Generating the training plans based on existing sports activities using swarm intelligence

Iztok Fister Jr., Iztok Fister

Abstract Planning the sports training sessions by an evolutionary computation and swarm intelligence based algorithms has been becoming an interesting topic for research. Recently, many methods and techniques were proposed in theory and practice in order to help athletes in sports training. In a nutshell, integrating these methods and techniques in the same framework has resulted in creating an artificial sports trainer with abilities similar to a human trainer. In this Chapter, we intend to extend the artificial sports trainer with an additional feature which enables athletes to generate a training plan on the basis of existing training courses tracked by mobile sports trackers. Experimental results suggest the usefulness of the proposed method.

1 Introduction

Once upon a time, many people began enjoying lighter sports activities. These activities were usually focused on walking, short jogging or even light cycling. Since the raising of mass sports advertisements and sports competitions/events, many people look at sports activities more seriously and, therefore, this has resulted in an expansion of participants in mass sports events [23]. In other words, casual events with a smaller number of participants suddenly became mass sport events. Typically, the mass sports event assembles from hundreds or thousands more participants. For example, running city marathons can be counted in this category of sports that recently became very popular [24]. All over the world, small and big cities organize running marathons where professional, amateur and newbie athletes compete or just

Iztok Fister Jr.
Faculty of Electrical Engineering and Computer Science, University of Maribor, Smetanova 17, 2000 Maribor, Slovenia, e-mail: iztok.fister1@um.si

Iztok Fister
Faculty of Electrical Engineering and Computer Science, University of Maribor, Smetanova 17, 2000 Maribor, Slovenia, e-mail: iztok.fister@um.si

participate together without any special racing goals. On the other hand, more complex sport disciplines as, for example, cycling and multisport disciplines such as triathlons, have also been becoming very popular for a wide range of participants.

Anyway, a participation in mass sports competitions is just one side of the coin. The other side of the coin is the fact that people have to be prepared in order to participate in such sports events. Preparation demands mostly proper sports training. The sports training is always a very complex and intricate process where athletes try to prepare themselves optimally for a sports event or competition. In this process, the presence of a sports trainer was inevitable until recently. The trainer prescribed his/her trainee training plans that need to be performed by athletes in order to increase their performances. To conduct a sports training efficiently, it is a very long process if trainers do not know the characteristics of the trainees' bodies in detail. Unfortunately, the trainers are generally very expensive.

To help athletes with a proper sports training program, an artificial sport trainer was designed that is an inexpensive variant of the real trainer. The artificial sports trainer consists of algorithms from a computational intelligence, which support a wide spectrum of tracking mobile devices and are based on the sports training theory [7]. At this time, it is able to plan sports training sessions very efficiently [7], detects an over-training phenomenon [9], recommends the most efficient food during endurance competitions [8], etc.

In the past 20 years, Computational Intelligence (CI) algorithms have been applied in dozens and dozens of applications. The following families of algorithms belong to the CI:

- Artificial Neural Networks (ANN) [14],
- Artificial Immune Systems (AIS) [6],
- Fuzzy Logic (FL) [26],
- Evolutionary Algorithms (EAs) [28],
- Swarm Intelligence (SI) [16, 13].

Recently, the most popular families of CI are EAs and SI-based algorithms. Typically, these stochastic population-based algorithms are inspired by natural systems. Thus, EAs mimic a Darwinian evolution [5] where the fitter individuals have more chances to survive in the cruel struggle for existence, while in SI, there is a bunch of unintelligent agents capable of performing the simplest tasks, but acting together in a community they exhibit collective behavior that enables them to survive, e.g., searching for a food by ants or building the magnificent buildings in which they live by termites.

Using computer technology to help athletes in training was also a vibrant topic for research in the past 10-15 years. In line with this, many useful applications emerged in theory with practical applicability. At the beginning, researchers applied machine learning techniques in the sport domain [22, 20, 21, 18]. On the other hand, a rapid development has been started of pervasive computing in sports [15, 2, 1, 19].

Mainly, the aim of this Chapter is to present a pioneering work in planning the sports courses by cycling for a some period of time (also cycle). For instance, to select a proper combination of training courses by cycling for a duration of one week is a very hard for sports trainers. In this work, a solution based on a swarm intelligence algorithm is proposed which performs optimal selection of sports courses from predefined course clusters. Initial experiments suggest that our solution is very helpful for athletes with many possibilities for further upgrade.

The structure of the remainder of the Chapter is as follows. Section 2 deals with a short description of the artificial sports trainer. In Section 3, the proposed SI-based algorithm for generating the training plans based on existing cycling courses is illustrated. The experiments and results are the subjects of Section 4. The Chapter finishes with summarizing the performed work and outlines possible directions for a future work.

2 Artificial sports trainer

Recently, the job of a human personal sports trainer has been becoming interesting and very popular. People start to hire personal trainers to help in training tasks due to different promotional activities for a healthy and low fat life in society. On the other hand, many amateur athletes tend to increase their performances in some sports disciplines. A good example of this is the case of the Ironman Triathlon, where thousands and thousands of competitors compete in various Ironman competitions every year to achieve places for the Ironman World Championships taking place in Hawaii. For qualifying on Hawaii an athlete must be very fast and, consequently, finish the race on the podium. To achieve this good result, he/she must be prepared brilliantly for one of the qualification races. Thus, many athletes hire one or more personal trainers who prepare them using the proper training plans. Some personal trainers are very good, but some are not so good. On the other hand, the trainers are pretty expensive and the most of the athletes could not afford them.

