



2007

The Hidden Beauty of the Quadratic Market Scoring Rule: A Uniform Liquidity Market Maker, with Variations

Michael B. Abramowicz

George Washington University Law School, abramowicz@law.gwu.edu

Follow this and additional works at: http://scholarship.law.gwu.edu/faculty_publications

 Part of the [Law Commons](#)

Recommended Citation

Abramowicz, Michael B., "The Hidden Beauty of the Quadratic Market Scoring Rule: A Uniform Liquidity Market Maker, with Variations" (2007). *GW Law Faculty Publications & Other Works*. Paper 244.

http://scholarship.law.gwu.edu/faculty_publications/244

This Article is brought to you for free and open access by the Faculty Scholarship at Scholarly Commons. It has been accepted for inclusion in GW Law Faculty Publications & Other Works by an authorized administrator of Scholarly Commons. For more information, please contact spagel@law.gwu.edu.

The Hidden Beauty of the Quadratic Market Scoring Rule: A Uniform Liquidity Market Maker, with Variations

Michael Abramowicz

The George Washington University Law School

2000 H St. NW

Washington, DC 20052

(202) 994-9704

abramowicz@law.gwu.edu

Classically, prediction markets depend entirely on the willingness of market participants to enter into trades with one another. Web sites like Tradesports and the Iowa Electronic Markets operate on the principle of a “continuous double auction.” Traders enter the prices at which they are willing to purchase or sell shares. Orders are matched where possible, or else put on “bid” and “ask” queues. The market prediction can be obtained by considering either the most recent price or the midpoint of the bid-ask spread. A limitation of this approach is that in some markets, bid-ask spreads can be very large. This may be because potential traders are worried that trading partners may have inside information, or simply because there is limited interest in trading. In any event, in these circumstances, classically structured prediction markets do not produce accurate point estimates of probabilities.

Because of this liquidity problem, there has been substantial interest in developing alternative market structures. Maxwell and Burns (1996) explains how to create a virtual specialist that adjusts market prices by identifying imbalances in pending buy and sell orders, while Abramowicz (1999) would require the most recent trader in effect to become a specialist at a price determined by the seller. In the past few years, two alternative classes of mechanisms have been developed with the virtue that mathematical formulas can calculate the price entirely on the basis of past transactions. First, Hanson (2003) has described a “market scoring rule,” and Hanson (2006) particularly emphasizes the virtues of one version of this approach, the logarithmic market scoring rule. The logarithmic approach has now been widely adopted, for example by Microsoft for its internal markets and by Inkling Markets, a provider of prediction market software on the web. Second, Pennock (2004) describes a variation on pari-mutuel betting, called the dynamic pari-mutuel market. Yahoo!, Pennock’s employer, uses this approach in its Tech Buzz market.

In this article, I will argue that while both the logarithmic scoring rule and the dynamic pari-mutuel market have uses in particular contexts, a different form of the market scoring rule, the quadratic market scoring rule, is better suited to serve as a foundation for many prediction market applications. The central point is that with the quadratic market scoring rule, the profit that a user can make by improving a prediction is constant across the probability spectrum (or, for nonprobabilistic estimates, the prediction spectrum), and so the subsidy is constant as well. The quadratic market scoring rule is thus equivalent to what I call a “uniform liquidity automated market maker.” Given the relatively small subsidies for prediction markets to date, what predictive improvements a subsidy rewards may not be so important, but the issue should have greater importance in potential future markets with greater subsidies, such as markets designed to provide information to corporations or governments.

Even if uniform liquidity is not desirable, the approach can be easily modified to allow for subsidy variability—either for different portions of the probability or prediction spectrum, or across time—in any way specified by the market administrator. Creators of subsidized markets should be considering directly how to distribute market subsidies, rather than relying implicitly on the default approach of the logarithmic market scoring rule (though at times, a logarithmic distribution may be exactly what is appropriate for the decision task) or the dynamic pari-mutuel market. In addition, the approach that I describe lends itself to a simple user interface that even those who do not understand markets should be able easily to understand, and the quadratic market scoring rule can be easily adapted to make nonprobabilistic predictions.

I. ROBIN HANSON’S MARKET SCORING RULE

Hanson’s market scoring rules are built on a statistics literature on “scoring rules,” of which DeGroot and Fienberg (1983) provide a useful overview. A scoring rule is simply a formula used to provide an incentive for a predictor to make an accurate prediction, where more accurate predictions generally earn greater amounts of money. “Strictly proper” scoring rules are scoring rules that should induce risk-neutral actors to make

honest probabilistic assessments, by providing them maximum expected payoffs from announcing accurate probabilistic beliefs.

