University of Zurich[UZH]

Agent-based Financial Economics
Lesson 2: The Farmer

Luzius Meisser, Prof. Thorsten Hens

luzius@meissereconomics.com

"What I cannot create, I do not understand."

- Richard Feynman

Today

- Discussion of exercise 1, the Hermit
- Special solution method: golden ratio search
- Classic theory: The Walrasian market
- Our orderbook based market
- Exponential search for prices
- The optimal number of firms
- Preparation of exercise 2: the Farmer

Quelle: Deutsche Fotothek

Moving from hermits to collaborative, free market farming.

Final Hermit Ranking

Rank	Consumer	Utility	Version	Method
1	team102-Hermit	4.487127668778711	Andrea on 2018-09-26T21:03:59Z	workFraction = (51.0/4.0)/24
2	team103-Hermit	4.487127668778711	DESKTOP-EQGLALN\\tobia on 2018-09-26T15:47:58Z	workFraction = 12.75/24
3	team105-Hermit	4.487127668778711	Tbrlan on 2018-09-26T15:03:37Z	workFraction = 0.53125
4	team101-Hermit	4.487127668778711	JustierNo1 on 2018-09-24T18:06:44Z	plannedLeisureTime = 11.25
5	team104-Hermit	4.48712766877717	NathalieTorrent on 2018-09-25T09:38:20Z	Hill climb & “binary search” (Note that this might not work)
6	team100-Hermit	4.487127668776791	Sommer1872 on 2018-09-27T12:14:30Z	Gradient ascent hours = hours + 0.1*(20.4-1.6*hours);

Well done! → Every team gets 10 out of 10 points for this exercise.

Analytical solution (team101)

$$max\, U(h_{work}) = log(24 - h_{work}) + log((h_{work} - 6)^{0.6} x_{land}^{0.2})$$

$$\frac{\delta U}{\delta h} = \frac{0.6}{h_{work} - 6} - \frac{1}{24 - h_{work}} \stackrel{!}{=} 0\; F.O.C$$

$$\frac{0.6}{h_{work} - 6} = \frac{1}{24 - h_{work}}$$

$$0.6(24 - h_{work}) = h_{work} - 6$$

$$14.4 - 0.6h_{work} = h_{work} - 6$$

$$20.4 = 1.6h_{work}$$

$$12.75 = h_{work}$$

Analytical Solution

$$max\ U(h_{work}) = log(24 - h_{work}) + log((h_{work} - 6)^{0.6} x_{land}^{0.2})$$

labor share

land is constant

$$U(h) = \log(24-h) + \log((h-h_c)^{\alpha_h} \cdot K) = \log(24-h) + \alpha_h \log(h-h_c) + K'$$

fixed cost

$$\frac{dU}{dh} = \frac{-1}{24-h} + \alpha_h \frac{1}{h-h_c} \overset{!}{=} 0$$

$$\Rightarrow \alpha_h(24-h) = h - h_c$$

$$\Rightarrow 24\alpha_h + h_c = h + \alpha_h h$$

$$\Rightarrow h = \frac{24\alpha_h + h_c}{1+\alpha_h}$$

```
protected double calculateWorkAmount(IStock currentManhours) {
    double weight = prodFun.getWeight(currentManhours.getGood()).weight;
    double fixedCost = prodFun.getFixedCost(currentManhours.getGood());
    return (currentManhours.getAmount() * weight + fixedCost) / (1 + weight);
}
```

Discussed implementations are now in: exercises/com/agentecon/exercise1 and named AnalyticHermit.java, AdaptiveHermit.java

Golden Ratio Search

Comparison Method for One-dimensional Optimization

Slides copied from MFOEC167 “Computational Economics and Finance” by Prof. Karl Schmedders.

Bracketing search method: **Golden ratio search**

Find local minimum of function f on interval $[a, b]$

Select two interior points c, d, such that $a < c < d < b$

Case 1: $f(c) \leq f(d)$ minimum must lie in $[a, d]$
replace b with d, new interval $[a, d]$

Case 2: $f(c) > f(d)$ minimum must lie in $[c, b]$
replace a with c, new interval $[c, b]$

Now repeat the iteration on the new interval

Question: How to choose c and d?

Golden Ratio Search

Choosing Points

Select c and d such that the intervals $[a, c]$ and $[d, b]$ have the same length, so $c - a = b - d$

$$\begin{aligned} c &= a + (1-r)(b-a) &= ra + (1-r)b \\ d &= b - (1-r)(b-a) &= (1-r)a + rb \end{aligned}$$

where $\frac{1}{2} < r < 1$ to ensure $c < d$

One of the old interior points will be an endpoint of the new interval; for efficiency, use the other interior point also as an interior point for the new subinterval; so in each iteration only one new interior point and only one new function evaluation is needed

If $f(c) \leq f(d)$ then these conditions require

$$\frac{d-a}{b-a} = \frac{c-a}{d-a}$$

Golden Ratio

$$\begin{aligned} & & \frac{d-a}{b-a} &= \frac{c-a}{d-a} \\ &\Longleftrightarrow & \frac{r(b-a)}{b-a} &= \frac{(1-r)(b-a)}{r(b-a)} \\ &\Longleftrightarrow & r &= \frac{(1-r)}{r} \\ &\Longleftrightarrow & r^2 &= 1-r \\ &\Longleftrightarrow & r^2 + r - 1 &= 0 \\ &\Longleftrightarrow & r &= \frac{-1 \pm \sqrt{5}}{2} \end{aligned}$$

