.
        }
        Sort the 1st dimension of subSolutionEvalData by descending order
            while sort 2nd dimension subSolutionEvalData by ascending
            order.
        TP =TP +first member of subSolutionSession.
}
return TP.
```

Results and discussion

The experimental results were analyzed with the aim of finding the best variable values that resulted in the training plan that raises the athletic performance the most, while maintaining the training feasibility according to all the physiological constraints. We investigated how the difference of parameter values affected both athletic performance and all the physiological constraints violations. The results were that each cyclist's training plan successfully raised the athletic performance to a high level, as illustrated in Figure 16. Unfortunately, the heuristic search algorithm failed to discover a feasible training plan for each cyclist that we simulated.

Nonetheless, to demonstrate the results of the heuristic search algorithm, we selected the 1st cyclist's training plan as the example case. The rising trend of athletic performance which resulted is illustrated in Figure 12. The evaluation of the results for sub-problem size parameter are described in the following section.

Sub-problem size

Sub-problem size is a critical part of the heuristic search algorithm because it generally affects the overall performance. The heuristic search algorithm iterates through the sub-problems, choosing the current best sequence of training sessions and appending that as its sub-solution. This sub-solution will keep appending the best sequence until it meets the complete solution. However, if in the early iteration the current sub-solution is too short, the evaluation of the sum constraint violation cannot be done and must be skipped. The criteria for choosing a sub-solution must be based on an objective value. By this circumstance, we cannot guarantee that the current sub-solution will be satisfied by all the constraints. The sub-solution can be evaluated correctly only after its length is long enough to be evaluated. If the current sub-solution is not satisfied by the constraints, the heuristic search algorithm cannot eliminate some sequence or sequences, nor step back to a previous sub-solution, and must continue appending the sequence of training sessions that create the new sub-solution that has both an optimal objective value and optimal sum constraint violation value. This is the drawback of the heuristic search algorithm, that sometimes it cannot arrive at the global optimal solution.

In our study, the sub-problem size is the length of days, and we used 1 day, 4 days, and 7 days in our simulations. Since all physiological constraints equations are directly evaluated by the sequence of training sessions, the sequence length can affect the value of the constraint violation. Each constraint violation function must have a minimal length training sequence otherwise it cannot be evaluated. The 7-day sequence was the shortest sequence length of training session for evaluating the monotony constraint, and 28 days was the length for evaluating the CTL ramp rate constraint, while we could a 1-day sequence as the shortest length of sequence for the daily TRIMP.

In our study, the heuristic search algorithm failed to create a feasible training plan (see Figure 14 and Figure 15).

None of the training plans derived using the heuristic search algorithm were feasible training plan. Each training plan violated at least one physiological constraint. The reason for the failure of heuristic search algorithm was that the building-and-evaluating process of the heuristic search algorithm was incomplete, even for the candidate solution. The reason for this is that in the early iterations of the algorithm, the incomplete candidate solution cannot be evaluated by the monotony constraint and the CTL ramp rate constraint, and the heuristic search algorithm skips the physiological constraints evaluation and selects the training sessions with a very high training load. This is done to raise the athletic performance as high as possible. This results in most of the early training sessions of the plan containing a monotonous pattern while at the same time accumulating a very high CTL (Figure 13). Once the candidate solution is long enough to evaluate the physiological constraint, the summed constraint violation of the candidate solution has gone too far. The heuristic search algorithm has no method to trace back the current candidate solution to previous state, but can only select and accumulate training sessions that make the current candidate solution feasible. As shown in Figure 13, the training load begins to reduce the later days of the program as the algorithm tried to correct the summed constraint violation. Unfortunately, the early training sessions still retained the monotonous pattern and the very high CTL, resulting in the heuristic search algorithm failing to discover a feasible training plan.

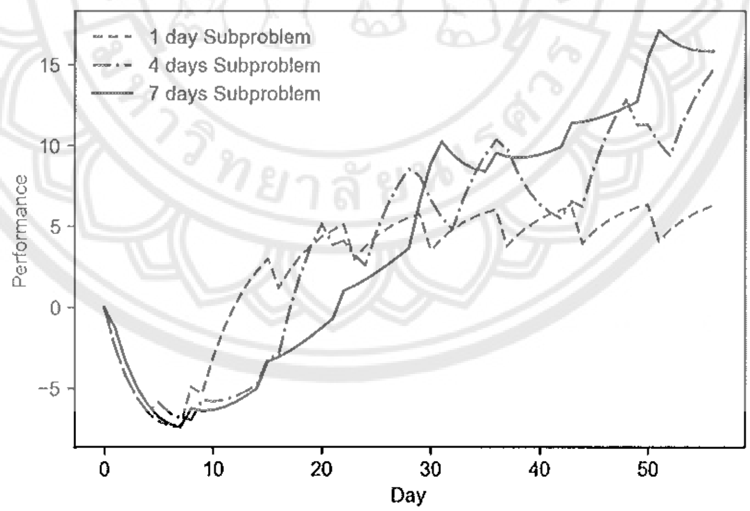

Figure 12 Rising trend of athletic performance by different sub-problem

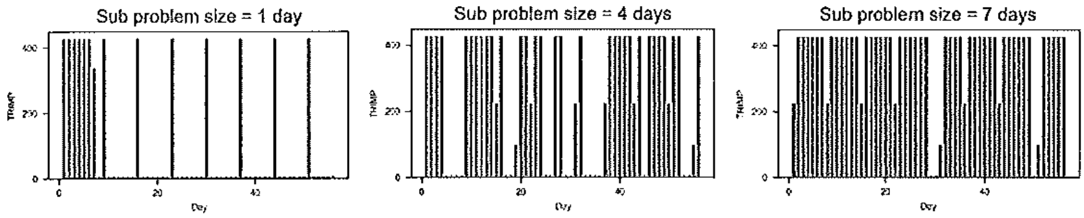

Figure 13 TRIMP by different sub-problem

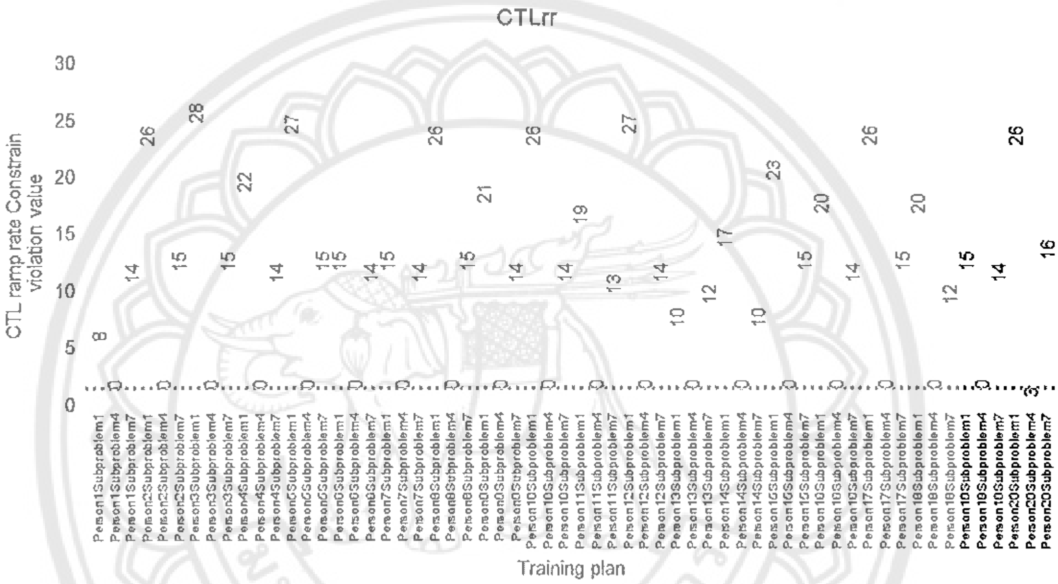

Figure 14 Chronic training load ramp rate for each cyclist's training plan by different sub-problem size

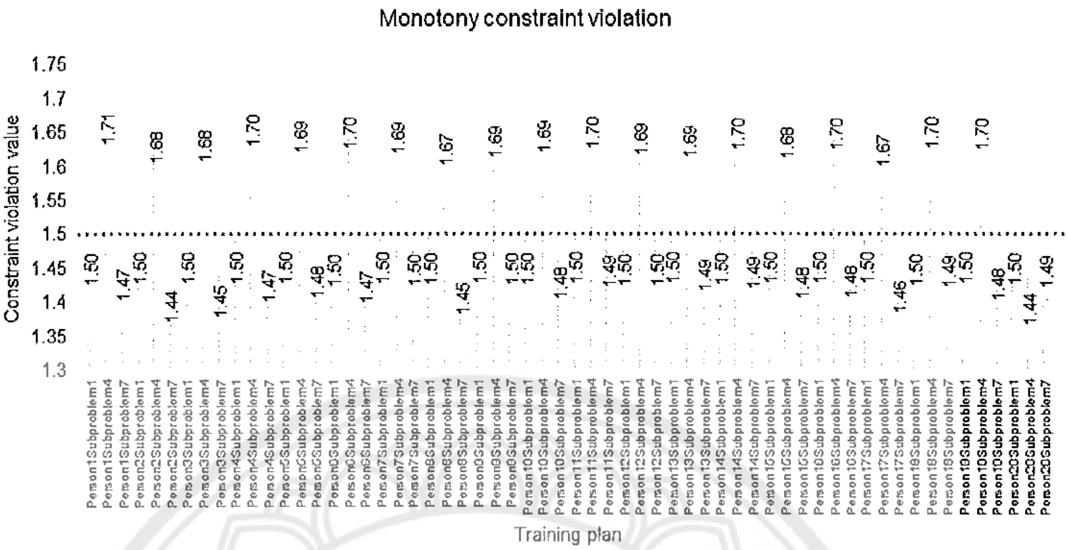

Figure 15 Monotony constraint for each cyclist's training plan by different sub-problem size

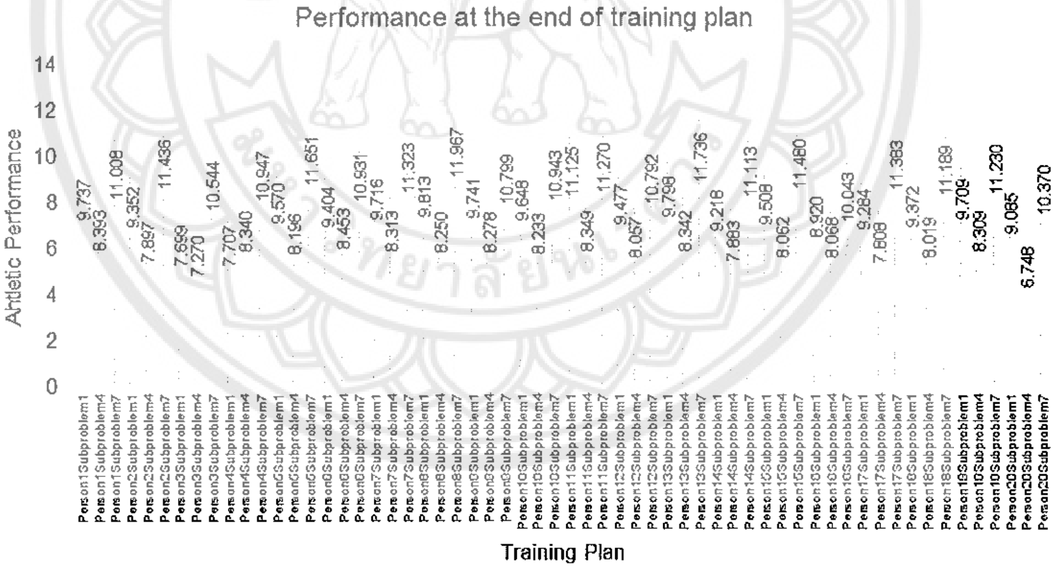

Figure 16 Athletic Performance for each cyclist's training plan by different sub-problem size

Conclusion

In this chapter, the sports training plan domain has been analyzed, and a new heuristic search algorithm that we developed for scheduling a sports training plan was proposed. The proposed algorithm adopted the ε-constraint methods to devise a training plan that recognized the physiological constraints of training monotony, CTL ramp rate, and daily TRIMP limitation. The best result of our proposed algorithm outperformed the standard training plan adopted (Britishcycling, n.d.). Unfortunately, the best result of our proposed algorithm fails to recognized the physiological constraints.

