# Large Scale Q-tables

**Kalle Prorok**
Department of Computing Science
Umeå University
SE-901 87 UMEÅ
Sweden
kalle.prorok@tfe.umu.se

**Abstract**

This article studies the possibilities for Q-learning to learn the bidding in the card-game bridge. The learnt bidding systems are quite useful but below the level of an average club-player. Some studies regarding the relation between the learning rate, the exploration rate and initial Q-values are also done.

## The Application

Bridge is a card-game normally played by four humans. It has some interesting properties making it suitable as a test platform regarding Collaborative Learning Agents. Each player is on his own but playing together pair-wise; a dynamic, collaborative and adversial game. Some agreements are made between the players on beforehand but most situations have to be taken care of by commonsense logic. This makes it difficult to pre-program a computerized player. Instead it makes it interesting to let the computer-player learn from exhaustive experiences. The optimal behavior is also unknown, which make the results from such an artificial player valuable.

My hope is that these experiments with Bridge will be applicable for other domains like network management (information transfer), compression algorithms (finding descriptions of i.e. a face automatically), electronic commerce (how to perform an auction) and robotics (learning to interact). Another area might be language learning; how did we invent a language "in the beginning".

Blocks world (Winston 1977) has been used as a small-scale model system for decades in the Artificial Intelligence community. Today's problems are of a different kind; co-operation, stochastic and/or non-linear systems, emergent properties etc. My suggestion is to use the international card-game Bridge as a new model world with some interesting properties from the real world but still on a reasonably small scale. Some of the things that can be studied using the Bridge platform are: Collaboration/Cooperation within pairs and teams; Competition between pairs, teams and countries; Communication - information transfer with limited bandwidth; Insecure domain - not all information is visible for all parts; Stochastic domains - there is a random factor (how the cards were dealt); The role of concentration; mental training; Planning (of bidding and playing); Emergent properties; single suit versus multi-suit plays, end-plays; Democratic (descriptive) or Captain (enquiry) based methods; Tactics, Strategy, Risk taking, "the only chance" or safety play "guarding"; Constructive versus destructive ways of doing things; Psychological aspects - to be unpredictable, when to fool partner or opponent; Philosophical aspects - try to find out what partner or opponents think; Memory; a complex bidding systems versus an easy to remember, understand and use bidding system; Getting information - what and when; Reducing amount of information to opponents who may take advantage; Learning from mistakes and successes.



The game is described in many standard textbooks and the rules can be learnt within a few days but requires several years of practice to become a master. The goal with the game is to score points related to the number of tricks your partnership is able to take and this is dependent of how good you are to reach a suitable contract; the game is thus separated into two phases: the bidding (auction) into a contract and then the play in such a contract. An example deal is presented in Fig. 1 and a bidding in Fig. 2.

## Background

Bridge is fairly international with similar rules all over the world and there are more than one million players in the world. Now it is also an Olympic game. There are bridge programs performing well and it is also a matter of time before the computer beats most human bridge players. The main reason for this is the technical advantage a computer has (remembering all cards during the play and the bidding system, possibilities of statistical simulations and calculations making it possible to play well) compensating for the lack of intuition and other psychological aspects by making fewer mistakes. Today's situation (December 2001) is that the play with the cards is very good but bidding lacks behind. The reason for this is that entering human judgement as rules in a bid-system database is very tedious, Fig. 3.

There are some computer programs around (reviewed in (Frank 1998)) able to play Bridge but most only at a low (beginners) level. One of the never and better candidates is AI-researcher professor Matthew L. Ginsberg's GIB (Ginsberg 1999), which he guess will be better than humans by the year of 2003 (Prorok 1999). It is able to do declarer play

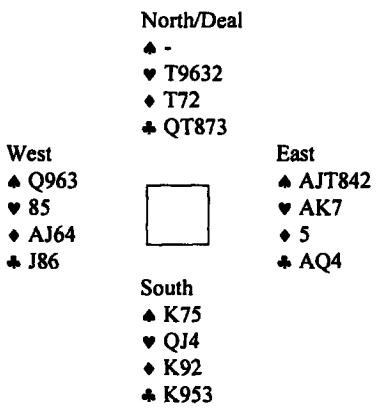


Figure 1: A typical deal in schematic form. It is dealt by player North who then starts the bidding with East next in turn. North is void in (have no) spades but has a weak hand and passes. East has a strong hand and normally opens the bidding by giving a bid, maybe 1 ♠. South has an average hand but not enough to enter the bidding with a bid other than Pass.

