Problem A: Multi-bit Large-scale Boolean Matching

Chung-Han Chou, Chih-Jen (Jacky) Hsu, Chi-An (Rocky) Wu, and Kuan-Hua Tu, Kei-Yong Khoo
(Cadence Design Systems, Inc.)

Q&A

Q1. I am writing to inquire about the evaluation criteria for Problem A.
In the "Evaluation Criteria" section, I noticed that it is stated that "The shorter runtime the contestant's program uses, the higher score the contestant will receive." However, I couldn't find any specific scoring rules that involve runtime.
Could you please clarify how runtime will be calculated in the score? I would greatly appreciate any information you can provide on this matter. Thank you for your time and attention.
A1. Thanks for your question. The runtime is a hard limit in this contest, and will not be calculated in the score. We will remove the relative description "The shorter runtime the contestant's program uses, the higher score the contestant will receive." in the next version.

Q2. I hope this email finds you well. I am writing to request clarification on some aspects of Problem A.

1) Firstly, in the "Output file format" section, I would like to know what the numbers 1 and 2, as well as the positive and negative signs inside the "INGROUP" and "OUTGROUP" tags, represent.

2) Secondly, I am wondering if the ports within a BUS have any specific order. For example, if circuit I has a bus with ports {a0, a1, a2}, and circuit II has a bus with ports {b0, b1, b2}, would the matching of a and b only be limited to {a0, a1, a2}={b0, b1, b2} and {a0, a1, a2}={b2, b1, b0}? Is it correct to assume that the order of the ports within the bus does not matter?

3) Thirdly, I would like to know if it is possible for buses with different lengths to match. For instance, if circuit I has a bus with ports {a0, a1}, and circuit II has a bus with ports {b0, b1, b2}, is it possible to match b2 to a constant value, resulting in a match such as {a0, a1, 1'b1}={b0, b1, b2}?

4) Lastly, I am curious if it is possible to connect or concatenate multiple buses to form a single bus. For example, if circuit I has a bus with ports {a0, a1, a2, a3}, and circuit II has two buses with ports {b0, b1} and {c0, c1}, respectively, is it possible to match both buses to the same bus, resulting in a match such as {a0, a1, a2, a3}={b0, b1, c0, c1}?

Thank you for your time.

A2. Thanks for your questions.

1) In this contest, we have 2 input circuit, Circuit I and Circuit II respectively. The 1 and 2 in the “Output file format” section denote to that the pin comes from the Circuit I or Circuit II. The positive/negative sign means the phase of the matching. We will add the description into the problem description in the next version.

2) The buses are ordered. The contestant need to map the LSB in Circuit I to LSB in Circuit II, and map MSB in Circuit I to MSB in Circuit II. As the result, if the correct order is “{a0, a1, a2} = {b0, b1, b2}”. Then “{a2, a1, a0} = {b2, b1, b0}” is a correct matching, but “{a0, a1, a2} = {b2, b1, b0}” not.

3) Yes, in your example, you can bind port “b2” to constant 1 or 0.

4) Yes, we do not force the contestant to map a bus with another bus. For example, it is acceptable if the contestant finds a solution which map a bus to scalers. Your mapping result in the example is also acceptable.

Q3. Thank you for your response regarding Problem A.
I have a follow-up question about the ordering of buses in Q2(2). In your previous response, you mentioned that the buses are ordered and that the contestant needs to map the LSB in Circuit I to the LSB in Circuit II, and the MSB in Circuit I to the MSB in Circuit II.
Could you please confirm whether the direction of the buses is fixed in the Input File, with the LSB always being in front and the MSB always being at the end?
Thank you for your time and attention. I appreciate any clarification you can provide on this matter.
A3. No. Contestants should not make such assumption. The input/output inside a bus group can be given in any order.

Q4. Problem A is said to be a "large-scale" problem, so I am wondering what is the maximum number of inputs and outputs in a circuit. Thank you very much.
A4. Contestants should not make assumption on the circuit size.

Q5. I am writing to inquire about the bus for Problem A .
In the previous Q&A, it is stated that "it is acceptable if the contestant finds a solution which map a bus to scalers."
My questions are as follows.
1) Is it allowed to only map a bit of a bus to scalers or a bit of other bus. For example, if circuit I has a bus with ports {a0, a1}, and circuit II has a bus with ports {b0, b1}, respectively, is it possible to match a0 to b0, but not match a1 to b1?
2) In the " Introduction" section, it is stated that " If we solve the Boolean matching problems by exhaustive search, the number of permutations could be 5! × 4! = 2880 ... can be significantly reduced to 64." If (1) is allowed, a0 can be mapped to y0, while a1 can be mapped to x1, so it cannot be reduced to only 64, however it may be still 2880. Is this correct?
Thank you.
A5. Here are the answers for these questions.
1) Yes, this matching result is acceptable.

