September, 2025

Jethro Update

Hello, everyone. It has been quite a while since my last update. I have been busy. I believe Jethro was already quite a good bidding engine before I started on this round of changes. I have made substantial improvements; it is better now.

This update is a synopsis of what I've done since January, 2025. There is likely far, far more information here than you care about. It gets geeky in places. As much as anything, I wanted to have a place to collect my notes and thoughts of the last 8 months. I don't really anticipate writing a book about Jethro, but in the unlikely event I decide to do so, I don't want to forget all this. I hope you find it at least a little bit interesting. Thanks for indulging me. I won't be offended if you don't make it to the end.

When I disappeared in January, I said I wanted to do a deep dive on Jethro's "evaluator" the part of the code that calculates how strong a given hand is. There are two parts, the initial evaluation – before any bidding occurs – and the dynamic evaluation, which is updated before every bid, incorporating information learned during the auction. I made substantial changes to both evaluators. Beyond the evaluators, I also made countless changes to the bidding algorithms, sometimes rewriting entire sections of code, or adding new code to handle specific sequences which had previously been handled with generic code.

I cloned the old code, creating a copy, then started making changes to the existing code base. The new version has about 50,000 lines of code, which, to be sure, is largely cut and pasted from the old code. But still. It's a lot.

The Bottom Line

I run matches of Jethro vs. Jethro, where one team uses the new version, and the other team uses the old version. My standard test file has 1,000,000 hands. It takes about 35 minutes to run a match of that size. The score on each hand is calculated using the double dummy result of the contract reached.

Here is the bottom line, from a 1,000,000 board match I ran recently.

TeamA is using the new version of code; TeamB is using the old version. “Bam” is the board-a-match (or matchpoint) score.

```
              TeamA    TeamB    Net
Total Imps: 818825 - 637672 181153 (778577 tied boards)
Total Bam:  129544 - 101522  28022 (768934 tied boards)

There were different auctions on 36.642% of the boards.
```

Not every different auction led to a different result. Sometimes different auctions led to the same contract. Sometimes different auctions led different contracts, but those different contracts scored identically.

As you can see, on average the new code gained one IMP every 5.5 boards vs. the old code. That seems like a lot, at least to me.

Looked at from yet another angle, the BAM score was (scaling down the above numbers by a factor of 10,000, and rounded) 77 ties, 13 wins for TeamA, 10 wins for TeamB. Which feels like a much narrower margin, but, in reality, is just as substantial.

Some Thoughts About Methodology

It is not always clear cut to determine if a particular change is for the best. Some changes led to an increase in the IMP score, while simultaneously decreasing the BAM score, or vice versa. You don’t make the games or slams you fail to bid, but bidding to higher contracts inevitably leads to going down more often. The converse, of course, is also true – bidding less leads to more made contracts, but you miss some making games and slams, losing substantial numbers of IMPs.

Here is an example from a 100,000 board match I ran early on, to test some long forgotten change. The new code lost IMPs, but gained on BAM.

```
             TeamA   TeamB  Net
Total Imps: 16879 - 17414 -535
Total Bam:   2750 -  2448  302 (94802 tied boards)
```

My first thought was that the scoring method should determine which version is better. If playing IMPs, you would prefer to be TeamB; if BAM, you would prefer to be TeamA.

But, this being bridge, nothing is ever straightforward. There is another factor which cannot be ignored. The length of the match makes a big difference. I took the *identical* 100,000 board match and sliced it into 100 consecutive matches of 1000 boards each. To be clear, *I did not run any new test*, I just repackaged the results I already had.

1000 Board Matches (100 total matches)

	TeamA Wins	TeamB Wins	Ties
Imps	47	53	0
Bam	67	31	2

Now the same data repackaged into 32 board matches:

32 Board Matches (3125 total matches)

	TeamA Wins	TeamB Wins	Ties
Imps	1262	1216	647
Bam	1126	945	1054

With shorter matches (32 boards or less, as it turned out in this example) TeamA actually prevailed at IMPs more often, despite *losing* IMPs over the entire 100,000 boards! The point is that even at IMP scoring, in a shorter match you would prefer to be TeamA, even though the overall IMP score was lower. In shorter matches the BAM score is more important, and needs to be weighted accordingly.

Regardless, I am not prepared to try to maintain separate code to maximize for IMPs vs BAM. Maybe some day I will add this as an option. But for now, I really want a single code base that works reasonably well for both. To that end, I created a simple algorithm. I compute a score, using both the IMP margin and the BAM margin, scaling the BAM margin by some magic scale factor. The scale factor I settled on was 4.5. That choice was fairly arbitrary; I suspect a better value for the scale factor might be a bit higher.

Using the above example, from the 100,000 board match:

Score = -535 + 302*4.5 = 824

Since the score is > 0, I consider it a good change. I am not entirely dogmatic about the scores. If they feel "too far out of whack" (that's a scientific term) in one direction or another I will generally go back to the drawing board, and try to find a similar change that will get both IMPs and BAM moving in the positive direction. Even that is subject to second guessing. If, for example, you gain 1000 IMPs, but the BAM score barely changes, you can be sure there are a lot of other hands going down, since the hands where IMPs are gained are also hands where BAM is gained.

Regarding the size of the test file. I have created a lot of random hand test files of different sizes. For many of the changes I've made recently, only a very tiny percentage of hands will be bid differently as a result of one particular change. In many cases if one hand in a thousand is bid differently, that is a lot. In a perfect world, I would create many more test files each focused on those particular hands, but that would be an enormous thankless task. The 1,000,000 random board test file I settled on was one that appeared, as best as I could discern, to be the most representative of many 1,000,000 board test files I created. I also created a ton of 100,000 board test files. I picked one I thought correlated well with the bigger file. I typically use the smaller file as a quick test to see if I think a change is likely to be successful. Unless it appears to be a clearly bad idea, I then run the test on the bigger file. Even so, it is not at all uncommon to see significant differences between the small test and the big test.