The market scoring rule adapts scoring rules to a context in which individuals make successive predictions. It reflects two clever insights. The first insight is that instead of paying each of the predictors according to the scoring rule, the market sponsor can pay each predictor according to the amount by which the predictor *improves* on the prediction. That is, each predictor receives the payoff according to the scoring rule applied to the predictor's own prediction, reduced by the payoff according to the scoring rule applied to the previous prediction. In effect, any new predictor can come along and make a new prediction, receiving a payout in the form of a scoring rule, as long as this predictor is willing to pay off the previous predictor according to the same scoring rule.

So far, this may not seem to have anything to do with prediction markets as classically defined. Hanson's second clever insight, however, was that for each market scoring rule, an automated market maker can be constructed that in effect implements the scoring rule. Based on the number of shares for each contingency being forecast, a formula can be used to calculate a price, which can be understood as a forecast if the scoring rule is strictly proper. The automated market maker will sell an infinitesimal share at this price. Each infinitesimal purchase raises the price, but with calculus, it is possible to derive formulas indicating for any given amount of money, how many shares can be purchased, and what the new price of the share will be afterward.

The underlying dynamics are precisely the same as with the market scoring rule in the absence of a market. When all is said and done, a trader's profit in a market using this automated market maker will be exactly the same as if the trader received payment under the market scoring rule based on the new price and in turn paid off the prior trader based on the old price. The correspondence, however, may not be intuitively obvious to traders. The amount that a user has to pay up front is not the prior trader's payoff under the market scoring rule – that amount will not be determined until the market closes. Instead, it is a price equal to the amount that the user will lose if the event corresponding to the shares purchased does not occur. The amount that the user eventually receives if the event occurs is not the payoff under the scoring rule, but, as in classically structured prediction markets, some fixed amount (such as \$1 per share).

These insights alone would have provided for a major improvement in prediction market technology from Hanson, the inventor of the classical prediction markets trading structure. But Hanson had another critical insight, that the market scoring rule could be used as the basis for a "combinatorial" market. In a combinatorial market, participants can trade not only on whether any particular event will occur, but also on whether a particular event will occur contingent on another event. The number of possible contingencies that individuals might like to trade on increases exponentially in the number of events, and past some point may become computationally unmanageable with current computers, but Hanson showed at least how an automated market maker could manage trading on all possible combinations for a relatively small number of events.

Ideally, in a combinatorial market, trading on the probability of *A* given *B* shouldn't result in a change in the probability of *B*, or a change in the probability of *C* given *A*. Hanson proved that this independence could be achieved with only one form of the market scoring rule, the logarithmic market scoring rule. Given this observation, it is not surprising that the logarithmic market scoring rule has been so widely adopted. Hanson provided a formula for the price function for the logarithmic market scoring rule, and Pennock (2006) has published the cost function as well. Meanwhile, Pennock concluded that the market scoring rule corresponding to the quadratic scoring rule "is not very interesting or useful in practice."

Apparently, Pennock came to this conclusion because he calculated a cost function for the quadratic market scoring rule based on a single market that could handle any number of contingencies.¹ Such an approach works well with the logarithmic market scoring rule. With this approach applied to the quadratic market scoring rule, however, as soon as one security reaches its upper limit, the market maker cannot allow purchases of other securities to drive down the other prices, because otherwise the probabilities corresponding to the various securities would add up to more than 1. As we will see, however, there is a simple solution to this problem. Where there are two possible events, the automated market maker can sell two different types of contracts implementing opposite quadratic market scoring rules to derive a single market price. For more than two events, additional markets can be deployed, with minimal additional computational complexity, and a subsidy intended for the predictive task can be divided among them.

But it may seem that there is relatively little reason to use a quadratic market scoring rule when the logarithmic market scoring rule already exists. In fact, however, the logarithmic scoring rule has a serious disadvantage for many applications. The amount of subsidy that a user receives for making a correction to a probability assessment varies dramatically across the probability spectrum, particularly near the ends of the spectrum. It can be shown, for example, that the amount of money that a user can expect from correcting a prediction from 0.10 to 0.11 (assuming that 0.11 is in fact correct) is only 13.8% as much as from correcting 0.01 to 0.02 (assuming that 0.02 is in fact correct).² Equivalently, with the automated market maker, a user will purchase far more shares in moving the price from 0.01 to 0.02 than in moving the price from 0.10 to 0.11. Because the logarithmic scoring rule provides infinite liquidity near the ends of the probability spectrum (although not infinite profit potential), trading can never move a probability all the way to 0 or 1, even if the event being predicted has already occurred (but the market administrator has not terminated the market).