2) It is correct that we allow any matching result if it doesn't violate the given rules. However, in your example, we assume that mapping {a0, a1} to {b0,b1} can be a good enough solution. The contestant can, but not necessary, to search the solution for cross-buses matching.

Q6. While participating in the competition, I have encountered some questions and I would appreciate your assistance.
The question is regarding the paths of the two circuits given in the input file. Are these paths relative to the input file path? For example, if the circuit path only contains the file name, is it located in the same directory as the input file?
Thank you very much for your help!
A6. The circuits file may located in the different directory from the input file. The path for input file and 2 circuit files may be given in absolute path or relative path. For relative path, it starts from the binary executable.

Q7. I have a question about problem A.
Could you please explain why there are 64 permutations in the following example? Thank you.
(Quoted from the problem statement) "However, if we know that there are four buses in the first circuit (Circuit I), {a0, a1}, {b0, b1}, {h0, h1}, {m0, m1}, and four buses in the second circuit (Circuit II), {x0, x1}, {y0, y1}, {u0, u1}, {w0, w1}, the permutations can be significantly reduced to 64."
A7. For input buses, there are 2 different matching result for bus:
{a0, a1} map to {x0, x1}, {b0, b1} map to {y0, y1}
{a0, a1} map to {y0, y1}, {b0, b1} map to {x0, x1}
for any of above, we further concern the permutation of {a0, a1} and {b0, b1}.
As the result, the permutations for input port matching is 2! (bus matching) x 2!(internal signal) x 2! (internal signal) = 8
For the output bus we also have 8 permutations, so the permutations in this example is 8x8=64.
However, polarity (phase) assignment is not shown in this example. The contestant need to assign polarity for each signal in this contest.

Q8. Thank you for your response regarding the paths of the two circuits given in the input file.
I have a follow-up question about the relative paths for the input file and the two circuit files. You mentioned that for relative paths, it starts from the binary executable. I would like to confirm whether "it starts from the binary executable" means that the relative path is based on the bmatch binary, or the current working directory.
For example, if the bmatch path is ~/A/bin/bmatch, and the current working directory is ~/A/, when I call the command ./bin/bmatch ./case.in /case.out in the working directory, will the paths for case.in and case.out be ~/A/bin/case.in and ~/A/bin/case.out, or ./A/case.in and ./A/case.out?
Similarly, I would appreciate clarification on the paths for the two circuit files mentioned in the input file.

Additionally, I have two more questions:
1) Will the gates in the test cases include a buffer (buf)?
2) Besides the NOT gate, are all the gates' inputs in the Verilog file provided expected to have only two inputs? For example, will there be gates like and (out, in1, in2, in3, in4)?

A8. It is more accurate that the relative path is based on the current working directory for all the paths(including input file, circuit files). We usually take the location of binary executable as working dir.
1) Yes, buf will be included in the netlist.
2) All the gates are 2 input (except not and buf)

Q9. I have some questions about problem A.
1) For buses, is it possible to match pins in the same bus to different buses? Like example in figure 2, is it possible for match {a0,x0}, {a1,y1} ?
2) For buses, is it possible to match pins in different order? Like example in figure 2, is it possible for match {a0,x1}, {a1,x0} ?
3) How should I represent pin matching for INGROUP and OUTGROUP ? For example, what does the "1+" in "1+<input_name1>" in figure 4(a) means ?
Thank you.

A9. Here are the answers for these questions.
1) Yes, this matching group is allowed.
2) Yes, this matching group is allowed.
3) ‘1’ (or ‘2’) means the pin comes from circuit I (or circuit II). ‘+’ or ‘-’ denotes to the matching phase(i.e., ‘-‘ means inverted matched).

Q10. I have the following two questions about problem A.
1) It seems that the bus information is only for efficiency and has nothing to do with correctness. Is it true that legal solutions can be obtained without the bus information?
2) Since multiple input ports of Circuit 2 can be mapped to one input port of Circuit 1, does Circuit 2 always have more input ports than Circuit 1?
Thank you.

A10. Here are the answers for these questions.
1) Yes. In the evaluation criteria, we only evaluate if the output groups are equivalence.
2) No. There may exists some extra ports in circuit I.

Q11. 目前團隊在準備A題，認為測資也是十分重要的一環，也才能確保演算法設計方向正確。
想請問A題（Multi-bit Large-scale Boolean Matching）的測資大約何時會釋出呢？
謝謝您的回覆。

A11. The testcases will be released on May 26th.