After identifying this problem two years ago, the development of an Artificial sports trainer [10, 7] has been started. The main purpose of the artificial sport trainer was to allow all people to have their own inexpensive personal trainers with capabilities comparable with the real sports trainer. It is based on algorithms of computational intelligence and it is able to cover almost all phases of the sports training. The following phases complete a traditional sports training (Fig. 1):

- Planning,
- Realization,
- Control,
- Estimation.

The plan of the specific sports training is accomplished in the first phase. Usually, this training plan is made for some time period, a so-called cycle (e.g., competition season, month, week) and depends on a training strategy. The training strategy is based on goals that need to be achieved at the end of the training period. As a result, all planned training sessions must comply with the strategy. Each training session prescribes a training type (e.g., aerobic, aerobic-anaerobic, anaerobic), magnitude,

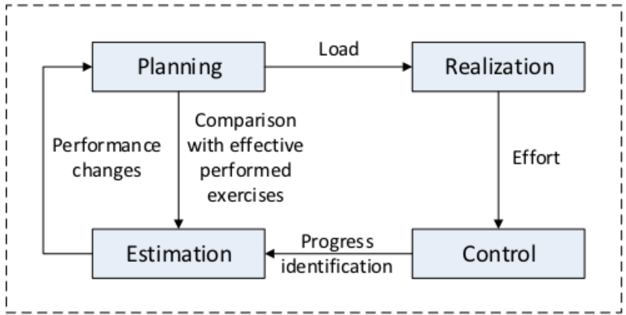


Fig. 1: A training cycle.

intensity and iterations thus defining an exercise load that an athlete needs to have accomplished during the training. In the realization phase, the training session is conducted by an athlete. All realizations of workouts, as demanded by the training session are, controlled by the trainer. In the past, the trainer measured an athlete's invested effort by tracking the workout with measures like a stopwatch for measuring the time, a measuring tape for measuring length, etc. Recently, the quality of realized workouts has been measured by the mobile devices that produce a lot of data by tracking the athlete performing the sports session. This amount of data is suitable for identifying the progress of the athlete by the artificial sports trainer. At the end of the training period, the training cycle is estimated by the trainer. This estimation is performed by comparing the prescribed values and the obtained values during the workouts. However, performance changes are indicated by this comparison that has an impact on the training plan in the next training cycle.

The following CI algorithms can be used for the training period by the artificial sports trainer at the time:

- Generate sports training sessions [7] in the planning phase,
- Generate fitness sessions [11] in the realization phase,
- Predict food [8] in the control phase,
- Help prevent over-training syndrome [9] in the estimation phase.

The efficiency of the artificial sport trainer was validated by a human sport trainer who confirmed that our solution is efficient enough. However, there are still many tasks for integration in the artificial sport trainer. One of these tasks is also generating the training plans based on existing sports activities as presented in the next Section.

3 Generating the training plans

An artificial sports trainer is capable of replacing the real trainer, especially in training phases that demand the processing of a large amount of data and making decisions on their basis. The best suited for an automation of the real trainer tasks are individual sports disciplines like running, cycling, triathlon, where tracking of athletes in training is performed by mobile devices (i.e., smart phones, sports watches, etc.) automatically. These mobile devices generate a huge amount of data saved in a GPS exchange (GPX) or Training Center XML (TCX) dataset formats. Consequently, the artificial sports trainer is capable of classifying, clustering, analyzing and data mining the training datasets and, on this basis, predicting those future training workouts that will most increase the current performances of athletes in training.

In line with this, generating the training plans based on an existing course is devoted to generating such courses that will comply the most with the training strategy on the basis of the current performance of an athlete. The architecture of the algorithm for generating the training plans is illustrated in Fig. 2.

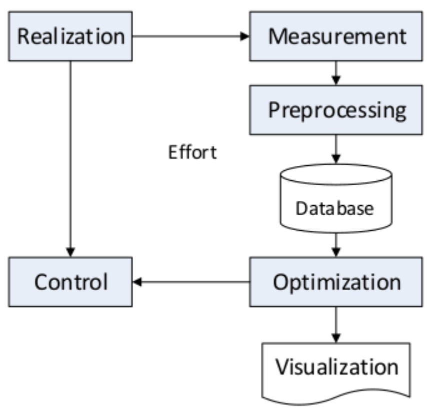


Fig. 2: Architecture of generating the sports training plans based on existing courses.

Input of this kind of sports training generation presents a set of training datasets in GPX and/or TCX format that are obtained in the realization training phase by tracking mobile devices. At first, these datasets are identified and preprocessed, where the characteristic values about the training session are parsed. Then, the identified training sessions are clustered according to the characteristic values of the training session into clusters and the obtained data are saved into a database. Finally, an optimization SI-based algorithm is launched that generates the optimal training course. On the one hand, this course is transferred to the control phase, while it can be visualized on the other hand.