There may be situations in which it makes sense to have very high subsidies for trading near zero. Suppose, for

¹ This is explained in an unpublished paper by Pennock, Yiling Chen, and Mike Dooley.

² This 13.8% figure can be calculated as follows:
$$\frac{((.11*\ln(.11)+(1-.11)*\ln(1-.11)) - (.11*\ln(.10)+(1-.11)*\ln(1-.10)))}{((.02*\ln(.02)+(1-.02)*\ln(1-.02)) - (.02*\ln(.01)+(1-.02)*\ln(1-.01)))}$$

example, that one were using a prediction market to assess the chance of a relatively low probability catastrophic, event. It might make a big difference for policy response whether the event has 0.001 or 0.002 probability, and it might be sensible to provide as great a subsidy for correcting a prediction from one level to the other as from, say, 0.01 to 0.02.

But there are many prediction markets for which this is not the case – it doesn't much matter just how outrageously low the Boston Celtics' chance of winning the NBA Championship are right now. Yet, using the logarithmic scoring rule to predict the NBA Champion would provide risk-neutral participants a relatively large incentive to work out those small numbers. The point is most obvious for prediction markets that are merely forecasting point estimates. (It is possible to use the logarithmic scoring rule to motivate forecasters to estimate an entire probability distribution, but for some applications, all the market sponsor really cares about is the mean of that distribution, and requiring each forecaster to enter an entire probability distribution is overkill.) Perhaps we would like to use a prediction market to forecast simply how many games the Celtics will win this year. In that case, we would likely want the same subsidy for each per game marginal improvement in the forecast.

Even where it may make sense to devote a relatively large portion of a market subsidy to the bottom of the probability spectrum, it might be desirable to allow the market administrator to specify just how the subsidy should be for each part of the probability spectrum. Suppose, for example, that Corporation *A* creates a prediction market to determine whether Corporation *B* will release a product competing with an offering from Corporation *A*. It may be that it will be worth it for Corporation *A* to respond (e.g., by accelerating its own project) if the probability is greater than 0.10, but that Corporation *A* will not respond at all for lower probabilities and will not augment its response for much higher probabilities. In that case, it makes sense to have the most liquidity in the direct vicinity of the 0.10 probability, not on the ends of the probability spectrum.

II. A UNIFORM LIQUIDITY AUTOMATED MARKET MAKER

This section describes a different automated market maker, one that provides uniform liquidity across the probability spectrum. Section II.A derives formulas for this market maker by conceptualizing this market maker as one that sells two kinds of shares, “high” and “low,” and places an “ask” offer for one kind or the other at every price along the probability spectrum. Section II.B derives formulas for the quadratic market scoring rule, and shows that it is equivalent. Some readers will merely be interested in the formulas provided in Table 1 below, but I provide the two derivations to explain how I obtained them and to emphasize the equivalence of the two approaches.³

A. The Uniform Automated Offers Approach

³ I am grateful to Robin Hanson for pointing out, when I described the uniform liquidity automated market maker, that there must be some version of the market scoring rule to which it would be equivalent.

Suppose that we are making prediction of an event that will produce a number p_{final} , which may range from p_{min} to p_{max} . (In the event of a binary prediction, $p_{final} = 0$ if the event does not occur and $p_{final} = 1$ if it does, so $p_{min} = 0$ and $p_{max} = 1$.) Let the current prediction be equal to $p_{current}$ (initially set to some default value $p_{default}$), and at all times, let $p_{current}^* = (p_{current} - p_{min}) / (p_{max} - p_{min})$, and let $p_{final}^* = (p_{final} - p_{min}) / (p_{max} - p_{min})$. The market maker is available at any time to sell at every price an infinitesimal number of shares whose maximum possible value is v . For prices above $p_{current}^*$, the market maker will sell an infinitesimal share for $v * p_{current}^*$ that will pay off $v * p_{final}^*$, and for prices below $p_{current}^*$, the market maker will sell an infinitesimal share for $v * (1 - p_{current}^*)$ that will pay off $v * (1 - p_{final}^*)$.

Let d equal the total number of shares that a user would need to purchase to move the price from p_{min} to p_{max} . Now suppose that a participant wishes to change the price from $p_{current}$ to p_{new} , where $p_{new} > p_{current}$. We let $p_{new}^* = (p_{new} - p_{min}) / (p_{max} - p_{min})$. Because the automated market maker's ask offers are uniformly distributed, the average price of a share in changing the prediction will be $v * (p_{current}^* + p_{new}^*) / 2$. The number of shares purchased $q = d * (p_{new}^* - p_{current}^*)$. So, the total cost $c = (v * (p_{current}^* + p_{new}^*) / 2) * d * (p_{new}^* - p_{current}^*) = (p_{new}^*^2 - p_{current}^*^2) * dv / 2$.