and thus r is the **golden ratio**,

$$r = \frac{-1+\sqrt{5}}{2} \approx 0.618034$$

Interior points

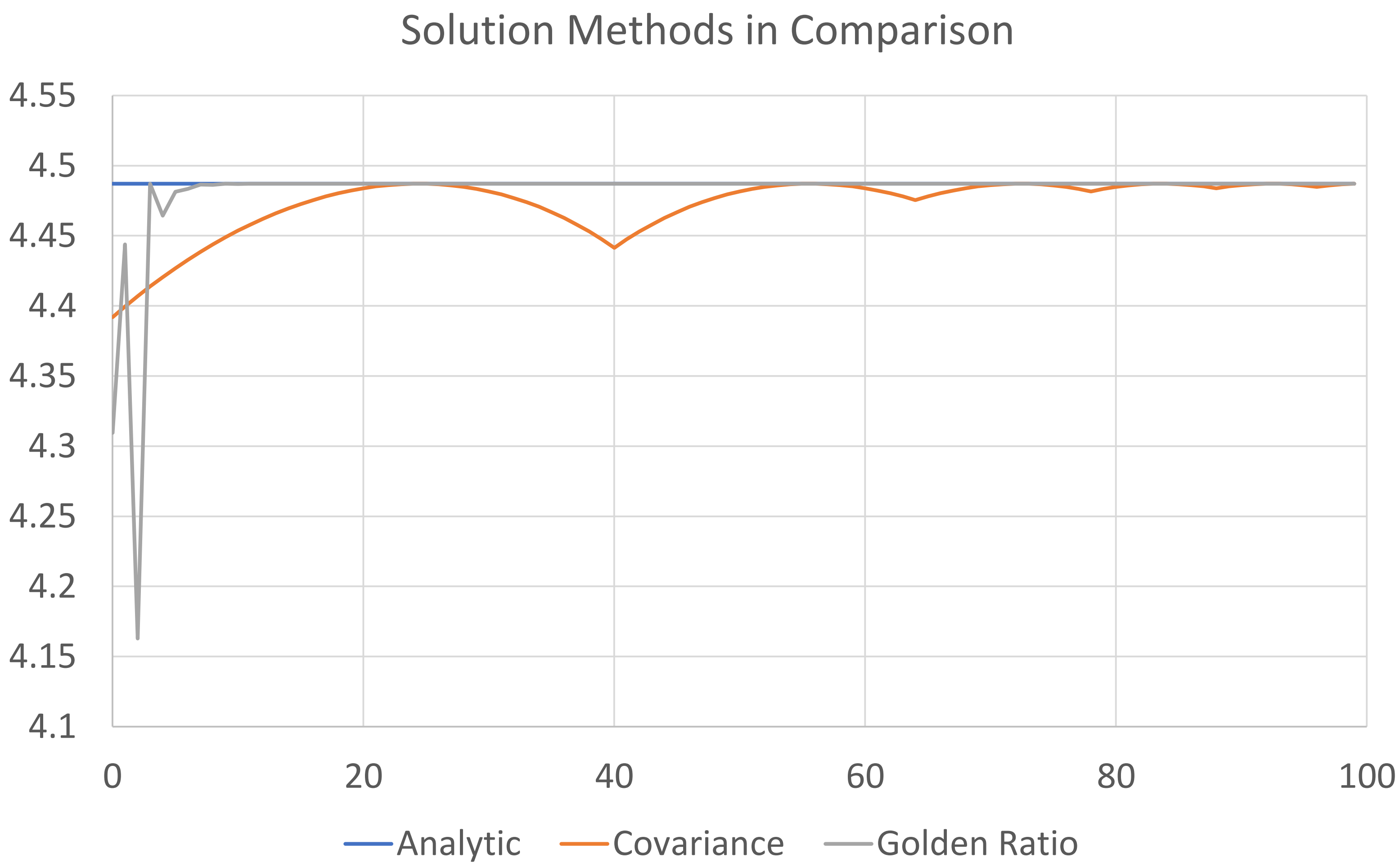
$$\begin{aligned} c &= a + (1-r)(b-a) \\ d &= a + r(b-a) \end{aligned}$$

Covariance Search

1. Start at some work amount
2. Observe resulting utility
3. Update covariance between work and utility (using a moving covariance, just like a moving average, that tends to “forget” old values over time)
4. Either work 1% more or less, depending on sign of covariance
5. Loop to step 2

```java
public class CovarianceControl implements IControl {

    private MovingCovarianceAlt cov;
    private IBelief belief;

    public CovarianceControl(double start, double memory) {
        this.belief = new ConstantFactorBelief(start, 0.01);
        this.cov = new MovingCovarianceAlt(memory);
    }

    public double getCurrentInput() {
        return belief.getValue();
    }

    public void reportOutput(double output) {
        this.cov.add(getCurrentInput(), output);
        boolean upwards = this.cov.getCovariance() > 0;
        this.belief.adapt(upwards);
    }

}
```

Result



Reproduce by running com.agentecon.exercise1.HermitComparison

The analytic method only works for cobb douglas production functions. The golden ratio method works for all convex functions. The covariance method even works when the function parameters change over time.

- → Trade-off between efficiency and adaptivity.
- → Exploration vs. exploitation!

Walrasian Market

- Consumers with endowment and utility function
- Firms with production function, maximizing profits and handing them back to the consumers
- An arbitrary number of goods

Tatonnement process:

1. Walrasian auctioneer proposes price vector p
2. Every agent i tells auctioneer how much he would buy or sell: x_i
3. If everything adds up, i.e. $\sum x_i = 0$, markets clear and we have found the equilibrium.
4. If things do not add up, the Walrasian auctioneer tries a different p.

An equilibrium must exist under fairly general conditions (Arrow, Debreu. 1954). However, finding the equilibrium is in PPAD complexity class (very similar to NP, https://people.cs.pitt.edu/~kirk/CS1699Fall2014/lect4.pdf).

Léon Walras

French economist

Marie-Esprit-Léon Walras was a French mathematical economist and Georgist. He formulated the marginal theory of value and pioneered the development of general equilibrium theory. Wikipedia

Born: December 16, 1834, Évreux, France

Died: January 5, 1910, Montreux

Parents: Auguste Walras, Louise Aline de Sainte-Beuve

Education: HEC Lausanne, Mines ParisTech

Influenced: Alfred Marshall, Vilfredo Pareto, Joseph Schumpeter, MORE

Market Equilibrium

Corresponding market equilibrium from the book «Economic Foundations of Finance» by Sabine Elmiger and Thorsten Hens.

In contrast to our model, consumers do not consume any man-hours themselves.

