Dynamic Liquidity-Based Security Design

Emre Ozdenoren[1] Kathy Yuan[2] Shengxing Zhang[3]

[1]LBS and CEPR, [2]LSE and CEPR, [3]LSE

Motivation

- The determinants of liquidity in a dynamic economy
  - random productivity or endowment shocks
  - adverse selection
  - type of liquidity technology: optimal security design
- The amount of liquidity
  - repo contracts
  - haircuts, interest rates

Motivation

- The determinants of liquidity in a dynamic economy
  - random productivity or endowment shocks
  - adverse selection
  - type of liquidity technology: optimal security design
- The amount of liquidity
  - repo contracts
  - haircuts, interest rates

Key Mechanism

- Adverse selection
  - Without collateral borrowers cannot commit to paying back.
  - Productive assets provide liquidity because they can be used as collateral but are subject to adverse selection.
- Inter-temporal feedback
  - Collateral value depends on the re-sale value of the asset.
  - Re-sale value itself depends on the collateral value of the asset.
  - Leads to fragility and volatility in asset price and real output.

Key Mechanism

- Adverse selection
  - Without collateral borrowers cannot commit to paying back.
  - Productive assets provide liquidity because they can be used as collateral but are subject to adverse selection.
- Inter-temporal feedback
  - Collateral value depends on the re-sale value of the asset.
  - Re-sale value itself depends on the collateral value of the asset.
  - Leads to fragility and volatility in asset price and real output.

Main Results

- Equity contracts: Fragility and self-fulfilling
  - Pooling equilibrium: more liquidity and output
  - Separating equilibrium: less liquidity and output
  - Multiple equilibria
- Security design: liquid repo-debt contract (under monotone payoff constraints)
  - Unique equilibrium: both high and low types issue repo-debt and debt is liquid; low type issues equity
    - Eliminates fragility and improves liquidity
  - Improves social welfare relative to the separating equilibrium under equity contract

Main Results

- Equity contracts: Fragility and self-fulfilling
  - Pooling equilibrium: more liquidity and output
  - Separating equilibrium: less liquidity and output
  - Multiple equilibria
- Security design: liquid repo-debt contract (under monotone payoff constraints)
  - Unique equilibrium: both high and low types issue repo-debt and debt is liquid; low type issues equity
    - Eliminates fragility and improves liquidity
  - Improves social welfare relative to the separating equilibrium under equity contract

Basic and z-Technology

- Agent I: investor or supplier; Agent O: entrepreneur
- Agent O has a CRS z-technology which produces $z > 1$ units of consumption goods with one intermediate good (capital, equipment) from Agent I
- Agent I produces intermediate goods 1-to-1 from labor
- Both have a basic technology that produces consumption good 1-to-1 from labor
- Agent O would like to borrow unlimited amount of intermediate goods from agent I.
  - because returns to scale of z-technology is $z > 1$
- ... but agent O's promise to pay back is not enforceable

Basic and z-Technology

- Agent I: investor or supplier; Agent O: entrepreneur
- Agent O has a CRS z-technology which produces $z > 1$ units of consumption goods with one intermediate good (capital, equipment) from Agent I
- Agent I produces intermediate goods 1-to-1 from labor
- Both have a basic technology that produces consumption good 1-to-1 from labor
- Agent O would like to borrow unlimited amount of intermediate goods from agent I.
  - because returns to scale of z-technology is $z > 1$
- ... but agent O's promise to pay back is not enforceable

Basic and z-Technology

- Agent I: investor or supplier; Agent O: entrepreneur
- Agent O has a CRS z-technology which produces $z > 1$ units of consumption goods with one intermediate good (capital, equipment) from Agent I
- Agent I produces intermediate goods 1-to-1 from labor
- Both have a basic technology that produces consumption good 1-to-1 from labor
- Agent O would like to borrow unlimited amount of intermediate goods from agent I.
  - because returns to scale of z-technology is $z > 1$
- ... but agent O's promise to pay back is not enforceable

Basic and z-Technology

- Agent I: investor or supplier; Agent O: entrepreneur
- Agent O has a CRS z-technology which produces $z > 1$ units of consumption goods with one intermediate good (capital, equipment) from Agent I
- Agent I produces intermediate goods 1-to-1 from labor
- Both have a basic technology that produces consumption good 1-to-1 from labor
- Agent O would like to borrow unlimited amount of intermediate goods from agent I.
  - because returns to scale of z-technology is $z > 1$
- ... but agent O's promise to pay back is not enforceable

Utilities

- Utility in period t is $U_t(x, l) = x - l$
- x is the consumption good
- l is labor
- Discount rate between periods β, with $0 < \beta < 1$.

