e-Commerce Bargain-Hunting
with an unbun Model

Raphael Yahalom and Stuart E. Madnick

Sloan WP#4067  CISTR WP #99-02
March 31, 1999

Sloan School of Management
Massachusetts Institute of Technology
Cambridge, MA 02142
Abstract

First-generation comparison-shopping software tools in general have limitations in dealing with bundled products. A new model for e-commerce comparison-shopping, based on a logical unBundling of sellers' offerings, is introduced. The model captures dominance relationship between offerings and consumer requirements and may serve as a foundation for a variety of shopping decision support software tools. The model provides a basis for defining notions of bargains and designing algorithms for searching for them. Various implications and remaining challenges are outlined.

1 Introduction

Bargains-hunting pursuits are as old as markets themselves. Consumers obviously always aim to get “good value for money”. However it seems that consumers derive special pleasure from buying a good for a price which is significantly lower than the price they perceive is the normal price for such a good.

How will the emerging electronic markets affect such bargain-hunting pursuits?

Consider a few potentially dominant trends:

- Markets are likely to become more efficient thus driving the prices of goods (especially undifferentiated ones) down towards their marginal costs, as well as potentially driving down the price variance of such goods across sellers;
- On the other hand, consumers' search costs are greatly reduced, enabling consumers to gain fast and easy access to many more potential global sellers;
Sellers have access to sophisticated technology which increasingly enables them to make comparison shopping more difficult by using techniques such as: dynamic pricing and price discrimination, bundling products to more complex packages, promoting intangible qualities such as brand and reputation, developing sophisticated customer loyalty programs, and attempting to block automatic shopping comparison agents;

On the other hand, consumers may have access to increasingly sophisticated shopping decision support software tools;

It is this last point which we believe will prove the most significant one, and is the one on which this paper focuses.

First-generation relatively simple comparison shopping tools have been emerging for the last few years (e.g. BargainFinder, bf.cstar.ac.com, Travelocity, www.travelocity.com, and Junglee www.junglee.com). The unBun model we present in this paper can be considered as a foundation for more sophisticated effective shopping decision support software tools.

Consider for example the travel industry. De-regulation and intense competition on certain popular routes, have accelerated the commoditization of flight seats, and have driven down the prices of flight ticket, as well as the price difference of tickets of competing airlines.

For example, on May 15th 1998 round-trip flight tickets Boston-London for a departure date of June 2nd 1998 were offered on the Web for $470 by British Airways (www.british-airways.com), for $470 by Virgin Atlantic (www.fly.virgin.com), and for $480 by American Airlines (www.aa.com).

On the other hand, these suppliers are also offering bundled packages, which are harder to compare, for which the price variance is significantly larger.

For example, on that same day, the following offerings were available at the above sites:

- A package for 2 adults which includes round-trip Boson-London flight on Virgin Atlantic, 6 nights at the White House Hotel (4 star – AA rating) in London with continental breakfasts, and airport transfers and a guidebook, for departure on June 2nd, was offered on that day by Virgin Atlantic for $2468 (www.fly.virgin.com).
- A package for 2 for the same target date, flying on American Airline and 6 nights at the same White House hotel, with airport transfers but also with full English breakfasts and a pair of theater tickets, though without the guidebook, was offered by American Airlines for only $2052 (www.aa.com).

Thus a rational choice of the first package may be justified if the consumer is willing to pay an extra of roughly $500 for the “Experience” of flying Virgin Atlantic.

However, on the same day and for the same target dates, flight-only offerings from Virgin Atlantic (www.fly.virgin.com) and an offering for a 6 night stay at the White House hotel offering from BHRC, a hotel booking agency (www.bhrc.com), enabled a consumer to create that package (including the Virgin Atlantic flight “Experience”) for only $1961.

Moreover, on the same day and for the same target date, a similar package with a British Airways flight, and a 6 nights stay at the Royal National Hotel was offered by British Airways (www.british-airways.com) for only $1543.
What is an effective model within which to compare such offerings?

What types of decision support tools are required to assist consumers in evaluating such offerings?

How can effective searches be conducted matching consumer requirements with offerings' feature representations?

How can related potentially interesting special opportunity bargains be identified and presented to the consumer?

In this paper we outline a foundation framework, unBun, which relies on fine-grained logical unbundling of offerings to their constituent components to support effective offering comparisons.

- Based on domain-specific ontologies for describing and assessing relevant features.
- Relies on state of the art approaches for general context mediation and wrapping to provide normalized offerings views.
- Represents uniformly product attributes as well as seller and buyer attributes.
- Defines a notion of quality dominance between products and between attributes.
- Maps between consumer requirements and lowest priced corresponding offerings.
- Augments basic consumer requirements to capture different categories of potential bargains.
- Enables controlled execution optimizations and complexity reductions.

The rest of this paper is organized as follows. In the next section we discuss previous work in a number of relevant areas. In section 3 the formal foundations of the unBun modeling of offerings are presented. In section 4 we specify the representation of consumer requirements and their relationship to offerings. In section 5 we introduce the notion of bargains and classify them to three categories. In section 6 we demonstrate the potential of the new framework in the context of realistic examples. Section 7 discusses certain optimization techniques. Finally in section 8 we provide some conclusions and outline areas of further work.

2 Related Work

A significant body of research has been carried out in a number of areas relevant to our approach.

