Problem Set 5

Instructor: Dean Corbae

Group Members: Shangjie Shi, Dondrup Drakpa, Junkai Sun,

Zachary Orlando, Cutberto Frias Sarraf

May 7, 2025

This problem is designed to have you compute the dynamic contracting problem in *DeMarzo and Fishman* (2007, *RFS*). The environment and associated programming problem are given in the handout on DeMarzo and Fishman on my website. As in that handout, the programming problem solves choose

$$\big\{d(a) \geq 0,\ p_i(a) \in [0,1],\ a_i'(a) \in [\underline{a}, \overline{a}] \mid i \in \{H, L\}\big\}$$

to solve:

$$\begin{aligned} b(a) = \max \Big\{ & \pi\,[\tilde{y}_H(a) - d_H(a) + p_H(a)\,L] + (1-\pi)\,[\tilde{y}_L(a) - d_L(a) + p_L(a)\,L] \\ & + \beta\big[\,\pi\big(1 - p_H(a)\big)\,b\big(a_H'(a)\big) + (1-\pi)\big(1 - p_L(a)\big)\,b\big(a_L'(a)\big)\big]\ \Big\}, \end{aligned} \tag{1}$$

subject to

$$\begin{aligned} a = {} & \pi\,[\,\lambda\big(Y^H - \tilde{y}_H(a)\big) + d_H(a) + p_H(a)\,R] \\ & + (1-\pi)\,[\,\lambda\big(Y^L - \tilde{y}_L(a)\big) + d_L(a) + p_L(a)\,R] \\ & + \delta\big[\,\pi\big(1 - p_H(a)\big)\,a_H'(a) + (1-\pi)\big(1 - p_L(a)\big)\,a_L'(a)\big], \end{aligned} \tag{2}$$

$$\begin{aligned} & \lambda\big(Y^i - \tilde{y}_i(a)\big) + d_i(a) + \big(1 - p_i(a)\big)\,\delta\,a_i'(a) + p_i(a)\,R \\ & \qquad \geq \lambda\big(Y^j - \tilde{y}_j(a)\big) + d_j(a) + \big(1 - p_j(a)\big)\,\delta\,a_j'(a) + p_j(a)\,R, \end{aligned} \tag{3}$$

$$\forall\, i, j \in \{H, L\},\ i \neq j,$$

$$p_i(a)\,R + \big(1 - p_i(a)\big)\,\delta\,a_i'(a) \geq R, \quad \forall\, i \in \{H, L\}, \tag{4}$$

$$Y^i \geq \tilde{y}_i(a), \quad \forall\, i \in \{H, L\}, \tag{5}$$

$$a \in [\underline{a}, \overline{a}]. \tag{6}$$

where

$$\overline{a} = \frac{\pi\, Y^H + (1-\pi)\, Y^L}{1-\beta} \quad \text{(the agent gets everything each period at the principal's rate)},$$

$$q = \max\bigl\{\pi\lambda\bigl(Y^H - Y^L\bigr) + R,\ \delta^{-1}R\bigr\} \quad \text{(the agent can always quit)}.$$

Algorithm

The algorithm you will use is:

1. Initialize the value function $b^i(a)$, with $i = 0$, to the first-best value.
2. Given $b^i(a)$, for each $a \in [\underline{a}, \bar{a}]$, solve (1)–(6) with MATLAB's `fmincon` to obtain decision rules that generate $b^{i+1}(a)$. The routine `fmincon` minimizes an objective subject to a constraint set. See the example program `fmincon_example.m` on my website, which minimizes `objf.m` subject to `const.m`.
3. If $\sup_a \bigl| \, b^{i+1}(a) - b^i(a) \, \bigr|$ exceeds your tolerance level, return to step 2; otherwise stop.

Parameters. All parameter values are taken from DeMarzo and Fishman:

$$\beta = \exp(-0.0953), \qquad \delta = \exp(-0.0998), \qquad L = 75, \qquad R = 0, \qquad \lambda = 1.$$

The cash-flow process is

$$Y^L = 0 \quad \text{with probability } 1 - \pi = \tfrac{1}{2}, \qquad Y^H = 20 \quad \text{with probability } \pi = \tfrac{1}{2}.$$

To do

1. Prove that, in this particular problem, all of the constraints are binding.
2. After solving the problem, graph the value function $b(a)$ and the decision rules $d_i(a)$, $p_i(a)$, and $a'_i(a)$. Interpret your results.

