AI phase 1 documentation

IDEA DESCRIPTION

The project aims to solve a faculty timetable scheduling problem using differential evolution.

The problem discussed is an NP-hard problem of generating a valid and highly optimal timetable for a faculty.

By valid we mean that there are no conflicts in the timetable, i.e. no two classes are in the same classroom at the same time, nor can a professor hold two classes at the same time, etc. Next, we describe the specific timetable format we will be analyzing. Classes are defined in the following fashion:

* Courses that is being taught
* The professor conducting the class
* All the student groups listening to the class together at the same time
* Length (1 to 4 hours) We assume that all classrooms are of the same size and have the required capacity. Valid hours for holding classes are from 9am to 6pm.

The task at hand is to assign a time and classroom for each of the classes given in the mentioned format.

FUNCTIONS

1. Assign classes to appropriate classrooms.
2. Assign proper time frame for each class.
3. Assign lecturers to their classes.
4. Ensure no classes are assigned more students than their capacity.
5. Ensure the most optimal schedule is generated based on the input data.

REAL WORLD APPLICATION

This is an application in South Africa regarding scheduling of nurses using a differential evolution approach.

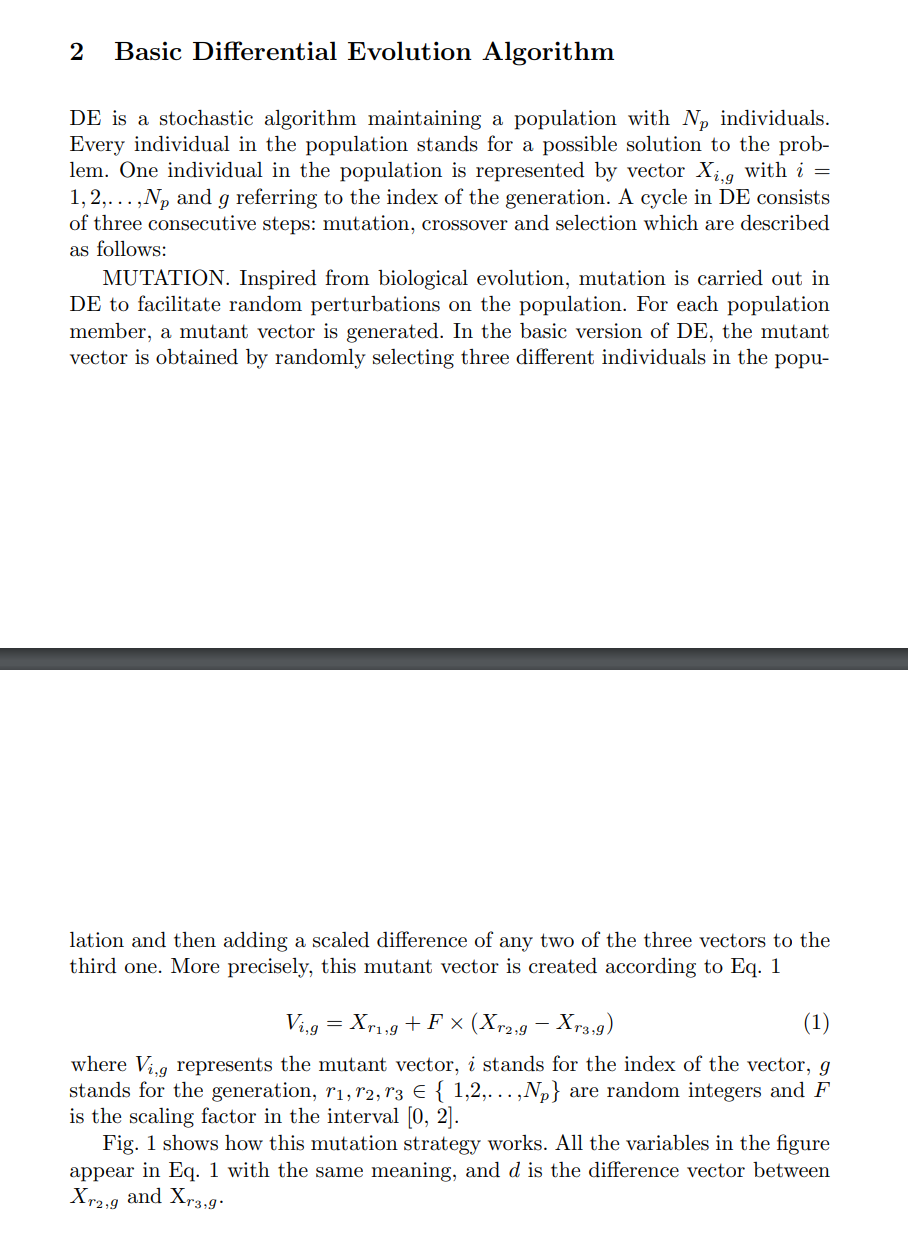
🡺<http://www.scielo.org.za/scielo.php?script=sci_arttext&pid=S2224-78902012000300008>

