Contents

Chapter 1

1X

1.1   1C

```
1C -
1D*: 4+H
1H*: 4+S
1S*: inv+, 4+D; or 5+D
   - 1N : min, 2-D or 4333 or some 4332
   - 2C : min unbal
   - 2D : 3+D
1N : 6-10
2C*: GF, 4+C  # inv if PH
   - 2D*: 12-14
   - 2M : 15+, nat, usually 3+M
   - 2N : 18-19
   - 3C : 15-17, 6+C
   - 3X*: spl
2D*: inv, 4+C  # constr if PH
   - 2M : 15+, nat, usually 3+M
   - 2N : nat min
   - 3C : nat min
   - 3X*: spl
   - 3N : nat  # 13-14/18-19
2M : 3-7, 6+M  # not interested in 16-17, 2-M
2N*: 7-9, 5+C
3C*: 0-6, 5+C
3X : 7+X, about 6-6.5 tricks
```

1.1.1   Gazzilli after 1D*/1H*

```
1C - 1D*;
1H*: 4+S  # highest priority unless 4+ fit
1S*: 16+; or 13-14(15), 6+C, GT  # may be 16-17, 2425 [3H]
   - 1N*: 8+
        - 2C*: 6+C, GT
             - P  : s/o
             - 2N : re-inv
             - 3C : re-inv
   - 2C : min
   - 2D : min
   - 2H : min
   - 2S*: min, 5+S  # implies 6+H
   - 2N+: C fit ST
   - 3H : solid H ST
1N : min bal
2C : min unbal
2D*: 14-16, 6+C, 4+D
```

```
2H : min, 4+H
2S*: inv+, 4+H, spl S
2N : 18-19, bal
3C : (15)16-17, 6+C, GT
3D*: inv+, 4+H, spl D
3H*: 18-19 bal, 4+H
3S*: 20-22, 4+H, some spl
    - 3N*: ask
3N*: GF, 4+H, S void
4C*: GF, 4+H, 6+C
4D*: GF, 4+H, D void
4H*: 4+H, 20-21 bal  # or good 18-19 bal
```

```
1C - 1H*;
1S*: 16+; or 13-14(15), 6+C, GT  # may be 16-17, 4225 [3S]
    - 1N*: 8+
        - 2C*: 6+C, GT
            - P  : s/o
            - 2N : re-inv
            - 3C : re-inv
    - 2C : min
    - 2D : min
    - 2H : min
    - 2S : min
    - 2N+: C fit ST
    - 3S : solid S ST
1N : min bal
2C : min unbal
2D*: 14-16, 6+C, 4+D
2H*: 14-16, 6+C, 4+H
2S*: min, 4+S
2N : 18-19, bal
3C : (15)16-17, 6+C, GT
3D*: inv+, 4+S, spl D
3H*: inv+, 4+S, spl H
3S*: 4+S, 18-19 bal
3N*: 20-22, 4+S, some spl
    - 4C*: ask
4C*: GF, 4+S, 6+C
4D*: GF, 4+S, D void
4H*: GF, 4+S, H void
4S*: 4+S, 20-21 bal  # or good 18-19 bal
```

1.1.2 XYZW

```
1C - 1D*; 1H*- 1S*: see utils/XYZW/PLOB
```

```
1C - 1DH; 1N - ...: see utils/XYZW/2wPCB
1C - 1S ; 1N - ...: no 2-way
```