| N | E | S | W |
|---|---|---|---|
| P | 1♠ | P | 2♠ |
| P | 4♠ | P | P |
| P | | | |

Figure 2: The schema of one possible bidding with the hands in Fig 1. Three passes ends the auction. 4 ♠ requires EW to take 6 ("the book") + 4 = 10 tricks out of the possible 13.

```
1*#08115 "  (5[[CDHS]:"          !.!
%0% %.% <0x1083> *2153*#08116 "   5[+#b]"              !DOPI!
%6% %:H[^:]*~8&[^:]*~8&% *2160*#08117 "   5[++#b]"
!DOPI!                            %7%
%:H[^:]*~8&[^:]*~8&[^:]*~8&% *2159*#08118 "   5[+++#b]"
!DOPI!                            %8% %:H~8&.~8&.~8&.~8&%
*2156*#08119 " 4N:$7"                !.!
%0% %.% *1501*#08120 "  (5N|[67][CDHSN]):"   !.!
%0% %.% <0x1083> *2153*#08121 "   X"                  !DOPI!
%3% %.% *2154*#08122 "   P"                  !DOPI!
%4% %:H[^:]*~8&% *2155*#08123 "   X"                  !DOPI!
%5% %:H[^:]*~8&[^:]*~8&% *2154*#08124 "   P"
!DOPI!                            %6%
%:H[^:]*~8&[^:]*~8&[^:]*~8&% *2155*#00126 "
.*{{{{{{{{{{{{[CDHS]<#j=3><#k=6>:P:4N"        !.!
%0% %.% *2037*#00127 "  :(P|X):"             !.!
%0% %.%#00128 "   @ACE_BLACKWOOD~"     !BLACKWOOD!
%1% %.% *1501*#00129 "   @ACE_RKCB~"          !RKCB!
%1% %.%#00130 " [123]{{{{{{{{{{{{{.:P:5N$12"      !GSF!
%0% %.% *2229*#00131 "  :,:"                 !.!
%0% %.%#00132 "   7#a"                 !.!
%5% %:H[^:]*~9>#a% *2034*#00133 "   7#a"                !.!
%5% %:H[^:]*~6=#a% *2034*#00134 "   6#a"                !.!
```

Figure 3: A small part of the GIB-database, showing how the Ace-asking bid Blackwood(4 NT) should be used.

on a very high level. This also makes the bidding better because it simulates the play to find the best or most probable contract.

GIB uses the Borel simulations algorithm (Ginsberg 1999):

> To select a bid from a candidate set $B$, given a database $Z$ that suggest bids in various situations:
>
> 1. Construct a set $D$ of deals consistent with the bidding thus far.
> 2. For each bid $b \in B$ and each deal $d \in D$ use the database $Z$ to project how the auction will continue if the bid $b$ is made. (If no bid is suggested by the database, the player in question is assumed to pass.) Compute the double-dummy result of the eventual contract, denoting it $s(b, d)$.
> 3. Return that $b$ for which $\sum_d s(b, d)$ is maximal.

There are dozens of other programs for playing, generating deals (Andrews 1999) etc. Ian Frank has made an end-play analyzer/planner FINESSE which seems to work well but too slow for practical use (written in interpreted PROLOG) and he has also written an excellent doctoral's thesis "Search and Planning Under Incomplete Information" (Frank 1998). PYTHON, (Sterling & Nygate 1990) as cited in (Björn Gambäck & Pell. 1993) is an end-play analyzer for finding the best play but not on how to reach these positions.

The former World Bridge Champion Zia Mahmood has offered one million pounds to the designer of a computer system capable of defeating him but has withdrawn his offer after the last match against GIB which was a narrow win for him.

Most programs basically use a rule-based approach. The first one (Carley 1962) with four (!) bidding rules with a performance like "The ability level is about that of a person who has played dozen or so hands of bridge and has little interest in the game". Wasserman (cited from (Frank 1998)) divides the rules into collection of classes that handle different types of bid, such as opening bids, responding bids or conventional bids. The classes are organized by procedures into sequences."Slightly more skillful than the average duplicate Bridge player at competitive bidding". MacLeod (MacLeod 1991)(cited from (Frank 1998)) tries to build a picture of cards a player holds but cannot represent disjunction; when a bid has more than one possible interpretation. Staniers (Stanier 1977)(cited from (Frank 1998)) introduces look-ahead search as an element of Bayesian planning and Lindelöf's COBRA (Lindelof 1983) uses quantitative adjustments. Lindelöf claims that COBRA bidding is of world expert standard. Ginsberg has invented a relatively compact but cumbersome (Fig. 3) way of representing bidding systems in his GIB-software. By using such representations he reduced a rule-base of about 30,000 rules into one with about 5,000. Such representations are of course prone to errors.

## The Rules of Bridge

Bridge is a card game played with a deck of 52 cards. The deck is composed of 4 suits (spades ♠, hearts ♡, diamonds ◇ and clubs ♣). Each suit contains 13 cards with the Ace as the highest value and then the King, Queen, Jack, Ten, 9,

..., 2. The first five are often abbreviated to A, K, Q, J, T. The game begins with some shuffling of the deck and then the cards are dealt to the four players, often denoted North, South, East and West (N, S, E, W) Fig. 1. N and S play together in a team against E and W.

Before the card play begins an auction (bidding) takes place. One of the teams[1] wins a *contract* to make at least a certain number of *tricks*. Some levels give bonus points.

To prevent opponents from taking a lot of tricks in a long suit of theirs, a trump-suit is looked for in the bidding. If no common trump suit is found, a game without trumps can maybe be played; No Trump (NT), which is ranked higher in the bidding than the suits which are ranked ♣ (lowest), ♢, ♡, to ♠ (highest).

