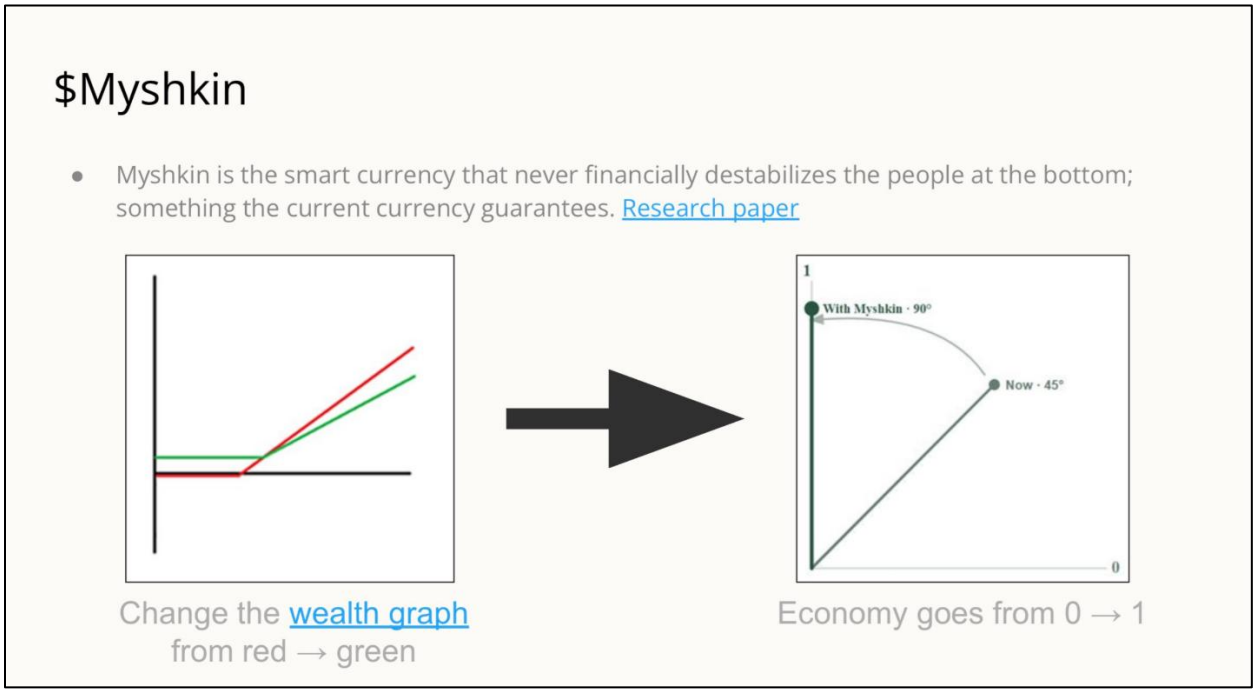
The Last Move

Aditya Narayanan



Abstract

If the rich *gave away* just enough money so the bottom wasn't financially destabilized then Elon Musk could sell 2 billion cars per year instead of 2 million.

The stock price and means of production skyrocket, so he would also get widely richer than he would otherwise stabilize at without this mechanism, despite giving some money away and owning a smaller share of the fiat capital.

Introduction

“I think if we sell Heaven to one another we can all have it for ourselves.”

