The authors study three problems related to the undirected multicut problem. The name “checkpoint problem” is motivated by the aim to minimize the number of ticket checks in public transport such that each path of a given set of paths in a graph is checked at least once. They consider the variant where the maximum number of checks is minimized (MCP) and that where the sum of checks of all paths is minimized (SCP). Furthermore they consider a related problem of finding a path in a graph with forbidden pairs while minimizing the number of violations (PAFP).

For MCP, if the graph is a tree and the paths do not connect different subtrees, they show that there is a simple LP that results in an exact solution and they provide an alternative algorithm with better worst-case running time. If arbitrary paths in trees are allowed, they show that the problem becomes hard: they show an approximation lower bound of 2 and they provide an approximation algorithm that computes a solution of at most 2 \( \text{opt} + 1 \), where \( \text{opt} \) is the value of an optimal solution. Furthermore they present an approximation algorithm for general graphs.

For SCP they show that the problem has the same approximability as the undirected multicut problem. The reduction is straightforward and may be seen as a note for completing the picture.

For PAFP they show a linear lower bound on the approximability.

The main part of the paper is concerned with the problem MCP. The result that the special case of MCP can be solved in polynomial time is due to a natural LP for the problem and the observation that its matrix is TU. The main contribution of section 2 is an alternative combinatorial proof that leads to a better worst-case running time.

Section 3 provides a gap-introducing reduction for MCP in trees that leads to a lower bound on the approximation ratio of 2. The crucial property of MCP that leads to such a high value is that even for large instances the value of an optimal solution can be one, whereas finding a solution with value smaller than two is as hard as solving 1-in-3-SAT.

Overall, the paper is well written and rich in results. I recommend an accept for this paper, but there are some minor issues to be clarified, listed in the following discussion.

(C1) Section 4:

The LP cannot be solved in the dual in polynomial time, because in the dual there are exponentially many variables. Instead, the LP can be solved in polynomial time by using the ellipsoid method in the primal.

To deal with the exponentially many constraints it is necessary to have a separation oracle. This point, however, can easily be fixed by computing the shortest paths for all pairs of end-points of paths using the LP-values as cost: if all paths have a value of at least 1, all constraints of (1) are satisfied; otherwise a too small path gives a concrete violation of (1). Since \( |P| \) is polynomial, (2) is no problem.

**Answer:** Correct. We have added your remark.

(C2) Section 4.2:

There should be a more careful consideration of \( \text{opt} \). I do not immediately see how one should guess \( \text{opt} \). Note that if, as suggested, \( \text{opt} \) is chosen as the value of LP1, it may be larger than the actual optimum value of the problem. This is because LP1 considers all possible paths which is not required for an optimal solution.

**Answer:** We are not sure we understand this comment. In the definition of the checkpoint problem we require that the set of edges chosen forms a multicut (every path in \( Q \) is covered) so the value of LP1 is a lower bound on \( \text{opt} \).

We added a remark regarding how the guessing works.
Section 6:

The label-cover problem was formally introduced by Arora, Babai, Stern, and Sweedyk in 1993.

For the problem 3-SAT-5 there is a constant lower bound on the approximation ratio, but the bound is not 7/8. In [14], the claimed lower bound is shown for E3-SAT. In fact, in Hstad, "On bounded occurrence constraint satisfaction" it is shown that there is a nontrivial approximation for all CPSs with a bounded number of occurrences. The hardness of 3-SAT-5 relies on an expander construction.

This issue affects Theorem 6.1 but not the more important Theorem 6.2 (since the constant c is not given explicitly).

The hardness of bounded degree label cover is well known. I did not find a proper reference, but one can find it for instance in the lecture notes of Guruswami’s lecture 2005 (CSE 533: The PCP Theorem and Hardness of Approximation, Lecture 14: Label Cover Hardness: Application to Set Cover) available in the internet. It does not seem necessary to state that reduction on page 12 since one can start immediately from label cover.