The effects of reduction of search costs on competition in electronic markets of differentiated good were investigated by [Ba97]. [Le98, CS98] present general pricing strategies for goods in electronic marketplaces.

An empirical price comparison analysis between Web retailers of books and CDs and brick-and-mortar ones surprisingly found (among other findings) significant price dispersion
across Internet retailers (as much as 47% difference between Internet retailers for the exact same book or CD) [BS99].

[BB97] investigated the circumstances under which multiple information goods are likely to be bundled as a single offering.

In [GM98a, GM98a] current shopping comparison approaches and tools are examined and criticized, mostly due to their over-emphasize on price and under-emphasis on respective features. The use of approaches such as multi-attribute utility theory [KR76], and distributed constraint satisfaction [Ts93] is advocated.

The challenge of consistent representation and interpretation of multiple offerings from global sources is related to work on semantic mediation in large scale heterogeneous environments (e.g. [Ma96]). It has been addressed by approaches which may generally be classified to tightly-coupled ones, to loosely-coupled ones, and to hybrid ones (e.g. [Go96, BFG97]).

Relatively recent efforts has focused on defining description languages and standards for electronic commerce goods, for example CommerceNet's XML exchange (www.xmlx.com) and RDF (www.w3.org). Web-Wrapping approaches such as [BB97] enable the mapping of unstructured and semi-structured Web data to a structured form.

Approaches for generic representation of different types of goods and services are beginning to emerge: for example, information services [MW98], electronic tickets [FN98], or sale-promotion coupons [KRJ98].

3 A Market-Offerings Model

Electronic markets contain a population of buyers and a population of sellers offering multiple goods.

As buyers (or software agents on their behalf) determine what to buy, from whom, and under which conditions, they view offerings from sellers.

The set of global offerings from multiple sources (sellers) is modeled as a distributed database. Each offering is represented as a record which is normalized according to domain-based conventions. That is, we assume that domain-specific ontologies are used together with context-mediation techniques or with global-schema techniques (cf. [Ma96, Go96, BFG97]), to convert heterogeneous seller representation to the normalized view form, while avoiding syntactic and semantic ambiguities and conflicts.

A particular offering from a seller to a (possibly specific) population of buyers, at a particular offer-publication point of time at a particular price, may be associated with information attributes about three dimensions:

1. information about the offered products (i.e. what precisely is the buyer promised to get);
2. information about the offering seller (i.e. who is the entity making the promises);
3. information about a potential offered buyer (e.g. is the offering limited to a particular
For example, as part of a special promotion British-Airways Airlines may offer flights during March 1999 from Boston to London at a special low fare to MIT affiliates.

The normalized representation of each of the three dimensions of any offering is based on domain-specific generic item classes. Each item class is associated with specific named attributes. Each particular item has an assigned value for each of the named attributes of its class.

For example, a class Flight's named-attributes include among others obvious ones such as: From (which may have values representing an airport at a city); To (which may similarly have values representing an airport at a city); Class (which may have values of economy, business, or first); and Departure Date (which may have a value of a date or a range of dates). However, the Flight class also has less obvious named attributes such as, among many others, Frequent-Flyer Miles (which may have values representing number of miles a passenger will receive), and Airport-Tax (which may also have values of yes or no representing whether such tax is already included in the offer).

Note that any feature and general information can in any case be represented within unstructured descriptive attributes, containing for example free text, a picture or a Web URL. Such attributes may provide additional information for ad-hoc consumer evaluations.

Any attribute of any offered good or service, tangible (e.g. meal) or not (e.g. reliability), can be captured in the specified normalized form. Some such attribute values may be based on an assessment ranking, which is possibly subjective. Any offering may contain one or multiple products items (For example a travel package offering may include a product item of class Flight and a product item of class Hotel-Room).

For the rest of this paper we focus mostly on offerings representing products and sellers attributes only. That is, for the most part we consider offerings which are available to any potential buyer. Our discussion below may be generalized in an analogous way to capture offerings which are customized for a particular buyer or a limited population of buyers.

We categorize each named-attribute in each item class to three categories, based on the possible values they may be instantiated with:

1. **Orderable superiority** - a total order is defined which represents an is_superior_than relationship between values. For example the attribute Class of Flight may be ordered as economy, business, and first, as an increasing superiority order. Similarly, the attribute-name TravelWatch Airline Quality Assessment may be ordered c, b and a representing increasing quality.

2. **Refinable specificity** - a hierarchy is defined which represent an is_part_of relationship between values. For example the attribute To of a item-class Flight may have a value of a city (e.g. NYC) or a higher specificity value of an airport (e.g JFK airport). The attribute Departure may have a value of such Feb 15th 1999 am or a higher specificity value such as Feb 15th 1999 9:35am. The attribute Hotel-Name of a item-class Tour-

---

1 Certain features such a refund-ability may involve more complex value structure. The representation and analysis of such more complex features is a focus of our current work which we do not address in this paper.
Package may have a set value such as \{Sheraton, Marriott, Hilton\} indicating multiple possibilities, or a higher specificity value such as Hilton\(^2\).

3. **Unstructured** - attributes whose values represent additional information for ad-hoc consumer evaluations but which are not interpreted and are not formally incorporated into the offering comparison model. Such attributes may contain free text, images, or Web URLs\(^3\).

### 4 Requirement Constraints

During a purchase decision process, a consumer normally has certain basic requirements for candidate offerings. Essentially, such requirements can be regarded as these features (of products and/or sellers) which she deems as a necessary foundation.