Timing

- Three dates in each period.
- Date 1: Intermediate good is produced
  - perishes at the end of the period
  - no direct utility
- Date 2: Consumption good is produced
  - via the productive or basic technology.
- Date 3: Consumption takes place.
  - Consumption good perishes at the end of the period.

Timing

- Three dates in each period.
- Date 1: Intermediate good is produced
  - perishes at the end of the period
  - no direct utility
- Date 2: Consumption good is produced
  - via the productive or basic technology.
- Date 3: Consumption takes place.
  - Consumption good perishes at the end of the period.

Timing

- Three dates in each period.
- Date 1: Intermediate good is produced
  - perishes at the end of the period
  - no direct utility
- Date 2: Consumption good is produced
  - via the productive or basic technology.
- Date 3: Consumption takes place.
  - Consumption good perishes at the end of the period.

Timing

- Three dates in each period.
- Date 1: Intermediate good is produced
  - perishes at the end of the period
  - no direct utility
- Date 2: Consumption good is produced
  - via the productive or basic technology.
- Date 3: Consumption takes place.
  - Consumption good perishes at the end of the period.

Asset

- Long lived asset pays s units of dividend as consumption good at date 3.
- Fixed supply of the asset is A.
- With prob. λ dividend distribution is F_L and $1-\lambda$ it is F_H.
  - $F_L, F_H \in \Delta[s_L, s_H]$, $0 \leq s_L < s_H$
  - F_H first order stochastically dominates F_L
  - Quality $Q \in \{H, L\}$ is i.i.d. over time
  - $\widetilde{F}_Q(s) = 1 - F_Q(s)$ for $Q \in \{L, H\}$

Asset

- Long lived asset pays s units of dividend as consumption good at date 3.
- Fixed supply of the asset is A.
- With prob. λ dividend distribution is F_L and $1-\lambda$ it is F_H.
  - $F_L, F_H \in \Delta[s_L, s_H]$, $0 \leq s_L < s_H$
  - F_H first order stochastically dominates F_L
  - Quality $Q \in \{H, L\}$ is i.i.d. over time
  - $\widetilde{F}_Q(s) = 1 - F_Q(s)$ for $Q \in \{L, H\}$

Asset

- Long lived asset pays s units of dividend as consumption good at date 3.
- Fixed supply of the asset is A.
- With prob. λ dividend distribution is F_L and $1-\lambda$ it is F_H.
  - $F_L, F_H \in \Delta[s_L, s_H]$, $0 \leq s_L < s_H$
  - F_H first order stochastically dominates F_L
  - Quality $Q \in \{H, L\}$ is i.i.d. over time
  - $\widetilde{F}_Q(s) = 1 - F_Q(s)$ for $Q \in \{L, H\}$

Collateral Asset and Adverse Selection

- Agent O uses the asset as collateral to borrow intermediate goods from agent I.
- Agent O privately observes asset quality O at the beginning of each period.
  - Adverse selection is within the period
  - Shown later, agent O purchases all collateral assets in equilibrium
  - Privately informed about the quality because of
    - opportunity to temper with quality
    - incentive to acquire private information

Collateral Asset and Adverse Selection

- Agent O uses the asset as collateral to borrow intermediate goods from agent I.
- Agent O privately observes asset quality O at the beginning of each period.
  - Adverse selection is within the period
  - Shown later, agent O purchases all collateral assets in equilibrium
  - Privately informed about the quality because of
    - opportunity to temper with quality
    - incentive to acquire private information

