# Quixote: Supporting Group Decisions through the Web

J.A. Rubio, D. Rios Insua, J. Rios, and E. Fernandez

Decision Engineering Lab
U. Rey Juan Carlos-DMR Consulting Foundation, Spain
jarubio@bayes.escet.urjc.es, {drios, jmrios, e.fernandez }@escet.urjc.es

**Abstract.** We describe a web-based architecture to support participation in group decision making. Emphasis is placed on security aspects related with our architecture, designed to enhance trust on the system.

## 1 Introduction

There is a current debate and many attempts to provide on-line support to democratic processes, so as to change the way people interact with governments. As an example, the URJC Strategic Planning Council is collecting feedback via computer from the URJC community (students, staff, lecturers) on how the university should be organized by year 2010.

Indeed, we view Internet as an opportunity to bridge the gap between governors and governees. Our current democratic institutions stem from times in which transportation and communications were difficult and time consuming. With the time, politics have evolved little and politicians have developed a style in which, except at political campaigns, they have little feedback from citizens.

We believe, however, that most ideas so far relating Internet and politics, are directed towards *facilitating* traditional political methods through new technologies: a political Internet discussion forum, rather than a political meeting; fundraising through the web, rather than through letters or telephone calls;... Our feeling is that there are ways to *transform*, rather than facilitate, politics: there are much more constructive and creative ways of involving citizens. Indeed, it is a tenet of ours that involving and communicating with the stakeholders at all stages of a decision making process leads to more consensual and better quality decisions.

Independently of Information Technology, many authors have discussed on pros and cons of increasing participation in political decision making, with the traditional debate between participatory and representative democracy. In [8] Pateman describes

> The current and widely accepted theory of democracy attaches very little importance to the concept of participation, and even emphasizes the dangers inherent in widespread popular participation in politics. The recent upsurge of demands for participation raises the question of its place in a realistic modern theory of democracy.

The discussion has indeed been enhanced by the perceived potentiality of new technologies to affect democratic processes. As an example, Brzezinski, see [1], even as early as in 1970, claimed

> We should increasingly ensure true participation in decisions which seem too complex and too far apart from the common citizen.

These ideas have been put to a extreme by the, now called, *cyberutopians*, one example of whom is Morris who, in his VOTE.COM, see [7], implicitly suggests a permanent e-referendum system:

> As direct democracy takes root, the American voter will become more involved and active. We do not have to wait anymore for the next election to express our view while the Congress makes decisions for us. We do not have to wait for a call from a pollster to speak our piece. We are going to take the Internet and tell our representatives what to do whenever we feel like it.

To counterbalance, we would like to just point out a recent note in *Wired*, referring to Bobbitt's *The Shield of Achilles* and entitled *Technology is killing democracy*!!!

We follow here an intermediate path, presenting a web based architecture aimed at distributing rationality to better resolve political decision making. By this, we mean helping groups through the web facilitating them the use of decision and negotiation analysis methods, see [2]. Ideologically, a close view is given in [5]:

> It's not a matter of allowing masses of individuals to vote instantaneously on simple questions posed by telegenic demagogues, but to promote collective and continuous elaboration of solutions and their cooperative solution, as close as possible to concerned groups.

The structure of the paper is as follows. In section 2, we provide a general description of our architecture and methods. We then provide a brief outline of its implementation. A crucial issue refers to trust and confidence in the system, which we promote through the use of cryptographically secure open truthful exchange (CSOTE) methods as compared with the FOTE and POTE frameworks described by Raiffa in [9]. We end up with a brief discussion.

## 2 Our Architecture

To some extent, we propose migrating to Internet the philosophy and methodology of decision conferencing, see [6], to support group decision processes. This may be seen as an asynchronous Internet based implementation of decision conferences. Note that standard decision conferences are synchronous and could be implemented, e.g., through videoconferences. But that would be yet another application of new technologies to standard political approaches.