In comparison to (Kumyaito et al., 2017; Kumyaito, Yupapin, & Tamee, 2018), our proposed heuristic search algorithm consumed less computational resources and is suitable to be implemented on a specific platform that may have restricted computational power and energy, such as mobile devices; smartphones, Pads and otherwise in the Internet of Things environment.

CHAPTER V

GENETIC ALGORITHM

Overview

The Genetic Algorithm (GA) is a well-known method that could be applied to optimize the effectiveness of a training plan, but such a training plan would still be subject to certain constraints that can limit its effectiveness. Algorithms that do not acknowledge and include the described physiological constraints which deficient the algorithms. To overcome these shortcomings, and to ensure the development of an effective training plan, we modified the GA algorithm to include the ε-constraint method, thereby enhancing the effectiveness of the training plan. The results showed that an optimized sports training plan, with the ultimate outcome of attaining heightened athletic performance, can be achieved by applying the GA, modified by the ε-constraint method.

Experiments

Creating a GA-based sports training plan that recognizes and acknowledges the physiological constraints, while at the same time enhancing athletic performance, can be achieved by the following processes:

GA with ε-constraint method

The population of chromosomes are randomly initiated and processed through tournament selection, one-point crossover and uniform integer mutation. The algorithm is illustrated in the following pseudo code. The physiological constraints are applied to the selection method by using a ranking of candidate solutions that acknowledge the objective and constraints violation.

```
evaluation(population)
for i = 1 to 500{
    population  =selectTournamentEpsilon (population
                , offspringSize=len(population)
                , tournamentSize=3)
```

```
    offspring =crossover(population
              , cxOnePoint
              , cxProbability=0.6)
    offspring =mutation (population
              , mutUniformInt
              , mutProbability=0.009)
    Evaluate(offspring)
    population=offspring
}
```

In our simulations we found that there is little or no improvement in the objective value beyond 500 generations, so we limited the number of evolutions in the GA to that figure in our subsequent simulations. The tournament selection method was chosen since it has shown, in previous research, better performance in increasing the hit rate of feasible solutions for each generation (Takahama, & Sakai, 2010). Reproduction methods included the one-point crossover method and uniform integer mutation method. Due to concerns regarding computing time, we set the rate of crossover and mutation close to the lower bound of its typical range, 0.6 to 0.9 and *1/population-size* to *1/chromosome-length*. By experimenting, we found the appropriate crossover rate to be 0.6 and mutation rate 0.009. These settings achieved good solutions within acceptable computation times.

Constraints violations and ε-level comparison methods

Takahama, & Sakai (2010) modified the GA by including ε-level comparisons which are defined as an order related on the pair of values, the objective function value and a constraint violation value. If the constraint violation of a chromosome is greater than the ε-level, the chromosome is not feasible and its worthiness is low. The following is pseudo code for the constraint violation calculation.

#Calculate sum constraint violation of each chromosome in particular generation

```
for i to total number of chromosome in population {
    for j to total number of constraints {
```

```
            chromosome_i.constraint_jViolation=
            evalConstraint_jViolation(chromosome_i)
    }
}
for i to total number of chromosome in population {
    for j to total number of constraints {
        normalized chromosome_i.constraintjViolation into 0-1 scale by
            all chromosomes in current generation
    }
    for k to total number of constraints {
        chromosome_i.SumConstraintViolation=+
            (chromosome_iconstraint_jViolation)^2 * constraint_kWeight
    }
}
```

After the summation of the constraint violations for each chromosome has been achieved, the next step is selecting a chromosome winner of the tournament with ε-level comparisons, which is calculated by the following pseudo code.

```
#Selecting chromosome with ε-level comparison
for each chromosome_i in tournament {
    if (both chromosome_i.SumConstraintViolation and
        winner.SumConstraintViolation are <=ε) {
        winner = findMaxFitness (winter, chormosome_i)
    }else if (chromosome_i.SumConstraintViolation ==
        winner.SumConstraintViolation){
        winner = findMaxFitness (winner, chromosome_i)
    } else {
        winner = findMinSumConstraintViolation (winner,
            chromosome_i)
    }
}
```

Physiological constraints on the sports training plan

Three physiological constraints adopted in this study were training monotony (Foster, 1998), Chronic Training Load's ramp rate (CTL_RampRate) (Coggan, 2008) and daily training load limitation. Foster suggested a standard value for training monotony not greater than 1.5, and Coggan suggested a CTL_RampRate score of 5-7 for a period of less than 4 weeks. We adopted the daily TRIMP from the benchmarking training plan of the British Cycling (Britishcycling, n.d.), of less than 600.

Results and discussion

The algorithms mentioned above were developed as Python language scripts, using the Distributed Evolutionary Algorithms (DEAP) framework version 1.0.2 (Fortin et al., 2012), and the Python scripts were implemented and tested on WinPython 64-bit version 3.4.3.5. Charts were created using MS Excel® 2016. A population of 300 chromosomes were evolved for 500 generations by GA as pseudo-code in the "Constraints violations and ε-level comparison methods" section.

The experiment was done by varying the tournament size between 3 to 7, the crossover rate between 0.6 and 0.9, the mutation rate between 0.00279 and 0.00892. The best training plan was discovered by using tournament size equals 3, crossover rate equals 0.6 and mutation rate equals 0.00892. The result of the experiments for the first person in dataset is discussed under the following headings: Training Pattern, Athletic Performance, and Constraint Handling.

Training pattern

The TRIMP values from a number of training sessions, calculated by our GA-based approaches of constrained and unconstrained optimization, are illustrated in Figure 17. These were adopted from the UK training plan from (Britishcycling, n.d.) as out standard plan.

Our tests showed that GA, without the ε-constraint method, created the hardest training plan consisting of many strenuous training sessions. All of these training sessions exceeded the physiological constraints. For example, most training sessions in the GA-based training plans exceeded the TRIMP value of 600. A cyclist following this training plan risks injury and overtraining. In the training plan designed using GA with ε-constraint method, the TRIMP value of each training sessions remained within the standard constraint values stated above, which reflected the UK training plan that we used as our benchmark.

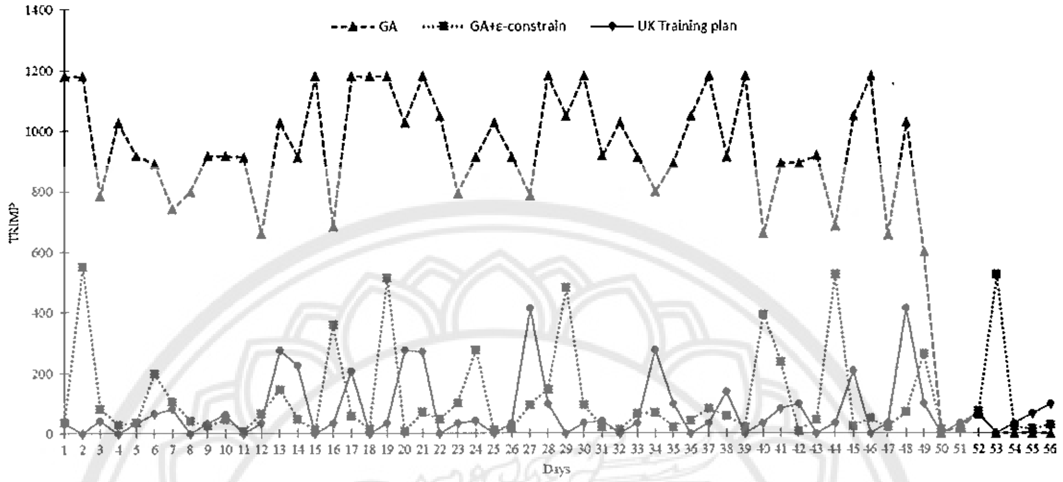

Figure 17 Comparison of TRIMP value between the UK training plan and GA-based training plans

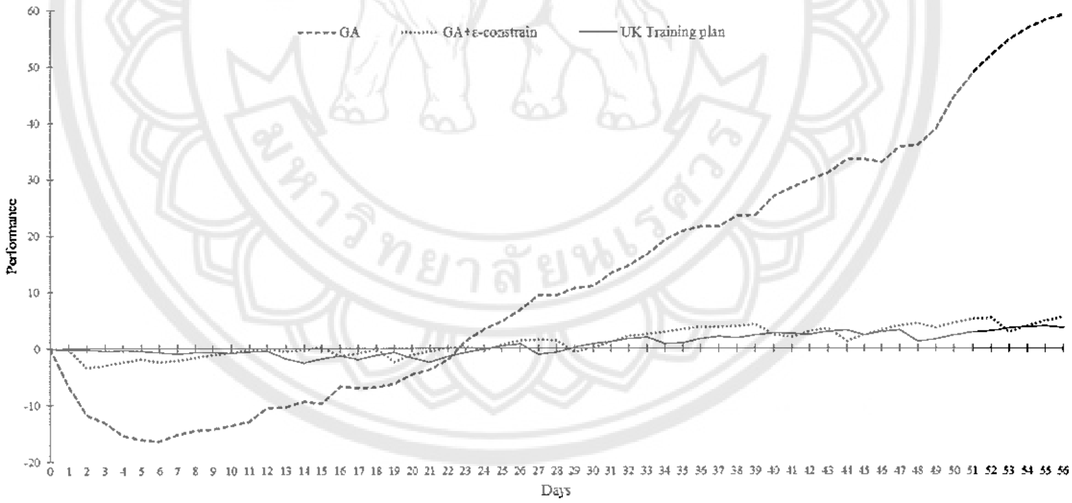

Figure 18 Comparison of athletic performance between the UK training plan and GA-based training plans

This means that they share the same training effort level, which reflects equality of their efficiency. In the plan derived from the GA without constraints, TRIMP was well above the acceptable levels.