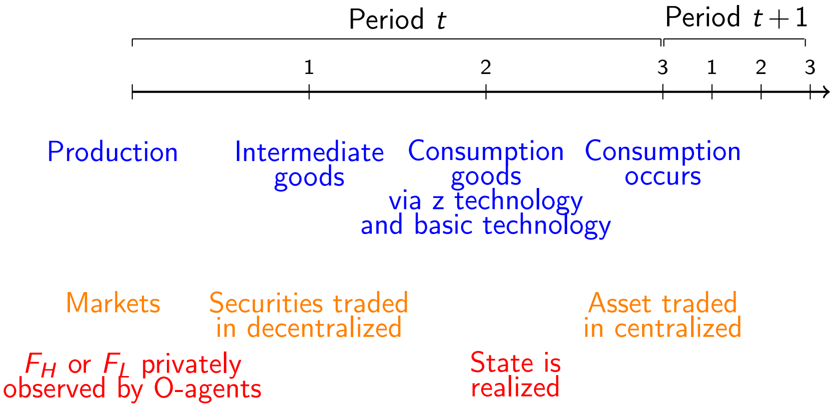
Trading Environment: Two Markets

- Markets for intermediate goods at date 1
  - An agent O randomly meets at least two agent Is
  - *in decentralized* market(s)
  - intermediate goods are traded for asset-based securities
  - i.e, borrowing against some forms of securities takes place
- Market for the collateral asset at date 3
  - After state is realized, asset price, ϕ_t, is determined
  - in a *centralized* market
  - ϕ_t= present value of all the future cash flows of the asset.

Trading Environment: Two Markets

- Markets for intermediate goods at date 1
  - An agent *O* randomly meets at least two agent *I*s
  - *in decentralized* market(s)
  - intermediate goods are traded for asset-based securities
  - i.e, borrowing against some forms of securities takes place
- Market for the collateral asset at date 3
  - After state is realized, asset price, ϕ_t, is determined
  - in a *centralized* market
  - $\phi_t =$ present value of all the future cash flows of the asset.

Timeline


Period t
Period $t+1$
1
2
3
1
2
3
Production
Intermediate goods
Consumption goods via z technology and basic technology
Consumption occurs
Markets
Securities traded in decentralized
Asset traded in centralized
F_H or F_L privately observed by O-agents
State is realized

Security Design

- Security Design is conducted ex ante before types are realised.
- An asset-backed security $y^j(s)$ is a state-contingent promise of consumption goods at date 3.
- Two cases of interest:
  - Equity $y(s) = s + \phi_t, \forall s \in [s_L, s_H]$
  - Set of monotone securities $\mathscr{I}_t(\phi_t) = \{y : y(s) \text{ increasing in } s, y(s) \leq s + \phi_t, \forall s \in [s_L, s_H]\}$

Security Design

- Security Design is conducted ex ante before types are realised.
- An asset-backed security $y^j(s)$ is a state-contingent promise of consumption goods at date 3.
- Two cases of interest:
  - Equity $y(s) = s + \phi_t, \forall s \in [s_L, s_H]$
  - Set of monotone securities $\mathscr{I}_t(\phi_t) = \{y : y(s) \text{ increasing in } s, y(s) \leq s + \phi_t, \forall s \in [s_L, s_H]\}$

Security Design

- Security Design is conducted ex ante before types are realised.
- An asset-backed security $y^j(s)$ is a state-contingent promise of consumption goods at date 3.
- Two cases of interest:
  - Equity $y(s) = s + \phi_t, \forall s \in [s_L, s_H]$
  - Set of monotone securities $\mathscr{I}_t(\phi_t) = \{y : y(s) \text{ increasing in } s, y(s) \leq s + \phi_t, \forall s \in [s_L, s_H]\}$

Equilibrium in Security j's Market

- Security trading occurs at date 1:
  - bilaterally between agent O and multiple agent Is
  - in dedicated sub-markets for each available security.
- Suppose agent I bids per-unit price q_t^j for security j.
- If highest bid, agent O offers him $a_t^Q(j)$ units of security j for $q_t^j a_t^Q(j)$ intermediate goods.
- In equilibrium, winning bid q_t^j
  - agent I: zero expected gain due to Bertrand Competition
  - IC for agent O: profitable for an informed O agent type