1.2   1D

```
1D -  # 11-21, 4+D unbal; 12-14/18-19, 4+D bal
1M : 6+, 4+M
1N : 6-10, nat
2C : inv, 6+C [3C]; or GF, 5+C
   - 2D*: 12-14  # may be 12-14 with good C support; ready to play 3N
        - 2H*: GF, ask
             - 2S*: 5D unbal
             - 2N : min bal
             - 3C : 3+C unbal
             - 3D : 6+D unbal
             - 3N : max bal
        - 2S+: nat GF
        - 3C : nat inv
   - 2M*: 15+ unbal, may be stopper
   - 2N : 18-19 bal
   - 3C : GF, 14+, 4+C
   - 3D : solid D ST
   - 3M*: GF, 4+C, spl
2D*: some 6+M pre  # shouldn't be interested in 18-19 bal
   # resp. similar to 2D/V (multi) except 3D
   - P  : s/o
   - 2H*: P/C
   - 2S*: P/C, may be interested in H
   - 2N*: ask  #
        - 3C*: min
             - 3D*: ask
             - 3H*: P/C
        - 3D*: max, H
        - 3H*: max, S
   - 3C : nat, NF
   - 3D : nat, inv  # then 3M = max and show suit
2H*: GF, 4+D  # inv if PH
   - 2S*: 15+, catchall  # GF if partner inv
   - 2N : 12-14 bal
   - 3C : nat
   - 3D : min unbal
   - 3M*: spl
   - 3N : 18-19 bal
2S*: inv, usually 4+D, not prefer to declare  # constr if PH
   - 2N : nat min
   - 3C : nat, F1
   - 3D : nat min
   - 3M*: nat, could be stopper
   - 3N : nat
2N : nat inv
```

```
3C*: 8-10, 4+D
3D*: (0)3-7, 4+D
3M : 7+M, about 6 tricks, NF
```

1.2.1 XYZW

```
1D - 1H; 1S - 2C*: see utils/XYZW/PLOB
1D - 1M; 1N - ...: see utils/XYZW/2wPCB
```

1.3 1H

```
1H -  # 12-21, 5+H
1S*: F1; 4-6, 3+H [2H]; inv, 3H [3H/4H]; constr. 5+H [4H]; 5-11, 2-S;
   12-14, 4-S bal [3N]
1N*: 5+S, F1
2C : GF, nat
2D : GF, nat
2H : 7-9, 3+S
2S*: 12+, 4+H
2N*: 8-9/12-13, C/S spl
   - 3C*: ask
       - 3D*: C spl
       - 3H*: S spl, 8-9
       - 3S+: S spl, 12-13, cue-bid
3C*: 10-11, 4+H bal; 10-11, 4+H spl  # the latter is GF
   - 3D*: ask, ST
       - 3H*: 10-11, 4+S bal
       - 3S+: spl  # 3N* replaces the highest spl, i.e. D
   - 3H : light opening; or min, 5H(332)
3D*: 8-9/12-13, D spl
3H : pre
3S*: 10-12, S void
3N*: 12-14, 3H, bal w/o 6m, CoG
4X*: 10-12, X void
4H : pre
```

1.3.1 Gazzilli after 1S*/1N*

```
1H - 1S*: F1
1N*: 16+, any; or 12-15, 6+H
2m : 12-15, nat
2H*: 12-15, 4+S
2S*: 14-16, 6+H, 4+S
2N*: 14-16, 6+H, 4+m
   - 3C*: ask
       - 3D : 4+D
       - 3H*: 4+C
3m : 14-16, 5+m, NF
3H : 15-17, (s-)solid 6+H
3S : inv, 5+S, NF
3N*: GF, solid 7+H
```

```
1H - 1N*: 5+S
2C*: 16+, any; or 12-15, 6+H
2D : 12-15, nat
```

```
2H*: 12-15, 4+C
2S : 12-15, 3+S
2N*: 14-16, 6+H, 4+X
   - 3C*: ask
        - 3D : 4+D
        - 3H*: 4+C
        - 3S : 4+S
3m : 14-16, 5+m, NF
3H : 15-17, (s-)solid 6+H
3S : inv, 6+H, 3+S
3N*: GF, solid 7+H
```

1.3.2   1H - 2S* (same as 1S - 2N*)

1.3.3   1H - (X)

```
1H - (X);
P  : weak or some unbal inv  # then bid/X = inv
XX : 10+, s-bal, usually not concentrated  # forcing pass on
1S : 5+S, F1  # system on, may be psyche but rare
1N*: 5+C, 6-10; or 4+C, GF
2C*: 5+D, 6-10; or 4+D, GF
2D*: 3+H, 4-6 or 10+
2H : 3+H, 7-9
2S+: system on
```