What the "best" size for a test file might be, I couldn't tell you, but I am sure 100,000 is not enough to decide for most minor changes. I do know this much: At one point I ran the same code, testing 9 different 1,000,000 board files. The IMP difference between the highest and lowest scoring test was about 16,000 IMPs, which was approximately 10% of the total IMP difference.

I spent a lot of time and energy working on the above methodology. Way more than I anticipated. It seems like a simple question to decide whether or not a

particular change is for the best. But like virtually everything else associated with this project, it just isn't so.

Changes to the Initial Evaluation

First, some background, and a recap of where I started in January.

Everybody is familiar with the traditional method of counting High Card Points. A=4, K=3, Q=2, J=1, T=0. It's really easy, and universally known. It is likely the first thing anyone remembers they learned about hand evaluation. It's also not entirely accurate. Two Queens really are not as good as one Ace, when it comes to taking tricks, which is, after all, the goal of the game. And it's just possible that Tens have some value.

Back when I started writing Jethro, I found a reference to an HCP evaluation called "BUMRAP." I am told BUMRAP stands for "Burnstine Uchida Martelli Reality Adjusted Pointcount."

BUMRAP evaluates high cards as follows: A=4.5, K=3, Q=1.5, J=.75, T=.25; Aces and Tens are upgraded, Queens and Jacks downgraded. The entire deck still has a total of 40 HCP.

The claim is that BUMRAP evaluations do a better job than the traditional HCP count, and, in fact, my testing has definitely confirmed that hypothesis. I have long used BUMRAP in my programs, internally using the somewhat pretentious name "HCP Accurate." I don't completely throw away the traditional HCP – sometimes it just seems too jarring to do so – but I rely on it far less.

Another correspondent, Nigel Kearney, suggested a slightly different HCP weighting, which puts even more emphasis on Tens and Jacks: A=4.4, K=2.8, Q=1.6, J=0.8, T=0.4. In honor of Nigel, I dubbed this the "Kearney Count." In tests I have run, this weighting does, in fact, seem better for BAM (matchpoint) scoring, but notably worse for IMP scoring. For now, I am keeping BUMRAP as my sole HCP evaluator. But it I ever get around to adding an option for "Scoring Method" I will definitely be revisiting the Kearney Count for BAM scoring.

Counting HCP, of course, is just one step in the initial hand evaluation. My original evaluator calculated three separate metrics: (1) Bergen Points, as defined by Marty Bergen, (2) the 4C's method (AKA Kaplan-Rubens) from an article in *The Bridge World* (October, 1982), and (3) a home grown method of my own creation, using BUMRAP HCP and short suit/long suit combinations. My final calculation was a weighted average of those three metrics, which turned into a number I called "Base Working Points." For contemplating NT contracts I also calculated a number I called "Base NT Working Points." I will spare you the details on how those values differed.

With my new iteration of the evaluator, I have abandoned those three metrics in favor of a new method. I found a reference for something called "Binky Points" along with a lot of data and an email link for the creator of Binky Points, a gentleman named Thomas Andrews. Thomas hadn't really looked at Binky Points in about 20 years, but I exchanged a couple of very pleasant emails with him. It always amazes me when I reach out to a complete stranger and they take the time to talk to me. Thanks, Thomas!

Binky Points made some inherent sense, and Thomas' data looked quite interesting, so I duly coded 'em up, and let 'er rip. Alas, Binky Points were not the panacea I had been hoping for. But it did give me some ideas about how to go about trying a different experiment I have had in the back of my mind for a few years.

It always made intuitive sense to me that honor cards in short suits were likely worth less than the same honor cards in somewhat longer suits. For example, think of KJ, vs. KJx, vs. KJxx, vs KJxxx (or longer). And it seems like multiple honors in the same suit should count extra; e.g., KJx should be worth than Kxx and Jxx in different suits. The 4C's method accounts for this idea to some extent.

Treating specific honor holdings differently, depending on the length in the suit is an evaluation method I wanted to try to expand on. I created a big two dimensional table, with specific honor card combinations in one dimension, and suit length in the other dimension. For the KJ(?) example above, the table has the following entries:

KJ	-.25
KJx	+.18
KJxx	+.20
KJxxx	+.22
KJxxxx	+.24
KJxxxxx	+.26 (also applies to longer suits)

Since a King has a BUMRAP value of 3.0, and a Jack has a BUMRAP value of 0.75, the KJ combination has a base value of 3.75 Working Points. So KJ doubleton gets an evaluation of 3.75 - .25 = 3.50 Working Points. KJx gets a value of 3.75 + 0 .18 = 3.93 Working Points. Etc.

Length points, for suits at least 5 cards long are also added in. Many evaluation methods add extra points for long suits. A very common method is to add 1 extra point for every card in a suit beyond the 4th card. Having experimented with a lot of different ideas, I haven't come up with anything that improves on that simple length addition.

Do this for all four suits, add up the values, and you have a measure I call "Total Combined Suit Length And Strength."

But I am not quite done yet. I make some further adjustments depending on the total number of Aces and the total number of Quacks (Queens plus Jacks) in the hand. Yet another adjustment is made depending on the hand's MLTC (Modern Losing Trick Count.)

