Development of a Heuristic Algorithm to Automate and Improve the Scheduling Process for Tennis Clubs

A Practical Operations Research Thesis with a Case Study at a Padel Club

MAX HEDERSTEDT LÖVENHEIM

JOHAN KRYLSTEDT
Development of a Heuristic Algorithm to Automate and Improve the Scheduling Process for Tennis Clubs

A Practical Operations Research Thesis with a Case Study at a Padel Club

MAX HEDERSTEDT LÖVENHEIM

JOHAN KRYLSTEDT
Abstract

The scheduling of a tennis school’s operation at a general tennis club is today done manually. This is a time consuming task, usually taking 2-4 weeks for the person responsible. This thesis aims to mathematically formulate the tennis school scheduling problem (TSSP), and to develop a computer based solution procedure to automate the scheduling process. The mathematical formulation of the problem has the structure of an optimization problem and the solution procedure that is developed is a heuristic algorithm for the specific purpose of this problem. Furthermore, a heuristic local search algorithm to improve the initial solution is developed, and the model is then tested on real data in a case study at a padel club in Stockholm. Furthermore, the result from the solution procedure is compared with the manually created schedule by the club on some key parameters from the mathematical model. The computerized solution procedure outperformed the manual scheduling procedure in terms of execution time and most of the times in the objective value of the generated schedule. In terms of the mathematical formulation neither the manually created schedule nor the schedules created by the algorithm gives feasible solutions to the problem. However, the padel club deems the schedules usable and will continue with the ongoing process to implement the model.

Keywords

Tennis Scheduling, Sports Scheduling, Operations Research, Heuristics, Algorithms
Utveckling av en heuristisk algoritm för att automatisera och förbättra schemaläggningsprocessen för tennisklubbar

June 20, 2019

Sammanfattning


Nyckelord

Schemaläggningsprocessen, Tennis, Systemteknik, Sökalgoritmer
Acknowledgements

We want to thank Johan Hjelmgren and Samuel Schwieler at Court1 Academy for participating in this project and providing us with the necessary data. We also want to thank Marianne Marescotti for her valuable insights in the world of a tennis organization. Lastly we want to thank our supervisor Per Enqvist for providing us with help when formulating the problem and guidance throughout the work to keep us on track.
Contents

1 Introduction ......................................................... 1
  1.1 Background ................................................. 1
  1.2 Problematization ......................................... 2
    1.2.1 Case Study ........................................ 3
  1.3 Purpose and Delimitations ......................... 3

2 Methodology ...................................................... 4
  2.1 Problem definition and gathering of data .......... 4
  2.2 Formulating a mathematical model ................. 4
  2.3 Develop a computer-based solution procedure .... 5
  2.4 Test and refine the model ......................... 5

3 Problem Definition and Data Gathering .................. 7
  3.1 Interviews ............................................. 7
  3.2 Related Works ......................................... 8
  3.3 Theory ................................................. 8
    3.3.1 Heuristic Algorithm Development ............. 8
    3.3.2 Heuristic Search Methods ..................... 9
  3.4 Requirement Specification .......................... 10
    3.4.1 Input Data ..................................... 10
    3.4.2 Coach Specific Requirements ................ 10
    3.4.3 Player Specific Requirements ............... 11
    3.4.4 Group Specific Requirements ................ 11

4 Mathematical Modelling ......................................... 12
  4.1 Further Simplifying Equations and Functions .... 12
  4.2 Coach Scheduling Problem ......................... 13
  4.3 Player Scheduling Problem ......................... 14

5 Computer Based Solution Procedure ...................... 16
  5.1 Additional Functions .................................. 16
  5.2 Schedule Coaches .................................... 17
    5.2.1 Schedule Coaches Pseudocode ............... 17
  5.3 Schedule Players ..................................... 20
    5.3.1 Schedule Players Pseudocode ............... 21
  5.4 Improvement Algorithm .............................. 23
    5.4.1 Improvement Algorithm Pseudocode .......... 26

6 Padel Club Case ................................................ 27
  6.1 Input Data ............................................. 27
  6.2 Results ................................................ 28
    6.2.1 Manually Produced Schedule by Court1 ...... 28
    6.2.2 Automated Schedule ........................... 28
    6.2.3 Improvement Algorithm ....................... 29
<table>
<thead>
<tr>
<th>Table 1: Table of Notations</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General</strong></td>
</tr>
<tr>
<td>$h \in H \triangleq {0, 1, 2, ..., 23}$</td>
</tr>
</tbody>
</table>
| $a_{h,d,court} \triangleq \begin{cases} 
1, & \text{if court } court \text{ is available to be scheduled at hour } h \text{ and day } d \\
0, & \text{otherwise} 
\end{cases}$ |
| **Player Categories**       |
| $cat \in Categories$, $Categories \triangleq \{Category_1, Category_2, ..., Category_\beta\}$ are the different player categories and $\beta$ is the number of categories |
| $\text{maxh}_{cat} \triangleq$ the maximum amount of hours a player of $cat$ can get |
| $\text{guaranteedh}_{cat} \triangleq$ the guaranteed amount of hours a player of $cat$ will get |
| $\text{maxhinarow}_{cat} \triangleq$ the maximum amount of hours a player of $cat$ can get in a row |
| $\text{maxping}_{cat} \triangleq$ the maximum number of players in a group of $cat$ |
| $\text{agediff}_{cat} \triangleq$ the maximum difference in age players in the same group of $cat$ can have |
| $\text{leveldiff}_{cat} \triangleq$ the maximum difference in level players in the same group of $cat$ can have |
| $\text{Cats}_{h,d,court} \subseteq Categories$, the categories allowed to play at hour $h$, day $d$ and court $court$ |
| **Coaches**                 |
| $c \in Coaches$, where $Coaches \triangleq \{Coach_1, Coach_2, ..., Coach_m\}$ is the prioritized set of coaches, where $m$ is the number of coaches |
| $l_{c} \in L$, the maximum player level coach $c$ has competence to coach |
| $\text{Cats}_{c} \subseteq Categories$, the set of categories coach $c$ has competence to coach |
| $\text{minh}_{c} \triangleq$ the minimum amount of hours a week coach $c$ will work |
| $\text{maxh}_{c} \triangleq$ the maximum amount hours a week coach $c$ can work |
| $\text{desiredh}_{c} \triangleq$ the desired amount of hours a week coach $c$ will work |
| $\text{minshift}_{c} \triangleq$ the minimum amount hours each work shift coach $c$ must have |
| $\text{employment}_{c} \triangleq$ the type of employment a coach has |
| $\text{priority}_{c} \in \{1, ..., m\}$, the prioritization order in which the the coaches shall be scheduled |
| $c_{h,d,court} \triangleq$ the coach $c$, if any, that is scheduled on court $court$, hour $h$ and day $d$ |
\[a_{c,h,d,\text{court}} = \begin{cases} 1, & \text{if coach } c \text{ is available to work hour } h \text{ and day } d \text{ at court} \\ 0, & \text{otherwise} \end{cases}\]

\[\text{cancat}_{c,\text{cat}} = \begin{cases} 1, & \text{if } \text{cat} \in \text{Cats}_c \\ 0, & \text{otherwise} \end{cases}\]

\[\text{cmatchp}_{c,p} = \begin{cases} 1, & \text{if } l_c \geq l_p \text{ and } \text{can} \text{cat}_{c,\text{cat},p} = 1 \\ 0, & \text{otherwise} \end{cases}\]

\[\text{notfullybooked}_{c,h,d,\text{court}} = \begin{cases} 1, & \text{if booked lesson hours for } c < \text{maxh}_c \\ 0, & \text{otherwise.} \end{cases}\]

\[\text{freeperiod}_c = \begin{cases} 1, & \text{if coach } c \text{ can have a free period in between work shifts} \\ 0, & \text{otherwise} \end{cases}\]

Players

\[p \in \text{Players}, \text{ where } \text{Players} \triangleq \{\text{Player}_1, \text{Player}_2, \ldots, \text{Player}_n\} \text{ is the prioritized set of players, where } n \text{ is the number of players} \]

\[l_p \in \text{L}, \text{ the level of player } p \]

\[\text{age}_p \triangleq \text{player age} \]

\[\text{cat}_p \triangleq \text{player category} \]

\[\text{prio}_p \in \{1,2,\ldots,n\}, \text{ the order in which player } p \text{ is prioritized} \]

\[w_p \triangleq 1/\text{prio}_p, \text{ a weight to keep track on how valuable it is to schedule player } p. \]

\[\text{desiredh}_p \triangleq \text{the desired amount of hours a week player } p \text{ would like to play} \]

\[\text{guaranteedh}_p \triangleq \text{the guaranteed amount of hours player } p \text{ will get. Player specific.} \]

\[\text{maxhinarow}_p \triangleq \text{the maximum amount of hours player } p \text{ can get in a row.} \]

\[\text{previous}_p = \begin{cases} 1, & \text{if player } p \text{ has been in the school the semester before} \\ 0, & \text{otherwise} \end{cases}\]

\[a_{p,h,d,\text{court}} = \begin{cases} 1, & \text{if player } p \text{ is available to play } (h,d) \text{ at court} \\ 0, & \text{otherwise} \end{cases}\]

\[\text{allowedcat}_{p,h,d,\text{court}} = \begin{cases} 1, & \text{if } \text{cat}_p \in \text{Cats}_{h,d,\text{court}} \\ 0, & \text{otherwise} \end{cases}\]

\[\text{twoinarow}_p = \begin{cases} 1, & \text{if player } p_j \text{ is available to play two hours in a row} \\ 0, & \text{otherwise} \end{cases}\]

\[\text{twoinaday}_p = \begin{cases} 1, & \text{if player } p \text{ is available to play two hours the same day} \\ 0, & \text{otherwise} \end{cases}\]

Groups

\[g \in \text{Groups}, \text{ where } \text{Groups} \triangleq \{\text{Group}_1, \text{Group}_2, \ldots, \text{Group}_k\}, \text{ where } k \text{ is the number of groups} \]

\[\text{players}_g \triangleq \text{set of players in } g \]

\[\text{cat}_g \triangleq \text{the category of the players in group } g \]

\[\text{pmatchg}_{p,g} = \begin{cases} 1, & \text{if } |\text{level}_p - \text{level}_{p_g}| < \text{leveldiff}_{\text{cat}_g} \text{ and} \\ |\text{age}_p - \text{age}_{p_g}| < \text{agediff}_{\text{cat}_g}, \text{ and } \text{cat}_g = \text{cat}_p \\ \forall p_g \text{ where } p_g \in \text{players}_g \\ 0, & \text{otherwise} \end{cases}\]
\[ p_{1matchp2_{p1,p2}} = \begin{cases} 
1, & \text{if } |\text{ level }_{p1} - \text{ level }_{p2}| < \text{ leveldiff}_{cat_{p1}} \text{ and} \\
& |\text{ age }_{p1} - \text{ age }_{p2}| < \text{ agediff}_{cat_{p1}} \text{ and } \text{ cat }_{p1} = \text{ cat }_{p2} \\
0, & \text{otherwise} 
\end{cases} \]

\[ p_{ingp,g} = \begin{cases} 
1, & \text{if player } p \text{ is in group } g \\
0, & \text{otherwise} 
\end{cases} \]

\[ c_{ingc,g} = \begin{cases} 
1, & \text{if coach } c \text{ is assigned group } g \\
0, & \text{otherwise} 
\end{cases} \]

**Decision Variables**

\[ B_{c,h,d,court} = \begin{cases} 
1, & \text{if coach } c \text{ is scheduled at hour } h, \text{ day } d \text{ and court} \\
0, & \text{otherwise} 
\end{cases} \]

\[ B_{p,h,d,court} = \begin{cases} 
1, & \text{if player } p \text{ is scheduled at hour } h, \text{ day } d \text{ and court} \\
0, & \text{otherwise} 
\end{cases} \]

\[ B_{g,h,d,court} = \begin{cases} 
1, & \text{if group } g \text{ is scheduled at hour } h, \text{ day } d \text{ and court} \\
0, & \text{otherwise} 
\end{cases} \]
1 Introduction

This thesis describes the process to automate a general tennis club’s way of constructing groups and a schedule for their tennis classes. The thesis takes an operations research approach to the problem and handles four areas of the development of the automation: Gathering of the relevant data to define the problem, formulating the problem mathematically, creating a computer-based procedure to derive solutions from the model and testing the procedure to analyze the results.

From an operations research point of view, the organization of a tennis club is interesting to analyze since there are different coaches that needs to be allocated, groups to construct evenly and to schedule these groups and coaches in a satisfactory manner. Hence, a great number of parameters needs to be taken into account when creating such a schedule and an operations research approach might be the way to streamline this task.

1.1 Background

Tennis clubs are more often than not organized with two main uses of their courts, their own organized practices, i.e. their tennis school, and most of the hours not used by the tennis school are bookable by members and/or the public. On an ordinary weekday for a tennis school there are classes between 07.00-09.00 and 15.00-21.00. Between 07.00-09.00 the students are adults or competitive juniors practising before work/school, then from 15, the youngest juniors start their practise followed by older juniors, competitive players and later from 19.00-21.00 it is adult students again. On Saturdays the classes starts at 09.00 for kids under 7 years old and continues until 12.00, then from 12.00-16.00 it is classes for juniors. There are no classes on Sundays. (Marescotti, 2018)

Consider the example of a club with 4 courts, 6 coaches and with the categories of players as "adults", "competition", "juniors", "kids". Then an example of a weekly schedule might look as in Figure 1.