Athletic performance

Figure 18 illustrates the estimated athletic performance achievable under the three plans; unmodified GA, GA with the ε-constraint method, and our standard plan. The highest athletic performance is a result of GA without the ε-constraint method. The training sessions in this plan are hard training sessions that exceed physiological constraints. As discussed elsewhere, the achievement of outstanding athletic performance is attended by the risks of injury or overtraining, therefore we suggest that this kind of training plan should be avoided. We consider that the GA with the ε-constraint method is the more effective training plan since its performance is more closely related to our standard.

Table 8 Benchmarking between GA-based training plans and the UK training plan

Dimension	MEAN		SSE		RMSE	
	GA	GA with ε-constraint	GA	GA with ε-constraint	GA	GA with ε-constraint
TRIMP	836.535	114.700	39431100.71	2182486.46	839.123	197.416
Performance	0.744	1.262	32714.402	130.661	24.170	1.527

Other comparative analyses of the solution are demonstrated in Table 8. A comparison of all the TRIMP results shows that the GA with the ε-constraint method is a better approach because it creates a better training plan in many aspects: first, the mean TRIMP of the entire training plan remains under the constraint, and second, the model provides a lower SSE and RMSE, which means it is more closely related to our standard. In addition, the comparative results of athletic performance also show that the GA with the ε-constraint method is a better approach because it has results in a higher mean performance, and lower SSE and RMSE.

Constraint handling

When comparing the outcomes of the three methods, we found that the GA with ε-constraint method produced a sports training plan that better acknowledged the physiological constraints and was more closely related to our standard plan. The plan that could result in higher athletic performance was produced by the GA without ε-constraint method which, however, created a sports training plan that violated the physiological constraints by including extremely intensive training sessions. This clearly implies that a cyclist undertaking a training program should avoid using a sports training plan that has been produced by the GA without ε-constraint method because of the risk inherent in that plan of becoming overtrained or injured.

By modifying the GA with the ε-constraint method, the GA was enhanced by this action. Existing unconstrained optimizer objective functions do not need to be extended for constrained optimization because the objective function and the constraint violation function are used separately. The lexicographic ordering action and the ε-level comparison process of the ε-constraint method were applied to the existing selection operation in the GA algorithm. This gave a more feasible candidate solution when applied to the comparison between the objective function and the sum of the constraints violation. In our experiment, the ε-constraint method was shown to enhance the effectiveness of the training plan when consideration of the physiological constraints was given.

Conclusion

Our results showed that the GA with ε-constraint method can create a suitable sports training plan that recognizes and acknowledges the physiological constraints. By applying the ε-constraint method, an ε-level comparison was adopted to modify existing tournament selections. The constraint violation value was successfully calculated by adopting the physiological constraints of training monotony, CTL ramp rate and daily TRIMP. Despite providing lower athletic training performance, the sport training plan created by GA with ε-constraint method has better similarity to the standard plan which we adopted.

We can therefore say with confidence that our study demonstrated that the GA with ε-constraint method produces a sports training plan that is, overall, more effective and more suitable for athletic training regimes.

CHAPTER VI

PARTICLE SWARM OPTIMIZATION

Overview

Adaptive Particle Swarm Optimization using ε-constraint methods was used to formulate a sports training plan by simulating likely performance outcomes.

Experiments

Adaptive Particle Swarm Optimization (PSO) was modified by Takahama, & Sakai (2006) by including ε-constraint methods. We adopted this approach to generate an optimal cycling training plan. The result was a cycling training plan that enhanced athletic performance by taking into account the physiological constraints: training monotony, CTL ramp rate and TRIMP, as well as the daily training load that we derived from the British Cycling's training plan.

Adaptive particle swarm with ε-constrained optimization

In the sports training plan optimization problem, physiological constraints should be handled in the optimization processes. Of the constrained optimization techniques, the techniques that separately evaluate the objective value and constraints violation value have shown good performance on various problems (Takahama, & Sakai, 2005; Takahama, & Sakai, 2000, 2003, 2004). Therefore, in our study, PSO use the ε-constrained method (Takahama, & Sakai, 2006) which separately uses the particle's objective value and the particle's constraints violation value. The ε-level comparisons are formulated as an order related on a pair of the objective function and constraint violation.

The ε-level comparison is used on separated objective values and constraints violation values to determine which the better particle is. Adjusting the ε-level value close to infinity makes the comparison mainly on the objective value. In contrast, adjusting the ε-level value close to zero makes the comparison mainly on the constraints violation value. In addition, this method limits the particle's maximum velocity adaptively to decrease the possibility of flying over a feasible region. All methods have been implemented as following pseudo code.

Randomly initiate a particle's position, x_i
Evaluate $f(x_i)$ and the particle's $\phi(x_i)$
Update p-best's position, x_i^*
Update p-best's constraints violation, $\phi(x_i^*)$
Randomly initiate the particle's velocity vector.
Initialize maximum velocity .(V_{max})
For $(i = 0; i \le T; i++)$ {
    Update the particle's velocity, v_{x_i} , limiting to V_{max} .
    Move the particle's position regarding to its velocity vector.
    Evaluate the particle's objective value.
    Evaluate the particle's sum of constraints violation
    Initiate the particle's best, x_i^* , and global best, x_G
    if ($f(x_i) <_\varepsilon f(x_i^*)$){
        if ($f(x_i) <_\varepsilon f(x_G)$){ $x_G = x_i$ }
        $x_i^* = x_i$
    }
    Update V_{max} corresponding to current iteration.
}
Return the global best particle

Results and discussion

For this study, the source code of Pyswarm (Abraham, 2015) was modified and represented as the pseudo code shown above, and the parameters for the ε-constraint method were defined by the constraint violation being given by the square sum of all constraints ($p = 2$). The ε-level is assigned the value 0 which means that the problems are solved in lexicographic order where the constraint violations precede the objective function. The number of groups N_g= 4, the number of particles in a group n_g = 25, the weight of the number of the currently feasible particle is F_λ = 0.2, the threshold of updating $F_\theta = 0.05$. The parameters for PSO are defined as the number of particles N = 100 (= 4×5), w^0 = 0.9, w^T = 0.4, the initial velocity is 0, and the maximum velocity V_{MAX_j} is adaptively controlled. The maximum number of iterations is 5,000 (50,000 fitness evaluations). Independent runs were performed 30 times. We selected and analyzed the run with the best athletic performance. The result of the experiments for the first cyclist in the dataset are discussed in terms of training patterns, athletic performance, and constraints handling.

Training patterns

The comparison of the PSO training plan against our standard training plan; the British Cycling training plan, is illustrated in Figure 19. The training load for each training session in the plan is represented as a bar chart. The solid bars located at left hand side belong to PSO's result while the striped bar at right hand side belong our standard's training load. As shown in Figure 19, both training plans share the same training pattern of alternating between high and low intensity training. Thus, the dynamic time wrapping (DTW) analysis was done as a similarity analysis. We bound the measured Euclidean distance between two similar agents in the two training plans at the same position to 1 and the two training plans that furthest apart to 0, PSO distance from standard training plan at 0.804. When it is close to 1, this indicates that the training plan produced by PSO is very similar to our standard training plan in terms of training patterns.

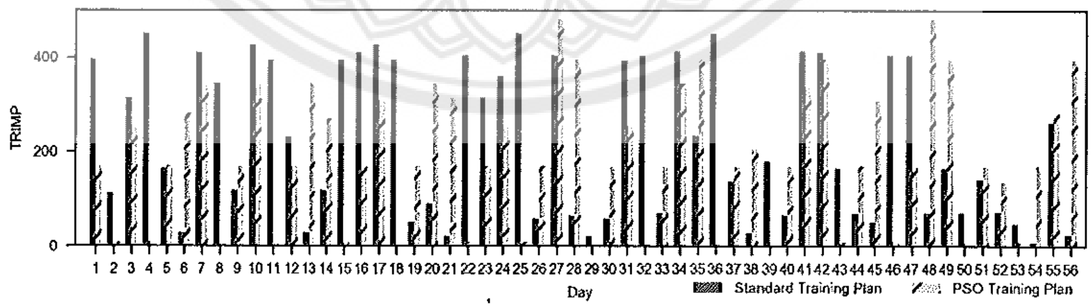

Figure 19 Comparison of PSO training plan against British cycling's training plan

Athletic performance

The athletic performance trend achieved over the training period, calculated each day, in each of the two training plans, is shown in Figure 20. Even though our results are closely related to the simulated standard training plan outcomes, the generated training plan outperformed the standard by raising athletic performance to a higher level of achievement. The PSO training plan's performance is 13.436 while our standard training plan's performance is 7.38 at Day 56, the last day. The PSO training plan satisfied all physiological constraints and achieved high athletic performance.

