

By Paul Belleflamme, 6 March 2014

## How do comparison shopping sites make a living?

**Industrial Organization Markets And Strategies (Paperback)**  
by BELLEFLAMME  
Rating: ★★★★★ (0)  
Starts at: ₹396 at **rediff BOOKS**

**About the book:**  
Contributors: BELLEFLAMME  
Language: Hindi  
Binding: Paperback

PRICE	BOOK DETAILS	REVIEWS				
Industrial Organization Markets And Strategies (Paperback) Price Filter by: <input type="checkbox"/> COD <input type="checkbox"/> EMI Price + Shipping						
Store	Price	EMI	COD	Shipping Cost	Shipping Time	Seller Review
ibeam.com	₹404		Yes	FREE	2-3 days	★★★★★
rediff BOOKS	₹396			₹30	4 days	★★★★★
snapdeal	₹436		Yes	FREE	4 days	★★★★★

Comparison shopping sites, also known as shopping robots or shopbots, have been around for about two decades. Sites such as [Shopping.com](#), [Shopper.com](#), [PriceGrabber](#), [Shopzilla](#), [Vergelijk](#) or [Kelkoo](#) help us find goods or services that are sold online by providing us with loads of information (products sold, price charged, quality, delivery and payment methods, etc.). As they present the information in an accessible way and display links to the vendors' websites, shopbots significantly reduce our costs of searching for the best deal.

The most common business model for shopbots is to charge sellers for displaying their information while letting users access the site for free. Fees can be computed in different ways, as explained by [Moraga and Wildenbeest](#) (2011, p. 4):

*Initially, most comparison sites charged firms a flat fee for the right to be listed. More recently, this fee usually takes the form of a cost-per-click and is paid every time a consumer is referred to the seller's website from the comparison site. Most traditional shopbots, like for instance PriceGrabber.com, and Shopping.com operate in this way. Fees typically depend on product category—current rates at PriceGrabber range from \$0.25 per click for clothing to \$1.05 per click for plasma televisions. Alternatively, the fee can be based on the execution of a transaction. This is the case of Pricefight.com, which operates according to a cost-per-acquisition model. This model implies that sellers only pay a fee if a consumer buys the product. Other fees may exist for additional services. For example, sellers are often given the possibility to obtain priority positioning in the list after paying an extra fee.*

How can such a business model be economically viable? To answer this question, we need to understand how shopbots create value for both retailers and consumers, so as to outperform their outside option, i.e., the possibility for them to find each other and conduct transactions outside the platform?

A quick journey through a number of landmark contributions to the theory of imperfect competition will help us understand the importance of price transparency and search costs. To use a simple setting, think of a number of firms offering exactly the same product, which they produce at exactly the same constant unit cost. They face a large number of consumers. If firms compete by setting the price of their product, [Joseph Bertrand](#) has shown in 1883 that the only reasonable prediction of this competition is that firms will set a price equal to the unit cost of production. This result is sometimes called the '*Bertrand Paradox*' as the competitive price is reached although there may be no more than two firms in the industry.

One important assumption behind this result is that there is full transparency of prices: all consumers are able to observe the prices of all firms without incurring any cost. As the firms offer products that are exactly the same, consumers only care about the price and (absent capacity constraint), they all buy from the cheapest seller, which generates this cutthroat competition.

What happens if it is assumed instead that consumers face a positive search cost if they want to observe and compare prices? Peter Diamond (Nobel Prize in economics, 2010) show in his [1971 paper](#) that the exact opposite result obtains: all firms will price at the monopoly level and no consumer will search. This result, known as the '*Diamond Paradox*', holds even if consumers have infinitesimal search costs. [Belleflamme and Peitz](#) (2010, p. 164) explain the intuition:

*In this equilibrium, consumers expect firms to set the monopoly price. A firm which deviates by setting a lower price certainly makes those consumers happier that learnt about it in the first place, but since the other consumers do not learn about it, this will not attract additional consumers. Given their beliefs, consumers have an incentive to abstain from costly search, so that a deviation by a firm is not rewarded by consumers.*

The next question that naturally arises is what happens between the previous two extremes. What if consumers have different search costs. In their "[bargains and ripoffs](#)" paper of 1977, Steven Salop and Joseph Stiglitz (Nobel Prize in economics, 2001) suppose that there are two kinds of consumers: the "informed" consumers can observe all prices for free, whereas the "uninformed" consumers know nothing about the distribution of prices. Under this assumption, they show that *spatial price dispersion* can prevail at equilibrium: some stores sell at the competitive price (and attract informed consumers) while other stores sell at a higher price (selling only to uninformed consumers).

In his [model of sales of 1980](#), Hal Varian (now chief economist at Google) establishes the possibility of *temporal price dispersion*. In his model, the equilibrium conduct for firms is to randomize over prices (to put it roughly, it is as if they were rolling a dice to determine which price to set). As a consequence, at any given period of time, firms set different prices for the same

product; also, any given firm changes its prices from one period to the next. This is why price dispersion is said to be temporal.



Let us now terminate our journey by introducing shopbots into the picture. In the previous models, firms didn't have to incur any cost to convey price information to consumers. It was consumers who had to search for the information, with some consumers incurring a larger search cost than others for some unknown reason. By adding a new player to the model, namely a shopbot, Michael Baye and John Morgan propose, in their [2001 paper](#), a more realistic model where the above assumptions are relaxed. In their model, a profit-maximizing intermediary runs a shopbot that mediates the information acquisition and diffusion process. This intermediary can charge both sides of the market for its services; that is, firms may have to pay the intermediary to advertise their price and consumers may have to pay to gain access to the list of prices posted on the shopbot. After observing the fees set by the intermediary, sellers decide whether or not to post their price on the shopbot and if so, which price, while consumers decide whether or not to visit the shopbot and learn the prices (if any) that are posted there.

Baye and Morgan show that price dispersion persists in this environment. This is because the intermediary optimally chooses to make sellers pay for advertising their price on the shopbot, while letting all consumers access the shopbot for free. This means that all consumers are 'fully informed' in the sense that they buy from the cheapest firm on the shopbot. Despite this fact, firms earn positive profits at equilibrium (this is due to the fact that they post randomized prices on the shopbot, as was the case in Varian's model of sales).

The predictions of this model square quite well with the business model that most shopbots have adopted and with the common observation that prices listed on shopbots are dispersed (even though the advertised products are very similar). So, the quick journey that we have made allows us, I believe, to understand better how shopbots can enter the market and survive in the long run.

[Important remaining questions are whether shopbots increase the competitiveness of product markets and enhance market efficiency. This is what I ask you to investigate.](#)