L: labor
w: price of labor (wages)
p: prices of goods
Y: production vector
C: consumption vector

A *market equilibrium* is an allocation of supply and demand $(L^{s*}, Y^*, L^{d*}, C^*)$ as well as a price-wage system (w^*, p^*), so that

the **firm maximizes profits**

$$(L^{d*}, Y^*) \in \underset{L^d, Y}{\operatorname{argmax}} \ \pi = p^* Y - w^* L^d$$
$$\text{s. t.} \quad Y = F(L^d),$$

the **household maximizes utility**

$$(L^{s*}, C^*) \in \underset{L^s, C}{\operatorname{argmax}} \ U(C)$$
$$\text{s. t.} \quad p^* C = w^* L^s + \pi^*, \ L^s \leq \bar{L},$$

and **markets clear**

$$L^{d*} = L^{s*} \text{ and } C^* = Y^*.$$

Note that the point, x, where $f(x)$ takes the maximum in a maximization problem $\max_x f(x)$ is called the $\operatorname{argmax}_x f(x)$.

Box 2.1: **Market equilibrium.** Definition of equilibrium in the basic economic model.

Market Equilibrium: Price Taking

The most important implicit assumption in the classic market setting is that of price-taking firms.

Does that make sense? Somewhat. ☺

The argument goes like this: if there is an infinite number of firms, none of can have an impact on prices, and thus they will exhibit pure price-taking behavior.

Obviously, there are not infinitely many firms in reality, but maybe it is a good approximation as most firms are small in comparison to the whole economy?

No, because the distribution of firm sizes follows a power-law that does not even have a well-defined average! Even if there are infinitely many firms, there will always be firms that can measurably impact prices.

Example: Samsung’s revenue is 17% of Koreas’ GDP and 20% of its exports.

Market Equilibrium Example

Take the Hermit economy, but separate production into a firm, so we have exactly one firm and one consumer.

Consumer maximizes utility given prices and endowment:

Consumer maximizes utility:

$$u(h_l, x) = \ln h_l + \ln x$$

└ potatoes

└ hours spent as leisure time

Subject to the budget constraint:

setting potatoe price to $p_x = 1$ without loss of generality

$$\underbrace{p_h \cdot (24 - h_l)}_{\text{wage received}} + \pi = \underbrace{p_x \cdot x}_{\text{amount spent on potatoes}} = x$$

└ dividends received from firm

Consumer maximizes utility:

$$u(h_l, x) = \ln h_l + \ln x$$

h_l: hours spent as leisure time

x: potatoes

Subject to the budget constraint:

$$\underbrace{p_h \cdot (24 - h_l)}_{\text{wage received}} + \Pi = \underbrace{p_x \cdot x}_{\text{amount spent on potatoes}} = x$$

Π: dividends received from firm

$= x$: setting potatoe price to $p_x = 1$ without loss of generality

$$u(x) = \ln h_l + \ln\left(p_h (24 - h_l) + \Pi\right)$$

$$\frac{du}{dh_l} = \frac{1}{h_l} + \frac{1}{p_h(24 - h_l) + \Pi} \cdot (-p_h) \overset{!}{=} 0$$

$$\Rightarrow \quad p_h(24 - h_l) + \Pi = p_h \cdot h_l$$

$$\Rightarrow \quad 24 + \frac{\Pi}{p_h} = 2h_l$$

$$\Rightarrow p_h h_l = \frac{24 p_h + \Pi}{2}$$

$\Rightarrow$ spend half of total income on leisure

In general: $U(x) = \sum_i \log x_i^{\alpha_i} = \sum_i \alpha_i \log x_i$ implies spendings proportional to α_i on each good i

Market Equilibrium Example

Firm maximizes profits given prices:

Firm maximizes profits Π:

$$\max_{h_w} \Pi(h_w) = \max_{h_w} \underbrace{p_x f(h_w)}_{\text{revenue}} - \underbrace{p_h \cdot \overset{\text{acquired man-hours}}{h_w}}_{\text{cost}}$$

$$\frac{\partial \Pi}{\partial h_w} = f'(h_w) - p_h \overset{!}{=} 0$$

$$\Rightarrow f'(h_w) = p_h$$

marginal production = marginal cost

Combining these equation with those of the consumer, we get the exact same result as in exercise 1.

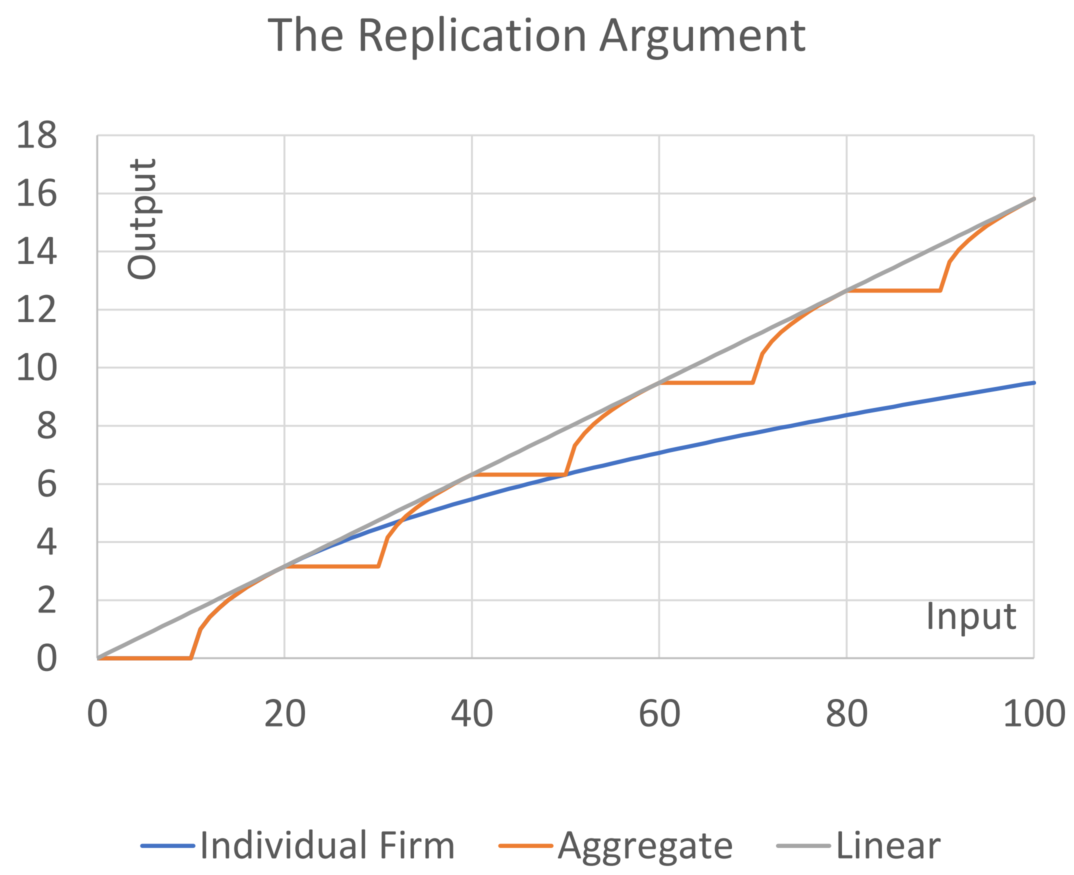
What if there are multiple firms?

