Checking Life-and-Death Problems in Go II: Results

Lei Shen and Thomas Wolf

Brock University, St. Catharines, Ontario, Canada

twolf@brocku.ca

May 29, 2007

Contents

1 Introduction **2**

2 Problems with problems **4**
2.1 Mistakes changing the status to life or death 4
2.1.1 Problem 33, solution 1 4
2.1.2 Problem 64, solution 4 4
2.1.3 Problem 69, solution 2 5
2.1.4 Problem 98, solution 3 5
2.1.5 Problem 101, solution 6 6
2.1.6 Problem 122, solution 3 7
2.1.7 Problem 129, solution 4 7
2.1.8 Problem 154, solution 4 8
2.1.9 Problem 189, solution 3 9
2.1.10 Problem 192, solution 4 9
2.1.11 Problem 201, solution 2 11
2.1.12 Problem 215, solution 3 11
2.1.13 Problem 296, solution 3 12
2.1.14 Problem 319, solution 4 13
2.1.15 Problem 342, solution 4 14
2.1.16 Problem 412, solution 3 15
2.1.17 Problem 427, solution 2 15
2.1.18 Problem 429, solution 4 16
2.1.19 Problem 458, solution 4 17
2.1.20 Problem 459, solution 3 17
2.1.21 Problem 468, solution 1 18
2.1.22 Problem 475, solution 1 19
2.2 Mistakes changing the status to ko 19

2.2.1 Problem 121, solution 6 19
2.2.2 Problem 128, solution 2 20
2.2.3 Problem 181, solution 1 20
2.2.4 Problem 202, solution 1 21
2.2.5 Problem 209, solution 4 22
2.2.6 Problem 227, solution 1 22
2.2.7 Problem 296, solution 1 23
2.2.8 Problem 303, solution 4 23
2.2.9 Problem 439, solution 1 24
2.2.10 Problem 469, solution 4 24
2.3 Problems with more than one best move 25
2.3.1 Problem 81 25
2.3.2 Problem 84 26
2.3.3 Problem 91 26
2.3.4 Problem 144 27
2.3.5 Problem 155 27
2.3.6 Problem 165 28
2.3.7 Problem 360 28
2.3.8 Problem 426 28
2.3.9 Problem 481 29

Abstract

In checking a collection of 500 life-and-death problems with the program ScanLD (built on GoTools) a list of problems with partially incorrect solutions has been obtained. In this paper the results of the computerized check are discussed. The program ScanLD is the subject of the earlier paper [1].

1 Introduction

The computer program GoTools solves life-and-death problems in Go but can also be used to generate new problems automatically ([2]). Although these problems are spacious and sometimes hard, they have an artificial touch and are not of much interest to some Go-players that want to practice with game relevant positions. In this paper the program ScanLD which is built on GoTools is used to pick from a collection [4] (in the following called 'the book') of human invented problems those which have some kind of fault in one of their solutions. We hope that Go players find the incorrect problems even more interesting than the already interesting problem collection of the book. After-all, the solutions contain moves that are non-obvious enough that they were overlooked by the book authors.

Details of the program ScanLD are described in the first paper [1] of this sequel. In the section below we give a long list of the mistakes that were found except those where a better move only changes the number of ko-threats that are needed to win or changes the side which needs ko-threats in order to win.

Category	Problems	Percentage
a) boundary not easily closed	9	1.8%
b) timed out	167	33.4%
c) agreement	245	49%
d) changes status to life/death	20	4%
e) missing ko	10	2%
f) change of # of ko-threats	42	8.4%
g) no error, more than 1 best move	7	1.4%

Table 1: A statistics of the 500 problems in the book

A few short comments that were already made in [1] are repeated here to save the purely Go interested reader from scanning all of [1].

We discuss only occurrences of non-optimal play that are not already identified to be wrong in the book.

Go-players can use the diagrams twice, once for solving the given problem and one more time by trying to find out what is wrong with the given solution.

The program ScanLD is based on the program GoTools and thus shares its weaknesses. One of them is that the problems have to be reasonably closed to be solved efficiently. The subroutine performing this closure is less accurate than the underlying solver but that is not a serious deficit as the cases where the boundary is too wide or too narrow can be corrected efficiently by hand. Another feature is that life and seki are treated alike. A complete list is given in the appendix of [1]. They are all not critical for the purpose of this investigation.

Readers can solve problems interactively and try out variations online at [3] but the boundary of problems has to be closed before suitably.

Table 1 shows problems where a) the boundary can not be closed easily without changing the status of the problem, b) the problem is currently too hard to be solved in a reasonably time, c) the program is in complete agreement with the book, d) the mistake changes the status to life or death, e) the mistake changes the status to ko, f) the mistake changes the number of external ko-threats that are needed to win or the side which needs external ko-threats to win, g) the problem has no bug in any solution but one or more additional best first moves which are not mentioned in the book.

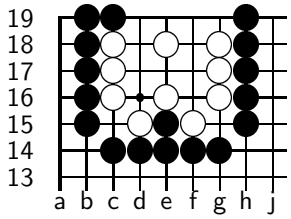
In the following 3 sections problems of type d), e) and g) are given. In all cases the first diagram shows the original problem as given in the book. The second diagram shows a solution from the book with non-optimal moves (which have not been identified as wrong in the book). For each such move follows a diagram showing a better move and what it can reach. If these diagrams do not already include the correct solution of the problem (because diagram 2 is not solution 1 in the book which in most cases is supposed to be the correct solution), then the solution with optimal play is added as the last diagram for this problem.

2 Problems with problems

2.1 Mistakes changing the status to life or death

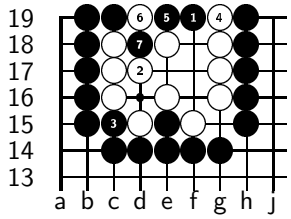
For all problems in the book ● plays first and what we call 'solution 1' below is the first solution in the book which always is supposed to be the solution where both sides plays optimally. Later solutions typically discuss non-optimal variations.

2.1.1 Problem 33, solution 1



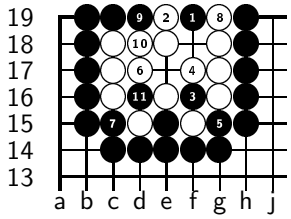
● to move.

Diagram 1.



Find any error.

Diagram 2.



⑫ @ ❸

Diagram 3.

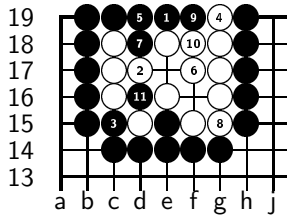
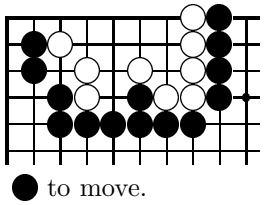


Diagram 4.

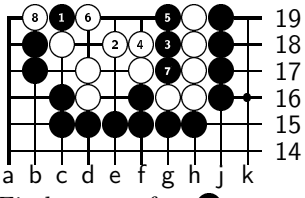
In Diagram 2, ◯ needs to win a ko at d19 to live. But ◯ can live by playing ② on e19 as shown in Diagram 3. The only ❶ that reaches at least a ko is shown in Diagram 4 and is missed in the book too.

2.1.2 Problem 64, solution 4



● to move.

Diagram 5.



Find errors after ❶.

Diagram 6.

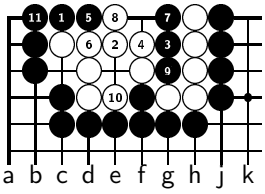


Diagram 7.

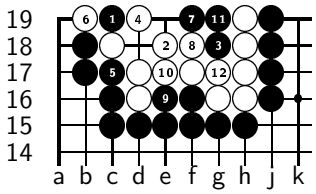


Diagram 8.

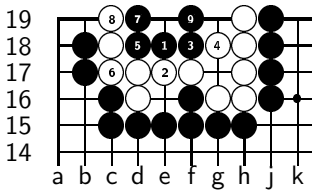
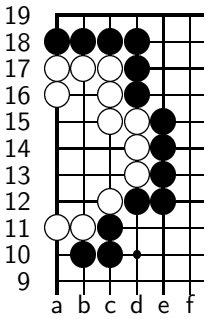


Diagram 9.