In our architecture, QUIXOTE, the decision analysis would be carried out by a decision analyst or team of analysts on a master system for the decision making problem owner (the president of a government, the CEO of a company,...). The system would provide support for the entire decision making cycle, using computer aided brainstorming, problem structuring techniques and various quantitative modelling techniques such as probability assessment tools, multi-attribute utility elicitation or outranking methods, as appropriate to an application. At various stages of the process, some or all of the models would be fed onto a server, which could be accessed by different stakeholders and the general public. The level of access would vary from stakeholder to stakeholder and at different stages of the decision process. Initially, the server provides pages simply stating that an issue is being addressed and inviting comments and submissions via email. Later, pages are developed actively which allow users to interact with the model to explore the implications of their individual perspectives and judgements. These explorations could be kept private if the user so wishes, but more usefully provide the problem owner with a summary of the stakeholders' views in a format entirely compatible with the decision model. The interactions are supported by the Internet with confidence built in through a cryptographically secure open truthful exchange (CSOTE) approach.

Note that, typically, as the problem owner and various stakeholders would have different values and beliefs, they would opt for different alternatives. It might be beneficial, therefore, to enter into a negotiation round, in which a more consensual solution might be sought. This, again, is supported through the web. Finally, in some contexts, it could happen that no consensus is reached after negotiations: we may appeal to a voting scheme supported as a way to fix a course of action.

From a methodological point of view, the key part of our system is the negotiation module. We describe now the underlying methods. Specifically, we aim at supporting a group of $n$ persons facing a joint decision making problem. Each person $i$, has to choose an action $a_i$ from a feasible set $A_i$, $i = 1, \dots, n$. This includes the case in which $A_i = A$, for all $i$, and parties should commonly implement the same alternative $a_i = a$. He also faces consequences $c_i(a_1, \dots, a_i, \dots, a_n, \theta)$, which depend on the actions of all persons and some common states of nature $\theta$. We assume that each person has his own utility function $u_i$ and probability distribution $p_i$ and is a expected utility maximiser, see [2] for a full exposition on this. This means that, assuming that the other persons fix their actions $a_1, \dots,$ $a_{i-1}, a_{i+1}, \dots, a_n$, the $i$-th negotiator will aim at solving the problem

$$\max_{a_i \in A_i} \Psi(a_i) = \int u_i(a_1, ..., a_{i-1}, a_i, a_{i+1}, ..., a_n, \theta) p_i(\theta) d\theta \qquad (1)$$

Therefore, we have the expected utility of each joint action $a = (a_1, \dots, a_n)$, obtained by each person and its optimal alternative.

In general, negotiators will not be able to maximise their expected utility simultaneously and we should enter into some kind of negotiation. We shall assume that the set of negotiators behave optimally in a Pareto sense. To wit, we

**Table 1.** This table shows the expected utility of each joint action $a$ obtained by each person and its optimal alternative

| | | Negotiator | | | |
|---|---|---|---|---|---|
| | | 1 | 2 | | $n$ |
| Action | $a$ | $\Psi_1(a)$ | $\Psi_2(a)$ | $\dots$ | $\Psi_n(a)$ |
| Optimal Action | | $a_1^\star$ | $a_2^\star$ | | $a_n^\star$ |

say that the choice $a = (a_1, \dots, a_n)$ is dominated by the choice $b = (b_1, \dots, b_n)$ if $\Psi_i(a) \leq \Psi_i(b), \forall i = 1, \dots, n$, with strict inequality for one of the negotiators. When convenient, we shall use $\Psi(a)$ to designate $(\Psi_1(a), \dots, \Psi_n(a))$. In consequence, an agreement should be sought within the nondominated set, as it guarantees that there is no other alternative unanimously preferred by all the negotiators. Typically, the set of nondominated solutions does not, however, include a unique solution. To help the negotiators in reaching consensus within the nondominated set, we propose using a modification of Raiffa's balanced increment method, which we briefly summarise here, see [10] for full details.

To do so, we need the concept of bliss point associated with an alternative $a$.

**Definition 1.** *For a given solution $a$, and each negotiator $i$, $i = 1, \dots, n$, consider the expected utility $\Psi_i^*$ of a feasible solution $x^i \in A$ which maximises the i-th's negotiator expected utility subject to $\Psi_j(x^i) \geq \Psi_j(a)$, $\forall j \neq i$. The bliss point associated with $a$ is $(\Psi_1^*, \dots, \Psi_n^*)$.*

To fully associate it with $a$, we designate it as $(\Psi_1^*(a), \dots, \Psi_n^*(a)) = \Psi^*(a)$. Note that, in some sense, the bliss point represents the ideal expected utilities achievable, should the current alternative or status quo be $a$. The diagonal linking $\Psi(a)$ and $\Psi^*(a)$, which we designate by $[\Psi(a), \Psi^*(a)]$, provides a balanced improvement direction, there being a nondominated alternative $K[\Psi(a)]$ whose associated expected utilities are in such diagonal, under appropriate technical conditions.

