Formalization of Gambler's Ruin Problem in Isabelle/HOL

Zibo Yang[1,2,*,†]

[1]Ecole Polytechnique, Rte de Saclay, 91120 Palaiseau, France
[2]Institut Polytechnique de Paris, 5 Av. Le Chatelier 2ème étage, 91764 Palaiseau, France

Abstract

The gambler's ruin problem is an exciting application of probability theory which is generally used to analyze various practical scenarios like the security of bitcoin protocol. In this article, we provide a complete analysis of the formalization of the gambler's ruin problem in Isabelle/HOL. First, we present a comprehensive background and pen-and-paper calculation. Second, we summarise how to quantify the gambler's ruin problem and prepare all necessary intermediate conclusions. Third, we explain the difficulties we faced during the final formalization and analyze the strategies to overcome these barriers. Our final result: The recursive probability equation aims to establish the complete quantitative analysis of random walks and assist others in developing advanced probability analysis based on what we endeavour here.

Keywords

Formal Verification, Gambler's Ruin Problem, Probability Theory, Random Walk, Theorem Proving

1. Introduction

Since proof assistants have been developed and become increasingly convenient to use, formalizing various mathematical branches to secure our basis of mathematics becomes more and more popular and influential. Due to the broad influence of probability theory, the formalization of different applications related to it attracts not only mathematicians but also other experts from economics and computer science.

What we formalize in Isabelle/HOL [1] is one of the most well-known problems about random walks application: Gambler's Ruin Problem. It asks about the probability of winning through random,unlimited gambling. In the following paper, we report on quantifying the problem and formalizing all the probability analysis involved. Basically, our work consists of four parts:

- Pen-and-paper introduction and analysis of the gambler's ruin problem(Section 2)
- Formalization of the gambler's ruin model(Section 3)
- Formalization of the necessary intermediate conclusions (Section 3.4)
- Formalization of final goal: the recursive probability equation (Section 3.5)

FMM 2021 – Fifth Workshop on Formal Mathematics for Mathematicians July 30–31, 2021, online
zibo.yang@polytechnique.edu (Z. Yang)
https://easychair.org/publications/preprint/k1hv (Z. Yang)

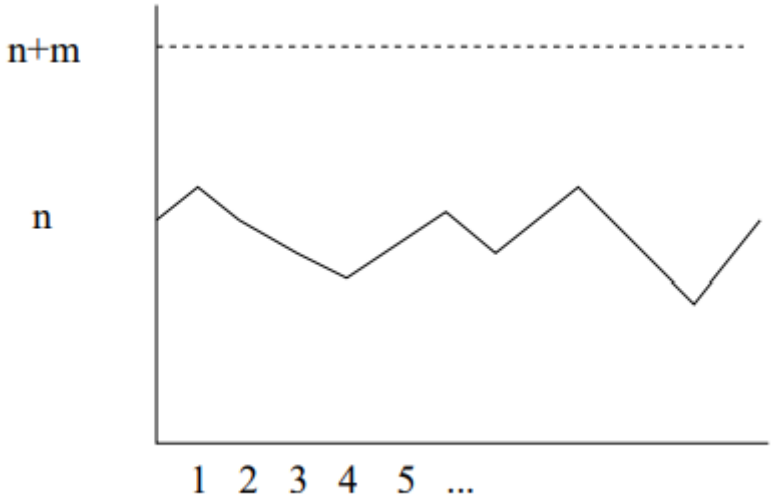

Figure 1: One-Dimensional Random Walk

2. Gambler's Ruin Problem

2.1. Problem introduction

Karl Sigman describes the gambler's ruin problem as follows [2]:

> Let N ≥ 2 be an integer and let $1 \leq i \leq N-1$. Consider a gambler who starts with an initial fortune of \$i and then on each successive gamble either wins \$1 or loses \$1 independent of the past with probabilities p and $q = 1 - p$ respectively. Let X_n denote the total fortune after the n^{th} gamble. The gambler's objective is to reach a total fortune of $\$N$, without first getting ruined (running out of money). If the gambler succeeds, then the gambler is said to win the game. In any case, the gambler stops playing after winning or getting ruined, whichever happens first.