Finally, I make one last adjustment, that I learned from Thomas Andrews. The overall shape of the hand is used to calculate an adjustment for both "Working Points" and "NT Working Points." For example, a 6421 shaped hand has an additional Working Points adjustment of +.30, but a NT Working Points Adjustment of -0.10.

I think that's everything. Are these adjustments all exactly "right?" Not a chance. But I have spent way more time than I care to admit trying out different values, and this is what I am going with at this time. I have run a lot of experiments.

These changes led to significant gains in the Jethro vs. Jethro matches when hands were bid to their conclusion.

Changes to the Dynamic Evaluation

Jethro has always had some kind of "dynamic adjustment" to make changes to the evaluation before every bid made, incorporating information learned from the auction. The questions, of course, are "in what circumstances" and "by how much"? What I had done was quite ad-hoc, making adjustments that "felt right" to me. I set out to formalize my approach, and see if I could make those adjustments more accurate in some verifiable way. I identified approximately 250 separate values I was using. I gave them clever names like "unsupportedQueenInPartnersShortSuit" and "downgradeForKingInLHOsuit" and "doubletonHonorInPartnersSingleSuitUpgrade."

I had always wanted to try writing some code to implement a Genetic Algorithm, as the concept seems extremely ingenious to me. What, you may ask, is a Genetic Algorithm? From a quick Google search: *"Genetic Algorithms (GAs) are a class of evolutionary algorithms inspired by the process of natural selection and genetics, often used in machine learning for optimization and search problems. They operate on a population of candidate solutions, iteratively evolving them towards better solutions based on a predefined fitness function."*

Optimizing Jethro's dynamic adjustments seemed to me like a good place to try a GA approach. Not that I really knew anything about Genetic Algorithms. I found a good book on the subject and went to work.

After two or three weeks, a couple of things became clear to me. First, I think the idea is theoretically sound. I believe the answers I was getting would eventually converge to a good solution. Second, and most unfortunately, "eventually" was going to be way, way too long. I was running into the same issues I had in the "methodology" discussion above. I just could not run a determinative number of hands in a reasonable amount of time to create hundreds (thousands?) of generations of parameter candidates.

Back to the drawing board. In the general (GA) approach I had tried, a major problem was that my set of test hands were entirely random. They had to be, in order to test all of the parameters simultaneously. That necessitated an unworkably large number of hands for each iteration of the test.

On most hands, however, only a small subset of parameters are actually used. (Most hands do not have, for example, exactly Qx in a suit LHO bid naturally.) I set out to collect data for hands where each specific parameter was actually used. My goal was to get 20,000 hands for each parameter. I mostly succeeded in the data collection phase. Sometimes the parameters were just not used very frequently, and I never did reach the 20,000 goal.

Then I wrote a program which tested each parameter using the 20,000 hand files specifically collected for use with that parameter. I varied the value of the parameter, keeping track of the results of bridge matches where the only difference was in the parameter value. In this way I was able to optimize individual parameter values.

There are some fundamental problems with this approach. First, and foremost, the method I used treats each of the parameters as independent entities. That's just not so, but I didn't have a sensible way to simultaneously test multiple parameters. Second, and very much related, I am sure the parameters I changed first were influenced by factors not exclusively related to their particular circumstances. More specifically, I suspect when I started that anything I did to make bidding more likely (usually by increasing the parameter value) was likely to be found to be a good idea. In other words, the parameters I changed first are likely overweighted.

At some point I am sure it would be a good idea to go back and run this entire set of experiments again, adjusting accordingly. I've no doubt enhancements would be found.

For now, however, I am moving on. Even with this flawed approach, very substantial improvements were apparent.

Changes to the Bidding Algorithms

Next, I dove into a wide variety of bidding algorithms. I got my fingers dirty dealing with innumerable nitty gritty details. In general, one theme stood out: Bidding More Is A Good Idea. Sure, sometimes this leads to bad results, but on balance it was distinctly better to bid than not to bid. Is this because Jethro isn't very good at dealing with opponents who mess up their nice clean auctions? That is certainly possible. But anytime you can put your opponents on a guess, they will get it wrong at least some of the time. The goal is always to put your opponent on the "last guess." That is just as true in the real world as in the bot world.

Of course, bidding more can be taken to silly extremes; at some point passing is best. As always, the challenge is to find the "just right" Goldilocks zone.

Places where loosening requirements led to better overall results:

- Takeout double thresholds are lower, especially after preempts
- Preempts are now more likely with strength outside the preempt suit

Takeout double auctions have been thoroughly revamped:

- Bidding requirements in response to a takeout double are more consistent
- Double and bid auctions (strong hands) were rewritten
- lebensohl responses (and follow-up) to takeout doubles were much expanded
- Cue bid responses (and follow-up) to takeout doubles were rewritten

Other areas where code has been refined, in no particular order:

- Slam bidding and control bidding
- Responsive doubles (and follow-up)
- Responding to support doubles and support redoubles
- Responding to a weak jump shift
- Preemptive auctions
- Bidding with an agreed fit
- Deciding whether a hand is good enough to make a jump rebid to the 3 level

- Responding to a quantitative NT inquiry
- Deciding when to make a DSI (Do Something Intelligent) double
- Responding to a DSI double
- Defensive bidding (and still… deciding when to whack the opponents remains a sore spot)
- Responding to Leaping Michaels
- Estimating partner's strength, based on the auction
- Making responder's 2nd bid in a strong 2 Club auction
- Advancing partner's two suiter
- Stopper showing auctions
- And many, many more common and not so common situations

Why all this attention to minute details matters so much

Perhaps the most important task for a software bidding engine is to accurately model expert hand evaluation. I would go so far as to say that accurately evaluating hands and judging their potential is a very large part of what makes an expert an expert. I am pretty sure expert bidders do not, however, try to assign exact numeric values to specific features of their hand as they make reassessments during the auction. But that is what a rules-based computer algorithm must do to attempt to model expert behavior.