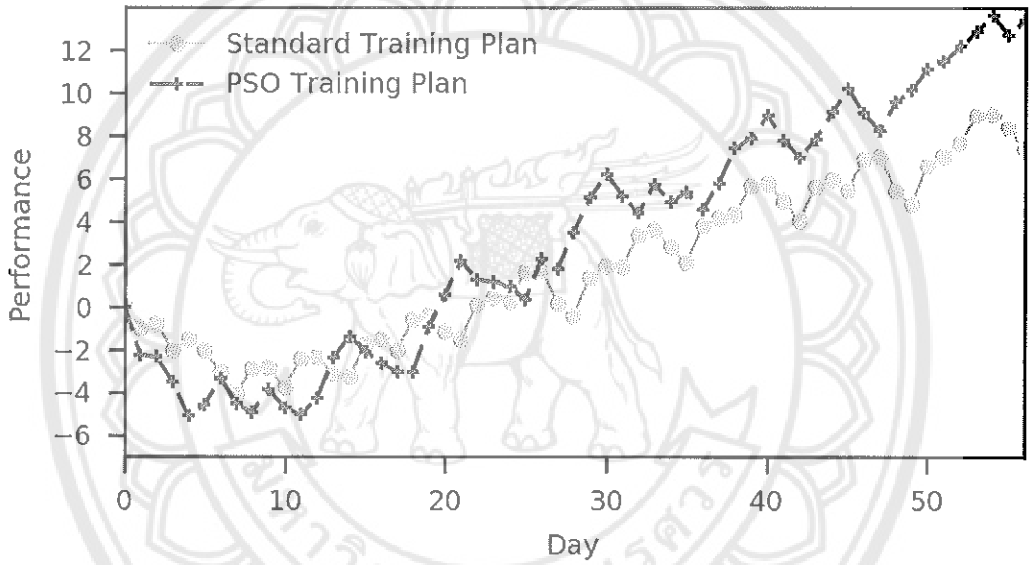

Figure 20 Performance corresponding to particular training plans constraints handling

The performance of the constraints handling mechanism in our study is illustrated in Figure 21 which represents the constraint violation values of each constraint calculated iteratively. In addition, the performance of the constrained optimization is analyzed in terms of the sum of the constraints violations by different iterations. In Table 9, we present the statistics of the sum of constraints violations, including the best, the worse, an average and a standard deviation.

The variation of a particle's velocity in the early iterations is very fast (Figure 21) with a brief scanning of their nearby area. The particle's velocity then slows down in each subsequent iteration as more detailed and fine searching occurs, seeking the best potential solution nearby the particle's current position. The purpose of the adaptive maximum limit of the particle's velocity is to avoid flying over better solutions. Table 9 illustrates the capability of this approach in each iteration. Particles are able to find feasible solutions and attract others to move toward their positions. The best particles satisfying all constraints of monotony, CTL ramp rate and daily TRIMP restriction. This means that the adaptive PSO generated training plan is considered as a practical sports training plan that minimizes the risk of becoming overtrained.

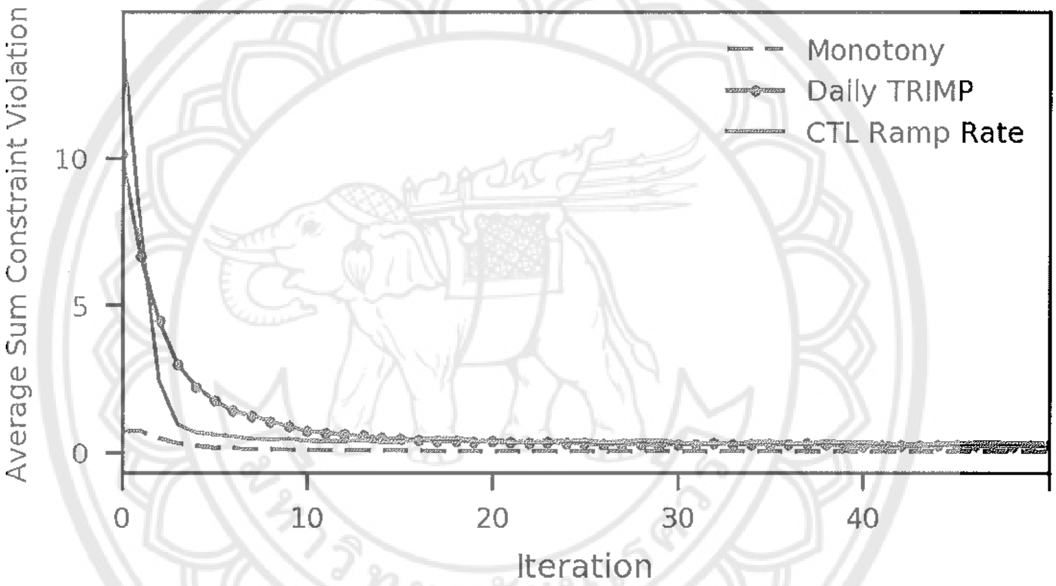

Figure 21 Convergence of summary constraints violation in early iterations

Table 9 All constraint violations in particular function evaluation times (FEs)

FEs	Monotony constraint				CTL ramp rate constraint				Daily Training Load constraint			
	Best	Worst	Average	S.D.	Best	Worst	Average	S.D.	Best	Worst	Average	S.D.
50	1.452	2.908	1.531	0.183	0	9	0.227	1.220	0	24	0.307	1.966
500	1.507	2.531	1.522	0.106	0	10	0.163	1.153	0	19	0.22	1.739
50,000	1.507	2.953	1.522	0.134	0	10	0.093	0.843	0	20	0.186	1.681

Conclusion

The adaptive PSO techniques for generating a sports training plan were presented. As the problem domain in our study is a cycling training plan, the cycling training-performance model and cycling physiological constraints were adopted. This work was divided into several processes including problem formulation, particle encoding, athletic performance model implementation as the objective function, physiological constraints adoption and implementation of an adaptive PSO with the ε-constraint method as the main optimization technique.

Our simulations demonstrated that the PSO-generated training plan significantly outperformed the standard plan which was based on a training plan from British Cycling, while satisfying all physiological constraints. These results mean that applying the Adaptive Particle Swarm Optimization method of deriving a training plan, and considering certain physiological constraints, produces a safe, high performance training plan.

CHAPTER VII

DISCUSSION

In this chapter the comparison of the results of the different proposed approaches is discussed. Several interesting findings are discussed; Personalization, Convergence, Computational Time, Quality of Solution, and Benchmarking against a commercial training plan.

Following this point in this thesis, for brevity, we use the simple term GA to mean the GA with ε-constraint methods.

Personalization

In order to illustrate the personalization of our proposed approach for creating cycling training plans, we simulated 20 cyclists with different physical fitness profiles (Table 4).

We created a personalized cycling training plan for each cyclist by using PSO algorithms with the best parameters value from our experiment. The personalized cycling training for each cyclists is illustrated in Figure 22.

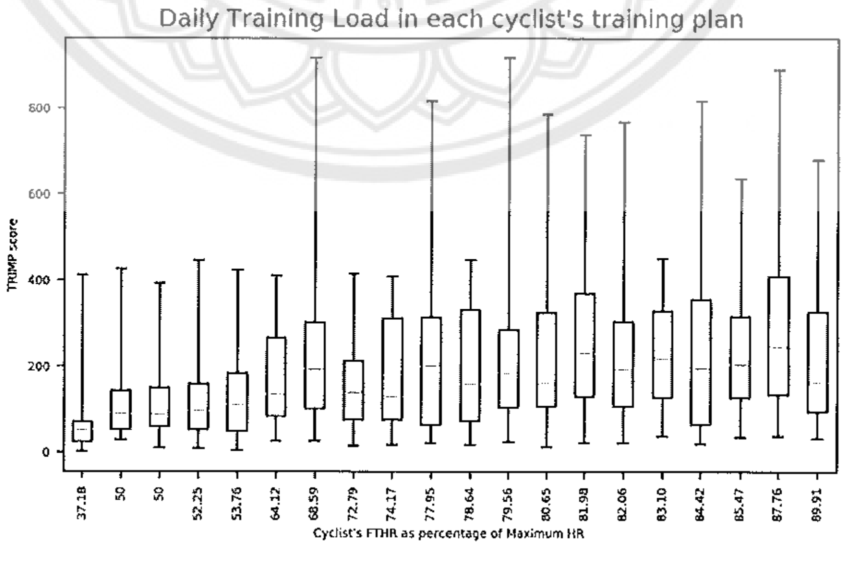

Figure 22 Daily training load in each cyclist training plan

Each cyclist was identified by their basic level of physical fitness as a proportion of their FTHR and maximum heart rate in X axis. The training load distribution of each cyclist's training plan was shown in Y axis.

The results showed that our algorithm successfully creates a training plan that is suitable for them, recognizing the cyclist's physical fitness level. The algorithm creates training plans that have a limited low training load for cyclists who have low physical fitness level while providing a higher training load for a cyclist who has a higher physical fitness level.

Convergence

In optimization, convergence is the phenomenon where, in both algorithms, the individual units (chromosome, particle) move from an initial point toward the global optima until they are all identical. Fast convergence to the result may be preferred but it does not guarantee that it will result in the optimal solution: fast moving individuals may step over the optimal solution. Therefore, the algorithm designer needs to emphasize the trading-off between fast convergence and the optimality of the solution.

For these reasons, convergence can be considered as an important evaluation matrix that represents the overall performance of the optimization algorithm. In this section, we compare the convergence between different optimization approaches. However, a simplistic comparison of each algorithmic approach, based only on the number of iterations performed in each algorithm, may lead to a failure to understand each algorithmic approach, because of the difference of complexity of each algorithm's iterative approach which significantly affects the convergence speed.

Therefore, in our simulations we used the number of Function Evaluation (FEs), which only count the number of times that individual evaluation functions are called. In our study we had an objective function and a sum constraint violation function which were called by the algorithm's selecting operations. FEs are considered as a fair performance comparison between our proposed techniques, GA and PSO, because for the same number of FEs, both techniques had almost equal estimated computation time (Engelbrecht, 2014).

For GA, the individual evaluation functions are called by the selection operation. For PSO, the individual evaluation functions are called by individual fitness evaluation function. Note that heuristic search algorithms use different strategies to find the optimal solution and cannot be compared to stochastic techniques like GA and PSO. the convergence comparison will be done between GA and PSO only.