Mathematical solution under homogeneity: determine optimal number of firms, divide inputs among them, multiply output again. Do the same for the consumers and their consumption.

There is an optimal number of firms, and the Cobb-Douglas production function turns into a linear production function in the aggregate! Economists call this the “replication argument”. Amazing!

Aggregation can change the quality of something. All firms in aggregate behave differently than the firms individually.

→ Emergence



Optimal number of firms

k: number of firms

$f(x)$: production function

$$\max_k \; \underbrace{k f\left(\frac{x}{k}\right)}_{\text{divide input among } k \text{ firms}} = \overbrace{F(x)}^{\text{aggregate production function}}$$

$$\frac{dF(x)}{dk} = f\left(\frac{x}{k}\right) + k \cdot f'\left(\frac{x}{k}\right) \cdot \left(-\frac{x}{k^2}\right) \overset{!}{=} 0$$

$$f\left(\frac{x}{k}\right) = \frac{x}{k} f'\left(\frac{x}{k}\right)$$

For $f(x) = (x-c)^\alpha$ we get:

$$\left(\frac{x}{k}-c\right)^\alpha = \frac{x}{k}\alpha\left(\frac{x}{k}-c\right)^{\alpha-1}$$

$$\left(\frac{x}{k}-c\right) = \frac{x}{k}\alpha$$

$$x - ck = x\alpha$$

$$\Rightarrow \quad k^* = (x - x\alpha)/c = x\frac{(1-\alpha)}{c}$$

→ In theory, seven hermits could team up and create six farms to produce more than before with the same effort.

$$(12.75-6)^{0.6}100^{0.2} = 7.899$$

$$\frac{6}{7}\left(\frac{7}{6}12.75-6\right)^{0.6}100^{0.2} = 7.97911$$

→ In the next exercise, we will find out whether they will succeed at that.

$$\Rightarrow \text{optimal } F^*(x) = k^* f\left(\frac{x}{k^*}\right) = x\frac{(1-\alpha)}{c} f\left(\frac{x \, c}{x(1-\alpha)}\right) = x\frac{(1-\alpha)}{c} f\left(\frac{c}{1-\alpha}\right)$$

$$= x\frac{(1-\alpha)}{c}\left(\frac{c}{1-\alpha}-c\right)^\alpha$$

$$= x \cdot \underbrace{c^{\alpha-1}\alpha^\alpha(1-\alpha)^{1-\alpha}}_{\text{constant}}$$

Break – 5 minutes

- Come to me if you team is broken!

- Up next: extending our model from hermits to collaborative farming.

Simulation Market Structure

We do not want any centralized decision taking.
→ No central planner
→ No Walrasian auctioneer

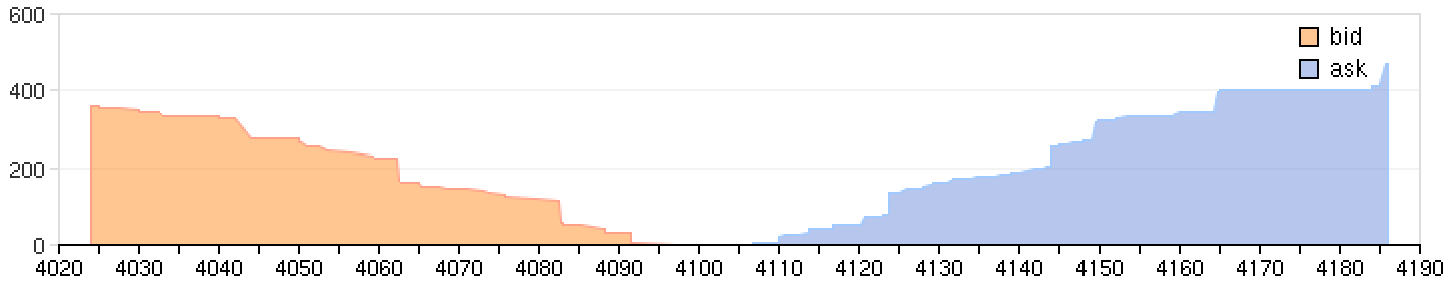
In the spirit of Hayek (1945): "In a system in which the knowledge of the relevant facts is dispersed among many people, prices can act to coordinate the separate actions of different people.“
[F. A. Hayek. The use of knowledge in society. The American Economic Review, 35(4), 1945.]

Order book

Firms as price makers (IGoodsTrader)
Consumers as price takers (IMarketParticipant)

Order books

- No centralized price setting mechanism!
- Law of one price only holds approximatively if everyone is rational.
- Limit order: “I sell 3 potatoes for 5 CHF.” Stays in the order book until matching trade found or cancelled. Price is certain, but not when or if you get it.
- Market order: “I want to spend 10 CHF on potatoes now.” Price is not certain, but you get it now.
- In our simulation, all firms are asked to place limit orders, and then the consumers enter the market doing market orders.



Typical market depth chart from a real market.
The price movement caused by a market order is roughly proportional to its size.