Most bids made by Jethro are made using logic like: if <some calculated value> exceeds <some threshold> do <something> else <do something else> etc.

In many instances, once a value falls on one side or another of a threshold, that typically sets the tone for the rest of the auction. Once a hand is deemed to be strong enough to force to game, for example, there's no putting on the brakes later. Deciding whether or not to look for slam can also depend on the slimmest of margins. On virtually any hand that falls very close to a bidding threshold -- say within .25 points either over or under the threshold -- it is typically quite easy to construct lies of the other players' cards that would make winners or losers out of the alternate actions. Bidding or passing (if not in a forcing auction) are almost

always both plausible actions. And hands that fall close to some threshold occur *quite* frequently. The goal is to make more winning bids in the long run.

Whatever it is that goes on inside experts' heads, they get these decisions right more often than non-experts. There is often something about a hand that will create a nudge in one direction or another, whether or not it can be can exactly articulated. The challenge, when creating a software bidding engine, is to quantify those things that create nudges. The software doesn't care how close to a threshold we come with some calculation; it is either over or under, and the indicated action is taken. There is no such thing as, "Well, it's close, and I really want to take a bid so that's what I will l do, even though I'm not technically strong enough to do so."

Taking a Step Back, and Looking at a Bigger Picture

When thinking about bridge bidding bots in general, I believe replicating the decision-making mechanisms of master players is not the only thing that is important.

I believe it is also important to ask the questions, "What are you looking for in a computer partner/opponent(s)?" What makes it fun? What makes it annoying? What is it that gets you to say, "I enjoyed that. I would like to do it again," or conversely, "That was a waste of time; I wish I had reorganized my sock drawer instead"?

Perhaps a useful comparison would be to a pick-up partner you hook up with at the partnership desk of a major tournament. Sometimes you hit it off, and form a lasting relationship. Other times you (hopefully) politely say, "Thank you for the game" and move on. In any case, in a casual partnership, if you are like me, you tend not to put your partner's actions under a microscope.

What does this mean in the context of a computer partner/opponents? Here are a few thoughts. I expect other people will have their own ideas about the definition of "what is and isn't fun."

I think there are a few things we should reasonably expect, in re: bidding:

- An understandable approach to bidding. Reasonable explanations for plausible bids that might be made.
- A built-in level of sophistication. This is a fuzzy topic best described by, "You know it when you see it (or don't see it)." For starters:
  - A lot of different bids and sequences that are part of the base system.
  - Sensible actions in competitive auctions.
  - The ability to occasionally pleasantly surprise us with an exceptionally good bid.
- Some level of customization. This is an endless topic. There are countless conventions and treatments. The more popular and necessary ones should be available as options; that list, of course, is quite subjective. Having said that, those same conventions and treatments inevitably lack consistent agreement.
- Dynamic re-evaluation of a hand as the auction proceeds. Sometimes our hand gets better, other times our hand gets worse, depending on the actions of the other players. Sometimes it is hard to tell.

There are also some things we should not expect:

- Whatever conventions and treatments are being played, be aware computer players may not play them exactly the same way you play them. With a human partner, you might be able to make adjustments on the fly. Not so with your computer partner. Do not expect to be able to teach the computer new tricks.
- Do not expect a computer bidder to figure out what you are trying to do when you make bids that do not mean what the computer thinks they mean. You may have a plan, and a person sitting across from you may be able to eventually decipher what you are trying to convey, but a computer partner is far less likely to be able to retrospectively figure it out. On the plus side, a computer partner won't get upset with you for violating system agreements.

- Do not expect a computer bidder to always make the same bid you would make with the same cards. Even experts often disagree about the best bid to make with a particular hand; see any bidding contest for innumerable examples. A human partner won't always make the same bid you would have either. A bid you would not have made is not necessarily wrong or terrible (though sometimes, of course, it is actually truly terrible!). In any case, think about how you would react if a pick-up partner made the same bid. Consider cutting your partner some slack even if your partner is silicon-based.
- Exploiting holes in your computer opponents' bidding system is likely easier than in real life. If you think that is fun, no one is going to tell you to stop, but don't blame the software for not figuring out everything. For example, do not expect a computer partner or opponents to gracefully handle psychs. If you show length in a suit you don't have, computer opponents may well be dissuaded from trying to play in that suit. Along those same lines, your computer partner may not be able to take that kind of a joke, and insist on supporting "your suit" at a level higher than you think reasonable.

Creating an Illusion, and Code That Goes "Clunk"

When you are watching a movie, or reading a book of fiction, the makers/author create their own version of reality, hoping you will have a "suspension of disbelief" for the parts in the story that probably wouldn't really happen that way in real life. If they are successful, you will still know it is fiction, but are willing to accept it as a temporary reality in the world portrayed. But if a character does something so egregiously unlikely that you stare at the screen or page and think, "Not a chance, give me a break!" the magic is lost, perhaps permanently. I think of that as a "Clunk" moment.

A bridge bidding bot is also a work of fiction, and it seems to me similar standards should apply. If I see Jethro make a bid I can't really imagine anyone making in real life, I think "Clunk." I try really hard to avoid those, and change things up when I find 'em. Of course other people undoubtedly have different standards for

what constitutes a “Clunk.” I would very much like to know what other people think of as a “Clunk.”