Capitalism Improperly Settles

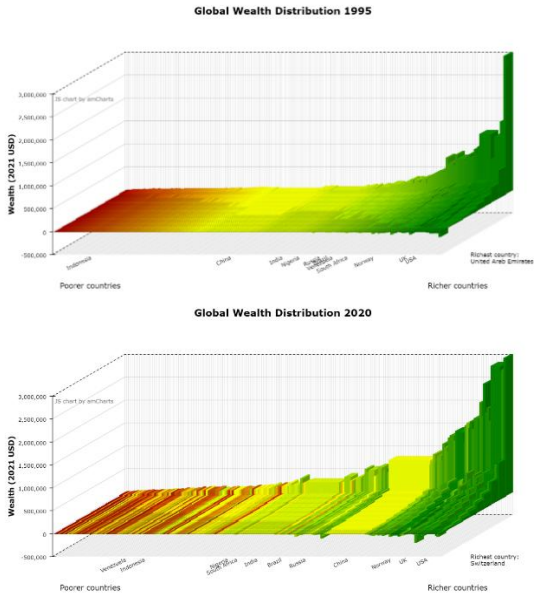


Figure 1: Wealth graphs, 1995 and 2020

The graphs show that over 25 years, the rich got a little bit richer, but otherwise things stayed the same. This is where capitalism currently stabilizes.

There’s a key here, which is that even though the rich got a little bit richer, the red countries *still exist*. This is useful information because if Elon Musk could sell a car to 2 billion people yearly, instead of 2 million currently, Tesla’s stock price would skyrocket, and his wealth would actually increase.

Most of his money is in stocks, allowing Tesla to flourish as a company by facilitating a world where all 8 billion people could afford a Tesla would make him 1000x richer.

I show that by facilitating a mechanism that allows the rich to give away fiat capital, they get way richer, *and* the bottom get some spending money.

The Mechanism

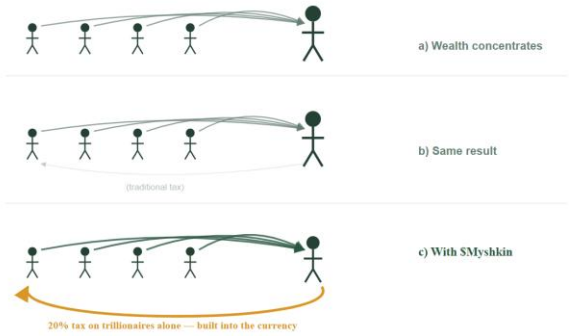


Figure 2: Visual effect of decentralized currency

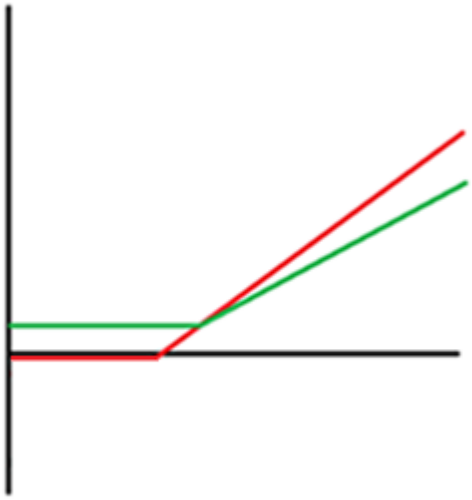


Figure 3: Wealth graph with and without Myshkin

Myshkin is a smart currency, the currency that never financially destabilizes the people at the bottom. Financially destabilizing the people at the bottom is something the current currency guarantees (red countries).

By giving some money away through a smart currency that auto-taxes trillionaires alone 20% yearly, the stock market can skyrocket as Tesla sells 2 billion cars instead of 2 million, and the rich get actually rich.

Every wallet has a tax-free allowance D (the "abundance floor").

Suggested starting value: $D = 100{,}000$ tokens ≈ $100,000 at $1/token launch price.

1 quadrillion tokens total, close to the 500 trillion in fiat capital globally currently.

For any balance $B > D$, the wallet pays a continuous quadratic tax on the excess $E = B - D$.

Instantaneous tax rate (in tokens per second):

$$\frac{dB(t)}{dt} = -k \times E^2(t) = -k\,(B(t) - D)^2$$

where k is the aggression parameter (suggested $k \approx 6.34 \times 10^{-21}$ for a 20% annual tax on someone holding 1 trillion tokens).

The total tax collected per second across the entire system is:

$$GlobalΔ(t) = \Sigma_i\, k\,(B_i(t) - D)^2$$

This entire amount $GlobalΔ(t)$ is immediately and perfectly evenly redistributed as a universal dividend:

$$\frac{Dividends}{person} = \frac{GlobalΔ(t)}{N(t)}$$

where $N(t)$ is the current number of registered wallets.