Consequently, the proposed algorithm consists of the following stages:

- Preprocessing,
- Optimization,
- Visualization.

In the remainder of the Chapter, the proposed stages are presented in detail.

3.1 Preprocessing

The preprocessing phase consists of four steps:

- Assembling and identifying a set of sports activities,
- Parsing the characteristics values in training data,
- Determining the training load indicators,
- Clustering.

A set of sport activity datasets was assembled and identified in the first step [25]. Then, the characteristics training values needed for the next step were parsed from the GPX and/or TCX datasets. Typically, these datasets are tracked by sport trackers (e.g., Garmin and Polar watches, or smart phones). Interestingly, athletes can upload a complete dataset on the server where it is archived. Additionally, there are many online services and applications to visualize these activities (Garmin Connect, Strava, Endomondo). A simple example of TCX dataset is presented in Table 1.

Table 1: An example of sport activity stored in a tcx file.

```
⟨Activity Sport="Biking"⟩
 ⟨Id⟩2012-03-06T14:16:05.000Z⟨/Id⟩
 ⟨Lap StartTime="2012-03-06T14:16:05.000Z"⟩
  ⟨TotalTimeSeconds⟩1902.72⟨/TotalTimeSeconds⟩
  ⟨DistanceMeters⟩15042.79⟨/DistanceMeters⟩
  ⟨MaximumSpeed⟩16.774999618530273⟨/MaximumSpeed⟩
  ⟨Calories⟩273⟨/Calories⟩
  ⟨AverageHeartRateBpm⟩
   ⟨Value⟩119⟨/Value⟩
  ⟨/AverageHeartRateBpm⟩
  ⟨MaximumHeartRateBpm⟩
   ⟨Value⟩153⟨/Value⟩
  ⟨/MaximumHeartRateBpm⟩
  ⟨Intensity⟩Active⟨/Intensity⟩
  ⟨TriggerMethod⟩Manual⟨/TriggerMethod⟩
  ⟨Track⟩
   ⟨Trackpoint⟩
    ⟨Time⟩2012-03-06T14:16:05.000Z⟨/Time⟩
    ⟨Position⟩
     ⟨LatitudeDegrees⟩46.07827073894441⟨/LatitudeDegrees⟩
     ⟨LongitudeDegrees⟩14.516056096181273⟨/LongitudeDegrees⟩
    ⟨/Position⟩
```

As can be seen from Table 1, each TCX dataset starts with identification of the training session (i.e., activity). Then, the summary data about the activity follows, i.e., a duration time, an overcame length, an average heart rate, etc. The dataset ends with a detailed description of the activity started with an XML tag ⟨Track⟩ and followed by a set of track-points each started with ⟨Trackpoint⟩ an XML tag determining the current position of an athlete at a specific time. Track-points are tracked in some time interval depending on the tracking device (e.g., each second).

In the parsing step, characteristic values are parsed from each activity dataset. These characteristics values are presented in bold in Table 1 and are needed for calculating the corresponding training load indicator. The training load is defined as the stress placed upon the body as a result of the training session [3]. There are more training load indicators. The basic TRIMP (TRaining IMPulse) was employed in this study. This training load indicator was at first proposed by Banister et al. in 1991 [3, 4] and it is simple, expressed as

$$\mathrm{TRIMP} = t \cdot \overline{HR}, \tag{1}$$

where t denotes a duration in minutes (min) and $\overline{HR}$ is an average heart rate in beats per minute (bpm). As can be seen from Eq. (1), the training load is defined as a product of duration and average heart rate. However, the main disadvantage of this indicator is that it is insensitive to the different levels of trainings. This means that long-term sports trainings of low intensity (i.e., with lower average HR) and the short-term sports trainings of high intensity (i.e., with higher average HR) have the similar TRIMP values.

Finally, the set of training sessions are clustered according to the identified TRIMP training load indicator in order to obtain groups of training sessions of the similar intensities. Here, k-means clustering [12] was taken into consideration. Thus, three TRIMP training zones are observed containing a low, medium and high intensive training sessions, where each point in the diagram on Fig. 3 presents the TRIMP training load indicator of a specific training session. Let us notice that points with green color represent the low intensive training activities, blue points the medium intensive training activities, while the red points are the high intensive training activities. Centers of clusters denoted as squares in the figure indicates the similar values of $HR = 130$ but of different duration. Therefore, the higher the duration higher the TRIMP. The other training sessions are dispersed around these centers by the average HR and duration t. The most intensive training sessions in each cluster are characterized by the high average HR and long duration t.