We represent such a consumer's set of requirements as a **Requirement Constraints set** (RC). An RC is a set of items with named-attributes values (identical to the structure of an item set in an offering, described above).

In practice, such RC will be generated and instantiated by the consumer support software based on explicit input by the consumer via an appropriate interactive interface, on internal deductions based on the consumer profile or history, or on default values defined for the particular circumstances.

Two general principles underlying our notion of an RC are:

1. The consumer is, in principle, indifferent to attributes and items which are not constrained by the RC.
2. Unless over-ridden by the consumer, the constraints in the RC are interpreted as **lower bounds**. That is, it is normally assumed that if a consumer constrained a hotel to 4 stars she is willing to stay at a 5 star one (if say, everything else of relevance is equal and the price of the latter is lower). Similarly, if say a consumer constrained an offering to include a Flight item (to London) and a Hotel-Room item (3 nights), it is normally assumed that given an offering that contains both of the above items, with an additional Tour item (in London, and in which the consumer does not have to participate) and is priced lower, such an offering may be considered an acceptable candidate by the consumer.

The following definition introduces a **satisfy** relation between item sets (which are associated with either offerings or RCs). Intuitively, the **satisfy** relation between items sets captures the notion of **at least as good for the consumer**.

\(^2\)The semantics of a low specificity value in an offering falls to two categories: **seller-refinable** (e.g. a package offering that promises a night at hotel whose general location is London), and **buyer-refinable** (e.g. an airline offering of Boston-London flights during March: it is up to the buyer to select the actual date within that interval).

\(^3\)In principle as more of an item's feature are structured within our normalized format the more effective the comparison support models and tools. It is our view that an increasingly many (but certainly not all) comparison-relevant features items will eventually become normalizable.
Informally, an item set of an offering satisfies an item set of an RC if the first set contains at least all the required items specified in the second, and at least the required superiority or specificity value levels for corresponding attribute-names.

Similarly an item set of an RC is said to satisfy an item set of another RC if these conditions hold (and refinability simply being interpreted as is a-subset-of, for analogous reasons as the ones discussed above).

**Definition 1:** An item set $T_2$ is said to satisfy an item set $T_1$ (denoted $T_2 \supseteq T_1$) if for every item $i_1$ in $T_1$ there is a unique corresponding item $i_2$ of the same class in $T_2$, such that for every value of an attribute in $i_1$, there is a corresponding value in $i_2$ which is superior (for ordered attributes), refined (for refinable attributes) or equal.

The following corollary establishes the transitivity of the satisfy relation between item sets.

**Corollary 1:**
Let $S_1$, $S_2$, and $S_3$ be sets of items.

If $S_2 \supseteq S_1$ and $S_3 \supseteq S_2$ then $S_3 \supseteq S_1$.

**Proof:** By contradiction. Assume that $S_3$ does not satisfy $S_1$ while the above two pre-conditions hold. Definition 1 thus implies that there is at least one item $i'_1$ in $S_1$ for which there is no corresponding unique item, equivalent, superior, or refined in $S_3$. However due to our first assumption there is such a unique item $i'_2$ in $S_2$, equal, superior, or refined to $i'_1$. It follows, that there is no corresponding item in $S_3$ which is equivalent, superior, or refined to $i'_2$ (otherwise that item would have also been equivalent, superior, or refined to $i'_1$) - contradicting our second pre-condition.

We assume a global network environment in which offerings may be posted by any site at any point of time. We model each offering as:

$$[oid, tsoid, osetoid, priceoid]$$

where

- $oid$ is a unique identifier of an offering.
- $tsoid$ is an offering-publication timestamp.
- $osetoid$ is a set of items of the offering.

---

4 *at least the required specificity* of a refinable attribute value is defined as being a-subset-of of the corresponding (lower refinement) value if the latter is seller refinable, or as intersecting-with the corresponding value if the latter is buyer refinable. That distinction is due to the fact that the satisfy relation reflects the consumer perspective of at-least-as-good relation.

5 Note that empty or don't care values need to be defined appropriately for refinable and ordered attributes so that their meaning will be consistent with this definition. Also note that this definition implies that there may be items in $T_2$ for which there is no corresponding items in $T_1$. 

7
price\textscript{oid} is the non-negative price of the offering.

From a consumer’s perspective, under some circumstances multiple offerings may be integrated, and considered for the sake of evaluation and comparison, as a single integrated offering. Such an integrated offering can be regarded as a proposal (jointly by a number of suppliers) to provide a set of items and a particular price. In fact, a single offering may be viewed as a special case of an integrated offering, and we normally use the term offering to refer to both.

We focus on the minimal price required to satisfy a consumer’s Requirement Constraints (RC) set. Intuitively, it is the minimal price of an integrated offerings which contains at least all the required items and attributes, and is associated at least with all the required corresponding superiority and refinement levels.

**Definition 2:** The minimal price of an Requirement Constraints item set RC in an environment, denoted min.\text{price}(RC), is

\[
\text{min.\text{price}}(RC) = \min (\text{price}_{\text{oid}_j})(\forall \text{oid}_j)(\text{oset}_{\text{oid}_j} \supset \text{RC})
\]

## 5 Bargains

When evaluating offerings, a consumer’s basic aim is to find the lowest priced offering which satisfies all her requirements (or the lowest n such offerings). We denote the set of all offerings (regular or integrated) in an environment which satisfy a consumer’s constraints RC as $S_{RC}$.