Net token flow for wallet i (excluding external trades/earnings):

$$\frac{dB_i}{dt} = -kE_i^2 + \frac{GlobalΔ(t)}{N(t)}$$

Which simplifies to:

$$\frac{dB_i}{dt} = -k(B_i - D)^2 + \frac{k}{N(t)} \times \Sigma_j\,(B_j - D)^2$$

This equation floats money until everyone has at least $100,000 in their bank account.

Results

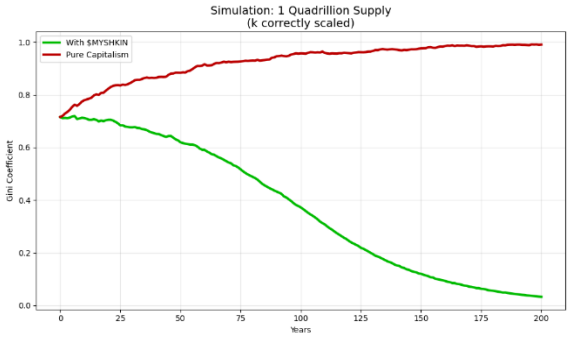


Figure 4: Simulation results, Gini coefficient with and without $MYSHKIN control system in place – Simulation A

The graph shows a collapse in the gini coefficient – desirable. The mechanism offers asymptotic time stability of the gini, basically it tends towards 0 if the economy ever completely stopped for forever.

By creating a currency that automatically redistributes wealth so that the bottom aren't left behind, we've solved the problem of economics at the differential level. Previously when a customer bought something, they'd lose the money to then be a customer again. This left Capitalism stuck in a state of wanting to move forward but the people having no money.

By implementing this control system, we resolve the Chinese finger trap Capitalism gets stuck in of the wealth concentration choking the flow of capital. The goal of allowing customers to be customers indefinitely is achieved.

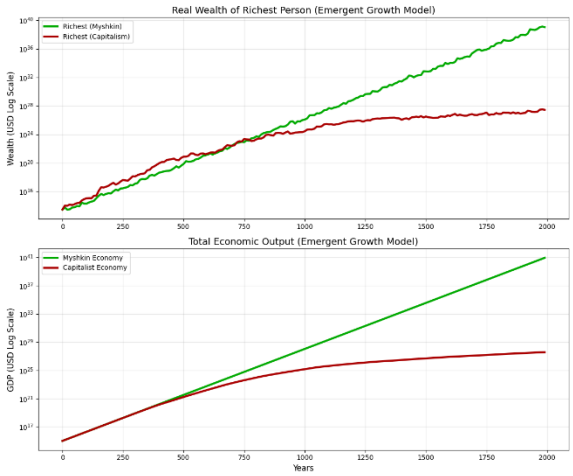


Figure 5: Real wealth of the richest person with and without Myshkin and total economic output with and without Myshkin – Simulation B

The graphs show an explosion in the wealth of the rich and real world output as Myshkin gets adopted as a currency.

Implications to Life

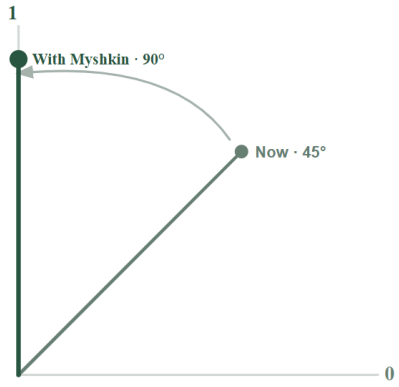


Figure 6: Economy going from 0 -> 1 in going from 1 -> 0 in what it has on us

Peter Thiel talks about life being that of going from "0 -> 1". This is because we must stand up and fight against the dying of the light that is life. Where life has 1 on us, we need to stand up such that we have 1 of our own.

