# *How* Optimal Banking Contracts Tolerate Runs

Karl Shell    Yu Zhang

Cornell University    Xiamen University

CU/PSU Macroeconomics Conference

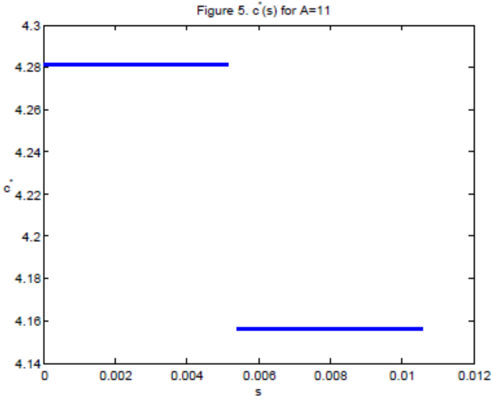
Sep. 26, 2015

# Introduction

- Bryant (1980) and Diamond and Dybvig (1983): "bank runs" in the *post-deposit* game
- Peck and Shell (2003): A *sunspot-driven* run can be an equilibrium in the *pre-deposit* game for sufficiently small run probability.

# Introduction

- For the 2-depositor banking model, the optimal contract is defined by $c$ – the withdrawal of the first in line in period 1.



# Introduction

- Shouldn't $c^*(s)$ become more conservative (i.e., strictly decreasing) in $s$ before it switches to the best run-proof contract?
- If yes, in which economies will we have this property and in which economies is $c^*(s)$ a step function?
- These issues are important to banks and regulators. Also important to the theory of SSE.
- Instead of relying solely on numerical examples, we provide the global comparative statics for this economy.

# The Model: Consumers

- 2 ex-ante identical vNM consumers and 3 periods: 0, 1 and 2.
- Endowments: $y$
- Preferences: $u(c^1)$ and $v(c^1 + c^2)$

$$u(x) = A\frac{(x)^{1-a}}{1-a}, \text{ where } A > 0 \text{ and } a > 1.$$

$$v(x) = \frac{(x)^{1-b}}{1-b}, \ b > 1.$$

- Types are uncorrelated (so we have aggregate uncertainty.): $p$

# The Model: Technology

- Bank Portfolio:

| $t = 0$ | $t = 1$ | $t = 2$ |
|---|---|---|
| $-1$ | $1$ | $0$ |
| $-1$ | $0$ | $R$ |

- Consumer storage option

# The Model

- Sequential service constraint (Wallace (1988))
- Suspension of convertibility.
- A depositor visits the bank only when he makes withdrawals.
- When a depositor makes his withdrawal decision, he does not know his position in the bank queue.
- If more than one depositor chooses to withdraw, a depositor's position in the queue is random. Positions in the queue are equally probable.

# Post-Deposit Game: Notation

- $c \in [0, 2y]$ is any feasible banking contract
- $\widehat{c} \in [0, 2y]$ is the unconstrained optimal banking contract
- $c^* \in [0, 2y]$ is the constrained optimal banking contract

## c

*early*

- A patient depositor chooses early withdrawal when he expects the other depositor, if patient, to also choose early withdrawal.

$$[v(c) + v(2y - c)]/2 > v[(2y - c)R]$$

- Let $c^{early}$ be the value of $c$ such that the above inequality holds as an equality. $c^{early}$ is the best run-proof $c$.

## c

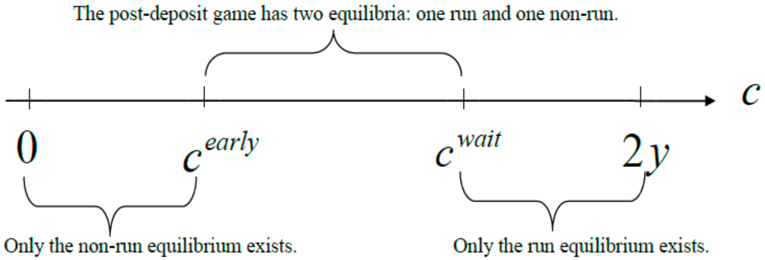
*wait*

- A patient depositor chooses late withdrawal when he expects the other depositor, if patient, to also choose late withdrawal. (ICC)

$$pv[(2y-c)R]+(1-p)v(yR) \geq p[v(c)+v(2y-c)]/2+(1-p)v(c).$$

- Let $c^{wait}$ be the value of $c$ such that the above inequality holds as an equality.