1.3.4   PH response

```
1H -  # 12-21, 5+H
1S : 4+S
   - 1N : min nat
   - 2C*: 16+, any; or 12-15, 6+H
        - 2D*: 8+
   - 2D : 12-15, nat
   - 2H*: 12-15, nat C
1N*: NF; 4-6, 3+H [2H]; 5-11, 2-S
   - 2C*: 16+, any; or 12-15, 6+H
        - 2D*: 8+
   - 2D : 12-15, nat
   - 2H*: 12-15, nat C
2C*: inv+, 3+H  # Drury
   - 2D*: min bal
   - 2H : light opening
   - 2S+: same as after 1H - 2H
2D : inv+, nat
```

```
2H : 7-9, 3+H
2S : 4-7, 6+S
2N*: 8-10, 4+H, some spl
3m*: 8-10, (54+) H and m
3H : pre
3S*: 10-12, S void
3N*: 12-14, 3H, bal w/o 6m, CoG
4X*: 10-12, X void
4H : pre
```

1.4 1N

temporarily use BTUBWS

1.4.1 comp

```
1N - (2C = nat or some 6+X) - X* : Stayman   # others system on
1N - (2D = nat) - X* : t/o
1N - (2M = nat) - X  : pen
# otherwise UwU
```

1.5 1S

```
1S -  # 12-21, 5+S
1N*: F1; 4-6, 3+S [2S]; inv, 3S [3S/4S]; constr. 5+S [4S]; 5-11, 2-S;
    12-14, 2-S bal [3N]
2C : GF, nat
2D : GF, nat
2H : GF, nat
2S : 7-9, 3+S
2N*: 12+, 4+S
3C*: 8-9/12-13, C/D spl
   - 3D*: ask
       - 3H*: C spl
       - 3S*: D spl, 8-9
       - 3N+: D spl, 12-13, cue-bid
3D*: 10-11, 4+S bal; 10-11, 4+S spl  # the latter is GF
   - 3H*: ask, ST
       - 3S*: 10-11, 4+S bal
       - 3N+: spl  # 3N* replaces the highest spl, i.e. H
   - 3S : light opening; or min, 5(332)
3H*: 8-9/12-13, H spl
3S : pre
3N*: 12-14, 3S, bal w/o 6m, CoG
4X*: 10-12, X void
4S : pre
```

1.5.1 Gazzilli after 1N

```
1S - 1N*: F1
2C*: 16+, any; or 12-15, 6+S
2D : 12-15, nat
2H : 12-15, 4+H
2S : 12-15, 4+C
2N*: 14-16, 6+S, 4+X
   - 3C*: ask
       - 3X : 4+X
       - 3S*: 4+C
3X : 14-16, 5+X, NF
3S : 15-17, (s-)solid 6+S
3N*: GF, solid 7+S
```

```
1S - 1N*; 2C*: 16+, any; or 12-15, 6+S
2D*: 8+, any
   - 2S : 12-15, 6+S
   - other : nat and GF
2H : 5-7, 5+H
```

```
2S : 4-7
# bids below applies to both 2H and 2S
   - 2N*: GF, some 6+S
        - 3C*: ask
              - 3X : 4+X
              - 3S*: 4+C
              - 3N : no 4-card suit
   - 3X : GF, 5+X   # except raise = inv
   - 3S : inv, 6+S
2N+: 5-7, nat   # 2N usually mms
3S : inv, 3S
4S : nat
```

1.5.2   1S - 2N*

```
1S - 2N*: 12+, 4+S
3X : X short   # then 3N = FF
3S : 14+, w/o shortness   # then 3N = FF
3N*: FF, 12-13, w/o shortness
(TBD) after 3S/3N, choose from spl or cue-bid
# 4X usually with controls in two other suits, otherwise bid 3X instead
4m : 5+m
   - +1*: ask
         - 4S : min
4H : 5+H, extra
4S*: 5+H, min
```

1.5.3   1S - (X)

```
1S - (X);
P  : weak or some unbal inv   # then bid/X = inv
XX : 10+, s-bal, usually not concentrated   # forcing pass on
1N*: 5+C, 6-10; or 4+C, GF
2C*: 5+D, 6-10; or 4+D, GF
2D*: 5+H, 6-10; or 4+H, GF
2H*: 3+S, 4-6 or 10+
2S : 3+S, 7-9
2N+: system on
```