This solution, once money is abundant, allows us to afford to automate ourselves out of the economy, with the ramp in robotics and automation coming soon.

As such, I propose a counter to Peter Thiel's claim, in fact I provide the right answer: in times of war, we "make our opponents go from 1 -> 0". By automating the economy, we will develop the system such that we may revert to before we felt the need to fight indefinitely, a fight all past men have lost. We shall be the first to win.

And the best part is that it's inevitable because everybody *needs* things to go up and to the right, and yet the system stagnates otherwise.

Time Placement in Civilization's Trajectory

I find that the arc of civilization follows these of sorts footprints in the sand, of

"Create Capitalism"

The wealth concentrates

And the next steps we take are:

"Make money abundant"

We can afford to automate ourselves out of the economy

The Transition to Heaven

1. The Bottom Receives
    a. Myshkin redistributes 20% from trillionaires to the bottom. For the first time, 8 billion people have spending power.
2. The Money Flows Up
    a. They spend – on Teslas, on goods, on services. The money still flows to the top, but now through 8 billion channels instead of 2 million.
3. The Rich Get Richer
    a. Tesla sells 2 billion cars. Stocks soar. Despite the 20% tax, owners hold stronger monopolies and more currency flows through them than ever.
4. We Fund Robotics and Automation
    a. With universal spending power, we collectively pay for the robotics and automation that replaces our own jobs – willfully, joyfully.
5. Heaven
    a. Everyone has financial abundance. The economy goes from 0 -> 1. We all retire. The economy works for us – not the other way around. One people, no division. We've stabilized at the second equilibrium.

Conclusion

The paper suggests a way to curb income inequality indefinitely. By investing in the customer base the end result is that nobody has to work ever again and we can be richer than we've ever been before.

