# EXPRESSING INTERVALS IN AUTOMATED SERVICE NEGOTIATION

Kassidy P. Clark, Martijn Warnier,
Sander van Splunter, Frances M.T. Brazier
*Systems Engineering*
*Faculty of Technology, Policy and Management*
*Delft University of Technology*
*The Netherlands*
[k.p.clark, m.e.warnier, s.vansplunter, f.m.brazier] @tudelft.nl

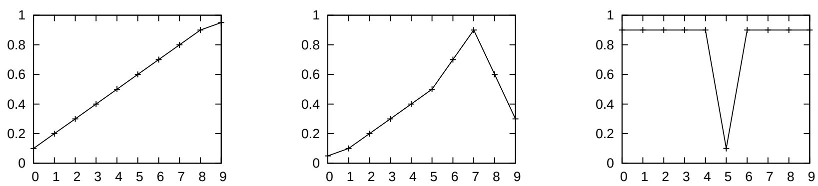
**Abstract** During automated negotiation of services between autonomous agents, utility functions are used to evaluate the terms of negotiation. These terms often include intervals of values which are prone to misinterpretation. It is often unclear if an interval embodies a continuum of real numbers or a subset of natural numbers. Furthermore, it is often unclear if an agent is expected to choose only one value, multiple values, a sub-interval or even multiple sub-intervals. Additional semantics are needed to clarify these issues. Normally, these semantics are stored in a domain ontology. However, ontologies are typically domain specific and static in nature. For dynamic environments, in which autonomous agents negotiate resources whose attributes and relationships change rapidly, semantics should be made explicit in the service negotiation. This paper identifies issues that are prone to misinterpretation and proposes a notation for expressing intervals. This notation is illustrated using an example in WS-Agreement.

**Keywords:** interval, semantics, negotiation, automation, ws-agreement, agents

# 1. Introduction

In the field of automated negotiation, the negotiation process is typically an exchange of offers between autonomous agents [1]. These agents have control over their own behavior and decision-making process. Furthermore, they can adapt to a changing environment, using different strategies and assuming different roles to achieve their goals. When an agent receives an offer, the agent evaluates the utility of the offer to determine the best course of action, such as accepting or proposing a counter-offer.

Offers specify values for the terms that can be negotiated. These terms can include discrete values, such as {*red, green, blue*} or intervals of values, such as {*between 10 and 100*}. During automated negotiation between autonomous agents, utility functions are most often used to evaluate the terms of negotiation. Evaluating the utility of a discrete value is well understood [2]; however, evaluating the utility of an interval of values is an area of ongoing research [3]. Figure 1 shows three possible utility functions with which autonomous agents can evaluate intervals. Using the example of negotiation in the energy market, each utility function can be used by agents in different roles: (a) an interval can have a rising utility, for instance, for a consumer it will hold that the lower the price, the better; or (b) utility only increases to a point and then decreases, for example, an energy consumer can store, for a low price, a certain amount of over capacity in a local battery. At the point that the battery is full the consumer can no longer profit from this cheap energy source and the utility will drop again; or (c) from the perspective of an energy provider, utility remains equal for all values, except one, such as when a value nears a sensitive threshold that requires extra effort to prepare additional resources. For instance, when a second power plant must be activated for only a small fraction of its capacity. Note that roles can change dynamically in this environment. The consumer from (b) can become a producer, by selling the energy stored in its local battery, in which case the *same* interval has a different utility.




(a) Rising (b) Single-peak (c) Dip

*Figure 1.* Examples of utility functions for interval evaluation, adapted from [3]

Many other utility functions are possible; however, to correctly compute the utility of an interval, it is first necessary to understand what is meant by that interval. It may be unclear: (1) if the choices are exclusive or inclusive; (2) if the values are real or natural; (3) if one value can be chosen or multiple, or

(4) if a sub-interval be can chosen or multiple sub-intervals. Without explicit semantics, the utility cannot be correctly interpreted.

One solution is to clearly define intervals offline, before agents enter into the negotiation arena. Another solution is to create very clear definitions and store these explicitly in a domain ontology that is available to agents during negotiation [4–5]. Domain ontologies are useful in static environments in which the domain of negotiable objects and relationships does not change quickly. However, to achieve automated negotiation in dynamic environments with autonomous agents in which resources are subject to constant change, a different approach is needed.