As discussed above, in a preliminary evaluation phase the differences in features between the offerings in $S_{RC}$ can be considered as essentially insignificant to the consumer. Consequently, in principle the primary selection criterion among offerings in this set is often the respective price. Thus we refer to the set of offerings in $S_{RC}$ as a set of offerings with products and features which in principle have equivalent utility to the consumer (in the sense that the user is indifferent to any respective item-feature differences).

Our approach assumes however that in some cases a consumer may also be interested to consider additional offerings to the lowest priced ones. That is, the consumer may potentially be interested in also examining offerings which either satisfy a somewhat relaxed RC, or offerings in which the price pay is higher than the Min.Price.

---

6 We do not elaborate explicitly in this paper on the fact that an integrated offering may be less convenient for a consumer than a single offering because it requires her to deal with multiple sellers. In general, such a feature can itself be represented explicitly within the item set of an offering. Furthermore, we assume that the consumer can specify some limit on the extent to which she is willing to transact with multiple sellers in a given purchase transaction.

7 In this paper we mostly ignore global consistency issues arising from the asynchronous nature of information transfer (e.g. offerings being posted, modified, or terminated while a minimal price computation algorithm is being performed). In general such aspects are part of the unBun framework and indeed serve as an important basis for optimization techniques.
We focus here on these cases in which a consumer is interested in considering such additional offerings if they represent exceptional favorable market opportunities, and are related (in a sense discussed below) to her RC.

Generally, A bargain is an offering which is significantly less expensive (i.e. less than some discount threshold value) than any other offering in some comparable reference set of offerings.

In order to incorporate notions of bargains we slightly augment our consumer-requirement model. Based on a consumer RC it is possible to specify additional envelope constraints. Each envelope is a set of constraints which is either a relaxation of RC, in which case the envelope is referred to as a Downgrade Envelope Constraints (DC), or is a tightening of RC, in which case the the envelope is referred to as a Upgrade Envelope Constraints (UC).

Intuitively a Downgrade Envelope Constraints defines sets of offering each of which is associated with some inferior utility value to the consumer (compared to the reference value of these offerings satisfying RC). However the consumer may be interested in considering as a candidate an offering satisfying (only) such a Downgrade Envelope Constraints but only if it is an attractive bargain.

Similarly an Upgrade Envelope Constraints defines sets of offering each of which is associated with some superior utility value to the consumer (compared to the reference value of these offerings satisfying only RC). The consumer may be interested in considering paying a higher price for an offering satisfying such an Upgrade Envelope Constraints but only if it is an attractive bargain.

So under what circumstances can such an offering which satisfies an envelope constraint set be considered as an attractive bargain?

Intuitively it needs to satisfy at least the Downgrade Envelope constraints (DC) and its price needs to be significantly lower (according to some specified discount threshold) than either:

- all other offerings which satisfy the consumer basic requirement constraints (RC) (great price compared to what the consumer would normally have paid), or
- all other offerings which satisfy at least the Downgrade Envelope constraints (DC) and which have the same level of consumer feature-utility as the reference offering (great price compared to what the consumer would be getting)

For example, a consumer may wish to stay a week in a 4-star hotel (RC), but may be willing to consider very attractive bargains, say associated with at least a 50% discount, within an envelope that includes 3-star hotels (DC) and 5-star ones (UC). The above first

---

8 That can be changed to refer to other aggregate price values of such a reference set, such as the average price, rather than the lowest price in that set. The principles discussed in this section can analogously be adjusted to such variants.

9 Note that RC ⊈ DC and so $S_{RC} \subseteq S_{DC}$ and similarly UC ⊈ RC and so $S_{UC} \subseteq S_{RC}$. Note also that allowing a refinable attribute to assume a different value (that its RC value) can be regarded as a Downgrade Envelope: the new constraint can be considered as a set which includes both values, thus a lower specificity value, thus potentially inferior to the consumer (consistently with the definition of satisfy).
category captures for example cases in which a 3-star hotel offering (DC) is cheaper by more than 50% than the lowest 4-star offering (RC).

The second category on the other hand captures for example cases in which a 3-star hotel (or a 5-star hotel) is cheaper by more than 50% than the next lowest 3-star offering (or, respectively, the next lowest 5-star offering).

With respect to the first category, we further find it useful to distinguish between two cases:

- bargains which are defined by the respective price difference only;
- bargains which are defined by the respective price difference in relation to the actual feature-differential value;

Thus accordingly for a given bargain discount threshold $d\%$ we classify bargains to the following 3 types $^{10}$:

1. **Same-Utility-Level Relative Bargain**: An offering which satisfies at least the envelope constraints DC, whose price is lower by at least $d\%$ than the price of any other offering satisfying at least DC which has features associated with the same features-utility for the consumer (as determined from DC, RC, and UC) $^{11}$.

2. **Base-Utility-Level Relative Bargain**: An offering which satisfies at least the envelope constraints DC, whose price is lower by at least $d\%$ than the price of any offering which satisfies the base requirements of the consumer (RC).

3. **Differential-Items Relative Bargain**: An offering which satisfies at least the envelope constraints DC, whose price differential in relation to Min.Price(RC) is lower by at least $d\%$ than the price of any offering which satisfies the items differential $^{12}$.

We note that in addition to bargains that are defined in relation to product items and seller items dimensions, bargains may in principle be similarly defined with respect to transaction time dimension and buyer dimension.