<table>
<thead>
<tr>
<th>Time</th>
<th>Court 1</th>
<th>Court 2</th>
<th>Court 3</th>
<th>Court 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>07:00</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
</tr>
<tr>
<td>08:00</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
</tr>
<tr>
<td>09:00</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
</tr>
<tr>
<td>10:00</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
</tr>
<tr>
<td>11:00</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
</tr>
<tr>
<td>12:00</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
</tr>
<tr>
<td>13:00</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
</tr>
<tr>
<td>14:00</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
<td>Adults/Competition</td>
</tr>
<tr>
<td>15:00</td>
<td>Competition/Juniors</td>
<td>Competition/Juniors</td>
<td>Competition/Juniors</td>
<td>Competition/Juniors</td>
</tr>
<tr>
<td>16:00</td>
<td>Competition/Juniors</td>
<td>Competition/Juniors</td>
<td>Competition/Juniors</td>
<td>Competition/Juniors</td>
</tr>
<tr>
<td>17:00</td>
<td>Competition/Juniors</td>
<td>Competition/Juniors</td>
<td>Competition/Juniors</td>
<td>Competition/Juniors</td>
</tr>
<tr>
<td>18:00</td>
<td>Competition/Juniors</td>
<td>Competition/Juniors</td>
<td>Competition/Juniors</td>
<td>Competition/Juniors</td>
</tr>
<tr>
<td>19:00</td>
<td>Adults</td>
<td>Adults</td>
<td>Adults</td>
<td>Adults</td>
</tr>
<tr>
<td>20:00</td>
<td>Adults</td>
<td>Adults</td>
<td>Adults</td>
<td>Adults</td>
</tr>
<tr>
<td>21:00</td>
<td>Adults</td>
<td>Adults</td>
<td>Adults</td>
<td>Adults</td>
</tr>
<tr>
<td>22:00</td>
<td>Adults</td>
<td>Adults</td>
<td>Adults</td>
<td>Adults</td>
</tr>
<tr>
<td>23:00</td>
<td>Adults</td>
<td>Adults</td>
<td>Adults</td>
<td>Adults</td>
</tr>
</tbody>
</table>

Figure 1: Example Schedule

Note that this schedule is merely an example schedule so the reader can get a grasp of what a typical schedule in the tennis school organization looks like.
1.2 Problematization

In most tennis clubs the schedule, such as the one shown in the Figure 1, is created manually by one or some of the employees at the club in question (Marescotti, 2018). The current practise for this is that the club chooses which hours, days and courts it would like to have available for the tennis school. It also sets which hours and courts it should be possible for the particular players categories to be scheduled. After the club has set this availability basis for the tennis school the coaches are scheduled. The coach with the highest priority gets its schedule set first, then the coach with the second highest priority gets its schedule set and so on until all of the predetermined available hours and courts have gotten a coach scheduled. The prioritization of the coaches is not well defined but usually the coaches with a certain responsibility for a category, the highest levels and the most restrictions in availability has the highest priority. After the coaches schedule has been set the players are scheduled. The players are also put in a prioritized list with the player with the highest priority scheduled first. This prioritization is done mainly based on whether the player has been taking classes the previous term as well or if it is a new student. The players that are guaranteed to play multiple hours a week are also prioritized higher in order to set their schedule first so that they are not lacking hours when the schedule is done. As was the case for the coaches, this prioritization is not well defined either and the scheduling of the players is done both based on the prioritized list but also on the schedule from the previous semester where some groups stays the same if the players have not expressed any dissatisfaction with the groups. There are some constraints when assembling the groups of players that needs to be taken into account. For example the players must match in a satisfactory manner both in age, level and category. Also, how much the age and level can differ in a group depends on which category the players belong to. The availability of the players is also a factor that needs to be taken into account but that today is not done in a systematic way (Schwieler, 2018).

The advantages with this manual scheduling procedure is that the person responsible for the scheduling can take many soft values into consideration when assembling groups and allocating coaches. For example players that are friends can be placed in the same groups and players that are known to favor a certain coach socially might have a higher chance of getting that coach. Hence the social aspects of training in groups with a coach can be taken into account (Marescotti, 2018). However, the disadvantages are many. First of all, having a person manually creating the schedule is very time consuming. Sometimes it takes up to a month for one or more employees to finish the schedule which is a big cost for a club. Since the constraints in availability of the students are not well defined and not taken into consideration in a careful manner, the schedule often needs to be remade several times before it is satisfying (Marescotti, 2018) and (Schwieler, 2018). A suspicion is also that the scheduling today might be done in a sub-optimal way based on some different objectives. For example there might be ways to create schedules where more players fit and by doing so increasing the income for the club or to assemble better matching groups and increasing the level of satisfaction of the players.

Throughout the report the full problem will be referred to as the tennis school scheduling problem and shortened TSSP. Furthermore, the scheduling of the coaches and the players will be referred to as the coach scheduling problem and player scheduling problem respectively and shortened CSP and PSP.
1.2.1 Case Study

A case study has been conducted in cooperation with Court1, a padel tennis club in Stockholm, Sweden. Padel tennis, or just padel, is a sport that is a mixture between tennis and squash hence Court1 is not a regular tennis club although the premises are the same as those of a tennis club. However when dealing with the problem in general it will be considered a tennis school or tennis club throughout the report.

At Court1’s padel school the grouping of the students, the scheduling of their classes and the allocation of their coaches is done manually in some spreadsheet software. This is a tedious task often done by one or several of the coaches. Commonly it takes up to two weeks and it has to be done before the start of every term/semester. This is a substantial cost for Court1 and as they continue to grow, a way to efficiently automate the procedure can lead to great long term savings.

Padel is one of the worlds fastest growing sports (Sandblom) and along with many other actors in Sweden, Court1 would like to be prepared for that growth. Today the club has 5 courts, 4 coaches and 176 students in its padel school.

1.3 Purpose and Delimitations

The main purpose of this thesis is to develop a heuristic algorithm that automates the scheduling process at a typical tennis club, with the aim to create a feasible schedule for the tennis school. Furthermore it will be examined if a heuristic local search algorithm can be developed to improve this feasible schedule based on the objective to maximize the amount of students fitted in the tennis school. These algorithms will then be tested in a case study at a Padel Club where the performance of the automated scheduling process will be compared to the manual one done by the club.

The first delimitation made for this thesis is that the problem will not be optimized by means of applying any standard optimization software on the model. There are several methods that might result in a mathematical model that is applicable in such a software, but the gain to the tennis organization is mainly the automation of the scheduling and grouping process, and to automate this process, it is more suitably done in an object oriented programming language. Furthermore, in discussions with the ones responsible for grouping and scheduling it became clear that they could not consider all parameters included in the process at once and would therefore find it difficult to assess two similar schedules. To optimize the schedule to such a degree was not what they were interested in. A second delimitation made is that the development of a "ready to use" product of the model developed and the implementation of such will not be handled in this thesis, nor more steps from the operations research approach as described in Introduction to Operations Research (Hillier and Lieberman, 2015).
2 Methodology

In this chapter the methodology applied throughout the thesis is presented. The methodology is specialized for operations research projects and is presented fully in *Introduction to Operations Research*, Frederick S. Hillier and Gerald J. Lieberman, Chapter 2, Page 10.

The outline of the methodology is as given in the list below

1. Define the problem of interest and gather relevant data.
2. Formulate a mathematical model to represent the problem.
3. Develop a computer-based procedure for deriving solutions to the problem from the model.
4. Test the model and refine it as needed.
5. Prepare for the ongoing application of the model as prescribed by management.
6. Implement.

However, in this report only the steps 1-4 are included since the steps 5 and 6 are considered to be beyond the scope of this thesis.

2.1 Problem definition and gathering of data

Unlike textbook examples of optimization or OR problems Hillier and Lieberman (2015) states that most real life problems are initially defined in a vague and imprecise way. Therefore, the first step in an OR project is to study the relevant system and to develop a well defined statement of the problem. Among other things this includes determining the appropriate objectives as well as the constraints on what can be done and also how meddling with the problem at hand might affect other areas of the business. The importance of making a good definition of the problem, Hillier and Lieberman (2015) points out that it is difficult to extract the "right" answer to the "wrong" problem.

Hillier and Lieberman (2015) also states that OR teams usually spends a great amount of time gathering the relevant data about the problem. Much of this data is needed to create further understanding of the problem but also data to provide input for the mathematical model being formulated in the next phase of the study. However much of the needed data is not available when the study begins and so it might be necessary to establish some computer based management information system to collect the necessary data on an ongoing basis.

2.2 Formulating a mathematical model

When having defined the problem properly and gathered the relevant data Hillier and Lieberman (2015) suggestion is to formulate a mathematical model of the problem. They state that all models are idealized representations of the real world. Mathematical models are also idealized representations, only that they are expressed in terms of mathematical symbols and expressions. The model includes quantifiable decision variables, such as $x_1, x_2, ..., x_n$, whose respective values are to be determined. Furthermore, the measure of the performance of the model is then expressed as a function of the decision variables. This is called the objective function and can look something like this: $P = 3x_1 + 2x_2 + ..., + 5x_n$. The restrictions on what values the decision variables can take are also expressed mathematically typically in terms of equations or inequalities such as...
These restrictions are called constraints. The constants of the model, i.e. the coefficients of the objective and the constraints and the right hand side of the constraints are called parameters. Determining the values to assign these parameters is a challenging and important part of the model building process. Unlike most textbook examples where the values are given this requires gathering of relevant data says Hillier and Lieberman (2015). Since the values of the parameters often are estimates it might also be important to examine how the solution might differ due to small changes in the values. This is called sensitivity analysis.

2.3 Develop a computer-based solution procedure

The next step in Hillier and Lieberman (2015) modelling approach is to develop an, often computer based, solution procedure to the problem. This can seem like the most extensive part of the work, however if the problem can be modelled in a way that is well known there are often many ready to use software packages that can be applied to solve the it. Hillier and Lieberman (2015) states that a common theme in OR is to find an optimal solution to the problem. They go on however, to point out that these optimal solutions are only optimal with respect the model being used which as described before is often an idealized representation of the real problem. To make any practical use of the OR study one should therefore, rather then be deluded into demanding the impossible of an utopian "best" solution to the problem make sure that the solution is better than what can be obtained by other means. Hillier and Lieberman (2015) goes on further and points out that satisficing might be more useful in practice than optimizing. Where satisficing is a coining of the words satisfactory and optimizing. The distinction between the two reflects the difference between theory and to implement that theory in practice. They go on and quote one of Englands pioneering OR leaders with that "optimizing is the science of the ultimate; satisficing is the art of the feasible". With this in mind OR-teams should consider the cost of the study and the disadvantages of delaying its completion when developing a more complicated solution procedure. The use of heuristic procedures (intuitively designed procedures that do not guarantee an optimal solution) in the search for a "good" suboptimal solution to the problem are often used in recognition to the concept of developing an unnecessarily complicated solution procedure.

2.4 Test and refine the model

Once the solution procedure seems to be formulated in a satisfying manner Hillier and Lieberman (2015) suggest that the next phase in the OR modelling approach is to test the model. They point out the similarities of developing a large mathematical model and the development on a computer program. When the first version of a computer program is made it inevitably contains many bugs and similarly the mathematical model with solution procedure contains many flaws. Therefore before implementing and using the model it has to be thoroughly tested to try to find and correct as many of the flaws as possible. This testing and improvement phase is often called model validation. How the model validation is done depends greatly on the nature of the problem Hillier and Lieberman (2015) says, however they give three suggestions of general procedures that often can be applied to examine the model. The first one is simple and does only involve looking over the full model again so as to "not lose the forrest for the trees", trees being the small details corrected when doing the first debuggings of the model. For this step they suggest that a person that has not been a part of the modelling is involved so as to get an impartial view. The second step is a more practical one where they suggest that further insight of the validity of model might be gained by varying the values of the parameters to see whether the output behaves in a plausible way. The third step is to do what they call a retrospective test where, if it is applicable, historical
data is used to reconstruct the past and then determining how well the model would have performed if it was used then. Comparing this hypothetical result with the historical result can then give an indication whether using this model would result in an improvement in performance over the current practice. It may of course also indicate areas where the model has shortcomings which might require modifications.
3 Problem Definition and Data Gathering

In this chapter the methods used for specifically defining the problem and the data gathering are presented. To get a further understanding of the actual problem at hand the first section describes an interview study that has been conducted. The Interviewees have long term experience of tennis organizations in general and have all had much insight in the scheduling processes of various tennis schools and similar sports. The second section is focused on related works to this thesis and although there are no related works that has handled the exact same topic, there are some that has inspired a portion of the ideas in this thesis. The next section handles some theoretical ideas regarding heuristic algorithm development in general and also a section which describes how local search strategies works in improvement procedures. In the fourth section, the first three sections are summarized in a requirement specification of the model. The requirement specification handles what input data is needed followed by specific requirements concerning the coaches, players and groups in a tennis school.

3.1 Interviews

When scheduling the tennis or padel school, the process is generally divided in three stages, says Schwieler (2018) and Marescotti (2018). First the courts and hours that the club wants to have available for the tennis school is determined. Then the coaches are scheduled first such that one coach is scheduled on every available court and hour. The last step is to schedule the players in suitable groups depending on age and level and to place the groups on courts where there is a coach with competence matching the level of the group. Both the coaches and players are scheduled from a list in a prioritized order, according to Schwieler (2018) and Marescotti (2018). The higher competence, if the coach is responsible for a certain category and the more restrictions the coach has in its availability the higher prioritized the coach is in general says Schwieler (2018). Schwieler (2018) also states that the players are prioritized higher if they are a previous player in the school, they have multiple guaranteed hours or if they belong to a certain category that has a higher priority than the others.

Schwieler (2018) goes on and says that the way to schedule the coaches also differs depending on what employment type they have got. Some coaches receives a fixed salary each month and can do other tasks at the club if they get free periods in between lessons, while others get paid by the hour and can normally not have free periods. This also affects the minimum work shift for the coaches. The ones that have other chores to do apart from coaching in the tennis school can usually take shifts that are only one hour whereas the ones getting paid by the hour usually needs longer shifts for them to make it worthwhile to work. Furthermore, every coach has limitations in how much and how little they can work every week, says Schwieler (2018) and Marescotti (2018).

