# **Software Documentation: An Automatic Game Designer**

Die nachfolgende Softwaredokumentation beschreibt Zweck, Aufbau und Funktionsweise des Automatic Game Designers (AGD). Sie ist im Zusammenhang mit der Gesamtdokumentation des Experiments zur Bestimmung des strategischen Entscheidungsverhaltens unter Zeitdruck – durchgeführt im Zeitraum Oktober/November 2014 – entstanden. An dieser Stelle sei zum Zwecke der weiterführenden Information auf die Gesamtdokumentation und die in diesem Zusammenhang entstandene wissenschaftliche Arbeit mit dem Arbeitstitel „Decision Making under Time Pressure – a process-based analysis approach“ verwiesen.

### Zweck

Der AGD unterstützt die Erstellung von Experimentdateien, indem er Inhalte gemäß den Vorgaben des Experimenteurs modifiziert. Das Experiment umfasst drei Design-Versionen, welche online verwendet wurden. Die zugehörigen Web-Pages des Experiment-Anteils sind mit dem AGD erstellt.

Das vornehmliche Ziel des Experiments ist die Gewinnung möglichst umfassender Daten zum Verhalten des Spielers im Entscheidungsprozess, die dann Rückschluss auf den mentalen Strategiewahlprozess geben können. Dazu gehören explizit Entscheidungsgeschwindigkeit, Strategiewahl, Reihenfolge der erfassten Informationen, Reihenfolge der Informationsverarbeitung und wünschenswerterweise der Eliminations- und Auswahlprozess. Hier bietet sich eine automatisierte Dokumentation des beobachtbaren Verhaltens vom Moment der Präsentation der Problemstellung bis zum Moment der Strategiewahl an.

Eine Software-gestützte Entwicklungsumgebung für Experimente der Verhaltenspsychologie ist MouseLabWeb. Als Gemeinschaftsprojekt der Verhaltenswissenschaftler Johnson und M. Willemsen ist es mit einer freien Lizenz (GNU-licensed) erhältlich. Durch ein Baukastenprinzip lassen sich Experimente flexibel selbst gestalten. Die geforderten Indikatoren, die Rückschlüsse auf den Entscheidungsprozess ziehen lassen, können bei MouseLabWeb integriert werden. Eine Designer-Oberfläche bietet, neben Standardangaben zum Projekt, die Möglichkeit, die Auszahlungsmatrix in Größe, Form und Inhalt nach den Bedürfnissen des Experiments zu gestalten. Weitere Stellschrauben sind Anzeigemodus der Matrixelemente, Anzeigedauer, Anzeigereihenfolge der Strategien und die (Art der) Strategieauswahl.

### Aufbau und Funktionsweise

Das eigentliche Experiment ist zusammengesetzt aus Experimentblöcken. Diese wiederum bestehen aus einer festzulegenden Anzahl an Einzelproblemen („Spielen“). Nachstehende Abbildung soll den Testablauf verdeutlichen. Für einen Experimentblock In Tabelle 1 sind die Parameter samt Einstellbereiche angegeben.

Die Experimentdateien von MouseLab lassen sich als \*.php-Dateien abspeichern und damit im Browser anzeigen. Dieses Dateiformat ist als Textdatei aufruf- und manipulierbar. Diese Eigenschaft wird zur automatischen Experimenterstellung ausgenutzt.

|  |  |
| --- | --- |
|  |  |

Abbildung 1 a) Bsp. Experimentblock im Detail b) Experimentablauf im Schema

#### Anforderungen an die Prozedur

Das Ziel der Prozedur ist die Erstellung der nötigen \*.php-Dateien eines Experimentabschnittes, wo die Einflussparameter fest sind, aber der Matrixinhalt in erster Instanz variiert. Die Prozedur ist in VBA (Visual Basic for Aplications) programmiert und läuft im Hintergrund einer EXCEL-Maske. In der EXCEL-Maske kann der Experimentator die Parametereinstellungen vornehmen und (per Knopfdruck) die Prozedur starten. Die Dateien werden dann automatisch an einem vordefinierten Speicherort abgelegt. In Abhängigkeit der statistischen Ansprüche an die Modellprüfung und damit an die Datengüte des Experiments werden die Anzahl von Studenten, Experimenten und zu erstellende \*.php-Dateien pro Parameterkonstellation ausgewählt. <Verweis auf statistische Berechnung der variablen Anzahlen> Der Experimentator stellt die Dateien zu Experimenten zusammen und lädt sie an entsprechender Stelle auf einem Server hoch. Per Link starten die Probanden das Experiment im Browser.