Equilibrium in Security j's Market

- Security trading occurs at date 1:
  - bilaterally between agent O and multiple agent Is
  - in dedicated sub-markets for each available security.
- Suppose agent I bids per-unit price q_t^j for security j.
- If highest bid, agent O offers him $a_t^Q(j)$ units of security j for $q_t^j a_t^Q(j)$ intermediate goods.
- In equilibrium, winning bid q_t^j
  - agent I: zero expected gain due to Bertrand Competition
  - IC for agent O: profitable for an informed O agent type

Equilibrium in Security j's Market

- Security trading occurs at date 1:
  - bilaterally between agent O and multiple agent Is
  - in dedicated sub-markets for each available security.
- Suppose agent I bids per-unit price q_t^j for security j.
- If highest bid, agent O offers him $a_t^Q(j)$ units of security j for $q_t^j a_t^Q(j)$ intermediate goods.
- In equilibrium, winning bid q_t^j
  - agent I: zero expected gain due to Bertrand Competition
  - IC for agent O: profitable for an informed O agent type

Equilibrium in Security j's Market

- Adverse selection index: higher R_t^j, lower adverse selection

$$R_t^j \equiv \frac{E_L y_t^j}{E_H y_t^j}$$

- Expected value of security j when both O types participate

$$\overline{q}^j = \lambda E_L y_t^j + (1-\lambda) E_H y_t^j$$

- High O type participates if adverse selection is low:

$$z\overline{q}^j - E_H y_t^j \gtreqqless 0 \text{iff} R \gtreqqless \zeta$$

where $\zeta \equiv 1-(z-1)/\lambda z$

Equilibrium Price in Security j's Market

- If $R_t^j > \zeta$, both high and low O types sell
  - $q_t^j = \lambda E_L y_t^j + (1-\lambda) E_H y_t^j$
  - $a_t^L(q_t^j) = a_t^H(q_t^j) = a$.
- If $R_t^j < \zeta$, only low type sells security
  - $q_t^j = E_L y_t^j$
  - $a_t^L(q_t^j) = a$ and $a_t^H(q_t^j) = 0$.

Security Design: Objective

Before learning asset quality, agent O chooses security design $\mathscr{J}_t(\phi_t) \subseteq \mathscr{I}_t(\phi_t)$ to maximize

$$
\begin{aligned}
V_{o,t}(a) = {} & \lambda \int \left(\sum_{j \in \mathscr{J}_t(\phi_t)} a_t^L(j) \left[z q_t^j - y_t^j(s) \right] \right) dF_L(s) \\
& + (1-\lambda) \int \left(\sum_{j \in \mathscr{J}_t(\phi_t)} a_t^H(j) \left[z q_t^j - y_t^j(s) \right] \right) dF_H(s) \\
& + \int a(s+\phi_t)\, d\left[\lambda F_L(s) + (1-\lambda) F_H(s) \right]
\end{aligned}
$$

Security Design: Constraints

- Each O type optimally chooses how much to supply:
  - Low type O agent always sells all since $E_L y_t^j(s) \leq E_H y_t^j(s)$

$$a_t^L(j) = a \text{ and } a_t^H(j) = \begin{cases} a & \text{if } R_t^j \geq \zeta \\ 0 & \text{if } R_t^j < \zeta \end{cases}$$

- The security design must be overall feasible

$$\sum_{j \in \mathscr{J}_t(\phi_t)} y_t^j(s) d\mu_{o,t} \leq s + \phi_t$$

- Price is determined via Bertrand competition

$$q_t^j = \begin{cases} \lambda E_L y_t^j + (1-\lambda) E_H y_t^j & \text{if } R_t^j \geq \zeta \\ E_L y_t^j & \text{if } R_t^j < \zeta \end{cases}$$

Security Design: Constraints

- Each O type optimally chooses how much to supply:
  - Low type O agent always sells all since $E_L y_t^j(s) \leq E_H y_t^j(s)$

$$a_t^L(j) = a \text{ and } a_t^H(j) = \begin{cases} a & \text{if } R_t^j \geq \zeta \\ 0 & \text{if } R_t^j < \zeta \end{cases}$$