The players are all divided into different categories such as "adults", "juniors", "kids" and "competition" says Marescotti (2018) and Schwieler (2018). They are scheduled in groups with other players of the same category and in the groups they have to match sufficiently in age and level. How much the player can differ in age and level respectively depends on the category to which they belong Schwieler (2018) says. Schwieler (2018) goes on and points out that it is also the case that the age and level difference in the categories can vary in the sense that for some categories it may be fitting that if you are younger than the other players, you must be at least at the same level as the older players, whereas for some categories the opposite is true. Marescotti (2018) says that these are naturally club-specific policies that can differ, which it has in the different clubs she
Hjelmgren (2018) says that when scheduling the players it is always good when the groups are perfectly matched, however, if the insistence to create perfectly matching groups is too high, fewer players would get a spot in school. Therefore, he says, it is more important that the groups match sufficiently good in age and level but that the objective should be that the largest amount of players will get a spot in the school.

### 3.2 Related Works

In this section some of the work that is related to this thesis is presented. Since the problematization in this thesis is specific there is no related work that handles the exact same problem but some that has been an inspiration for the ideas in this thesis.

Farmer et al. (2007) handles the problem of scheduling the umpire crews at tennis tournaments. Their scheduling problem, though different in many aspects, has its similarities with some details of the scheduling problem dealt with in this thesis. When scheduling the umpire crews Farmer et al. (2007) has to allocate umpires to many different courts simultaneously, hence, no umpire can be scheduled at more than one court at the same time, much alike for the coaches and players in this thesis. Furthermore there have to be enough umpires at each court every match which also resembles the situation in this thesis for the players in their respective groups but also that there has to be a coach on every court every hour.

Knust (2010) deals with a case of scheduling non professional table tennis leagues. The table tennis league consists of a certain number of teams which play a double round robin tournament during a season, i.e. each team plays against each other team twice (once at home, once away). Some constraints of Knust (2010) model resembles the ones for the players in this thesis where for example each table tennis team is to be scheduled a maximum number of times a day and a limited amount of times a week. Knust (2010) both solves the problem with an ILP solver but also uses an heuristic solution algorithm. In the outline for the heuristic solution procedure the problem is decomposed into subproblems which are then treated hierarchically.

Similarly to the problem of Knust (2010), Croce et al. (1999) handles the scheduling of round robin tennis tournament at a tennis club. Much alike some constraints faced in this thesis and in the other related works Croce et al. (1999) has to model constraints such that each court is assigned a maximum of one match at a time and that each participant is to be scheduled only one match at a time which is the case for the courts and players in this thesis.

### 3.3 Theory

In this section the most relevant mathematical and algorithmic theory will be presented in order for the reader to be able to follow the work throughout the report.

#### 3.3.1 Heuristic Algorithm Development

Algorithms that either give nearly the right answer or provide a solution (not for all instances of the problem) are called heuristic algorithms (Kokash, 2005). Heuristics have been used in the last century or so, to indicate a practical decision rule or a practical way to find a solution to
a problem, relying upon experience and common sense. These rules do not aim at satisfying any formal or theoretical property; sometimes “heuristics” have been defined just as practical alternatives in contrast to formal mathematical techniques (Righini). To model the application in terms of precisely described, well understood problems, it must be described abstractly in terms of procedures on fundamental structures. Many such structures can be described as combinatorial objects, for example:

- **Permutations** - which are arrangements or orderings of items. For example \{1, 4, 3, 2\} and \{4, 3, 2, 1\} are two different permutations of the same set. Permutations are likely to be the object when dealing with "ordering", "tours", "sequences" or "arrangements".

- **Subsets** - which represents selections from a set of items. For example \{1, 3, 2\} and \{4\} are two different subsets of \{1, 2, 3, 4\}. Order does not matter with subsets and so \{1, 2, 3\} is considered the same as \{3, 2, 1\}. Subsets are likely to be the object in question when dealing with "clusters", "groups", "selections" and so on (Skiena, 2008).

Heuristic algorithms are used to solve large instances of computationally difficult problems because the computation of an exact solution would require an excessive amount of computing time. They are thus effective on combinatorial optimization problems because such problems are usually difficult. These algorithms can roughly be divided into two types:

- Specific algorithms for specific problems → Heuristics
- General ideas for almost any problem → Meta-heuristics (Righini)

A heuristic (heuristic rule, heuristic method) is a rule of thumb, strategy, trick, simplification, or any other kind of device which drastically limits search for solutions in large problem spaces. Heuristics do not guarantee optimal solutions; in fact, they do not guarantee any solution at all; all that can be said for a useful heuristic is that it offers solutions which are good enough most of the time. A heuristic program is a program that employs heuristics in solving complex problems (Feigenbaum and Feldman, 1963).

The three most common forms of algorithmic notation are (1) English, (2) pseudocode, or (3) a real programming language. All three methods are useful because there is a natural tradeoff between greater ease of expression and precision (Skiena, 2008).

### 3.3.2 Heuristic Search Methods

Most heuristic search methods have two things in common:

- **Solution space representation** - This is the complete description of the set of all possible solutions to the problem.

- **Cost function** - Search methods need a cost or evaluation function to assess the quality of each element of the solution space.

A local search method employs local neighborhood around every element in the solution space. The search proceeds from the present solution to a neighbouring solution that is better or more promising in some way. (Skiena, 2008)
3.4 Requirement Specification

Based on the conducted interviews and the ideas from the related works and theory presented earlier in this chapter, a generalized requirement specification for the problem has been constructed. The required input data is defined and the specific requirements for the coaches, players, groups, and categories are described with words. The reader is advised to have the table of notations at hand for this part so as to get familiar with the notation when its meaning is described more thoroughly than in the table of notations and in other sections.

3.4.1 Input Data

Some of the data input required for the application is club specific, some is coach specific, some is player specific and some data input needs to be agreed upon and determined together by the club and the coach and so on. Not every required data input will be specifically printed here but together with the table of notation it should be clear.

Club specific: First the club needs to determine what days, hours and courts that are available for tennis school. It also has to define the different categories the players can be assigned and the attributes of the different categories that are listed in the table of notation. The club then determines the set of different player levels, which category each player belongs to and set each player’s level if the player is known to the club, if not, the player itself approximates its own level. It also sets the competence level of the coaches and the different categories the coach has the competence to coach. The club then needs to provide a list of coaches and a list of players, both ordered by priority.

Coach specific: Each coach needs to specify its availability, the maximum, minimum and desired amount of hours it can work a week. Each coach also needs to specify if it can have free periods in between lesson, how long or short shift it can work.

Player specific: Each player needs to specify its availability and desired amount of hours it would like each week. The player also states its age, its level (if necessary), the maximum amount of hours in a row it is prepared to play and if it is okay to get scheduled twice the same day.

3.4.2 Coach Specific Requirements

A coach:

- of higher priority will if possible be scheduled before a coach of lower priority.
- can only be scheduled when available.
- cannot be scheduled on a court unless the court is available and the coach have competence to coach the allowed categories on that court.
- will not work unless the coach is assigned its minimum work hours.
- cannot work more than maximum work hours.
- would like desired amount of work hours.
- cannot be scheduled for shifts less than minimum shift hours or longer than maximum shift hours.
- cannot have free period during the work shift if that is specified.
- needs to have required competence to match groups that it coaches.
3.4.3 Player Specific Requirements

A player:

- of higher priority will if possible be scheduled before a player of lower priority.
- cannot be scheduled when it is not available.
- cannot be scheduled on a court unless the court is available and the player’s category is in the list of allowed categories on that court.
- cannot be scheduled for more than that player’s category’s specific allowed maximum hours in a day or exceed its maximum hours in a row.
- cannot be scheduled more than that player’s specific maximum hours in a row.
- should get at least its guaranteed hours, if possible.
- cannot be scheduled more than its desired hours.
- will not get free periods in between lessons in the morning or in the afternoon but can be scheduled both in the morning hours and in the afternoon hours the same day.

3.4.4 Group Specific Requirements

A group:

- is assigned the same category as the players in it.
- will not be initiated and scheduled unless the minimum amount of players fits in the group. The minimum amount of players is a category specific attribute.
- cannot have more than its maximum number of players in it. The maximum amount of players is a category specific attribute.
- must have each evenly matched players in it based on age and level. The maximum allowed age and level difference and how to apply them are player category specific attributes.
4 Mathematical Modelling

In this chapter the problem is modelled mathematically based on the problematization and requirement specification. The first section describes some additional functions to simplify the understanding and writing of the rest of the model. The following sections then divides the problem into two parts, coach scheduling and player scheduling. The coach scheduling is executed prior to the player scheduling since it renders some input data for the PSP. The problems are formulated as optimization problems and although they are not modelled in a solver to produce an optimal solution, this yields a more precise formulation of what is desired and what is required from the model as well as provide an easier way to formulate the computer based algorithm. Most of the notation is stated with logical names; however, the reader is advised to have the table of notations at hand when reading this chapter. Throughout the chapter there are several summations over an entire set without specifying which variable is connected to which set, this is however described in the table of notations and omitted to improve readability of the equations.

4.1 Further Simplifying Equations and Functions

Most functions are needed for the mathematical modelling, and some are needed to more accurately describe the computer based solution.

Equation 4.1 is the sum of all hours that are available for tennis school lessons. The first remark is that if \( n \) courts are available at hour \( h \) this implies \( n \) available school hours. The second remark is that if a court is scheduled with a coach and a group, it is no longer available, but \( a_{h,d,court} \) is still equal to one.

\[
\text{available school } h = \sum_W \sum_H \sum_{\text{Courts}} a_{h,d,court} \quad (4.1)
\]

Equation 4.2 determines if a coach can be scheduled on a certain court, hour and day. It gives 1 if true and 0 if false. This value is not constant and varies before and after a coach has been scheduled.

\[
\text{can be scheduled } c,h,d,court = a_{c,h,d,court} \cdot a_{h,d,court} \cdot \text{can cat } c, cat \quad (4.2)
\]

\[\forall c, h, d, \text{court and cat } \in \text{Cats}_{h,d,\text{court}}.\]

Equation 4.3 determines if a player can be scheduled on a certain court, hour and day. It gives 1 if true and 0 if false. This value is not constant and varies before and after a player has been scheduled.

\[
\text{can be scheduled } p,h,d,court = a_{p,h,d,court} \cdot \text{allowed cat } p_{h,d,court} \cdot \text{c match } p_{h,d,court}, p \quad (4.3)
\]

\[\forall h, d, \text{court}.\]

Equation 4.4 determines unused capacity (UC), i.e. scheduled coaches where no group is scheduled.

\[
UC = \sum_W \sum_H \sum_{\text{Courts}} \sum_{\text{Coaches}} \sum_{\text{Groups}} (B_{c,h,d,court} - B_{g,h,d,court}) \quad (4.4)
\]

Equation 4.5 determines how many player guaranteed hours (GH) that are not fulfilled, in total.

\[
GH = \sum_{\text{Players}} \max(\text{guaranteed } h_p - \sum_{\text{Week}} \sum_H \sum_{\text{Courts}} B_{p,h,d,court}, 0) \quad (4.5)
\]
4.2 Coach Scheduling Problem

Minimize the sum of the deviations of a coach’s scheduled work hours and the coach’s desired amount of work hours.

\[ Z_1 = \min_{Coaches} \left( \sum_{W} \sum_{H} \sum_{Courts} B_{c,h,d,court} \cdot \left( \sum_{W} \sum_{H} \sum_{Courts} B_{c,h,d,court} \right) - \text{desired}_{h_c} \right)^2 \]  \hspace{1cm} (4.6)

Subject to:

Coach constraints

A coach can only be scheduled at one court at a time.

\[ \sum_{Courts} B_{c,h,d,court} \leq 1, \forall c, h, d. \]  \hspace{1cm} (4.7)

Each coach has a minimum number of hours they need to be scheduled to assure that they still want to work for the club.

\[ \sum_{W} \sum_{H} \sum_{Courts} B_{c,h,d,court} \geq \text{min}_{h_c}, \forall c \in \text{Coaches}. \]  \hspace{1cm} (4.8)

A coach can not be scheduled more than its given maximum work hours.

\[ \sum_{W} \sum_{H} \sum_{Courts} B_{c,h,d,court} \leq \text{max}_{h_c}, \forall c \in \text{Coaches}. \]  \hspace{1cm} (4.9)

To ensure that a coach only is scheduled if it also can satisfy its required hours of a minimum shift

\[ \sum_{h=h_0}^{h_0+\text{min}_{shift_{c}}-1} \sum_{Courts} B_{c,h,d,court} = \begin{cases} \text{min}_{shift_{c}}, & \text{if coach } c’s \text{ shift starts at hour } h_0 \\ \text{day } d \text{ and court } court \\ 0, & \text{otherwise.} \end{cases} \]  \hspace{1cm} (4.10)

To avoid free periods for those whose employment type does not allow free periods, a coach must be booked hour \( h \), day \( d \) and court \( court \) if it is booked the previous or following hour.

\[ B_{c,h,d,court} = 1, \text{ if } B_{c,h-i,d,court} = 1 \text{ and } B_{c,h+j,d,court} = 1, \forall \text{ court and } \]  \hspace{1cm} (4.11)

\[ \forall h \in (h-i, h+j) \text{ where } [h-i, h+j] \subseteq \text{am or pm and } i = 1, 2, ..., j = 1, 2, ... \]

One and only one coach shall be scheduled at each court each available hour

\[ \sum_{Coaches} B_{c,h,d,court} \cdot a_{h,d,court} = 1 \forall h, d, court. \]  \hspace{1cm} (4.12)

The sum of all booked hours for the coaches must equal the sum of all available school hours

\[ \sum_{Coaches} \sum_{W} \sum_{H} \sum_{Courts} B_{c,h,d,court} = \text{available}_{schoolh}. \]  \hspace{1cm} (4.13)
A coach shall only be scheduled if it is available, the court is available for school and the coach has competence to coach some of the allowed categories.

\[
\sum_{W} \sum_{H} \sum_{Courts} \sum_{Cats, h, d, court} B_{c, h, d, court} \cdot \text{canbescheduled}_{c, h, d, court} = \sum_{W} \sum_{H} \sum_{Courts} B_{c, h, d, court}.
\] (4.14)

4.3 Player Scheduling Problem

Maximize the total sum of, priority weighted, scheduled hours for the players.

\[
Z_2 = \max \sum_{Players} \sum_{W} \sum_{H} \sum_{Courts} w_p \cdot B_{p, h, d, court}
\] (4.15)

Subject to:

**Player Constraints**

A player shall not get less than its guaranteed hours nor more than its desired or max hours.

\[
guaranteedh_p \leq \sum_{W} \sum_{H} \sum_{Courts} B_{p, h, d, court} \leq \min(desiredh_p, maxh_{cat_p}).
\] (4.16)

A player shall not get more hours in a row than allowed.

\[
\sum_{h} \sum_{h\in\text{maxhinarow}_p} B_{p, h, d, court} \leq \text{maxhinarow}_p, \forall h \in H
\] (4.17)

A player, much like a coach whose employment type does not allow free periods, shall not have a free period in between lessons. A player is however allowed to have one morning lesson and one afternoon lesson. The constraints on i and j

\[
\sum_{j=1}^{h_{end}-h} B_{p, h+j, d, court} = 0, \text{ if } B_{p, h-i, d, court} = 1 \text{ and } B_{p, h, d, court} = 0
\]

where \(i \in \{1, 2, ..., 11\}\), \(\forall h\), where \([h-i, h+j] \subseteq \text{am or pm}\) and \(h_{end} = 12\) or \(h_{end} = 24\).