Simulation Day

1. Consumers are endowed with 24 man-hours
2. Firms pay dividends to their owners.
3. Firms place limit orders, hoping to sell newly produced potatoes and to acquire man-hours for the next production.
4. Consumers enter the market in random order and optimize their utility given the prices they encounter. Unless some money is explicitly put aside, they will spend everything.
   (The function **tradeGoods(IPriceTakerMarket market)** does that for you.)
5. Firms adjust their price believes depending on how successful their orders were. The class MarketingDepartment that does this for you.
6. Consumers consume their inventory, enjoying utility.
7. Firms use the acquired input to produce over night.

Simulation Market Structure

How do firms set prices?
Ask the marketing department. ☺

In the exercise, you get a ready-made marketing department, so you can focus on the other non-trivial problems:

- How much money do you want to spend on acquiring inputs?
- What dividend do you want to pay to the founder (your farmer)?
- When should you give up and declare bankruptcy?

```
public class Farm extends Producer {

    private MarketingDepartment marketing;

    public Farm(IAgentIdGenerator id, IShareholder own

    @Override
    public void offer(IPriceMakerMarket market) {
        double budget = calculateBudget();
        marketing.createOffers(market, this, budget);
    }

    private double calculateBudget() {

    @Override
    public void adaptPrices() {
        marketing.adaptPrices();
    }
```

Simulation Market Structure

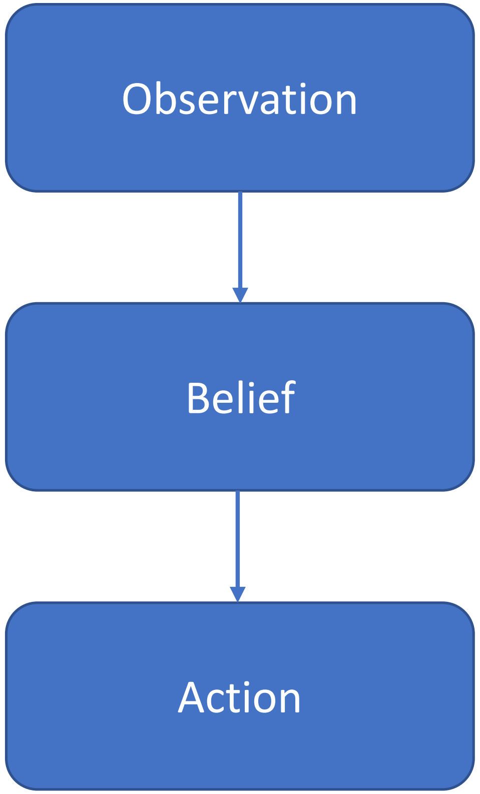
How do firms set prices?
Ask the marketing department. ☺

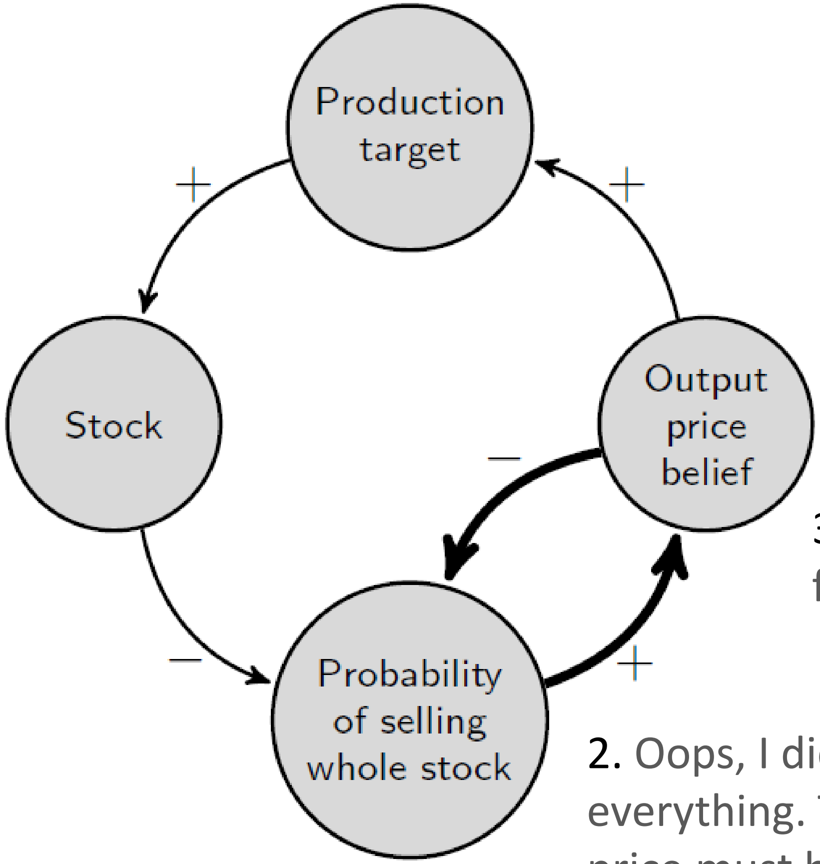
In the exercise, you get a ready-made marketing department, so you can focus on the other non-trivial problems:

- How much money do you want to spend on acquiring inputs?
- What dividend do you want to pay to the founder (your farmer)?
- When should you give up and declare bankruptcy?

```java
public class Farm extends Producer {

    private MarketingDepartment marketing;

    public Farm(IAgentIdGenerator id, IShareholder own

    @Override
    public void offer(IPriceMakerMarket market) {
        double budget = calculateBudget();
        marketing.createOffers(market, this, budget);
    }

    private double calculateBudget() {

    @Override
    public void adaptPrices() {
        marketing.adaptPrices();
    }