We now define the concept of a balanced increment solution, see [9] for further details.

**Definition 2.** *The balanced increment solution $R_\alpha(\Psi^0)$ is the limit point in the utility set of the sequence $\{\Psi^t\}$ defined by:*

$$\Psi^t = \Psi^{t-1} + \alpha(\Psi(K[\Psi^{t-1}]) - \Psi^{t-1}) \tag{2}$$

*where $\alpha \in (0, 1)$ and $\Psi^0$ represents the utilities achieved if negotiators fail to agree. A continuous version of the solution is obtained by letting $\alpha \to 0$, $R_\alpha(\Psi^0) \to R(\Psi^0)$.*

Technically, the continuous BIM solution is obtained as a double limit. We implement it in the discrete version by starting at a given solution, computing its bliss point and moving a certain fraction, say one eighth, in the line segment between $\Psi^t$ and $\Psi(K[\Psi^t])$. Also, at each step we may offer $K[\Psi^t]$, the nondominated solution in the diagonal with the bliss point, to reach a consensus if parties accept it. We also introduce, as stopping rule, that the process terminates when the parties agree on the solution $K[\Psi^t]$ offered, or $\Psi^t$ is close enough, in terms of expected utilities, to the nondominated set, reaching no consensus.

To sum up, the modified balanced increment algorithm we implement is

1. Initialisation:
   - Start at $a^0$ with $\Psi^0 = \Psi(a^0)$, $t = 0$.
   - Compute $K[\Psi^0]$ (the nondominated solution in the diagonal with the bliss point).
2. Repeat
   - If parties agree on $K[\Psi^t]$, stop.
   - If $\Psi^t$ is close to $K[\Psi^t]$, stop.
   - Ow, move fixed fraction $\alpha$ in $[\Psi^t, \Psi(K[\Psi^t])]$, obtaining $\Psi^{t+1}$.
   - $t = t + 1$
   - Compute bliss point of $\Psi^t$ and $K[\Psi^t]$.

As indicated, when the algorithm does not lead to an agreement we may apply a voting scheme.

## 3 System Modules

The current implementation uses a web based system using the LAMP (Linux, Apache, MySQL, PHP) environment. There are three basic types of users:

- the problem owner, the entity which aims at solving a decision making problem, structures and publicizes it.
- the stakeholders or participants, who provide input (beliefs, preferences, votes) to the decision making process.
- the administrator, who takes technical care of the process development, from supporting the problem owner to structure the problem, to providing access rights to stakeholders, to defining time windows for voting.

Appropriate safety mechanisms are available, as we shall later describe in detail. In particular, participants will use secure validation mechanisms. The following system modules are included:

### 3.1 Problem Structuring

The system includes a module that allows the problem owner to build an influence diagram to structure the incumbent decision making problem, in terms of uncertainty, decision and value nodes. If needed, the problem owner may be aided by the administrator to build the diagram, which acts, in this case, as a decision analyst or facilitator. The module allows for:

- Adding and removing nodes.
- Adding and removing arcs.
- Editing and modifying the tables associated with the nodes.
- Saving a diagram for later evaluation.

The stakeholders will use the same structure to explore issues of interest concerning the problem.

### 3.2 Preference Modelling

The system includes a module that allows users to build their preference model. It is assumed that any user (problem owner, stakeholder) may build his own utility function, given the usability of the model developed. Each user will assess his utility function privately and communicate it to the system. Without much loss of generality, we assume that the users' preferences may be modelled through a weighted additive utility function, see [2] for details. The system allows for:

- Specification of basic properties of (multiple) objectives by the problem owner: number of objectives, their scale and range, whether the objective is to be minimised or maximised. It is assumed that all participants will share these objectives; some participants may disregard some of these objectives, by giving them zero weight.
- Assessment of each component utility function. For each objective, and each user, the utility of some attribute values is assessed with the probability equivalent method. Then, a concave-convex or convex-concave (piecewise exponential) utility function is fitted through least squares.
- Assessment of the weights of the additive utility function, again with the aid of the probability equivalent method.
- Saving the utility function for later purposes.

Users are expected to provide their preferences within a given time window. If agreed, summaries and or comparisons of utility functions may be obtained.