During these two phases, the bidding and the play, there is cooperation in pairs playing versus each other with four players at each table. The pairs communicate via bidding where the bids are limited to a few (15) "words", but many sequences "sentences/dialogs" are possible. Opponents are kept informed[2] about the *meaning* (semantics) of the bid or sequence. This can be seen as a mini-language with words like 4, 1, Spades, Double, Pass etc. and these words can be combined via simple grammatical rules into short "sentences" like 3 Spades. The sentences are put in a sequence "conversation" called the bidding. It can look like N:Pass - E:1 Spade - S:Pass - W:2 Spades, N:Pass - E:4 Spades - S:Pass - W:Pass, N:Pass with semantic meanings like "I have a hand with spades as the longest suit and above average number of Aces, Kings and Queens" - "I have support for your spades but not so good hand", "I have some extra strength" - "I have nothing to add". The first non-pass bid is denoted *opening*. A full bidding schema is presented in Fig. 2.

When the bidding is finished, a pair handles the final contract (4 spades here) and the play with the cards can begin by taking tricks. After the play, the pairs are given points according to the *result*. In tournament play the points are compared to the other pairs playing a *duplicate*[3] version of the deal, almost compensating the factor of luck with "good cards".

In our work we have studied undisturbed bidding which means the pair is free to bid on their own without disturbing interventions from the opponents. There is no need for preemptive bids and sacrificing or to keep in mind the importance of reducing the amount of information transferred to the opponents. By having these limitations we are able to learn better in shorter time. Less memory is also needed.

## Representation

A bridge-hand as a human views it is seen in Fig. 4. A shorter, standard, representation with almost the same meaning (losing the graphic "personality" of the cards) is

[1] With the highest bid.

[2] If they ask.

[3] Or replayed later by storing the deal in a wallet.

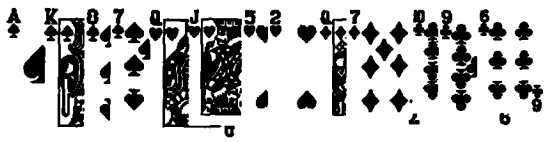

Figure 4: A typical hand

| Card | High card points (h.c.p.) |
|---|---|
| Ace | 4 |
| King | 3 |
| Queen | 2 |
| Jack | 1 |

Table 1: The estimated strength of the high cards.

♠*AK*87
♡*QJ*52
♢*Q*7
♣*T*96

or, simply, AK87, QJ52, Q7, T96 with the order of the suits understood. The total number of possible hands are $\binom{52}{13} \approx 6 * 10^{11}$. There are 8192 different suit holdings (AK92, AK82, etc.), and only 39 hand patterns (5431,5422, etc. disregarding suit-order) (Andrews 1999) and 560 shapes (5431, 4513, 5422,..) in total. To estimate the strength in a hand, Milton Works invented the high card points (h.c.p.) (table 1).

The total number of h.c.p. in a deck is 40 and the average strength per hand is exactly 10. A hand with AKQJ, AKQ, AKQ, AKQ contains 37 points which is the maximum. Such evaluations are needed to estimate the number of tricks a pair of hands are able to take. Robin Hillyard (Hillyard 2001) has calculated the expected number of tricks at no trump, presented in table 2 based on GIB-simulations and the results are close to basic rules of thumb used by bridgeplayers. Thomas Andrews (Andrews 1999) has made some analysis showing that the value of the Ace is slightly underestimated in the Milton Works 4321-scale.

Of course this scale is just to make an rough estimate of a hand and every bridgeplayer adjust (reestimate) the value of the hand, maybe dynamically during the bidding due to fit etc.

One add-on adjustment invented by Charles Goren is the distributional strength (Table 3) where the total value of our hand is 12 $(h.c.p)$ + 1 (doubleton ♢) = 13 points. The

| H.c.p | 20 | 21 | 22 | 23 | 24 | 25 |
|---|---|---|---|---|---|---|
| Tricks | 6.1 | 6.7 | 7.2 | 7.6 | 8.2 | 8.7 |
| H.c.p | 26 | 27 | 28 | 29 | 30 | |
| Tricks | 9.1 | 9.7 | 10.1 | 10.6 | 11.1 | |

Table 2: The average number of tricks with a given strength.

| Number of cards in a suit | Distributional value |
|---|---|
| no card (void) | 3 |
| one (singleton) | 2 |
| two (doubleton) | 1 |

Table 3: Distributional points.

idea behind is that shortage in a suit makes it possible to ruff in a trump contract and thereby increases the strength of the hand. These distributional points are normally not added during a search for a no trump (NT) contract. The strength is used to define a bidding system in terms of shape-groups and strength intervals.

Our hand can be represented as a 4423-shape (4 spades, 4 hearts, 2 diamonds and 3 clubs) with the strength of 12 h.c.p. The number of possible hands in this representation is $560 * 38 = 21180$. In our approach we reduce it further by just keeping the distribution of the cards (the shape), and the *range* of strength:

$$4432, medium$$

where medium might show 12-14 (Goren)points.