1.6   2C

```
2C*-
2D*: waiting
   - 2H*→2S*- 2N : 20-21 bal
              - 3X : 22+, nat, 5+H
              - 3N : 22-24, 5H(332)
        - 2N*: 4-7, 5+S
   - 2S : 22+, 5+S
   - 2N : 22-24
   - 3m : GF, 5+m
   - 3M : nat solid ST
   - 3N : 25-27
   - 4m : nat solid ST
   - 4M : 8.5+ tricks
   - 5m : 8.5+ tricks
2M : 8+, 5+M
2N : 8-10
3m : 8+, (5)6+m
3M : 4-7, 6+M
3N : 11-12
4m : 8+, solid (6)7+m, ST
4M : 4-7, 7+M
```

Chapter 2

pre

2.1 2D/NV (multi)

```
2D - # 3-9, 6+M
P  : s/o
2M/3M/4H*: P/C
4S : s/o
2N*: F1
   - 3C*: min
         - 3H*: P/C
   - 3D*: H, mid
   - 3H*: S, mid
   - 3S*: H, max
   - 3N*: S, max
3C : s/o
3D*: some 5+M, GF
   - 3H : 6+H
         - 3S : 6+S
         - 3N : 5S
   - 3S : 6(2-)xx
   - 3N*: 6322
   - 4C*: 63xx+, spl D
   - 4D*: 63xx+, spl C max
   - 4H*: 63xx+, spl C min
3N : s/o
4C*: fit in both majors, slam interest, ask
   - 4D*: H
   - 4H*: S
4D*: s/o 4M, not allowing re-raise
```

2.2 2H/NV (Ekren)

```
2H*- # 3-8, 4+MMs
2S : s/o
2N*: ask
   - 3C*: min
      - 3D*: ask, GF
            - 3M*: 5+oM
            - 3N : 44Ms
            - 4m*: 55Ms, spl m
   - 3D*: mid+, 55Ms
       - 3M : s/o
   - 3M*: max, 5+oM
   - 3N*: max, 44Ms
3C*: 4+H, inv+
   - 3D*: re-inv
```

```
    - 3H*: min
    - 3S*: shape-FF, 4H but max
    - 3N+: 5H, super max, cuebid
    - 4H : 5H, max, not super max
3D*: 4+S, inv+
    - 3H*: re-inv
    - 3S*: min
    - 3N*: shape-FF, 4S but max
    - 4C+: 5S, super max, cuebid
    - 4S : 5S, max, not super max
3M : s/o
3N : s/o
4C*: ask, ST
4D*: pick a 4M, not allowing re-raise
4M : s/o
```

2.3 2S/NV (wild)

```
2S - # 0-8 (11 if partner PH), 5+S
2N*: inv+
    - 3m : 3+m, min
    - 3H*: 6+S, max
    - 3S*: 6+S, min
    - 3N*: 5S, max
3C*: 5+H, GF; 2(5+)xx, inv
    - 3D*: 5(1-)xx, min; 52xx, inv; 5(2-)xx, max  # then rebid 3N/4H
        - 3H : 26xx, inv
        - 3S : s/o  # usually 25xx inv
        - 3N : s/o
    - 3H*: 5(2+)xx, min
    - 3S*: 6(2-)xx, min
    - 3N*: 6(2-)xx, max
    - 4C+: 3+H, max
3D*: 16xx, inv; 6+H, GF
3H*: 3+S, inv+
3S : s/o
3N : s/o
4C*: ORKC
4D*→4S : re-raise not allowed, usually with strength  # could be preempt
4H : s/o
4S : s/o
```

2.4   2D/V (multi)

```
2D - # 3-7, 6+M
P  : s/o
2M/3M/4H*: P/C
4S : s/o
2N*: F1
   - 3C*: min
         - 3H*: P/C
   - 3D*: H, mid
   - 3H*: S, mid
   - 3S*: H, max
   - 3N*: S, max
3C : s/o
3D*: some 5+M, GF
   - 3H : 6+H
         - 3S : 6+S
         - 3N : 5S
   - 3S : 6(2-)xx
   - 3N*: 6322
   - 4C*: 63xx+, spl D
   - 4D*: 63xx+, spl C max
   - 4H*: 63xx+, spl C min
3N : s/o
4C*: fit in both majors, slam interest, ask
   - 4D*: H
   - 4H*: S
4D*: s/o 4M, not allowing re-raise
```