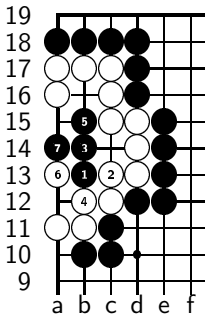
In Diagram 6, ❺ on g19 is wrong in that ◯ lives. With ❺ on d19 in Diagram 7, ◯ dies. However, already ④ is wrong because ④ on d19 as in Diagram 8 can ensure life. (Diagram 9 from the book shows the best play of ● .)

2.1.3 Problem 69, solution 2



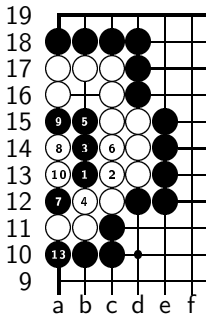
● to move.

Diagram 10.



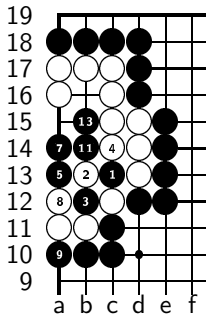
Find error after ❸ .

Diagram 11.



⑪ @ ❼ ⑫ @ ⑩

Diagram 12.

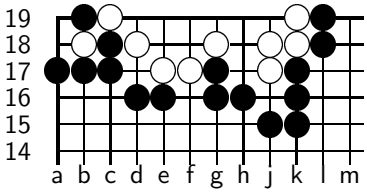


⑥ @ c12 ⑩ @ ❸
⑫ @ ❶

Diagram 13.

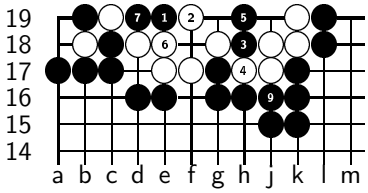
⑥ on a13 in Diagram 11 was wrong in that ◯ needs to win a ko to live. However, with ⑥ on c14 in Diagram 12, ◯ lives. (Diagram 13 from the book shows the best play of ● .)

2.1.4 Problem 98, solution 3



● to move.

Diagram 14.



⑧ @ c19 ⑩ @ ❼
Find error after ❶.

Diagram 15.

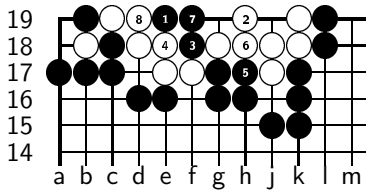


Diagram 16.

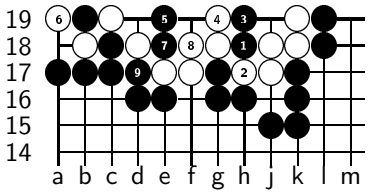
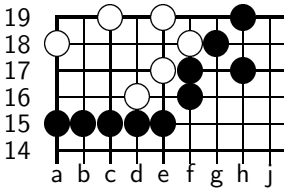


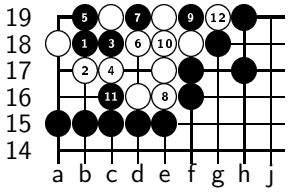
Diagram 17.

In Diagram 15, although ● kills the right 4 ○ stones, ○ saves the other group. But ○ could save the right 5 stones as well by playing ② on h19 in Diagram 16 . (Diagram 17 from the book shows the best play of ● which kills.)

2.1.5 Problem 101, solution 6



● to move.
Diagram 18.



Find error after ❶ .
Diagram 19.

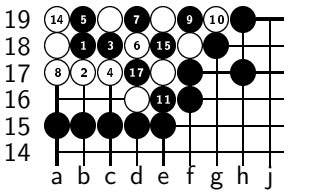


Diagram 20.

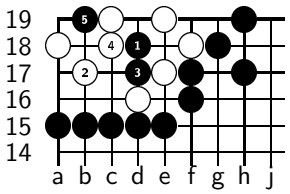
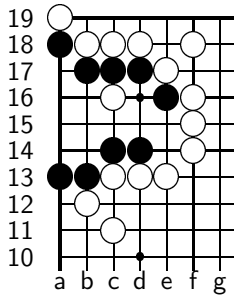


Diagram 21.

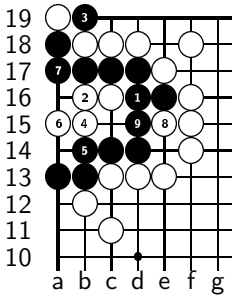
In Diagram 19, ○ needs to win a ko to live. But with ⑧ on a17 as shown in Diagram 20, ○ lives. (Diagram 21 from the book shows the best moves for ● in that ○ dies.)

2.1.6 Problem 122, solution 3



● to move.

Diagram 22.



Find errors after ②.

Diagram 23.

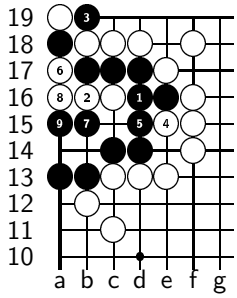
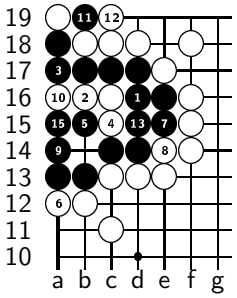


Diagram 24.

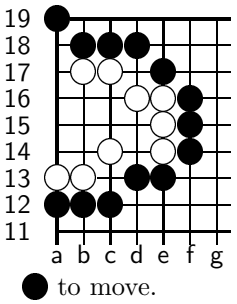


⑭ @ a19

Diagram 25.

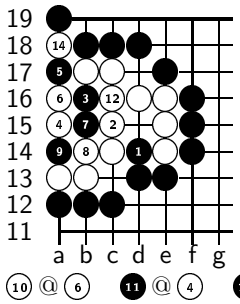
In Diagram 23, ④ on b15 is wrong in that ● is safe in seki. By playing ④ on e15 as in Diagram 24, ● needs to win a ko to live. However, ❸ on b19 in Diagram 23 is wrong already, whereas ❸ on a17 as in Diagram 25 guarantees ● to live. (❶ in Diagram 25 is the correct first move in agreement with the book.)

2.1.7 Problem 129, solution 4



● to move.

Diagram 26.



⑩ @ ⑥ ❶❶ @ ④ ❶❸ @ ⑥
❶❺ @ ④

Find error after ❶ .

Diagram 27.

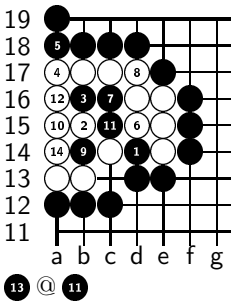


Diagram 28.

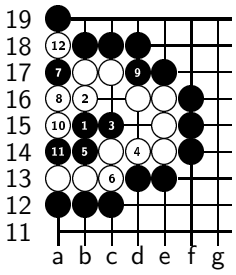
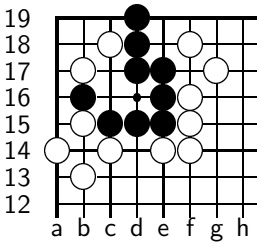


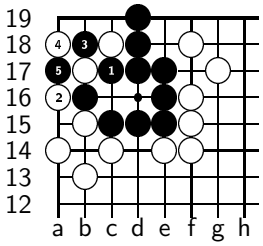
Diagram 29.

In Diagram 27, ◯ dies with ② on c15. This is not a mistake in the book because it makes a comment about ② not being optimal but also the book does not say which ② would be better. By playing ② on b15 as shown in Diagram 28, ◯ lives. (Diagram 29 from the book shows the best ❶ on b15 in that ◯ needs to win a ko to live.)

2.1.8 Problem 154, solution 4



● to move.
Diagram 30.



Find error after ❶ .
Diagram 31.