(4.18)

A player shall not be scheduled when not available.

\[
\sum_{W} \sum_{H} \sum_{Courts} B_{p, h, d, court} \cdot \text{canbescheduled}_{p, h, d, court} = \sum_{W} \sum_{H} \sum_{Courts} B_{p, h, d, court}, \forall p.
\] (4.19)

A player shall only be scheduled on a certain hour, day and court if the coach that is scheduled at that hour, day and court has the competence to coach the player

\[
\sum_{W} \sum_{H} \sum_{Courts} B_{p, h, d, court} \cdot \text{cmatch}_{p, c, h, d, court} = \sum_{W} \sum_{H} \sum_{Courts} B_{p, h, d, court}, \forall p.
\] (4.20)

**Group Constraints**
All players in a group must be of the same category

\[ cat_p = cat_g, \forall \ p \in \text{players}_g, \ \forall \ g. \]  \hspace{1cm} (4.21)

A group must have more than or equal to its minimum number of players and less than or equal to its maximum number of players

\[ \sum_{\text{Players}} ping_{p,g} \leq \max ping_{\text{cat}}, \forall \text{ groups}. \]  \hspace{1cm} (4.22)

Each group is assigned one coach

\[ \sum_{\text{Coaches}} cing_{c,g} = 1, \forall \text{ groups}. \]  \hspace{1cm} (4.23)

Each group is to be scheduled once

\[ \sum_{H} \sum_{W} \sum_{\text{Courts}} B_{g,h,d,court} = 1, \forall \ g. \]  \hspace{1cm} (4.24)

The difference in level among the players in one group cannot exceed the allowed difference in level for that category

\[ |l_{p_1} - l_{p_2}| \leq \text{leveldiff}_{cat_g}, \forall \text{ pairs of players } (p_1, p_2) \subseteq \text{players}_g, \forall \ g. \]  \hspace{1cm} (4.25)

The difference in age among the players in one group cannot exceed the allowed difference in age for that category

\[ |age_{p_1} - age_{p_2}| \leq \text{agediff}_{cat_g}, \forall \text{ pairs of players } (p_1, p_2) \subseteq \text{players}_g, \forall \ g. \]  \hspace{1cm} (4.26)
5 Computer Based Solution Procedure

This chapter describes the heuristic algorithms that have been developed to produce a solution to the TSSP. The first section introduces some additional functions that simplify the notation and the understanding of the algorithms that will follow. The following sections are then: Schedule Coaches, Schedule Players and Improvement Algorithm. In each of those sections the algorithms are first described in words followed by a description in pseudocode as suggested by Skiena (2008).

For this chapter the reader is, once again, advised to have the table of notations at hand.

5.1 Additional Functions

To simplify the pseudocode and make it easier to follow if one is not familiar with the terms and format used, some equations are defined.

Coach Related

Equation 5.1 returns the number of booked hours for a specific coach, $c$.

$$ bookedh_c = \sum_H \sum_W \sum_{\text{Courts}} B_{c,h,d,court}. \tag{5.1} $$

Equation 5.2 returns 1 if no coach is scheduled this hour, day and court and 0 otherwise.

$$ \text{nocoachscheduled}_{h,d,court} = \begin{cases} 1, & \text{if } \sum_{\text{Coaches}} B_{c,h,d,court} = 0 \\ 0, & \text{otherwise}. \end{cases} \tag{5.2} $$

Equation 5.3 returns 1 if a coach $c$ is not scheduled above its’ $\text{maxh}_c$ and 0 otherwise.

$$ \text{notfullybooked}_{p,h,d,court} = \begin{cases} 1, & \text{if } \text{bookedh}_p < \text{desiredh}_p \\ 0, & \text{otherwise}. \end{cases} \tag{5.3} $$

Equation 5.4 determines whether it is possible to schedule coach $c$ at hours $[h,...,\text{minshift}_c - 1]$ on day $d$ i.e for enough hours to cover the minimum amount required for a minimum shift. It returns 1 if it is possible and 0 otherwise.

$$ \text{canscheduleminshift}_{c,h,d,court} = \begin{cases} 1, & \text{if } \sum_{h}^{\text{h}+\text{minshift}_c - 1} \sum_{\text{Courts}} \text{canbescheduled}_{c,h,d,court} = \text{minshift}_c \\ 0, & \text{otherwise}. \end{cases} \tag{5.4} $$

Equation 5.5 checks whether coach $c$ already has been scheduled at hours $[\text{minshift}_c,...,h - 1]$ on day $d$ i.e that it is already scheduled for enough hours to cover the minimum amount required for a minimum shift.

$$ \text{alreadyscheduledminshift}_{c,h,d,court} = \begin{cases} 1, & \text{if } \sum_{h}^{\text{h} - \text{minshift}_c} \sum_{\text{Courts}} B_{c,h,d,court} = \text{minshift}_c \\ 0, & \text{otherwise}. \end{cases} \tag{5.5} $$

To make better use of $\text{canbescheduled}$, it is redefined, in order to make the pseudocode more
readable. It is defined as

\[
\text{canbescheduled}_{c,h,d,court} = a_{c,h,d,court} \cdot a_{h,d,court} \cdot \text{cancat}_{c,cat} \cdot (5.6)
\]

\[
\text{nocoachscheduled}_{h,d,court} \cdot \text{not fullybooked}_{c},
\]

\[
\forall c, h, d, court, cat \in \text{Cats}_{h,d,court}.
\]

To clarify, equation 5.6 now includes the condition that only one coach is to be booked per available court, hour and day and ensures the coach does not get fully booked.

**Player Related**

As for the coaches, booked hours for a player is defined as

\[
\text{bookedhp} = \sum_{H} \sum_{W} \sum_{\text{Courts}} B_{p,h,d,court}.
\]

(5.7)

A new set must be defined to explain how the computer based algorithm schedules the players. Instead of looping through available school hours, the least popular hours are scheduled first. These are defined as a set of sorted hours.

\[
\text{SortedWeek} \triangleq [(h_{1,d}), (h_{2,d}), ..., (h_{k,d})], \text{ where } (h_{i,d}) \text{ is the i:th least popular hour among the players and } k \text{ is the number of available hours.}
\]

(5.8)

### 5.2 Schedule Coaches

The very general rules of thumb, as Feigenbaum and Feldman (1963) described a heuristic as, that set out the fundamental framework for the coach scheduling algorithm are as follows:

1. Pick the first non-picked coach in the priority ordered set of coaches.
2. Schedule the coach on feasible work shifts greater than or equal to the minimum work shift length. Try to proceed until the coach’s desired amount of work hours is met, then take the next coach and repeat for all coaches.
3. If there are still hours and courts available that have not gotten a coach then repeat 1 and 2 until the coach’s maximum amount of work hours is met.

For a computer based solution procedure; however, the rules of thumb are not enough and so a more precise way of how to actually go about and schedule the coaches needs to be formulated. Hence, with the rules of thumb as a ground a systematic algorithm that considers the heuristic is developed in the section below.

#### 5.2.1 Schedule Coaches Pseudocode

The coach scheduling algorithm is split into two parts. Algorithm 1 tries to schedule the coaches until their desired amount of work hours is met or until there is no coach that can be scheduled. If there are coaches that can be scheduled after their desired amount of work hours are met but there are still available courts and hours for the tennis school without a scheduled coach, then Algorithm 2 schedules the coaches until all hours are filled or until there is no coach that can be scheduled.
**Data:** Coaches

**Result:** Preliminary Coach schedule

initialization;

\[ \text{begin} \]

\[ \text{WeeklySchedule}_{d,h,court} \leftarrow \emptyset, \forall d, h, court \]

\[ \text{while} \ \exists (c, h, d, court) \text{ such that } \text{desired}_c < \text{booked}_c \text{ AND canbescheduled}_{c,h,d,court} \text{ do} \]

\[ /* \text{Coaches have desired hours free to book} */ \]

\[ \text{for } c \in \text{Coaches} \text{ do} \]

\[ \text{for } d \in \text{Week} \text{ do} \]

\[ \text{for } court \in \text{Courts} \text{ do} \]

\[ \text{for } h \in H \text{ do} \]

\[ \text{if } \text{freeperiod}_c = 1 \text{ AND canbescheduled}_{c,h,d,court} \text{ then} \]

\[ \text{B}_{c,h,d,court} = 1 \]

\[ \text{Append to WeeklySchedule} \]

\[ \text{else if alreadyscheduledminshift}_c \text{ AND canbescheduled}_{c,h,d,court} \text{ then} \]

\[ \text{B}_{c,h,d,court} = 1 \]

\[ \text{Append to WeeklySchedule} \]

\[ \text{else if canscheduleminshift}_c \text{ then} \]

\[ \text{for } \text{hour} \in [h, h + \text{minshift}_c - 1] \text{ do} \]

\[ \text{B}_{c,h,d,court} = 1 \]

\[ \text{Append to WeeklySchedule} \]

\[ \text{else} \]

\[ \text{pass} \]

\[ \text{end} \]

\[ \text{end} \]

\[ \text{end} \]

\[ \text{end} \]

\[ \text{end} \]

\[ \text{end} \]

**Algorithm 1:** Schedule Coaches until Coaches’ Desired Hours Are Met
**Data:** Coaches, Preliminary Coach Schedule, Week, H and Courts

**Result:** Coach schedule

initialization;

begin

while \( \exists c, h, d, \text{court such that canbescheduled}_{c, h, d, \text{court}} = 1 \) and \( \text{coachalreadyscheduled}_{c, h, d, \text{court}} = 0 \) do

/* Coaches can be scheduled, the school hours are not fully booked */

for \( c \in \text{Coaches} \) do

for \( d \in \text{Week} \) do

for \( \text{court} \in \text{Courts} \) do

for \( h \in \text{H} \) do

if \( \text{freeperiod}_c = 1 \) AND \( \text{canbescheduled}_{c, h, d, \text{court}} \) then

\( B_{c, h, d, \text{court}} = 1 \)

DailySchedule

else if \( \text{alreadyscheduledminshift}_c \) AND \( \text{canbescheduled}_{c, h, d, \text{court}} \) then

\( B_{c, h, d, \text{court}} = 1 \)

Append to WeeklySchedule

else if \( \text{canscheduleminshift}_c \) then

for hour \( \in [h, h + \minshift_c - 1] \) do

\( B_{c, h, d, \text{court}} = 1 \)

Append to WeeklySchedule

end

else

pass

end

end

end

end

end

end

**Algorithm 2:** Schedule Coaches to Fill "availableschoolh"
5.3 Schedule Players

To formulate the problem of scheduling the players in a more "abstract" way, as suggested by Skiena (2008) it can be viewed as the combinatorial objects subsets that was described in section 3.3. Consider the set $\text{Players} = \{p_1, p_2, ..., p_n\}$, this set can be split up in different subsets such as $Q, g_1, g_2, ..., g_k$ where $Q$ represents the queue of players that want more school hours and the $g$-sets are of course different scheduled groups. To solve this is to solve the PSP.

As was done for the CSP some rules of thumb as for how to deal with the PSP was formulated. An important thing to repeat is that the coaches have been scheduled prior to the players aiming to get as close to the desired amount of "possible" work hours as possible; however, a coach has no work unless groups of players are scheduled on on its court. The rules of thumb are formulated as:

For every hour, day and court:

1. Go through the player queue and try to assemble a feasible group with the highest prioritized player as possible. If the players $\{p_1, p_2, p_3, p_4\}$ are in prioritized order and either $\{p_1, p_4\}$ or $\{p_2, p_3\}$ can be assembled then $\{p_1, p_4\}$ will be scheduled first.

2. Schedule groups of players on the courts based on the priority of the coaches. That is the highest prioritized coach gets its groups first.

3. Schedule a group when it is full or when it is full "enough".

4. If a feasible group cannot be found with the highest prioritized player, discard it from the queue at that hour, day and court and try the next player in the prioritized queue.

5. Switch to schedule groups to a lower prioritized coach when the minimum amount of work hours for the current coach have been covered.

6. Repeat the scheduling process by trying to schedule groups at the most unpopular hours first to avoid scheduling players at popular hours if they are able to attend the unpopular ones. Continue with this process until all coaches are fully booked, or until there are no more available tennis school hours.