2.5   2M/V (nat)

```
2M - # 8-11, 6+M
2S : nat F
2N*: ask feature
   - 3X : cue
3M : s/o
3N : s/o
4C*: ORKC
4D*: s/o 4M, not allowing re-raise
4M : s/o
```

Chapter 3

comp

3.1 Gladiator

```
(1M) - 1N - (P) -
2C*→2D*: tranfer accepted
        - 2M*: GF, 4+oM
        - 2oM: s/o
        - 2N*: inv
        - 3C : s/o
        - 3X : 6+X inv
        - 3M*: GF, 4oM, short in M
        - 3N*: 54+mms CoG, usually extra
2X : inv, 5+X
2M*: inv, 4+oM
2N : inv, 5+C, NF  # because may not able to pen. 2M
3X : GF, 5+X
3M*: GF, w/o 4oM, short in M  # 2C* then 3M is 4oM
```

```
(1m) - 1N - (P) -
# similar, but D is the cuebid suit
2C*→2D*: tranfer accepted
        - 2M : s/o
        - 2N*: inv
        - 3C : s/o
        - 3X : 6+X inv
        - 3N*: 54+mms CoG, usually extra
2D*: inv+, Stayman
2M : inv, 5+M
2N : inv, 5+C, NF
3X : GF, 5+X
```

3.2   good bad 2NT

```
1X - (1Y)- [bid/X] - (2Y)
2N*→3C*: comp
3X : good hand  # if [bid/X] shows inv+ then GF, otherwise inv
```

```
# other situations that applies
 1X -(P) - 1Y -(2M = jump overcall) - 2N*: good-bad
 1X -(2Y)- X* -(P) - 2N*: good-bad
(1X)- 1Y -(2X)- 2N*: good-bad
(1X)- X* -(2X)- 2N*: good-bad
(1X)- 2Y -(2X)- 2N*→3C*: tr. Leb.  # notice the difference!
```

3.3   overcall

```
(1m) -
# style: the stronger the hand, more shapely to pre 2M

# V/NV
P  : # may be lengthy but not enough values, ex: AJxxxxx.xxx.xx.x
1M : (8)9-17, 5+M  # rebid 2M = 14-17
2M : 9-13(14), 6+M  # promise defensive values

# V/V
1M : 8-17, 5+M  # could be weaker (LD) if partner PH
2M : 6+M, pre  # usually not 6322; unless ex: KQJ9xx.xx.Kxx.xx

# NV/NV
1M : 8-17, 5+M  # 4oM or afraid of losing game
2M : (5)6+M, pre  # usually unbal and partner PH if 5M
# KQxxx.JTxx.x.Qxx: 2S
# KJxxx.xxx.QJTx.x: 1S/2S
# KQJxx.xxx.Qxx.xx: 1S/1-2S

# NV/V
1M : 8-17, 5+M  # 4oM or afraid of losing game
2M : 5+M, pre  # wild if partner PH
# KQJxx.xxx.Qxx.xx: 1S/2S
# KJxxx.xx.xxx.xxx: P/2S
```

3.4   unusual

```
# if we are able to overcall the higher suit at 1-level, then range = 3-10
   or 16+; otherwise 10+
(1M) - 2M*: 5+oM, 5+m
(1M) - 2N*: 55+ms
(1m) - 2m*: 55+Ms; 54+ if NV/V
(1m) - 2N*: 5+H, 5+om
```

```
(1X) - (1Y) -
X*  : 44+ unbid suits or strong
1N*: 54+ unbid suits
```

```
(1X)- 2m -(2X)- P  -(P) - 2N*: 64+ mms
```