Tom Leighton and Ronitt Rubinfeld further generalize, in their lecture notes of random walk [3] , the problem into one-dimensional random walk mode as below:

> This problem is a classic example of a problem that involves a one-dimensional random walk. In such a random walk, there is some value, say the number of dollars we have, that can go up or down or stay the same at each step with some probabilities. In this example, we have a random walk in which the value can go up or down by 1 at each step [1].

2.2. Basic calculation of Gambler's Ruin Problem

According to the description above, we could formalize the gambler's ruin problem as a one-dimensional model below.

Assume we start with $n > 0$ and randomly walk of $+1$ with the possibility of p or -1 with the possibility of $1 - p$, then we end this game once we get $m > n$ or 0 . The possibility of reaching m from initially n without any stop is what we want to calculate and formalize in the end.

Let P_n be the possibility of successfully reaching m with initial n. Then it is fairly easy to derive the equations:

$$P_n = pP_{n+1} + (1-p)P_{n-1} \text{ if } 0 < n < m$$
$$P_0 = 0$$
$$P_m = 1$$

Proof: $P_0 = 0$ since we have already ruined with initial $0.P_m = 1$ since we've already won with initial m. Let E be the events that first step is plus 1. Let initial denotes the initial number and target denotes the target ending number, then:

$P_n = \Pr(\text{ target} = m \mid \text{init} = n)$

$= \Pr(\text{ target} = m \mid \text{init} = n \bigwedge E) \Pr(E \mid \text{init} = n) + \Pr(\text{ target} = m \mid \text{init} = n \bigwedge \bar{E}) \Pr(\bar{E} \mid \text{init} = n)$

$= p \Pr(\text{ target} = m \mid \text{init} = n + 1) + (1 - p) \Pr(\text{ target} = m \mid \text{init} = n - 1)$

$= pP_{n+1} + (1 - p)P_{n-1}$

3. Formalization of Gambler's Ruin Problem

In Gambler_Ruin_Problem.thy, we will construct the formalization of a specific random walk model coordinated with gambler's ruin problem.

3.1. Theory Infinite_Coin_Toss_Space

In order to construct the formal model in gambler's ruin problem, we start with the existing formalization in the Theory Infinite_Coin_Toss_Space[4] which constructed the probability space on infinite sequences of independent coin tosses.

```
locale infinite_coin_toss_space =
fixes p : real and M :: "bool stream measure"
assumes p_gt_0: "0 ≤ p''
and p_lt_1: "p ≤ 1"
and bernoulli: "M = bernoulli_stream p "
```

The only concept that needs to be elaborated here is bernoulli assumption. 'a stream is the type of infinite sequence with all elements of type ' a. The bernoulli stream is a stream measure with space Ω composed of all the elements of type bool stream, power set A filled with all the measurable subsets under this specific measure and measure function μ produced