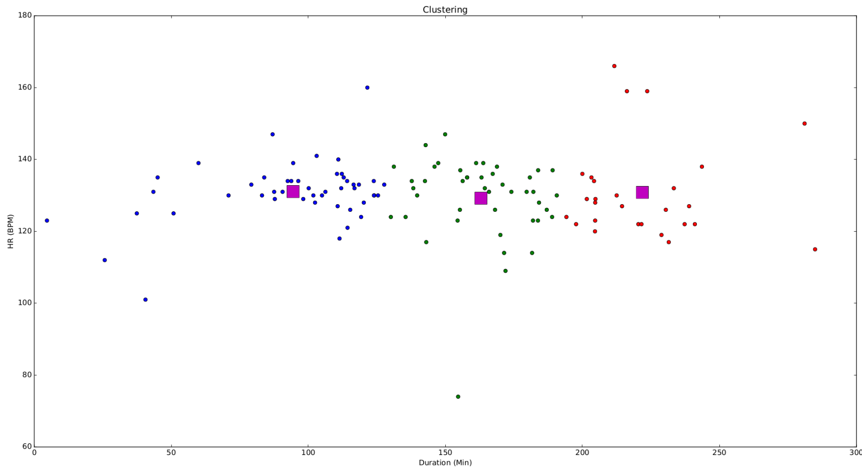


Fig. 3: An example of clustering

Intensity zones determine the intensity of the sports training session. However, the intensity load cannot be measured directly, because it is the result of complex chemical reactions carried out in an athlete's organism. Typically, it is estimated indirectly based on the average heart rate *HR*. Usually, two measures are employed to determine which stress is placed upon the athlete's body by performing some workout, i.e., *HR* zone and % max *HR*. In this study, four intensity zones were defined that cover the *HR* zones as illustrated in Table 2.a. The k-means clustering splits identified sports training sessions into three clusters as presented in Table 2.b.

Intensity zone	*HR* zone	% max *HR*
1	60-100	30-50%
2	100-140	50-70%
3	140-180	70-90%
4	180-200	90-100%

(a) Training intensity zones.

Cluster	Average *HR*	Duration *t*	TRIMP
1	130 min	90 bpm	11,700
2	130 min	160 bpm	20,800
3	130 min	215 bpm	27,950

(b) TRIMP training zones.

Table 2: Characteristics of the TRIMP clusters.

Let us notice that the *HR* zones and % max *HR* are calculated for an athlete with the % max $HR = 200$ in Table 2.a. Interestingly, the addressed set of sports activity datasets includes only training sessions in an aerobic *HR* zone (i.e., $50-70$ % max *HR*) and an aerobic-anaerobic *HR* zone (i.e., $70-90$ % max *HR*), while anaerobic *HR* zone (i.e., $90-100$ % max *HR*) is not presented in the current dataset. Training sports sessions of intensity $30-50$ % max *HR* indicate a resting phase. Interestingly, the training sessions of all observed intensity zones are observed in each cluster. As can be seen from Table 2.b, the TRIMP training load indicator increases by raising the duration t.

The obtained clusters are saved into a database, from which they serve as an input for an optimization phase.

3.2 Optimization process

The task of an optimization process is to accomplish a sports training plan for the duration of a training cycle (e.g., one week) in which a training sessions of different intensities are distributed such that the less intensive training sessions are planned at the start and the end of the cycle, while the more intensive are in the middle of the cycle. The clustering algorithm assembles the training sessions into three clusters where the intensity of training sessions increases from the first to the third cluster. As a result, the third cluster comprises the most intensive sports sessions. The optimization algorithm needs to include the proper training sessions from different clusters into the training plan so that the demanded intensity distribution is considered.

At the moment, we are a witnessing the emerging the highly effective computational intelligence algorithms which find their inspiration from nature. These so-called nature-inspired algorithms are very useful for solving the continuous, as well as discrete, optimization problems. A recent member of this family is a Bat Algorithm (BA) developed by Yang [29] that is, besides its simplicity, also capable of solving complex optimization problems efficiently. In this Chapter, we took the BA for searching for the optimal training courses. It was used successfully for many tasks in design and development of the artificial sport trainer and, thus, it is also appropriate for generating the training plans based on existing courses.

The original BA is presented in the remainder of the Chapter. Then, the modified BA is discussed for the optimal training plan generation.

3.2.1 Bat algorithm

The BA was developed by Yang in 2010 [29], where the phenomenon of echolocation arisen by some kind of micro-bats is explored as an inspiration for this stochastic population-based optimization algorithm. Mainly, the phenomenon is used by the micro-bats for orientation in the dark. In order to apply it to the algorithm, the author simplified the complex behavior of the natural bats with the following three rules:

- All bats use echolocation to sense distance to target objects.
- Bats fly randomly with the velocity v_i at position x_i, the frequency $Q_i \in [Q_{min}, Q_{max}]$ (also the wavelength λ_i), the rate of pulse emission $r_i \in [0,1]$, and the loudness $A_i \in [A_0, A_{min}]$. The frequency (and wavelength) can be adjusted depending on the proximities of their targets.

- The loudness varies from a large (positive) A_0 to a minimum constant value A_{min}.

