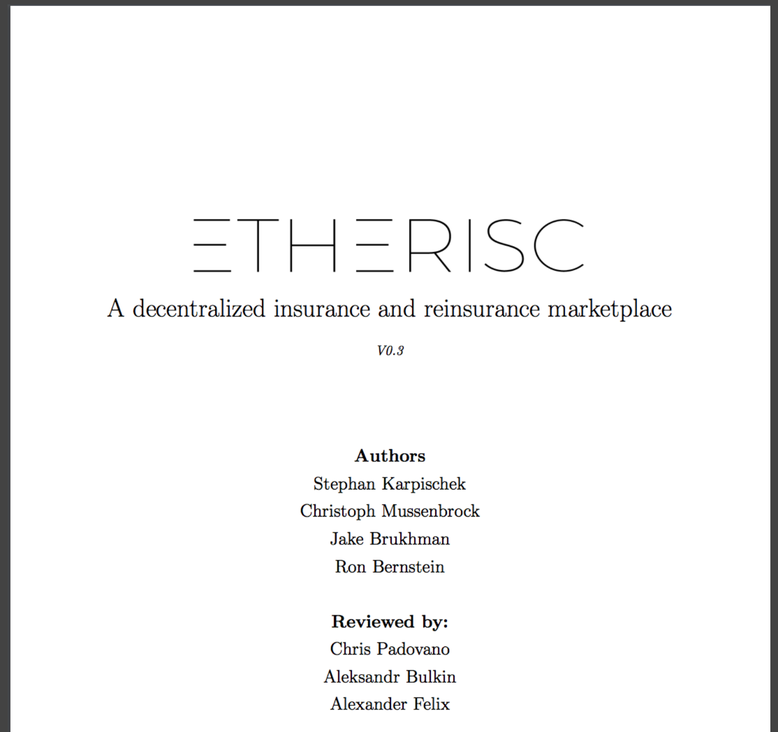
CoinFund

A simple probability model for decentralized insurance

Jake Brukhman, CoinFund
jake@coinfund.io



ETHERISC

A decentralized insurance and reinsurance marketplace

V0.3

Authors
Stephan Karpischek
Christoph Mussenbrock
Jake Brukhman
Ron Bernstein

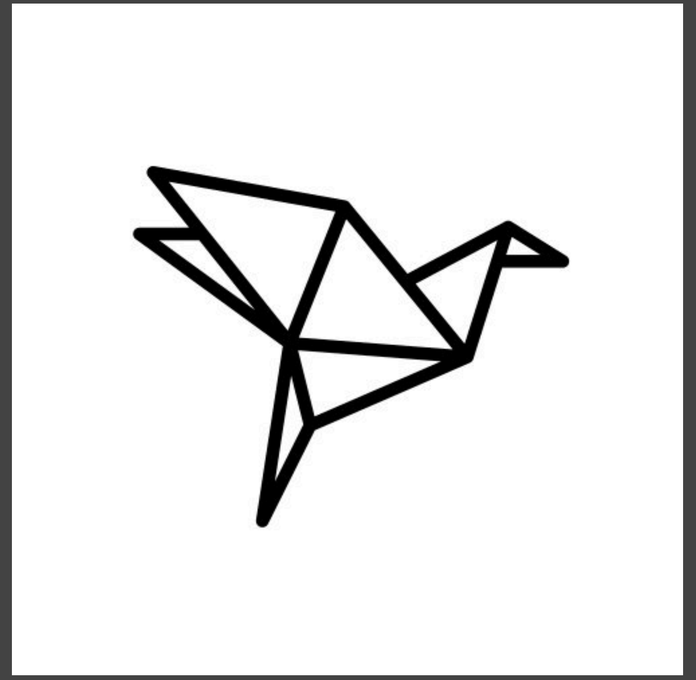
Reviewed by:
Chris Padovano
Aleksandr Bulkin
Alexander Felix

1. **Background**
    1. Mathematics and computer science
    2. Quantitative research, trading, and engineering on Wall Street
    3. TPM & Engineer at Amazon.com
    4. CTO @ Triton Research
2. **Founder @ CoinFund**
    1. One of the first **cryptofunds** (2015) and **cryptoeconomics advisory teams** (2016)
    2. **Lead advisory** with companies building new blockchain-based business models (Kik Interactive, YouNow/PROPS, Sweetbridge, Civil)
    3. **Research in decentralized systems** and token models (Mediachain, AugmentPartners, StorJ, Aragon, Decentraland, etc.)