**Allowed systems** In most real tournaments the possible systems are limited by assigning penalty points (dots) to non-natural opening bids (Svenska Bridgeförbundet, SBF and World Bridge Federation,WBF). This is so because it is difficult for a human opponent to find a reasonable defence within a few minutes to a highly artificial bidding system. A certain number of points are allowed depending on the type of tournament.

## The Learning

The Q-learning (Watkins & Dayan 1992) updating rule

$$q_{situat,bid} \quad : \quad = q_{situat,bid} + lr * \qquad (1)$$
$$(r + \gamma \max_{bid'} q_{situat',bid'} - q_{situat,bid})$$

was used for all three (response, rebid, 2nd response) bidding positions (when applicable) after each deal. The shape, strength, position (E,W) and bid-sequence is encoded as part of the situation and the $q$-values were stored in a multidimensional integer array. $q$ is the average resulting score having this hand and following the policy, $lr$ is the learning rate, $\gamma$ the discount rate and $r$ is the reward. The bid with the highest $q$ was selected in a particular situation unless exploration was actual, on which any bid was selected with equal probability. If several bids had the same $q$-value, one of them was randomly selected.

In practice this meant one out of 2048 opener were combined with one of the 2048 responders according to the current example in the database and they had to learn together with their partners; thence multi-agent reinforcement learning. The pair's q-values were updated when the bidding was finished. This was repeated multiple times.

### Reinforcement signal

Some preliminary tests with the reward was done.

What value of the reward $r$ should the evaluator give the learners?

a) Calculate the difference between the actual score to the score achievable on this combination of hands:

$$r = score - achievable \qquad (2)$$

It works badly due to high exploration. Assuming a start value of Q with zero "optimistic start", almost all tried bids will start with negative scores (strange contracts) making them worse than hitherto untested alternatives. This will lead to a huge, time consuming exploration phase of all alternatives.

b) Difference as above but added 200 points:

$$r = score - achievable + 200 \qquad (3)$$

A contract not worse than -200 is denoted better than the untested cases. The Pass always system evolved guaranteeing a score of 200. Maybe a long run will improve this policy and the pass-system is a kind of eye of a needle?

c) Absolute score. Simply reinforce the learner by the actual score:

$$r = score * 10 \qquad (4)$$

It worked best, at least on our short (one day or two) runs. The score was scaled with 10 to try to avoid integer round-off errors when multiplied by the learning rates.

d) Heuristic reinforcement.

$$r' = r - 100 * isToHighBid \qquad (5)$$

To avoid high-level bidding with bad cards there was an extra negative reinforcement on those bids that were above the optimal contract.

### Discount/Decay

How far-sighted should the bidders be? By selecting different values for $\gamma$ in the Q-learning different behaviors was obtained. A low value make the learners prefer a short bidding sequence and resulted in guessing into 3NT more often. Maybe a value higher (not tested) than the normal limit of one will encourage a more informative, long sequence? Eligibility traces was not used but the result after the bidding was used for all positions.

## Results

Some of the ideas above were implemented in a program called "Reese", after the famous bridge player Terence Reese who died 1996. He was known for his excellent books and simple and logical bidding style. Regarding our bidding program there was hope for finding a super-natural bidding system but so was not the case. This program requires a computer with at least 348 MB of memory with the described configuration. The experiments can be seen as what can be achieved with these methods in a complex two-learner task and also gives some hints of how to chose learning parameters.

## Introduction

By using the huge database of GIB-results when playing against itself for 717.102 deals, we found a reasonable evaluative function. The database was generated using an early version of GIB but both the declarer and the opponents were not so strong so maybe the result is quite reasonable anyhow. We have used the 128 most common shapes which includes all hands with at most a six-card suit except for the rare 5440, 5530, 6511, 6430 and 6520 shapes. This covers about 91% of the hands and, neglecting the inter-hand shape influence, $0.91 * 0.91 = 83\%$ of the hand-pairs. The strength was classified into the following 16 intervals (although this is easy to modify):

| **Class** | **0** | **1** | **2** | **3** |
|---|---|---|---|---|
| h.c.p. | 0 – 3 | 4 – 6 | 7 – 8 | 9 |
| **Class** | **4** | **5** | **6** | **7** |
| h.c.p. | 10 | 11 | 12 | 13 |
| **Class** | **8** | **9** | **10** | **11** |
| h.c.p. | 14 | 15 | 16 – 17 | 18 – 19 |
| **Class** | **12** | **13** | **14** | **15** |
| h.c.p. | 20 – 21 | 22 – 24 | 25 – 27 | 28 – 37 |

The intervals were heuristically selected in accordance to frequency of appearing.

