



Erstellung ein Stundenplan mit Genetic Algorithmus

Suat Köroğlu

e190503043@stud.tau.edu.com

INFORMATIK

Erstellung ein Stundenplan mit Genetic Algorithmus

Suat Koroğlu

1. Einführung

- Stundenplanung kann mithilfe eines genetischen Algorithmus erstellt werden.
- Nachdem die erforderlichen Informationen eingegeben wurden, wird der Algorithmus ausgeführt. Sowohl für Lehrer als auch für Schüler wird eine angemessene Planung durchgeführt

2. Methoden

- Zuerst Permutation erstellen.
- Ausführung von Fitnessfunktion
- Ausführung von Crossover- und Mutationsalgorithmen
- Optimierung des Algorithmus

3. Ergebnisse

- Unter Verwendung eines genetischen Algorithmus mit geeigneten Funktionen zeigen wir, dass ein ein Stundenplanung Erstellung möglich ist(Fig 2, Fig 3)

4. Diskussion

- Die Erhöhung der Anzahl von Unterrichtsstunden, Lehrern und Klassen wird die Arbeitszeit des Algorithmus erhöhen.
- We taken a major step towards proving the viability of this sensing mechanism

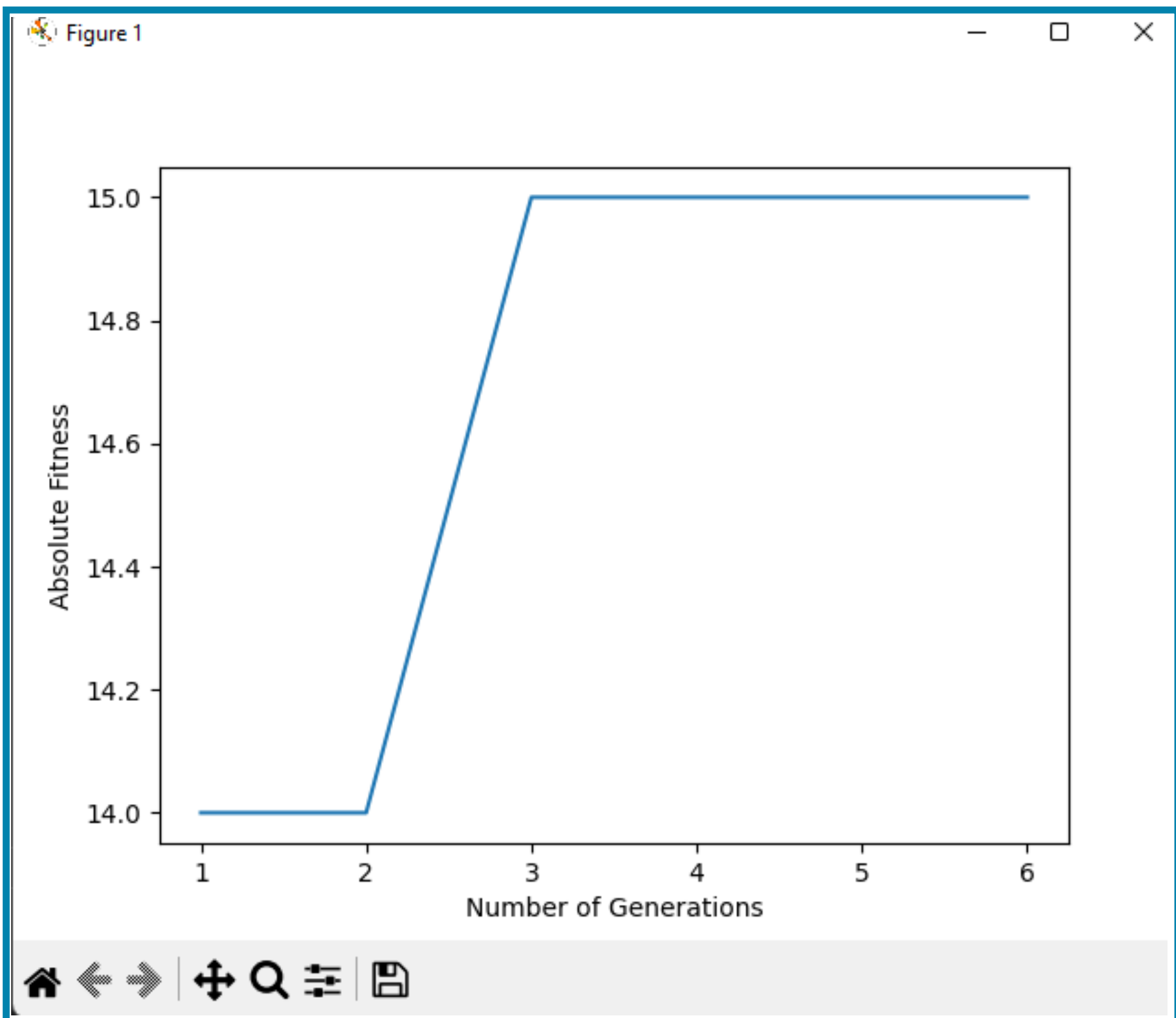


Figure 1 Arbeitsdiagramm des Genetischen Algorithmus

	A	B	C	D	E
1		grade_1	grade_3	grade_5	grade_7
2	Period: 1	INF107	INF205	INF506	INF506
3	Period: 2	MAT103	INF203	INF714	INF714
4	Period: 3	INF101	INF209	INF701	INF517
5	Period: 4	INF103	INF201	INF523	INF523

Figure 2 Ergebnismachweis Studienplan mit genetischem Algorithmus

	A	B	C	D	E	F	G
1		bircu_class	ruk_class	dyo_class	nan_class	mel_class	re_class
2	Period: 1	INF103	INF107	DEU121	INF205	BAU103	INF506
3	Period: 2	MAT103	INF211	ENG201	INF517	INF303	INF714
4	Period: 3	INF101	INF209	TUR001	INF701	INF203	ETE103
5	Period: 4	INF211	INF103	ENG101	INF201	INF523	INF211

Figure 3 Ergebnismachweis Studienplan mit genetischem Algorithmus

Codes und Ergebnistabelle des für die Unterrichtsplanung geschriebenen **Genetischen Algorithmus**.

```
#Erstellung von alle Permutationen der Stundenpläne der Schüler