3.5   vs 1N

```
(1N)-
# case 1: non-pass
X  : 15+
2C*: H and another suit, 54+ either-way  # if MM then should be 54xx+
   - 2D*: P/C  # ask the 5-card suit
2D*: S and another suit, 54+ either-way  # if MM then should be 45xx+
   - 2H*: P/C
2M : nat, (5)6+M
2N*: 55(54)+mms
3X : nat, comp
# case 2: PH
X* : S and another suit, 54+ either-way
   - 2C*: P/C
2m*: H and m, 54+ either-way
2M : nat, (5)6+M
2N*: 55(54)+mms
3X : nat, comp
```

3.6   vs pre

```
1m - (3M) - X*: ask for stopper; may have 4oM
1m - (3M) - P - (P) - X*: ask for stopper; may have 4oM
# resp: pen with KJxx+ (IP) or equivalent values, otherwise
# 5 good oM [3/4oM] > 1.5 stop [3N] > 4 good oM [3/4oM] > 1 stop [3N]
```

Chapter 4

util

4.1   Forcing Pass

4.1.1   XX = Q

4.2 suit GT

```
[fit in 2S] -
2N*: HSGT or ST, no shortness
    - 3X*: Qxx or xxx in X  # partner having ctrl/xx is consider helped
    - 3S : min
    - 3N+: max  # may need to cuebid since partner may want to ST
3S : s/o
# case 1: bidder only shows one suit
3C*: C spl GT; or H spl GT (lo); or C spl ST
    - 3D*: inv C spl
        - 3H*: H spl (lo)
        - 3S : C spl (lo)
        - 3N+: C spl ST
        - 4S : C spl (hi)
    - 3H*: inv H spl
3D*: D spl GT or ST
    - 3H*: inv
3H*: H spl GT (hi) or ST
# case 2: if bidder already shows another suit X, spl X replaced by long X
```

```
[fit in 2H] -
2S*: HSGT or ST, no shortness
    - 2N+: Qxx or xxx in X  # partner having ctrl/xx is consider helped
    - 3H : min
    - 3S+: max  # may need to cuebid since partner may want to ST
3H : s/o
# case 1: bidder only shows one suit
2N*: S spl GT; or D spl GT (lo); or S spl ST
    - 3C*: inv S spl
        - 3D*: D spl (lo)
        - 3H : S spl (lo)
        - 3S+: S spl ST
        - 4H : S spl (hi)
    - 3D*: inv D spl (lo)
3C*: C spl GT or ST
    - 3S*: inv
3D*: D spl GT (hi) or ST
# case 2: if bidder already shows another suit X, spl X replaced by long X
```

4.3   Lebensohl

```
 1N -(2M)- 2N+: Leb
(2M)- X* - 2N+: Leb
 1m -(2M)- 2N+: Leb
```

```
Leb over (2M)-
2S : NF
2N*→3C*- P* : s/o
        - 3D*: s/o
        - 3oM: s/o
        - 3M*: ask stop
        - 3N : half stop
   - 3X : 18+, nat
3C*→3D*: inv+, 4+D
3D*→3oM: inv+, 5+oM
# 3M* depends on context:
# 1N -(2M): 4oM, GF
#(2M)- X* : 4oM, GF
# 1m -(2M): m fit, ask for stopper (since we have neg X)
3oM: 6+oM, inv  # 4oM if partner t/o
3N : s/o
```

```
1H -(2S)-
2N*→3C*- P* : s/o
        - 3D*: s/o
        - 3H : comp raise
        - 3S*: ask stop
        - 3N : half stop
   - 3X : nat 18+
3C*: inv+, 5+D
3D*: inv+, 3+H
3H : constr raise
3S+: spl
```

4.4   maximum X

```
fit in 2M - (opp. comp to 3X) -
# if X = M - 1
X*   : inv+  # allow pen with low probability
# otherwise
3M-1: inv+
```

```
(1m) - 2H [V/NV] - (3D) - X*   : 2+H, inv
(1m) - 2H [V/NV] - (3C) - 3D* : 2+H, inv
```