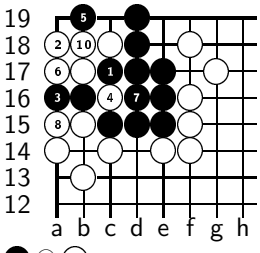


Diagram 32.

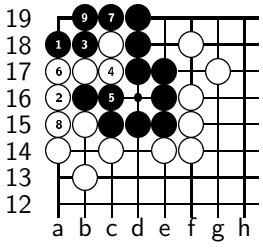
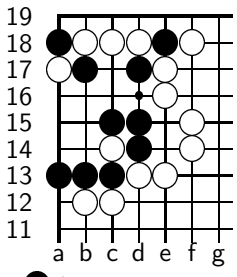


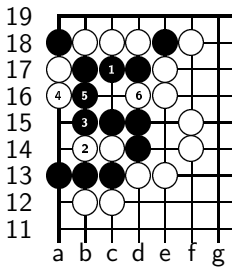
Diagram 33.

In Diagram 31, ◯ needs to win a ko to kill. But with ② on a18 in Diagram 32, ◯ kills. (Diagram 33 from the book shows the correct move ❶ on a18 enabling ● to live.)

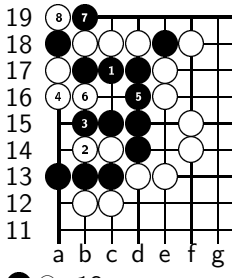
2.1.9 Problem 189, solution 3



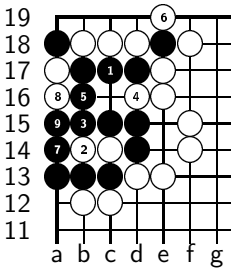
● to move.
Diagram 34.



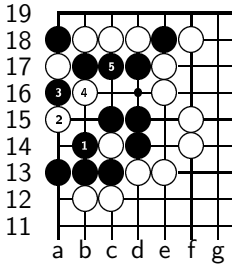
Find error after ❶ .
Diagram 35.



❾ @ a18
Diagram 36.



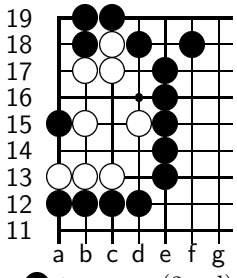
⑩ @ a17
Diagram 37.



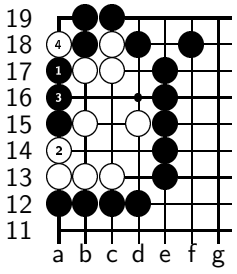
⑥ @ a17
Diagram 38.

Although ● dies in Diagram 35, ④ is wrong because after ❺ on d16 in Diagram 36, ○ needs to win a ko to kill. Better is ④ on d16 in Diagram 37 which kills. (The solution with optimal play of ● from the book is given in Diagram 38 enabling ● to live in ko.)

2.1.10 Problem 192, solution 4



● to move (2 sol).
Diagram 39.



Find errors after ❶.
Diagram 40.

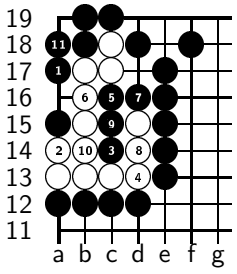


Diagram 41.

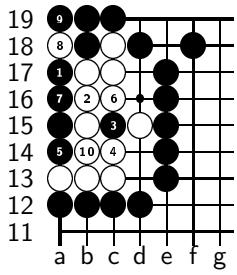


Diagram 42.

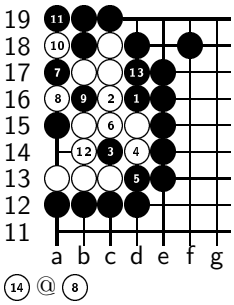


Diagram 43.

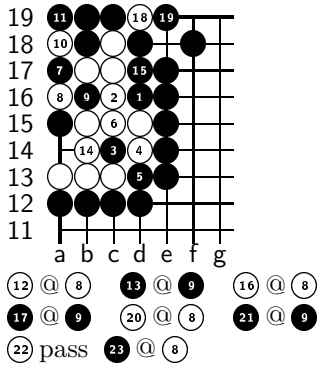


Diagram 44.

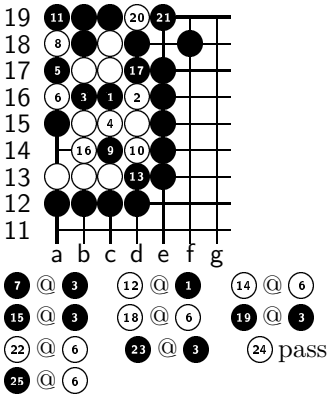
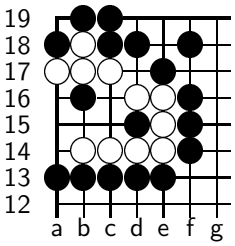


Diagram 45.

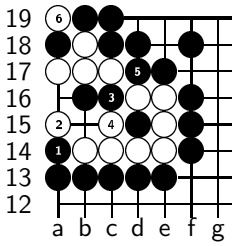
In Diagram 40, ● 3 on a16 is wrong in that ○ lives. By playing ● 3 on c14 as in Diagram 41, ○ dies. However, ○ 2 on a 14 in diagram 40 is wrong already, and by playing ○ 2 on b16 as in Diagram 42, ○ lives. Diagram 43 from the book shows optimal play apart from ○ 12 which should be on a16 after which ● needs 3 external ko-threats as shown in Diagram 44. In Diagram 43 ● needs only 2 external ko-threats to win (when ○ uses the internal threat on d19). The book authors surely know that but according to the book ○ 12 is done to get more liberties when instead playing on a16 is better.

A similar but different optimal play of ● which is missed in the book is shown in Diagram 45 in which ● also needs 3 external ko-threats.

2.1.11 Problem 201, solution 2



● to move.
Diagram 46.



Find error after ❶ .
Diagram 47.

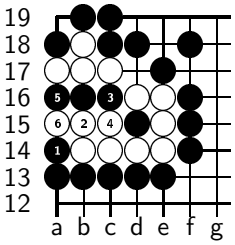


Diagram 48.

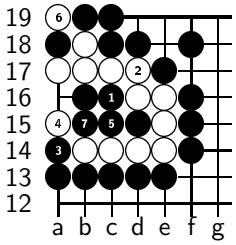
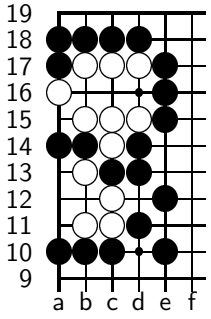


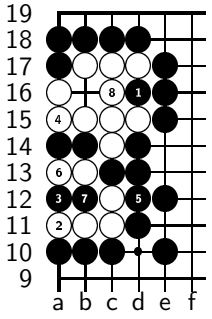
Diagram 49.

In Diagram 47, ② on a15 is wrong in that ◯ needs to win a ko to live. By playing ② on b15 as in Diagram 48, ◯ lives. (Diagram 49 from the book shows the best ❶ which kills.)

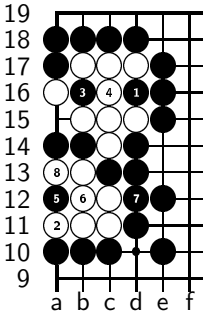
2.1.12 Problem 215, solution 3



● to move.
Diagram 50.



Find errors after ❶.
Diagram 51.



❾ @ ❺
Diagram 52.

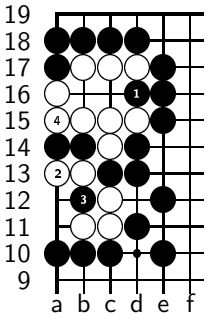


Diagram 53.