> def create_all_permutations(students):...

#erstellt eine Anfangspopulation von n Stundenplänen, indem die Stundenpläne jedes Lehrers n-mal gemischt werden
> def initial_population(teachers_list, population_size):...

#Gehe alle möglichen Schülerpläne durch und überprüfe, welcher der beste ist, der seine Punktzahl zurückgibt
> def current_offspring_fitness(students_list, teachers_lists, fitness_weights):...
> def crossover (offspring, offspring_fitness):...

> def hall_of_fame(offspring, offspring_fitness, top_students):...

> def mutation(offspring, mutation_chance, n):...

> def absolute_fitness(top_dog, students_list, fitnesses, index):...

> def genetic_algorithm_function(generations, students, teachers, mutation_chance, number_of_offspring, print_results, graph_results, fitness_weights):...

genetic_algorithm_function(6, students_, teachers_, 0.4, 20, False, True, student_weights)
```

```
INF101 ED-A-2-3
MAT103 ED-B-2-1
INF103 ED-A-2-2
INF211 ED-A-3-5
INF107 ED-B-1-1
INF209 ED-B-2-5
DEU121 ED-B-2-4
ENG101 ED-A-2-4
TUR001 ED-A-1-3
ENG201 ED-B-1-4
INF201 ED-B-3-1
INF205 ED-A-2-1
INF517 ED-A-2-5
INF701 ED-B-3-5
INF203 ED-B-1-3
INF523 ED-A-1-4
INF303 ED-B-3-4
BAU103 ED-A-3-4
INF506 ED-A-3-3
INF714 ED-B-3-2
ETE103 ED-A-3-1
6 Generations and 0.8431956768035889 seconds needed
```

Erstellung ein Stundenplan mit Genetic Algorithmus

Liste alle Eigenschaften des Problems

Suat Koroğlu 190503043

Vorlesungen

INF101, MAT103, INF103, INF211, INF107, INF209, DEU121, ENG101, TUR001, ENG201, INF201, INF205, INF517, INF701, INF203, INF523, INF303, BAU103, INF506, INF714, ETE103

Professoren

- Canan Yıldız
- Faruk Bağcı
- Burcu Yıldız
- Emre Işık
- Emel Maden Yılmaz
- YDYO

Räume

- ED-A-1-1 bis ED-A-1-5
- ED-A-2-1 bis ED-A-2-5
- ED-A-3-1 bis ED-A-3-5
- ED-B-1-1 bis ED-B-1-5
- ED-B-2-1 bis ED-B-2-5
- ED-B-3-1 bis ED-B-3-5
- TOTAL: 30 KLASSE

Jahrgänge

- 1. Semester
- 3. Semester
- 7. Semester