Here is one thing I’ve noticed which isn’t exactly a “Clunk” but might catch you by surprise. Any hand with 5332 shape, 14 HCP (traditional) and two or more Aces is very likely to be upgraded and opened 1NT (nominally 15-17). Even if the 5 card suit is a major. The same concept extends to other, less shapely hands, too. This chunky 4333 hand, AT92 KT4 AT8 KT5, is also opened 1NT. I spent quite a while doing extensive testing on 14 HCP hands Jethro had upgraded. I could not come up with any specific criteria which made it better not to open 1NT.

I like to think I am pretty good bridge bidder, but I have never claimed “expert” status, whatever that means. There are many, many situations and sequences I have “put under a microscope” with Jethro, when bids were made which surprised me. At the end of the day, I rely heavily, but not exclusively, on the double dummy results to guide me, even when I hear a faint “clunk.”

A Few Thoughts About Bridge Bidding and A/I

Whenever I talk to friends in academia about Jethro, they often gently tell me there is no future for rules based algorithms. They tell me the future of computer bridge is all A/I, all the time. But they tend to be awfully short on details, beyond creating huge databases of played hands and doing more extensive Monte Carlo (random) simulations. I won’t claim to know anything specifically about how neural networks might be used in this application, but I do not believe anyone is particularly close to a breakthrough solution.

The cover article from the May, 2025 *ACBL Bridge Bulletin*, was titled “In Search of Smart Bridge Robots.” Author Sinan Tatlicioglu presented an easily digestible introduction to A/I as it relates to bridge bidding and play. The author sums up the problem succinctly, *“… bridge playing AI must make accurate, statistically sound decisions by modeling countless potential outcomes based on its communication with partners and opponents.”*

And therein lies the rub. I have looked at thousands of hands. When I see Jethro make a bid I disagree with, I routinely try to figure out why, and, if possible, make

a change to the code so that a bid more to my liking is chosen. Then I will try to see if that change leads to unintended, less desirable consequences on other hands. It is not the least bit unusual to discover a specific change I implement only makes a difference once in ten or twenty thousand random hands. The problem is that there are countless different situations which can arise when bidding completely random hands to their conclusion. The effect is that one time in ten thousand suddenly does not *feel* like such an unusual event, because for practical purposes there is an unlimited supply of different potentially problematic "events."

I do not know how the A/I wizards deal with this problem. Just identifying which of the millions of "expert" auctions they have access to apply to the specific problem at hand would seem to be a daunting, but perhaps not impossible, task. I know they are very smart people, and I wish them well. There may be different solutions I have not considered.

Another thought, regarding Monte Carlo (random) sampling. To try to answer questions by simulating, I think the basic idea is to create a bunch of hands that fit the auction -- including the opponents' bidding -- then run Double Dummy analyses of the combined hands. That's a fairly CPU intensive process. Even if you can do 10 DD calcs/second -- which is notably faster than I can do them -- in 10 seconds you've only done 100, which I'm thinking isn't really nearly enough. And ten seconds feels like a long time to wait for a computer bridge player to do something.

As always, garbage in --> garbage out. If you do not simulate using the correct representative hands, you will not get a good answer.

Creating simulation hands for partner and the opponents requires some good judgment/guesswork. It is an interesting, frustrating exercise. Having tried to do this, I can tell you it is not nearly as easy as it sounds. Create a very large number of random hands, then calculate a "similarity score" for each of them and sort them based on that score, to get what you hope is a representative sample of the hands you are actually interested in. What's your "similarity score" algorithm? Good question. Just because a player makes a bid that shows, say, 5-5 in the

majors, they might have fudged a bit, and only have 5-4. Or maybe they have either more or less than strength than they putatively promised. The point is you cannot exclusively rely on information from the auction when creating your "similar" hands. You must expand the universe of hands you are looking at. Sensibly incorporating negative inferences – bids that were *not* made, for example – is particularly challenging. The basic concept sounds great, and it is, but it is just not simple to effectively implement when you dive into the details.

If you want to take the "similarity score" exercise one step further, you could bid your sample hands up to the point in the auction where you have the next decision to make, to see which hands make the same bids. That would necessarily rely on other assumptions, including how your opponents would bid specific hands. The number of iterations to try would quickly become very large.

I spent hundreds of hours of computing time calculating the initial values and parameters used by my evaluators. I ran a lot of tests overnight. That is time already invested, which hopefully makes up a bit for all the analyses I am not doing "on the fly." I like to think of this as having run my simulations in advance, to some extent. If nothing else, the approach I am using has the advantage of being lightning fast.

On Bidding Contests

I am not a big fan of bidding contests, especially as related to Jethro. I don't think bidding contests are really the best way to judge software bidders. Not that I have any particularly good ideas about how to do so. I have spent a little time (but not very much) trying to think about how to best "score" a bidding engine on a set of random hands. It is a daunting task, one that I have not come remotely close to even defining.

In any event, having looked at more than a few contests, it appears a lot of good scores in bidding contests are the result of getting to the "right" game or slam when more than one makes. Or somehow bidding either more or less than a normal evaluation of the cards would suggest.

That is something I just don't care about. Getting to 3NT with a 9 card M fit, instead of 4M when both contracts take 10 tricks; or getting to a Moysian major game that takes 10 tricks vs. 9 in NT; or getting to 6NT instead of a different good slam -- are inconsequential to me. As far as I am concerned, it is just fine to not be in the "highest" scoring contract, as long as you get to a good spot. But it is often really important for scoring well in bidding contests.