#### Die Module der Prozedur

Für die Initialisierung der Prozedur wird vom Experimentator auf der EXCEL-Eingabemaske folgender Input benötigt:

* Grundspiel (Chicken-Game, Falke-Taube-Spiel, Gefangenen-Dilemma, Stackelberg’s Leadership und Throwing Hands)
* Anzahl der Spieler (1,2)
* Anzahl der Strategien pro Spieler (2,…,5)
* Zeitdauer pro Experiment
* Anzahl der Experimentdateien pro Parameterkonstellation (1,…,256)
* weitere Parameter bei den Grundspielen Falke-Taube-Spiel (z, v) und Gefangenen-Dilemma (R, S, T)

Die Grundspiele sind in Form einer 5x5-Matrix für 2 Spieler in dem ersten Tabellenblatt („Grundspiele“) der berechnenden EXCEL-Datei abgelegt. 5x5 ist die maximal geplante Größe der Matrizen. Jedes größere Limit ist technisch abbildbar. Die Aufgaben für die Probanden verkomplizieren sich jedoch mit der Matrixgröße. Das wirkt sich mindestens unmittelbar auf die Zeit aus, die die Probanden für die Lösung eines Problems benötigen. Im schlimmsten Fall ist das Problem für die Probanden nicht lösbar. (, was u.U. die Motivation senkt, weitere Aufgaben zu bearbeiten.)

Bei Start der Prozedur muss eine geeignete Modelldatei im Zielverzeichnis der Experimentdateien abgelegt sein. Die Modelldateien bilden die Vorlage für die Experimentdateien. Sie werden dabei durch die Prozedur gemäß der Parameterkonstellation abgeändert und unter einem neuen Namen am Speicherort abgespeichert. Erstellt werden die Modelldateien mit der MouseLab-Designer-Funktion. Pro Matrixgröße wird eine Modelldatei benötigt.

Die Prozedur besteht aus einer Hauptroutine und drei ausgelagerten Routinen. In der Hauptroutine werden die drei ausgelagerten Routinen in einer verschachtelten For-Schleife eingebunden. Die ausgelagerten Routinen

* berechnen die Daten der Matrix („Matrix\_erstellen“),
* erstellen die Dateien („txt\_TempFileName“) und
* schreiben die neuen Informationen in die Dateien („write\_Line“).

Die For-Schleife besitzt drei Verschachtelungen. Die äußere For-Schleife ist für die Anzahl Dateien, die mittlere für die Anzahl der zu erstellenden Zeilen und die innere für die Erstellung einer Modular aufgebauten Zeile.

Die Komplexität wird bestimmt durch die Komplexität der ausgelagerten Routinen und ihrem Aufruf in der Hauptroutine. So ergeben sich folgende für die Routinen folgende Komplexitätswerte in den Werten der verwendeten Variabeln (O: Landau-Symbol; Maß für die Komplexität eines Algorithmus‘ in Rechenschritten):

* O(Matrix\_erstellen) = rows X cons
* O(txt\_TempFileName) = 1
* O(write\_Line) = Anz. Zeilen im Basisdokument
* O(Experiment\_erstellen) = Anz\_Dateien X rows X O(write\_Line) = Anz\_Dateien X rows X Anz. Zeilen im Basisdokument

Die Anzahl Dateien und die Anzahl Zeilen werden durch den Experimentator festgelegt. Die Zeilen im Basisdokument sind abhängig von der Matrixgröße und eventuell zusätzlichen Angaben zum Experiment.