The BA maintains a swarm of the virtual bats, where each bat in the swarm represents a solution of the problem to be solved. Thus, each solution is represented as a real-coded vector

$$\mathbf{x}_i = [x_{i,1}, \ldots, x_{i,D}]^T, \quad \text{for } i = 1, \ldots, Np, \tag{2}$$

where t is a current generation, D denotes a dimensionality of the problem, and Np the number of virtual bats in the swarm. Actually, each vector determines a position of the virtual bat in a search space. Bats move towards the best bat position and thus explore the new regions of the search space. The BA supports two strategies of an exploration of the search space. The former moves the virtual bat according to the equation

$$\begin{aligned} Q_i^{(t)} &= Q_{min}^{(t)} + \left(Q_{max}^{(t)} - Q_{min}^{(t)}\right) \cdot \beta, \\ \mathbf{v}_i^{(t+1)} &= \mathbf{v}_i^{(t)} + \left(\mathbf{x}_i^{(t)} - \mathbf{x}_{best}^{(t)}\right) \cdot Q_i^{(t)}, \\ \mathbf{x}_i^{(t+1)} &= \mathbf{x}_i^{(t)} + \mathbf{v}_i^{(t)}. \end{aligned} \tag{3}$$

where a pulse frequency can vary in the interval $Q_i^{(t)} \in [Q_{min}, Q_{max}]$, a random number $\beta \in [0,1]$ specifies the output pulse, and $\mathbf{x}_{best}^{(t)}$ presents currently the best solution. The latter improves the current bat position according to the equation

$$\mathbf{x}_i^{(t+1)} = \mathbf{x}_{bast} + \varepsilon \cdot L(s, \alpha), \tag{4}$$

where $\varepsilon > 0$ is the step size scaling factor, $L(s, \alpha)$ the Lévy flight alpha-stable distribution with parameters scale s and exponent $\alpha \in (0,2]$. The distribution reduces to Gaussian distribution for $\alpha = 2$ and to Cauchy distribution for $\alpha = 1$. The mentioned strategy is more exploitative than those illustrated in Eq. (4), and presents a kind of random walk that is focused primarily focused on exploring the vicinity of the current best solution.

Let us notice that both exploration strategies are balanced in the search process using the parameter pulse rate $r_i^{(t)} \in [0,1]$. The closer the bat to the prey, the higher the pulse rate and vice versa. The evaluation function evaluates the quality of the generated solutions and models the optimization problem into the BA. Interestingly, the better trial solution replaces the on the same position laid solution in the current swarm only conditionally, i.e., according to the loudness $A \in [A_0, A_{\min}]$. The motivation behind using this parameter lays in simulated annealing [17], where the better trial solution replaces the current solution only under some probability in order to avoid getting stacking into local optima.

The algorithm's pseudo-code is presented in Algorithm 1, whose main algorithm components are summarized as follows:

- *Initialization* (lines 1-3): Initializing the algorithm parameters, generating the initial population, evaluating this, and, finally, determining the best solution $\mathbf{x}_{best}$ in the population,

Algorithm 1 Bat algorithm

Input: Bat population $\mathbf{x_i} = (x_{i1}, \ldots, x_{iD})^T$ for $i = 1 \ldots Np$, MAX_FE.
Output: The best solution $\mathbf{x}_{best}$ and its corresponding value $f_{min} = \min(f(\mathbf{x}))$.
1: init_bat();
2: *eval* = evaluate_the_new_population;
3: f_{min} = find_the_best_solution($\mathbf{x}_{best}$); {Initialization}
4: **while** termination_condition_not_meet **do**
5:   **for** $i = 1$ **to** Np **do**
6:     $\mathbf{y}$ = Generate_new_solution($\mathbf{x}_i$);
7:     **if** $\text{rand}(0,1) > r_i$ **then**
8:       $\mathbf{y}$ = Improve_the_best_solution($\mathbf{x}_{best}$)
9:     **end if**{ Local search step }
10:     f_{new} = Evaluate_the_new_solution($\mathbf{y}$);
11:     $eval = eval + 1$;
12:     **if** $f_{new} \leq f_i$ **and** $\text{N}(0,1) < A_i$ **then**
13:       $\mathbf{x}_i = \mathbf{y}$; $f_i = f_{new}$;
14:     **end if**{ Save the best solution conditionally }
15:     f_{min}=Find_the_best_solution($\mathbf{x}_{best}$);
16:   **end for**
17: **end while**

- *Generate_the_new_solution* (line 6): Moving the virtual bats in the search space according to the physical rules of bat echolocation,
- *Local_search_step* (lines 7-9): Improving the best solution using the Random Walk Direct Exploitation (RWDE) heuristic,
- *Evaluate_the_new_solution* (line 10): Evaluating the new solution,
- *Save_the_best_solution_conditionaly* (lines 12-14): Saving the new best solution under some probability A_i,
- *Find_the_best_solution* (line 15): Finding the current best solution.

The original BA is devoted primarily for continuous optimization. When the algorithm is applied to another kind of problems some modifications need to be applied into it. Therefore, necessary modifications for developing the BA for sports training course generation are presented in the next subsection.

3.2.2 BA for generating the sports training plan

Mathematically, generating the sports training plans based on existing sports activities can be defined as follows: Let a set of clusters $C = \{C_k\}$ for $k = 1, \ldots, n$ and an ordered list of D-tuples $(C_{\pi_1}, \ldots, C_{\pi_D})$, where n is the number of clusters and $\pi_j \in [1, n]$ for $j = 1, \ldots, D$ determines the cluster defined by j-th tuple and D is dimension of the problem denoting the length of the training cycle. Thus, the tuple is defined as $C_k = \{s_{k,1}, \ldots, s_{k,n_k}\}$ where $s_{k,l}$ for $l = 1, \ldots, n_k$ denotes a specific training session and n_k is a size of k-th cluster. In addition, vectors $\mathbf{x}_i = [x_{i,1}, \ldots, x_{i,D}]^T$ for $i = 1, \ldots, Np$ are given, where each element $x_{i,j} \in [0, 1]$ is mapped to the corresponding training session. The training session $s_{\pi_j,l}$ is obtained from the element $x_{i,j}$ by determining the index l, as follows