These rules of thumb then forms a ground for the algorithm developed to schedule the players as will be presented in the section below.
5.3.1 Schedule Players Pseudocode

**Data:** Coach schedule, Players

**Result:** Preliminary Player Schedule

**Algorithm 3: Schedule Players until Coaches Minimum Hours are Filled**

```plaintext
begin
  while ∃ c and (h, d, court) such that bookedh_c < desiredh_c AND a_h,d,court=1 do
    for c ∈ Coaches do
      while desiredh_c > bookedh_c do
        for d ∈ Week do
          for court ∈ Courts do
            for h ∈ H do
              if B_c,h,d,court = 1 AND B_g,h,d,court = 0 then
                for p1 ∈ Players do
                  if canbescheduled_{p1,h,d,court} = 1 = 1 then
                    Initiate group g
                    Add p to players_g
                    Add g to Groups_{p1}
                    B_{p1,h,d,court} = 1
                    cing_{c,g} = 1
                end
              end
            end
          end
        end
      end
    end
  end
end
```
Data: Coach Schedule, Preliminary Player Schedule

Result: Player Schedule

initialization;

begin

while ∃ c and (h, d, court) such that \( B_{c,h,d,court} = 1 \) AND \( B_{g,h,d,court} = 0 \), ∀ g do

for \((h,d) \in \text{SortedWeek}\) do

for court ∈ Courts do

  if \( B_{c,h,d,court} = 1 \) AND \( B_{g,h,d,court} = 0 \) then

  for \( p_1 \in \text{Players} \) do

    if canbescheduled\(_{p_1,h,d,court} = 1 \) then

      Initiate group g

      Add \( p \) to players\(_g\)

      Add g to Groups\(_{p_1}\)

      \( B_{p_1,h,d,court} = 1 \)

      \( cing_{c,g} = 1 \)

  end

  for \( p_2 \in \text{Players} \) do

    if \( p_1 \neq p_2 \) AND canbescheduled\(_{p_2,h,d,court} \) * pmatch\(_{p_2,g} = 1 \) then

      Add \( p_2 \) to players\(_g\)

      Add g to Groups\(_{p_2}\)

      \( B_{p_2,h,d,court} = 1 \)

      if length of players\(_g\) = maxping\(_{cat}\), then

      | Go to next available h, day and court in Sortedweek

    end

  end

end

if length of players\(_g\) < minping\(_{cat}\), then

Remove g from groups\(_p\)

Set \( B_{p,h,d,court} = 0 \), ∀ p ∈ g

\( cing_{c,g} = 0 \)

end

end

end

Algorithm 4: Schedule Players until All School Hours are Booked or until No Player can be Scheduled
With the notation from the abstract formulation of the PSP a solution from Algorithm 3 can then look like in example 1.

**Example 1. Example solution to PSP**

Consider the set of players \( \text{Players} = \{p_1, p_2, \ldots, p_n\} \) as input to the algorithm. An example of the generated output (or solution) can look something like this:

\[
\text{Players} \Rightarrow Q = \{p_4, p_{13}, p_{50}\}, \quad G = \{g_1 = \{p_1, p_3, p_{12}, p_8\}, \quad g_2 = \{p_2, p_4, p_9, p_{45}, p_{65}\}, \ldots, \quad g_k = \{p_{15}, p_{37}, p_{50}\}\}
\]

where \( Q \) is the queue of players and \( g_1, \ldots, g_k \) are the scheduled groups of players, all associated with an hour, day, court and coach.

**Remark:** the first thing to note about Example 1 is that \( p_4 \) and \( p_{50} \) are both in the queue and in group \( g_2 \) and \( g_k \) respectively. This can be the case if \( p_4 \) and \( p_{50} \) have been scheduled less than their desired or guaranteed amount of hours but also less than their maximum amount of hours. Hence, a person is only discarded from the queue if it has been scheduled its desired or maximum amount of hours. Another thing to note is that all the possible subsets, i.e different representations of \( Q \) and \( G \), forms the solution space representation as was formulated by Skiena (2008) in Section 3.3.

### 5.4 Improvement Algorithm

As was stated by Feigenbaum and Feldman (1963) in Section 3.3, when employing a heuristic algorithm to produce a solution to a problem it is not guaranteed that the solution is optimal or even feasible. When analyzing the solutions obtained by the algorithms above to the CSP and PSP it is obvious that by scheduling in the hierarchical order of priority there might be coach schedules and group constellations that are better in terms of the objectives stated in equations 4.6 and 4.15 but that these constellations are not tested. There might be many improvements to the schedule that comes to mind; however, the algorithm will seek to improve the solution to the PSP in terms of equation 4.15.

With the notation from example 1, consider the solution before applying the improvement algorithm being \( S_1 = (Q_1, G_1) \) and the solution after applying the improvement algorithm being \( S_2 = (Q_2, G_2) \). The only way for \( S_2 \) to be a better solution than \( S_1 \) is if the value of \( Z_2(S_2) > Z_2(S_1) \), where \( Z_2 \) is as given in equation 4.15 and can be seen below:

\[
\text{maximize } Z_2 = \sum_{\text{Players}} \sum_{W} \sum_{H} \sum_{\text{Courts}} w_p \cdot B_{p,h,d,court}
\]

or in words *if more players from the queue are scheduled*. Hence, \( Z_2 \) then represents the cost function as given by Skiena (2008) in Section 3.3.

The rules of thumb formulated for the improvement heuristic was stated as:

1. Do not alter the coach schedule
2. Take the highest prioritized player remaining in the queue and try to rearrange the players in the groups such that the player from the queue can get a feasible spot.
3. Only move one player at a time.
4. The number of groups should not be increased.

5. A player from the queue should not get a spot at the expense of a higher prioritized player losing its spot.

There might be many ways to make this kind of improvement to the schedule, one idea being explained in Example 2 where a potential improvement to the solution in Example 1 is proposed.

**Example 2. Possible improvement of example solution to PSP**

Let the solution, \( S_1 = (Q_1, G_1) \), from the PSP be the one from example 1, that is:

\[
Q_1 = \{ p_4, p_{13}, p_{50} \}, \quad G_1 = \{ g_1 = \{ p_1, p_3, p_{12}, p_8 \}, \quad g_2 = \{ p_2, p_4, p_9, p_{45}, p_{65} \}, \ldots, \quad g_k = \{ p_{15}, p_{37}, p_{50} \} \}
\]

Let \( g_1 \) and \( g_k \) be groups such that they can have a maximum of 4 players in them and so there is a spot left in \( g_k \). What if \( p_{13} \) could play with the players in \( g_1 \) but since they have higher priority the group got full and obviously \( p_{13} \) can not be scheduled in \( g_k \) or any other group and so it is left in \( Q \). Now what if any of the players in \( g_1 \), for example \( p_{12} \), could just as well play in \( g_k \)? Then, by moving \( p_{12} \) to \( g_k \) and let \( p_{13} \) fill the vacant space, so the solution could become:

\[
Q_2 = \{ p_4, p_{50} \}, \quad G_2 = \{ g_1 = \{ p_1, p_3, p_{13}, p_8 \}, \quad g_2 = \{ p_2, p_4, p_9, p_{45}, p_{65} \}, \ldots, \quad g_k = \{ p_{12}, p_{15}, p_{37}, p_{50} \} \}
\]

this would not violate the order of priority among the players and \( S_2 \) is obviously better than \( S_1 \). △

The improvement that Example 2 forms the general idea behind the improvement algorithm developed for PSP; however, there is one additional expansion to add. Consider the case where \( p_{12} \) from example 2 could not fit in \( g_k \) then, instead of immediately giving up, maybe \( p_{12} \) could repeat the process that \( p_{13} \) did and replace a player from a full group and then that player could be tried to fit in a group with a spot left. This is the full idea of the improvement algorithm and the number of times that any player can not fill a group with spots left but replace a player in a full group will be referred to as *depth* and each step towards that depth will be referred to as a *layer*. This procedure is visualized in Figure 2, and a more thorough explanation of the systematic execution of this procedure is described in pseudocode in the next section.
Figure 2: Improvement Algorithm. Performed after initial scheduling.
5.4.1 Improvement Algorithm Pseudocode

**Data:** Player Schedule, \( Q_1 \)

**Result:** Improved Player Schedule

begin

\[ \text{for } p_1 \in Q_1 \text{ do} \]

\[ \text{layer} = 1 \]

\[ \text{for } g \in \text{Groups, where cat}_g = \text{cat}_{p_1} \text{ do} \]

\[ \text{compatible players} = [ ] \]

\[ \text{for } p_2 \in \text{players}_g \text{ do} \]

\[ \text{if } p_1 \text{match}_{p_2, p_1} = 1 \text{ then} \]

\[ \text{Append } p_2 \text{ to } \text{compatible players} \]

end

if length of compatible players = 0 then

\[ \text{Pass} \]

else

\[ \text{Get } (h, d, \text{court}) \text{ from } B_{g, h, d, \text{court}} = 1 \]

\[ \text{Remove } g \text{ from Groups} \]

\[ \text{Put } p_1 \text{ in } g \]

\[ \text{for } p_2 \in \text{compatible players do} \]

\[ \text{Remove } p_2 \text{ from } g \]

\[ \text{for } g_2 \in \text{Groups do} \]

\[ \text{if } \text{length of players}_{g_2} < \text{maxping}_{\text{cat}_{g_2}} \text{ AND } p_2 \notin \text{players}_{g_2} \text{ then} \]

\[ \text{Get } (h, d, \text{court}) \text{ from } B_{g_2, h, d, \text{court}} = 1 \]

\[ \text{if } \text{canbescheduled}_{p_2} \text{ AND } p_1 \text{match}_{g_2} \text{ then} \]

\[ \text{Put } p_2 \text{ in } g_2 \]

\[ B_{p_2, h, d, \text{court}} = 1 \]

\[ \text{Repeat with next player in } Q_1 \]

end

if layer < depth then

\[ \text{layer} = \text{layer} + 1 \]

\[ \text{Remove } g_2 \text{ from Groups} \]

\[ \text{Repeat from the 2nd for-loop using } p_2 \text{ as } p_1 \]

end

\[ \text{Put } g_2 \text{ in Groups} \]

end

\[ \text{Put } p_2 \text{ in } g \]

end

end

end

**Algorithm 5:** Improves the Player Schedule
6 Padel Club Case

This chapter describes the testing of the model on real data provided by the padel club Court1 Academy in Stockholm Sweden. The computer based solution procedures from the previous chapter have been set up with the ? programming language and the coach and player data is passed to the program as csv-files. The first section gives an overview of the input data where the players availability data to some extent is uncertain and so two approaches for generating the availability data have been conducted; one simplified, deterministic model where the players are assumed to be available at all times and one stochastic model. The stochastic model randomizes the availability of each player from a distribution that is based on an "availability survey" (Appendix 10.6) performed on the members of Court1. This is in line with what Hillier and Lieberman (2015) suggested when all data required is not available in the beginning of an OR-project. The stochastic model is then simulated multiple times to analyze how the model performs on average. In the second section of the chapter the relevant results of the two methods are presented.

6.1 Input Data

General Club Data

The general data for Court1 Academy is shown in table 2. The availability choice for the courts are given in Appendix 10.4.

<table>
<thead>
<tr>
<th>L</th>
<th>Courts</th>
<th>( a_{h,d,court} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>{1,2,3,4}</td>
<td>{Court1, Court2, ..., Court5}</td>
<td>see appendix 10.4</td>
</tr>
</tbody>
</table>

Table 2: General Club Data

Categories

The categories and associated data as defined by Court1 is given in table 3. What categories that are allowed to be scheduled at what hours are also given in Appendix 10.4.

<table>
<thead>
<tr>
<th>Categories</th>
<th>maxh</th>
<th>guaranteedh</th>
<th>maxhinarow</th>
<th>minping</th>
<th>maxping</th>
<th>agediff</th>
<th>leveldiff</th>
</tr>
</thead>
<tbody>
<tr>
<td>Junior</td>
<td>4</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Adult</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>30</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 3: Category Data

Coaches

The data associated with the coaches at Court1 is given in the table 4.

<table>
<thead>
<tr>
<th>Coaches</th>
<th>level</th>
<th>Cats</th>
<th>minh</th>
<th>maxh</th>
<th>desiredh</th>
<th>minshift</th>
<th>freperiod</th>
</tr>
</thead>
<tbody>
<tr>
<td>( c_1 )</td>
<td>4</td>
<td>Junior, Adult</td>
<td>10</td>
<td>15</td>
<td>15</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>( c_2 )</td>
<td>4</td>
<td>Junior, Adult</td>
<td>10</td>
<td>25</td>
<td>20</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>( c_3 )</td>
<td>3</td>
<td>Junior, Adult</td>
<td>29</td>
<td>33</td>
<td>33</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>( c_4 )</td>
<td>2</td>
<td>Junior, Adult</td>
<td>9</td>
<td>10</td>
<td>9</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 4: Coach Data

Players

At Court1 Academy 176 players applied for a spot in the padel school. The structure of the player data is given in table 5; however, the entire player data set is given in Appendix 10.5. As for the stochastic model of the distribution from which the availability of the players is randomized is defined in Appendix 10.7.
The attributes that determines in which order the players are prioritized can vary depending on the club policy, Court1 sorts their players as

1. Individual priority (set by the head of scheduling)

2. Previous player

3. Guaranteed hours (the more guaranteed hours the higher priority)

One remark to the sorting is that since "guaranteed hours" is the last attribute on which the set is sorted it means that the players with the highest priority are those that have the most guaranteed hours.