**Answer:** We fixed the references. We do not claim a specific ratio like the tight 7/8 ratio that was claimed before. Rather we rely on the $1 - \delta$ gap of the PCP theorem, with $\delta$ being some universal constant. Applying an idea of Papadimitriou and Yannakakis that uses expanders, this result can be extended to a $1 - \epsilon$ gap for 3-SAT-5 with a universal constant $\epsilon$. We now cite this Papadimitriou and Yannakakis paper in the text. This implies a $1 - \delta/3$ gap for the one-round two-provers, that is based on the Papadimitriou and Yannakakis result. Here we cite the chapter on hardness of approximation of Arora and Lund in the book edited by Hochbaum. This chapter explicitly gives this simple reduction, that translates the gap for 3-SAT-5 to $1 - \epsilon/3$ gap for one-round two-provers protocol and in its Max-Rep representation.

Page 3 in "Our results" there is some unrelated text: "Therefore we are interested in purely combinatorial.

**Answer:** fixed

Page 4, first paragraph of Section 2: Since e is an edge and f(p) is a vertex, they are incident and not adjacent.

**Answer:** fixed

Algorithm APPROXIMATING-MCP, line 1: fraction should be fractional.

**Answer:** fixed
Response to referee report 2 on “The Checkpoint Problem”

The authors examine two checkpoint problems that differ only in the objective function. These problems combine the nature of the minimum membership set cover problem with that of the multicut problem, yielding an interesting combination. More precisely, given a set of paths in a graph between some s-t pairs, the objective is to find a multicut disconnecting all s-t pairs and that minimizes a function on the size of the overlap between the multicut and any single path. When the said function is simply the maximum, we have the Maximum Checkpoint problem (MCP) and when it is the sum we have the sum checkpoint problem (SCP). While the SCP is shown in this paper to be equivalent to the Undirected Multicut problem, the MCP raises many interesting questions. First, the authors give a $O(\sqrt{n \log n/\text{opt}})$-approximation for the MCP problem on general graphs, based on a sphere-growing technique. Solving a previously raised open problem, they then show that, even when the graph is a tree, the problem cannot be approximated within a factor better than 2. Furthermore, they show that this bound is essentially tight, by giving an asymptotic LP-based 2-apx algorithm. Furthermore, for the case that all s-t paths are vertical in the tree, the authors give an exact algorithm that is combinatorial (an obvious but slow LP-based exact algorithm follows from total unimodularity). On the negative side, the authors show a strong hardness result for the related Path with Min forbidden Pairs problem on undirected graphs. They provide a PCP reduction establishing $cn$-inapproximability for any constant $c > 0$ in the classical sense, i.e. unless P=NP. It is remarkable that this strong result is at the same time the first hardness of approximation result on this problem. Furthermore, it also carries over to directed acyclic graphs, establishing the same result for this variant of the problem.

The authors provide a nice set of results on an interesting problem that has natural applications. The techniques they are using are interesting, and especially the hardness results are very strong. My only complaint is that there are many typos and minor errors in this paper that need to be cleaned up. I give a vote for acceptance and provide a list of suggestions to help the authors polish their paper before it is published.

(C1) page 1 abstract: magnitudes $\rightarrow$ magnitude  Answer: fixed

(C2) totally $\rightarrow$ total  Answer: fixed

(C3) (Siam ... $\rightarrow$ [9]  
Answer: Here we are following the convention that abstracts should be stand alone pieces of text, so we prefer to leave the complete reference.