CoinFund

WORK WITH ETHERISC

1. Got to know Stephan and Christoph on the CoinFund Slack
2. Joined as team member for EtherCamp hackathon (11/2016)
3. Helped to construct first Etherisc whitepaper and PoC **probability risk model**
4. Watched team navigate financing, PoCs, cryptoeconomics, regulatory compliance
5. Continue to be involved as stakeholder and advisor



WHY DECENTRALIZED INSURANCE?

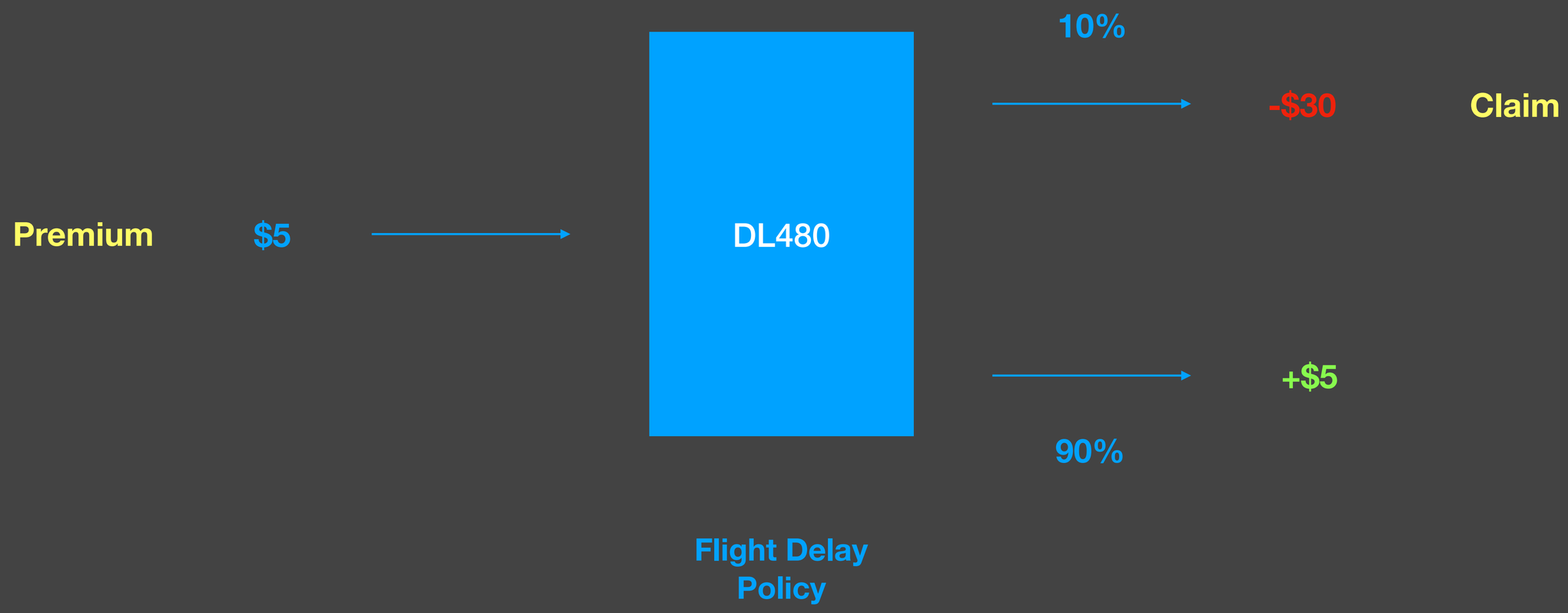
1. **Overall Thesis: “The medium term upside is in consumer-proximate applications with real user bases and use cases, not TCP/IP-proximate protocols.”**
    1. “Fat protocols are not an investment thesis” (CoinFund Blog, http://blog.coinfund.io)
2. **Traditional perspective**
    1. Huge market opportunity
    2. Lots of new consumer use cases
    3. We **want** to live in a world where we can efficiently offset all kinds of risks
3. **Blockchain perspective**
    1. Desirable: fundamental cryptoasset pricing methods based on mathematical models
    2. Regulatory guidance, frameworks



1. **Goals**
    1. Simple model that demonstrates the principle of insurance
    2. Show that it is computationally tractable
    3. Show that it is feasible to implement in a decentralized system