# Post-Deposit Game



# Post-Deposit Game

- $c^{early} < c^{wait}$ if and only if

$$b < \min\{2, 1 + \ln 2 / \ln R\}$$

- When $b$ and $R$ satisfy the above inequality, bank runs matter in the *post-deposit* game for $c \in (c^{early}, c^{wait}]$.
- When $b$ and $R$ don't satisfy the above inequality, $c^{early} \geq c^{wait}$, which implies that any implementable allocation is strongly implementable; hence bank runs do not matter.

# Pre-Deposit Game

- Whether bank runs occur in the *pre-deposit* game depends on whether the optimal contract $c^*$ belongs to the set $(c^{early}, c^{wait}]$.
- To characterize the optimal contract, we divide the problem into three cases depending on $\widehat{c}$, the contract supporting the *unconstrained efficient allocation*.
  - $\widehat{c} \leq c^{early}$ (Case 1)
  - $\widehat{c} \in (c^{early}, c^{wait}]$ (Case 2)
  - $\widehat{c} > c^{wait}$ (Case 3)

# Impulse parameter A and the 3 cases

- $\widehat{c}$ is the $c$ in $[0, 2y]$ that maximizes

$$\widehat{W}(c) = p^2[u(c) + u(2y - c)] + 2p(1 - p)[u(c) + v[(2y - c)R]] \\ + 2(1 - p)^2 v(yR).$$

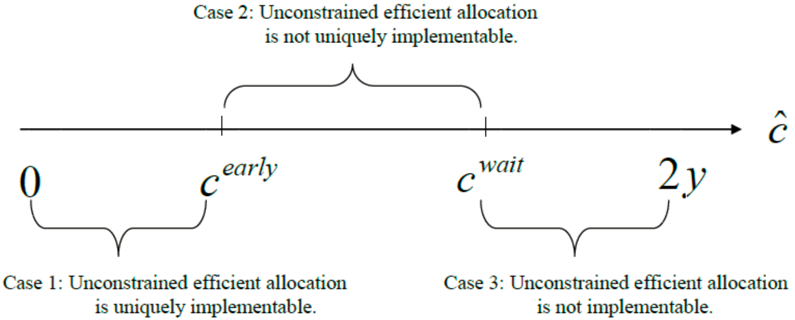
-

$$\widehat{c} = \frac{2y}{\{p/(2 - p) + 2(1 - p)/[(2 - p)AR^{b-1}]\}^{1/b} + 1}.$$

- $\widehat{c}(A)$ is an increasing function of $A$.

# Parameter A and the 3 Cases

- Neither $c^{early}$ nor $c^{wait}$ depends on $A$



# Example

- The parameters are

$$a = b = 1.01; p = 0.5; y = 3; R = 1.5$$

- We see that $b$ and $R$ satisfy the condition which makes the set of contracts permiting strategic complementarity non-empty. We have that $c^{early} = 4.155955$ and $c^{wait} = 4.280878$.
- $A_1 = 6.217686$ and $A_2 = 10.277988$.
- If $A \leq A_1$, we are in Case 1; If $A_1 < A \leq A_2$, we are in Case 2; If $A > A_2$, we are in Case 3.