First, we analyzed the total number of FEs for any iteration of GA which can be described by the equation:

$$FE_{GA} = N_{selection} \tag{17}$$

where the total number of function evaluations for GA's i-th iteration (FE_i) is equal to the number of times that the GA's selection operation ($N_{selection}$) is called. The number of selection operation executions is described by:

$$N_{selection} = n_{tournament} \times \left(\left(size_{tournament} - 1 \right) \times n_{evaluation} \right) \tag{18}$$

The total number of selection operation executions equals the number of tournaments ($n_{tournament}$) multiplied by the number of functions called for selecting the winner in each tournament ($size_{tournament}$ -1 times) multiplied by the number of functions called for each chromosome evaluation ($n_{evaluation}$). In this comparison, $n_{tournament}$ equals the population total divided by the tournament size, which is 100/3=33.33 (we use the configuration from the simulation that returned the best result). The $n_{evaluation}$ is 2, calling an objective function once and calling a sum constraints violation function once. Therefore, $N_{selection}$ equals

$$N_{selection} = 33.33 \times \left(\left(2 \right) \times 2 \right) = 133.32$$

Therefore, the total number of function evaluations for any GA's iteration is:

$$FE_{GA} = N_{selection} = 133.32$$

For PSO, the individual evaluation functions are called by the PSO's fitness function. The total number of function evaluations for any iteration in PSO can be calculated by:

$$FE_{PSO} = (N_{population} \times N_{evaluation}) \tag{19}$$

The total number of evaluations for any iteration in PSO equals the number of calls to the individual evaluation functions ($N_{evaluation}$) multiplied by the number of individuals in the population. $N_{evaluation}$ is equal to 2, by calling an objective function once and calling a sum constraints violation function once. In the configuration from the selected PSO simulation with the best result, there were 300 individuals in the population. Therefore, the total number of function evaluations for any iteration in PSO equals;

$$FE_{PSO} = (300 \times 2) = 600$$

The best performance result from each approach for the first cyclist in the dataset is shown in Figure 23. The athletic performance by PSO was 13.4511 while the athletic performance by GA was 6.8877. The athletic performance from PSO obviously outperforms the athletic performance from GA. In addition the convergence of the different approaches shown Figure 23, indicates that the GA technique converges to its optimal solution slightly faster than the PSO technique.

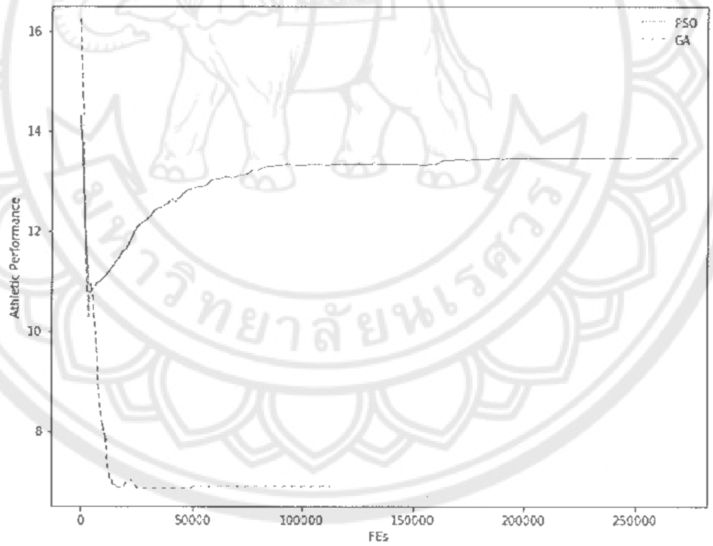

Figure 23 Performance profile based-on average of best objective value by FEs

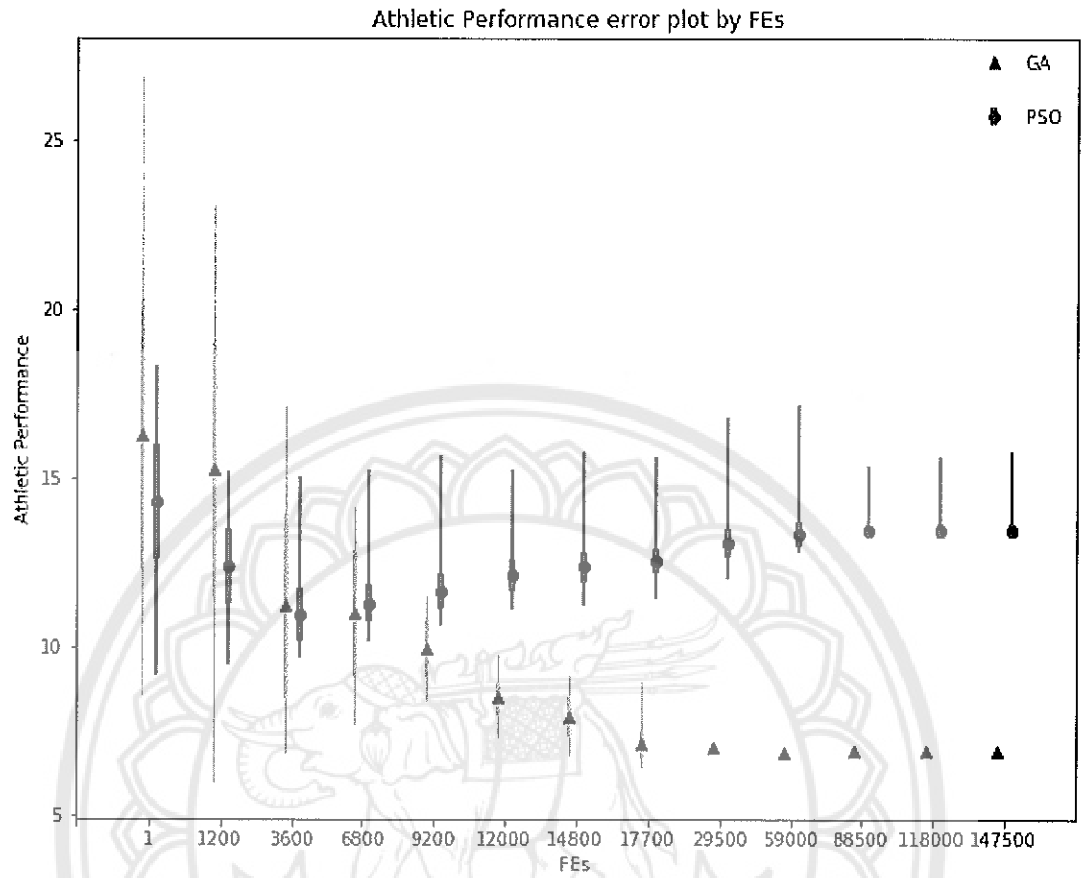

Figure 24 Comparison of population's error by FEs

The main reason to support the fast convergence of GA is the complexity decrement mechanism that limits the probability of triggers for GA's key operations, including the crossover operation, the mutation operation, and the selection operation. The frequency that these operations perform is limited by their probabilistic parameters. For a comparison of the results, see Figure 24.

The operational probability for the crossover operation is 0.6 and for the mutation operation is 0.01. In addition, the GA's tournament selection operation with a setting of about the tournament size, which is equal to 3 chromosomes, also decreases the complexity to 1/3 of the number of chromosomes in the population. For the PSO approaches, the convergence happens by the selection of the g-best particle and then updating the velocity vector for each particle. There is no complexity decrement mechanism in PSO. Therefore, in terms of convergence, the GA outperforms the PSO.

The distribution of objective values in the entire population, as illustrated in Figure 24, were also investigated. In the initial iterations, both techniques share the common situation of having their populations' objective values widely distanced from their mean value. Through time, each technique continually tries to achieve better candidate solutions by applying the reproduction, selection and evaluation processes to their population. These methods provide improved candidate solutions over time, per iteration, for each member of the population, by means of both the problem objectives and constraints. Thus, each member approaches the global optima. The iterative process terminates when the exit criteria are met, and the solution developed at that point is adopted as the best solution.

Computational time

The computational time that produces the best result is shown in Table 10 and Figure 25.

In the heuristic search algorithm, the tradeoff is between solution quality and computational time, meaning that this algorithm outperforms the other algorithms in computational time, but perhaps provides a less feasible solution that the other algorithms. The main reason for the fast computational time of the heuristic search algorithm is in the mechanism by which the algorithm accumulates the current best sub-problem leading to the complete solution, which is fast because of our propose methods had reduced the number of possible combination to be computed. In addition, heuristic search algorithm also benefits from more efficient usage of the available computer memory by storing the objective value of all possible sub-problems in memory, minimising the need for recalculating the objective function. For a training program with a small number of training days, a small amount of memory space is required, yet still allowing fast computational time. Table 10 and Figure 25

Table 10 Computation time for different approaches

Approaches / Evaluation	Heuristic Search Algorithm (1day sub-problem, 7days taper period)	Genetic Algorithm (NPOP=100,Tournsize=3, CX=0.6,MUT=0.01)	Particle Swarm Optimization
Time (ms)	0:00:07	0:03:18	0:02:07

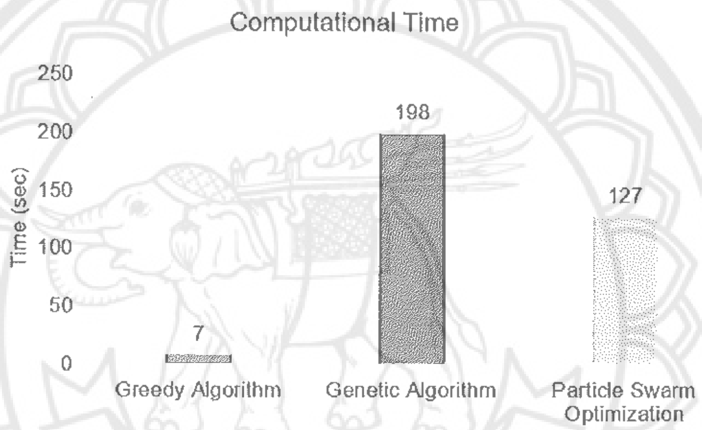

Figure 25 Computation Time of different approaches

The second fastest algorithm is the PSO algorithm which has moderate computational complexity. but provided the best quality solution in our study. In comparison with the GA, the PSO utilises reproduction methods and selection methods, which are simpler computational mechanisms. A good quality solution is one that is widely spread across the entire search space which illustrated in Figure 9.

The search for a better solution, by moving particles across the wider search space, seems to work in this kind of problem.

In our study, the slowest algorithm was the GA. It has many methods to fine tune, including the selection methods, the crossover methods and the mutation methods. The candidate solutions are found by manipulating chromosomes by the crossover methods when new and better solutions are sought by exploring different regions of the search space. Mutation is considered as exploitation that looking for better candidate solution that lie nearby the best-known solutions. These methods, however, are computationally more complex, therefore are slower, making this algorithm unable to outperform others.