We can now calculate the maximum subsidy s , which equals the maximum gain for traders if the market were to start at a default value of p_{min} , and trading closed at $p_{max} = p_{final}$, or vice versa. If $p_{current}^* = 0$ and $p_{new}^* = p_{final}^* = 1$, then $c = dv / 2$, with a payoff w (for “winnings”) of dv , and thus the subsidy $s = dv - dv / 2 = dv / 2$.

We continue to assume that a participant wishes to change the price from $p_{current}$ to p_{new} , where $p_{new} > p_{current}$. Substituting into the equations above, $c = s(p_{new}^*^2 - p_{current}^*^2)$, and the ultimate payout w equals $2s * p_{final}^* * (p_{new}^* - p_{current}^*)$. Profit $\pi = w - c = s * (2 * p_{final}^* * (p_{new}^* - p_{current}^*) - (p_{new}^*^2 - p_{current}^*^2))$. It can similarly be shown that if a participant wishes to change the price from $p_{current}$ to p_{new} , where $p_{new} < p_{current}$, $c = s * ((1 - p_{current}^*)^2 - (1 - p_{new}^*)^2)$, and $w = 2s * (1 - p_{final}^*) * (p_{current}^* - p_{new}^*)$. Note that these equations do not include d or v ; simply specifying a maximum subsidy s is sufficient to calculate all variables that are needed to determine costs and payoffs.

The relevant formulas are summarized in Table 1.

Table 1. Formulas for Implementing the Uniform Liquidity Automated Market Maker

Initial state variables			
p_{min}	The minimum possible outcome	Given	
p_{max}	The maximum possible outcome	Given	
s	The maximum possible subsidy	Given	
v	The maximum possible payoff of a share	Given	
$p_{default}$	The default prediction	Given	
$p_{current}$	The current prediction	At beginning of market:	

		$p_{default}$
		After new prediction is processed: p_{new}
d	The density, or the total number of shares that a forecaster would need to purchase to move the prediction from p_{min} to p_{max}	$2s/v$
$p_{current*}$	The scaled current prediction	$(p_{current} - p_{min}) / (p_{max} - p_{min})$

Variables realized after occurrence of event

p_{final}	The final result of the event (for a binary event, 0 if the event does not occur and 1 if it does occur)	Determined by the ultimate event, or by the market sponsor as a result of the early termination of the market
p_{final*}	The scaled final result	$(p_{final} - p_{min}) / (p_{max} - p_{min})$

If predictor wishes to set a new prediction higher than the current prediction

p_{new}	The new prediction; must confirm that $p_{min} \leq p_{new*} \leq p_{max}$.	Given by predictor
p_{new*}	The scaled new prediction	$(p_{new} - p_{min}) / (p_{max} - p_{min})$
c	The cost to the predictor (or, equivalently, the maximum loss that the predictor faces)	$s(p_{new*}^2 - p_{current*}^2)$
q	The number of "high" shares received by the predictor	$(2s/v) * (p_{new*} - p_{current*})$
w	The predictor's gross winnings, i.e. the payoff that the predictor receives after occurrence of the event	$2s * p_{final*} * (p_{new*} - p_{current*})$.
π	The predictor's net winnings or profit	$w - c$

If predictor wishes to set a new prediction lower than the current prediction

p_{new}	The new prediction; must confirm that $p_{min} \leq p_{new*} \leq p_{max}$.	Given by predictor
p_{new*}	The scaled new prediction	$(p_{new} - p_{min}) / (p_{max} - p_{min})$
c	The cost to the predictor (or, equivalently, the maximum loss that the predictor faces)	$s((1 - p_{new*})^2 - (1 - p_{current*})^2)$
q	The number of "low" shares received by the predictor	$(2s/v) * (p_{current*} - p_{new*})$
w	The predictor's gross winnings, i.e. the payoff that the predictor receives after occurrence of the event	$2s * (1 - p_{final*}) * (p_{current*} - p_{new*})$.
π	The predictor's net winnings or profit	$w - c$

If predictor wishes to purchase "high" shares for a given cost

c	The cost to the predictor (or, equivalently, the maximum loss that the predictor faces)	Given by predictor
p_{new*}	The scaled new prediction; must confirm that $p_{min} \leq p_{new*} \leq p_{max}$.	$\text{sqrt}(p_{current*}^2 + (c/s))$
p_{new}	The new prediction (from which other variables can be	$p_{min} + p_{new*} * (p_{max} - p_{min})$

calculated, as above)