6.2 Results

In this section the most relevant results from the models are presented. A value that will be presented for every schedule in the results that is not yet defined is the total amount of booked player hours. This is similar to the value of the objective function to the PSP but without the priority weight assigned to each player and will be defined as $Z_3$ in Equation 6.1.

$$Z_3 = \sum_{p} \sum_{W} \sum_{H} \sum_{\text{Court}} B_{p,h,d,court}$$

6.2.1 Manually Produced Schedule by Court1

The metrics for the manually created schedule made by the head of scheduling at Court1 (Schwieler, 2018) is shown in table 6. The value $|Q|$ denotes the cardinality of the players queue i.e the number of players left in the queue.

| Execution Time | $Z_2$ | $Z_3$ | $|Q|$ | GH | Feasible |
|----------------|-------|-------|--------|-----|----------|
| 3 weeks        | 11.47 | 165   | 41     | 17  | No       |

Table 6: Results Manual Schedule

6.2.2 Automated Schedule

The metrics for the automated schedule of the deterministic is shown in Table 7. For the deterministic model, the value of $a_{p,h,d,court} = 1 \forall p, h, d, \text{court}$.

Deterministic Model

| Execution Time | $Z_2$ | $Z_3$ | $|Q|$ | UC | GH | Feasible |
|----------------|-------|-------|--------|-----|-----|----------|
| 24 seconds     | 12.29 | 201   | 7      | 9   | 2   | No       |

Table 7: Result Automated Schedule - Deterministic Model
Stochastic Model
In the stochastic model the hours for which the value $a_{p,h,d,court} = 1$ for player $p$ each day $d$ are randomized from the distribution in appendix 10.7. The scheduling process was then simulated 10 times with the results shown in table 8.

| Runtime | Execution Time (seconds) | $Z_2$ | $Z_3$ | $|Q|$ | UC | GH | Feasible |
|---------|--------------------------|-------|-------|------|----|----|----------|
| 1       | 11.38                    | 11.54 | 126   | 78   | 27 | 59 | No       |
| 2       | 10.29                    | 11.52 | 135   | 71   | 25 | 58 | No       |
| 3       | 17.88                    | 11.90 | 164   | 45   | 17 | 28 | No       |
| 4       | 19.91                    | 10.37 | 116   | 75   | 30 | 67 | No       |
| 5       | 13.70                    | 11.49 | 125   | 77   | 27 | 60 | No       |
| 6       | 9.97                     | 11.78 | 160   | 49   | 19 | 31 | No       |
| 7       | 15.69                    | 10.85 | 145   | 65   | 26 | 55 | No       |
| 8       | 12.85                    | 10.18 | 122   | 80   | 32 | 71 | No       |
| 9       | 10.19                    | 11.39 | 154   | 49   | 25 | 37 | No       |
| 10      | 20.33                    | 11.76 | 158   | 44   | 23 | 33 | No       |
| Average | 14.22                    | 11.28 | 140.5 | 63.3 | 25.1 | 49.9 | No |

Table 8: Result Automated Schedule - Stochastic Model

6.2.3 Improvement Algorithm
The improvement algorithm developed in chapter 5 was applied on the automated schedules produced above. The results when applied on the deterministic and stochastic model are shown in table 9 and 10 respectively. The depth used was 10 for the deterministic model and 5 for the stochastic model.

Deterministic Model

| Depth | Execution Time | Swaps | $Z_2$ | $Z_3$ | $|Q|$ | UC | GH | Feasible |
|-------|----------------|-------|-------|-------|------|----|----|----------|
| 10    | 0.64 seconds   | 3     | 12.37 | 204   | 6   | 9  | 1  | No       |

Table 9: Result Deterministic Model - After Improvement

Stochastic Model

| Runtime | Depth | Execution Time | Swaps | $Z_2$ | $Z_3$ | $|Q|$ | UC | GH | Feasible |
|---------|-------|----------------|-------|-------|-------|------|----|----|----------|
| 1       | 5     | 99 seconds     | 6     | 11.66 | 132   | 72   | 27 | 54 | No       |
| 2       | 5     | 107 seconds    | 2     | 11.60 | 137   | 69   | 25 | 58 | No       |
| 3       | 5     | 86 seconds     | 5     | 12.06 | 169   | 41   | 17 | 26 | No       |
| 4       | 5     | 149 seconds    | 4     | 11.26 | 120   | 75   | 30 | 65 | No       |
| 5       | 5     | 123 seconds    | 5     | 11.66 | 130   | 73   | 27 | 57 | No       |
| 6       | 5     | 181 seconds    | 3     | 11.93 | 163   | 48   | 19 | 30 | No       |
| 7       | 5     | 75 seconds     | 6     | 10.98 | 151   | 63   | 26 | 54 | No       |
| 8       | 5     | 57 seconds     | 4     | 11.19 | 126   | 79   | 32 | 69 | No       |
| 9       | 5     | 128 seconds    | 7     | 11.68 | 161   | 44   | 25 | 34 | No       |
| 10      | 5     | 113 seconds    | 5     | 12.01 | 163   | 42   | 23 | 29 | No       |
| Average | ...   | 112 seconds    | 4.7   | 11.60 | 145.2  | 60.6 | 25.1 | 47.6 | No |

Table 10: Result Stochastic Model - After Improvement
7 Discussion

In this chapter the results from the case study in the former chapter are discussed. Also some advantages and limitations of the model and solution procedure are reviewed. The chapter ends with some final thoughts of further research and how the thesis could be further improved and developed.

7.1 Case Study Results

When reflecting back on section 1.3 and the purpose of the thesis; the main purpose of this thesis is to develop an (heuristic) algorithm that automates the scheduling process at a typical tennis club, with the aim to create a feasible schedule for the tennis school, and analyzing the results from the padel club case it is obvious that the word "feasible" is the main issue. None of the schedules created by the algorithm developed were feasible, not even after the improvement algorithm had been applied. The schedule from the deterministic model was close to being feasible but even if it was only 6 players left in the queue after the improvement algorithm had been applied one of those 6 had not gotten its guaranteed hours satisfied. The guaranteed hours was not only a problem for the schedules created in this thesis, but also for the head of scheduling at the padel club. As shown in Table 6 and Appendix 10.5 the manually created scheduled that is currently in use is not feasible either, in terms of the constraints concerning the guaranteed hours. A question then arises: can a schedule be usable even though it is not feasible?. The answer is obviously "yes" and the proof of that is that such a schedule is being used at the padel club. The schedules created by the algorithms were also discussed with the head of scheduling and as he thought many of them were usable it was concluded that the guaranteed hours constrains, that have been modelled as hard constraints in this model, could maybe have been modelled as soft constraints to make the model more realistic.

The biggest advantage of the model and solution procedure created in this thesis is, as could have been expected, the time it saves to automate the scheduling procedure instead of having a person manually create it. As this is the biggest issue, when it comes to scheduling, for the tennis clubs in general and Court1 in particular (Marescotti, 2018)(Hjelmgren, 2018), it seems like the result of this thesis is a step in the right direction.

When comparing the stochastic model to the manually created schedule, the biggest conceivable downside of using the algorithm is that it is more players in the queue and a lower value of $Z_3$ on all occasions. One reason for this could be as simple as that when the members of Court1 replied to the availability survey in Appendix 10.6 they were more restrictive to say when they were available to play than what they are in real life. In real life they realize that "if I do not take this spot I will not get anything else" and so when the head of scheduling calls them and asks if they can reconsider they agree to be scheduled, when they first said that they could not. Another reason is that the model in this thesis focuses more on the players of higher priority, trying to satisfy their desires, than the head of scheduling did when making the schedule. When the solution procedure developed in this thesis tries to satisfy the desires of the highly prioritized, it tries to satisfy the desires of those players to whom the desires are most difficult to satisfy. Since the queue is formed by players that have not gotten their desires met yet, the way to reduce that queue would be to start by scheduling the players that are the easiest to satisfy and that is the opposite to what the prioritization and solution procedure in this thesis does based on the club policy.

Although the schedule from using the deterministic availability model was close to being feasi-
ble, both that one and all the ones from the stochastic model were infeasible. As can be seen in
the result tables in Chapter 6, from the value of $UC$, this is not due to lack of available courts
or coaches. In the deterministic case the schedule becomes infeasible due to a matching problem,
where the algorithm cannot find suitable groups for all the players with guaranteed hours. Since
the algorithm does not take an "all possible solutions approach" it could be the case that there is
in fact a feasible solution to the problem but that the algorithm does not find it. However, it is
easy to think of cases when the problem is not feasible at all. One example would be if there is a
player that have been guaranteed two hours and there are only three other players that the player
matches with, then if two or more of the other players that the player matches with only desire
to play one hour the problem immediately becomes infeasible. In the stochastic case the possible
matching problem remains but it also adds the problem that a player might have many players
that would be fitting to play with but that not enough of them are available at the same time to
form large enough groups.

7.1.1 Implementation

Although the model does not provide the club with a schedule superior to the manually created
one, the advantage in time consumption from using the algorithm to create a usable schedule
makes the club want to continue the development of the algorithm and prepare to implement the
model for the upcoming semester. To do this, a simple and effective process to gather and store
all the necessary data for the coaches and players must be implemented. As the data gathering
is one of the things that makes the manual scheduling so time consuming this process must also
be computerized and the responsibility to provide the club with such data should be transferred
to the coaches and the players to the extent that it is possible. Also it should be investigated if
additional improvement algorithms can be developed to further improve the schedule by applying
them to the first output solution of the PSP algorithm.

7.2 Solution Procedure

The choice to develop a heuristic solution procedure to deal with the TSSP was made based both
on the complexity of the problem and on what was requested by Court1. To understand the
complexity of the problem one could consider a case similar to the one at the padel club, only
considering the PSP: If the club has 200 players, 3 available courts, 8 hours a day for 5 days this
would yield $200 \times 8 \times 3 \times 5 = 24000$ variables of the binary kind $B_{p,h,d,court}$ when using the same
notation as in this thesis. Include the CSP, with 3 coaches, 3 courts, 5 days and 8 hours and we
have $3 \times 8 \times 3 \times 5 = 360$ more variables to determine. The number of solutions is unreasonably
large and the problem is not fitting to solve using methods that assesses each possible solution.
Furthermore, what Court1 thought would be most beneficial to them was not to necessarily create
an optimal schedule but to automate the process and produce a schedule that is "good enough"
(Hjelmgren, 2018). This is in line with what was pointed out by Hillier and Lieberman (2015) in
section 2.3, that in practise satisficing is sometimes more useful than optimizing. Therefore, in
order to deal with the complexity of the TSSP and to approach the problem in a practical manner
the choice to develop a heuristic algorithm to produce a solution to the problem was made.

7.2.1 Purpose

When turning back to the purpose of the thesis in Section 1.3 it is recognized that a solution proce-
dure to automate the scheduling process at a typical tennis club has been developed, an algorithm
that improves this schedule on all test occasions has been developed and these algorithms have been
tested on real data and the results have been compared with the results from the process that is currently used at a padel club. However, the aim with the algorithms was that they would produce a feasible schedule to the problem which they have not done when testing them on the real data in the padel club case. It is easy to understand that the problem can be infeasible with certain data, for example a player that is guaranteed hours but never available when the players it is compatible with are available will not get its hours and so this will give an infeasible solution. However, there are also flaws in the solution procedure such that the problem might be feasible but the algorithm does not yield a feasible solution anyway. This is actually the case in the deterministic model in the padel club case and an instance of the issue in that case can be exemplified in the following way: Consider the priority ordered players \( \{p_1, p_2, \ldots, p_6\} \), all compatible to play with each other and all have got guaranteed hours. Further assume that the maximum number of players in a group are 4 and the minimum amount is 3. The solution procedure developed would then produce the solution \( g = \{p_1, \ldots, p_4\} \) and \( Q = \{p_5, p_6\} \) and the improvement algorithm can not do anything about this since it can not move any of the players in \( g \) to another group. So if the players are all available all the time as in the deterministic model a feasible solution to the problem would be \( Q = \{\}, g_1 = \{p_1, p_2, p_3\}, g_2 = \{p_4, p_5, p_6\} \). One of the rules of thumb for the improvement algorithm was that the number of groups should not be increased. A positive property of this is that it aims to fill the groups further without increasing the work load for the coaches; however, for the obvious reason of increasing the chance of a feasible solutions it should be further investigated if an improvement algorithm that can split up and create more groups in order to create additional spots for players in the queue can be developed.

A relevant thought when analyzing the results from the padel club case is that maybe the purpose should have been formulated differently with the aim of the algorithms being to create a usable schedule rather than a feasible schedule since the schedule that Court1 uses is not deemed feasible according to the mathematical formulation of the problem.

7.3 Mathematical Model

When discussing the results with the owner and the head of scheduling at Court1 some concerns with the objective function of the PSP arises. The concern is that it might not value to schedule a low priority player enough and so this might result in a lower value of the sum of total booked hours for the players, \( Z_3 \). This would significantly lower the income for the club since all players pays the same amount to participate. A suggestion might be to investigate if the objective function could be to maximize \( Z_3 \) and add a less severe penalizing factor to the function to deal with the prioritization of the players.

There are some complicated constraints in the model concerning the minimum work shift for the coaches, the free periods for coaches and the free periods for the players, these are represented in equations 4.10, 4.11 and 4.18 respectively. These constraint are difficult to formulate if the problem is to be solved using some optimization software. A suggestion might be to try and relax those constraints and to reformulate the objective functions so that the existence of too short work shifts and free periods in the schedules are penalized rather than fixed constraints.

7.4 Further Research

A delimitation in this thesis was that the problem would not be solved using some existing optimization software. With greater insight and understanding for the problem this could be an
interesting area to further investigate. The problem is as pointed out before complex and large; however, it is also very sparse and so it might be reason to look into some sparse optimization theory and methods and see if that could render some additional ideas on how to go about with the modelling. It is also possible that if simplifying the mathematical formulation of the problem and solving it using an optimization software this could produce a good starting solution to the more complex problem which could then be tried to improve.