In contests where the objective is to make a single bid, as opposed to bidding an entire hand to completion, the “winning” answer frequently seems to be either some kind of nebulous cue bid, or not necessarily well-defined double, punting the decision back to your partner.

Short digression:

> Jethro will make a number of different kinds of cue bids, but they are always explicitly described at the time they are made. Most cue bids made directly after partner shows a major for the first time are “fit-showing.” In other well-defined competitive auctions, Western cue bids (both “asking” and “telling”) are made at the three level. Sometimes a bid which looks like a Western cue bid is made, but subsequent actions will reveal the previous “Western” description to be spurious. Finally, some cue bids are just defined as “ambiguous” at the time they are made. Partner is expected to bid naturally, and hopefully the confusion can be sorted out later.
>
> At the time they are made, doubles which are not otherwise explicitly defined (e.g., a negative double or a takeout double) are always described as either DSI (Do Something Intelligent) or Penalty. There is extensive logic to decide which is which, according to the context of the auction. The objective is to not leave anyone at the table guessing as to the doubler’s intent.
>
> Most low level doubles are DSI; the doubler will either have at least some support for unbid suits, or a hand so strong they believe they cannot pass. Jethro will pull a penalty double if it is judged to be right to do so. Converting a DSI double to a penalty double by passing is not uncommon.

End of digression.

Having said all that about not liking bidding contests, I do, on occasion turn Jethro loose on bidding contest hands to see what happens. If nothing else I am likely to identify some hands (often those one in ten thousand) where I might want to try something else.

For no particular good reason I went back to the very first "Challenge The Champs" contest from the January, 1967 issue of *The Bridge World*. The teams were Peter Leventritt/Howard Shenken vs. Alvin Roth/Bill Root, well known luminaries of the day. I own books written by at least 3 of them.

By contemporary standards, there was a lot of timid bidding, especially by (no surprise) Roth/Root. Alvin Roth was a notoriously "sound" bidder, and encouraged/expected his partners to bid similarly. There were also some systemic bids that looked a bit odd to me, though the commentary suggested they were just fine.

From the introductory description. (Remember, this is the first time CTC appeared. Astonishingly, the format looks pretty much identical today.)

"In theory, the match-point awards range from a top of twelve to zero. In practice, though, the best score awarded will usually be ten or less; higher match-point results generally come from opponents' gifts, unavailable here. Hence, the maximum total available in each match is 100 points. If you compute your own total, we suggest that you regard it as your percentage *score in a big one-session pair tournament: 70% or more is magnificent and figures to win; a score in the sixties is one to be proud of; a score in the fifties is better than average."*

On the 10 boards, Leventritt & Shenken prevailed over Roth & Root:

"Final Score: Schenkin-Leventritt 70, Roth-Root 64.
The losing score was an excellent one (Roth-Root reached only two really poor contracts, and were in many exceptionally good ones) but the winners were even better."

And how, you may ask, did Jethro fare on those 10 boards? Drumroll, please.....

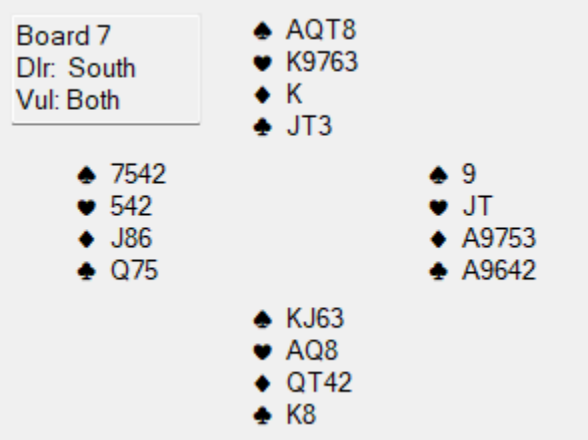
Jethro's score added up to 81! Yeehaw! I was not expecting that! Only one board scored below average, when Jethro got to 3NT (as did Schenken/Leventritt) on a hand where each player had 12 HCP, and West had a decent 6 card suit (♥QJT853).

Let me stress this result is NOT typical for Jethro in a bidding contest. I did not think these were a particular challenging set of hands. I believe a good – not necessarily expert -- modern pair should be expected to bid them as well as Jethro. But it sure was fun for me to see! Now I just need that time machine to take me back to the 1960's.

Some Fun Hands

No update would be complete without including a few hands that caught my fancy.

1) A bit of a hodge-podge of decisions to make for multiple players at the table. Showing multiple different aspects of Jethro's bidding.



Here is how Jethro bid all four hands:

West	North	East	South
			1NT
Pass	2♣	2NT	Pass
3♣	X	Pass	3♠
Pass	4♠	Pass	Pass
Pass			

1NT is a typical 15-17HCP opener for South.

West certainly has nothing to say.

North knows they want to be in game, but what game? 4♠ or 4♥ or 3NT are all in play. North starts with Stayman, expecting to raise 2M to game, or bid 3M (Smolen or not, depending on what is on the N/S convention card) over 2♦ to force to game. The North hand is judged to be just a bit shy of the strength to go slam hunting.

East has other ideas, getting in there with a 2NT bid, showing the minors. This *might* turn out to lead to a -1100 (or worse) disaster, and many players would not choose to bid 2NT. My experience with Jethro, however, has led me to believe that bidding more is a better idea than bidding less in many circumstances, especially with shapely hands. Maybe 2NT leads to a making 3m, or to 3m down 1, when N/S were making +110 in 2M. Maybe N/S get too high. Disasters won't occur on *every* hand. Let's see what happens here.