A **Time Relative Bargain** is an offering whose price is lower by at least $d\%$ than the price of any offering with features associated with the same consumer feature utility which was offered within some time interval.

$^{10}$The discount threshold value, like any other input value, can either be set by the consumer once and for all (say a 50% threshold), can be set or overridden by a consumer in relations to specific envelopes, or may not be explicitly set by the user at all (implying some default value, or a value derived from the consumer environment).

$^{11}$We omit from this paper some of the technical details however in principle same-feature-utility offerings are defined as ones in which attributes corresponding envelope constraints (i.e. relaxed or tightened) need to have the same values in the bargain and in its reference set. Other attributes may range across the values defined by the RC. The intuition behind such definition is based on our model in which the set $(SDC-SRC)$ represents offerings which are acceptable to the consumer but of reduced feature-utility in varying degrees, the set $(SRC-SUC)$ represents offerings which are of equal features-utility for the consumer, and the set $SUC$ represents offerings which are of increased feature-utility in varying degrees for the consumer.

$^{12}$Again, technical details are omitted here for the sake of brevity. Intuitively, the differential price associated with the upgraded differential items (corresponding to the envelope constraints) is compared to the price of similar differential in other offerings.
Similarly, a *Buyer Relative Bargain* is an offering whose price is lower by least $d\%$ than the price of *any* offering with features associated with the same consumer feature utility which is *offered to other population of buyers*.

6 Bargain-Hunting Examples

We demonstrate the potential of the above unBun concepts using an environment consisting of certain travel packages Web offerings.

The presentation focuses on the internal core model level representation of attributes and requirements rather than on user interface level. Shopping support tools designed on top of such core model would permit for example various consumer requirement values to be deduced from the consumer profile, past behavior, or domain defaults, in addition to these selected explicitly (or over-ridden) by the consumer.

We note again that discount threshold values, like any other input, can be set by the software tool, or overridden by the consumer for particular envelopes. Note that such value will only affect whether (and which) *additional*, potentially interesting, opportunity offerings will be presented to the consumer for considerations.

The following are representations of offerings which were available (among many others) on the Web, on May 15th 1998, for round-trips from Boston to London for 2 adults, leaving on June 2nd 1998, for 6 nights.\(^\text{13}\)

\(^\text{13}\)For the sake of brevity we omitted any explicit reference to certain attributes such as timing-related ones.
## Flights + Hotel Offerings

<table>
<thead>
<tr>
<th>Offering 1</th>
<th>Offering 2</th>
<th>Offering 3</th>
<th>Offering 4</th>
<th>Offering 5</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Airline</strong>: Virgin</td>
<td><strong>Airline</strong>: American</td>
<td><strong>Airline</strong>: British-Airways</td>
<td><strong>Airline</strong>: British-Airways</td>
<td><strong>Airline</strong>: British-Airways</td>
</tr>
<tr>
<td><strong>Class</strong>: economy</td>
<td><strong>Class</strong>: economy</td>
<td><strong>Class</strong>: economy</td>
<td><strong>Class</strong>: economy</td>
<td><strong>Class</strong>: economy</td>
</tr>
<tr>
<td><strong>#-of-flights</strong>: 2 round-trips (Bos-Lon)</td>
<td><strong>#-of-flights</strong>: 2 round-trips (Bos-Lon)</td>
<td><strong>#-of-flights</strong>: 2 round-trips (Bos-Lon)</td>
<td><strong>#-of-flights</strong>: 2 round-trips (Bos-Lon)</td>
<td><strong>#-of-flights</strong>: 2 round-trips (Bos-Lon)</td>
</tr>
<tr>
<td><strong>Hotel</strong>: White House</td>
<td><strong>Hotel</strong>: White House</td>
<td><strong>Hotel</strong>: Russel</td>
<td><strong>Hotel</strong>: St Giles</td>
<td><strong>Hotel</strong>: Royal National</td>
</tr>
<tr>
<td><strong>AA-Rating</strong>: 4 star</td>
<td><strong>AA-Rating</strong>: 4 star</td>
<td><strong>AA-Rating</strong>: 4 star</td>
<td><strong>AA-Rating</strong>: 3 star</td>
<td><strong>AA-Rating</strong>: 2 star</td>
</tr>
<tr>
<td><strong>Location</strong>: central London</td>
<td><strong>Location</strong>: central London</td>
<td><strong>Location</strong>: central London</td>
<td><strong>Location</strong>: central London</td>
<td><strong>Location</strong>: central London</td>
</tr>
<tr>
<td><strong>Breakfast</strong>: continental</td>
<td><strong>Breakfast</strong>: English</td>
<td><strong>Breakfast</strong>: continental</td>
<td><strong>Breakfast</strong>: continental</td>
<td><strong>Breakfast</strong>: continental</td>
</tr>
<tr>
<td><strong>#-of-nights</strong>: 6 (double room)</td>
<td><strong>#-of-nights</strong>: 6 (double room)</td>
<td><strong>#-of-nights</strong>: 6 (double room)</td>
<td><strong>#-of-nights</strong>: 6 (double room)</td>
<td><strong>#-of-nights</strong>: 6 (double room)</td>
</tr>
<tr>
<td><strong>Transfers</strong>: standard</td>
<td><strong>Transfers</strong>: standard</td>
<td><strong>Transfers</strong>: standard</td>
<td><strong>Transfers</strong>: standard</td>
<td><strong>Transfers</strong>: standard</td>
</tr>
<tr>
<td><strong>#-of-tickets</strong>: 2 trips</td>
<td><strong>#-of-tickets</strong>: 2 trips</td>
<td><strong>#-of-tickets</strong>: 2 trips</td>
<td><strong>#-of-tickets</strong>: 2 trips</td>
<td><strong>#-of-tickets</strong>: 2 trips</td>
</tr>
<tr>
<td><strong>Theater Tickets</strong>: standard</td>
<td><strong>Theater Tickets</strong>: standard</td>
<td><strong>Theater Tickets</strong>: standard</td>
<td><strong>Theater Tickets</strong>: standard</td>
<td><strong>Theater Tickets</strong>: standard</td>
</tr>
<tr>
<td><strong>#-of-tickets</strong>: 2</td>
<td><strong>#-of-tickets</strong>: 2</td>
<td><strong>#-of-tickets</strong>: 2</td>
<td><strong>#-of-tickets</strong>: 2</td>
<td><strong>#-of-tickets</strong>: 2</td>
</tr>
<tr>
<td><strong>Total Price</strong>: $2468</td>
<td><strong>Total Price</strong>: $1962</td>
<td><strong>Total Price</strong>: $2142</td>
<td><strong>Total Price</strong>: $1762</td>
<td><strong>Total Price</strong>: $1543</td>
</tr>
</tbody>
</table>

