

Gewinnen beim Monopoly Spiel – Alles nur Zufall? Oder gibt es doch ein paar Muster, die man kennen sollte?

Gerhard Svolba
SAS Austria
Mariahilfer Straße 116
A-1070 Wien
Sastools.by.gerhard@gmx.net

Zusammenfassung

Das Monopoly Brettspiel erfreut sich nach vielen Jahrzehnten nach wie vor hoher Beliebtheit. Das Spiel verknüpft Zufallsereignisse mit der Notwendigkeit strategischer Entscheidungen und beinhaltet eine Fülle von Handlungsanweisungen und Vorgaben in Form von Regeln. Das Monopoly Spiel ist aber auch ein hervorragendes Beispiel für die Anwendungsmöglichkeit von Monte Carlo Simulationen. So können Fragen wie die Verteilung der Besuchshäufigkeit auf den unterschiedlichen Feldern, die Profitabilität und das Risiko verschiedener Grundstücke oder die dynamische Veränderung bestimmter Zusammenhänge während des Spiels simuliert und erkannt werden.

Der Beitrag schließt an den KSFE 2016 Beitrag "Simulationen und Mathematische Programmierung mit SAS" an und zeigt, wie ein SAS Datastep verwendet werden kann um eine umfangreiche Simulationsstudie für die Besuchshäufigkeiten sowie die Einnahmen und Ausgaben pro Spieler und pro Feld aufzubauen. Im Vortrag sind Tipps und Tricks enthalten, wie ein SAS Datastep mit SAS Formaten, Arrays, Funktionen, Zufallszahl-Generatoren oder Do-Loops programmiert werden kann, um komplexe Systeme zu simulieren.

Die Erkenntnisse dieses Beitrags sind nicht nur auf das Monopoly Spiel limitiert sondern können für die Simulation vieler praktischer Aufgaben verwendet werden.

Schlüsselwörter: Monte Carlo Simulationen, komplexe Systeme, SAS Arrays

1 Einleitung und Fragestellung

Manche Zusammenhänge in komplexen Systemen können nicht auf einfache Weise in mathematischen Formeln umgesetzt werden. Hier bietet sich die Durchführung einer Monte Carlo Simulation an.

Das Paper „Simulationen und Mathematische Programmierung mit SAS“ von der KSFE 2016 [2] beschreibt einige Möglichkeiten der Simulation mit SAS. In diesem Paper wird ein konkretes Beispiel auf Basis des Monopoly Brettspiels gezeigt. In dieser Simulation wird die Monopoly Deluxe Version verwendet, diese hat im Gegensatz zum klassischen Monopoly Spiel 52 statt 40 Felder. Die Simulation des Monopoly Brettspiels ist detailliert mit SAS Code in der Case Study 8 des Buchs „Applying Data Science – Business Case Studies Using SAS“ [1] beschrieben.

2 Das Monopoly Spiel als Basis für komplexe Simulationen

Das Monopoly Brett Spiel kann als komplexes System betrachtet werden.

- Das Spielbrett stellt ein Rahmenwerk für Möglichkeiten und Ereignisse dar.
- Es gibt Regeln für die Abläufe und möglichen Aktionen in der Spielanleitung.
- Es gibt Karten mit Anweisungen die Einfluss auf den Spielablauf haben.
- Die beiden Würfel stellen eine zufällige Komponente dar.
- Der Speed-Würfel stellt eine dynamische Komponente dar, sein Einfluss auf das Spiel ändert sich im Spielablauf.
- Die Grundstücke sind durch Kosten und Gewinne charakterisiert und stellen somit eine monetäre Komponente dar.

Die Verteilung der Besuchshäufigkeit auf den einzelnen Feldern des Brettspiels sowie die Profitabilität der einzelnen Grundstücke kann schwer nur durch alleiniges Studium der Spielregeln beschrieben werden. Simulationen helfen hier ein Bild über die Zusammenhänge und Verteilungen zu erhalten.



Abbildung 1: Überblick über das Monopoly Spiel

3 Verteilung der Besuchshäufigkeit auf den Feldern – Simulationsergebnisse

Die Verteilung der Besuchshäufigkeit am Monopoly Spielbrett ist nicht gleichverteilt. Abhängig von der Anzahl der Einflussfaktoren, die berücksichtigt werden, wie Ereigniskarten oder „Gehe-ins-Gefängnis“ Anweisungen, ändert sich die Verteilung markant.

3.1 Nur Würfeln - Fast eine Gleichverteilung

Berücksichtigt man keine der Anweisungen, so ist die Verteilung auf die 52 Felder fast gleichverteilt (siehe Abbildung 2). Es ist ein leichter Buckel auf den Feldern 6-10 zu sehen. Dieser erklärt sich aus dem Fakt, das ein Spiel für alle Spieler immer am Feld 1 beginnt und somit bei der Würfelsumme aus 2 Würfeln die Werte 5,6,7,8,9 häufiger vorkommen als die Werte 2,3,4 oder 10,11,12.