```

Price Adaption Mechanism



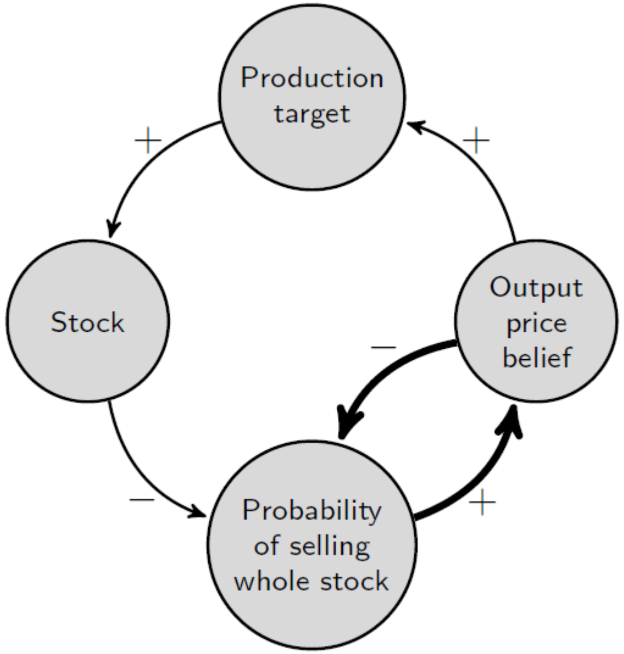


1. I believe the right price for a potatoe is 10$.

3. I believe the right price for a potatoe is 9.95$.

2. Oops, I did not sell everything. The right price must be lower.

System Dynamics Crash Course



+ indicates a positive effect. A higher x leads to a higher y.

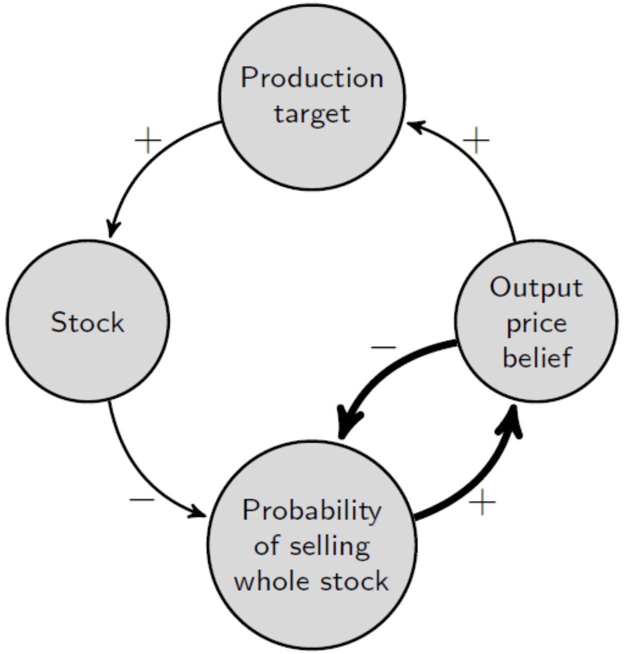
- indicates a negative effect. A higher x leads to a lower y.

Loops with an even number of - signs are positive or self-reinforcing feedback loops.

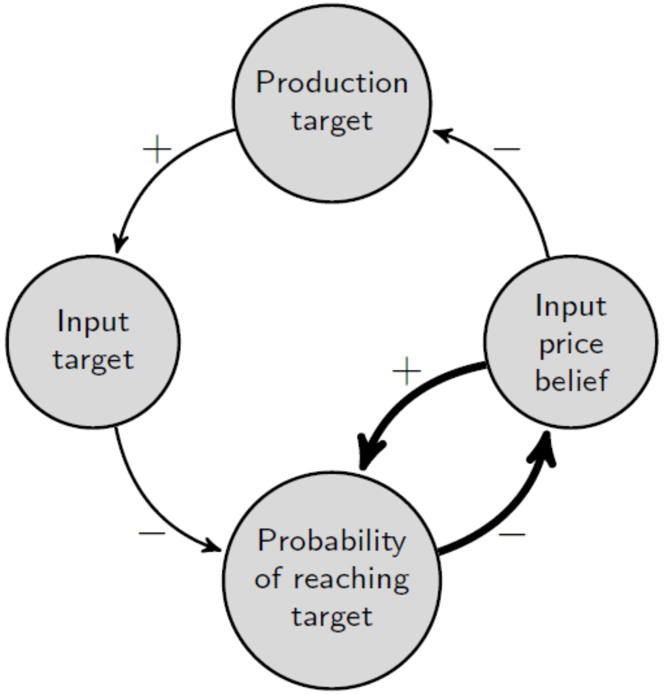
Loops with an odd number of - signs are negative or balancing feedback loops.

→ Simple and powerful tool to analyze the dynamics of a system.

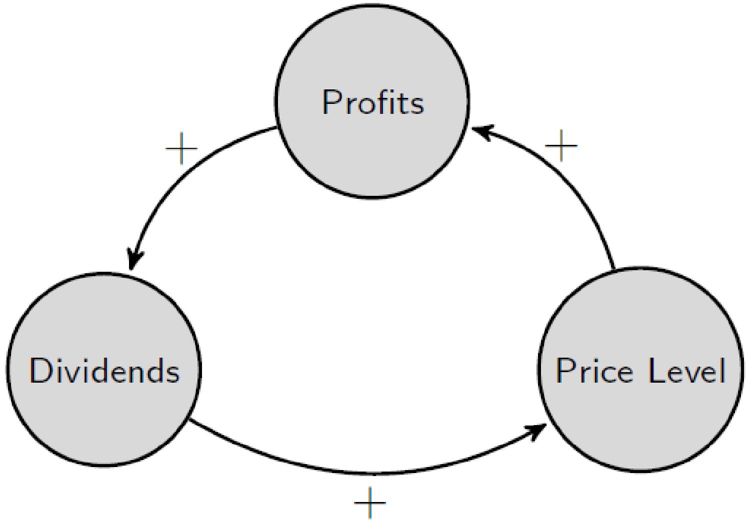
System Dynamics of Prices



Two balancing feedback loops around output prices.

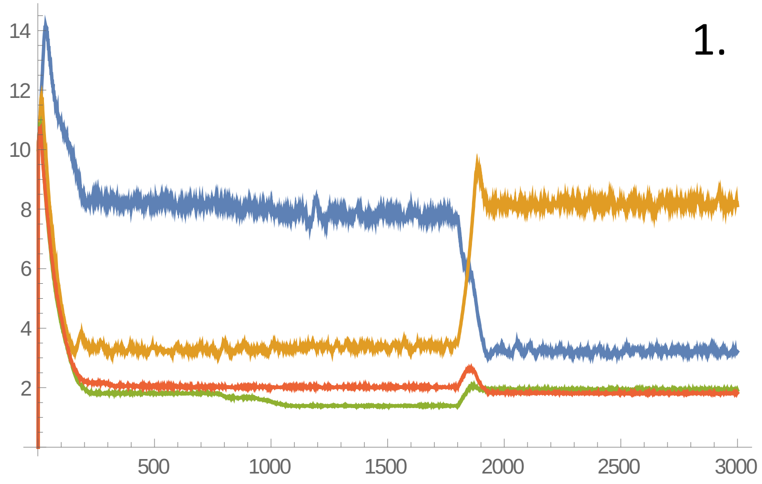


Two balancing feedback loops around input prices.