To reduce memory requirements, learning time and simplify the restriction of learning into allowed bidding systems, the opening bids was defined by the user as a small two page table. Part of the definition of the opening bids in bidsyst.txt is (the strengths 9, 13, 14 and 28 were removed here for ease of showing):

```
Shape 0  4  7  10 11 12 15 16 18 20 22 25
6313  P 3S 2D    1S                2S
6241  P          1S                2S
6232  P 3S 2D    1S                   2S
6223  P 3S 2D    1S                   2S
6214  P          1S                2S
6142  P          1S                2S
6133  P 3S 2D    1S                2S
6124  P          1S                2S
5521  P          1S                2S
5512  P          1S                2S
5431  P             1S                2S
5422  P                1S             2S
5413  P             1S                2S
5341  P             1S                2S
5332  P                1S 1N    1S 2C 2N 2S
```

The bidsyst.txt can be rearranged (preferably with a program) from this matrix representation into an input file with examples. Each strength 0..37 is used to generate an example, resulting in $128 * 38 = 4864$ examples, which look like this (high card points, number of Spades, Hearts, Diamonds, Clubs, recommended opening-bid):

```
0,6,4,2,1,pass
1,6,4,2,1,pass
2,6,4,2,1,pass
3,6,4,2,1,pass
4,6,4,2,1,pass
...
```

It can be represented as a decision tree or ruleset but this introduces errors (due to some pruning; about 2% is erroneous), is larger (five and ten pages respectively) and considerably more complex to read.

The See5, inductive-logic, software (available via http://www.rulequest.com) generates the following (slightly modified) decision tree:

```
See5 [Release 1.13] Wed Mar 14 2001

    Options:
        Generating rules
        Fuzzy thresholds
        Pruning confidence level 50%

4864 cases (5 attributes) from open.data

Decision tree:

pts in [20-38]:
:.s in [5-6]:
: .pts in [25-38]: 2S (390)
:  pts in [0-24]:
:  : :.pts in [0-21]:
:  :   :.s in [1-5]:
:  :   : :.d in [5-6]: 2S (4)
:  :   : : d in [1-4]:
:  :   : : :.h in [5-6]: 2S (4)
:  :   : :   h in [1-4]:
:  :   : :   :.h in [1-2]:
:  :   : :     :.d in [1-2]:
:  :   : :     : :.d = 1: 2S (2)
:  :   : :     : : d in [2-6]:
:  :   : :     : : :.h = 1: 2S (2)
:  :   : :     : :   h in [2-6]: 1S (2)
:  :   : :     : d in [3-6]:
:  :   : :     : :.h = 1: 1S (4)
:  :   : :     :   h in [2-6]:
:  :   : :     :   :.d in [1-3]: 2C (2)
:  :   : :     :     d in [4-6]: 1S (2)
:  :   : :     h in [3-6]:
:  :   : :     :.h in [4-6]: 1S (6)
... (5 pages)
```

And the following set of rules (my comments on right of the *):

```
Rule 1: (512, lift 3.9)
        pts in [0-3]
        -> class pass  [0.998]
* Pass with a very weak hand

Rule 2: (132/2, lift 3.8)
        pts in [0-11]
        s in [1-3]
        h in [1-4]
        d in [5-6]
        d in [1-5]
        -> class pass  [0.978]
* Pass even up to average and 5 diamonds
```

```
Rule 3: (39, lift 3.8)
        pts in [0-12]
        s in [4-6]
        s in [1-4]
        h in [2-6]
        h in [1-2]
        d in [1-3]
        -> class pass  [0.976]
* Pass with up to average and 4 spades
... (10 pages)
```

The program uses the original *bidsyst.txt* and it is doubtful if the tree or the rules can be useful to a human reader.

No opponents were taken into account except for the specified opening bids which included preemptive bids like 2 diamonds (showing a strong hand with diamonds or a weak hand with a six-card major) and 3 in a suit showing a weak hand with a six-card suit.

The database contains a vector with the actual number of tricks taken in any suit or no trumph. We have assumed East to be declarer in all cases due to performance reasons. The maximum number of bids in a row was set to four (a pass was automatically added as a fifth bid if applicable) and the number of possible bids was limited to sixteen.

Learning rates were encoded as integers and shifted right ten steps, effectively scaling them down with a factor of 1024. Exploration rates was integer coded in parts per thousand. A $\gamma$-value of zero was used and the initial values of the Q-tables for east and west respectively was a command line argument. The examples were separated in a training and a test set and training examples were selected in random order to avoid biasing effects. The desired number of examples were selected in order from the database except for not representable uncommon shapes which were simply skipped. Exploration was not used during test-evaluations but used during training evaluations.

The score was calculated from the number of tricks taken and final contract. The vulnerability was "none" and the dealer was always East. Undoubled part-scores, games, small-slams and grand slams was calculated and contracts going more than two down was assumed to be doubled.

We tried to find the best possible bidding system by doing three experiments; first a survey run with different parameters, a second survey with some new, better, parameter values and then a longer run with the parameters estimated to be best by the human observer. The parameters to be selected were the learning rate (common for both hands), exploration rate (also common) and initial values for the Q-tables.