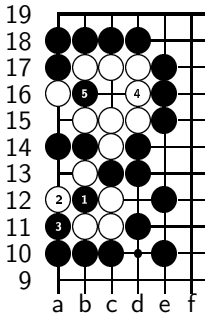
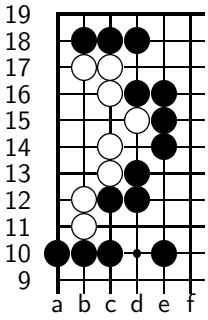


Diagram 54.

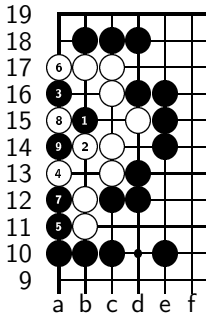
In Diagram 51, ❸ on a12 is wrong in that ◯ lives. By playing ❸ on b16 as in Diagram 52, ● kills. However , ② on a11 in Diagram 51 is wrong already and with ② on a 13 as in Diagram 53, ◯ lives. (Diagram 54 from the book shows the best play of ● which kills.)

2.1.13 Problem 296, solution 3



● to move.

Diagram 55.



Find error after ❶ .

Diagram 56.

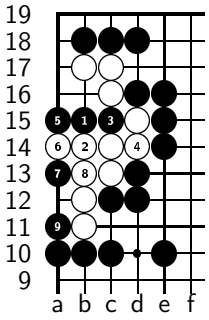


Diagram 57.

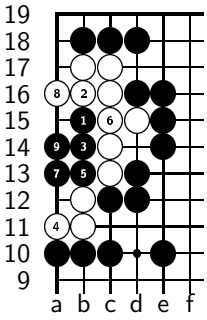
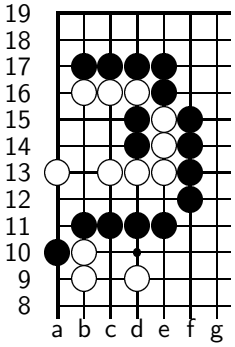


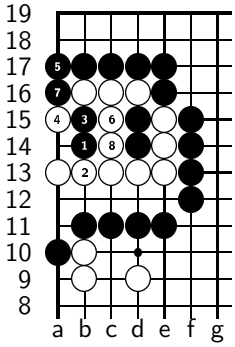
Diagram 58.

In Diagram 56, ❸ on a16 is wrong in that ◯ can still get a ko. By playing ❸ on a13 as in Diagram 57, ● kills. A variation with a different ② in which ● kills too is shown in Diagram 58 answering to a different ②. A Diagram like the last 2 should replace solution 1 in the book because of another mistake regarding the same problem (see the comments following Diagram 119 in the next subsection).

2.1.14 Problem 319, solution 4



● to move.
Diagram 59.



Find errors after ❶.
Diagram 60.

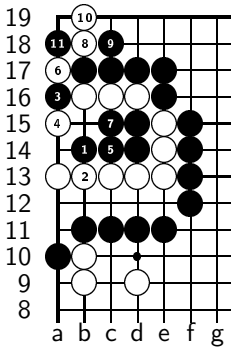


Diagram 61.

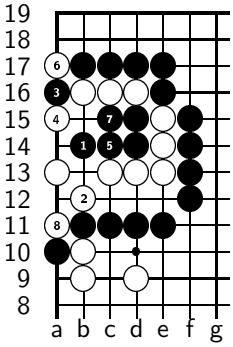


Diagram 62.

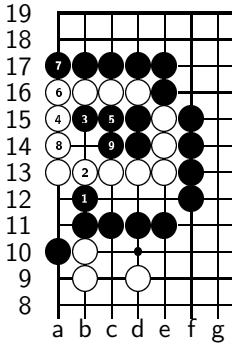
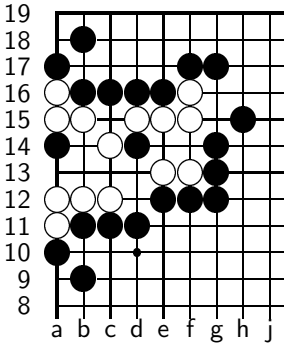


Diagram 63.

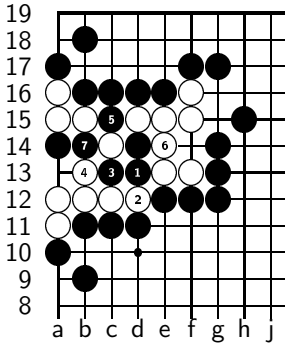
❸ on b15 in Diagram 60 is wrong in that it fails to kill ○ . With ❸ on a16 in Diagram 61, ● kills. Already earlier ② on b13 in Diagram 60 is not optimal. ② on b12 in Diagram 62 reaches a ko. (Optimal play of ● that kills is shown in diagram 63 from the book.)

2.1.15 Problem 342, solution 4



● to move.

Diagram 64.



⑧ @ ❶

Find error after ❶ .

Diagram 65.

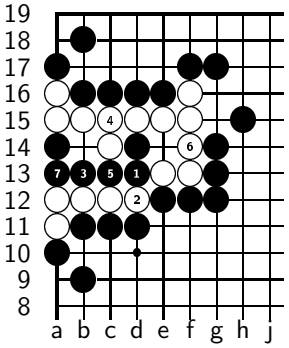
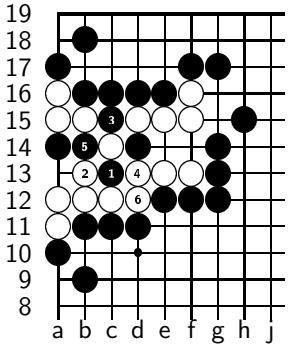


Diagram 66.

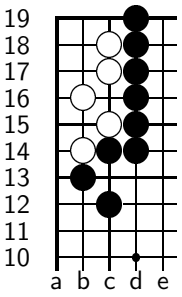


❼ @ ❶

Diagram 67.

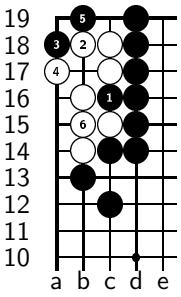
❸ on c13 in Diagram 65 is wrong in that it fails to kill ◯ . But with ❸ on b13 as in Diagram 66, ● kills. (Diagram 67 from the book shows optimal ● play leading to ko.)

2.1.16 Problem 412, solution 3



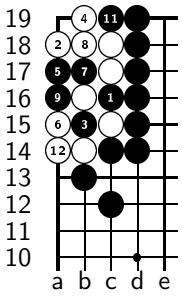
● to move.

Diagram 68.



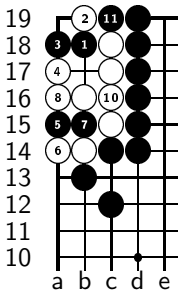
Find error after ● .

Diagram 69.



⑩ @ b16

Diagram 70.

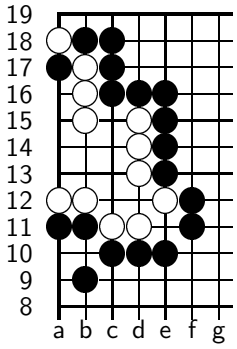


● @ ●

Diagram 71.

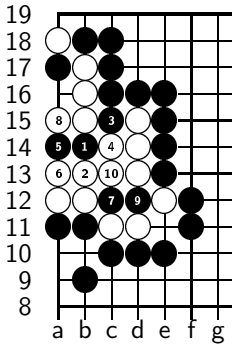
In Diagram 69, ◯ needs to win a ko to live. But ◯ can live by playing ② on a18 in Diagram 70. (Diagram 71 from the book shows optimal play of ● that kills.)

2.1.17 Problem 427, solution 2



● to move.

Diagram 72.



● @ ●

Find error after ② .

Diagram 73.

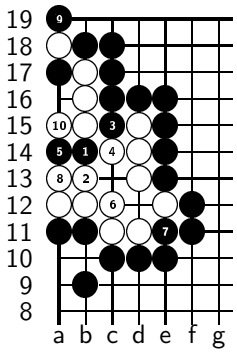


Diagram 74.