- The security design must be overall feasible

$$\sum_{j \in \mathscr{J}_t(\phi_t)} y_t^j(s) d\mu_{o,t} \leq s + \phi_t$$

- Price is determined via Bertrand competition

$$q_t^j = \begin{cases} \lambda E_L y_t^j + (1-\lambda) E_H y_t^j & \text{if } R_t^j \geq \zeta \\ E_L y_t^j & \text{if } R_t^j < \zeta \end{cases}$$

Security Design: Constraints

- Each O type optimally chooses how much to supply:
  - Low type O agent always sells all since $E_L y_t^j(s) \leq E_H y_t^j(s)$

$$a_t^L(j) = a \text{ and } a_t^H(j) = \begin{cases} a & \text{if } R_t^j \geq \zeta \\ 0 & \text{if } R_t^j < \zeta \end{cases}$$

- The security design must be overall feasible

$$\sum_{j \in \mathscr{J}_t(\phi_t)} y_t^j(s) d\mu_{o,t} \leq s + \phi_t$$

- Price is determined via Bertrand competition

$$q_t^j = \begin{cases} \lambda E_L y_t^j + (1-\lambda) E_H y_t^j & \text{if } R_t^j \geq \zeta \\ E_L y_t^j & \text{if } R_t^j < \zeta \end{cases}$$

Dynamic Security Design Equilibrium

A stationary dynamic equilibrium consists of

- $\mathscr{J}_t(\phi_t)$ solves the security design problem
- security price q_t^j satisfies the submarket Bertrand competition
- asset price ϕ_t solves the Euler equation given by:

$$\phi_t = \beta \left[z \left(\sum_{j \in P_t} q_t^j + \lambda \sum_{j \in \mathscr{J}_t(\phi_t) \setminus P_t} q_t^j \right) + (1-\lambda) \sum_{j \in \mathscr{J}_t(\phi_t) \setminus P_t} E_H y_t^j \right]$$

where $j \in P_t$ iff $R_t^j \geq \zeta$.

Benchmark: Dynamic Lemons Market

- Collateral asset is the only security: No security design
- security price depends on $\frac{E_L s+\phi_t}{E_H s+\phi_t} \gtreqless \zeta$:
  - Pooling: $q_t^P = \phi_t + \lambda E_L s + (1-\lambda) E_H s$
  - Separating: $q_t^S = \phi_t + E_L s$ otherwise.
- asset price depends on $\frac{E_L s+\phi_t}{E_H s+\phi_t} \gtreqless \zeta$:
  - Pooling: $\phi^P = \beta z q_t^P = \frac{\beta z(\lambda E_L s+(1-\lambda)E_H s)}{1-\beta z}$
  - Separating: $\phi^S = \beta\left[z\lambda q_t^S + (1-\lambda)\left(\phi^S + E_H s\right)\right] = \frac{\beta[\lambda z E_L s+(1-\lambda)E_H s]}{1-\beta(\lambda z+1-\lambda)}$
- $\phi^P > \phi^S > PV = \frac{\beta[\lambda E_L s+(1-\lambda)E_H s]}{1-\beta}$

Benchmark: Dynamic Lemons Market

- Collateral asset is the only security: No security design
- security price depends on $\frac{E_L s+\phi_t}{E_H s+\phi_t} \gtreqless \zeta$:
  - Pooling: $q_t^P = \phi_t + \lambda E_L s + (1-\lambda) E_H s$
  - Separating: $q_t^S = \phi_t + E_L s$ otherwise.
- asset price depends on $\frac{E_L s+\phi_t}{E_H s+\phi_t} \gtreqless \zeta$:
  - Pooling: $\phi^P = \beta z q_t^P = \frac{\beta z(\lambda E_L s+(1-\lambda)E_H s)}{1-\beta z}$
  - Separating: $\phi^S = \beta \left[z\lambda q_t^S + (1-\lambda)\left(\phi^S + E_H s\right)\right] = \frac{\beta[\lambda z E_L s+(1-\lambda)E_H s]}{1-\beta(\lambda z+1-\lambda)}$
- $\phi^P > \phi^S > PV = \frac{\beta[\lambda E_L s+(1-\lambda)E_H s]}{1-\beta}$