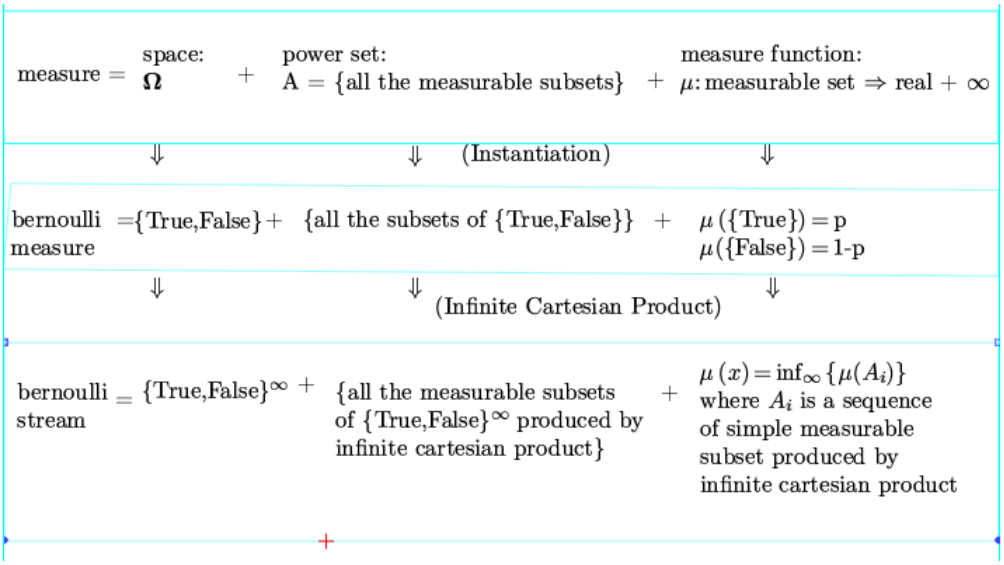

Figure 2: Construction of Bernoulli Stream

by infinite Cartesian product. All the probability of occurrence of elements in a specific set m can be described as the measure value of m if and only if m is the measurable set of this bernoulli stream M. In fact, it is set up by producting countable measures of boolean space with measuring {True} to p and { False } to $1-p$. Despite the complicated and tedious construction of the bernoulli stream, we make a specific diagram [2] here to clarify the structure and mechanism of the bernoulli stream M which we will mention later frequently.

3.2. Gambler model

fun (**in** infinite_coin_toss_space) gambler_rand_walk_pre: "int ⇒ int ⇒ int ⇒ (nat ⇒ bool stream ⇒ int)" **where**
base: "gambler_rand_walk_pre u d v 0 w = v"
step1: "gambler_rand_walk_pre u d v (Suc n) w = $((\lambda True \Rightarrow u \mid False \Rightarrow d)(snth\ w\ n))$+ gambler_rand_walk_pre u d v n w "

fun (**in** infinite_coin_toss_space) gambler_rand_walk:: "int ⇒ int ⇒ int ⇒ (enat ⇒ bool stream ⇒ int)" **where**
"gambler_rand_walk u d vnw = (case n of enat n ⇒ (gambler_rand_walk_pre u dvnw) $/\infty \Rightarrow -1$)"

The function gambler_ran_walk extends the fourth parameter by adding ∞ as new input. We define it because we found it very tough to describe the position where the specific random walk stops, for the first time, by reaching the threshold if the natural number is the only allowed input as what gambler_ran_walk_pre defines. Since some infinite random walks will never stop, we must allocate ∞ as the input coordinated with those nonstop cases and extend the type of

steps from nat to enat. Nevertheless, if someone wants to base their further analysis on our endeavour here, please be cautious of or even avoid discussing the case that the initial number and target number is negative since we map ∞ to -1. The lemma exist demonstrates that non-stop random walks will never succeed in reaching the target, which is the best explanation why we assign -1 as the output of ∞ in gambler_ran_walk.

> **locale** gambler_model = infinite_coin_toss_space +
> **fixes** geom_proc::"int ⇒ bool stream ⇒ enat ⇒ int"
> **assumes** geometric_process:"geom_proc init x step = gambler_rand_walk 1 (-1) init step x"

3.3. Basic functions

Here we define all the basic functions which will play an important role in the further probability analysis.

- Fun reach_steps describes all the steps where the input random walk reaches the threshold 0, target
- Fun stop_at describes the first step in the reach_steps sets, which means exactly the stopping point in gambler's ruin problem. Note that, here the type of output has been extended to enat, which means stopping point will be ∞ (equivalent to non-existence)
- Fun success describes the random walk reaching the target number rather than ruining at stopping point

3.4. Important intermediate conclusions

In this section, we will specify many necessary intermediate lemmas to lay a solid foundation for further analysis. Regardless of plenty of complex lemmas formalized, there are two significant things we pursue:

- Trying to clarify that our definition for gambler_rand_walk is reasonable by proving the abnormal situation doesn't exist:
- lemma exist demonstrate that the weird situation where random walk succeeds at ∞ will disappear once we set target to be positive, which is the reason why we set -1 as the output of ∞ over function geom_proc
- Demonstrating the ways we count are independent with the numeral results we calculate:
- lemma additional1 states that the reaching number doesn't change if we want to calculate from the second step
- lemma conditional 2 states that whether a random walk succeeds or not doesn't change if we calculate from second step
- lemma fst_true_plus_one states that we add 1 to initial number if first step is true.
- lemma conditional_set_equation states that the set where all random walk in it succeeds and their first step are True doesn't change if we calculate from second step.

3.5. Probability equation

In this section, we start to analyse the probability of successful random walk. There are two lemmas which pose enormous difficulty and deserve more explanation from different aspects: Lemma success_measurable and Lemma semi_goal_true

probability_of_win is the function describing the possibility of successful random walks with initial number and target number as inputs

fun probability_of_win:: "int ⇒ int ⇒ ennreal " **where**
"probability_of_win init target = emeasure M {$x \in$ space M. success init x target}"

3.5.1. Successful random walk set is measurable

First, we introduce this most challenging lemma we have met during this model formalization. Lemma success_measurable asserts that successful random walks set under the assumption "0 $\leq$ initialnumber $\leq$ targetnumber" is a measurable set for measure M.

On the one hand, the lack of discrete analysis library might have been the most significant factor causing the trouble here. Since the probability theory has been set up based on the measure theory, every specific set must be proved to be measurable concerning fixed measure before we calculate the probability of the set, which severely hinders most scholars and experts from formalizing the security analysis related to the probability since it is extremely difficult to prove why your set is measurable. That is precisely why our endeavour matters to provide an example to overcome the difficulty.

On the other hand, the chaos given from the gambler's ruin model itself leads to a large part of difficulties here. The topological space where we base our discussion here is the infinite Cartesian product which makes our formalization deteriorate since it tends to be more challenging to clarify the topology on the infinite Cartesian product. The function success we rely on is not the easy one to construct(see our discussion in section 3.3. Its complex definition increases the difficulties when we handle the measurability.

we will briefly explain here the way we prove this lemma since it is nontrivial even for pen-and-paper proof.

- Lemma finite_stake_measurable states that for the function (λw. stake nw) listing the first n steps of random walk, the preimage of a finite set is measurable for measure M.
- Lemma finite_image states that sets filled with all bool lists of fixed length n is finite. - Lemma success_measurable 2 is the most important intermediate lemma prepared for lemma success_measurable. It clarifies that any list in the image of successful random walk over function stake will never contain another shorter list corresponding to another successful random walk, which sets up the bijection between successful random walks stopping at fixed-step and preimage of successful bool list with identical length.
- Lemma success_measurable1 demonstrates that a set of successful random walks is a countable union of sets of successful random walks stopping at some step.

Combining theses 4 lemmas together proves the set of successful random walks is measurable.

These lemmas are surprisingly hard to prove. Honestly,the successful formalization of this part cannot be accomplished without the current stochastic process theory library established just in 2021 by Mnacho Echenim, the author of theory infinite_coin_toss_space.

3.5.2. Probability of successful random walk with its first step True

The lemma semi_goal_true is the second difficulty we have overcome during the model formalization. It asserts that the probability of sets of successful random walk with first step True is equal to probability of sets of random walk times probability of sets of successful random walk with initial number plus 1 .

According to the definition of measure value of subset in the infinite Cartesian product, we need to calculate a sequence of measure value of simple measurable sets constructed by multiplying its infinite projections and carefully drives to the limit of that sequence as the measure value we desire. As everyone knows, this approach is not feasible since the set we measure is too irregular to find the sequence of simple measurable sets to cover it. So how to calculate such an irregular measurable set?