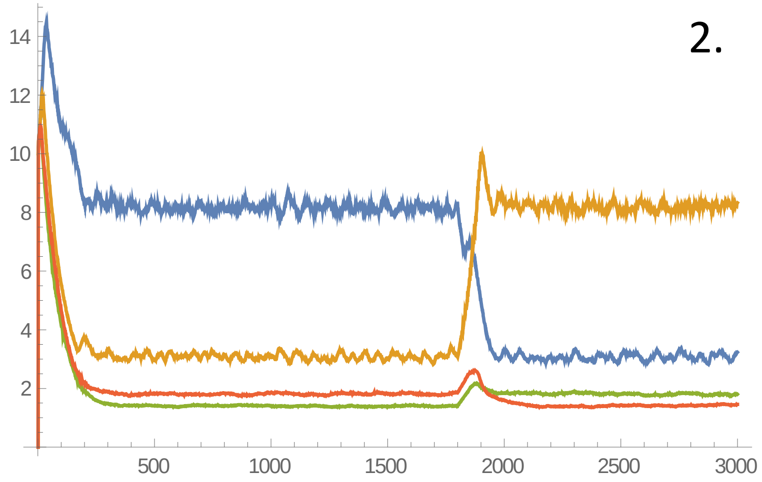


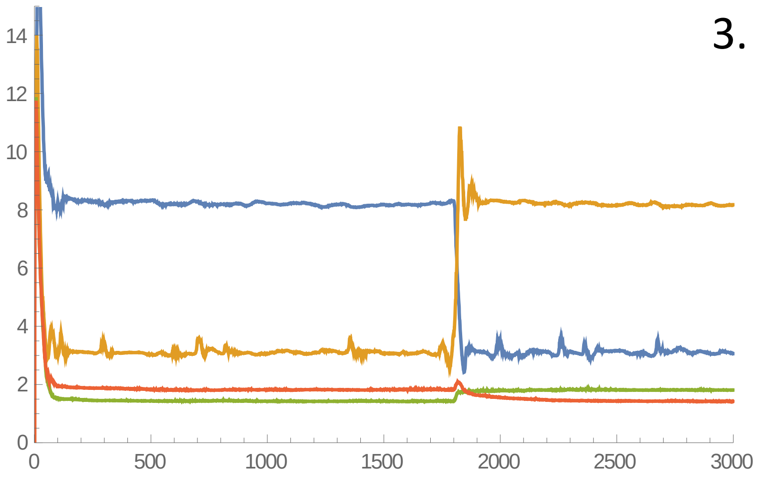
A self-reinforcing feedback loop that might cause hyper-inflation!

Price Adaption Comparison



1. Constant factor
2. Constant, randomized
3. Exponential Price Search





Exponential Price Search

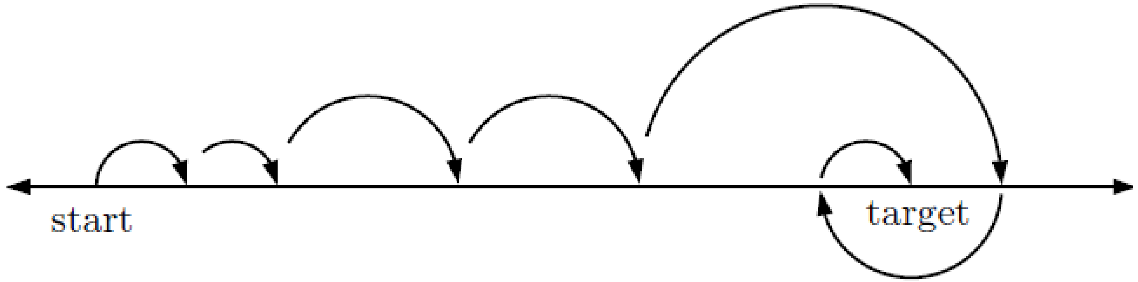


Fig. 4 Adapting price beliefs with exponential search: increasing the adjustment factor after every second step in same direction, decreasing on turns