In the first run we used the learning rates 5, 10 and 20, exploration rates 5, 10 and 20 and initial Q-values -100 for East and -200, 0 and +200 for West. Each run was repeated three times with different random number seeds (13579, 5799 and 8642 respectively). This resulted in a total of $3*3*3*3 = 81$ cases. Ten million training examples were selected in each epoch and 200 epochs was run in each case. Each case took about 41 minutes and the total CPU-time was about 55 hours on a 1.5 GHz Intel Pentium IV running Windows 2000 and equipped with 512 MByte memory. Each training occasion required about one microsecond including evaluation. Every

| | $\epsilon = 0.005$ | $\epsilon = 0.010$ | $\epsilon = 0.020$ |
|---|---|---|---|
| **Lr = 5/1024** | $27.2 \pm 8.7$ | $27.6 \pm 0.2$ | $30.7 \pm 9.9$ |
| **Lr = 10/1024** | $24.9 \pm 1.8$ | $40.4 \pm 6.1$ | $41.7 \pm 3.7$ |
| **Lr = 20/1024** | $33.8 \pm 3.6$ | $50.1 \pm 10.0$ | $63.6 \pm 3.1$ |

Table 4: The average of the last ten test-evaluations, averaged over three runs. q0 is zero

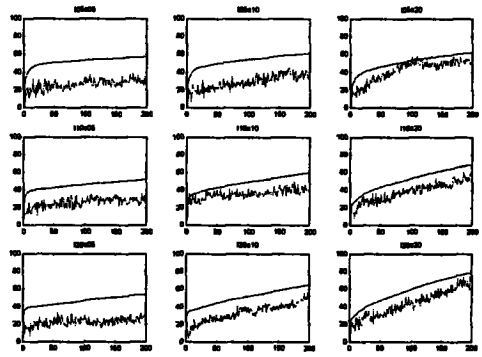

Figure 5: Runs with Q0 = -200. The smooth curve is for the training set and the rugged curve the test-set. 200 epochs, each 10 miljon tries, were run and repeated three times to be averaged into these curves.

99th epoch, a bidding system was emitted into two files; responses.txt (115 kB) and rebid.txt (525 kB), the second response was not stored. The generation of the rebid-file takes the opening restrictions into consideration to reduce the size. A log-file (6 kB) with the time, average training-sum and average test-sum was generated and the standard output was redirected into a file (1.4 MB) with the actual bidding sequences from all the test-cases for future reference. About 2MB*81=162MB of disk space were used.

550,000 training examples and 200 test examples (positioned from 710,000 and forward) were used. 109,940 (16.7%) were skipped, reasonably close to the assumption above (83% kept).

## Selecting Learning Rate, Exploration and Initial Q-value

The first run was made with a $q_0$ of 0. The results are shown in Table 4. The highest value for both of them resulted in the best performance; a average value of $63.6 \pm 3.1$ with an learning rate of 0.019 and an exploration of 0.020, denoting a 2% risk of selecting a random bid. The experiments were rerun with a $q_0$ of +200, meaning a untested alternative is worth 200 points. Because most bridge-results are in the $-100..140$ region, this is the optimistic approach resulting in an (more) exhaustive search and slightly worse results.

A third run with $q_0 = -200$ where done with better results. This value (-200) results in less search of alternatives because a reasonable result like -100 is preferred to untested alternatives. The training and test-curves (Fig.5) seems to

| | $\epsilon = 0.005$ | $\epsilon = 0.010$ | $\epsilon = 0.020$ |
|---|---|---|---|
| **Lr = 5/1024** | $29.3 \pm 5.1$ | $36.7 \pm 13.6$ | $51.4 \pm 6.9$ |
| **Lr = 10/1024** | $28.4 \pm 5.6$ | $40.9 \pm 6.2$ | $54.6 \pm 3.2$ |
| **Lr = 20/1024** | $27.2 \pm 2.4$ | $48.3 \pm 8.3$ | $68.7 \pm 4.9$ |

Table 5: The average of the last ten test-evaluations, averaged over three runs. q0 is -200

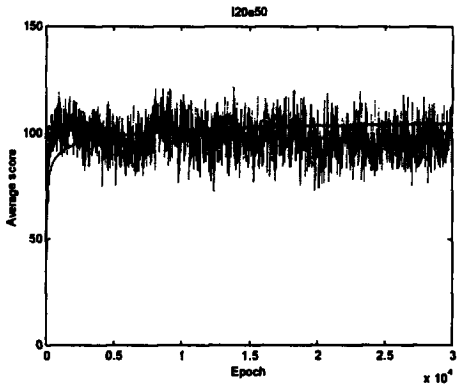


Figure 6: A 5-day CPU run of learning. Lr is 20/1024 , $\epsilon = 0.050$ and $q_0 = -200$.The maximum training case has a score of 104.89, max test-score is 121.7 and average on the last ten is 94.65.

follow each other but at a different level. The test-curve is almost always below the training curve although no exploration is used. The training-curve is also more smooth due to the large number of examples (10 million, some identical). The best result (Table 5; $68.7 \pm 4.9$) was also obtained with the highest parameter values; $Learning\_rate = 20/1024 \approx 0.019$ and $\epsilon = 0.020$.

A long, single run with lower learning rate but high exploration is shown in Figure 6. The levels seems to more or less stabilize around 100.

Some of the interesting examples (many removed) from the bidding are presented in table 6.