Appendix A – Simulation A

```
import numpy as np
import matplotlib.pyplot as plt

# ==========================================================

# MYSHKIN SIMULATION — 1 QUADRILLION TOKENS
# FIXED: k scaled to 2e-14 to handle 1Q supply naturally
# ==========================================================

# ---------------------- GINI FUNCTION ----------------------
def compute_gini(w):
    w = np.sort(w)
    n = len(w)
    if n == 0: return 0.0
    cumw = np.cumsum(w)
    if cumw[-1] == 0: return 0.0
    return (2 * np.sum(np.arange(1,n+1) * w) / (n * cumw[-1]) - (n+1)/n)

# ---------------------- PARAMETERS ----------------------
TOTAL_SUPPLY = 1_000_000_000_000  # 1 QUADRILLION tokens scaled down because N was scaled down
N            = 10_000             # agents scaled down from 10 billion global population
D            = 100_000            # demurrage-free floor

# --- THE FIX ---
# Original k was at ~1Q people. for 1e6 less people, multiply k by 1e6
# To scale in time from seconds to years, scales by 3.1563e7.
k            = 6.34e-23*1e6*3.1563e7
# ----------------

mu           = 0.05               # 5% drift
sigma        = 0.25               # volatility
T            = 200.0              # years
dt           = 0.01               # time step (3.65 days)
steps_per_year = int(1/dt)

# ------------------- INITIAL DISTRIBUTION ------------------
np.random.seed(42)
xm     = D * 0.1
alpha  = 1.05
```

```
wealth = xm * (1 - np.random.rand(N)) ** (-1 / alpha)

# Force exact total supply
wealth *= (TOTAL_SUPPLY / wealth.sum())

wealth_with    = wealth.copy()
wealth_without = wealth.copy()

print(f"Total supply         : {TOTAL_SUPPLY:,.0f}")
print(f"Average Wealth       : {wealth.mean():,.0f}")
print(f"k (Aggression)       : {k:.1e}")
print(f"Initial Gini         : {compute_gini(wealth):.4f}\n")

# ---------------------- STORAGE ----------------------
gini_with    = [compute_gini(wealth_with)]
gini_without = [compute_gini(wealth_without)]
times        = [0]
dividend_yr  = [0]
div_times    = [0]

# ========================= SIMULATION =========================
for step in range(int(T/dt)):
    dW = np.random.normal(0, np.sqrt(dt), N)

    # --- 1. Pure Capitalism ---
    wealth_without *= np.exp((mu - 0.5*sigma**2)*dt + sigma*dW)

    # --- 2. Myshkin System ---
    # Market movement
    wealth_with *= np.exp((mu - 0.5*sigma**2)*dt + sigma*dW)

    # Quadratic Demurrage (Natural Math)
    excess = np.maximum(wealth_with - D, 0)

    # With k=2e-14, this tax is now naturally < 100% of wealth
    tax = k * (excess ** 2) * dt

    # Redistribution
    total_tax = np.sum(tax)
    dividend  = total_tax / N

    wealth_with -= tax
    wealth_with += dividend
```

```
    # Prevent numerical zeros just for log plot safety (not for solvency)
    wealth_with = np.maximum(wealth_with, 1e-2)

    # Record Data
    if (step+1) % steps_per_year == 0:
        year = (step+1) * dt
        times.append(year)

        gw = compute_gini(wealth_with)
        gwo = compute_gini(wealth_without)

        gini_with.append(gw)
        gini_without.append(gwo)

        annual_div = dividend * (1/dt)
        dividend_yr.append(annual_div)
        div_times.append(year)

        print(f"Year {year:2.0f} | Gini Myshkin {gw:.4f} | "
              f"Gini Cap {gwo:.4f} | Div {annual_div:,.0f}")

# ========================= PLOTS ==========================
plt.figure(figsize=(10,6))
plt.plot(times, gini_with,    label="With $MYSHKIN", color="#00bb00", lw=3)
plt.plot(times, gini_without, label="Pure Capitalism", color="#bb0000", lw=3)
plt.title(f"Simulation: 1 Quadrillion Supply\n(k correctly scaled)", fontsize=14)
plt.xlabel("Years")
plt.ylabel("Gini Coefficient")
plt.legend()
plt.grid(alpha=0.3)
plt.tight_layout()
plt.savefig("myshkin_fixed_k.png", dpi=100)
plt.show()

# Dividend Plot
plt.figure(figsize=(10,5))
plt.plot(div_times, dividend_yr, color="#006400", lw=3)
plt.title("Universal Dividend (Annualized)", fontsize=14)
plt.ylabel("Tokens / Year")
plt.xlabel("Years")
plt.grid(alpha=0.3)
plt.tight_layout()
plt.show()
```

```
print("\n" + "="*50)

print(f"Final Gini with $MYSHKIN :
{gini_with[-1]:.6f}")

print(f"Final Annual Dividend    :
{dividend_yr[-1]:,.0f}")

print("="*50)
```