# The Optimal Contract

- $$c^*(s) = \arg \max_{c \in [0, c^{wait}]} W(c; s),$$

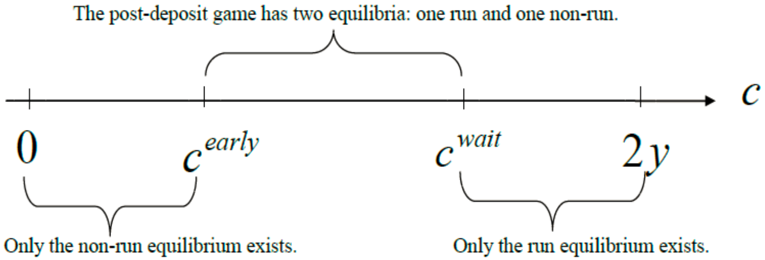
  where

$$W(c;s) = \begin{cases} \widehat{W}(c) & \text{if } c \leq c^{early}. \\ (1-s)\widehat{W}(c) + sW^{run}(c) & \text{if } c^{early} < c \leq c^{wait}. \end{cases}$$

and

$$\begin{aligned} W^{run}(c) = {} & p^2[u(c) + u(2y-c)] \\ & + p(1-p)[u(c) + v(2y-c) + v(c) + u(2y-c)] \\ & + (1-p)^2[v(c) + v(2y-c)]. \end{aligned}$$

# The Optimal Contract



# The Optimal Contract: Case 1

- Case 1: The *unconstrained efficient allocation* is strongly implementable, i.e., $\widehat{c} \leq c^{early}$.
- It is straightforward to see that the optimal contract for the *pre-deposit* game supports the *unconstrained efficient allocation*

$$c^*(s) = \widehat{c}.$$

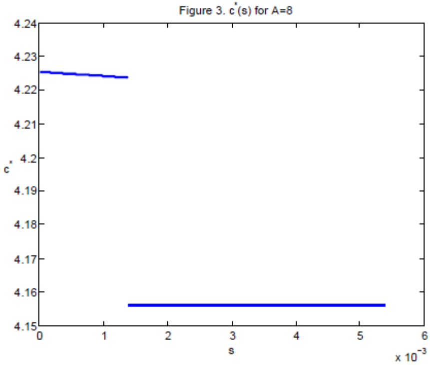
and that the optimal contract doesn't tolerate runs.

# The Optimal Contract: Case 2

- Case 2: The *unconstrained efficient allocation* is weakly implementable, i.e., $c^{early} < \widehat{c} \leq c^{wait}$.
- The optimal contract $c^*(s)$ satisfies: (1) if $s$ is larger than the threshold probability $s_0$, the optimal contract is run-proof and $c^*(s) = c^{early}$. (2) if $s$ is smaller than $s_0$, the optimal contract $c^*(s)$ tolerates runs and it is a strictly decreasing function of $s$.

# The Optimal Contract: Case 2

- Using the same parameters as the previous example. Let $A = 8$. (We have seen that we are in Case 2 if $6.217686 < A \leq 10.277988$.)
- $c^*$ switches to the best run-proof contract (i.e. $c^{early}$) when $s > s_0 = 0.001382358$

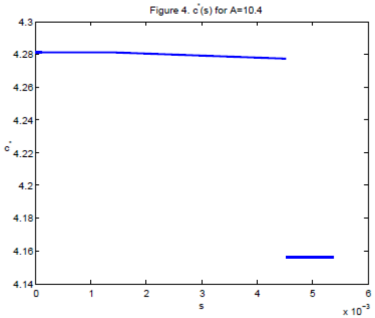


# The Optimal Contract: Case 3

- Case 3: The *unconstrained efficient allocation* is not implementable, i.e., $c^{wait} < \widehat{c}$.
- The optimal contract $c^*(s)$ satisfies: (1) If $s$ is larger than the threshold probability $s_1$, we have $c^*(s) = c^{early}$ and the optimal contract is run-proof. (2) If $s$ is smaller than $s_1$, the optimal contract $c^*(s)$ tolerates runs and it is a weakly decreasing function of $s$. Furthermore, we have $c^*(s) = c^{wait}$ for at least part of the run tolerating range of $s$.