At a later position (1000) under the same training conditions, a bidding system was emitted. Only a very small part of it (part of the responses when partner has started with a Pass) is presented in table 7. It is illustrative because it indicates what suit to open with and which strengths are associated with a particular shape. Many cases are innovative and maybe unknown to most human bridge-bidders. Some examples are: opening with only seven h.c.p. when having six spades, two hearts and a four card minor (club or diamond) suit, opening with the lower of adjacent suit if weak (then pass) or strong (continue bidding) and higher suit if medium strength, pass if balanced and less than fourteen h.c.p., open one or three NT with strong hands and any shape, 2 hearts and spades(stronger) shows a good hand with eight+ cards in the majors, open with nine h.c.p. and unbalanced with six-card major or diamonds or when having a five-five shape,..

| Deal | East | Hp | West | Hp | E | W | E | W | Score |
|---|---|---|---|---|---|---|---|---|---|
| 2. | 2614 | 16 | 2353 | 15 | 1H | 3N | 4N | 7H | 1510 |
| 5. | 2434 | 9 | 5323 | 14 | P | 1S | 2C | 2H | 110 |
| 6. | 4522 | 10 | 1255 | 13 | P | 1C | 1S | 2C | 110 |
| 7. | 3145 | 7 | 2434 | 15 | P | 1H | 1N | P | 90 |
| 13. | 1543 | 9 | 5332 | 16 | P | 1S | 2C | 3N | 400 |
| 16. | 3514 | 12 | 3253 | 14 | 1H | 3N | P | | 430 |
| 24. | 3334 | 6 | 3343 | 14 | P | 1D | P | | -50 |
| 26. | 5152 | 4 | 3613 | 10 | P | 1H | P | | 110 |
| 27. | 4243 | 2 | 2641 | 22 | P | 3N | P | | -100 |
| 28. | 2434 | 12 | 4243 | 12 | 1H | 3N | P | | -50 |
| 29. | 3613 | 13 | 3262 | 15 | 1H | 3N | P | | 520 |
| 33. | 5134 | 6 | 3622 | 11 | P | 1H | 1N | 2H | -50 |
| 34. | 3514 | 11 | 4135 | 13 | P | 1C | 1N | P | 150 |
| 42. | 3334 | 10 | 5512 | 13 | P | 1H | 1S | 4S | 480 |
| 52. | 2623 | 10 | 2236 | 7 | 2D | 2H | P | | 110 |
| 53. | 2254 | 13 | 2632 | 10 | 1D | 3N | P | | 400 |
| 54. | 4342 | 2 | 3514 | 11 | P | 1C | 1D | 1H | -50 |
| 56. | 3163 | 12 | 5422 | 15 | 1D | 3N | P | | 430 |
| 60. | 4144 | 0 | 3325 | 17 | P | 1N | 2H | 2S | -100 |
| 64. | 4441 | 9 | 5431 | 11 | P | 1S | 1N | 2S | 110 |

Table 6: Some interesting examples of bidding from a reasonably good run with Lr = 0.02 and exploration set to 0.02, q0 is set to -200. An average score of 87 on the training set and 100 on the test set is achieved after 562 epochs.

Response to P:

| | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 18 | 20 | 22 | 25 | 28 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3613 | 1H | 1H | 1H | 1H | 1H | 1H | 1H | 1H | 2C | 2D | ? | ? |
| 3541 | P | 1H | 1H | 1H | 1H | 1H | 1H | 3D | 2S | 2H | 2S | ? |
| 3532 | P | P | P | P | 1H | 1H | 1H | 1N | 3N | 3N | 2C | ? |
| 3523 | P | P | P | 1H | 1H | 1H | 1H | 1H | 3N | 2C | 2H | 3C |
| 3514 | P | 1C | 1H | 1H | 1C | 1H | 1H | 1H | 3N | 3N | 3C | ? |
| 3451 | P | 1D | 1D | 1D | 1D | 1D | 1D | 1N | 2H | 3D | ? | ? |
| 3442 | P | P | P | P | 1D | 1D | 1H | 1N | 3N | 3N | 3N | ? |
| 3433 | P | P | P | P | 1C | 1H | 1H | 1N | 1N | 3N | 2H | 3N |
| 3424 | P | P | P | 1C | 1C | 1C | 1H | 1N | 3N | 3N | 3N | ? |
| 3415 | P | 1C | 1C | 1C | 1H | 1C | 1C | 1H | 1N | 3N | ? | ? |
| 3361 | P | 1D | 1D | 1D | 1D | 1D | 1D | 1N | 1N | 3N | 3S | ? |
| 3352 | P | P | 1D | 1D | 1D | 1D | 1N | 1D | 3N | 3N | 2D | ? |
| 3343 | P | P | P | P | 1D | 1D | 1D | 1N | 3N | 3N | 3N | 2S |
| 3334 | P | P | P | 1C | 1C | 1C | 1C | 1N | 3C | 3N | 3N | ? |
| 3325 | P | P | 1C | 1C | 1C | 1C | 1C | 1N | 3N | 3N | 3N | ? |
| 3316 | 1C | 1C | 1C | 1C | 1C | 1N | 1C | 1C | 3H | 3N | ? | ? |
| 3262 | P | P | 1D | 1D | 1D | 1D | 1N | 3N | 3N | 3N | 2N | 3H |

Table 7: Some of the responses when partner has started with pass, showing a weak hand without extreme shape. Question-mark indicate unknown case (no example).