Moreover, in autonomous systems, agents are able to adapt to changing situations by assuming different roles at different phases of negotiation. For instance, in a dynamic energy market with decentralized co-generation of power, a consumer of electricity can also be a provider [8]. In these cases, no clear distinction can be made between the roles of client and server. As such, the range of possible actions and intentions is more difficult to define in a static ontology.

This notion of adding additional semantics to intervals has already been discussed for scheduling compute jobs on the Grid [6]. Time is an object that is inherently continuous and thus scheduling compute tasks typically consists of defining and selecting not one instant in time, but rather intervals of time. This specific scenario has been addressed by adding semantics to describe two intervals: (1) the duration of the task and (2) the larger scheduling interval between the earliest possible time to start the job and the latest possible time the job must be completed [7]. Whilst in this scenario the specific case of scheduling is addressed, a more generic set of semantics is needed for more general scenarios.

The contributions of this paper are a generic notation to express semantics to facilitate interval evaluation during automated negotiation between autonomous agents in a dynamic environment. This paper is organized in the following way. In Section 2, a negotiation scenario is presented to motivate the necessity of interval semantics. Section 3 presents the precise issues that must be addressed and describes the proposed generic notation for clarifying them. This generic notation is expressed in Section 4 using the WS Agreement specification. Finally, the conclusions and areas of future work are discussed in Section 5.

## 2. Interval Semantics

The following scenario illustrates the need for interval semantics. In this scenario, two autonomous agents negotiate the provision of electricity in a dynamic, open energy market [8]. Agent (A) wishes to purchase electricity from agent (B). Agent (B) specifies the available electricity as an offer. This document shows the available sources and attributes of electricity as illustrated in Figure 2. The assumptions made in this example include that *Provider* is an exclusive choice, as a contract is either signed with one provider or the other, but not both. Another assumption is that the *Source* is not exclusive, as a contract can contain both solar energy during the day and coal energy during the night. More chances of misinterpretation become apparent with the intervals *Base Rate* and *Quantity*. The assumption is that the price is continuous with a precision of several digits past the decimal point. However, should a single value be chosen? Or perhaps choosing a sub-interval would be better, such as between 10 and 20. In contrast, the quantity of kilowatt hours is not typically specified with such a level of precision and this interval may actually only contain discrete choices in increments of 1000. These semantics, however, are not explicit and could cause incorrect assumptions.

```
                        OFFER

Base Rate = {0 - 100}
Quantity = {0 - 10000}
Provider = {A, B, C}
Sources = {Nuclear, Coal, Gas, Wind, Solar}
Green Percent = {0 - 100}
Availability = {75 - 100}
CO2 Compensation = {green investment}
Buy-back Rate Factor = {50 - 500}
```

*Figure 2.* Resource offering

Some intervals may be described with exclusive choices. For instance, *Base Rate* is described as starting at zero, yet this is not a valid choice, but rather an exclusive lower limit. The first valid choice may actually be 1 or 0.5 or some other positive number.

When multiple choices are presented, the order of the choices may have meaning. For instance, *Source* may be ordered according to price or carbon emission. In contrast, when there is no order, it can be useful to express this fact explicitly, as well.

It is also unclear when only one value should be chosen, such as *Provider* or when multiple values should be chosen, such as *Sources*. Regarding continuous intervals, it is unclear when a value should be chosen, such as *Green Percent*, or when a sub-interval should be chosen, such as *Base Rate*. Thus, there is a chance of misinterpretation in both the meaning of the terms (e.g. inclusive or exclusive) as well as the types of choices can be made (e.g. single value, multiple values or sub-interval). Misinterpreting the meanings of these choices can lead to a suboptimal or unacceptable offer.

Often relationships and dependencies between choices need to be specified. For instance, some options may be inclusive, such as *Solar* energy can only be chosen in combination with a second energy source. Similarly, relationships between terms are important. For instance, if *Nuclear* energy is chosen, then only providers *A* and *B* are available. For intervals, the higher the *Availability*, the higher the *Price*. While these relationships could conceivably be derived from several rounds of negotiation, showing them explicitly could make for faster negotiation.

In some cases, an agent may reveal their utility function and/or preferences to its counterpart from the start [9]. For instance, one agent may inform its counterpart that, of the five sources, *Solar* is their first choice, and *Gas* is their second choice.