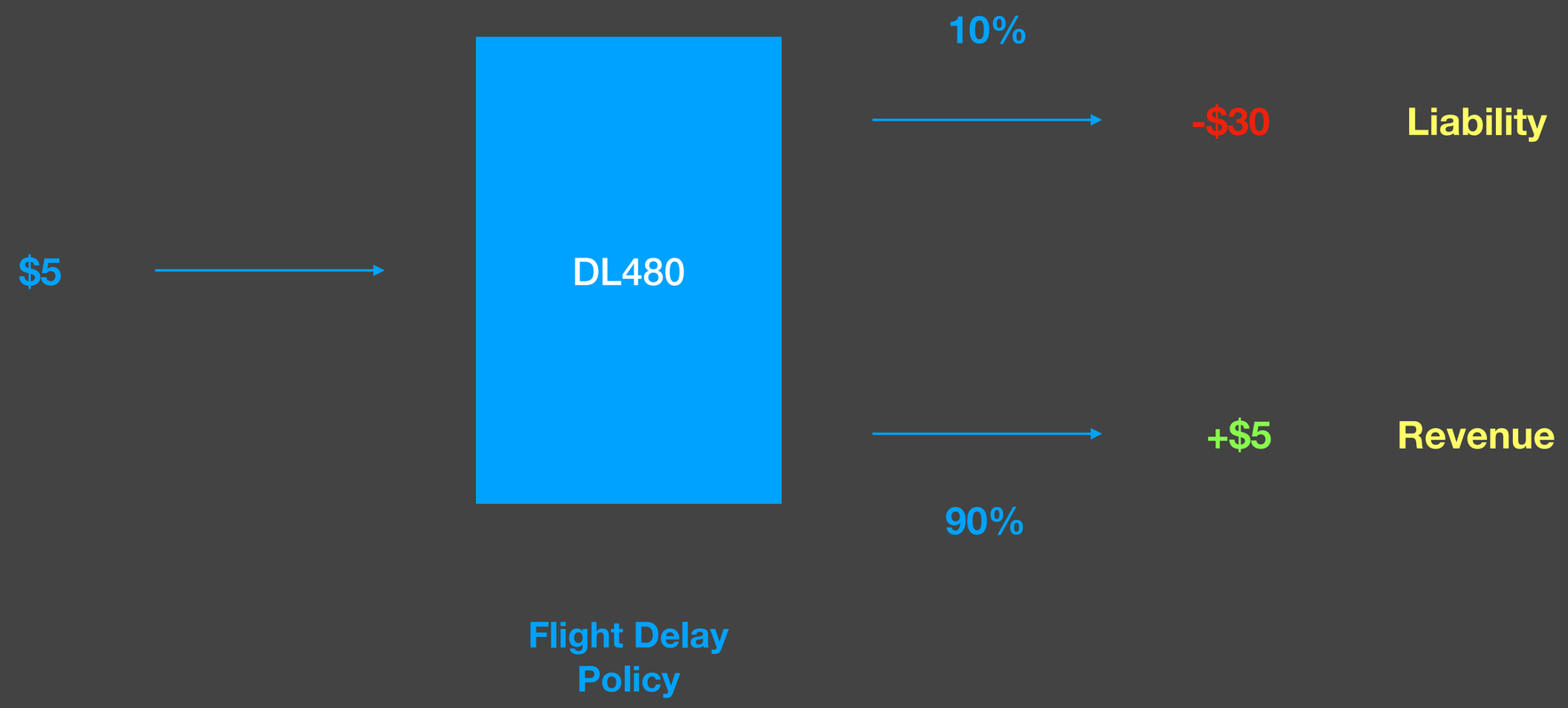
CoinFund

A FLIGHT DELAY INSURANCE MODEL IN 10 MINUTES



CoinFund

A FLIGHT DELAY INSURANCE MODEL IN 10 MINUTES



CoinFund

A FLIGHT DELAY INSURANCE MODEL IN 10 MINUTES



$50 - $30 = $20

CoinFund

A FLIGHT DELAY INSURANCE MODEL IN 10 MINUTES



Revenue: +$50
Max Liability: -$300
Expected Liability: -$30
Expected Revenue: +$20

1. **Strategy for a simple probability model**
    1. Fight delays modeled as independent Bernouli random variables
        1. Independent random variables
        2. **Not** identically distributed (i.e. the delay probabilities are different)
    2. Find **expected liability** up to some very, very high confidence interval
        1. 99.9999%
    3. Back the insurance pool with enough capital to cover expected liability
    4. Keep the rest as profit
    5. Offload the long-tail risk to a reinsurance market

CoinFund

A FLIGHT DELAY INSURANCE MODEL IN 10 MINUTES



2 Probability model

Let us assume that our risk pool portfolio contains n policies which are insuring against n insurable events, modeled as independent but not necessarily identically distributed random variables X_i $(i = 1, \ldots, n)$. Suppose that P_i^* $(i = 1, \ldots, n)$ is a fixed set of desired payouts where P_i^* corresponds to the policy i. Furthermore, suppose that $X_i \in \{0, 1\}$ are Bernoulli random variables with event probability p_i $(i = 1, \ldots, n)$. That is, $P(X_i = 1) = 1 - P(X_i = 0) = p_i$ for $i = 1, \ldots, n$.