Here we briefly explain the deduction. The following two lemmas are the most important intermediate conclusion we prepare:

- Lemma semi_goal1 states the equation between the measure value of set of successful random walks calculated from either first step or second step, which facilities our prepare for the function with first step as input and measure value as output.
- Lemma semi_goal2_final declares the function we set up in lemma semi goal1 can be calculated in integral prefectly.

Thanks to the lemma emeasure_stream_space provided by Mnacho Echenim, the author of infinite_coin_toss_space, we can finally use the integral rather than tediously break down the countable product to calculate the probability. The things we cooperate to formalize here are providing the function for the integral (lemma semi_goal1) and proving its relevant properties(lemma semi_goal2_final). Although our formalization went well after establishing this two lemmas, it still took us more than two weeks to accomplish the lemma semi_goal_true

3.5.3. Final goal: establish the recursive probability equation

The final probability equation we want to formalize:

$$P_n = pP_{n+1} + (1-p)P_{n-1}$$

lemma Recursive_probability_equation:
fixes init target
assumes "0< init" "init < target"
shows "probability_of_win init target = $p*$ (probability_of_win (init $+1$) target)
$+(1-p)*$ (probability_of_win (init - 1) target)"
⟨ proof ⟩

4. Conclusions

It seems that the difficulties of formalizing the gambler's ruin problem are beyond our initial imagination. By describing and explaining what we endeavour to construct through all the chapters, we have experienced this arduous but exciting formalization journey again. Fortunately, we succeed in formalizing all the quantitative probability analysis of this model and resolving the difficulties such as securing the measurability of sets calculated later and calculating the measure value of the sets by integral equation. Our success will not come true without many excellent existing entries like theory In finite_Coin_Toss_Space and our persistence.

The difficulties we faced arise from multiple aspects. However, the two most critical aspects are the inconvenience originating from the problem itself and the lack of appropriate entries and libraries preparing enough discrete probability analysis. Honestly, we have no desire to establish such a perfect library of discrete probability analysis ultimately. However, we hope that our contribution here could offer an example for other excellent scholars when they face a similar challenge like us in the future.

5. Future work

The short-term goal shortly is formalizing the complete security analysis of the bitcoin mechanism. In the bitcoin whitepaper, Satoshi proposed the link between Gambler's ruin problem and an attacker model of bitcoin network [5]:

The probability of an attacker catching up from a given deficit is analogous to a Gambler's Ruin problem. Suppose a gambler with unlimited credit starts at a deficit and plays potentially an infinite number of trials to try to reach breakeven. We can calculate the probability he ever reaches breakeven or that an attacker ever catches up with the honest chain.

So the next step for us is trying to apply our well-prepared gambler's ruin model to the security analysis of the bitcoin network by formalizing the analysis of the attacker model there. Since right now all the fundamental models and mathematical tools have been formalized in AFP libraries [1] and our work here, We have the confidence to accomplish this goal before the end of internship at around 27^{th} August.

6. Acknowledgements

The author was supported by the ERC Advanced Grant ALEXANDRIA (Project GA 742178) funded by the European Research Council and led by Professor Lawrence Paulson. The author was also supervised by Prof. Lawrence Paulson and Dr. Anthony Bordg at the University of Cambridge, UK.

References

[1] T. Nipkow, M. Wenzel, L. C. Paulson, Isabelle/HOL: a proof assistant for higher-order logic, Springer, 2002.

[2] K. Sigman, Gambler's ruin problem, 2016. URL: http://www.columbia.edu/~ks20/FE-Notes/4700-07-Notes-GR.pdf.
[3] T. Leighton, R. Rubinfeld, Random walk lecture notes, 2016. URL: http://web.mit.edu/neboat/Public/6.042/randomwalks.pdf.
[4] M. Echenim, Pricing in discrete financial models, Arch. Formal Proofs 2018 (2018).
[5] S. Nakamoto, A. Bitcoin, A peer-to-peer electronic cash system, Bitcoin.–URL: https://bitcoin. org/bitcoin. pdf 4 (2008).