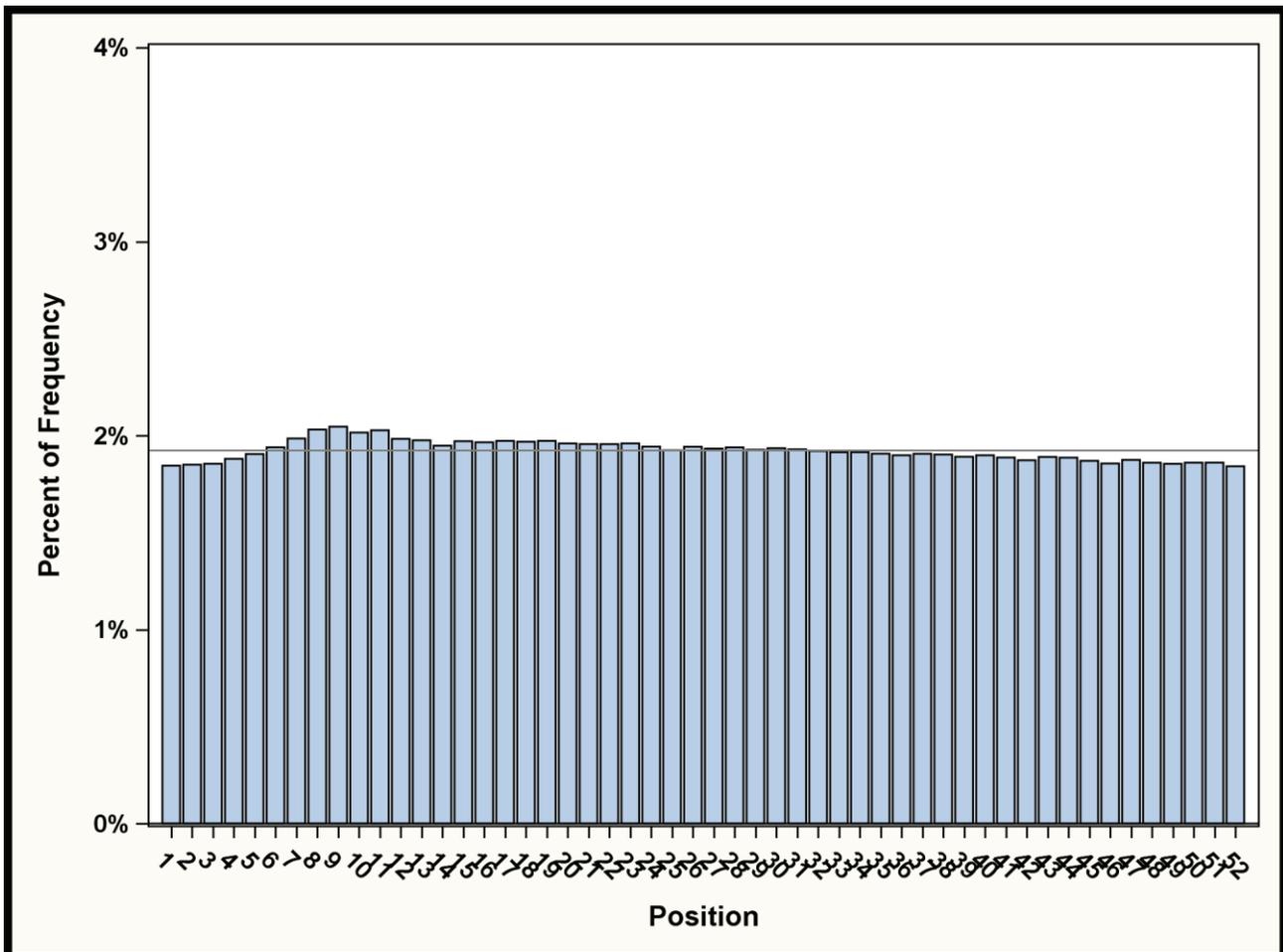


Abbildung 2: Besuchshäufigkeit bei alleiniger Berücksichtigung der Würfel

3.2 Berücksichtigung der „Gehe ins Gefängnis“ Anweisung

Berücksichtigt man die „Gehe ins Gefängnis“ Anweisung, so sieht man die Besuchshäufigkeit auf dem Feld 40, das diese Anweisung enthält, auf 0 sinkt. Gleichzeitig erhöht sich die Besuchshäufigkeit auf dem Feld 13 (Gefängnis).

Eine weitere Konsequenz dieser Anweisung ist, dass höhere Besuchshäufigkeit auf dem Feld 13 zu einer höheren Besuchshäufigkeit auf den Feldern 18-22 führt, weil vom Feld 13 häufiger losgestartet wird. Gleichzeitig ist die Besuchshäufigkeit auf den Feldern 46-51 geringer, weil vom Feld 40 nie losgestartet wird und somit hier Besuchshäufigkeiten fehlen.