Quality of solution

The best results from the 3 difference approaches: the heuristic search algorithm, GA and PSO, are discussed in this section by comparing the athletic performance and constraints violations achieved in each approach. The athletic performance and constraints violations of the best training plan produced by each approach, together with our standard training plan - the UK training plan, are demonstrated in Table 11 and Figure 26.

Table 11 Quality of the best solution from different approaches

Approaches / Evaluation	Heuristic Search Algorithm	Genetic Algorithm	Particle Swarm Optimization	UK Training Plan
Athletic Performance	10.83166	7.75507	13.43600	7.38231
Monotony	1.095431	1.26277	1.45200	1.26011
CTL ramp rate (times)	0	0	0	17
Daily Training Load (times)	0	0	0	0

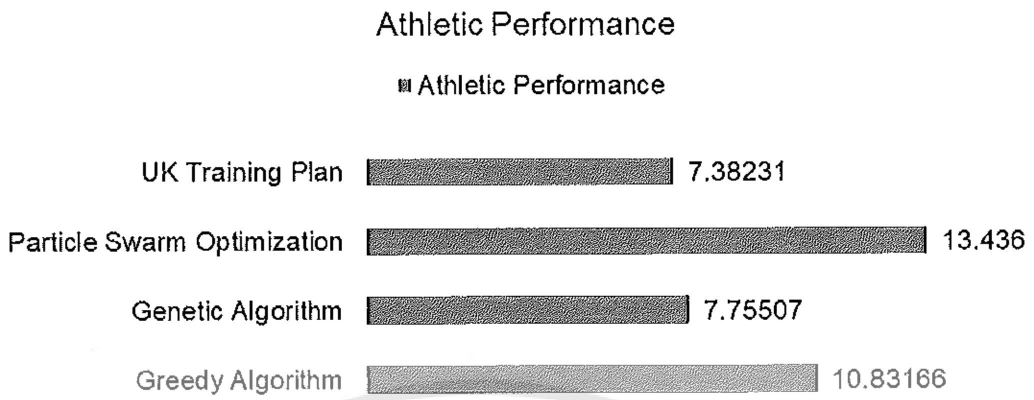

Figure 26 Athletic performance from different approaches

For athletic performance, all approaches found better solutions than achieved in our standard. The lowest athletic performance was in the training plan from the GA algorithm with ε-constraint methods, which is close to the level of athletic performance achieved in our standard training plan. However, the GA algorithm with ε-constraint methods provided a better solution in terms of feasibilities. The GA algorithm with ε-constraint methods found the best solution with regard to all of the physiological constraints.

The second best solution in athletic performance was the result of the heuristic search algorithm which raised athletic performance by 10.83, which is 46.72% higher than our standard plan, with regard to all physiological constraints.

The best solution that raised the highest athletic performance among all proposed approaches while properly taking regard of all physiological constraints, was the plan produced by the PSO algorithm. That training plan raised athletic performance by 13.436, or 82.01%, higher than our standard.

Benchmarking with a commercial training plan

Several commercial applications present some good training plans, such as Sufferfest®, TrainingPeak®, and Today's Plan®. For our benchmarking action, we used the 8-WEEKS UCI GRAN FONDO TIME TRIAL training plan (Henderson, & Cassin, n.d.), provided by Sufferfest®, as a benchmarking training plan. The Sufferfest®'s training plan was preprocessed and analyzed by the same method as the

British Cycling's training plan. The comparison between the British Cycling's training plan and the Sufferfest®'s training plan is shown in Table 12 and Figure 27.

The Sufferfest®'s training plan raised athletic performance higher than the UK training plan by 38.42% while, at the same time, properly considering the monotony constraint and CTL ramp rate constraint. However, The Sufferfest®'s training plan presented many high training load sessions which had TRIMP up to 850. This is the reason for the very high athletic performance arising from the Sufferfest®'s training plan.

Table 12 Comparison between British cycling's training plan and sufferfest training plan

Training plan	Performance	Monotony	CTL ramp rate (times)
Sufferfest's Training Plan	10.21891	0.93583	3
UK's Training Plan	7.38231	1.26011	17

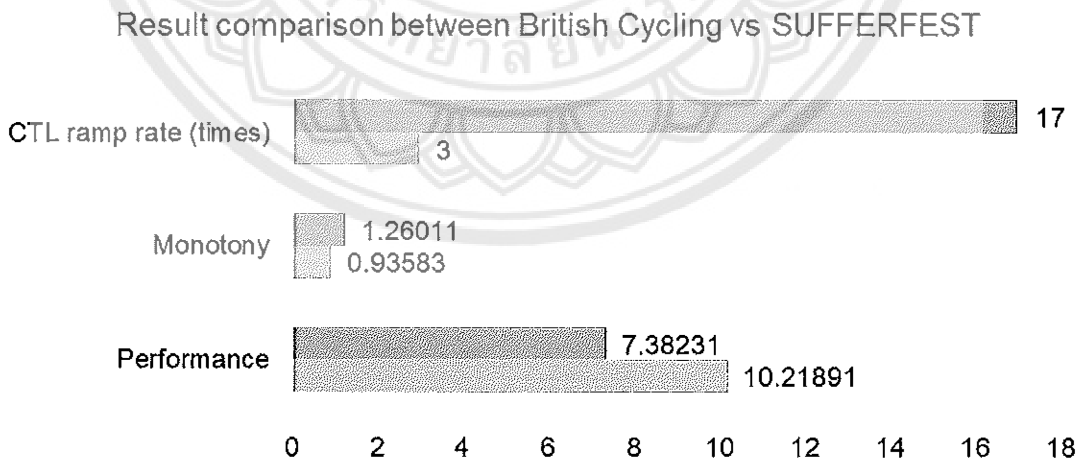

Figure 27 Result comparison between British cycling vs Sufferfest application

In our proposed approach, we needed to adjust the daily TRIMP constraint's threshold to 850 before benchmarking against the Sufferfest®'s training plan. The result of our proposed approach, after adjustment, is shown in Table 13 and Figure 28.

Our modifed approach outperformed the Sufferfest®'s training plan by raising athletic performance by 80.21%, while taking proper regard of all physiological constraints.

Table 13 Comparison between revised PSO training plan and sufferfest training plan

Training plan	Performance	Monotony	CTL ramp rate (times)	Daily TRIMP (times)
Sufferfest's Training Plan	10.21891	0.93583	3	0
PSO Training Plan	18.41585	1.35790	0	0

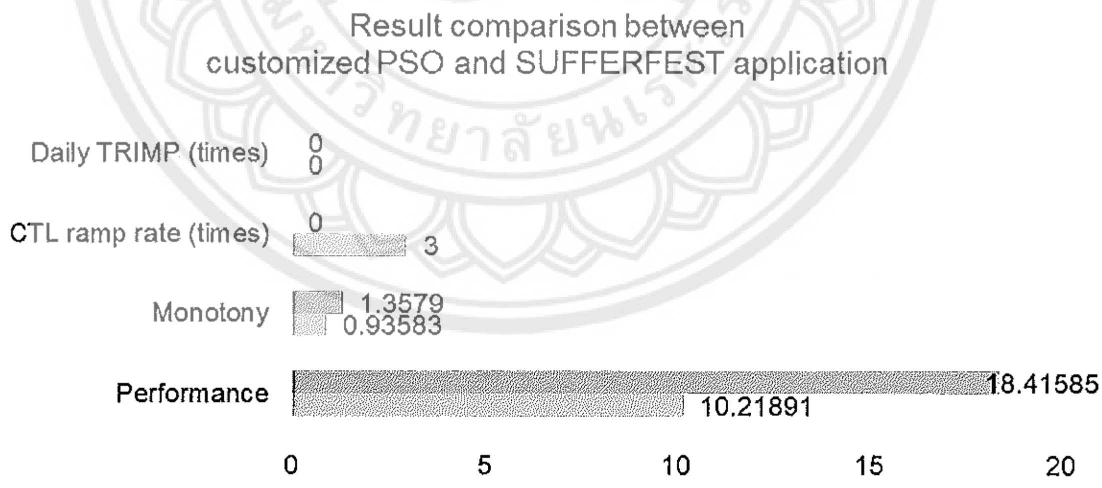

Figure 28 Result comparison between customized PSO and Sufferfest application

CHAPTER VIII

CONCLUSION

Summary of research

In our study, we used several algorithmic approaches to develop an optimal cycling training plan for developing a high level of athletic performance. We included the requirement for all physiological constraints, training monotony, CTL ramp rate, and daily TRIMP limitation, to be considered in the training plan.

We first carefully studied the problem of scheduling a sports training plans. Coggan's training zone, emphasising personal heartrate, was used to encode the personalized training data. Banister's TRIMP, based on average heartrate, was used to quantify the training load of a training session. Then, Banister's Training-Performance Interaction Model was used to estimate the potential rise in athletic performance achievable in a sequence of training sessions prescribed in a training plan.

We identified and studied several optimization techniques that have been previously published, including the heuristic search algorithm, the genetic algorithm (GA), and the Particle Swarm Optimization algorithm (PSO). Each technique applies the ε-constraint method as the constraint handling method.

In our comparative analysis, we determined that the heuristic search algorithm, a deterministic technique, results in a sports training plan that raises athletic performance substantially. However, this algorithm does not consider the physiological constraints. As well, the heuristic search algorithm outperforms all others techniques in terms of computation time. A further drawback of this algorithm is that the researcher needs to spend considerable effort on problem analysis, which is critical in deterministic optimization, and requires significant expertise on the part of the researcher.

The two other techniques, GA and PSO, are stochastic optimization techniques which also showed good potential in searching for the optimal sports training plan. The proposed GA algorithm resulted in a training plan that raised athletic performance to the same level as our standard plan that was based on the

British Cycling organisation's standard plan, and also outperformed our standard in terms of proper consideration of all physiological constraints The overall best sports training plan, which potentially raised athletic performance to a high level, while properly handling all constraints, was the PSO algorithm.

Contribution of PhD research

1. Based on our benchmarking, the PSO algorithm presented the best training program. This algorithm can be applied to the creation of sports training plans for any athletic activity, especially endurance dominant sports such as running, swimming and cycling. The algorithm can be customized to meet user-defined criteria, such as training plan length, desired athletic performance outcomes, and considering the remaining time until their goal event.