Q12. I have a follow-up question about Problem A.
If Circuit 1 has more input ports than Circuit 2, then there will be unmatched input ports in Circuit 1.
How is the equivalence determined in this case?
Thank you.
A12. The equivalence is determined by the fanin cone of an output port. If some output ports can't be solved under the matching rule, just leave those output ports unmatched.

Q13. I want to ask the detail about previous question.
If we want to say Input port x in circuit 1, map to port y in circuit 2, should we write as follows?

INGROUP
1+x
2+y
END

Also, should we write a INGROUP-END block for each mapping ?
That is, would there be multiple INGROUP-END block written in output file, or all input matching is written in one block ? Thank you.
A13. Yes. The output file would look like this:

```
====================
INGROUP
1 + n1
2 - n2
END
INGROUP
1 + n3
2 + n4
2 + n5
END
OUTGROUP
1 + n7
2 + n8
END
...
CONSTGROUP
+ n9
+ n10
- n11
END
====================
```

Q14. I'm curious to know if the initial bus defined in the problem can include constants like {a0, a1, 1'b0}. Is this a feasible scenario?
Thank you very much for your help!
A14. Yes, the original design might have something like
==========
wire [7:0]bus;
assign bus={VDD, VDD, VDD, a0, a1, a2, VDD, a3};
==========
In this case, we'll say a0, a1, a2, a3 are in a group of signal in a bus.

Q15. Thank you for the reply. I have one more question as follows.
The formulation of problem A is very different from the traditional Boolean matching problem, which allows many-to-one mapping and accepts partial mapping results.
Could you please give an example of how this can be used or describe where this problem is formulated from?
Thank you.
A15. Thanks for the question. In both industrial and academic, there are lots of problems that need to match a (subset of) design to another design where the designs are highly optimized. Take equivalence checking and ECO for example. The synthesis tools may perform aggressive optimization on the netlist, and some registers or module boundary pins may be optimized as a constant, which may be merged with another reg/pin, or even be removed. In this case, the mapping problem would be difficult unless many-to-one mapping and partial mapping can be achieve. As the result, we are eager for making traditional boolean matching algorithm have the ability to deal with non-exact and projective problems.

Q16. In Problem A, I have some follow-up questions about the bus in Q2(2), Q2(3), and Q7. Thank you.
1) In the previous response, the contestant needs to map the LSB in Circuit I to LSB in Circuit II and map MSB in Circuit I to MSB in Circuit II in the response of Q2(2). However, in the response of Q2(3), it's mentioned that matching such as {a0, a1, 1'b1} = {b0, b1, b2} is legal. Does such matching violate the rule that the LSB matches to the LSB and MSB matches to the MSB? Could you help clarify that?
2) As mentioned, signals in the bus are ordered. However, calculation of the possible permutations for the input port in Q7 has a term labeled as "internal signal" and the formula uses "factorial" to calculate it. Does it mean that the signals are disordered?
3) The last question is: consider that bus signals are ordered, is matching like {a0, c0, 1'b1, c1, a1} = {b0, b1, b2, b3, b4} or {a0, c0, a3, c3} = {b0, b2, b4, b6} valid?
A16. Here are the answers for these questions.
1) The final goal is to get as many equivalence as possible in the primary output.
Assume the function of circuit 1 is {s3,s2,s1,s0}={a2,a1,a0}+{b2,b1,b0} and the function of circuit 2 is {z3,z2,z1,z0}={x2,x1,x0}+{y2,y1,y0}.
The contestant can get equivalence to match {a2,a1,a0} to {x2,x1,x0} and so on.

However, if the function of circuit 1 is {s3,s2,s1,s0}={a2,a1,1'b1}+{b2,b1,b0}, to make the output equivalence, the contestants need to match {a2,a1,1'b1} with {x2,x1,x0}.
2) The bus signal might be meaningful and have certain order in the design, for example it could be an integer or floating number. But they are not sorted in the testcases. The contestants need to match each of the bits inside a bus.
3) The bits of the bus is not given in ordered. The contestant can match a bit from circuit 1 to any of a bit(s) from circuit 2. The final goal is to make primary output equivalence.

Q17. I have a question about problem A. Will the current 10 public cases be included in the final evaluation? Thank you.
A17. We will select part of the public testcases into the final evaluation.

Q18. Could you please tell me if there is testbench or any other way / data to verify the results? I will be highly obliged if I could get some help verifying my results.
A18. The contestants can use some formal verification tools to evaluate their results.