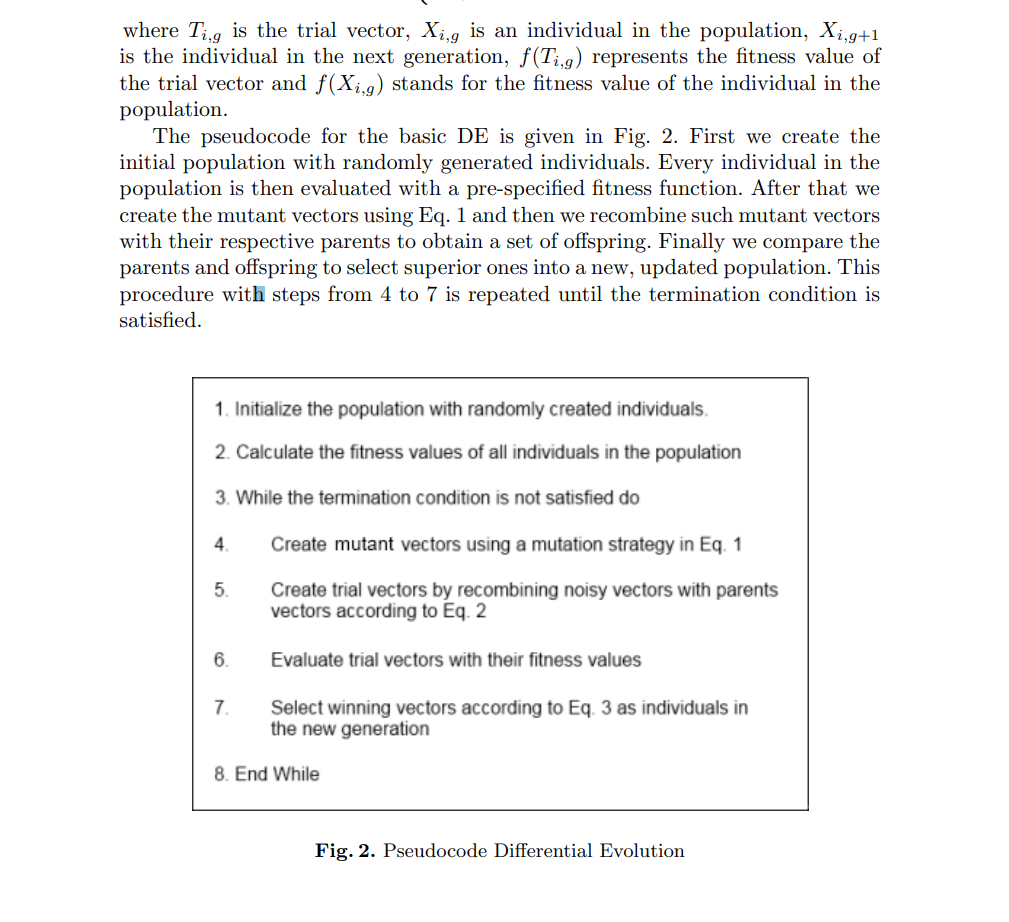
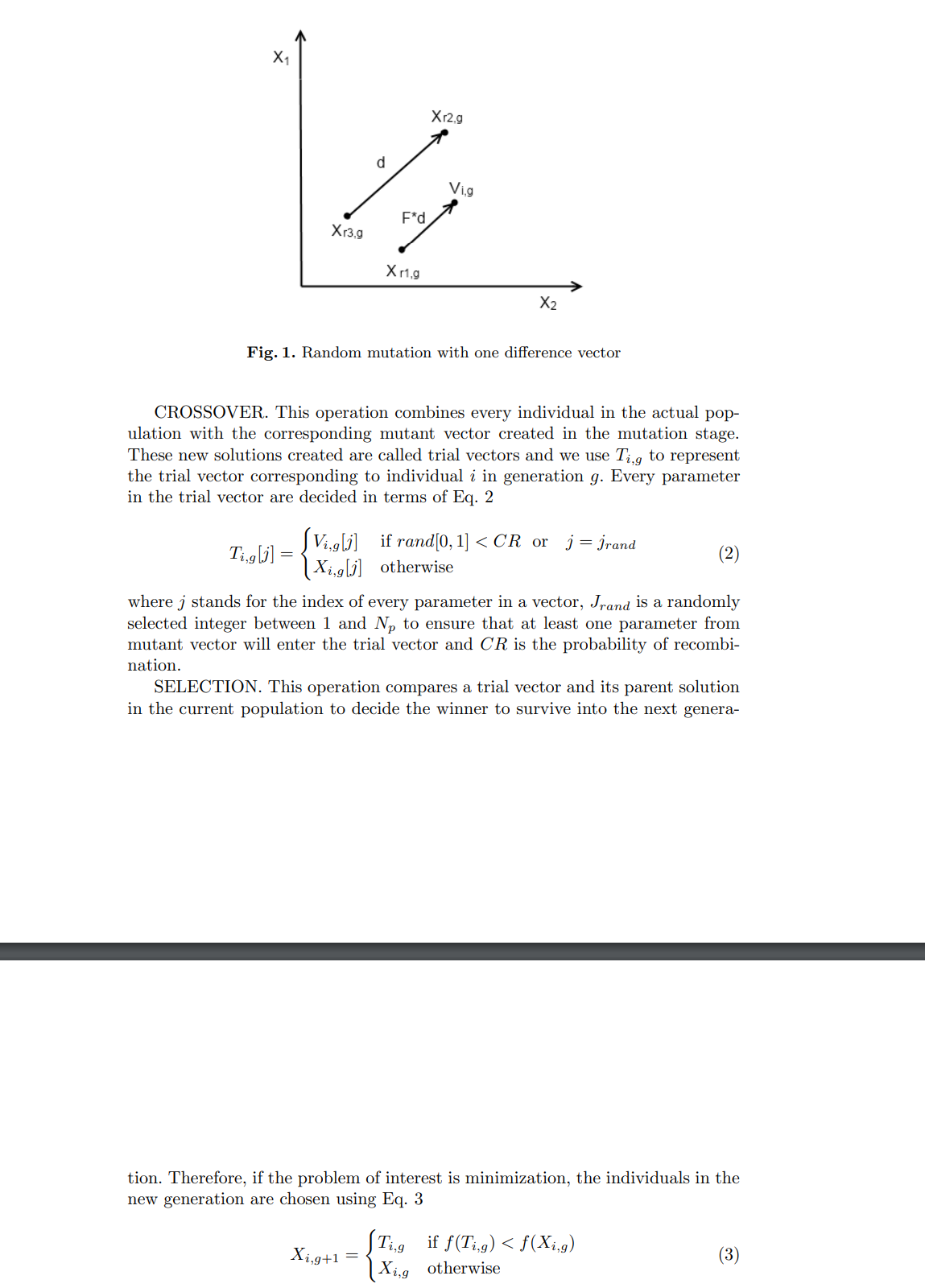
This is an application on the scheduling of generator maintenance in Nigeria.