2. The PSO algorithm can create sports training plan that considers any given physiological constraints potentially experienced in a training program. These constraint parameters can be modified to meet the user's requirements, such as matching daily training load with the athlete's starting level of physical fitness. The approach can also optimise the ε-constraint method by adjusting the ε-level value. This fine-tuning mechanism enables the algorithm to emphasize the objective value or the sum constraint violation. The result from this approach can be very flexible in meeting the user's requirements while ensuring that the training plan minimizes training risk.

3. The PSO algorithm can also be applied to other combinatorial problems that adaptively negotiate the strength of constraints handling, in real-time, in some cases.

Recommendations for future research

Training-performance interaction model

Before using the PSO algorithm in the real environment, the athlete should be aware that this training-performance interaction model was fitted by collecting data from samples which were cyclists who had extensive training experience. Choosing the parameters values, such as decay rate of both fatigue and fitness and gained coefficient of both physical body fitness and fatigue, is preferred to ensure the precision of the model result. The parameter values that we adopted from the original work from the British Cycling organization are acceptable for estimating the current fitness, fatigue, and athletic performance.

However, if the difference of the decay rate of fatigue between the athlete, and Banister's sample, is too large, the athlete will run the risk of becoming overtrained or detrained, both undesirable outcomes, even though the athlete trains according to the plan.

Future research on this topic could consider using some biological markers that represent the fatigue level, and develop the method to transform this marker into a user- defined model parameter.

An adaptive training plan

Cyclists normally have 'a day job', which requires their attendance at their place of work during normal business hours. Many athletes are likewise family people, with family and home activities. This may mean that, sometimes, they must skip some training sessions because of matters arising in their job, such as unexpected overtime job, or family tasks arising from schooling, or children's illness. These unexpected tasks delay the peak performance time. The training plan should be automatically adjusted to use the remaining time until the race day to raise the athletic performance as high as possible.

Mobile-specific optimization techniques.

Mobile computing, via smartphones, tabs and lightweight laptops are now prevalent, allowing AnyTime/Anywhere access to online services and apps which are also now available in great number and style for mobile platforms. These platforms do, however, have some operational limitations, such as relatively limited battery life, lower computing power, and small screen area. Given that the PSO algorithm that we recommend demands quite high computational power, we need more efficient and highly optimized languages if we are to develop a PSO-based app for mobile devices.

Alternatively, as we have demonstrated in our benchmarking of the traditional approach, it consumes less CPU power due to its less complexity. Further research into the analysis and design the traditional algorithm to make it more effective is suggested. This would approach the mobile platform conundrum stated from the direction of algorithmic ability while taking advantage of its low demands for computing power, for resource-limited platforms.

REFERENCES

REFERENCES

Abraham, L. (2015). *Pyswarm - particle swarm optimization (PSO) that supports constraints (Version 0.6).* Retrieved May 9, 2016, from https://github.com/tisimst/pyswarm

Ackland, J. (1999). *The complete guide to endurance training* (3rd ed.). London: A & C Black.

Allen, H., & Coggan, A. (2012). *Training and racing with a power meter.* Boulder, US: VeloPress.

Alves, R. T., Delgado, M. R., Lopes, H. S., & Freitas, A. A. (2004, September). An artificial immune system for fuzzy-rule induction in data mining. In *International Conference on Parallel Problem Solving from Nature* (pp. 1011-1020). Berlin, Heidelberg: Springer.

Anonymous. (n.d.). *TSS based on heart rate - app at Movescount.com.* Retrieved October 6, 2016, from http://www.movescount.com/apps/app10040232-TSS_based_on_heart_rate

Arif, A., Javed, F., & Arshad, N. (2014). Integrating renewables economic dispatch with demand side management in micro-grids: a genetic algorithm-based approach. *Energy Efficiency, 7*(2), 271–284.

Atan, T., & Hüseyinoğlu, O. P. (2017). Simultaneous scheduling of football games and referees using Turkish league data. *International Transactions in Operational Research, 24*(3), 465-484.

Babraj, J. A., Cuthbertson, D. J., Smith, K., Langberg, H., Miller, B., Krogsgaard, M. R., … Rennie, M. J. (2005). Collagen synthesis in human musculoskeletal tissues and skin. *American Journal of Physiology-Endocrinology and Metabolism, 289*(5), E864–E869.

Banister, E. W. (1991). Modeling Elite Athletic Performance. In Macdougal, J. D., Wenger, H. A., & Green, H. J. (eds.) *Physiological Testing of Elite Athletes* (pp. 403–424). Champaign, Illinois: Human Kinetics.

Behm, D. G., & Sale, D. G. (1993). Intended rather than actual movement velocity determines velocity-specific training response. *Journal of Applied Physiology, 74*(1), 359–368.

Biethahn, J., & Nissen, V. (2012). *Evolutionary algorithms in management applications.* Berlin: Springer Science & Business Media.

Bompa, T., & Buzzichelli, C. (2015). *Periodization Training for Sports* (3rd ed.). Champaign, IL: Human kinetics.

Britishcycling. (n.d.). *Introduction to the foundation plan for intermediate / advanced riders.* Retrieved October 12, 2016, from https://www.britishcycling.org.uk/knowledge/article/izn20140929-Training-Introduction-to-the-Foundation-Plan-for-Intermediate---Advanced-0

Brzostowski, K., Drapała, J., Dziedzic, G., & Swiatek, J. (2015). Algorithm to Plan Athlete's Prolonged Training Based on Model of Physiological Response. In *Asian Conference on Intelligent Information and Database Systems* (pp. 273–283). Cham: Springer.

Busso, T., Denis, C., Bonnefoy, R., Geyssant, A., & Lacour, J. R. (1997). Modeling of adaptations to physical training by using a recursive least squares algorithm. *Journal of Applied Physiology, 82*(5), 1685–1693.

Chang, F. S., Wu, J. S., Lee, C. N., & Shen, H. C. (2014). Greedy search based multi objective genetic algorithm for emergency logistics scheduling. *Expert Systems with Applications, 41*(6), 2947–2956.

Chen, J. C., Wu, C. C., Chen, C. W., & Chen, K. H. (2012). Flexible job shop scheduling with parallel machines using genetic algorithm and grouping genetic algorithm. *Expert Systems with Applications, 39*(11), 10016–10021.

Coggan, A. (2008). *The Science of the Performance Manager*. Retrieved October 3, 2016, from http://home.trainingpeaks.com/blog/article/the-science-of-the-performance-manager

Coggan, A. R. (2003). *Training and racing using a power meter: An introduction, site Inspired Performance Multisports.* N.P.: n.d.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to Algorithms* (3rd ed.). Cambridge: MIT Press.

Crameri, R. M., Langberg, H., Teisner, B., Magnusson, P., Schrøder, H. D., Olesen, J. L., … Kjaer, M. (2004). Enhanced procollagen processing in skeletal muscle after a single bout of eccentric loading in humans. *Matrix Biology : Journal of the International Society for Matrix Biology, 23*(4), 259–264.

Devos, O., Downey, G., & Duponchel, L. (2014). Simultaneous data pre-processing and SVM classification model selection based on a parallel genetic algorithm applied to spectroscopic data of olive oils. *Food Chemistry,* 148, 124–130.

Doessing, S., & Kjaer, M. (2005). Growth hormone and connective tissue in exercise. *Scandinavian Journal of Medicine & Science in Sports, 15*(4), 202–210.

Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. In Micro Machine and Human Science. In *Proceedings of the Sixth International Symposium* (pp. 39–43). Nagoya: IEEE.

Engelbrecht, A. P. (2014). Fitness function evaluations: A fair stopping condition?. In *Symposium on Swarm Intelligence* (pp. 1–8). Orlando, FL: IEEE.

Faghihi, V., Reinschmidt, K. F., & Kang, J. H. (2014). Construction scheduling using genetic algorithm based on building information model. *Expert Systems with Applications, 41*(16), 7565–7578.

Fister, I., Rauter, S., Yang, X. S., & Ljubič, K. (2015). Planning the sports training sessions with the bat algorithm. *Neurocomputing,* 149, 993-1002.

Fortin, F.-A., Rainville, D., Gardner, M.-A. G., Parizeau, M., & Gagné, C. (2012). DEAP: Evolutionary algorithms made easy. *The Journal of Machine Learning Research, 13*(1), 2171–2175.

Foster, C. (1998). Monitoring training in athletes with reference to overtraining syndrome. *Medicine & Science in Sports & Exercise, 30*(7), 1164–1168.

Friel, J. (2009). *Estimating Training Stress Score (TSS).* Retrieved June 27, 2018, from https://www.trainingpeaks.com/blog/estimating-training-stress-score-tss/

Hassan, R., Cohanim, B., de Weck, O., & Venter, G. (2005). A comparison of particle swarm optimization and the genetic algorithm. In *46th AIAA/ASME/ASCE/ AHS/ASC Structures, Structural Dynamics and Materials Conference* (Vols. 1–0). Austin, TX: American Institute of Aeronautics and Astronautics.

Hejazi, F., Toloue, I., Jaafar, M. S., & Noorzaei, J. (2013). Optimization of earthquake energy dissipation system by genetic algorithm. *Computer-Aided Civil and Infrastructure Engineering, 28*(10), 796–810.

Henderson, S. N., & Cassin, M. (n.d.). *UCI Gran fondo world championship time trial training plan, Sufferfest*. Retrieved April 3, 2017, from https://app.thesufferfest.com/plans

Hoseini, P., & Shayesteh, M. G. (2013). Efficient contrast enhancement of images using hybrid ant colony optimisation, genetic algorithm, and simulated annealing. *Digital Signal Processing, 23*(3), 879–893.

Howard, J. D., Ritchie, M. R., Gater, D. A., Gater, D. R., & Enoka, R. M. (1985). Roundtable: Determining factors of strength physiological foundations. *Strength & Conditioning Journal, 7*(6), 16–22.

Huang, C. C., Liu, H.-M., & Huang, C. L. (2015). Intelligent scheduling of execution for customized physical fitness and healthcare system. *Technology and Health Care, 24*(s1), S385–S392.

Jarboui, B., Ibrahim, S., Siarry, P., & Rebai, A. (2008). A combinatorial particle swarm optimisation for solving permutation flow shop problems. *Computers & Industrial Engineering, 54*(3), 526–538.

Jolliffe, I. T. (1986). Principal Component Analysis and Factor Analysis. In *Principal Component Analysis* (pp. 115–128). New York: Springer.