Benchmark: Dynamic Lemons Market

- Collateral asset is the only security: No security design
- security price depends on $\frac{E_L s+\phi_t}{E_H s+\phi_t} \gtreqless \zeta$:
  - Pooling: $q_t^P = \phi_t + \lambda E_L s + (1-\lambda) E_H s$
  - Separating: $q_t^S = \phi_t + E_L s$ otherwise.
- asset price depends on $\frac{E_L s+\phi_t}{E_H s+\phi_t} \gtreqless \zeta$:
  - Pooling: $\phi^P = \beta z q_t^P = \frac{\beta z(\lambda E_L s+(1-\lambda)E_H s)}{1-\beta z}$
  - Separating: $\phi^S = \beta\left[z\lambda q_t^S + (1-\lambda)\left(\phi^S + E_H s\right)\right] = \frac{\beta[\lambda z E_L s+(1-\lambda)E_H s]}{1-\beta(\lambda z+1-\lambda)}$
- $\phi^P > \phi^S > PV = \frac{\beta[\lambda E_L s+(1-\lambda)E_H s]}{1-\beta}$

Benchmark: Dynamic Lemons Market

- Collateral asset is the only security: No security design
- security price depends on $\frac{E_L s+\phi_t}{E_H s+\phi_t} \gtreqless \zeta$:
  - Pooling: $q_t^P = \phi_t + \lambda E_L s + (1-\lambda) E_H s$
  - Separating: $q_t^S = \phi_t + E_L s$ otherwise.
- asset price depends on $\frac{E_L s+\phi_t}{E_H s+\phi_t} \gtreqless \zeta$:
  - Pooling: $\phi^P = \beta z q_t^P = \frac{\beta z(\lambda E_L s+(1-\lambda)E_H s)}{1-\beta z}$
  - Separating: $\phi^S = \beta \left[z \lambda q_t^S + (1-\lambda) \left(\phi^S + E_H s \right) \right] = \frac{\beta[\lambda z E_L s+(1-\lambda)E_H s]}{1-\beta(\lambda z+1-\lambda)}$
- $\phi^P > \phi^S > PV = \frac{\beta[\lambda E_L s+(1-\lambda)E_H s]}{1-\beta}$

Fragility of Dynamic Lemons Market

- There can be multiple equilibria in a dynamic lemons market.
- Occurs when $\frac{E_L s+\phi^S}{E_H s+\phi^S} < \zeta \leq \frac{E_L s+\phi^P}{E_H s+\phi^P}$.
- Plugging for ϕ_S and ϕ_P we obtain the condition for multiplicity as

$$\frac{\zeta-\beta}{1-\beta} < \frac{E_L s}{E_H s} < \frac{\zeta-\beta\left[1-(1-\lambda)(z-1)\right]}{1-\beta\left[1-(1-\lambda)(z-1)\right]}$$

- Easy to see that for intermediate values of $E_L s/E_H s$ both equilibria exist.

Fragility of Dynamic Lemons Market

- There can be multiple equilibria in a dynamic lemons market.
- Occurs when $\frac{E_L s+\phi^S}{E_H s+\phi^S} < \zeta \leq \frac{E_L s+\phi^P}{E_H s+\phi^P}$.
- Plugging for ϕ_S and ϕ_P we obtain the condition for multiplicity as
$$\frac{\zeta-\beta}{1-\beta} < \frac{E_L s}{E_H s} < \frac{\zeta-\beta\left[1-(1-\lambda)(z-1)\right]}{1-\beta\left[1-(1-\lambda)(z-1)\right]}$$
- Easy to see that for intermediate values of $E_L s/E_H s$ both equilibria exist.