### 3.3 Problem Solving

Once with the preferences of a participant, we may proceed to compute his optimal alternative. For that purpose, the system includes a module that allows users to evaluate the influence diagram, based on his utility function, to obtain his preferred maximum expected utility course of action. The problem owner may find out his optimal alternative privately, as the stakeholders may do. If wished, they may make public their solutions. Alternatively, summaries of the obtained solutions may be provided.

### 3.4 Negotiating

Typically, the various parties involved (problem owner, stakeholders) will reach different optimal solutions. Consequently, a round of negotiations may be undertaken to try to reach a consensus. The negotiation is driven by our modification of the balanced increment method as explained above.

At each iteration, the system offers a solution to participants and, if accepted, it stops, that being a consensus. Alternatively, the procedure stops when two of the subsequent solutions offered are close enough. If the last one is accepted, a consensus is reached. At each iteration, users are expected to communicate whether they accept or not an offer within a given time window.

### 3.5 Voting

Our (automatic) negotiating scheme converges to a nondominated solution, but it is conceivable that participants may not accept such solution, neither the sequence of solutions offered. This deadlock could be solved through voting. For that reason, our system includes a voting module, which permits the design of a voting session, with several voting rules available, and its execution. Specifically, the voting rules implemented are:

- Plurality: Participants may vote for just one alternative. The winner is that receiving the biggest number of votes.
- Approval voting: Participants may provide at most one vote to as many alternatives as they feel like. The winner is that receiving the biggest number of votes.
- Cumulative voting: Each voter has $m$ votes which he may distribute however he wants among the alternatives. The winner is that receiving the biggest number of votes.
- Borda count: Given that there are $k$ alternatives, Borda count asks voters to rank them increasingly ($k$ to the best,..., 1 to the worst alternative).The winner is that receiving the biggest number of votes.

Users are expected to vote within a specific a time window.

## 4 Csote concepts in Quixote

It has been frequently discussed that a critical issue in applications related with e-democracy and e-government is the confidence and trust among system users. This is, for example, stressed in the recent report *Development of the Information Society in Spain (2002).* For this reason, we have built on what we call the CSOTE framework, see [11] for details, in contrast with the FOTE and POTE frameworks described by Raiffa, see [9].

By CSOTE, we understand cryptographically secure open truthful exchange of information among participants and the system, enhancing reliability of all processes, achieved through:

- Confidentiality of system data, which will be accessible to only authorised parts.
- Communications security, therefore protecting bidirectional channels user-system.
- Data integrity, so that they are only modifiable by data owners.
- Accesibility, that is, potential ways of mitigating system attacks.

These issues may be achieved through cryptographical methods. Specifically, we shall sketch how public key cryptographical methods, see [3, 4], aid us in developing a more open, flexible and reliable framework for negotiations, in which the involved parts may reveal their real objectives and, possibly, achieve satisfactory agreements. This also aids us in automating negotiation processes as we may support all the issues we are interested in.

### 4.1 Interaction with the System

As we have mentioned, at various points, a participant must send his information to the system and, possibly, may wish to obtain a summary of the opinions of other users. For these purposes we use:

- *Partial secret revealing techniques.* They are based on the global knowledge of a function $f(x_1, ..., x_n)$, with each party knowing and revealing only a part $b_i$ of the information about the domain of $f$ and ignoring the rest, their aim being finding out the value of $f(b_1, ..., b_n)$. Two applications are:
  - Facilitate comparison of the utility functions of participants, should they wish to, with minimal revealing of information. For example, should they all have constant risk averse utility functions, they could find out which of them is more risk averse.
  - In a similar fashion, users could compare their weightings of various objectives, without revealing their exact values, to find out, e.g., which participant gives bigger importance to a criterion.
- *Zero knowledge techniques.* Specifically, we use ElGamal's cryptosystem as a basis to determine the equality or inequality of two discrete logarithms and build knowledge proofs to verify users' information. One application is:
  - The users may obtain any kind of partial information about other users' preferences, with them revealing no more than the necessary information, and therefore protecting the rest of it. The users might undertake these tests interactively, or not. In the latter case, the preferences and actions of the users could be verified publicly by system users.