$$l = \lceil x_{i,j} \cdot n_{\pi_j} \rceil. \tag{5}$$

In fact, each element $s_{\pi_j,l}$ denotes the l-th training session in π_j-th cluster. As a result, the fitness function of the modified BA algorithm is expressed as follows

$$f(\mathbf{x}_i) = \sum_{j=1}^{D} w_i \cdot \text{TRIMP}(s_{\pi_j,l}), \tag{6}$$

where w_i denotes the weight of the corresponding cluster and $\text{TRIMP}(s_{\pi_j,l})$ denotes the TRIMP training load indicator of the corresponding training session calculated by Eq. (5).

Let us suppose, a training cycle of one week is given, where five days are devoted for the active training sessions (i.e., $D = 5$) and two days for resting. The distribution of clusters during the training cycle is defined as $C = \{C_1, C_2, C_3, C_2, C_1\}$. Consequently, each element of vector $\mathbf{x}_i$ can be mapped to a specific training session $x_{i,j} \mapsto s_{\pi_j,l}$. The weight values and distribution of clusters are presented in Table 3.

Table 3: Distribution of clusters and their influence on the fitness function.

Day	Cluster	Weight w_i	Cluster size	% max HP
Monday	-	-	-	-
Tuesday	1	0.1	27	30-50%
Wednesday	2	0.2	49	50-70%
Thursday	3	0.4	44	70-90%
Friday	-	-	-	-
Saturday	2	0.2	49	50-70%
Sunday	1	0.1	27	30-50%

As can be seen from Table 3, the distribution of the intensity of the training sessions follows predefined demands, because the training sessions from the low intensive cluster one are taken in the second and seventh training days, the more intensive cluster in the third and sixth training days and the high intensive cluster in the fourth training day (i.e., in the middle of the active training period). Two days in the training plan (i.e., Monday and Friday) are left for resting.

From the mathematical definition of the problem it follows that the modified BA algorithm differs from the original BA in the interpretation of the solution. Indeed, each element of solution vector $x_{i,j}$ is mapped to the corresponding j-th training session in i-th cluster that represents a basis for the evaluation of the TRIMP training load indicator.

The task of the modified BA is to find the maximum value of the fitness function, in other words, the $\max f(\mathbf{x})$ is searched for by the BA. Let us notice that the algorithm was implemented in Python programming language.

3.2.3 Visualization

The sports training courses proposed by the modified BA are visualized in this phase. Thus, the appropriate training course for each day of the week is selected and the figure is drawn from the corresponding GPX or TCX dataset using a Google Maps with the help of a visualizer [27]. In this manner, an athlete, as well as a potential trainer, obtain a feeling of how stressful the training session is without waiting for he/she to realize it during the training cycle. On the other hand, the athlete can allocate his/her strength properly along the entire training course.

4 Experiments

The goal of our experimental work was to show that the BA can be used as a generator of sports training plans based on existing courses. Thus, the quality of the generated sports training plans needs to be not worse than those planned by real trainers manually. In line with this, the archive of tracked TCX and GPX datasets obtained by the professional cyclists were taken into consideration during a period of one year. These datasets are clustered by *k*-means clustering into three groups according the TRIMP training load indicator. Then, the BA was applied for generating the sports training plans. Finally, courses proposed in the optimal sports training plan were visualized. This method is useful for each sports discipline where the training course can be tracked by the mobile sports trackers (i.e., running, biathlon, duathlon, triathlon, etc.).

During the experiments, the parameters of the BA were set as follows. The frequency was drawn from the Gaussian distribution in the interval $Q_i \in [0.0, 1.0]$, the loudness was fixed at $A_i^{(t)} = 0.7$, while the emission rate was $r_i^{(t)} = 0.2$. Let us notice that the population size was set to $Np = 70$ and algorithm can spend the maximum 10,000 function evaluations to terminate and that only one run of the BA was used for evaluation of the results.

The best result of the optimization with BA for generating the sports training plans obtained during the experimental work is illustrated Table 4. The total intensity load overcome by an athlete during this training cycle amounts to TRIMP=$124,758.56$.

As can be seen from Table 4, the more intensive training courses from each cluster were incorporated into the training plan, i.e., thus the maximum duration is pursued by the maximum average *HR*. As matter of fact, the value of the TRIMP training load indicator increases from the Tuesday to Thursday training sessions and decreases from the Thursday to Sunday training sessions. As said before, Monday and Friday remain for resting. Although it seems that the duration *t* has higher impact on the TRIMP value, also average *HR* is above the value 130 in the last three training sessions. This training plan was also confirmed by two real trainers for cycling.

Table 4: Generated sport training plan as proposed by the BA.