South has four spades, but does South really want to take an immediate 3♠ bid over 2NT? North's Stayman bid did not promise any particular values; with some bad hands, North will start with 2♣ and try to get out in 2M via a "crawling Stayman" sequence. East's 2NT bid promises only modest values. For all South knows, West could be sitting behind them with a good hand. So South passes, knowing that North has another bid coming.

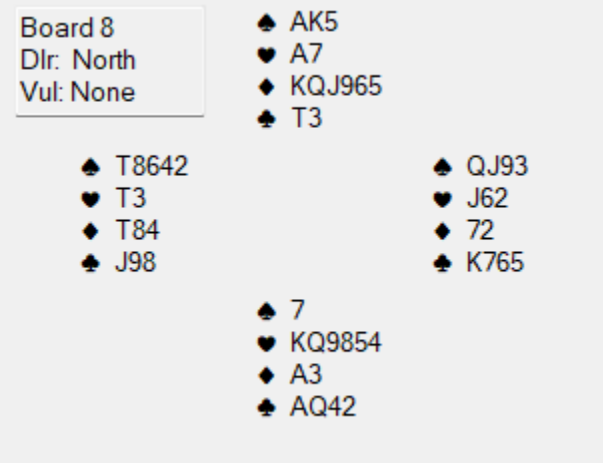
West most assuredly does not have a big hand, and chooses 3♣, bidding up the line with the same number of cards in each minor.

North has to do something, but what, exactly is not clear. They punt the decision back to partner with a "Do Something Intelligent" double. As mentioned above, a lot of thought and energy has gone into deciding when a double is definitely for penalty, as opposed to the more nebulous DSI.

Should South sit for the double? Maybe. But with 4 Spades and 10HCP in the majors, South is happy enough to show their Spade suit. North has no trouble bidding 4♠.

So N/S get to the same 4♠ contract they would have gotten to without the interference. The interference, however, made them work a bit harder, and gave them some chances to make wrong decisions. If, for example, North held a hand like ♠AQTx ♥JTxxx ♦x ♣Jxx their decision over 3♣ might not have been quite so clear cut.

2)



West	North	East	South
	1NT	Pass	4♦
Pass	4♥	Pass	4NT
Pass	5♥	Pass	5NT
Pass	7NT		

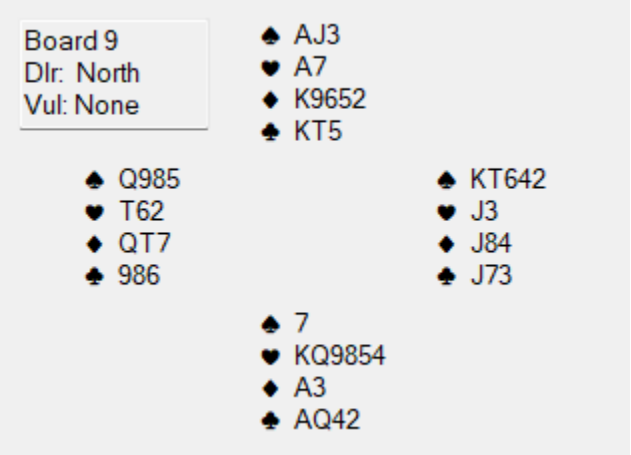
North is a bit heavy for 1NT, but what is the alternative? I think the hand is too strong for 1♦ then 3♦, and 1♦ then jump to 2NT (or 3NT??) is flawed as well, with 2 doubletons, one of them worthless. So I am ok with a heavy 1NT. If you don't think it is optimal, I hope you will agree it is plausible.

Over 1NT, South is immediately thinking slam. With a perfecto North hand – A♠, A♥, K♦, and K♣ – 13 tricks are within sight. Texas transfer then Blackwood seems just right. 5NT confirms all the keycards, plus the Q♥. Now it is North who can count 15(!) tricks – 2♠ + 6♥ + 6♦ + 1♣ – so they just blast away to 7NT!

On a bad day Hearts won't break, and you'll go down. Most days you will be happy when all the tricks come rolling home.

However…

Suppose North has a different hand. A much more modest 1NT opener, with only 15 HCP, but in all the right places:



West	North	East	South
	1NT	Pass	4♦
Pass	4♥	Pass	4NT
Pass	5♥	Pass	5NT
Pass	6♣	Pass	6♦
Pass	7♥	Pass	Pass

Over 5NT, 6♣ shows the K♣ (showing specific Kings, not number of Kings is Jethro's default agreement, which you can change if you desire). 6♦ asks North to bid the grand if they have the K♦. North obliges.

This is not quite a total slam dunk, as there might be a fourth round club loser, but in practice it will usually be a good bet. North might have the K♠, or the Q♦, or the J♣ or be able to set up a long suit, or other possibilities in the club suit. Thirteen tricks will usually make, as long as the Hearts split.