If predictor wishes to purchase "low" shares for a given cost

c	The cost to the predictor (or, equivalently, the maximum loss that the predictor faces)	Given by the predictor
p_{new*}	The scaled new prediction; must confirm that $p_{min} \leq p_{new*} \leq p_{max}$.	$1 - \text{sqrt}((1 - p_{current*})^2 + (c/s))$
p_{new}	The new prediction (from which other variables can be calculated, as above)	$p_{min} + p_{new*} * (p_{max} - p_{min})$

If predictor wishes to purchase a particular number of "high" shares

q	The number of "high" shares purchased by the predictor	Given by predictor
p_{new*}	The scaled new prediction; must confirm that $p_{min} \leq p_{new*} \leq p_{max}$.	$p_{current*} + q/d$
p_{new}	The new prediction (from which other variables can be calculated, as above)	$p_{min} + p_{new*} * (p_{max} - p_{min})$

If predictor wishes to purchase a particular number of "low" shares

q	The number of "low" shares purchased by the predictor	Given by the predictor
p_{new*}	The scaled new prediction; must confirm that $p_{min} \leq p_{new*} \leq p_{max}$.	$p_{current*} - q/d$
p_{new}	The new prediction (from which other variables can be calculated, as above)	$p_{min} + p_{new*} * (p_{max} - p_{min})$

B. The Quadratic Market Scoring Rule Approach

It is straightforward to derive the same formulas using a quadratic market scoring rule. The quadratic scoring rule is defined as follows (Murphy and Winkler 1970): Assume that the forecaster announces a probability distribution $p=(p_1, \dots, p_n)$, i.e. announces p_i , $1 \leq i \leq n$, for each of n events, where $\sum p_i = 1$. If the event that occurs is j , the forecaster's payoff $Q_j(p) = a + 2b * p_j - b * \sum(p_i)^2$.

Let $n = 2$ and let p_1 correspond to the probability that the event occurs. Then, if the event does occur, the forecaster receives payout $Q_1(p) = a + 2b * p_1 - b * (4p_1 - 2p_1^2 - 1)$, and if the event does not occur, the payout is $Q_2(p) = a + b(1 - 2p_1^2)$. Note that a and b represent arbitrary constants.

Assume as before that the prior forecaster has announced $p_1 = p_{current*}$, and the new forecaster has announced $p_1 = p_{new*}$, where $p_{new*} > p_{current*}$. The cost for the new forecaster is the maximum possible loss, which is the loss in the event that the event does not occur. Thus, $c = Q_2(p_{current*}) - Q_2(p_{new*}) = 2b(p_{new*}^2 - p_{current*}^2)$. Similarly, the profit for the new forecaster is the gain if the event does occur, where $p_{new*} > p_{current*}$. It follows that $\pi = Q_1(p_{new*}) - Q_1(p_{current*}) = 2b * (2(p_{new*} - p_{current*}) - (p_{new*}^2 - p_{current*}^2))$.

It is apparent that this is a special case of the formulas above, where $p_{final*} = 1$ and $s = 2b$. It is straightforward to show a similar equivalence where $p_{new*} < p_{current*}$.

III. EXTENSIONS

This part explains how to extend the uniform liquidity market maker. Section III.A explains how the subsidy can be varied over the probability or prediction spectrum, as well as over time. Section III.B, meanwhile, explains how the uniform liquidity market maker can be used when making probabilistic predictions and there are more than two mutually exclusive outcomes. Section III.C explains how the market maker can be run over multiple rounds to create a “deliberative market,” and Section III.D shows how the deliberative market design makes it straightforward to implement conditional markets.

A. Variable Subsidy

As noted above, the uniform liquidity automated market maker provides an equal incentive for a marginal prediction improvement anywhere along the probability or prediction spectrum. For many applications, this will be what is desired, but there may be some situations when it is not. Suppose, for example, that a software company is deciding whether to create a new version of its software program, and that decision will depend in part on the sales of the current version of the software program. If the expected sales of the current version are below some low threshold, it definitely will not be worth creating a new version, and if the expected sales are above some high threshold, it definitely will be worth creating a new version. If the company is creating a prediction market to forecast sales of the current version, and needs the result only because it will help determine whether to make a new version of the software, it might not make much sense to provide a large subsidy below the low threshold or above the high threshold. Once it is clear that the actual number will not be between the thresholds, the precise value does not much matter.

Meanwhile, it also often does not make sense to provide the same subsidy over time. In particular, it will often be sensible to provide a lower subsidy at the very beginning of the market, and to raise this subsidy to the maximum level afterward. The reason for this is that there may be a number of predictors who can provide a rough approximation of a prediction very cheaply, and so a small subsidy will be enough to induce some predictor to make this approximation if it is thought that otherwise some other predictor will do so. The market sponsor, however, ultimately may want greater accuracy eventually, and so the subsidy can rise, inducing greater levels of research by predictors. (Of course, a predictor might do research in advance but then wait to trade on it until the subsidy is sufficiently high.) In many contexts, the subsidy can be set to reach its maximum level quite quickly.