The semantics of an offer can also change during different phases of negotiation due to a dynamic market: products change and preferences change. A shift in demand could cause green sources of energy to become scarce, thus affecting price. Similarly, providers could add incentives, such as *CO2 Compensation*, to interest consumers in using more gray energy. A domain ontology could not store such dynamic, situation-specific information. However, these changes could be directly reflected in the semantics of the current service offer.

When negotiations are automated, any issues that require assumptions can lead to misinterpretation and an incorrect evaluation of the utility of the offer. As the example above illustrates, these semantics are not always clear. Therefore, explicit semantics are needed to describe the offer being made and the choices that can be taken. Furthermore, these semantics must be expressed in the offer rather than in a domain ontology, so an offer can react to the changes of a dynamic environment.

## 3. Expressing intervals

Each issue that is prone to misinterpretation, as summarized in Table 1, requires a clear notation that conveys the intended meaning. This notation can be added to service offers and subsequent responses to indicate the exact meaning of a term to facilitate correct interpretation and evaluation.

Figure 3 shows the same offer as before, but with added semantics. To differentiate an ordered list from an unordered list, an ordered list is surrounded by ‘$<$’ and ‘$>$’, whereas an unordered list is surrounded by ‘$\{$’ and ‘$\}$’. To indicate whether an interval’s limits are inclusive or exclusive, standard mathematical notation is used. This requires a ‘(’ or ‘)’ for inclusive and a ‘[’ or ‘]’ for exclusive. Indifference is indicated with ‘$* * *$’. All other issues use annotations that take predefined values.

Whether an interval is continuous or discrete is indicated with the annotation ‘`CD`’ that takes a letter and number as its value. If continuous, the letter ‘`C`’ is followed by a number indicating the precision. If discrete, the letter ‘`D`’ is followed by a number indicating the size of the increments.

*Table 1.* Issues prone to misinterpretation

| | |
|---|---|
| **Ordered or Unordered** | Are multiple values ordered or unordered? If ordered, what is the meaning of the order? |
| **Inclusive or Exclusive** | Are the limiting values of an interval inclusive or exclusive? |
| **Continuous or Discrete** | Is an interval continuous or discrete? If continuous, to what precision? If discrete, what are the increments? |
| **Value or Interval** | Should choices be in the form of a single value or a sub-interval? How many of each? |
| **Preference** | Is there a preference for one choice above another? |
| **Indifference** | Is a user indifferent to the value of a certain term? |
| **Relationships of choices** | Are there relationships between multiple choices? |
| **Relationships of terms** | Are there relationships between different terms? |

Whether a value or interval should be chosen is indicated with the annotation ‘VI’ that takes a letter and number as its value. If a value, the letter ‘V’ is followed by the number of values that may be chosen. If an interval, the letter ‘I’ is followed by the number of sub-intervals that may be chosen.

If one choice is preferred over another, the ‘PC’ annotation is used. This indicates that the order of the values conveys the preference.

To indicate that a relationship exists between two choices, the ‘RC’ annotation is used. This takes the value of ‘TERM:RELATIONSHIP:TERM’ where ‘RELATIONSHIP’ is a predefined term, such as ‘INCREASES’ or ‘REQUIRES’. Similarly, the ‘RT’ annotation is used to indicate relationships between two terms. This takes the value of ‘RELATIONSHIP:TERM’ and uses a set of predefined relationships, such as ‘AND’ or ‘ONLY’.