The *total liability* L of the portfolio is the sum of all payouts the portfolio is underwriting, and we define

$$L(n) := \sum_{i=1}^{n} P_i^*.$$

We are interested to find the *required collateral* $C(n)$, that is, the amount of capital the portfolio will hold to handle capital outflows due to claims. Typically, $C(n)/L(n) < 1$, reflecting the fact that the portfolio is taking on credit risk. Our goal is to keep enough collateral in the portfolio to be able to pay a reasonably expected number of policy claims. We define the total capital outflow due to claims as

$$X := \sum_{i=1}^{n} X_i P_i^*.$$

X is a random variable that has a non-trivial distribution which is the weighted sum of Bernoulli variables with non-uniform probabilities. Let us define π to be the desired confidence level (probability) of portfolio solvency, and we note that typically π will be high ($\pi \approx 1$). Let F_X be the cumulative distribution function of X. Our model is then defined by setting the required collateral C to the π-percentile probable capital outflow due to claims:

$$C(n) = F_X^{-1}(\pi). \tag{1}$$

Let us now to turn to calculating the set of premiums P_i $(i = 1, \ldots, n)$. For one, we must have that $\sum_{i=1}^{n} P_i = C(n) = F_X^{-1}(\pi)$, but we are free to choose how to distribute the

2.1 Model inputs

1. A vector of insurable event probabilities $\mathbf{p} = \langle p_i \rangle$, $(i = 1, \ldots, n)$.
2. A vector of desired payouts $\mathbf{P}^* = \langle P_i^* \rangle$, $(i = 1, \ldots, n)$.
3. A confidence level for portfolio solvency π, where $.5 < \pi < 1$.

2.2 Model outputs

1. **Total Liability.** Total portfolio liability is given by

$$L(n) := \sum_i P_i^*$$

.

2. **Collateral.** Required minimal collateral is given by

$$C(n) := F_X^{-1}$$

.

3. **Excess Liability.** The excess portfolio liability on offer to a reinsurance market is

$$\tilde{L}(n) := L(n) - C(n)$$

.

4. **Premiums.** A vector of premiums corresponding to the payouts $\mathbf{P}^*$, given by

$$\mathbf{P} := \left\langle \frac{p_i P_i^*}{\sum_j p_j P_j^*} F_X^{-1}(\pi) \right\rangle .$$

5. **Capital Outflow.** The expected capital outflow due to claims is $E(X) = \sum_i p_i P_i^*$. The standard deviation of capital outflow due to claims is

$$\sigma_X = \sqrt{\sum_{i=1}^{n} p_i(1-p_i)(P_i^*)^2}$$

.

6. **Expected Revenue.** The expected revenue for the portfolio is $E(R) = C(n) - \sum_{i=1}^{n} p_i P_i^*$. The standard deviation of revenue is

$$\sigma_R = \sigma_X = \sqrt{\sum_{i=1}^{n} p_i(1-p_i)(P_i^*)^2}$$

Etherisc insurance calculation

n: 60
mu: 1412.63
sd: 561.27
L: $15000.00
C: $3750.00
%: 25.00
r: 4.00
R: $2337.37

	prob	premium	payout
DL_762_ATL_MDW	0.018182	12.066488	250
WN_349_RIC_ATL	0.022727	15.083110	250
WN_349_ATL_CMH	0.022727	15.083110	250
WN_203_BNA_SAT	0.026316	17.464685	250
DL_780_ATL_CVG	0.040000	26.546313	250
DL_762_MDW_ATL	0.040000	26.546313	250
KL_724_HAV_AMS	0.040816	27.088050	250
DL_132_TPA_DTW	0.046512	30.867805	250
SK_904_EWR_ARN	0.048387	32.112468	250
DL_160_MSP_AMS	0.048387	32.112468	250
DL_132_DTW_AMS	0.052632	34.929329	250
WN_203_MDW_BNA	0.052632	34.929329	250

NEXT STEPS

1. **Scale**
    1. System becomes viable at scale
2. **More sophisticated models**
    1. Don’t assume independence
    2. Take factors like weather, regulatory activity into account
    3. Make the insurance system able to run in perpetuity
3. **Reinsurance**
    1. No matter how good your model is, assumptions hold until they don’t
    2. Sell long-tail risk to a reinsurance market
4. **Cryptoasset Pricing**
    1. Mean, variance of future revenue allows for better fundamental pricing
    2. Discounted future cash flows with risk measurement/confidence

CoinFund

THANK YOU