One virtue of the dynamic pari-mutuel market is that the profit from betting will generally rise over time in this way. The greater the number of bets that have been placed in the dynamic pari-mutuel market, the greater the liquidity. But if a dynamic pari-mutuel market is subsidized by the placement of “seed wagers” at the beginning of the market, much of the subsidy will effectively be distributed at the start of the market. At the least, it is not easy to figure out how the dynamic pari-mutuel market can be modified so that a given level of subsidy

can be provided at a particular time, let alone for different portions of the probability spectrum.

Table 2 shows how to implement a general automated market maker on top of the uniform automated market maker. It introduces various new variables representing the relative subsidy to be given over the probability spectrum and over time. In contrast to Table 1, the maximum possible subsidy is now represented by the variable σ , and the variable s now represents an adjusted subsidy variable. Once calculated, this s variable can be used wherever s appears in Table 1 to calculate various other variables based on a new prediction specified by the predictor. Unfortunately, there is no simple formula for determining the new prediction based on a cost or number of shares input by a predictor. If this is desired, however, it can be straightforwardly calculated by creating a two-dimensional array of possible new predictions and the corresponding costs.

Table 2. Formulas for a General Automated Market Maker

New variables and the calculation of the adjusted subsidy measure		
$f(p)$	The subsidy weight for a marginal change to a given prediction p	Given for all p between p_{min} and p_{max} , where $0 <= f(p)$ for all p .
$F(p)$	The cumulative density function for $f(p)$	Can be calculated from $f(p)$. Note that $F(p_{min}) = 0$, and where $p_{min} + \epsilon <= p <= p_{max}$, $F(p) \approx F(p-\epsilon) + \epsilon f(p)$.
$f_{mean}(a,b)$	The average value of $f(p)$ from a to b	$(F(b) - F(a))/(b-a)$
$g(t)$	The subsidy weight at time t , where $0 <= g(t) <= 1$ for all t .	Given
t	The time at which the current transaction takes place	Given
σ	The maximum possible subsidy	Given
s	Adjusted subsidy variable	$\sigma * g(t) * f_{mean}(p_{current}, p_{new}) / f_{mean}(p_{min}, p_{max})$

There may be some circumstances in which it makes sense to change the relative subsidy levels based on variables such as the current price or the current time. Consider, for example, a prediction market being used to predict the outcome of a sports contest. The sponsor of the market might want new predictions to be announced during the event, but might not want to offer larger subsidies closer to the current prediction level. Otherwise, there will be a particularly large award for whichever predictor happens to respond to a major event in the contest by entering a new prediction the fastest. This modification should be straightforward to implement by changing the subsidy distribution $f(p)$ over time. Note that the distribution should be centered around the price with a short time lag, lest users be able to capture the full subsidy by rapidly entering numerous incremental predictions. This approach limits the ability of any market participant to receive a large subsidy simply for acting first on news available to all participants.

B. More Than Two Mutually Exclusive Events

The above analysis makes it straightforward to forecast either a continuous variable or a binary variable, that is to predict whether a single event A_1 will or will not occur. If either A_1 will occur or A_2 will occur, then the approach above can be used simply to forecast A_1 , since $A_2 = 1 - A_1$. Equivalently, however, one could run two markets, allowing the automated market maker to sell “low” and “high” shares in both A_1 and A_2 . A user could maximize profit from a given correction to the relevant probabilities by trading on both markets, and arbitrageurs should generally ensure that the market prices add up to unity. A simpler strategy, however, is to have any trading in A_1 or A_2 automatically result in trading in A_1 and A_2 . So, for example, when a user requests to make a purchase that will change the prediction of A_1 from 0.30 to 0.31, the system reports the cost of changing the prediction of A_1 from 0.30 to 0.31 (by buying “high” shares in A_1), plus the cost of changing the prediction of A_2 from 0.70 to 0.69 (by buying “low” shares in A_2). The system would, of course, report the exact same price if the user requests to make a purchase that will change the prediction of A_2 from 0.70 to 0.69. The subsidy for each of these two markets could be set to half of the original intended maximum subsidy.