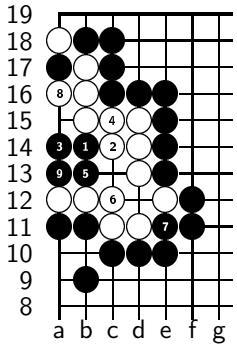
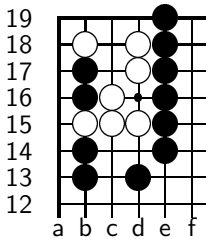


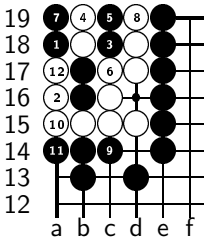
Diagram 75.

⑥ on a13 in Diagram 73 is wrong in that ◯ dies. But ◯ lives by playing ⑥ on c12 in Diagram 74 . (Diagram 75 from the book shows another variation of ② in which ◯ dies.)

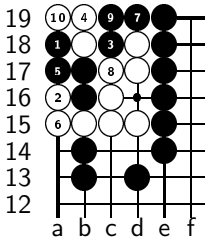
2.1.18 Problem 429, solution 4



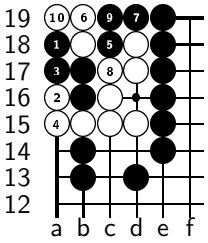
● to move.
Diagram 76.



Find error after ❸ .
Diagram 77.



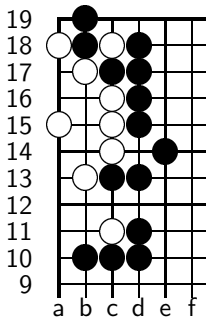
⓫ @ b17
Diagram 78.



⓫ @ b17
Diagram 79.

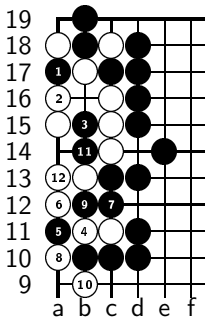
❺ on c19 in Diagram 77 is wrong in that ◯ lives. By playing ❺ on a17 as in Diagram 78, ● kills. Thus, contrary to the statement in the book, ❸ on c18 in Diagram 77 is correct. (A variation of Diagram 78 is solution 1 from the book in Diagram 79 in which ● kills too.)

2.1.19 Problem 458, solution 4



● to move.

Diagram 80.



Find error after ❺ .

Diagram 81.

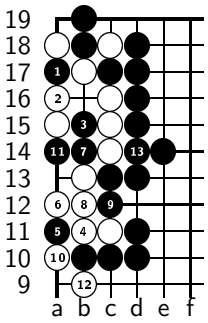


Diagram 82.

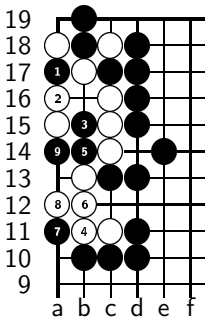
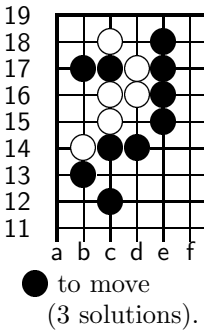


Diagram 83.

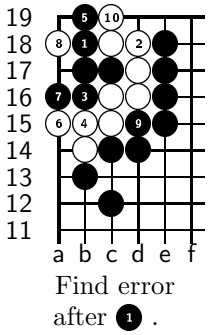
In Diagram 81, ❼ on c12 is wrong in that ◯ lives through escape. By playing ❼ on b14 as in Diagram 82, ● at least captures the majority of ◯ . (Optimal play of ● that kills is shown in Diagram 83 from the book.)

2.1.20 Problem 459, solution 3



● to move (3 solutions).

Diagram 84.



Find error after ❶ .

Diagram 85.

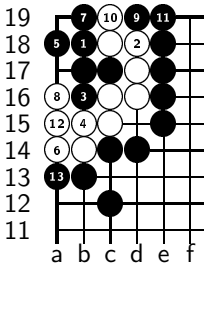


Diagram 86.

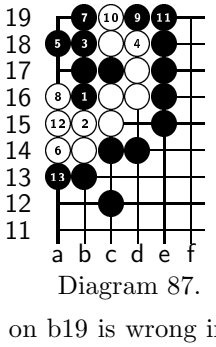


Diagram 87.

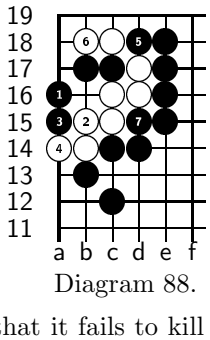
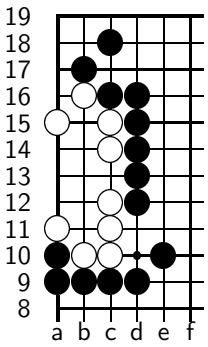


Diagram 88.

In Diagram 85, ❺ on b19 is wrong in that it fails to kill ◯ . With ❺ on a18 in Diagram 86, ● kills ◯ . However, ❶ on b18 is not the only move that kills.

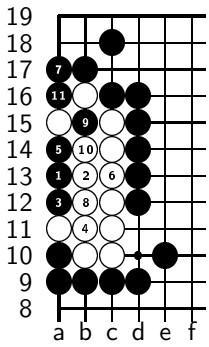
Reversing the order of ❶ and ❸ works as well, see Diagram 87. Another killing ❶ is shown in Diagram 88 from the book.

2.1.21 Problem 468, solution 1



● to move.

Diagram 89.



⑫ @ ❶ ❿ @ ❾

Find any error.

Diagram 90.

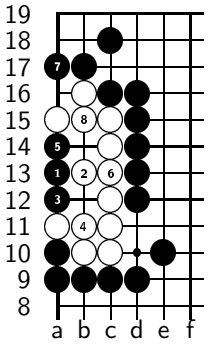
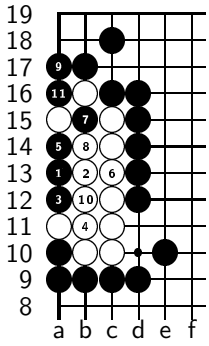


Diagram 91.

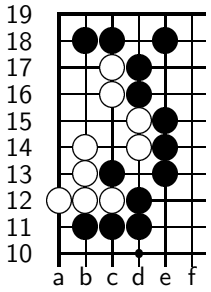


⑫ @ ❶ ❿ @ ❼

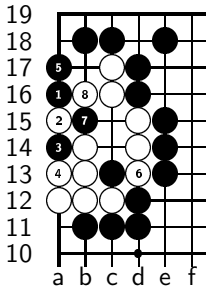
Diagram 92.

In Diagram 90, ⑧ on b12 is wrong in that ◯ needs to win a ko to live. By playing ⑧ on b15 as in Diagram 91, ◯ lives. However, ❼ on a17 is already wrong, and by playing ❼ on b15 in Diagram 92, ◯ needs to win a ko to live. The last diagram shows optimal play and should replace solution 1 in the book.

2.1.22 Problem 475, solution 1



● to move.
Diagram 93.



Find any error.
Diagram 94.

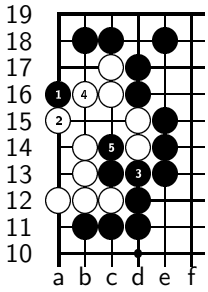


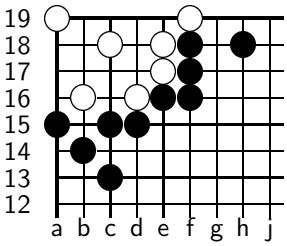
Diagram 95.

Diagram 94 (which is supposed to show optimal play of ● in the book) results in a ko. But by playing ❸ on d13 in Diagram 95 ● can kill.

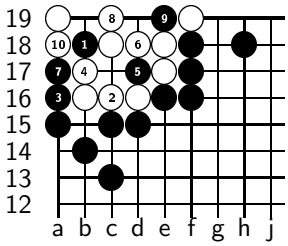
2.2 Mistakes changing the status to ko

In the problems of this subsection moves are missed which would provide a ko for the side that otherwise would lose unconditionally.