Figure 3 is interpreted in the following way: *Base Rate* is an interval that excludes the lower limit and includes the upper limit. Furthermore, it is continuous to five digits past the decimal point and one sub-interval can be chosen. *Quantity* is also an interval that excludes the lower limit and includes the upper limit. Furthermore, it is discrete with increments of 100 and a single value can be chosen. *Provider* is an unordered list and only one value can be chosen. *Sources* is an unordered list and two values can be chosen. Furthermore, either *Wind* or *Solar* can be chosen, but not both. *Green Percent* is an interval of continuous natural numbers with inclusive limits. A single sub-interval can be chosen and as this value increases, *Availability* decreases. *Availability* is an interval with an inclusive lower limit and an exclusive upper limit. Further-

```
                              OFFER

Base Rate = (0 - 100] | CD:C5, VI:I1
Quantity = (0 - 10000] | CD:D100, VI:V1
Provider = {A, B, C} | VI:V1
Sources = {Nuclear, Coal, Gas, Wind, Solar} | VI:V2, RC:Wi:OR:So
Green Percent = [0 - 100] | CD:C0, VI:I1, RT:DECREASES:Availability
Availability = [75 - 100) | CD:C0, VI=V1, RT:DECREASES:Green Percent
CO2 Compensation = {green investment} | VI:V1, RT:ONLY:A
Buy-back Rate Factor = [0.1 - 4] | CD:C1, VI:I1

                           RESPONSE

            Base Rate = [5.5 - 12]
            Quantity = {5000}
            Provider = {A}
            Sources = <Solar, Gas> | PC:YES
            Green Percent = [***]
            Availability = {99}
            CO2 Compensation = {green investment}
            Buy-back Rate Factor = [1 - 2]
```

*Figure 3.* Resource offer and response with added semantics

more, it is continuous with zero digits of precision and a single value can be chosen. *CO2 Compensation* is only available from provider “A”. Finally, *Buy-back Factor* is an interval with inclusive upper and lower limits. Furthermore, it is continuous with one digit of precision and a single sub-interval can be chosen.

The response made to the offer also uses added semantics. *Base Rate* contains an interval with inclusive limits. *Sources* is an ordered list ordered by preference. Furthermore, the agent is indifferent to the value of *Green Percent*.

## 4. Expressing intervals in WS Agreement

The WS Agreement specification defines a language with which to express agreements and a protocol to support negotiation of Service Level Agreements (SLAs) between parties [10]. WS Agreement uses XML to specify *Templates* that advertise available services and list additional constraints. Based on these templates, *Agreement Offers* are proposed until the parties reach an agreement.

Continuing the earlier example of energy provision, annotations will take the form of XML tags and can be added to the XML schema to resolve issues prone to misinterpretation. Figure 4 shows the same service offer and response using XML for WS Agreement. Two parties negotiate the provision of energy. The provider advertises the available choices in a template using additional semantic tags, as introduced above.

Instead of the ‘(’ and ‘)’symbols, the *min-* and *maxExclusive* tag is used, and instead of the ‘[’ and ‘]’symbols, the *min-* and *maxInclusive* tag is used. This can be seen in the *baseRate* and *quantity* items. To express ordering, an

TEMPLATE

```
<wsag:Item wsag:name="baseRate" CD="C5" VI="I1">
  <minExclusive="0"/>
  <maxInclusive="100"/>
</wsag:Item>
<wsag:Item wsag:name="quantity" CD="D100" VI="V1">
  <minExclusive="0"/>
  <maxInclusive="10000"/>
</wsag:Item>
<wsag:Item wsag:name="provider" VI="V1">
  <list ordering="NONE">
    <enum value="A"/>
    <enum value="B"/>
    <enum value="C"/>
  </list>
</wsag:Item>
<wsag:Item wsag:name="sources" VI="V2">
  <list ordering="NONE">
    <enum value="Nuclear"/>
    <enum value="Coal"/>
    <enum value="Gas"/>
    <enum value="Wind"/>
    <enum value="Solar"/>
  </list>
  <RC="Wi:OR:So">
</wsag:Item>
<wsag:Item wsag:name="greenPercent" CD="C0" VI="I1">
  <minInclusive="0"/>
  <maxInclusive="100"/>
  <RT="DECREASES:availability"/>
</wsag:Item>
<wsag:Item wsag:name="availability" CD="C0" VI="V1">
  <minInclusive="75"/>
  <maxExclusive="100"/>
  <RT="DECREASES:greenPercent"/>
</wsag:Item>
<wsag:Item wsag:name="co2Comp" VI="V1">
  <enum value="green investment"/>
  <RT="ONLY:A"/>
</wsag:Item>
<wsag:Item wsag:name="buyBackFac" CD="C1" VI="I1">
  <minInclusive="0.1"/>
  <maxInclusive="4"/>
</wsag:Item>
```