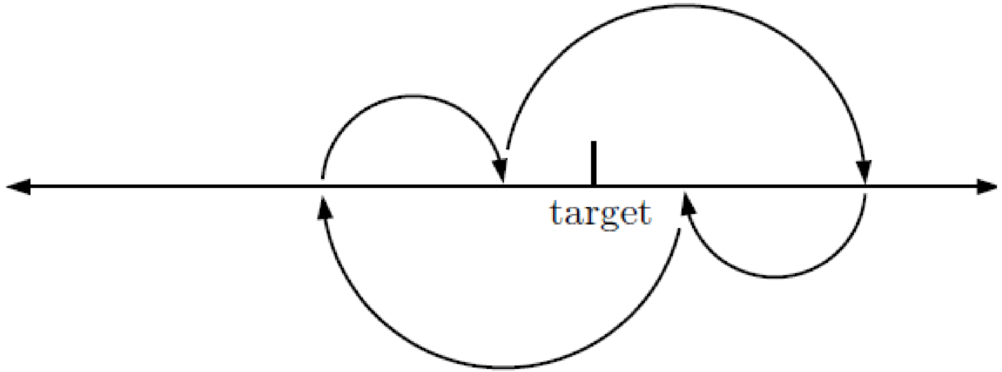


Fig. 5 Trap: no convergence when increasing the adjustment factor too early

Sensor Prices: Problem

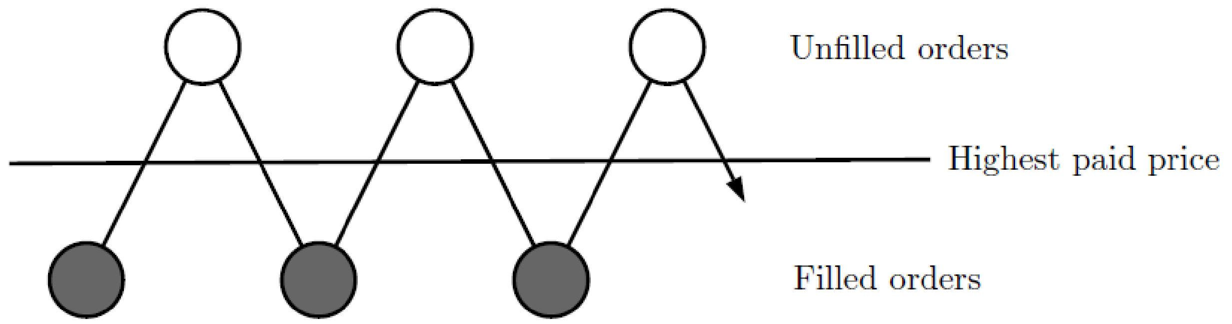


Fig. 6 Typical price adaption heuristics lead to filled orders only half of the time, alternating between a price below and above what the market can bear.

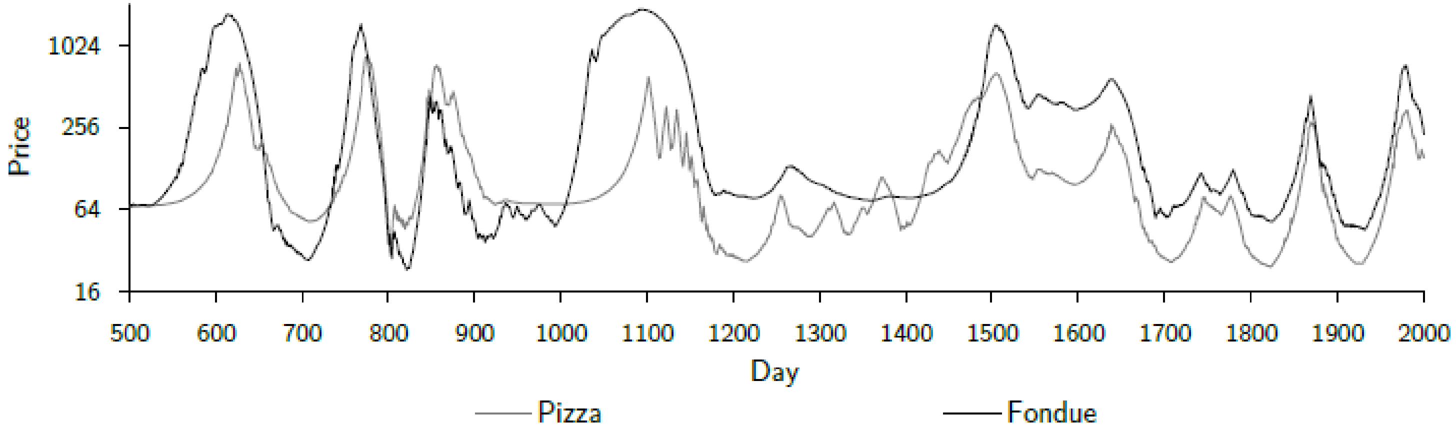


Fig. 10 Disabling sensor pricing in the default configuration can cause chaos.

Sensor Prices: Solution

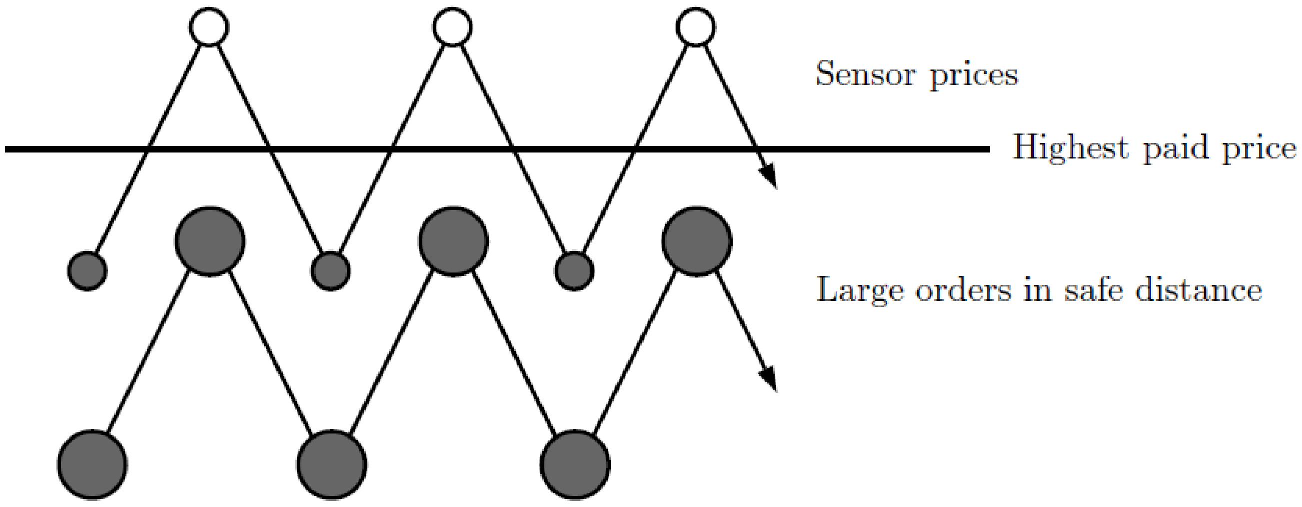


Fig. 7 With sensor prices, only a small fraction of volume is sacrificied for price exploration, whereas the bulk can reliably drive revenue.

From: Meisser, L. & Kreuser, “An Agent-Based Simulation of the Stolper-Samuelson Effect”, Computational Econonomics (2016). https://doi.org/10.1007/s10614-016-9616-x

Recurring them: exploration vs exploitation.

Demo

- How to run the whole simulation on your computer (excluding the agents of the other teams)
- Only way to compete against the other teams is to upload your agents
- Everything else – including the web server - can be run locally