| Epoch | Training-level | Lr | Difference |
|---|---|---|---|
| 849 | 84.7 | 6/1024 | 170.7 |
| 951 | 83.4 | 5/1024 | 204.8 |
| 1074 | 79.1 | 4/1024 | 256 |
| 1221 | 67.3 | 3/1024 | 341.3 |
| 1413 | 45.7 | 2/1024 | 512 |
| 1683 | 35.6 | 1/1024 | 1024 |

Table 8: Integer round-off error

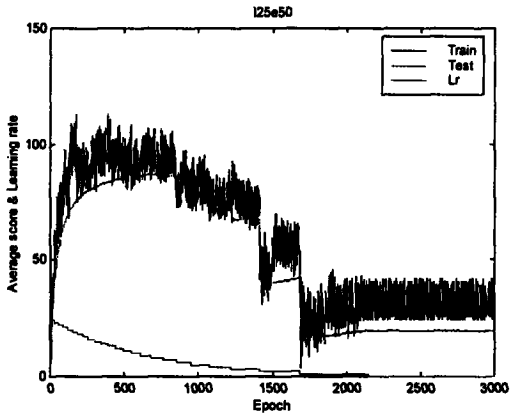


Figure 7: The scores are actually decreasing step-wise when the learning rate reduces.

### Reducing the Learning Rates

The single learner Q-learning theory says that convergence is assured if the learning rate is reduced slowly enough. An attempt to improve the average score is done by reducing the learning rate with a decay factor. In our cases, the learning seems to drop off at about epoch 150. By setting the learning rate to 20/1024 at this point a experiments was run with a slow decay (Figure 7). There are jumps in the performance and if viewed in detail those jumps are positioned according to table 8. When the learning rate steps down to small values, the required difference between score and q-value has to be at least one when scaled with the learning rate according to Equation 1. The first noticeable effects appear when the learning rate is $6/1024 \approx 0.006$ and the required difference is then $1024/6 \approx 170.7$. In bridge terms this means that the learner is not able to see the difference between part-scores and concentrates on the large contracts like games and slams. When learning rate gets below 3/1024, the game-contract abilities are also lost. This problem with learning could be avoided by using floating-point representation but this consumes more memory (4 times in this case; $384 * 4 = 1536$ MB) and requires more computing time.

## Conclusions and Further Work

The runs shows it is possible to find a reasonable bidding system in a limited time. A relatively high learning rate combined with high exploration resulted in the best performance. The test-set evaluations are quite close to the training evaluations, indicating that no over-fitting is achieved yet. An initial $q_0$ value set to a reasonable lower acceptable threshold seem to steer the learning into realistic areas. Good runs with decreasing learning rate was not possible to achieve due to the integer nature of this implementation.

The resulting bidding system, studied in detail is also interesting. The responder have learnt, "understood", that an opener shows hearts or spades or a strong hand with the two diamond opening bid and thereby never passes and do not bid two hearts with heart support. The no-trump bidding is also interesting; two-bid responses seems to be sign off, two NT is never used and three in a suit seems to be highly conventional and never passed by the NT-opener. Three NT is often the response with medium strength hands and not an extreme shape and four in suit is a transfer bid. Maybe an interesting NT-bidding part of the system can be developed if the learning were concentrated on this?

Anyhow; the learnt system is not practically useful until the defensive bidders are put into consideration. More information will be available in my Licentiate thesis (ongoing work) including a Genetic approach.

## References


Andrews, T. 1999. Double dummy bridge evaluations. http://www.best.com/ thomaso/bridge/valuations.html.

Björn Gambäck, M. R., and Pell., B. 1993. Pragmatic reasoning in bridge. Technical Report 299, University of Cambridge, Computer Laboratory, Cambridge, England.

Carley, G. 1962. A program to play contract bridge. Master's thesis, Dept. of Electrical Engineering, Massachussets Institute of Technology,Cambridge,Massachussets.

Frank, I. 1998. *Search and Planning Under Incomplete Information.* London: Springer-Verlag.

Ginsberg, M. L. 1999. GIB: Steps toward an expert-level bridge-playing program. *Proceedings from Sixteenth International Joint Conference on Artificial Intelligence* 584–589.

Hillyard, R. 2001. Extending the law of total tricks. *http://www.bridge.hotmail.ru/file/TotalTricks.html.*

Lindelof, E. 1983. *The Computer Designed Bidding System - COBRA.* Number ISBN 0-575-02987-0. London: Victor Gollancz.

MacLeod, J. 1991. Microbridge - a computer developed approach to bidding. *Heuristic Programming in AI - The First Computer Olympiad* 81–87.

Prorok, K. 1999. Gubben i lådan (in swedish). *Svensk Bridge* 31(no 4, october):26–27.

Stanier, A. 1977. *Decision-Making with Imperfect Information.* Ph.D. Dissertation, Essex University.

Sterling, L., and Nygate, Y. 1990. Python: An expert squeezer. *Journal of Logic Programming* 8:21–40.

Watkins, C. J. C. H., and Dayan, P. 1992. Q-learning. *Machine Learning* 8:279–292.

Winston, P. H. 1977. *Artificial Intelligence.* Addison-Wesley.