* <https://core.ac.uk/reader/229178154>



ACADEMIC LITERATURE RIVEIW



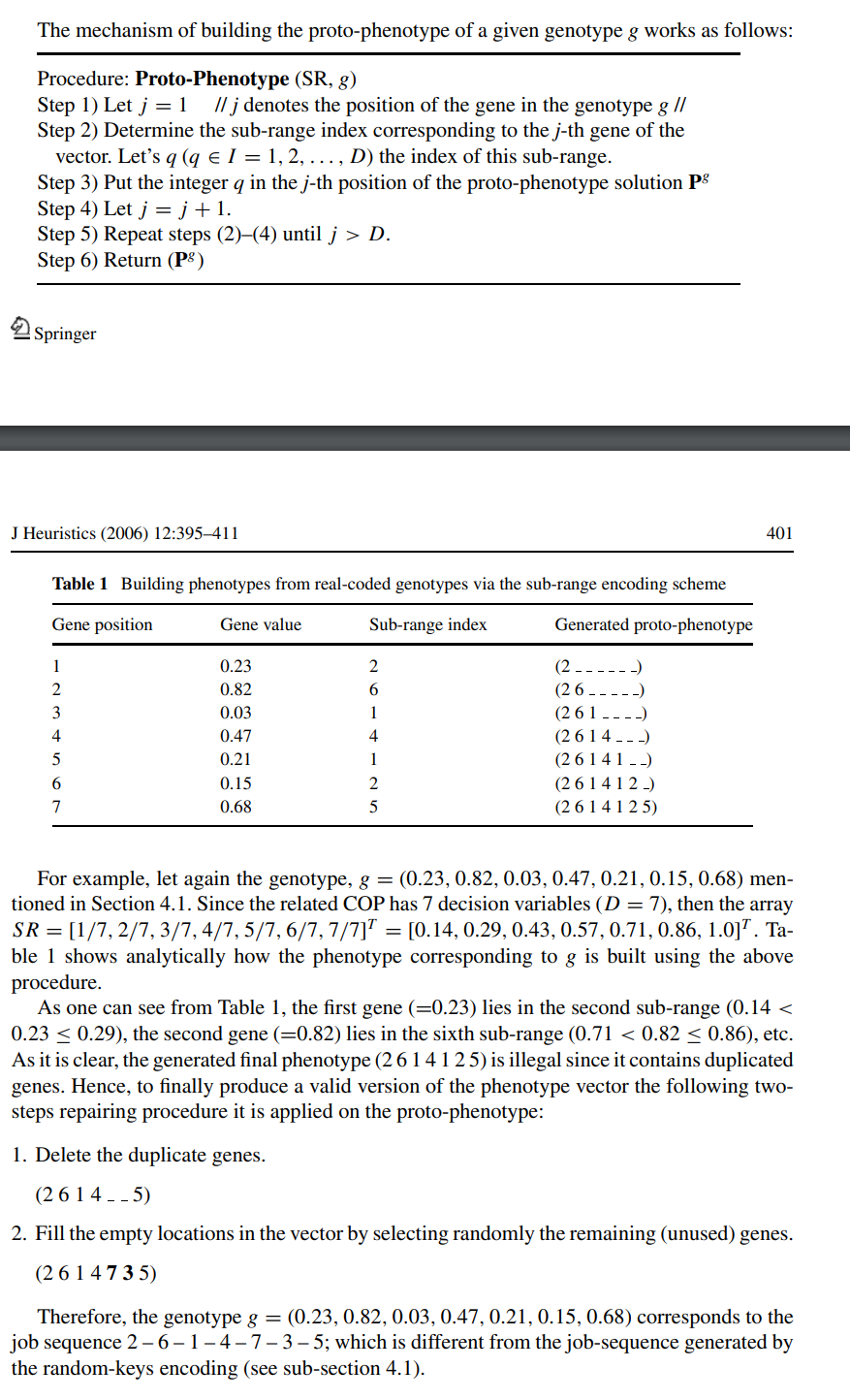