Of course, this modest added complexity has little value when there are only two mutually exclusive events to be forecast, but it can be quite valuable when there are three or more. From a user interface perspective, the user can enter new probabilities for each (or for all but one) event, or alternatively, the user can simply change one probability, and the system can calculate the probabilities to which the user is setting the other events. A simple approach is by default to change the other predictions proportionately so that all of the predictions continue to add up to 1, and to report the price for conducting all of the transactions simultaneously. (A useful user interface feature might allow the user to move sliders for any one event, with the sliders for all other events moving responsively.) So, let $p_{current}^*(i)$ and $p_{new}^*(i)$ represent the current and new predictions for some event i , $1 \leq i \leq n$, where n is the total number of mutually exclusive events. Then, for each j , $1 \leq j \leq n$, where $i \neq j$, $p_{new}^*(j) = p_{current}^*(j) + (p_{current}^*(i) - p_{new}^*(i)) * p_{current}^*(j) / (1 - p_{current}^*(i))$.

Two technical details: First, if the maximum total subsidy is t , then the maximum subsidy σ corresponding to the market for each mutually exclusive possibility should be set to $t/2$. This is true regardless of the number of mutually exclusive events. Second, there is no simple formula for determining the new prediction based on a cost or number of shares input by a predictor. Once again, however, if this is desired, it can be straightforwardly calculated by creating a two-dimensional array of possible new predictions and the corresponding costs, and interpolating between them.

C. Deliberative Markets

Abramowicz (2006) described the operation of a deliberative prediction market. In such a market, a participant’s performance depends not just on ultimately being proved right

by the event being predicted, but on actually succeeding in moving the market along some potentially smaller interval. To succeed in such a market, a trader must convince other market participants that one is correct.

This approach is straightforward to implement in conjunction with the market design described here. The system must pay off shares held by predictors at some time after the prediction is made, but potentially before the event occurs. The exact amount of time is randomized to prevent manipulation. Suppose that the prediction is made at time t . Then, the system calculates a reimbursement time of $t_{final} = t_{min} - h * \ln(\text{rand}()) / \ln(2)$, where t_{min} is a minimum amount of time, h is the half life after this minimum is reached, $\ln()$ is the natural logarithm function, and $\text{rand}()$ is a pseudo-random number generator producing a floating point number uniformly distributed between 0 and 1. At time t_{final} , the shares for the predictor are paid off by setting p_{final} for that predictor only to $p_{current}$. Of course, if the event actually occurs, all remaining shares are paid off based on p_{final} .

Two technical points: First, the subsidy s in Table 1 is no longer the maximum theoretically possible subsidy. Second, h should be set to an amount of time long enough so that if someone did try to manipulate the market, others with high probability would successfully counter such manipulation. It may often make sense for h to fall closer to the conclusion of an event. For example, more traders are likely to be sitting by their computers looking at market prices during the fourth quarter of a basketball game being forecast than hours before, and so h might be as short as, say, a minute.

The deliberative market approach may work effectively even where producing deliberation is not a goal of the market. One problem with traditional prediction market designs is that predictors’ money is unavailable when a market is running. This is particularly problematic where a market may not resolve itself for many years. Although there are some possible solutions—such as investing money in a mutual fund—making t_{min} and h relatively short is a simple one.

Note that market participants will still have incentives to forecast the ultimate event because they will recognize that predictors just before the event occurs will have such incentives, and that predictors just before that will have an incentive to anticipate the predictions of later predictors, and so forth. This may seem attenuated, but it is in fact no different from many security markets where the payoff will not occur for a long time. Even if each individual trader expects to be in the market only a short period of time, the market fundamentals depend on the eventual event.

D. Conditional Markets

The most significant disadvantage of the quadratic market scoring rule, as well as of the variations described here, is that it is not well suited to combinatorial markets. With a combinatorial market, a single subsidy can be allocated automatically to subsidize all possible combinations of events. But conditional markets can still be used to make discrete conditional probability forecasts. Suppose, for example, that prediction markets are used to forecast whether each of 12 battleground states will vote Democratic or Republican in the

next Presidential election. A combinatorial market would effectively allow trading on all permutations and combinations, including for example the probability that Florida will go Democratic if Ohio and New Mexico go Republican, and it would ensure that the market results are consistent. Conditional prediction markets, meanwhile, can be used to make any particular conditional evaluation specified by the market sponsor, but there is no mechanism for providing automatic consistency among large number of markets. Such conditional markets will satisfy the needs of many, though perhaps not all, potential market sponsors.

One way to implement conditional markets is to simply run markets on different possibilities. For example, one might create a market on whether the Republican candidate will win both the Presidency and Florida, as well as a market on whether the Republican will win Florida. The probability of the first divided by the probability of the second provides an estimate of the probability that the Republican will win the Presidency contingent on the Republican winning Florida. Mathematically, this approach is straightforward, and the user interface could present the user with the derived conditional prediction.