Answer to Question 1

(Following the handout) Assume the resource feasibility constraint is not binding and consider an alternative,

$$\hat{y}_i^*(a) = \hat{y}_i(a) + \varepsilon, \qquad \hat{d}_i^*(a) = d_i(a) + \lambda\varepsilon, \qquad \varepsilon > 0.$$

Then

$$-\lambda\hat{y}_i^*(a) + \hat{d}_i^*(a) = -\lambda\big(\hat{y}_i(a) + \varepsilon\big) + d_i(a) + \lambda\varepsilon = -\lambda\hat{y}_i(a) + d_i(a).$$

Notice that the IC constraint (3) is identical under both allocations and the promise keeping constraint is also unaffected. However, the objective is weakly higher for any $\lambda <= 1$. Hence, assuming the resource constraint binds will achieve the optimal contract (the agent reports truthfully).

The no quitting constraint will not bind for this problem as our policy functions for the second question exceed $\delta^{-1}R$ (which is equivalent to the RHS of the no quitting constraint after simplification).

The IC constraint clearly binds for the high type, as the first best allocation is not the solution to the constrained problem (shown in our solution to Q2). The low-type does not have an incentive to pretend to be another type, so their constraint will not bind.

Answer to Question 2

The policy function for promised utility is kinked because after that level of a, the marginal cost of promised utility tomorrow exceeds the cost of the static incentive (which is always 1). This is due to the concavity of the value function being most pronounced for small values of a.

We can see current incentives begin to rise after the kink in that type/message's promised utility policy function.

The investor never liquidates a high-type, and uses the threat of liquidation at low levels of promised utility to ensure compliance (the "stick").