Q19. 您好，想請問的是我們競賽平台上所可以使用的 memory? 以及 Problem A 是否有**特別限制**使用多少 memory？謝謝。
A19.
1) 競賽平台系統配置是 128G，但由於系統機台自動備份機制會使用部分，所以不建議佔用過大。
2) 我們沒有特別限制 memory 的使用，請以平台提供的限制為準。

Q20. I have a question about problem A.
I'm thinking that case8 has an output provided by two input BUS, but CIR1 are provided by more than 3. Is it possible to say that in the case of the same number of BUS provided by the organizer, the node in the same bus will not be only matching a BUS of another circuit. The following conditions will be occurred:
Ex.
Cir1 bus{a1,a2} bus{b1,b2} bus{z1,z2}
Cir2 bus{x1,x2,x3} bus{y1,y2,y3}
Correct answer :
x1->a1
x2->a2
x3->z1
y1->b1
y2->b2
y3->z2
Therefore, if we use BUS, it will mislead our design. Because the two buses of Cir2 must match the three buses of Cir1.

Thank you.

A20. The mapping in your example is legal in this contest. However, there exist a 1 to 1 bus mapping that can achieve the best matching result.

Q21. According to the QA20, then I have the next question.
If there exists a 1 to 1 bus mapping in every case.
I still have a question in case8, whether the n84 output is on the wrong bus.
The output node n29 of cirII is provided by 2 buses, but the output of the other circuit is provided by more than 3 buses.
n29 cannot be matched for all output nodes. It will not be an optimal solution.

A21. We looked into case8 but didn't find the described situation.
1) n84 is not an output, it's an input pin of circuit 1.
2) n29 is an output in circuit 2, however it doesn't belong to any buses.
Making sure we are looking at the same case, the output of Circuit 1 is "n5 , n63 , n78 , n126 , n142 , n144 , n153" and the output of Circuit 2 is "n29 , n76 , n152 , n194 , n224 , n245 , n248".

Q22. According to the QA21, I thought that I had a question for my description.
Actually, What I want to make sure is that the input bus is working right. Because all the output nodes are supported by more than three input buses in CIRI, and I have checked that the output node was simplified.
The meaning is that all output nodes are supported by more than three input buses in CIRII , but I found that the n29 only supported by two buses.
If I put the n84 to another bus in CIRI, it will have some output nodes supported by two buses. it will let n29 match some output nodes.

A22. Thanks for your question and observation. We need to modify the previous answer to " there are indeed some situation that we need to perform matching algorithm across to different buses. "
We take the bus information as an assistance and we allow cross bus matching in this contest.

Q23. I am writing to inquire about the Beta Test of Problem A in the CAD Contest.
I would like to know whether hidden test cases will be included during the Beta Test, in addition to the 10 sets of public test cases.
Understanding the importance of thorough testing, I want to make sure that my solution can perform reliably in various scenarios before the contest's official commencement.
Thank you for your time and attention to this matter.

A23. In Beta evaluation, we only applied the 10 public cases. Hidden cases will apply only in final evaluation.

Q24. I have two questions about the input file:
1). The input of case01 in an alpha test benchmark contains a nonexistent port name: "01", It shold be "b1"?

Is this an error? Or do we allow port names that don't exist?
circuit_1.v
4
2 a0 a1
2 b0 01
2 h0 h1
2 m0 m1
circuit_2.v
4
2 x0 x1
2 y0 y1
2 u0 u1
2 w0 w1
2) What is the practical use of the bus constraint? Can you give me an example?
Thanks ahead!

A24.

1) This is a typo and has been fix in 6/14 testcase update. Please refer to the updated testcase file.
2) We take the bus information as an assistance, not a constraint. In the problem description, we show a case that bus information do help reducing the complexity of Boolean matching problem in introduction:

"For example, Figure 2 shows two designs with five input ports and four output ports, respectively. If we solve the Boolean matching problems by exhaustive search, the number of permutations could be 5! × 4! = 2880. However, if we know that there are four buses in the first circuit (Circuit I), {a0, a1}, {b0, b1}, {h0, h1}, {m0, m1} , and four buses in the second circuit (Circuit II), {x0, x1}, {y0, y1}, {u0, u1}, {w0, w1}, the permutations can be significantly reduced to 64."

Q25. Understood that part of public cases and hidden cases will be selected as the final score. Is it possible to announce the actual number of choices that will be made from the public and hidden? Thank you.

A25. The final evaluation may include 10 public cases and 10 hidden cases. However, this is not confirmed yet.

Q26. I have a question about problem A. If my process is timeout but I have generated a temporary output file during the process. Will the output file be scored although the process is timeout? Thank you!

A26. Yes, we will perform evaluation on the output file no matter how the process is terminated. However, the "temporary output file" should be named as the given output-file name.