4.5   Rubens

```
(1X)- 1M - (P/X)-
XX : 10+, near bal
1N : nat
# if Y < X
2Y : 10+, nat, F1
# if Y >= X
2Y*→2Y+1: 10+, 5+[Y+1]   # if Y+1 = M then it means good raise
2X : comp. raise
```

```
(1H)- 2C - (P/X)-
XX : 10+, near bal
2D*: 10+, F1   # usually C fit or 5+D
2H*: 10+, 5+S, F1
2S*: 6-9, 5+S
2N+: nat
```

```
(1S)- 2C - (P/X)-
XX : 10+, near bal
2D*: 10+, 5+H, F1
2H : 6-9, 5+H
2S*: 10+, (3)4+C
2N+: nat
```

4.6 Slam bidding

4.6.1 cuebid

```
cue = 1/2nd ctrl
# if opener shows a suit (unless 1C - 1X; 1N/2N), then
opener's cue on that suit = 2 of AKQ, usually source of tricks
resp's cue on that suit = never shortness, can be Q
```

4.6.2 FF

```
[fit in 3M] -
4M  : min
3M+1*: FF, mild slam interest
4X*: cuebid, strong slam interest
```

4.6.3 kickback RKC

```
[fit in S] - 4N*: ask number of keycards   # 4 Ace + Trump K
5C*: 0/3 keycards
5D*: 1/4 keycards
5H*: 2/5 keycards w/ Trump Q
5S*: 2/5 keycards w/o Trump Q
# case 1: resp. didn't show another suit
5N*: 0/2/4 keycards, some void
   - 6C*: ask
       - 6X*: void in X
6X*: 1/3 keycards, void in X
# case 2: resp. already shows another suit, now only two possibilities
5N*: void in H (if resp. shows D) or D
   - 6C*: ask, no Trump Q
       - 6D*: 0/2/4 keycards w/ Trump Q
            - 6H*: inv
       - 6H*: 1/3 keycards w/ Trump Q
       - 6S*: w/o Trump Q
   - 6D*: want to inv 1/3 keycards
       - 6H*: extra but no Trump Q
       - 6S*: no extra
       - 7S : Trump Q, extra
   - 6H*: have Trump Q, want to inv 0/2/4 keycards
   - 6S*: s/o
6C*: 0/2/4 keycards, void in H (if resp. shows C) or C
   - 6D*: ask Trump Q
       - 6H*: Trump Q, no extra
```

```
        - 6S*: no Trump Q
        - 7S : Trump Q, extra
6D*: 1/3 keycards w/o Trump Q, void in C or H
   - 6H*: inv
6H*: 1/3 keycards w/ Trump Q, void in C or H, extra
6S*: 1/3 keycards w/ Trump Q, void in C or H
```

```
[fit in X] - [4X+1]*: ask number of keycards
# similar responses, 5N replaces void in [X+1]
```

4.6.4 ERKC

```
[fit in X] -
# if opener already shows non-void Y, then it replaces the highest ERKC
5Y*: ask number of keycards, excluding Y
   - +1*: 0/3 keycards
   - +2*: 1/4 keycards
   - +3*: 2 keycards
```

4.6.5 Obvious ERKC

```
[opp. bids Y (or bidder showed shortness in Y) and we fit in X] -
4X+2*: ask number of keycards, excluding Y
# 5Y replaces ERKC in X+2 or the highest ERKC (if X+2 is NT)
```

4.6.6 ORKC

```
preempt in X (not C) - 4C*: ORKC
4D*: min
4H+: same as resp. to RKC
```

4.6.7 2-suied RKC

```
1M - 2X; 3X - 3M; ... [4M+1]: 2-suited RKC
# Queen of M and X act as 0.5 keycards
+1*: 0/3/6 keycards  # may +0.5
   - +2*: ask if there's extra 0.5
        - 5M*: no
+2*: 1/4/7 keycards  # may +0.5
   - +3*: ask if there's extra 0.5
        - 5M*: no   # +4 = 5M
+3*: 2/5 keycards
```

`+4`*`: 2.5/5.5 keycards`

4.7   UwU

TBD (low-low, high-high)