Fragility of Dynamic Lemons Market

- There can be multiple equilibria in a dynamic lemons market.
- Occurs when $\frac{E_L s+\phi^S}{E_H s+\phi^S} < \zeta \leq \frac{E_L s+\phi^P}{E_H s+\phi^P}$.
- Plugging for ϕ_S and ϕ_P we obtain the condition for multiplicity as

$$\frac{\zeta-\beta}{1-\beta} < \frac{E_L s}{E_H s} < \frac{\zeta-\beta\left[1-(1-\lambda)(z-1)\right]}{1-\beta\left[1-(1-\lambda)(z-1)\right]}$$

- Easy to see that for intermediate values of $E_L s/E_H s$ both equilibria exist.

Fragility of Dynamic Lemons Market

- There can be multiple equilibria in a dynamic lemons market.
- Occurs when $\frac{E_L s+\phi^S}{E_H s+\phi^S} < \zeta \leq \frac{E_L s+\phi^P}{E_H s+\phi^P}$.
- Plugging for ϕ_S and ϕ_P we obtain the condition for multiplicity as

$$\frac{\zeta-\beta}{1-\beta} < \frac{E_L s}{E_H s} < \frac{\zeta-\beta\left[1-(1-\lambda)(z-1)\right]}{1-\beta\left[1-(1-\lambda)(z-1)\right]}$$

- Easy to see that for intermediate values of $E_L s/E_H s$ both equilibria exist.

Intuition for Dynamic Multiplicity

- There is a dynamic feedback loop.
- If agents anticipate the asset to be traded in a pooling eqm in the decentralized market, then price is high.
- In turn, when the price is high, the H-type O agent is willing to pool.
- If agents anticipate the asset to be traded in a separating eqm in the decentralized market, price is low.
- In turn, when the price is low, the H-type keeps the asset.

Intuition for Dynamic Multiplicity

- There is a dynamic feedback loop.
- If agents anticipate the asset to be traded in a pooling eqm in the decentralized market, then price is high.
- In turn, when the price is high, the H-type O agent is willing to pool.
- If agents anticipate the asset to be traded in a separating eqm in the decentralized market, price is low.
- In turn, when the price is low, the H-type keeps the asset.

Intuition for Dynamic Multiplicity

- There is a dynamic feedback loop.
- If agents anticipate the asset to be traded in a pooling eqm in the decentralized market, then price is high.
- In turn, when the price is high, the H-type O agent is willing to pool.
- If agents anticipate the asset to be traded in a separating eqm in the decentralized market, price is low.
- In turn, when the price is low, the H-type keeps the asset.

Intuition for Dynamic Multiplicity

- There is a dynamic feedback loop.
- If agents anticipate the asset to be traded in a pooling eqm in the decentralized market, then price is high.
- In turn, when the price is high, the H-type O agent is willing to pool.
- If agents anticipate the asset to be traded in a separating eqm in the decentralized market, price is low.
- In turn, when the price is low, the H-type keeps the asset.

Optimal Security Design

- We call a security traded in a pooling equilibrium in the decentralized market a liquid security.
- First we show that Agent O is weakly better-off selling only one liquid security.
- This is because if two securities are both liquid, combination is also liquid and generates at least as much value for Agent O.
- Also if security design is optimal, the feasibility constraint is binding.
- W.l.o.g. can restrict attention to a liquid security $y(s)$ and an illiquid one $s + \phi - y(s)$.

Optimal Security Design

- We call a security traded in a pooling equilibrium in the decentralized market a liquid security.
- First we show that Agent O is weakly better-off selling only one liquid security.
- This is because if two securities are both liquid, combination is also liquid and generates at least as much value for Agent O.
- Also if security design is optimal, the feasibility constraint is binding.
- W.l.o.g. can restrict attention to a liquid security $y(s)$ and an illiquid one $s + \phi - y(s)$.

Optimal Security Design

- We call a security traded in a pooling equilibrium in the decentralized market a liquid security.
- First we show that Agent O is weakly better-off selling only one liquid security.
- This is because if two securities are both liquid, combination is also liquid and generates at least as much value for Agent O.
- Also if security design is optimal, the feasibility constraint is binding.
- W.l.o.g. can restrict attention to a liquid security $y(s)$ and an illiquid one $s+\phi-y(s)$.