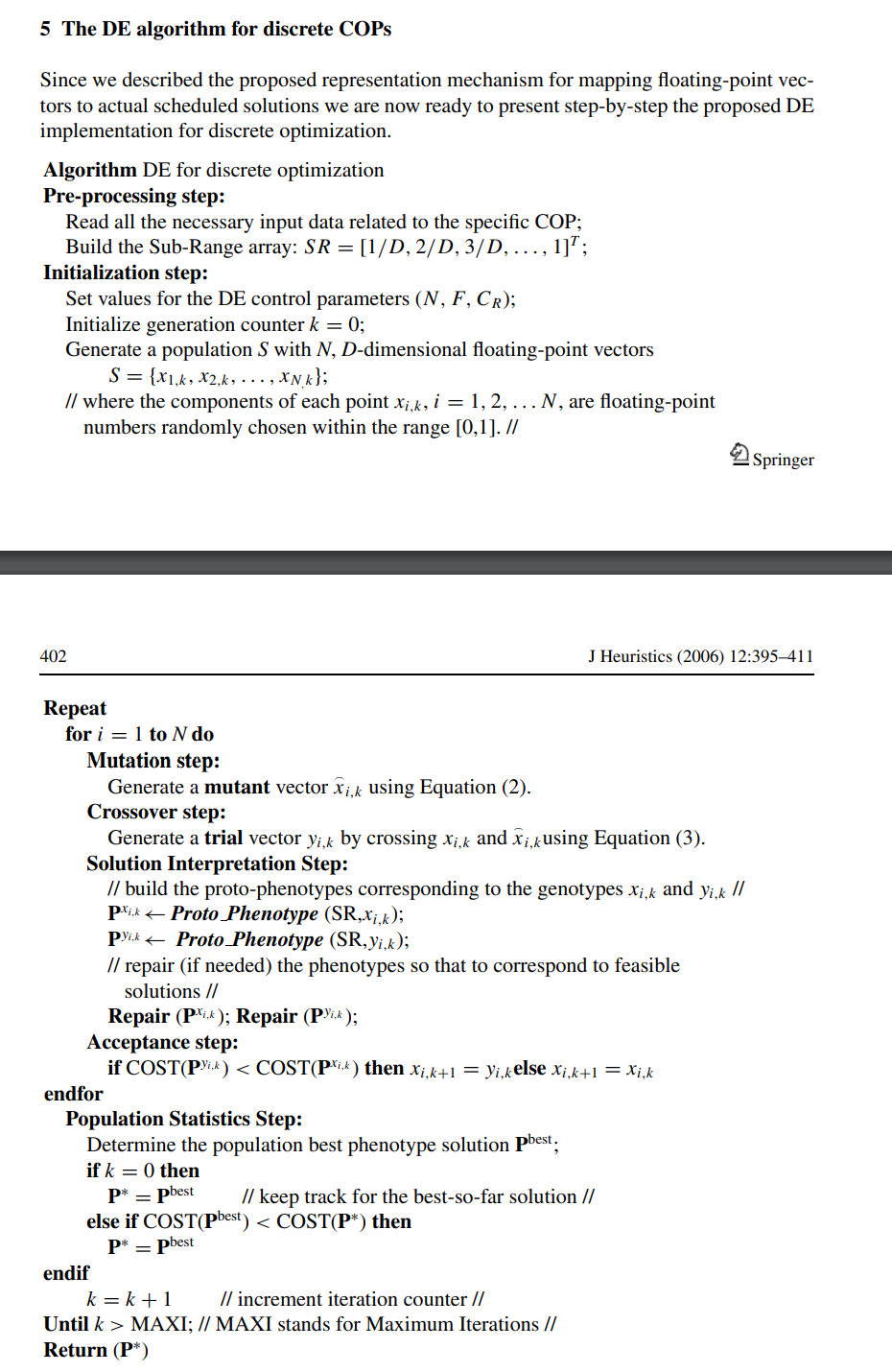
This discusses a basic idea of a differential evolution