# The Optimal Contract: Case 3

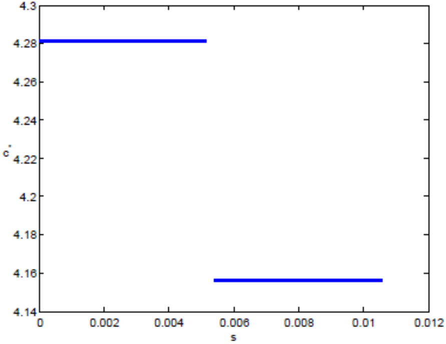
- Using the same parameters as in the previous example. Let $A = 10.4$. (We have seen that we are in Case 2 if $A > 10.277988$.)
- $c^*$ switches to the best run-proof (i.e. $c^{early}$) when $s > 0.004524181$.
- ICC becomes non-binding when $s \geq 0.001719643$.



# The Optimal Contract: Case 3

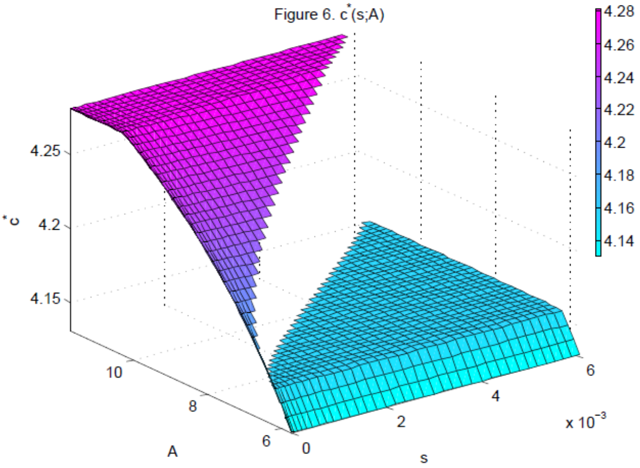
- Let $A = 11$. (PS case)
- $c^*$ switches to the best run-proof (i.e. $c^{early}$) when $s >$ 0.005281242.

Figure 5. $c^*(s)$ for A=11



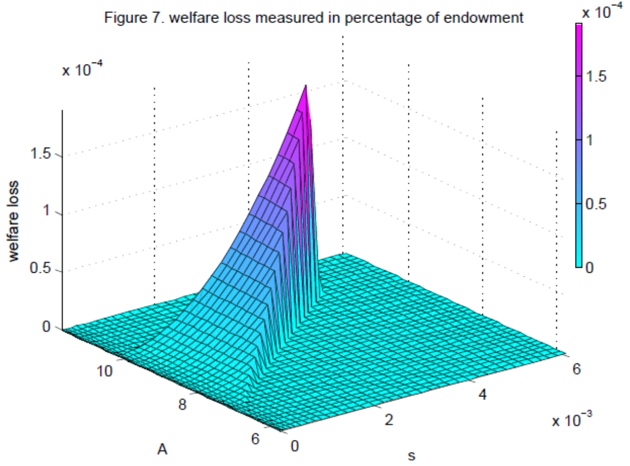
# The Optimal Contract

- $c^*$ versus $s$ and $A$



# The Optimal Contract

- welfare loss from using the corresponding optimal bang-bang contract instead of $c^*(s)$



# Summary and Concluding Remark

- The general form of the optimal contract to the *pre-deposit* game is analyzed.
- The *unconstrained efficient allocation* falls into one of the three cases:
  - (1) strongly implementable
  - (2) weakly implementable
  - (3) not implementable.

# Summary and Concluding Remark

- In Cases 2 and 3, the optimal contract tolerates runs when the run probability is sufficiently small:
- In Case 2, the optimal contract adjusts continuously and becomes strictly more conservative as the run probabilities increases.
  - The optimal allocation is never a mere randomization over the *unconstrained efficient allocation* and the corresponding run allocation from the *post-deposit* game. Hence this is also a contribution to the sunspots literature: another case in which SSE allocations are not mere randomizations over certainty allocations.

# Summary and Concluding Remark

- In Case 3, the ICC binds for small run-probabilities, which forces the contract to be more conservative than it would have been without the ICC. Hence, for Case 3, the optimal contract does not change with $s$ until the ICC no longer binds.
  - For small $s$, the optimal allocation is a randomization over the *constrained efficient allocation* and the corresponding run allocation from the *post-deposit* game.