Day	Cluster	Training Session	Average *HR*	Duration *t*	TRIMP
Monday	-	-	-	-	-
Tuesday	1	9	130	124.04	16,125.20
Wednesday	2	2	128	184.18	23,575.04
Thursday	3	22	150	281.08	42,162.00
Friday	-	-	-	-	-
Saturday	2	27	137	189.15	25,913.55
Sunday	1	47	133	127.69	16,982.77

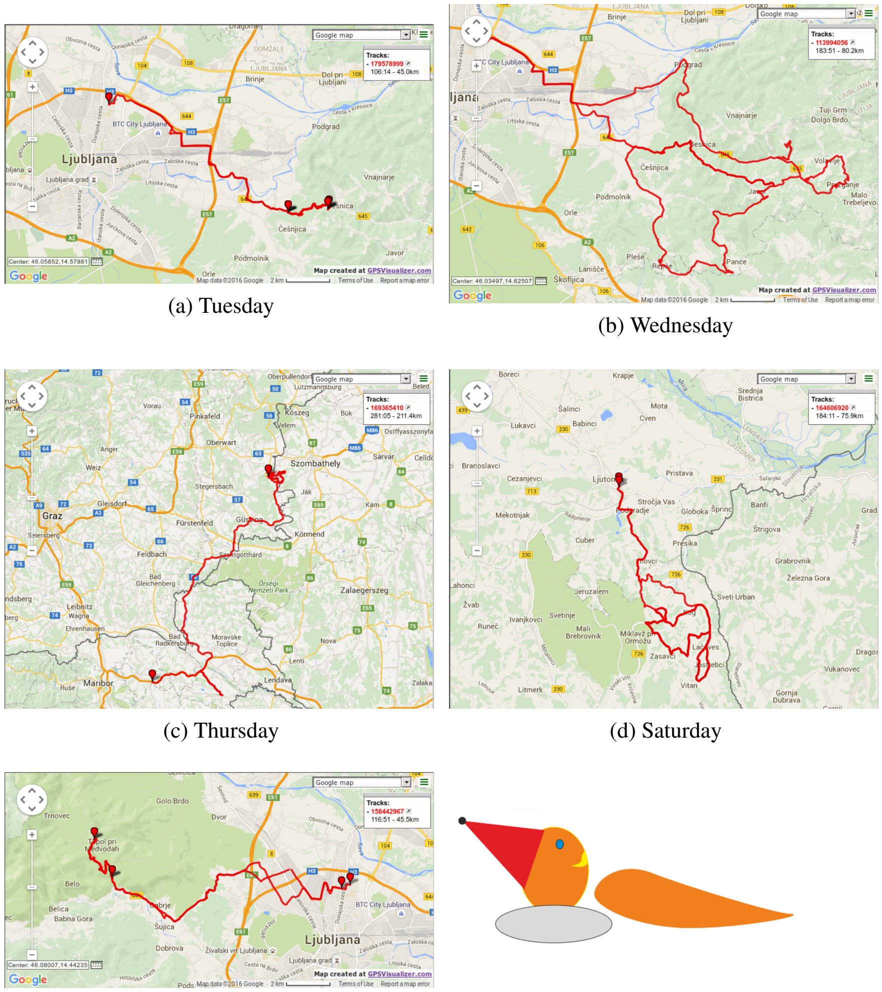
Visualization of the results from Table 4 is illustrated in Fig. 4, where specific maps are presented to an athlete in training to prepare themselves easily in mental preparation for the upcoming closures of training sessions in the next cycle. Training courses in these 5 maps are very vibrant, because they consists of small and medium hills, while there are also flat segments demanding the higher speeds.

Anyway, there a problem has been discovered after experiments due to some courses that are not generated within the same geographic area (e.g., Central Slovenia or Northeastern Slovenia). However, this is not a problem, when the competition (e.g., National Championship) is planned in another geographic region and athletes would like to try it before the competition takes place. Unfortunately, this mode of training is often unusual for realizing the normal training sessions. As a result, the generator must be upgraded in the future with some geographical filter where all activities laying outside of particular geographical region are removed from an observation.

5 Conclusion with future ideas

This Chapter presented an innovative solution for generating the sports training plans based on existing sports activities that expands the functionality of the artificial sports trainer. In particular, swarm intelligence algorithms were used to search for the best combination of training sessions within a cycle of duration one week. Indeed, the training sessions are clustered according to their TRIMP training load indicators into three clusters, from which the appropriate training sessions are included into the training plan such that obey the prescribed distribution of training intensities. Practical experiments showed that the proposed solution is very bright for use in such a domain.

In the future, there is still room for improvement and extension of this approach. Firstly, it would be good to take more training sessions into account. Additionally, the design of a prototype for generation of training plans based on existing sports activities in multisport disciplines would be a good way for future development. On the other hand, it might be very promising to study the influence of the other values extracted from GPX or TCX datasets besides duration and average heart rate, also,



(a) Tuesday (b) Wednesday

(c) Thursday (d) Saturday

(e) Sunday (f) Monday, Friday

Fig. 4: Visualization of sports training courses.

e.g. altitude, power meters, cadence, etc. Moreover, considering the other training types, like intervals, hills, power, should be outlined in more detail.

References

1. Arnold Baca, Peter Dabnichki, Mario Heller, and Philipp Kornfeind. Ubiquitous computing in sports: A review and analysis. *Journal of Sports Sciences*, 27(12):1335–1346, 2009.