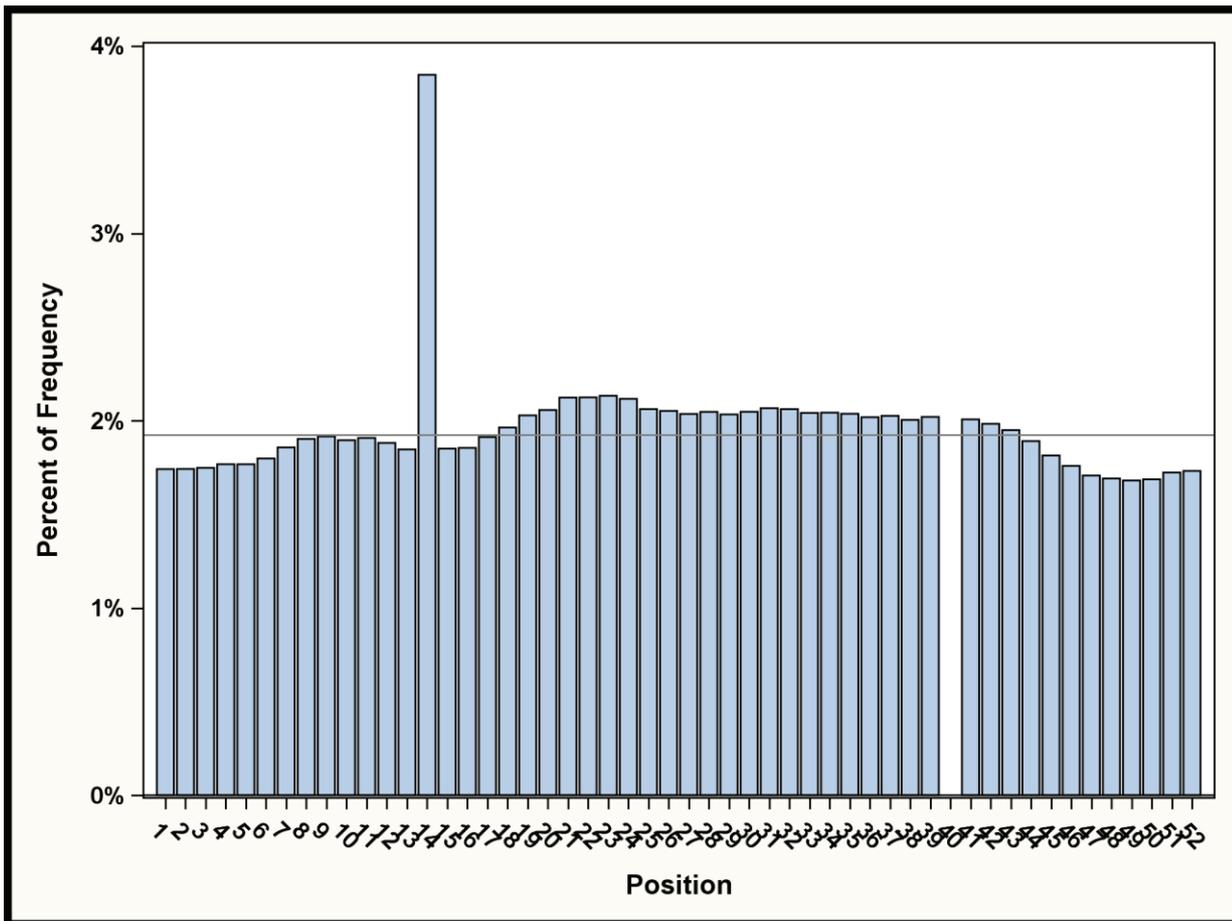


Abbildung 3: Besuchshäufigkeit bei Berücksichtigung des „Gefängnisses“

3.3 Berücksichtigung der Ereigniskarten

Berücksichtigt man auch die Ereigniskarten, wie in Abbildung 4 gezeigt, ergibt sich bereits ein deutlich unterschiedlicheres Bild. Ereigniskarten setzen in manchen Fällen die Figur des Spielers um und führen so zu einer niedrigeren Besuchshäufigkeit auf diesen Feldern und zu einer gestiegenen Besuchshäufigkeit auf den Zielfeldern.

3.4 Berücksichtigung des Speed-Würfels

Der Speed-Würfel in der Monopoly Deluxe Version bringt eine weitere Variabilität in die Besuchshäufigkeit der Felder, siehe Abbildung 5. Wenn der Monopoly Mann am Speed Würfel erscheint, rückt der Spieler seine Figur auf das nächste freie Grundstück vor. Sind alle Grundstücke bereits verkauft, rückt er auf das nächste Grundstück vor.

Dies hat folgende Konsequenzen:

- Grundstücke werden generell häufiger besucht als andere Felder.
- Der Effekt des Speed Würfels ändert sich im Verlauf des Spiels, abhängig davon, wie viele Grundstücke bereits verkauft sind. Wenn alle Grundstücke verkauft sind, werden die Figuren immer auf das nächste freie Grundstück versetzt.
- Dadurch werden Grundstücke die nach „Nicht-Grundstücks-Feldern“ liegen, häufiger besucht, da wenn alle Grundstück verkauft sind, diese ein häufigeres Ziel der Umsetzung der Figur sind.