3)

Board 2
Dlr: East
Vul: N/S

	♠ T62 ♥ 976 ♦ JT54 ♣ AK2	
♠ A953 ♥ KT3 ♦ 863 ♣ 986		♠ J7 ♥ AQ5 ♦ K7 ♣ QJ7543
	♠ KQ84 ♥ J842 ♦ AQ92 ♣ T	

West	North	East	South
		1♣	X
1♠	1NT	2♣	Pass
Pass	2♦	Pass	Pass
3♣	Pass	Pass	Pass

Double Dummy Analysis

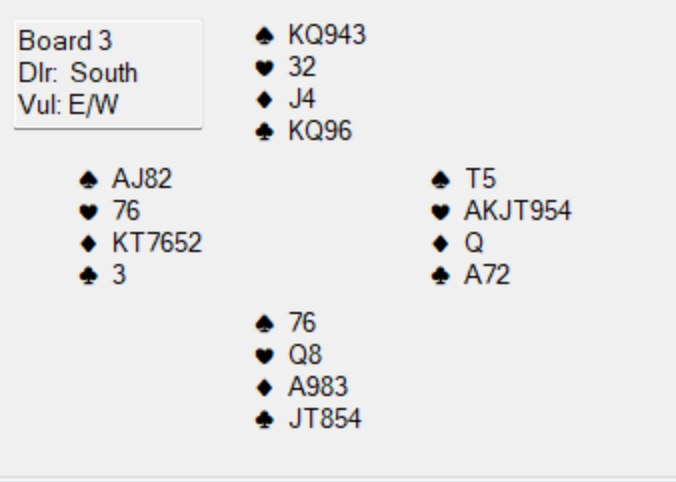
	W	N	E	S
NT	5	8	5	8
♠	5	8	5	8
♥	5	7	5	7
♦	4	9	4	9
♣	8	5	8	5

Pretty well done, all around, in a competitive auction. With a double stopper in Clubs, North initially responds 1NT to the takeout X, then balances in later with a 2♦ bid. West is willing to take the push to 3♣ with a known 9 card fit.

Other choices might have been made along the way. This sequence seems plausible.

4)

A somewhat mundane kind of hand, correctly evaluated to get to a good spot:



West	North	East	South
			Pass
Pass	1♠	2♥	Pass
3♦	Pass	4♥	Pass
Pass	Pass		

South's initial pass is beyond reproach.

I suppose some maniacs might open a weak 2♦ with the W hand, but with ♠AJxx, I don't think many decent players would consider doing so. There is just too much chance of missing a potential Spade fit.

North's 3rd seat 1S opener white vs red opener seems fine to me. I like 1♠ much better than either 2♠ or Pass.

East has quite a good hand. It is actually not all that far from a X and bid Hearts hand, but East settles for a simple 2♥ overcall.

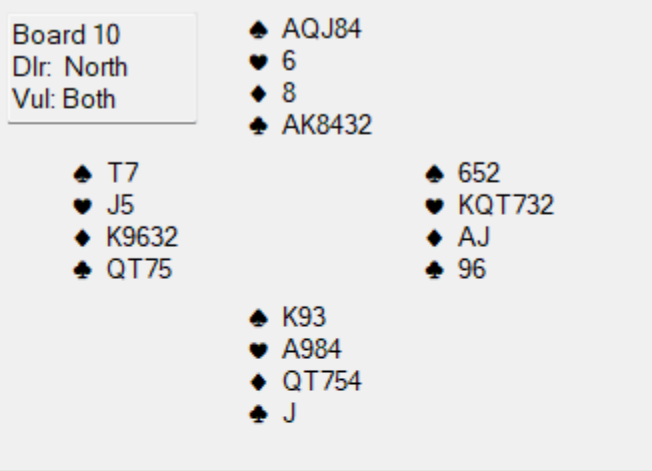
South is not judged to have enough for either 3♣ or a negative X; ♥Qx is a dubious value after East's overcall. I am not sure how 2NT by South should be best interpreted, but Jethro plays it as natural. In any case South passes, which looks plausible to me.

Does W have enough for a non-forcing 3♦? Jethro thinks it does.

Over 3♦, which should show decent values, East bids 4♥, what they think they can make. East's stiff ♦Q is upgraded after West shows ♦. West's stiff ♣ is an unexpected treat.

How would other players / bots have bid these hands?

5)



West	North	East	South
	1♣	2♥	2NT
Pass	3♠	Pass	4♠
Pass	4NT	Pass	5♥
Pass	6♠	Pass	Pass
Pass			

Two suiters can be difficult to evaluate. Jethro tends to evaluate them very highly, especially if the suits are stellar, as they are here. Occasionally this leads to silly results. On this hand, after the ♠ fit is found, all North really cares about are keycards.

Getting to a great slam with 24 HCP. A little careful play should earn 12 tricks. Jethro plays South's 2NT here as natural, but not forcing. For Jethro, North's 3♠ rebid promises this shape, since South denied 4 Spades when they failed to make a negative X.

What Comes Next?

First, I need to roll all of my changes into the base code. That will be tedious, and require close attention to detail, but shouldn't be overly onerous. I am guessing maybe a week or so should get 'er done. Please let me know if you would like copies of the latest programs once everything is integrated.

I believe I have made substantial progress. There will always be room for improvement. When I sit myself in the South seat now, and play a match vs. Jethro I usually, but not always, win. If I am tired I am much more likely to lose. Of course, I am intimately familiar with how Jethro is likely to bid any given hand, and am no doubt blind/unreasonably tolerant to Jethro's quirks. Other players' experience will vary.

Going forward, I would somehow very much like to get Jethro in front of a wider audience. I think I am justifiably proud of what I have created. I believe that expanding Jethro's audience is only going to be possible if there is some kind of web-based interface. Which is still something I don't know anything about. I do know, however, a standalone Windows executable only program is not a viable long term solution. Suggestions are welcome. I have no clue what I am going to do about this, but it has become my top priority.

At this time I am not really looking for additional feedback from the long-suffering beta testers. Thank you very much for sticking with Jethro all the while. The simple reality is that it is not particularly difficult for me to find hands where I think changes could be effectively made. All those different "one in ten thousand hands" really do add up. If I open it up again for everyone to send me hands where they disagree with the bidding, I fear being overwhelmed.

If you made it all the way to the end, thanks for your patience and indulgence.

-Bob

♠♥♦♣