2.2.1 Problem 121, solution 6



● to move.
Diagram 96.



Find error after ❶ .
Diagram 97.

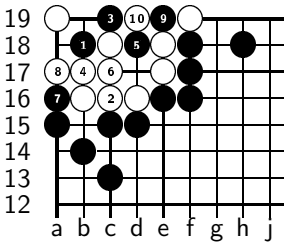


Diagram 98.

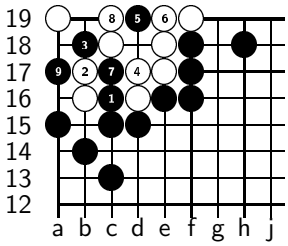
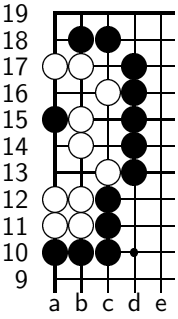


Diagram 99.

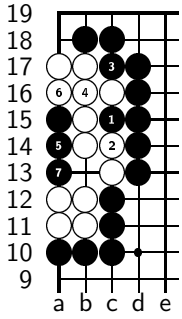
In Diagram 97, ❸ on a16 is wrong in that ○ lives. However, by playing ❸ on c19 in Diagram 98, ● could kill ○ by winning a ko. (Diagram 99 from the book shows optimal ● play that kills.)

2.2.2 Problem 128, solution 2



● to move.

Diagram 100.



Find error after ② .

Diagram 101.

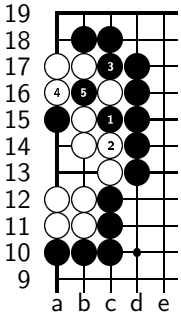


Diagram 102.

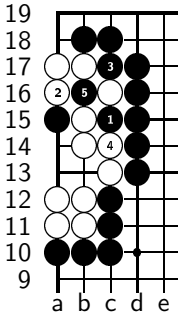
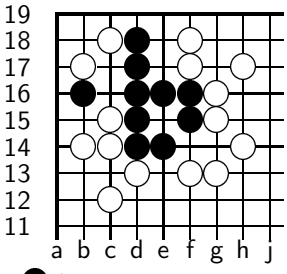


Diagram 103.

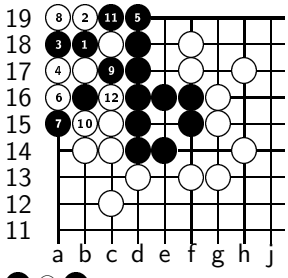
In Diagram 101, ④ on b16 is wrong in that ○ dies. By playing ④ on a16 as in Diagram 102, ○ could live by winning a ko. (Diagram 103 shows a variation as shown in the book for optimal play of ● with the same outcome as in diagram 102.)

2.2.3 Problem 181, solution 1



● to move.

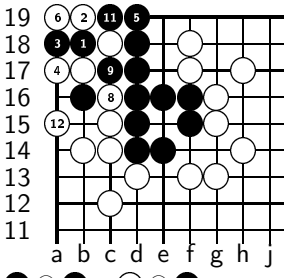
Diagram 104.



● 13 @ ● 1

Find any error.

Diagram 105.



● 7 @ ● 3 ○ 10 @ ● 1

Diagram 106.

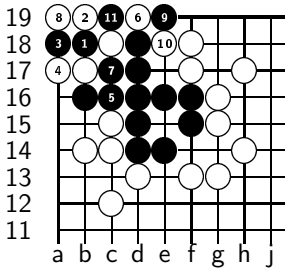


Diagram 107.

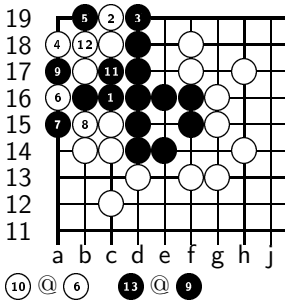
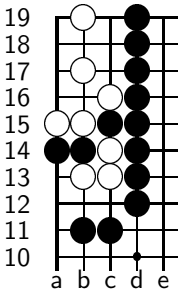


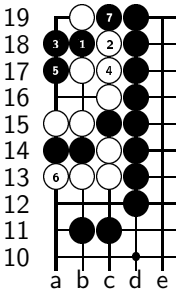
Diagram 108.

In Diagram 105 the move (6) is not optimal as it enables ● to live. Better is (6) on a19 as shown in Diagram 106 which kills. One move earlier, ● 5 is an error too as it could have prevented death by playing on c16 in Diagram 107 leading to a favourable ko for ● . A slightly better and optimal play of ● is shown in Diagram 108 where ○ needs one more ko-threat to kill. The solution in the last Diagram is equivalent to solution 3 in the book which due to the above mistakes is described there falsely as non-optimal.

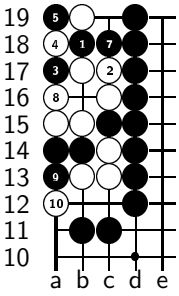
2.2.4 Problem 202, solution 1



● to move.
Diagram 109.



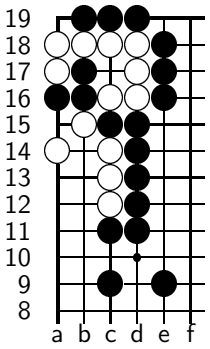
Find any error.
Diagram 110.



(6) @ (4)
Diagram 111.

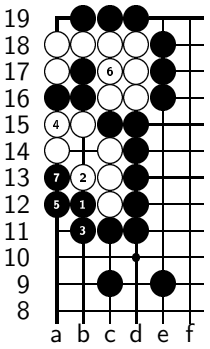
In Diagram 110, ○ dies. But by playing (2) on c17 as in Diagram 111, ○ could live by winning a ko. Diagram 111 replaces Diagram 110 from the book as the correct solution.

2.2.5 Problem 209, solution 4



● to move.

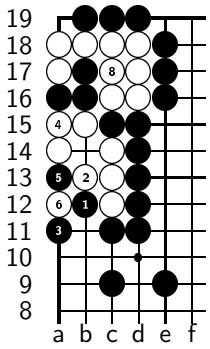
Diagram 112.



⑧ @ b16

Find error after ❶ .

Diagram 113.



❼ @ ❺ ❾ @ b16

⑩ @ ⑥

Diagram 114.

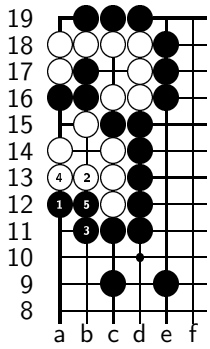
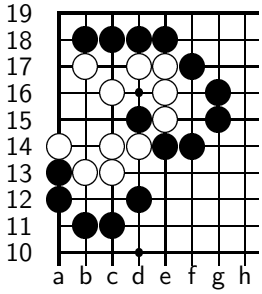


Diagram 115.

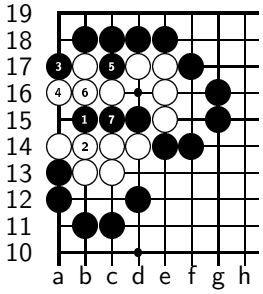
In Diagram 113, ❸ on b11 is wrong in that ◯ lives. By playing ❸ on a11 as in Diagram 114, ◯ has to win ko to live. (Diagram 115 from the book shows optimal ● play that kills.)

2.2.6 Problem 227, solution 1



● to move.

Diagram 116.



Find any error.

Diagram 117.

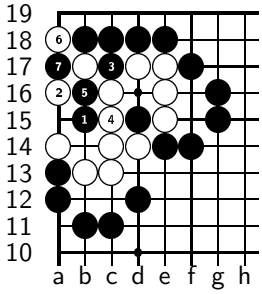
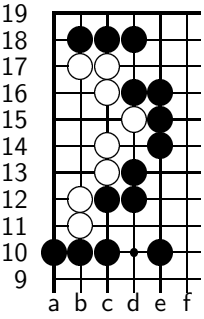


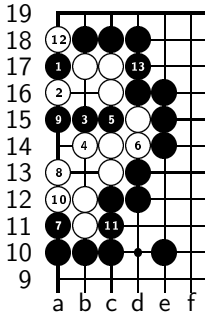
Diagram 118.