2. Marc Bächlin, Kilian Förster, and Gerhard Tröster. Swimmaster: a wearable assistant for swimmer. In *Proceedings of the 11th international conference on Ubiquitous computing*, pages 215–224. ACM, 2009.
3. EW Banister. Modeling elite athletic performance. *Physiological testing of elite athletes*, pages 403–424, 1991.
4. EW Banister, JB Carter, and PC Zarkadas. Training theory and taper: validation in triathlon athletes. *European journal of applied physiology and occupational physiology*, 79(2):182–191, 1999.
5. Charles Darwin. On the origins of species by means of natural selection. *London: Murray*, page 247, 1859.
6. Dipankar DasGupta. *An overview of artificial immune systems and their applications*. Springer, 1999.
7. Iztok Fister, Samo Rauter, Xin-She Yang, Karin Ljubič, and Iztok Fister Jr. Planning the sports training sessions with the bat algorithm. *Neurocomputing*, 149:993–1002, 2015.
8. Iztok Fister Jr., Iztok Fister, Dušan Fister, Karin Ljubič, Yan Zhuang, and Simon Fong. Towards automatic food prediction during endurance sport competitions. In *Soft Computing and Machine Intelligence (ISCMI), 2014 International Conference on*, pages 6–10. IEEE, 2014.
9. Iztok Fister Jr, Goran Hrovat, Samo Rauter, and Iztok Fister. Am i overtraining? a novel data mining approach for avoiding overtraining. In *Computer Science Research Conference*, pages 47–52, 2014.
10. Iztok Fister Jr., Karin Ljubič, Ponnuthurai Nagaratnam Suganthan, Matjaž Perc, and Iztok Fister. Computational intelligence in sports: Challenges and opportunities within a new research domain. *Applied Mathematics and Computation*, 262:178–186, 2015.
11. Iztok. Fister Jr, Samo Rauter, Karin Ljubič Fister, Dušan Fister, and Iztok Fister. Planning fitness training sessions using the bat algorithm. In *15th conference ITAT 2015 (CEUR workshop proceedings, ISSN 1613-0073, Vol. 1422)*, pages 121–126, 2015.
12. John A Hartigan and Manchek A Wong. Algorithm as 136: A k-means clustering algorithm. *Applied statistics*, pages 100–108, 1979.
13. Aboul Ella Hassanien and Eid Alamry. *Swarm Intelligence: Principles, Advances, and Applications*. CRC Press, 2015.
14. Mohamad H Hassoun. *Fundamentals of artificial neural networks*. MIT press, 1995.
15. Jonathan Hey and Scott Carter. Pervasive computing in sports training. *Pervasive Computing, IEEE*, 4(3):54, 2005.
16. James Kennedy, James F Kennedy, Russell C Eberhart, and Yuhui Shi. *Swarm intelligence*. Morgan Kaufmann, 2001.
17. Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.
18. Tian-Syung Lan, Pin-Chang Chen, Ying-Jia Chang, and Wen-Hsu Chiu. An investigation of the relationship between performance of physical fitness and swimming ability based on the artificial neural network. In *Innovation in Design, Communication and Engineering: Proceedings of the 2014 3rd International Conference on Innovation, Communication and Engineering (ICICE 2014), Guiyang, Guizhou, PR China, October 17-22, 2014*, page 181. CRC Press, 2015.
19. Florian Michahelles and Bernt Schiele. Sensing and monitoring professional skiers. *Pervasive Computing, IEEE*, 4(3):40–45, 2005.
20. Hristo Novatchkov and Arnold Baca. Machine learning methods for the automatic evaluation of exercises on sensor-equipped weight training machines. *Procedia Engineering*, 34:562–567, 2012.
21. Hristo Novatchkov and Arnold Baca. Artificial intelligence in sports on the example of weight training. *Journal of sports science & medicine*, 12(1):27, 2013.
22. Jürgen Perl and Arnold Baca. Application of neural networks to analyze performance in sports. In *Proceedings of the 8th annual congress of the European College of Sport Science, S*, volume 342, 2003.
23. Samo Rauter and Mojca Doupona Topič. Perspectives of the sport-oriented public in slovenia on extreme sports. *Kineziologija*, 43(1):82–90, 2011.

24. Samo Rauter and Mojca Doupona Topič. Runners as sport tourists: The experience and travel behaviours of ljubljana marathon participants. *Collegium antropologicum*, 38(3):909–915, 2014.
25. Samo Rauter, Iztok Fister, and Iztok Fister Jr. How to deal with sports activity datasets for data mining and analysis: Some tips and future challenges. *International Journal of Advanced Pervasive and Ubiquitous Computing (IJAPUC)*, 7(2):27–37, 2015.
26. Timothy J Ross. *Fuzzy logic with engineering applications*. John Wiley & Sons, 2009.
27. Adam Schneider. Gps visualizer. *http://www. gpsvisualizer. com*, 2009.
28. Dan Simon. *Evolutionary optimization algorithms*. John Wiley & Sons, 2013.
29. Xin-She Yang. A new metaheuristic bat-inspired algorithm. In *Nature inspired cooperative strategies for optimization (NICSO 2010)*, pages 65–74. Springer, 2010.