## Hotel Offerings

<table>
<thead>
<tr>
<th>Offering 6</th>
<th>Offering 7</th>
<th>Offering 8</th>
<th>Offering 9</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hotel</strong>: White House</td>
<td><strong>Hotel</strong>: Russel</td>
<td><strong>Hotel</strong>: James</td>
<td><strong>Hotel</strong>: Savoy</td>
</tr>
<tr>
<td><strong>AA-Rating</strong>: 4 star</td>
<td><strong>AA-Rating</strong>: 4 star</td>
<td><strong>AA-Rating</strong>: 4 star</td>
<td><strong>AA-Rating</strong>: 5 star</td>
</tr>
<tr>
<td><strong>Location</strong>: central London</td>
<td><strong>Location</strong>: central London</td>
<td><strong>Location</strong>: central London</td>
<td><strong>Location</strong>: central London</td>
</tr>
<tr>
<td><strong>Breakfast</strong>: continental</td>
<td><strong>Breakfast</strong>: continental</td>
<td><strong>Breakfast</strong>: continental</td>
<td><strong>Breakfast</strong>: continental</td>
</tr>
<tr>
<td><strong>#-of-nights</strong>: 6 (double room)</td>
<td><strong>#-of-nights</strong>: 6 (double room)</td>
<td><strong>#-of-nights</strong>: 6 (double room)</td>
<td><strong>#-of-nights</strong>: 6 (double room)</td>
</tr>
<tr>
<td><strong>Total Price</strong>: $1020</td>
<td><strong>Total Price</strong>: $840</td>
<td><strong>Total Price</strong>: $1040</td>
<td><strong>Total Price</strong>: $1400</td>
</tr>
</tbody>
</table>

## Flights Offerings

<table>
<thead>
<tr>
<th>Offering 10</th>
<th>Offering 11</th>
<th>Offering 12</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Airline</strong>: Virgin</td>
<td><strong>Airline</strong>: American</td>
<td><strong>Airline</strong>: British-Airways</td>
</tr>
<tr>
<td><strong>Class</strong>: economy</td>
<td><strong>Class</strong>: economy</td>
<td><strong>Class</strong>: economy</td>
</tr>
<tr>
<td><strong>#-of-flights</strong>: 2 round-trips (Bos-Lon)</td>
<td><strong>#-of-flights</strong>: 2 round-trips (Bos-Lon)</td>
<td><strong>#-of-flights</strong>: 2 round-trips (Bos-Lon)</td>
</tr>
<tr>
<td><strong>Total Price</strong>: $940</td>
<td><strong>Total Price</strong>: $960</td>
<td><strong>Total Price</strong>: $940</td>
</tr>
</tbody>
</table>

## Theatre Tickets Offering and Airport-Transfer Offering

<table>
<thead>
<tr>
<th>Offering 13</th>
<th>Offering 14</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Transfers</strong>: standard</td>
<td><strong>Transfers</strong>: standard</td>
</tr>
<tr>
<td><strong>#-of-tickets</strong>: 2 trips</td>
<td><strong>#-of-tickets</strong>: 2 trips</td>
</tr>
<tr>
<td><strong>Theater Tickets</strong>: standard</td>
<td><strong>Theater Tickets</strong>: standard</td>
</tr>
<tr>
<td><strong>#-of-tickets</strong>: 2</td>
<td><strong>#-of-tickets</strong>: 2</td>
</tr>
<tr>
<td><strong>Total Price</strong>: $100</td>
<td><strong>Total Price</strong>: $70</td>
</tr>
</tbody>
</table>
Assume the following constitutes a consumer’s Requirement Constraints set RC ( * represents don’t care) :

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Airline:</td>
<td>*</td>
</tr>
<tr>
<td>Class:</td>
<td>economy</td>
</tr>
<tr>
<td># of flights:</td>
<td>2 round-trips (Bos-Lon)</td>
</tr>
<tr>
<td>Hotel:</td>
<td>*</td>
</tr>
<tr>
<td>AA Rating:</td>
<td>2 star</td>
</tr>
<tr>
<td>Location:</td>
<td>Greater London</td>
</tr>
<tr>
<td>Breakfast:</td>
<td>*</td>
</tr>
<tr>
<td># of nights:</td>
<td>6 (double room)</td>
</tr>
</tbody>
</table>

The Min.Price of the set offerings which satisfy this Constraint set is $1534 (corresponding to offering 5).