In Diagram 117 ② on b14 is wrong in that ◯ dies. Diagram 118 shows the better move ② on a16 achieving a ko. It should replace Diagram 117 in the book as the correct solution.

2.2.7 Problem 296, solution 1



● to move.
Diagram 119.



Find any error.
Diagram 120.

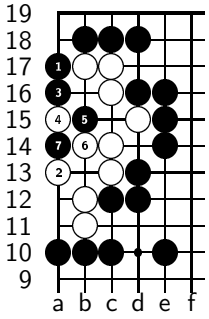
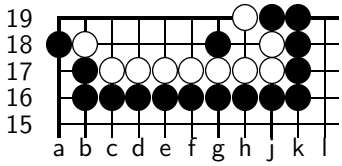


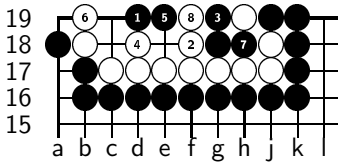
Diagram 121.

In Diagram 120, ② on a16 is wrong in that ◯ dies. By playing ② on a13 as in Diagram 121, ◯ could live by winning a ko. For optimal ● play see Diagrams 57 and 58 in the previous section.

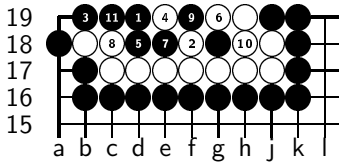
2.2.8 Problem 303, solution 4



● to move.
Diagram 122.



Find error after ❶ .
Diagram 123.



⑫ @ ④
Diagram 124.

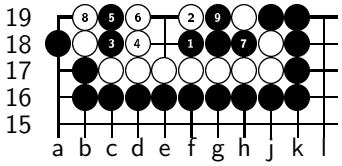
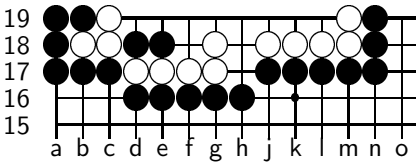


Diagram 125.

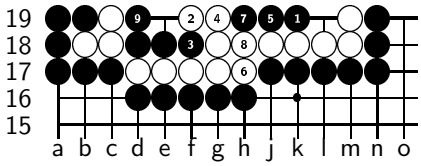
In Diagram 123, ❸ is wrong in that ◯ lives. By playing ❸ on b19 as in Diagram 124, ◯ needs to win ko to live. (Diagram 125 from the book shows how ● can kill.)

2.2.9 Problem 439, solution 1



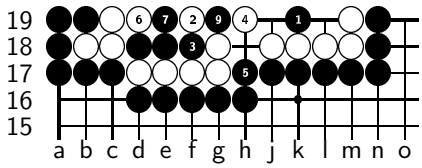
● to move.

Diagram 126.



Find any error.

Diagram 127.

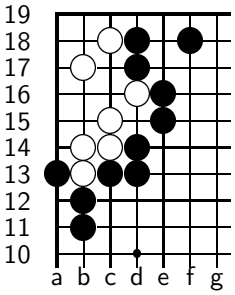


(8) @ c18

Diagram 128.

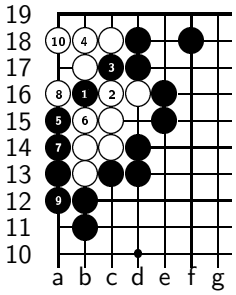
According to the book Diagram 127 is supposed to show the correct solution but (4) on g19 is wrong in that ○ dies. By playing (4) on h19 as in Diagram 128, ○ could live by winning a ko. This is the best ● can reach. It is better than what the other solutions in the book get.

2.2.10 Problem 469, solution 4



● to move.

Diagram 129.



Find error after ❶ .

Diagram 130.

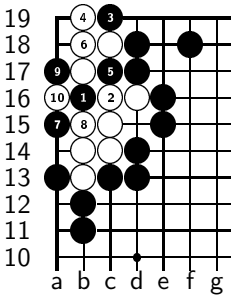
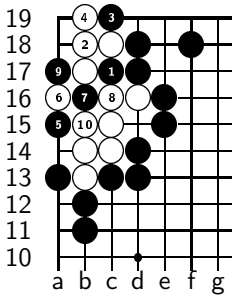


Diagram 131.



❶❶ @ (6)

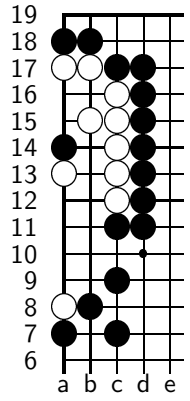
Diagram 132.

In Diagram 130, ❸ on c17 is wrong in that ◯ lives. By playing ❸ on c19 as in Diagram 131, ◯ needs to win ko to live. (Diagram 132 from the book shows how ● can kill.)

2.3 Problems with more than one best move

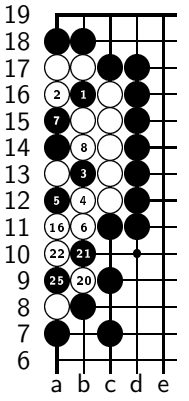
Some problems in the book are somewhat less suitable for a tsume go collection because they have at least two equally good first moves. If there are any mistakes in their solutions in the book then these problems have been discussed in the previous sections. Here we list the additional error-free problems with non-unique solutions. The extra winning move is often an strong threat which has to be answered before the best move from the book is made. Another source of multiple best moves is the situation where the order of ❶ and ❸ can be reversed.

2.3.1 Problem 81



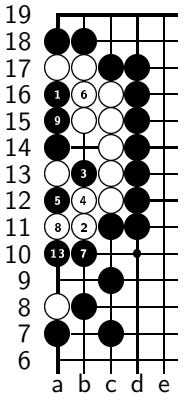
● to move.

Diagram 133.



❾ @ ❶ ⑩ @ a13 ❶❶ @ ❸
⑫ @ b17 ❶❸ @ a17 ⑭ @ a13
❶❺ @ ❸ ❶❼ @ a13 ⑱ @ ②
❶❾ @ a13 ❷❸ @ a14 ㉔ @ b17

Diagram 134.

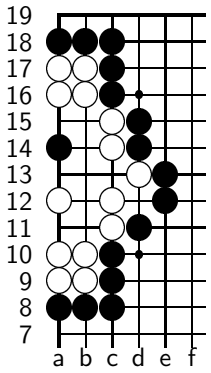


⑩ @ a13 ❶❶ @ ❺
⑫ pass

Diagram 135.

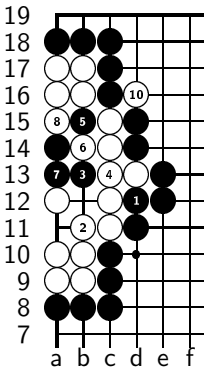
In addition to ❶ on b16 in Diagram 134, ❶ on a16 as in Diagram 135 also reaches the status that ● needs two external ko-threats to win the ko although in Diagram 135 ◯ can afford to pass one more time (which the computer program ScanLD does not take into account). The moves 1-10 in Diagram 134 are from the book, the extra moves are from the program to show that ● needs in total two external ko-threats to win.

2.3.2 Problem 84



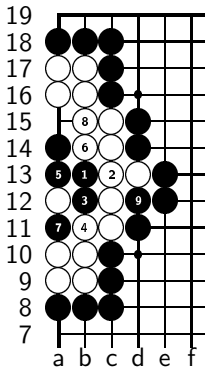
● to move.

Diagram 136.



❾ @ ❺ ⓫ @ a16

Diagram 137.

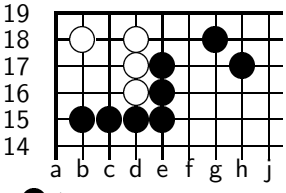


⑩ @ a12

Diagram 138.