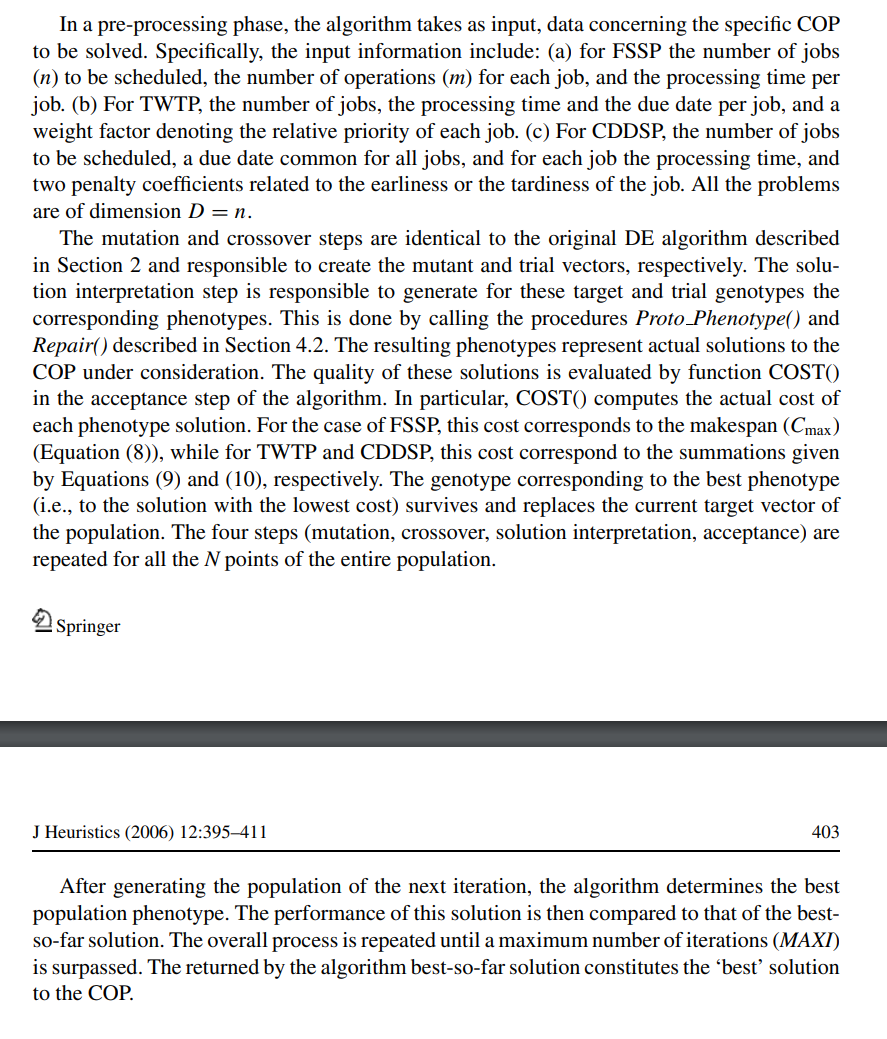
Strategy.

From the following paper









This explains a differential evolution approach using a proto phenotype that is being generated.

From the following paper



APPROACH/ALGORITHIMS

First we start by defining our constraints

1. Resources must not overlap in time:
   * No professor can hold two classes at the same time
   * No student group can attend two classes at the same time
   * No classroom can host to classes at the same time
   * Note: the term "same time" does not only mean the starting time of a class, but what also must be taken into account is the length of a class. If a resources is taken at time T1 and the class lasts for t, then the resource can be taken again only at time T2.

The constraints must be met (hard constraints)

2. Additional possible soft constraints

* Minimize total idleness for each group (pauses between classes)
* Minimize total idleness for each professor (pauses between classes)
* Provide one free hour a week with no classes, for a professors union meeting
* Minimize daily load for professors and groups (less than 6 hours of class a day)

Classes:

1. Data (contains data and some preformatting functions)
2. Schedule (to be populated with the data of each generation by the program)
3. Population
4. Course
5. Instructor
6. Classroom
7. ClassTime
8. Department
9. Class

Algorithms:

* Differential algorithm:

1. Data formatting and discerning algorithm
2. Cross over algorithm
3. Mutation algorithm
4. Fitness calculation algorithm
5. An algorithm to select the optimum solution(Solution Selection algorithm)