Two other interesting areas, if proceeding with the heuristic and meta heuristic approach, that could give further theoretical depth to this research could be clique problems and stable marriage problems. If able to formulate the TSSP and its’ subproblems CSP and PSP in a way such that any already existing meta heuristic method could be applied this would most definitely add theoretical value.
8 Conclusion

This thesis has investigated the scheduling process of the tennis school at a general tennis club, with the aim to formulate it as a mathematical model and develop a solution procedure such that the scheduling can be automated. The solution procedure, an heuristic algorithm, was then tested on real data at a padel club and rendered arguably satisfying results. As no case that handled the same problem could be found, this thesis should be seen as a first step to get deeper insight and understanding for the specific problem such that better models and solution procedures can be developed in the future. The greatest benefit from using the model and solution procedure developed in this thesis is that it saves time compared with when manually creating the schedule and the groups. Although there are certainly many parts handled in this thesis that could be further researched, it can be concluded that a computerized solution procedure that automates the scheduling process of the tennis schools would definitely be of value.
9 References


Giovanni Righini. Introduction heuristic algorithms.


Samuel Schwieler. Interview in person, 2018.

10 Appendices

10.1 Interview - Owner Court 1

**Interviewers:** Johan Krylstedt and Max Lövenheim  
**Interviewee:** Owner at Court1 Padel Club  
**Interview setting:** At the premises of Court1 Padel Club the 5th of November 2018

**Q:** To start off, can you describe what are the main things that you want our help with?

**A:** Mainly we feel that the scheduling of our padel school is not efficient enough. Even now when we have recently started it can take the person responsible for scheduling 2 weeks to make the schedule.

**Q:** What is the reason that it takes so much time would you say?

**A:** Well I have worked before with scheduling myself and I would say that it is due to that it is so many things to consider such as when the student is available, the parameters that makes it fit in a group, whether it can be matched with a good enough coach and so on. Every time you change one thing in the schedule you automatically change it in many other places as well. Also it is done manually which is what we want to get away from.

**Q:** What would you say that the main objective is for you? Is it to create better matching groups? To fill the school with more players? Or anything else?

**A:** First of all I would say that the objective is to automate the scheduling. To save the time that it takes to create the schedule today is more important than to get better matching groups or to fill the school with even more players.

**Q:** And if you have to choose between getting better matching groups or to fill the school with more players, what would be more important?

**A:** Of course the players in the groups will have to be an okay match but if it comes to getting "perfect" groups but then a few players will not get their spot or if the groups are okay but then many more player will get a spot it is preferable to fill the school with more players.

**Q:** Does the schedule of the players and the schedule of the coaches affect each other?

**A:** Yes.. how shall I put it? The coach schedule affects the player schedule in the way that not all coaches have the competence to coach all levels and all categories. So a player is not scheduled on a court where it does not match the scheduled coach.

**Q:** Okay, so the schedule for the coaches is done before you schedule the players?

**A:** Yes. Or at least that is how I did it before. I think it is still done that way.

**Q:** Okay, we will ask your colleague about how he do it now also. But is the coach schedule...
prioritized higher than that of the players?

A: Well.. the demand to play is very high and it is really difficult to find good and reliable coaches, so I would say yes. It is more important that our coaches are satisfied with their schedule than that the students get all their wishes fulfilled.
10.2 Interview - Head Coach Court 1

**Interviewers:** Johan Krylstedt and Max Lövenheim  
**Interviewee:** Head Coach, responsible for scheduling  
**Interview setting:** At the premises of Court1 Padel Club the 12th of November 2018

**Q:** When you create the schedule for the padel school, where do you start?

**A:** start by setting the coaches hours when they are available so that I get a good view of what hours I have to work with when assembling the groups.

**Q:** Do you start with a particular coach or can you start with anybody?

**A:** I think I can start with anybody...

**Q:** So they are not prioritized in different ways?

**A:** Well maybe.. I mean there are coaches that we want because they are very good and that only can work certain hours so I think, yeah, I schedule them first.

**Q:** Okay so the coaches that have the highest quality and that can only take a few hours are scheduled first to ensure that they get to work and to ensure that you have a coach that can take high level players?

**A:** Yes!

**Q:** When you schedule the coaches do you try to schedule them one day at a time or do you have any other system?

**A:** Well I start of by scheduling the prioritized coaches, as I said, the days and hours it has to get. After that, I start from the first day of the week and try to schedule coaches on all our available courts.

**Q:** Okay and can you schedule a coach on hour just to fill it and then it does not have any more hours that day?

**A:** Well, that depends on the coach. It is never desireable I would say. But me and NN can work in the cafeteria or clean or do other work. But NN2 cannot do that.

**Q:** Why is that?

**A:** Well because if you are paid by the hour then we want you to work on the court or not at all. We cannot have people paid by the hour and then they get a free period or they come in to only work one hour.

**Q:** Okay, so for consultants and coaches paid by the hour, no free periods between the work shifts and not to short work shifts?
A: Exactly! Although it can be special cases but then we have to make a special deal or agreement.

Q: So then you fill the scheduling, trying to avoid free periods and to short shifts. Are there maximum shifts aswell?

A: Not really. We usually have lessons for a maximum of 3 hours in the morning and then max 6 hours in the afternoon so if you work both morning and evening that will be 9 hours and then you can work less another day.

Q: Okay then, lets talk about when you schedule the players then. How do you start off?

A: Well I start off with the player list and then I Start making the most obvious groups that I know will match.

Q: How do you know they will match?

A: Because I know what age they are, I know their level and I know if they are friends and so on.

Q: Okay so you have that information in a spreadsheet?

A: Not right now. I keep it in my head. But I starting to organize it in a spreadsheet now for the future.

Q: So if you quit how would it be done by the next person?

A: That is exactly what happened when I got this responsibility... The person that was responsible before had no system so I had to create my own while learning everything quick about the players.

Q: Okay but you match them according to age and level right?

A: Yeah, and a little bit on how they are socially as well.

Q: And the player list, is it in any prioritized order or something or which player do you start with?

A: Well you can say that, or as of now I just do it from my head and begin with the players that also were in the school last semester and if they should have more than one hour and so on.

Q: Okay, so you are prioritized if you played the semester before?

A: Yes!

Q: And the players, can they be guaranteed a spot or guaranteed multiple hours?

A: Yes, if you played last semester you are guaranteed a spot. Of course if the player were not to be available the hours we suggest then it can happen that you do not get a spot anyway but
you are always given a suggested spot from us. Also, if you have been in line for an extra hour for long you can after a while be guaranteed multiple hours and we never take away hours from one semester to another.

**Q:** So you prioritize based on if it is a previous player in the school, how many guaranteed hours it is supposed to have. Anything else?

**A:** I do not know, maybe. Some are just prioritized higher because they have been in the queue longer.

**Q:** So first come, first served?

**A:** Yes, exactly!

**Q:** How do you know if they are able to play the time that you are assigning them?

**A:** Well, some of them you have spoken to in advance so you know approximately which days and hours they can play but others you give a time and hope that they can take it.

**Q:** And if they can not?

**A:** Then I have to rearrange a bit.

**Q:** Okay but when you make your judgement on how good they match. How do you rank their level for example? And how much can they differ in level or age?

**A:** I have a ranking system from 1 to 4 where 4 is the best level. How much they can differ...? Well one level maximum I would say. In age that depends on how old they are...

**Q:** Do you divide them in categories depending if they are grownups or children?

**A:** Yes, we have two categories "Juniors" and "Adults".

**Q:** So how much can they differ in age in the junior category?

**A:** One year maybe. Well if there is for example a 12 year old of level 4 maybe he can play with 14 year olds of level 3...

**Q:** Okay so the allowed age difference depends on the level and vice versa?

**A:** Yes. But for the juniors age is more important than level but both are important.

**Q:** I see! So for the adults, what can the age and level difference be?

**A:** For the adults the level is more important than age but if you are 60 years old and are going to play with a 30 year old you can maybe not be a level lower also. **Q:** Could you say that for juniors you have to be better or equal than your fellow group members if you are younger and for
adults you have to be better or equal if your older? A: Yes, that would be about right.

Q: Ok, lets move on to the groups. How many players can it be in each group?

A: For the juniors it is maximum 6 and for the adults it is 4.

Q: Do they have to be completely full for you to schedule them?

A: No, not really... that depends on whether I think I can fill it later during the semester.

Q: Okay, but even if you know that you will not be able to fill a group can you schedule them anyway?

A: Yeah, if it is full enough. I would say that adults can be 3 throughout the semester and juniors can be 4-5. But usually I can start adult groups with 2 or sometimes even 1 person because I have a feeling that I will be able to fill it later. I almost always start a group with 4 juniors, sometimes even 3.
10.3 Interview - Standard Tennis Club Schedule and Organization

**Interviewers:** Johan Krylstedt and Max Lövenheim  
**Interviewee:** Former Head of Gustavsbergs Tennis Club  
**Interview setting:** Restaurant Stockholm 30th of October 2018

Q: So I am going to begin with some questions about how the organization that is involved with the tennis school looks like, some data about what the club looked like and so on. So to begin, how many courts were there?

A: It was 4 courts.

Q: And how many players participated in the tennis school?

A: It varied but was about 600

Q: Would say that, that is a lot compared to other clubs with the same size and number of courts?

A: Yes, it is one of the largest clubs in Sweden if measuring members per court.

Q: And are there tennis school classes everyday?

A: Not on Sundays but otherwise yes.

Q: Okay, so how would the schedule look like a normal week?

A: On an ordinary weekday for a tennis school there are classes between 07.00-09.00 and 15.00-21.00. Between 07.00-09.00 the students are adults or competitive juniors practising before work/school then from 15, the youngest juniors start their practise followed by older juniors, competitive players and later from 19.00-21.00 it is adult students again. On Saturdays the classes starts at 09.00 for kids under 7 years old and continues until 12.00, then from 12.00-16.00 it is classes for juniors.

Q: As you describe it now there are many different categories of players. How do you categorize them?

A: We have 4 main categories with Adults, Juniors, Kids and Competetive players. Then juniors are divided in three categories ranging from the youngest that are about 7 years old to the oldest that are 18. We call the categories orange, green, and yellow depending on which level of ball they are playing with because there are different balls that differ in difficulty playing with.

The competitions category is also divided in three sub-categories depending on age, level and commitment. We call them Future, Challenger and Grand Slam.

Q: Okay, and from your previous working experience would you say that it is common to divide the players in such categories?
A: Yes, I would say that that is the standard and you almost have to do it to have some kind of structure in the tennis club. The names can of course be different but the main structure is the same.

Q: What is the difference between the categories more than age and level and so on?

A: Well, the different categories can get different amount of hours and the competition categories are guaranteed a certain amount of hours. Other than that I would just say that the categories can play different times. Some of them are allowed to be scheduled the same time but for example we do not schedule juniors after 19 and adults not before 19 and there are a couple of other such specifications that I think differ a bit from club to club. Then the group sizes varies in our categories. In some category for the youngest players they can be 8 players in each group and for the adults and competitive players it is a maximum of 4 players per group.

Q: Okay, so lets move on to the coaches. Are all coaches employed full time?

A: No, we have only two coaches that are employed full time, then we have some part time employees, some consultants and some coaches that are paid by the hour.

Q: Can all your coaches coach every category or are there different levels among the coaches as well?

A: All coaches cannot coach every category or level of player. We have some coaches that are still very young and so they only coach kids. Then we also have the highest level coaches that usually have been competing earlier in their career and they coach the competing juniors and the highest level of adults but can also coach players of lower level to get more hours.

Q: Are some of the coaches responsible for the different categories and if so do they have to be on the court every time that category plays?

A: Well, yes.. There are coaches responsible for the different categories but all of them do not have to be scheduled every time their category is playing. However the coaches that are responsible for the competing categories are almost always scheduled when that category plays because they have a bigger responsibility to observe those players. Also on weekends when we have practises for the kid category most of the coaches are very young so the the coach that is responsible have to be schedule to keep an eye on the work that is being done on all courts.

Q: When scheduling the tennis school, in what way is it done? Is it done manually or does the person responsible have a more sofisticated way of doing it?

A: Well it is done by one or two of the coaches that has that responsibility. It is done manually in Excel where they have the schedule from the previous semester as a base for the new schedule.

Q: Would you say that, that is how it works in most other clubs as well?

A: Yes, or at least I have not heard about it being done in a very different way.
Q: And to schedule the school how long time does it take?

A: Usually at least a couple of weeks but it can take up to a month.

Q: What can be an advantage that a person does the schedule manually instead of for example automate it based on level and age?

A: Well one thing that comes to mind is that such an automization may not take social aspects into consideration. When our person responsible scheduled the players he knew who they where and could match them in groups where he thought they would have a good time socially as well as the matching of the tennis level. However since there were so many students this also is problem since it makes the scheduling even more complex and it was always some flaws in the schedule anyway so I do not know if it is such an advantage to be honest.

Q: When the tennis school is not scheduled, are your courts possible to lease by the hour for any player then?