In addition to ❶ on d12 in Diagram 137, ❶ on b13 as in Diagram 138 also reaches the status that ● needs one external threat to win the ko.

2.3.3 Problem 91



● to move.

Diagram 139.

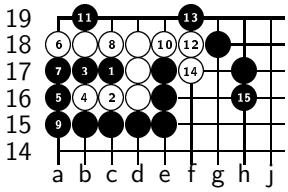


Diagram 140.

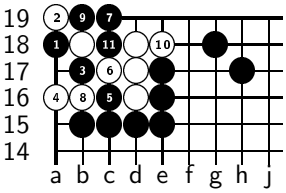


Diagram 141.

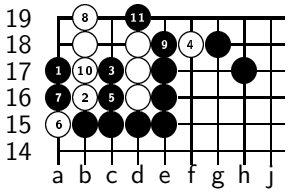
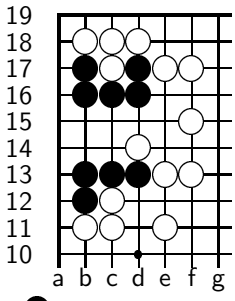


Diagram 142.

In addition to ❶ on c17 in Diagram 140, also ❶ on a18 in Diagram 141 and ❶ on a17 in Diagram 142 kill.

2.3.4 Problem 144



● to move.

Diagram 143.

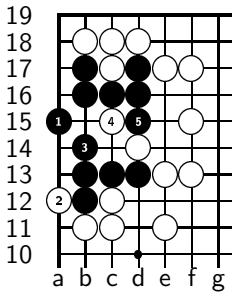


Diagram 144.

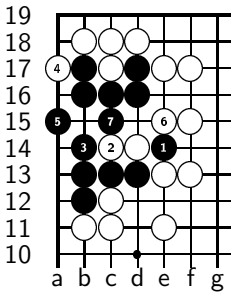
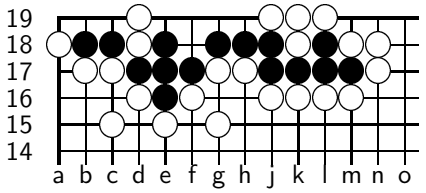


Diagram 145.

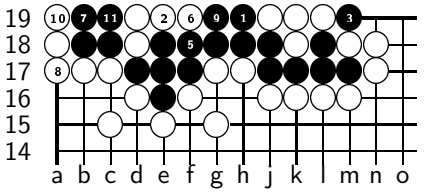
Apart from ❶ in Diagram 144 there is another ❶ with one possible follow-up sequence shown in Diagram 145 which also guarantees life.

2.3.5 Problem 155



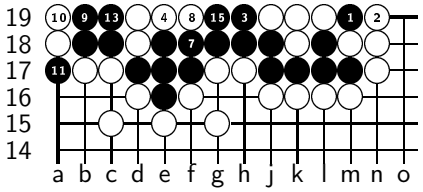
● to move.

Diagram 146.



④ @ k19 ⑫ @ ② ❸ @ l19 ⑭ @ d18
❺ @ ⑥ ⑯ @ d19 ❼ @ c18

Diagram 147.



❺ @ ❶ ⑥ @ k19 ⑫ @ l19
⑭ @ a18 ⑯ @ ④ ❼ @ d19

Diagram 148.

In addition to ❶ on h19 in Diagram 147 also ❶ on m19 as in Diagram 148 reaches the status that ● lives.

2.3.6 Problem 165

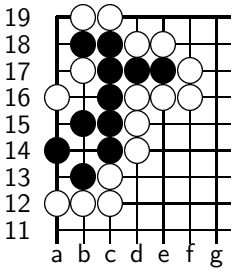


Diagram 149.

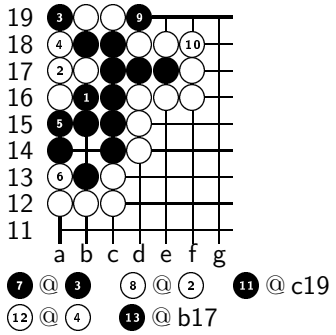


Diagram 150.

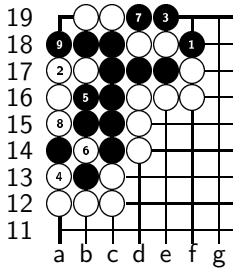


Diagram 151.

In addition to ❶ on b16 in Diagram 150, Diagram 151 shows another best ❶ on f18 also ensuring life.

2.3.7 Problem 360

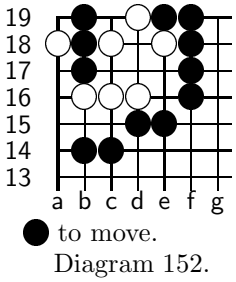


Diagram 152.

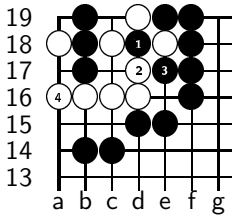


Diagram 153.

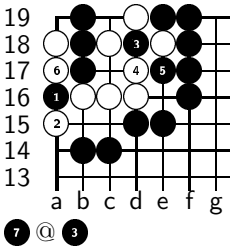


Diagram 154.

In addition to ❶ on d18 in Diagram 153 also ❶ on a16 as in Diagram 154 reaches the status that ○ needs one external threat to win the ko.

2.3.8 Problem 426

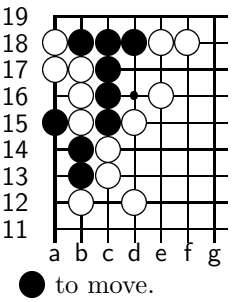


Diagram 155.

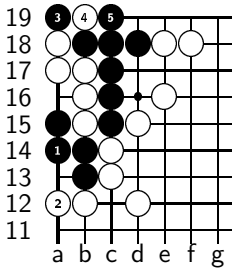


Diagram 156.

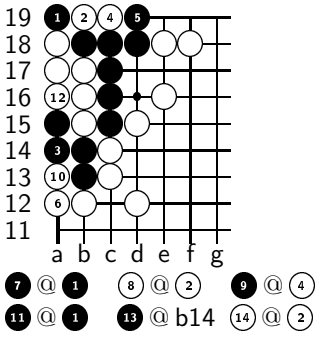
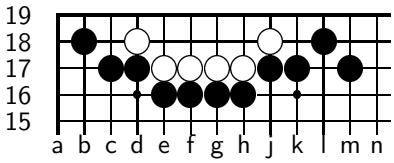


Diagram 157.

In addition to ❶ on a14 in Diagram 156 also ❶ on a19 as in Diagram 157 reaches the status that ● needs one external threat to win the ko.

2.3.9 Problem 481



● to move.

Diagram 158.

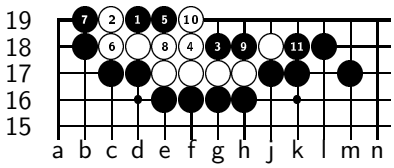


Diagram 159.

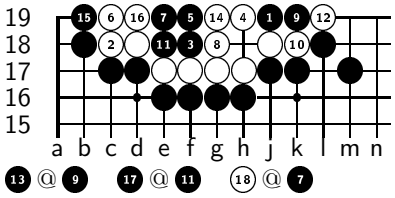


Diagram 160.

In addition to ❶ on a14 in Diagram 159 also ❶ on j19 as in Diagram 160 reaches the status that ● kills.

References

[1] Wolf, T and Shen, Lei: Checking Life-and-Death Problems in Go I: The Program ScanLD.
`http://lie.math.brocku.ca/twolf/papers/bugsintro.ps`

[2] Wolf, T: Quality improvements in tsume go mass production, in the Proceedings of the Second GO & Computer Science Workshop, Cannes 1993, publ. by INRIA Sophia-Antipolis, 11 pages.

[3] Vesinet, J P and Wolf, T: GoTools online,
`http://lie.math.brocku.ca/gotools/index.php?content=go_applet`

[4] Li, A and Li, Y: Korean Life-and-Death Problems (in Chinese) Beijing Sport University, ISBN 7811000601, 271 pages (2005).