OFFER

```
<wsag:Item wsag:name="baseRate">
  <minInclusive="5.5"/>
  <maxInclusive="12"/>
</wsag:Item>
<wsag:Item wsag:name="quantity">
  <enum value="5000"/>
</wsag:Item>
<wsag:Item wsag:name="provider">
  <enum="A"/>
</wsag:Item>
<wsag:Item name="sources" PC="YES">
  <list ordering="PREFERENCE">
    <enum value="Solar" rank="0"/>
    <enum value="Gas" rank="1"/>
  </list>
</wsag:Item>
<wsag:Item name="greenPercent">
  <enum value="***"/>
</wsag:Item>
<wsag:Item name="availability">
  <enum="99"/>
</wsag:Item>
<wsag:Item wsag:name="co2comp">
  <enum="green investment"/>
</wsag:Item>
<wsag:Item wsag:name="buyBackFac">
  <minInclusive="1"/>
  <maxInclusive="2"/>
</wsag:Item>
```

*Figure 4.* WS-Agreement template and offer with semantics

additional *ordering* tag is used, instead of ‘{’ and ‘$<$’ as used earlier. Additionally, a “rank” value with ascending order is added to each element in the list as XML does not natively support ordering of elements. This is illustrated in the *sources* item of the offer. When an agent wishes to indicate preferential ordering, the value of this tag is modified in the offer.

## 5. Conclusion

Understanding the meaning of offers is crucial to negotiation. This is especially true for automated negotiation between autonomous agents in dynamic environments. Offers often contain intervals of choices, which are prone to misinterpretation and thus require additional semantics. This extra knowledge is often stored in a static domain ontology. However, this solution is not well suited to a highly dynamic environment of autonomous agents, in which resources, relationships and user's roles are constantly changing. This paper focuses on the specification of intervals of choices during negotiation. The interval semantics proposed are domain independent and self-contained in negotiation offers. In addition to preventing misinterpretation of intervals, these semantics also express dynamic relationships between intervals.

Future work on this topic will investigate how to better express relationships and preferences that change dynamically. For instance, as a deadline approaches, the need for a successful agreement increases and the need for a particular attribute decreases. Furthermore, research will focus on the challenge of adapting negotiation strategies and utility functions during dynamic negotiations.

## Acknowledgments

This work is supported by the NLnet Foundation (`www.nlnet.nl`).

## References

[1] N. Jennings, P. Faratin, A. Lomuscio, S. Parsons, M. Wooldridge, and C. Sierra. Automated negotiation: prospects, methods and challenges. *Group Decision and Negotiation*, 10(2):199–215, 2001.

[2] H. Raiffa. *The art and science of negotiation: How to resolve conflicts and get the best out of bargaining*. Belknap Press, 2002.

[3] R. C. van het Schip, S. van Splunter, and F. M. T. Brazier. Template evaluation and selection for ws-agreement. *Service Level Agreements in Grids Workshop*, 2009.

[4] G. Tondello and F. Siqueira. The QoS-MO ontology for semantic QoS modeling. In *Proceedings of the 2008 ACM symposium on Applied computing*, pages 2336–2340. ACM New York, NY, USA, 2008.

[5] H. Jin and H. Wu. Semantic-enabled specification for Web Services agreement. *International Journal of Web Services Practices*, 1(1-2):13–20, 2005.

[6] H. Ludwig, T. Nakata, O. Wäldrich, P. Wieder, and W. Ziegler. Reliable orchestration of resources using ws-agreement. *Lecture Notes in Computer Science*, 4208:753, 2006.

[7] D. Battrë, O. Wäldrich. Time Constraints Profile, Version 1.0. In *Global Grid Forum GRAAP-WG, Draft, October*, 2009. `http://www.ogf.org`.

[8] E. Ogston and F. M. T. Brazier. Apportionment of control in virtual power stations. *In the proceedings of the international conference on infrastructure systems and services 2009: Developing 21st Century Infrastructure Networks*, 2009.

[9] E. Oliveira, J. Fonesca, and A. Steiger-Carca. Multi-criteria negotiation on multi-agent systems. In *Proceedings of the First International Workshop of Central and Eastern Europe on Multi-agent Systems (CEEMAS'99)*, 1999.

[10] A. Andrieux, K. Czajkowski, A. Dan, K. Keahey, H. Ludwig, J. Pruyne, J. Rofrano, S. Tuecke, and M. Xu. Web Services Agreement Specification (WS-Agreement). In *Global Grid Forum GRAAP-WG, Draft, August*, 2004.