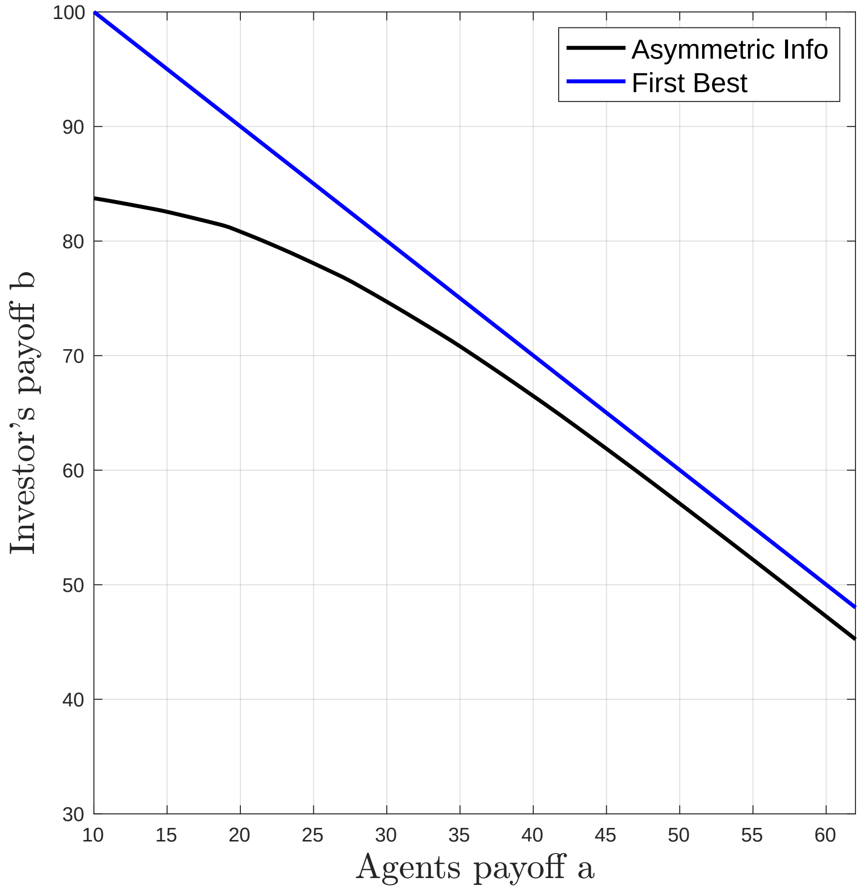

Figure 1: Value function for investors

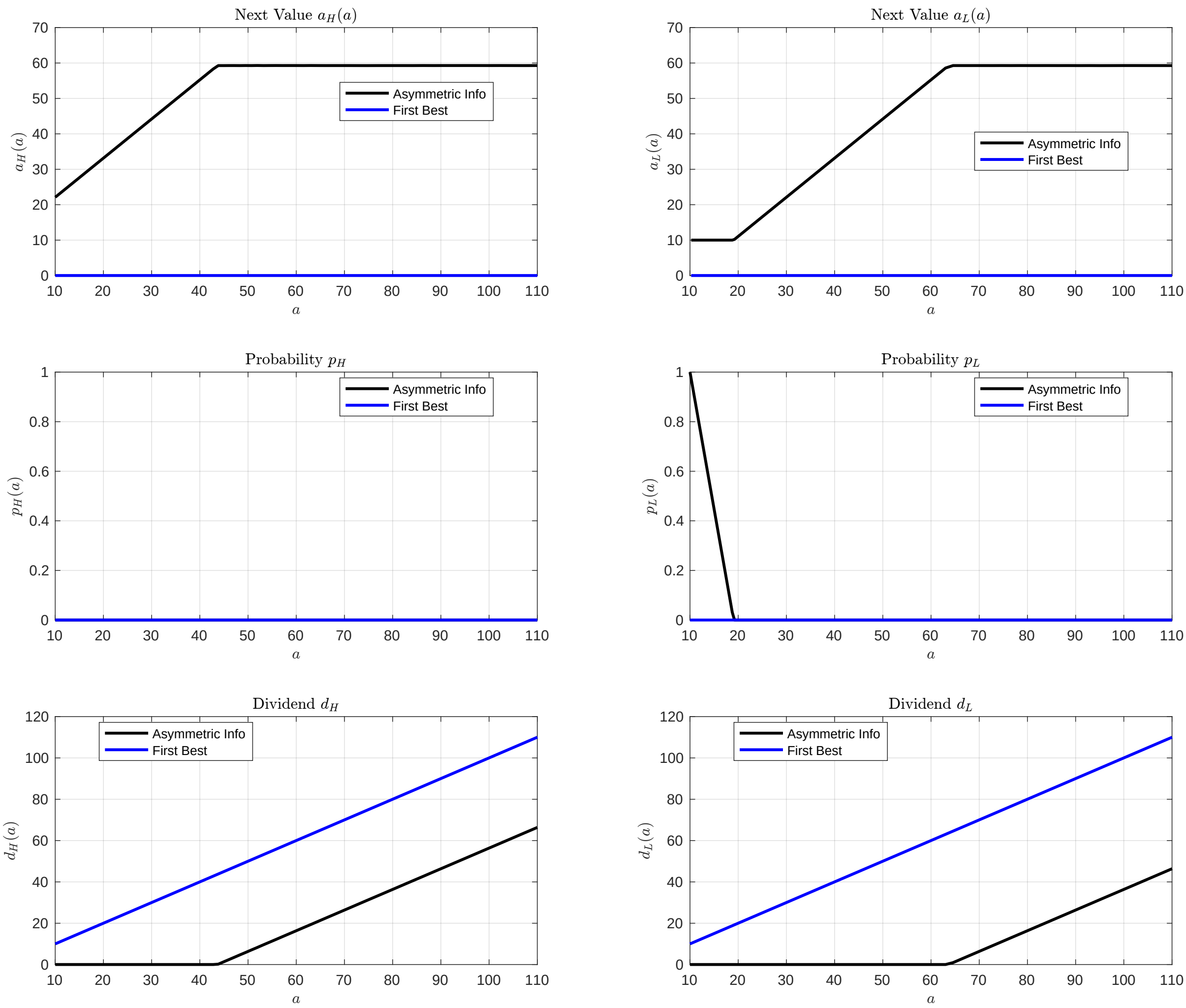

Figure 2: Policy functions