### 4.2 Negotiation Phase

CSOTE is specially relevant during the negotiation rounds, the key part of our architecture. Once the users' preferences have been stored, we may proceed to look for alternatives satisfying the involved parts. For that purpose, we use the following methods:

- *$(k, n)$-threshold schemes.* An interesting piece of information (a secret) is divided in parts, in such a way that we only need to know $k < n$ of them to recover the secret. We can do this using modular arithmetic, the Chinese Remainder Theorem or Lagrange coefficients. We apply them, for example, to ensure that the choice of a certain action is made only if $k$ users out of the existing $n$ go for it.
- *Selling of secrets techniques.* With them, a buyer chooses what (secrets) to buy, with the seller not knowing what he has sold. We use the RSA cryptosystem, generating $k$ cryptosystems, with special features, together with Jacobi symbols to enhance transparency in negotiations. To wit, with them, users may undertake exchanges and, moreover, negotiate over any set of elements or objectives in a transparent manner. Moreover, each user has the possibility of weighting their choices, as well the importance of the elements to negotiate.

- *Matching protocols.* We use ElGamal's cryptosystem, Diffie-Hellman's key agreement technique and zero knowledge proofs to check correctness of computations. For example, we undertake privately computations about users' objectives or preferences protecting locally the users' data. We could also use them to create groups with certain affinity in terms of their choices.
- *Verifiable public auction schemes.* We use mechanisms such as signatures to prove knowledge of a discrete logarithm as a mechanism for anonymous signatures, which serves us as anonymous certifier for all users. In such a way, for example, we induce auction schemes generating honesty, as, at each round, we would only need to know the highest stake. As the process would be totally blind to users, they would leave apart strategic behavior.
- *Strong proxy signatures.* With them, a user may delegate all his negotiation capacity to an agent, which would take care of such process within the system. Note that with the use of delegated signatures, we could condensate in a single agent the requirements of a large number of users, which may be specially relevant if many users take place in the negotiation process, as could happen, e.g., in a participatory budget elaboration.

### 4.3 Voting Phase

Again CSOTE supports the voting phase, in case no consensus is reached during negotiations. To undertake this process, we must take into account the transparency, the capacity of the user to verify his vote and, last but not least, the capacity to avoid that a voter is not able to give a proof of his vote to another entity, so as to avoid vote dealing.

Indeed, this is an area in which developers of e-voting systems are spending lots of effort. However, most known schemes are based on critical assumptions which make them vulnerable, including the existence of truthful communications and a truthful third part. Alternatively, we opt for using schemes which include a Tamper-Proof Randomizer (TPR), which allows us to randomise votes, leaving aside the need to use reliable third parts.

Once this randomisation has taken place, we prove to the voter the correctness of the process through re-encryptation proofs through a designated verifier, which, if appropriate, generates a valid vote certificate based on validity and difference proofs. Finally, these proofs and the final vote will be signed digitally by the voter's TPR to provide credibility to the vote.

## 5 Conclusions

We have outlined issues concerting trust and confidence, which are key in e-democracy applications, within a web based group decision support system. Case studies within the area of participatory budget elaboration will be reported in a forthcoming paper.

## Acknowledgments

Research supported by grants from MEC(TSI2004-06801-C04-01), the Decision Engineering Lab (URJC - DMR Consulting Foundation) and the ESF-TED program.

## References

1. Brzezinski, Z.: *Between two ages: America's role in the technotronic era.* Viking Press (1970)
2. French, S.,Rios-Insua, D.: *Statistical Decision Theory.* Arnold (2000)
3. Goldreich, O.: *Modern Cryptography, Probabilistic Proofs and Pseudorandomness.* Springer-Verlag (1999)
4. Lee, B.: *Zero-knowledge proofs, digital signature variants, and their applications.* PhD thesis, ICU University (2001)
5. Levy, P.: *L'Intelligence Collective.* Dunod (1995)
6. McCartt, A., Rohrbough, J.: Evaluating group decision support system efectiveness. *Decision Support Systems* **5** (1989) 243–253
7. Morris, D.: *Vote.com: How Big-Money Lobbyists and the Media are Losing Their Influence, and the Internet is Giving Power to the People.* Renaissance Books (1999)
8. Pateman, C.: *Participation and Democratic Theory.* Cambridge University Press (1970)
9. Raiffa, H.: *Negotiation Analysis.* Harvard UP (2002)
10. J. Rios and D. Rios-Insua. Negotiation over influence diagrams. Tech Rep on Statictis and Decison Sciences, Rey Juan Carlos University (2004)
11. Rubio, J. A.: Sobre métodos criptográficos en análisis de negociaciones. Master tesis, Rey Juan Carlos University (2004)