Kanan, H. R., & Nazeri, B. (2014). A novel image steganography scheme with high embedding capacity and tunable visual image quality based on a genetic algorithm. *Expert Systems with Applications, 41*(14), 6123–6130.

Kennedy, J., & Eberhart, R. (1995). *Particle swarm optimization. In International Conference on Neural Networks* (pp.1942-1948). Nagoya: IEEE.

Kjaer, M., Langberg, H., Miller, B. F., Boushel, R., Crameri, R., Koskinen, S., … Hansen, M. (2005). Metabolic activity and collagen turnover in human tendon in response to physical activity. *Journal Musculoskelet Neuronal Interact, 5*(1), 41–52.

Kjær, M., Magnusson, P., Krogsgaard, M., Møller, J. B., Olesen, J., Heinemeier, K., … Esmarck, B. (2006). Extracellular matrix adaptation of tendon and skeletal muscle to exercise. *Journal of Anatomy, 208*(4), 445–450.

Kumyaito, N., Yupapin, P., & Kreangsak, T. (2017). Personalized sports training plans with physiological constraints using the ε- constraint method with a genetic algorithm. *Far East Journal of Electronics and Communications, 17*(2), 475–496.

Kumyaito, N., Yupapin, P., & Tamee, K. (2018). Planning a sports training program using adaptive particle swarm optimization with emphasis on physiological constraints. *BMC Research Notes*, 11, 9.

Langberg, H., Ellingsgaard, H., Madsen, T., Jansson, J., Magnusson, S. P., Aagaard, P., & Kjær, M. (2007). Eccentric rehabilitation exercise increases peritendinous type I collagen synthesis in humans with Achilles tendinosis. *Scandinavian Journal of Medicine & Science in Sports, 17*(1), 61–66.

Martis, R. J., Prasad, H., Chakraborty, C., & Ray, A. K. (2014). The Application of Genetic Algorithm for Unsupervised Classification of ECG. In *Machine Learning in Healthcare Informatics* (pp. 65–80). Berlin, Heidelberg: Springer.

Meng, F.-W., Chen, K.-C., Lin, K.-C., & Chen, R.-C. (2014). Scheduling volleyball games using linear programming and genetic algorithm. *Information Technology Journal, 13*(15), 2411.

Mezyk, E., & Unold, O. (2011). Machine learning approach to model sport training. *Computers in Human Behavior, 27*(5), 1499–1506.

Miller, B. F., Olesen, J. L., Hansen, M., Døssing, S., Crameri, R. M., Welling, R. J., … Rennie, M. J. (2005). Coordinated collagen and muscle protein synthesis in human patella tendon and quadriceps muscle after exercise. *The Journal of Physiology, 567*(Pt 3), 1021–1033.

Neill, S. P., Hashemi, M. R., & Lewis, M. J. (2014). Optimal phasing of the European tidal stream resource using the greedy algorithm with penalty function. *Energy, 73*(Supplement C), 997–1006.

Novatchkov, H., & Baca, A. (2013). Fuzzy logic in sports: A review and an illustrative case study in the field of strength training. *International Journal of Computer Applications, 71*(6), 8–14.

Pan, Q. K., Tasgetiren, M. F., & Liang, Y. C. (2008). A discrete particle swarm optimization algorithm for the no-wait flowshop scheduling problem. *Computers & Operations Research, 35*(9), 2807–2839.

Perl, J. (2001). PerPot: A metamodel for simulation of load performance interaction. *European Journal of Sport Science, 1*(2), 1-13.

Rashedi, R., & Hegazy, T. (2015). Capital renewal optimisation for large-scale infrastructure networks: genetic algorithms versus advanced mathematical tools. *Structure and Infrastructure Engineering, 11*(3), 253–262.

Rygula, I. (2006). Artificial neural networks as a tool of modeling of training loads. In *Engineering in Medicine and Biology Society, 2005* (pp. 2985-2988). Shanghai, China: IEEE.

Seiler, S., & Tønnessen, E. (2009). Intervals, thresholds, and long slow distance: The role of intensity and duration in endurance training. *Sport Science,* 13, 32–53.

Sha, D. Y., & Hsu, C. Y. (2008). A new particle swarm optimization for the open shop scheduling problem. *Computers & Operations Research, 35*(10), 3243–3261.

Sivanandam, S. N., & Deepa, S. N. (2007). *Introduction to genetic algorithms.* Berlin, Heidelberg: Springer Science & Business Media.

Song, M. X., Chen, K., He, Z. Y., & Zhang, X. (2014). Optimization of wind farm micro-siting for complex terrain using greedy algorithm. *Energy, 67*(Supplement C), 454–459.

Song, M. X., Chen, K., Zhang, X., & Wang, J. (2015). The lazy greedy algorithm for power optimization of wind turbine positioning on complex terrain. *Energy, 80*(Supplement C), 567–574.

Takahama, T., & Sakai, S. (2000). Tuning fuzzy control rules by the α constrained method which solves constrained nonlinear optimization problems. *Electronics and Communications in Japan (Part III: Fundamental Electronic Science), 83*(9), 1-12.

Takahama, T., & Sakai, S. (2005). Constrained optimization by applying the alpha; constrained method to the nonlinear simplex method with mutations. *IEEE Transactions on Evolutionary Computation, 9*(5), 437–451.

Takahama, T., & Sakai, S. (2010). Constrained optimization by the ε-constrained differential evolution with an archive and gradient-based mutation. In *Congress on Evolutionary Computation* (pp. 1–9). Barcelona, Spain: IEEE.

Takahama, Tetsuyuki, & Sakai, S. (2003). Learning fuzzy control rules by α-constrained simplex method. *Systems and Computers in Japan, 34*(6), 80–90.

Takahama, Tetsuyuki, & Sakai, S. (2004). Constrained optimization by a constrained genetic algorithm (AGA). *Syst. Comput. Japan, 35*(5), 11–22.

Takahama, Tetsuyuki, & Sakai, S. (2006). Solving constrained optimization problems by the ε constrained particle swarm optimizer with adaptive velocity limit control. In *Cybernetics and Intelligent Systems* (pp. 1–7). Bangkok: IEEE.

Tamee, K., & Kumyaito, N. (2016). Intelligence planning for aerobic training using a genetic algorithm. In *International Symposium on Natural Language Processing* (pp. 76). Cham: Springer.

Tang, W., Shi, Z., & Wu, Y. (2014). Regularized simultaneous forward-backward greedy algorithm for sparse unmixing of hyperspectral data. *IEEE Transactions on Geoscience and Remote Sensing, 52*(9), 5271–5288.

Trainingpeaks. (n.d.). *Find a training plan.* Retrieved April 3, 2017, from https://www.trainingpeaks.com/training-plans/cycling

Vidal, T., Crainic, T. G., Gendreau, M., & Prins, C. (2013). A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows. *Computers & Operations Research, 40*(1), 475–489.

Wang, Y. (2012). A new hybrid genetic algorithm for job shop scheduling problem. *Computers & Operations Research, 39*(10), 2291–2299.

Welikala, R. A., Fraz, M. M., Dehmeshki, J., Hoppe, A., Tah, V., Mann, S., … Barman, S. A. (2015). Genetic algorithm based feature selection combined with dual classification for the automated detection of proliferative diabetic retinopathy. *Computerized Medical Imaging and Graphics,* 43, 64–77.

Xie, F., & Bovik, A. C. (2013). Automatic segmentation of dermoscopy images using self-generating neural networks seeded by genetic algorithm. *Pattern Recognition, 46*(3), 1012–1019.

Xu, Y., Li, K., Hu, J., & Li, K. (2014). A genetic algorithm for task scheduling on heterogeneous computing systems using multiple priority queues. *Information Sciences*, 270, 255–287.

GLOSSARY

GLOSSARY

Athletic Performance	:	Medical thesaurus that governed by U.S. National Library of Medicine or Medical Subject Heading (MeSH®) had defined Athletic performance as "Carrying out of specific physical routines or procedures by one who is trained or skilled in physical activity. Performance is influenced by a combination of physiological, psychological, and socio-cultural factors."
Fitness	:	Positive outcome from physical exercises. Human gain fitness by the adaptation as preparation to upcoming training load. The coefficient of fitness gain is less than the coefficient of fatigue gain. However, decay rate of fitness is less than decay rate of fatigue. Therefore, correctly training and sufficient resting can elevate the athletic performance.
Fatigue	:	The negative that gained by physical exercise. It's both coefficient of gain term and decay rate are greater than fitness. Thus, training too heavy and too often can cause an overtraining.
Physiological Constraints	:	These constraints determined as threshold that limit to physical body. Training patterns that go beyond this limitation consequence an overtraining or an injury. Cyclists may need to extend their resting period and delay their peak performance. In severe case, cyclist may need a long complete rest period and eliminate all racing events entire season.

GLOSSARY (CONT.)

Constrained Optimization	:	Methods or procedures in searching for a global optimal solution with regarding to problem's constraints. Constraints can be classified as equality constraint and inequality constraint.
Heuristic Search Algorithm	:	This algorithm is classified as deterministic algorithm which search for a global optimal solution in efficient manner. Heuristic search algorithm begins by divide a complete solution's structure into sub structures. The main process is evaluating all possible current sub structures and select the best current sub structure. The selected sub structure will append to previous substructures to construct the complete solution. Finally, the complete solution will be determined as the solution of heuristic search algorithm.
Genetic Algorithm	:	An optimization algorithm that classified as stochastic technique. This algorithm is based on Charles Darwin's evolution theory. Its results are estimated by selection methods and reproduction methods. The algorithm continues their processes until the exit criteria is meet. The best solution in the final iteration determine as genetic algorithm's solution.

GLOSSARY (CONT.)

Particle Swarm Optimization	:	A stochastic algorithm that inspired by nature, a swarm of birds or a school of fishes, in moving all members to their target destination. The concept of PSO begins by randomly initiate all individuals as vectors within search space. The fitness of individuals will be iterated evaluate and consequence as an update of individual's the direction and velocity toward the current best individual. When exit criteria is meet, the best individual will be selected as a solution.
Training Plan	:	Sports fitness training plans are the strategies for achieving peak performance. The objective of training plan is to reach a high level of performance (peak performance) and an athlete has to develop skills, biomotor abilities and psychological traits in a methodical manner.
Reliable Organization	:	A well-known organization in sports cycling, such as national cycling organization and the owner of the famous application for cycling training. In this research, British Cycling and Sufferfest® is defined as reliable organization.