4.8 XYZW

4.8.1 2wPCB

(https://www.ptt.cc/man/BridgeClub/D6D1/D49B/D130/M.924860463.A.html)

```
1X - 1Y; 1N
2C*→2D*: transfer accepted
        - P  : s/o
        - 2M : s/o, choose a partial [M <= Y]; inv, 5+Y, 4+M [M > Y]
        - 2N*: inv
        - 3Z : inv, 6+Z [Z = Y] or 4+Z [Z = X] or 5+Z and 4Y [otherwise]
        - 3N*: 5332, CoG  # different from BTUBWS
   - 2Y*: max, 3Y
2D*: GF, ask
   - 2M : 3M [M = Y] or 6M [M = X] or 4M [otherwise]
   - 2N : nat
   - 3X : good 5+X
2M : inv, 5+M [M = Y] or 4+M [otherwise], NF
2N*→3C*: transfer accepted
        - P  : s/o
        - 3D+: ... (TBD)
3Z : ST, 4+Z [Z = X] or semi-solid 6+Z [Z = Y] or 5+Z [otherwise]
3N : s/o
4C+: ... (TBD)
4Y : s/o
```

4.8.2 PLOB

not done yet

```
1C - 1D*; 1H*-
# 2S* is usually F1 only
1S*: any (9)10-14
   - 1N : 12-14, 2H bal
        - 2C : s/o
   - 2C : 12-14, 2-H, (5)6+C
   # bids below applies to both 1N and 2C
        - 2D*: F, not prefer to declare NT
        - 2H : s/o
        - 2S : s/o
        - 2N+: nat inv
   - 2D*: GF ... (TBD)
   - 2H : F, 3H
        - 2S*: F
        - 3S : inv
```

```
   - 2S*: GF, not prefer to declare NT
1N : nat NF
2X : s/o
   - 2S*: F
   - 2N+: nat inv
   - 3S*: 6+C, 5+S, F
2N*: 15+, catchall
3C*: fit in C, ST
3D*: 5+H, 5+D, ST
3H*: 6+H, ST
3S*: 4+S, ST
3N*: 18-19, 4H
```

```
1D - 1H; 1S -
1N : nat NF
2C*: any (9)10-14
   - 2D : 12-14, 2-H
      - P  : s/o
      - 2H : s/o
      - 2S*: F, not prefer to declare NT
         - 2N : min
         - 3N : max
      - 2N+: nat inv
   - 2H : F, 3H
   - 2S*: general GF
   - 2N+: nat GF
2X : s/o
2N*: 15+, catchall
3C*: fit in D, ST
3D*: 5+H, 5+C, ST
3H*: 6+H, ST
3S*: 4+S, ST
3N*: 18-19, 4H
```

Chapter 5

cardplay

5.1 lead

```
# lead vs suit: 3-low
# A/K = ask for count/att, but vs slam K = ask for count
A : AK, Ax, AKQ+, (AK+)
K : AKx, Kx, KQ+, (AK+)
Q : Qx(+), QJ+, (AQJ+)
J : Jx(+), JT+, AJT+, KJT+, (AJx), (KJx)
T : Tx, T9+, AT9+, KT9+, QT9+
hi-X: Xx, xXx+
lo-X: xXx+, HxX, HxXx, HxxxX, HxXxxx..., (xxX+)
# lead vs NT: 4-th
A : Ax, AKQ+, AK
K : Kx, KQ+, AKx, AKJT+
Q : Qx(+), QJ+, AQJ+, KQT9+, (AQx)
J*: Jx(+), JT+
T*: Tx(+), T9+, AJT+, KJT+
9*: 9x(+), AT9+, KT9+, QT9+, [AKT9+], [AQT9+]  # exceptionally
hi-X: Xx, xXx+, (Xx+)
lo-X: HxX, HxxX+, HHxX+, xXx+
```

5.2   signal

```
UDCA: lo = enc/even
# vs suit
partner's lead: att, s/p
declarer's lead: cnt, s/p
discarding: att
# vs NT
partner's lead: att, s/p
declarer's lead: cnt, s/p
discarding: lav
```

5.2.1   Smith echo

```
# vs NT, declarer wins the lead, then on the next round available for
    signal,
# if first round x - x - H - win
both defenders' lo: enc leading suit
# if first round x - win - [att] - x
leader's lo: enc leading suit
# note: on/off may depends on dummy/lead
```

5.2.2   other

```
# vs suit, knowing partner empty
hi/lo = S/P
```