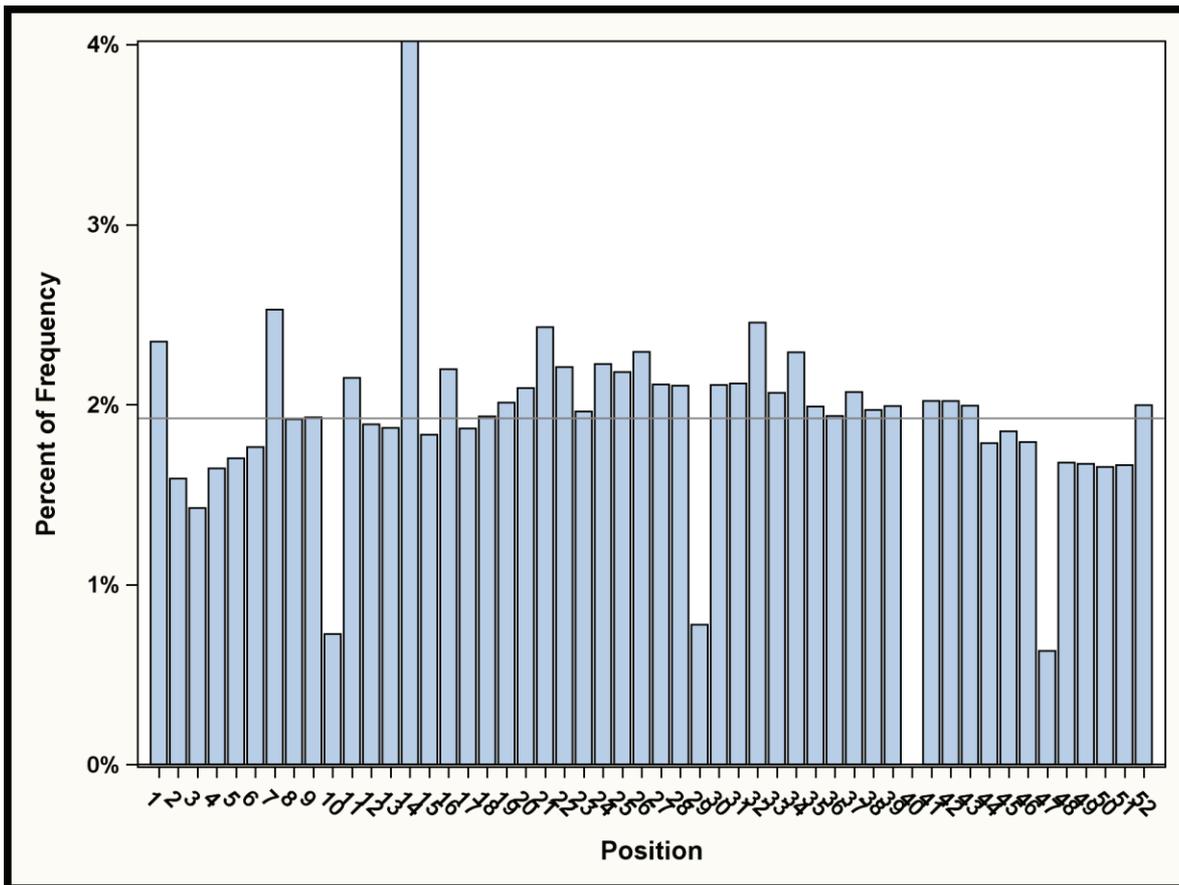


Abbildung 4: Besuchshäufigkeit bei Berücksichtigung der Ereigniskarten

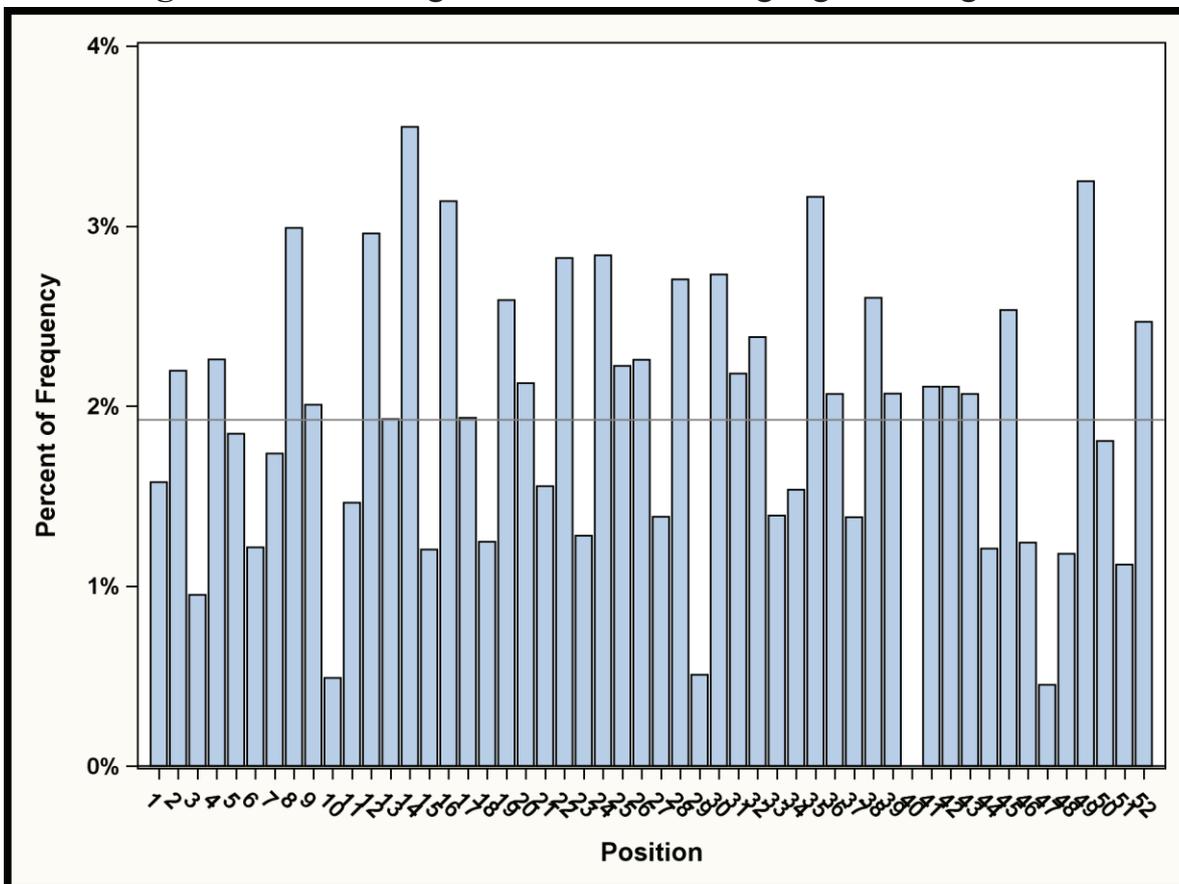


Abbildung 5: Besuchshäufigkeit bei Berücksichtigung des Speed-Würfels

3.5 Ergebnisse der Profit-Simulation

Bei der Profit-Simulation zeigt sich, dass der erwartete Profit für ein Grundstück von unterschiedlichen Faktoren abhängt:

- Anzahl der Spieler, die am Spiel teilnehmen,
- Anzahl der vereinbarten Spielrunden,
- Vereinbarung, oder der Speed Würfel verwendet wird oder nicht.

Abbildung 6 und 7 zeigen die Verteilung der Profitabilität der einzelnen Grundstücke bei 10.000 Simulationsläufen und mit 4 Spielern, einmal nach 40 Spielrunden und einmal nach 70 Spielrunden.