A predictor could correct misaligned conditional probabilities by placing bets on both markets. For example, suppose that a predictor believes that the market is underestimating the probability that the Republican will win the Presidency contingent on the Republican winning Florida. Suppose, though, that the predictor has no idea which of the two markets is misaligned. Then, the predictor could purchase shares in the “both” security, and sell short shares in the Florida security. If done in just the appropriate proportions, the predictor will end up with no gains or losses if in fact the Republican does not win in Florida. So, the predictor is truly making a conditional assessment. A drawback, however, is that the predictor will have tied up money for potentially the full duration of the market, and that may be undesirable, especially if the condition is unlikely.

An alternative way to implement conditional markets with the quadratic market scoring rule and other automated market makers is to unwind a market if the condition is not met. For example, a single prediction market can be maintained that forecasts whether the Republican will win the Presidency if Florida votes Republican. If Florida does not vote Republican, any money charged for shares is refunded, so traders will assume that Florida in fact will vote Republican for the purposes of this market. This approach, however, has the same drawback as before, that individuals will often not want to tie up their money for long periods of time, especially for unlikely contingencies.

The deliberative market structure described above, however, largely overcomes this problem. As soon as it becomes apparent that the contingency will not be met, the market unwinds only those predictions that have not yet produced a payoff. If round times are sufficiently brief, traders can generally profit on predictions even when the contingency is not ultimately met. For example, if the minimum round time is a week and the half life is equal to a few hours, then a conditional prediction made in October will with very high probability be resolved by the election.

The other advantage of this approach is consistent with the benefit of the general market maker discussed in this paper, that it makes it possible to target subsidies to the questions of greatest interest. A market sponsor might sometimes be interested in the probability of A conditional on B , but not otherwise interested in the probability of B . Creating markets in $A \& B$ and B effectively allocates a portion of the subsidy to users who can improve the market’s information on whether B will occur. By using the deliberative market structure and unwinding only once it becomes clear that a condition will not occur, the market sponsor is providing financial incentive only for research on A conditional on B . Of course, the market sponsor might be interested in subsidizing a market on B as well, but in that case can determine how great a subsidy to allocate to provide incentives for research on the probability of B .

IV. CONCLUSION

No experimentation is required to demonstrate that the quadratic market scoring rule should give market participants incentives to move prices toward the values they perceive as correct. Risk-averse participants might not move prices all the way to the values that they perceive as correct given all information (Sonnemans and Offerman 2001), but at least should move prices in the correct direction. Ideally, however, experimentation should be done to assess the degree to which the quadratic market scoring rule may improve on alternatives in providing greater subsidies for particular parts of the probability spectrum. It seems quite plausible that in lab experiments, many automated market makers will perform comparably, in part because it may take participants a long time to figure out how to optimize their research and prediction activities. The quadratic market scoring rule’s effects should be greatest in markets with very large subsidies, where participants will have sufficient incentives to tend to these details, but to date prediction market subsidies have been relatively small.

Bibliography

- M. Abramowicz (1999). The Law-and-Markets Movement. *American University Law Review* 49: 327.
- M. Abramowicz (2006). Deliberative Information Markets for Small Groups. *Information Markets: A New Way of Making Decisions* (AEI Press): 101.
- M.H. DeGroot & S.E. Fienberg (1983). The Comparison and Evaluation of Forecasters. *Statistician* 32: 12.
- R. Hanson (2003). Combinatorial Information Market Design. *Information Systems Frontiers* 5: 107.
- R. Hanson (2006). Logarithmic market scoring rules for modular combinatorial information aggregation. *Journal of Prediction Markets* 1: 1.
- K.T. Maxwell & M.R. Burns (1996). Computer-Implemented Securities Trading System with a Virtual Specialist Function. U.S. Patent No. 5,950,176 (issued Sept. 7, 1999).

A.H. Murphy & R.L. Winkler (1970). Scoring Rules in Probability Assessment and Evaluation. *Acta Psychologica* 34: 273-286.

D.M. Pennock (2004). A Dynamic Pari-Mutuel Market for Hedging, Wagering, and Information Aggregation. in *Proceedings of the 5th ACM Conference on Electronic Commerce* 170.

D.M. Pennock (2006). Implementing Hanson's Market Maker. <http://blog.oddhead.com/2006/10/30/implementing-hansons-market-maker/>.

R.J. Shiller (2003). From Efficient Markets Theory to Behavioral Finance. *Journal of Economic Perspectives* 17: 83.

J. Sonnemans and T. Offerman (2001). Is the Quadratic Scoring Rule Really Incentive Compatible? <http://www1.fee.uva.nl/creed/pdf/files/qspaper5.pdf>.