Appendix B – Simulation B

```
import numpy as np

import matplotlib.pyplot as plt


# ==================================================

# MYSHKIN SIMULATION: EMERGENT GROWTH PROOF

# Growth is calculated by Market Participation, not hardcoded.

# ==================================================


# --- Parameters ---

N          = 10_000

years      = 2000          # Extended to see long-term compounding

dt         = 0.5

steps      = int(years / dt)

TOTAL_SUPPLY = 1_000_000_000_000_000 # 1 Quadrillion (Fixed)

D          = 100_000

k          = 6.34e-23*3.1563e7

mu_baseline  = 0.03


# --- Initial Distribution ---

np.random.seed(42)

xm     = D * 0.1

alpha  = 1.1

wealth0 = xm * (1 - np.random.rand(N)) ** (-1 / alpha)

wealth0 *= (TOTAL_SUPPLY / wealth0.sum())


# Initialize Worlds

tokens_myshkin = wealth0.copy()

tokens_cap     = wealth0.copy()

real_gdp_myshkin = TOTAL_SUPPLY

real_gdp_cap     = TOTAL_SUPPLY


# Baseline Participation (How many start with >= D)

initial_active = np.sum(wealth0 >= D)


# Storage

time_axis = []

rich_real_myshkin = []

rich_real_cap     = []

gdp_myshkin_hist  = []

gdp_cap_hist      = []


print(f"{'Year':<5} | {'Active (M)':<10} | {'Active (C)':<10} | {'GDP M (Log)':<12} | {'Rich M (Log)':<12}")
```

```
for step in range(steps):
    t = step * dt
    dW = np.random.normal(0, np.sqrt(dt), N)
    sigma = 0.20

    # --- World A: Pure Capitalism ---
    tokens_cap *= np.exp((-0.5*sigma**2)*dt + sigma*dW)
    tokens_cap *= (TOTAL_SUPPLY / tokens_cap.sum())

    # CALCULATED GROWTH: Growth is mu * % of people who can still afford products
    active_cap = np.sum(tokens_cap >= D)
    participation_cap = active_cap / initial_active
    growth_cap   = mu_baseline * participation_cap
    real_gdp_cap *= np.exp(growth_cap * dt)

    # --- World B: Myshkin ---
    tokens_myshkin *= np.exp((-0.5*sigma**2)*dt + sigma*dW)
    excess = np.maximum(tokens_myshkin - D, 0)
    tax    = k * (excess**2) * dt
    tax    = np.minimum(tax, tokens_myshkin)
    tokens_myshkin -= tax
    tokens_myshkin += (tax.sum() / N)
    tokens_myshkin = np.maximum(tokens_myshkin, 1e-5)
    tokens_myshkin *= (TOTAL_SUPPLY / tokens_myshkin.sum())

    # CALCULATED GROWTH: In Myshkin, participation stays near 100%
    active_myshkin = np.sum(tokens_myshkin >= D)
    participation_myshkin = active_myshkin / initial_active
    growth_myshkin = mu_baseline * participation_myshkin
    real_gdp_myshkin *= np.exp(growth_myshkin * dt)

    # Record Data
    if (step % int(10/dt)) == 0:
        time_axis.append(t)

rich_real_myshkin.append((np.max(tokens_myshkin)/TOTAL_SUPPLY) * real_gdp_myshkin)

rich_real_cap.append((np.max(tokens_cap)/TOTAL_SUPPLY) * real_gdp_cap)
```

```
gdp_myshkin_hist.append(real_gdp_myshkin)
        gdp_cap_hist.append(real_gdp_cap)

        if int(t) % 500 == 0:
            print(f"{int(t):<5} | {active_myshkin:<10} | {active_cap:<10} | {real_gdp_myshkin:.1e} | ${rich_real_myshkin[-1]:.1e}")

# ====================== PLOTTING ======================
plt.figure(figsize=(12, 10))

# Subplot 1: Real Wealth (LOG SCALE)
plt.subplot(2, 1, 1)
plt.plot(time_axis, rich_real_myshkin, color="#00aa00", linewidth=3, label="Richest (Myshkin)")
plt.plot(time_axis, rich_real_cap, color="#aa0000", linewidth=3, label="Richest (Capitalism)")
plt.yscale('log')
plt.title("Real Wealth of Richest Person (Emergent Growth Model)", fontsize=14)
plt.ylabel("Wealth (USD Log Scale)", fontsize=12)
plt.legend()
plt.grid(True, which="both", alpha=0.3)

# Subplot 2: GDP (LOG SCALE)
plt.subplot(2, 1, 2)
plt.plot(time_axis, gdp_myshkin_hist, color="#00aa00", linewidth=3, label="Myshkin Economy")
plt.plot(time_axis, gdp_cap_hist, color="#aa0000", linewidth=3, label="Capitalist Economy")
plt.yscale('log')
plt.title("Total Economic Output (Emergent Growth Model)", fontsize=14)
plt.ylabel("GDP (USD Log Scale)", fontsize=12)
plt.xlabel("Years", fontsize=12)
plt.legend()
plt.grid(True, which="both", alpha=0.3)

plt.tight_layout()
plt.savefig("myshkin_emergent_growth_2000y.png")
plt.show()
```