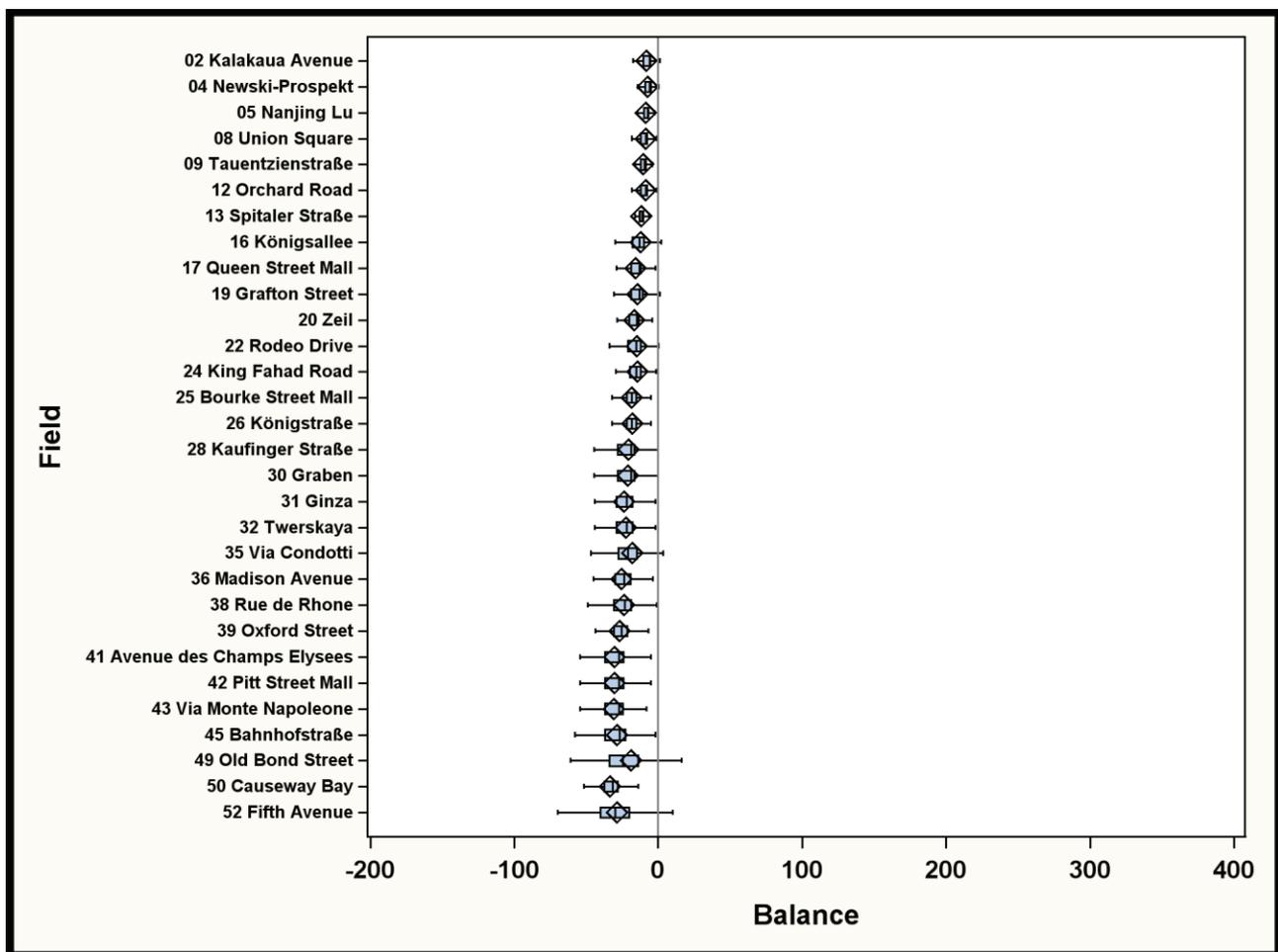


Abbildung 6: Profitverteilung nach 40 Runden

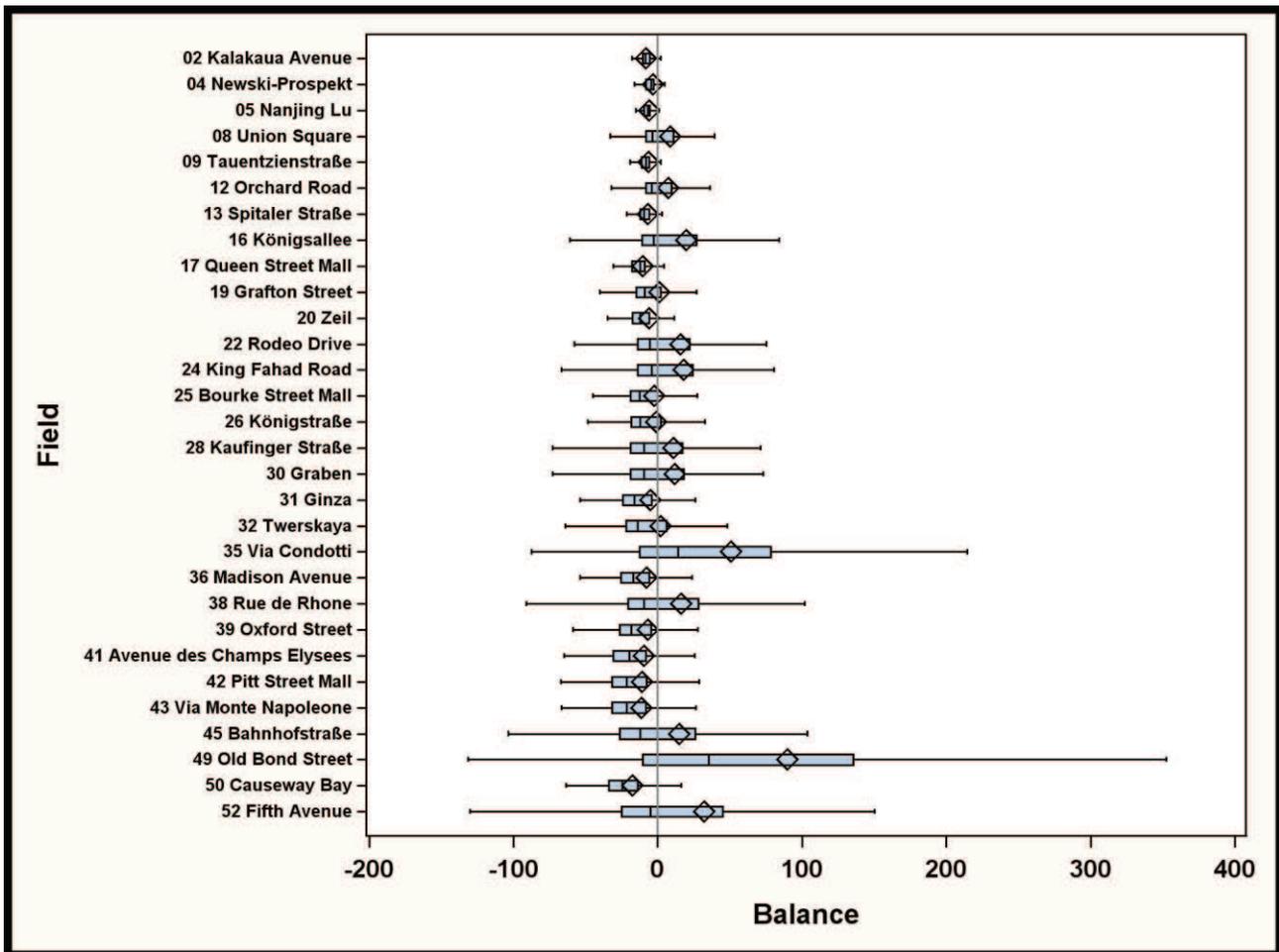


Abbildung 7: Profitverteilung nach 70 Runden

- Es ist zu sehen, dass manche Felder erst in den positiven Profitabilitätsbereich kommen, wenn eine Mindestanzahl von Runden gespielt ist.
- Weiter ist zu erkennen, dass manche Felder zwar eine höhere durchschnittliche Profitabilität, aber auch eine höhere Variabilität haben.
- Felder, welche durch den Speed Würfel begünstigt werden, wie z.B. 16, 22, 24, 35 oder 49, haben eine höhere Profitabilität, weil die Wahrscheinlichkeit steigt, dass sie von anderen Spielern besucht werden.