Assume now the following consumer’s RC :

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Airline:</td>
<td>Virgin</td>
</tr>
<tr>
<td>Class:</td>
<td>economy</td>
</tr>
<tr>
<td># of flights:</td>
<td>2 round-trips (Bos-Lon)</td>
</tr>
<tr>
<td>Hotel:</td>
<td>White House or Grosvenor</td>
</tr>
<tr>
<td># of nights:</td>
<td>6 (double room)</td>
</tr>
<tr>
<td>Airport transfers:</td>
<td>2 trips</td>
</tr>
<tr>
<td>London Guide book:</td>
<td>1</td>
</tr>
</tbody>
</table>

Further assume that a discount threshold is specified as 15% and that the following Downgrade Envelope Constraints (DC) is specified as the following possible relaxations to the RC:

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td># of nights:</td>
<td>6 → 5</td>
</tr>
<tr>
<td>London Guide book:</td>
<td>1 → exclude</td>
</tr>
</tbody>
</table>

The Min.Price satisfying RC is $2468 (corresponding to offering 1). However in addition to that offering, the consumer will also be presented with an additional bargain offering for consideration: the integrated offering { 10, 6, 13 }. That offering’s price is $2060 and it is a Base-Utility-Level Relative Bargain: It satisfies DC (essentially identical product items without the London guidebook) and its price is discounted by more than 15% compared.
to the *Min.Price* of RC. Note that this breakdown implies that in offering 1 the consumer is essentially being charged $408 for a single London Guide Book.

Assume the following consumer’s RC:

<table>
<thead>
<tr>
<th>Hotel:</th>
<th>*</th>
</tr>
</thead>
<tbody>
<tr>
<td>AA-Rating:</td>
<td>5 star</td>
</tr>
<tr>
<td>Location:</td>
<td>Greater London</td>
</tr>
<tr>
<td>Breakfast:</td>
<td>continental</td>
</tr>
<tr>
<td>#-of-nights:</td>
<td>6 (double room)</td>
</tr>
</tbody>
</table>

and the following *Downgrade envelope (DC)*:

| AA-Rating: | 5 star → 4 |

Offering 9 whose price ($1400) corresponds to the *Min.Price* of offerings satisfying RC will be presented to the user.

However offering 7 will be presented as well, as a *Same-Utility-Level Relative Bargain*. Its price of $840 is lower by more than 15% than the price of the next higher offerings ($1020) of the set of downgraded offerings with equal utility (which contains offerings 6 and 8).

Offering 8 constitutes a *Base-Utility Relative Bargain* as well.

Assume the following consumer’s RC:

<table>
<thead>
<tr>
<th>Airline:</th>
<th>*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class :</td>
<td>economy</td>
</tr>
<tr>
<td>#-of-flights</td>
<td>2 round-trips (Bos-Lon)</td>
</tr>
<tr>
<td>Hotel:</td>
<td>*</td>
</tr>
<tr>
<td>AA-Rating:</td>
<td>3 star</td>
</tr>
<tr>
<td>#-of-nights:</td>
<td>6 (double room)</td>
</tr>
</tbody>
</table>

and the following *Upgrade Constraints envelope (UC)*:

| AA-Rating: | 3 star → 4 |

Offering 4 whose price ($1762) is the *Min.Price* of the offerings set satisfying RC will be presented.

The integrated offerings \{ 10, 7 \} and \{ 12, 7 \} with a price ($1780) will be presented as well as a *Differential-Items Relative Bargain*. Its incremental price ($18) represent over a
90% discount with respect to the next higher incremental price between a similar 3 star hotel room to 4 start hotel room (which is $199).

Finally assume the following consumer’s RC:

<table>
<thead>
<tr>
<th>Airline:</th>
<th>*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class:</td>
<td>economy</td>
</tr>
<tr>
<td>#of-flights:</td>
<td>2 round-trips (Bos-Lon)</td>
</tr>
<tr>
<td>Hotel:</td>
<td>White House</td>
</tr>
<tr>
<td>#of-nights:</td>
<td>6 (double room)</td>
</tr>
</tbody>
</table>

and the following Upgrade envelope (UC):

| Theater Tickets (2) | include |

The integrated offerings \{10, 6\} and \{12, 6\} with a price ($1960) will be presented as lowest priced offer to satisfy RC.

Offering 2 will be presented as well as a Differential-Item Relative Bargain. Its incremental price of $1 represents over a 98% discount with respect to the next lowest incremental price of a pair of theater tickets (offering 14).

7 Execution Optimizations

Optimization techniques fall into two complementary categories:

- Techniques which may eliminate in a controlled manner certain sets of offerings (or offering combinations) from consideration relying instead on appropriate sampling or approximations and providing results which are (very likely) close to optimal.
- Techniques which rely on identifying and eliminating computation steps which can be deduced as redundant.

The first category of techniques enables fine-tuning of extra computation costs in a way that reflects the expected potential gain (from increasingly better offerings). The design and analysis of such techniques involves experimentation in the context of particular application domains.