A: Yes of course. That I would say is the case in every tennis club although some only allow their members to play. Tennis clubs usually have to businesses, one that is just leasing of their courts and one that is the tennis school where you get the opportunity to get coached in a group.
### 10.4 Tennis School and Court Availability

#### Monday

<table>
<thead>
<tr>
<th></th>
<th>Court 1</th>
<th>Court 2</th>
<th>Court 3</th>
<th>Court 4</th>
<th>Court 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>['Kids', 'Junior']</td>
<td>['Kids', 'Junior']</td>
<td>['Kids', 'Junior']</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>['Kids', 'Junior', 'Adult']</td>
<td>['Kids', 'Junior', 'Adult']</td>
<td>['Kids', 'Junior', 'Adult']</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>['Junior', 'Adult']</td>
<td>['Junior', 'Adult']</td>
<td>['Junior', 'Adult']</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>['Adult']</td>
<td>['Adult']</td>
<td>['Adult']</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>['Adult']</td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>['Adult']</td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>['Adult']</td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>['Adult']</td>
</tr>
</tbody>
</table>

Figure 3: Allowed Categories for Tennis School

#### Tuesday

<table>
<thead>
<tr>
<th></th>
<th>Court 1</th>
<th>Court 2</th>
<th>Court 3</th>
<th>Court 4</th>
<th>Court 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td>['Kids', 'Junior']</td>
<td>['Kids', 'Junior']</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>['Kids', 'Junior', 'Adult']</td>
<td>['Kids', 'Junior', 'Adult']</td>
<td>['Kids', 'Junior', 'Adult']</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td>['Junior', 'Adult']</td>
<td>['Junior', 'Adult']</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td>['Adult']</td>
<td>['Adult']</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td>['Adult']</td>
<td>['Adult']</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td>['Adult']</td>
<td>['Adult']</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
<td>['Adult']</td>
<td>['Adult']</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td>['Adult']</td>
<td>['Adult']</td>
<td></td>
</tr>
</tbody>
</table>

Figure 4: Allowed Categories for Tennis School
### Wednesday

<table>
<thead>
<tr>
<th></th>
<th>Court 1</th>
<th>Court 2</th>
<th>Court 3</th>
<th>Court 4</th>
<th>Court 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>8</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>9</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>10</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>11</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>12</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>13</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>14</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>15</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>16</td>
<td>[]</td>
<td>['Kids', 'Junior']</td>
<td>['Kids', 'Junior']</td>
<td>['Kids', 'Junior']</td>
<td>['Kids', 'Junior']</td>
</tr>
<tr>
<td>17</td>
<td>[]</td>
<td>['Kids', 'Junior', 'Adult']</td>
<td>['Kids', 'Junior', 'Adult']</td>
<td>['Kids', 'Junior', 'Adult']</td>
<td>['Kids', 'Junior', 'Adult']</td>
</tr>
<tr>
<td>18</td>
<td>[]</td>
<td>['Junior', 'Adult']</td>
<td>['Junior', 'Adult']</td>
<td>['Junior', 'Adult']</td>
<td>['Junior', 'Adult']</td>
</tr>
<tr>
<td>19</td>
<td>[]</td>
<td>['Adult']</td>
<td>['Adult']</td>
<td>['Adult']</td>
<td>['Adult']</td>
</tr>
<tr>
<td>20</td>
<td>[]</td>
<td>['Adult']</td>
<td>['Adult']</td>
<td>['Adult']</td>
<td>['Adult']</td>
</tr>
<tr>
<td>21</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>22</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>23</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
</tbody>
</table>

Figure 5: Allowed Categories for Tennis School

### Thursday

<table>
<thead>
<tr>
<th></th>
<th>Court 1</th>
<th>Court 2</th>
<th>Court 3</th>
<th>Court 4</th>
<th>Court 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>8</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>9</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>10</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>11</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>12</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>13</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>14</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>15</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>16</td>
<td>[]</td>
<td>['Kids', 'Junior']</td>
<td>['Kids', 'Junior']</td>
<td>['Kids', 'Junior']</td>
<td>['Kids', 'Junior']</td>
</tr>
<tr>
<td>17</td>
<td>[]</td>
<td>['Kids', 'Junior', 'Adult']</td>
<td>['Kids', 'Junior', 'Adult']</td>
<td>['Kids', 'Junior', 'Adult']</td>
<td>['Kids', 'Junior', 'Adult']</td>
</tr>
<tr>
<td>18</td>
<td>[]</td>
<td>['Junior', 'Adult']</td>
<td>['Junior', 'Adult']</td>
<td>['Junior', 'Adult']</td>
<td>['Junior', 'Adult']</td>
</tr>
<tr>
<td>19</td>
<td>[]</td>
<td>['Adult']</td>
<td>['Adult']</td>
<td>['Adult']</td>
<td>['Adult']</td>
</tr>
<tr>
<td>20</td>
<td>[]</td>
<td>['Adult']</td>
<td>[]</td>
<td>[]</td>
<td>['Adult']</td>
</tr>
<tr>
<td>21</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>22</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
<tr>
<td>23</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
<td>[]</td>
</tr>
</tbody>
</table>

Figure 6: Allowed Categories for Tennis School
10.5 Player Data Received by Court 1 Padel Club

<table>
<thead>
<tr>
<th>Name</th>
<th>Age</th>
<th>Category</th>
<th>Level</th>
<th>Previous Player</th>
<th>Priority</th>
<th>Desired hours</th>
<th>Guaranteed Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>13</td>
<td>Junior</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>P2</td>
<td>13</td>
<td>Junior</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>P3</td>
<td>13</td>
<td>Junior</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P4</td>
<td>12</td>
<td>Junior</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P5</td>
<td>9</td>
<td>Junior</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>P6</td>
<td>10</td>
<td>Junior</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P7</td>
<td>11</td>
<td>Junior</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P8</td>
<td>12</td>
<td>Junior</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P9</td>
<td>11</td>
<td>Junior</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P10</td>
<td>13</td>
<td>Junior</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P11</td>
<td>13</td>
<td>Junior</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>P12</td>
<td>11</td>
<td>Junior</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P13</td>
<td>10</td>
<td>Junior</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>P14</td>
<td>13</td>
<td>Junior</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P15</td>
<td>12</td>
<td>Junior</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P16</td>
<td>13</td>
<td>Junior</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P17</td>
<td>13</td>
<td>Junior</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>P18</td>
<td>41</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P19</td>
<td>39</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P20</td>
<td>49</td>
<td>Adult</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P21</td>
<td>43</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P22</td>
<td>42</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P23</td>
<td>46</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P24</td>
<td>49</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P25</td>
<td>55</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Page</td>
<td>ID</td>
<td>Age</td>
<td>Gender</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>------</td>
<td>----</td>
<td>------</td>
<td>--------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>P26</td>
<td>36</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P27</td>
<td>37</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P28</td>
<td>47</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P29</td>
<td>64</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P30</td>
<td>44</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P31</td>
<td>13</td>
<td>Junior</td>
<td></td>
<td>2</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>P32</td>
<td>12</td>
<td>Junior</td>
<td></td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P33</td>
<td>13</td>
<td>Junior</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P34</td>
<td>13</td>
<td>Junior</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P35</td>
<td>13</td>
<td>Junior</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P36</td>
<td>13</td>
<td>Junior</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P37</td>
<td>48</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P38</td>
<td>44</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P39</td>
<td>42</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P40</td>
<td>48</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P41</td>
<td>53</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P42</td>
<td>19</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P43</td>
<td>56</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P44</td>
<td>47</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P45</td>
<td>39</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P46</td>
<td>44</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P47</td>
<td>49</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P48</td>
<td>49</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P49</td>
<td>19</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P50</td>
<td>47</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P51</td>
<td>44</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P52</td>
<td>47</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P53</td>
<td>51</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P54</td>
<td>41</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P55</td>
<td>12</td>
<td>Junior</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P56</td>
<td>12</td>
<td>Junior</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P57</td>
<td>12</td>
<td>Junior</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P58</td>
<td>9</td>
<td>Junior</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P59</td>
<td>10</td>
<td>Junior</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P60</td>
<td>9</td>
<td>Junior</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P61</td>
<td>17</td>
<td>Junior</td>
<td></td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>P62</td>
<td>16</td>
<td>Junior</td>
<td></td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>P63</td>
<td>17</td>
<td>Junior</td>
<td></td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>P64</td>
<td>17</td>
<td>Junior</td>
<td></td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>P65</td>
<td>28</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P66</td>
<td>32</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P67</td>
<td>31</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P68</td>
<td>35</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P69</td>
<td>21</td>
<td>Adult</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>P70</td>
<td>43</td>
<td>Adult</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P71</td>
<td>41</td>
<td>Adult</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P72</td>
<td>42</td>
<td>Adult</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P73</td>
<td>44</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P74</td>
<td>39</td>
<td>Adult</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P75</td>
<td>19</td>
<td>Adult</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P76</td>
<td>29</td>
<td>Adult</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P77</td>
<td>14</td>
<td>Junior</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>P78</td>
<td>14</td>
<td>Junior</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>P79</td>
<td>12</td>
<td>Junior</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>P80</td>
<td>11</td>
<td>Junior</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P81</td>
<td>11</td>
<td>Junior</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P82</td>
<td>11</td>
<td>Junior</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P83</td>
<td>11</td>
<td>Junior</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P84</td>
<td>42</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P85</td>
<td>37</td>
<td>Adult</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P86</td>
<td>42</td>
<td>Adult</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P87</td>
<td>42</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P88</td>
<td>48</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P89</td>
<td>47</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P90</td>
<td>48</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P91</td>
<td>49</td>
<td>Adult</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P92</td>
<td>38</td>
<td>Adult</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P93</td>
<td>40</td>
<td>Adult</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P94</td>
<td>45</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P95</td>
<td>39</td>
<td>Adult</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P96</td>
<td>54</td>
<td>Adult</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P97</td>
<td>52</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P98</td>
<td>42</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P99</td>
<td>43</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P100</td>
<td>48</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P101</td>
<td>53</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P102</td>
<td>53</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P103</td>
<td>51</td>
<td>Adult</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P104</td>
<td>11</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P105</td>
<td>42</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P106</td>
<td>34</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P107</td>
<td>40</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P108</td>
<td>47</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P109</td>
<td>16</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P110</td>
<td>54</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P111</td>
<td>41</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P112</td>
<td>43</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P113</td>
<td>11</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Age</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>----</td>
<td>------</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td></td>
</tr>
<tr>
<td>P114</td>
<td>39</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P115</td>
<td>36</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>P116</td>
<td>33</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P117</td>
<td>59</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P118</td>
<td>48</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P119</td>
<td>57</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P120</td>
<td>43</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P121</td>
<td>45</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P122</td>
<td>43</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P123</td>
<td>45</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P124</td>
<td>55</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P125</td>
<td>36</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P126</td>
<td>57</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P127</td>
<td>45</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P128</td>
<td>58</td>
<td>Adult</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P129</td>
<td>39</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P130</td>
<td>51</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P131</td>
<td>39</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P132</td>
<td>40</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P133</td>
<td>38</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P134</td>
<td>13</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P135</td>
<td>11</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P136</td>
<td>9</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P137</td>
<td>13</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P138</td>
<td>40</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P139</td>
<td>49</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P140</td>
<td>48</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P141</td>
<td>47</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P142</td>
<td>49</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P143</td>
<td>55</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P144</td>
<td>49</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P145</td>
<td>42</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P146</td>
<td>54</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P147</td>
<td>49</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P148</td>
<td>44</td>
<td>Adult</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P149</td>
<td>43</td>
<td>Adult</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P150</td>
<td>44</td>
<td>Adult</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P151</td>
<td>45</td>
<td>Adult</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P152</td>
<td>50</td>
<td>Adult</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P153</td>
<td>51</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P154</td>
<td>39</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P155</td>
<td>44</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P156</td>
<td>46</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>P157</td>
<td>39</td>
<td>Adult</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>Player Id</td>
<td>Age</td>
<td>Gender</td>
<td>Total 1</td>
<td>Total 2</td>
<td>Total 3</td>
<td>Total 4</td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>-----</td>
<td>--------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td></td>
</tr>
<tr>
<td>P158</td>
<td>53</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P159</td>
<td>35</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P160</td>
<td>55</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P161</td>
<td>40</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P162</td>
<td>59</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P163</td>
<td>39</td>
<td>Adult</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P164</td>
<td>23</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P165</td>
<td>49</td>
<td>Adult</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P166</td>
<td>47</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P167</td>
<td>27</td>
<td>Adult</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>P168</td>
<td>37</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P169</td>
<td>47</td>
<td>Adult</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P170</td>
<td>14</td>
<td>Junior</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>P171</td>
<td>14</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P172</td>
<td>13</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P173</td>
<td>11</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P174</td>
<td>10</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P175</td>
<td>10</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>P176</td>
<td>10</td>
<td>Junior</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Player Data
10.6 Availability Survey

The format of the player availability survey can be seen in Figure 8
The frequency of the answers from the availability survey is shown in figure 9

Figure 9: Availability Survey Answers
10.7 Stochastic Model - Distribution

The stochastic model is done based on the replies from the survey which are summarized in Appendix 10.6, Figure 9 and can be viewed in full in ???. The probabilities of being available on certain hours are; however, not calculated exactly since it would require an extensive conditional probability analysis that would be beyond the scope of this project. Instead some general observations of the replies of the members are made to make out the most common availability possibilities and some approximate probabilities to be assigned each possibility. Since the padel club has chosen not to have padel school in the weekends those possibilities are omitted. The general observations being:

- it is unlikely to be available in the mornings, but if you are the hours 7 and 8 are most popular with approximately 1/6 of the members being available.
- it is unlikely that you are available only one hour and nothing more on a day and very unlikely that you are available one hour, not available the next and then available again.
- as a junior you are very unlikely to be available in the mornings and after 19. The most popular hours as a junior are 16 and 17.
- as an adult you are unlikely to be available before 17 in the afternoons. The most popular hours being 18 and 19.
- it is likely that if you are available on the unpopular hours you are also available on the popular hours.

Based on these observations the following availability possibilities for the juniors (JAP) and the adults (AAP) are given by equation 10.1 and 10.2 respectively.

\[
JAP = \{\emptyset, \{16, 17\}, \{16, 17, 18\}, \{16, 17, 18, 19\}, \{17, 18, 19\}\} \tag{10.1}
\]

\[
AAP = \{\emptyset, \{7, 8\}, \{7, 8, 16, 17, 18, 19\}, \{16, 17, 18, 19\}, \{17, 18, 19\}, \{18, 19\}, \{19\}\} \tag{10.2}
\]

The approximated probabilities associated with each availability outcome are given in equations 10.3 and 10.4 for the juniors and adults respectively.

\[
P(JAP_i) = \{1/5, 2/5, 1/5, 1/10, 1/10\}, \text{ where } i = 1, \ldots, 5 \tag{10.3}
\]

\[
P(AAP_i) = \{1/8, 1/16, 1/16, 1/16, 1/8, 3/8, 1/8, 1/16\}, \text{ where } i = 1, \ldots, 8 \tag{10.4}
\]