4 Implementierung in SAS

4.1 Überblick

Abbildung 7 gibt einen Überblick über den allgemeinen Simulationsablauf. Es sind 3 Loops (Spiele, Runden und Spieler) ineinander verschachtelt. Innerhalb eines Loops werden jeweils die Würfel geworfen, die Anweisungen befolgt, die Statistiken und Counter aktualisiert und der Record gespeichert.

In Wicklin [3] und [4] wird gezeigt, dass solche Loops wenn möglich in einem einzigen gesamthaften Datastep erfolgen und nicht durch Macro Loops und ersetzt werden sollen.

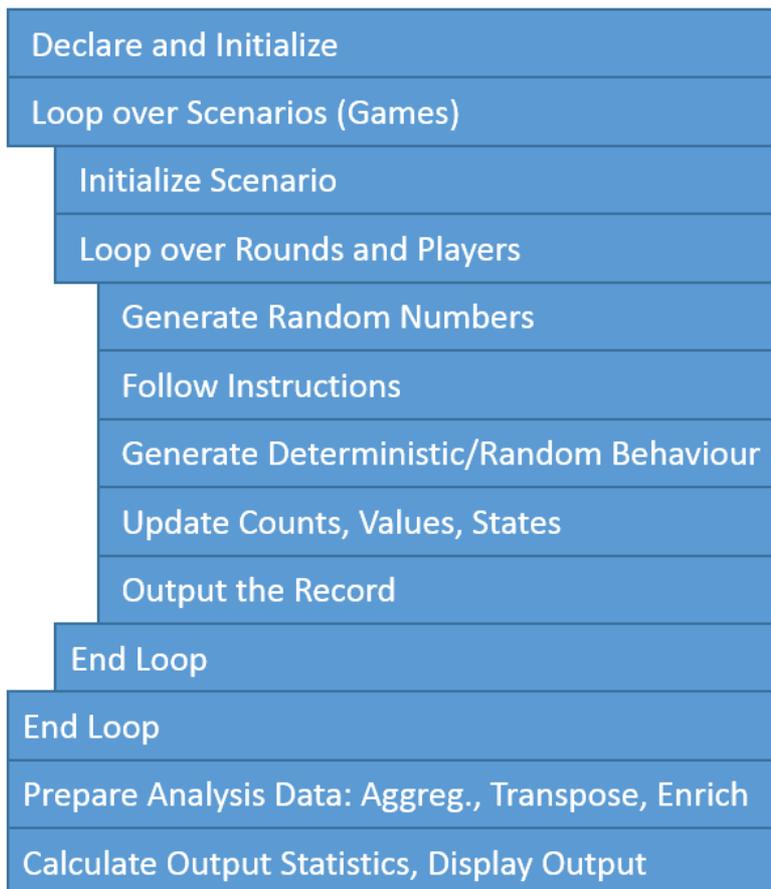


Abbildung 8: Simulationsablauf

Der gesamte Ablauf ist in einem SAS Datastep programmiert, wie in Abbildung 9 schematisch dargestellt.

```

data Monopoly;
  array PlayerPos {4} PlayerPos1 - PlayerPos4;
  do Game = 1 to 10000;
    do Round = 1 to 70;
      do Player = 1 to 4;
        Dicel = ceil(rand('Uniform')*6);
        if PlayerPos[Player]=40 then
          PlayerPos[Player]=14;
        output;
      end;
    end;
  end;
run;
proc transpose data=Monopoly ...;run;
proc sgplot data=Monopoly_TP;

```

Abbildung 9: Schematische Darstellung des SAS Codes für die Simulation

Nach dem SAS Datastep für die Simulation folgen Datenmanagement Schritte für die Auswertung und die Ausgabe der Ergebnisse z.B. in Form von Graphiken.

4.2 Verwaltung der Simulationsdaten in SAS ARRAYS

In der Simulation im SAS Datastep werden die Variablen z.B. für die Spielerpositionen, Salden, Einkommen, Ausgeben der Spieler sowie der Felder als SAS Arrays verwaltet. Dies erlaubt eine flexible Programmierung der Zuweisung der jeweiligen Werte zu den Variablen.