The second category of techniques relies on established formal properties to reduce, where possible, redundant computation steps.

Consider for example a software shopping agent of a consumer that has direct access to certain Minimal Price values (e.g. as a result of previous pre-computed query, or as a result of intentional caching of commonly requested items). In particular let us assume that the
agent has access to the fact that the minimal price for round-trip, economy, NY-Paris flight in March is $550. Furthermore, assume that the agent also has access to the fact that the minimal price for a package for March which includes round-trip, economy, NY-Paris flight as well as a 2 night stay at a 5-star hotel is $1050.

Now assume that the consumer herself is interested in a trip to Paris for 2 nights at 4 star hotel. She specified that she is willing to settle for only a 3 star hotel if there is a particular good bargain (say if she is required to pay 50% or less for the 3-star hotel package relative to the lowest 4-star offering).

A naive execution of the above requirement would compute the minimal price of the base (4 star) requirements, followed by an evaluation of all appropriate 3-star offerings (to determine if any is cheaper by at least 50% than the minimal one of the basic requirement).

However it can immediately be deduced that no such bargain can exist and so that the computation may stop right after completing the calculation of the minimal price for the 4-star case. This is due to the fact that there exist 2 minimal prices for constraint sets, one of which represents looser constraints (flight only) than the ones corresponding to our target bargain (flight and 3-star hotel), and the other which represents tighter constraints (flight and 5-star hotel) than the reference set of the bargain (flight and 4-star hotel) - and the ratio between the min.prices corresponding to these extreme constraint sets is higher than 50%.

The following Lemma establishes this property in general, and enables to deduce in certain cases that certain bargains cannot exist.

**Lemma 1:**

Let $S_1$, $R_1$, $R_2$ be constraint item sets and oidk be an offering.

If all the following three conditions hold

1. $\frac{\text{min.price}(S_1)}{\text{min.price}(R_1)} \geq \frac{100-d}{100}$
2. $\text{osetoid}_k \gg S_1$
3. $R_1 \gg R_2$

Then

$$\frac{\text{priceoid}_k}{\text{min.price}(R_2)} \geq \frac{100-d}{100}$$

**Proof:** By contradiction.

(1): Assume the three conditions hold and $\frac{\text{priceoid}_k}{\text{min.price}(R_2)} < \frac{100-d}{100}$

(2): According to condition 3: $\text{osetoid}_k \gg S_1$

(3): Thus from definition of $\text{Min.Price}$: $\text{priceoid}_k \geq \text{min.price}(S_1)$

(4): According to condition 1: $\frac{\text{min.price}(S_1)}{\text{min.price}(R_1)} \geq \frac{100-d}{100}$

(5): We deduce from (3) and (4): $\frac{\text{priceoid}_k}{\text{min.price}(R_1)} \geq \frac{100-d}{100}$
(6): From (5) and (1): \( \min.price(R_2) > \min.price(R_1) \)

(7): According to condition 2: \( R_1 \gg R_2 \)

(8): Let oid_m be the offering of \( \min.price(R_1) \)

(9): By definition of \( \min.price \): \( oset_{oid_m} \gg R_1 \)

(10): But from (7) and Corollary 1 we deduce: \( oset_{oid_m} \gg R_2 \)

(11): Consequently by definition of \( \min.price \): \( \min.price(R_2) \leq price_{oid_m} \)

(12): However from (8) and (11) we deduce: \( \min.price(R_2) \leq \min.price(R_1) \), a contradiction with (6). \( \square \)

This and other such properties may be used to significantly reduce in some cases the computation by early elimination of redundant steps.

8 Conclusions and Further Work

In general, consumers often do not make offering selection decisions based purely on the price difference. However, the nature of electronic markets and the availability of tools based on frameworks such as \( unBun \) will increase the degree of rational quantitative selection criteria, for two main reasons:

1. It will increasingly enable the decomposition of virtually all aspects of an offering into explicit, quantifiable, dimensions (e.g. reputation of the supplier, reliability of product, etc.).

2. Applications can perform increasingly complex quantitative analysis, based on paradigm such as the ones presented here, on behalf of consumers, while retaining their conceptual and user-friendly appeal.

In this paper we have presented a new model, \( unBun \), which may serve a foundation for effective consumer shopping-support tools which assist in searching and comparing relevant offerings.

Normally a consumer inputs some requirements and is presented with the least expensive offerings which satisfy it. However, our approach enables to also find and present particular attractive opportunities which are close to the user basic requirements, in the sense that they are within some envelope defined around these requirements.

The satisfies relation in \( unBun \) between offerings and requirements and the dominance implied, enable to define notions of bargains in a meaningful and consistent manner. In consumer support software tools which are based on the \( unBun \) model, many of the consumer requirements can be derived automatically, or be over-ridden by explicit input from the user. Similarly, the acceptable bargain envelopes around the requirements can often be based on default notions of value acceptable proximity, which can also be over-ridden as required.
Optimizations techniques enable to fine-tune the performance and complexity to reflect expected potential gains.

Our current work on the unBun approach is focused on a number of main areas:

- Incorporation within unBun notions of more complex item features such as options, after-the-fact changes, cancellation, insurance, discount coupons, and others.
- Design and implementation of a Web-based prototype system.
- Empirical analysis of potential bargain offerings in selected industries.
- Experimentation with real consumers in realistic offering comparison settings.

References