Optimal Security Design

- We call a security traded in a pooling equilibrium in the decentralized market a liquid security.
- First we show that Agent O is weakly better-off selling only one liquid security.
- This is because if two securities are both liquid, combination is also liquid and generates at least as much value for Agent O.
- Also if security design is optimal, the feasibility constraint is binding.
- W.l.o.g. can restrict attention to a liquid security $y(s)$ and an illiquid one $s + \phi - y(s)$.

Optimal Security Design

- We call a security traded in a pooling equilibrium in the decentralized market a liquid security.
- First we show that Agent O is weakly better-off selling only one liquid security.
- This is because if two securities are both liquid, combination is also liquid and generates at least as much value for Agent O.
- Also if security design is optimal, the feasibility constraint is binding.
- W.l.o.g. can restrict attention to a liquid security $y(s)$ and an illiquid one $s + \phi - y(s)$.

Optimality of Debt

Proposition

Assume that $\frac{f_L(s)}{f_H(s)}$ *is decreasing in* s*. The optimal security is a unique standard debt contract* y_D *such that*

$$y_D(s) = \phi + \min(s, s^*),$$

for some $s^* \in (s_L, s_H)$.

Characterizing the Debt Contract

Proposition

Assume that $\frac{f_L(s)}{f_H(s)}$ is decreasing in s.

- If $\frac{E_L s}{E_H s} < 1 - \frac{z-1}{z}\frac{1}{\lambda(1-\beta)}$,
  - ie, the separating region in the dynamic lemons market
  - a unique equilibrium and non-trivial tranching with $D \in (s_L, s_H)$ and ϕ solve:

$$\phi = \frac{z}{z-1}\lambda \int_{s_L}^{D} \left[\widetilde{F}_H(s) - \widetilde{F}_L(s)\right] ds - \int_{s_L}^{D} \widetilde{F}_H(s) ds - s_L \qquad (1)$$

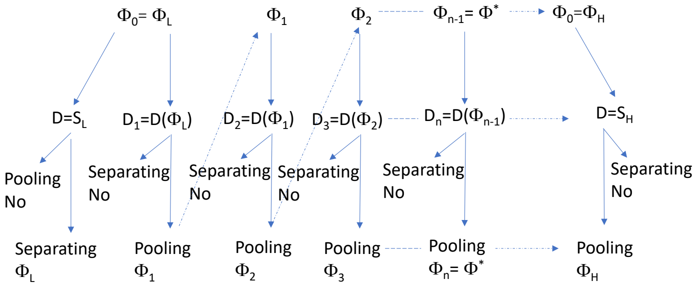
$$\phi = \frac{\beta}{1-\beta z}\left\{z\left[\lambda E_L s + (1-\lambda)E_H s\right] - (1-\lambda)(z-1)\int_{D}^{s_H} \widetilde{F}_H(s) ds\right\} \qquad (2)$$

- Otherwise, a "pass-through security" that promises the entire value of the asset and replicates the pooling equilibrium in dynamic lemons market
  - $D = s_H$ and $\phi = \frac{\beta}{1-\beta z} z\left[\lambda E_L s + (1-\lambda)E_H s\right]$.

Discussions on Liquidity and Fragility

- We show that security design equilibrium Pareto dominates all equilibria of the case in dynamic lemons market
  - more liquidity, more real output and less fragility
  - even if only issue a "pass-through security" that mimics equity – replicate the pooling

Eliminates Low Liquidity Equilibrium



Discussions on Fragility and Robustness

- Unravelling results when security design option is introduced.
  - Suppose low asset price,
  - tranche a small senior liquid debt, asset price $\uparrow$, which allows more liquid tranching $D \uparrow$, which leads to asset price $\uparrow$, ... converges to optimal.

Repo Features

- Face value: $D + \phi$
- Repo rate: $\frac{D+\phi-q_D}{q_D}$
- Haircut: equity tranch, q_E

Conclusion

Optimal security design in a dynamic lemons market

- Unique equilibrium: both high and low types issue repo-debt and debt is liquid; low type issues equity
- Eliminates fragility and improves liquidity
- Improves social welfare relative to the separating equilibrium under equity contract
- Endogenous amplification of shocks to asset quality and productivity