Definition der ARRAYS

Macro Variable `&PLAYERS` ist hier mit dem Wert 4 initialisiert. Es werden somit jeweils 4 Variablen für `PlayerPos` (`PlayerPos1` bis `PlayerPos4`), `PlayerBalance`, `PlayerIncome`, `PlayerExpense` erstellt.

```
Array PlayerPos      {&players} PlayerPos1-PlayerPos&players.
Array PlayerBalance {&players} PlayerBalance1-PlayerBalance&players;
Array PlayerIncome  {&players} PlayerIncome1-PlayerIncome&players.;
Array PlayerExpense {&players} PlayerExpense1-PlayerExpense&players;

Array Field          {52} Field1          - Field52          ;
Array FieldSetup     {52} FieldSetup1     - FieldSetup52     ;
Array FieldRevenue   {52} FieldRevenue1   - FieldRevenue52   ;
Array FieldCost       {52} FieldCost1     - FieldCost52     ;
Array FieldBalance   {52} FieldBalance1   - FieldBalance52   ;
```

Für die Variablengruppen `Field`, `FieldSetup`, `FieldRevenue`, `FieldCost` und `FieldBalance` werden jeweils 52 Variablen angelegt (jeweils eine pro Feld).

Verwendung der ARRAYS

Die Arrays können nun flexibel verwendet werden:

- `PLAYERPOS[2]` bezeichnet die Position von Spieler 2 und referenziert Variable `PLAYERPOS2`.
- `PLAYEREXPENSE[Player]` referenziert die `PlayerExpense` Variable für den jeweiligen Spieler `PLAYER`.
- `FIELDREVENUE[PLAYERPOS[PLAYER]]` referenziert den Revenue jenes Feldes, wo der aktuelle Spieler (`PLAYER`) gerade platziert ist.

4.3 Konvertierung der Lookup-Tabelle in ein SAS Informat

Für jedes Grundstück sind Kosten und Mieterträge für unterschiedliche Anzahl von Häusern in der Spielanleitung enthalten. Diese Daten wurden in eine SAS Tabelle eingegeben, welche auszugsweise in Abbildung 10 dargestellt ist.

	Field	M0	M1	M2	M3	M4
1	2	0.2	1	3	9	16
2	4	0.4	2	6	18	32
3	5	0.5	3	8	24	36
4	8	0.6	3	9	27	40
5	9	0.6	3	9	27	40
6	12	0.6	3	9	27	40
7	13	0.8	4	10	30	45
8	16	1	5	15	45	62
9	17	1	5	15	45	62
10	19	1	5	15	45	62
11	20	1.2	6	18	50	70

Abbildung 9: Lookup-Tabelle für Kosten und Einnahmen pro Grundstück

Diese Daten können nun einfach in ein SAS-Informat konvertiert werden. Die Idee ist es dieses Informat im Programm Code dann aufzurufen und so obige Tabelle, wie eine Lookup Tabelle zu verwenden. Zuerst wird für jedes Informat eine Steuertabelle angelegt:

```
data m0;
  set Property_CostRevenue;
  fmtname = 'm0_';
  type = 'i';
  rename field=start m0=label;
run;
```

Diese wird dann mit der FORMAT Prozedur eingelesen:

```
proc format cntlin=m0 ;
run;
```

Das INFORMAT M0_ steht nun zur Verfügung und kann im Code mit der INPUT Funktion verwendet werden. Der Aufruf von INPUT(8,M0_) returniert den Wert 0.6, da dieser für das Feld 8 in obiger Tabelle in der Spalte M0 eingetragen ist.

4.4 Verwendung der SAS Formate und der SAS ARRAYS in der Simulation

SAS Informats lassen sich in der Simulation mit einem SAS Datastep sehr gut mit SAS Arrays verknüpfen. Dies ist in folgenden Code illustriert.

```
if Field[PlayerPos[Player]] = 0 then do;
  Field[PlayerPos[Player]] = Player
  FieldCost[PlayerPos[Player]] = input(PlayerPos[Player],K0_.);
  PlayerExpense[Player] = PlayerExpense[Player]
    + input(PlayerPos[Player],K0_.);
end;
```

- Für das Feld, wo der Spieler sich gerade befindet, wird seine Spieler-Nummer als Besitz Indikator eingetragen.
- Die Kosten die auf diesem Feld lasten erhöhen sich um jenen Wert, der aus dem Informat für diese Spielerposition (= Feldnummer) erhalten wird.
- Die Ausgaben des Spielers werden ebenfalls um diesen Wert erhöht.

5 Zusammenfassung

Das Monopoly Brettspiel kann sehr gut verwendet werden, um die Simulation eines komplexen Systems zu illustrieren. Allein bei der Analyse der Besuchshäufigkeiten der Felder ergeben sich interessante Einblicke in die Verteilung der Häufigkeiten über die Spielfelder. Der SAS Datastep ist hervorragend für die Durchführung solcher Simulationsstudien geeignet. DO-Loops, SAS Arrays und SAS Informats lassen sich hier ideal kombinieren.

Literatur

- [1] G. Svolba: Applying Data Science: Business Case Studies Using SAS: SAS Press 2017, Cary NC
- [2] G. Svolba: Simulationen und Mathematische Programmierung mit SAS, Proceedings der 20. KSFE (2016)
- [3] R. Wicklin: Simulating Data With SAS: SAS Press 2013, Cary NC
- [4] R. Wicklin: Ten Tips for Simulating Data with SAS: Paper SAS1387-2015, Proceedings des SAS Global Forums 2015.