(C4) p 1 intro l 6 : carry small $\rightarrow$ carry a small  
Answer: fixed

(C5) p 2 sec 1.1 par 4 l 4 : restricted to $\rightarrow$ restricted in the way we  
Answer: fixed

(C6) p 3 Def 2 : pair s,t $\rightarrow$ pair (s,t)  
Answer: fixed

(C7) p 3 Our results, l 2 : ”or vice versa” is redundant (by renaming s t)  
Answer: fixed

(C8) p 2: atex: Command not found,  
Answer: fixed
(C9) cost opt + 1 → cost at most opt + 1
   Answer: fixed

(C10) p 3 Our results, p 3 : this special case → this special case, but still on trees,
   Answer: fixed

(C11) nearly → asymptotically
   Answer: fixed

(C12) that the every → that every
   Answer: fixed

(C13) of a set S → of a set S of vertices
   Answer: fixed

(C14) p 4 Sect 2: adjacent to f(p) → incident at f(p)
   Answer: fixed

(C15) a set paths → a set of paths
   Answer: fixed

(C16) 5. return A → 5. return F(A)
   Answer: fixed. We also changed the statement of Lemma 2.1, which referred to what
   the algorithm returns.

(C17) solution: For → solution: for
   Answer: We rather capitalize whole sentences after a colon.

(C18) p 5 Proof of Lem 2.1 : intervals → paths
   Answer: fixed

(C19) : "ancestor" is not so common for edges...
   Answer: we changed the text in several parts so that the ancestor relation is only over
   pairs of vertices.

(C20) : in A are pairwise → in A contained in P are pairwise
   Answer: fixed

(C21) : they could not be disjoint) → would not have been added).
   Answer: fixed

(C22) p 5 : one extra checkpoint than → one checkpoint more than (also, this sentence is a bit
   misleading as the +1 is in the min-max objective)
   Answer: fixed. we changed the wording.

(C23) p 5 Proof of Lem 2.2 : F(A) are there → F(A) there are
   Answer: fixed

(C24) p 6 Sec 3 : gap-inducing → gap-introducing
   Answer: fixed
(C25) p 7 Proof of Thm 3.1 : l 3, after "... ∈ S.” mention that this is well-defined.
Answer: fixed

(C26) : l-4, $(y^3_j, \ell^3_j) \rightarrow (y^3_j, \ell^1_j)$
Answer: fixed

(C27) p 8 Sec 4.1: be consistent in the order of the pairs, i.e. $(s_i, a_i) \rightarrow (a_i, s_i)$ everywhere.
Answer: fixed

(C28) : number of edges can be → number of edges that can be
Answer: fixed

(C29) : (same line) all paths → all paths in the new instance
Answer: fixed

(C30) p 9 : Let us finishing → Let us finish
Answer: fixed

(C31) : ”and $\delta_1 > 0$” is redundant.
Answer: fixed

(C32) : its all neighbors → all its neighbors
Answer: fixed

(C33) p 10 : thought off → thought of
Answer: fixed

(C34) : existing the while → exiting the while
Answer: fixed

(C35) : increases by → increases by at least
Answer: fixed

(C36) p 11 Proof of Thm 4.5 : due to edges to → due to edges added to
Answer: fixed

(C37) : and Lemma 4.2 together → and Lemma 4.3 together
Answer: fixed

(C38) : (next line) $cp_p(Sol) = cp_x(Sol) \rightarrow cp_{Sol}(p) = cp_x(p)$
Answer: fixed

(C39) p 11 Sec 5 denotes the edge → denoted the weight of the edge
Answer: fixed, but we kept present tense

(C40) p 11 Proof of Thm 5.1 : trough → through
Answer: fixed

(C41) : We show that → We show that the
Answer: fixed
(C42) : having capacity \( w(e)/p(e) \) → having weight \( c(e)/p(e) \)

Answer: fixed

(C43) : Thus → Overall

Answer: We are not sure which one you meant. We changed the wording of the last occurrence in that part of the text.

(C44) p 12 Background : We provide hardness → We provide hardness of approximation

Answer: fixed

(C45) : two-provers system → two-prover system

Answer: fixed

(C46) : for a fixed prover into a super-vertices → for the first prover into super-vertices

Answer: fixed

(C47) : (next line) other prover → second prover

Answer: fixed

(C48) p 13 Sec 7 : no-trivial → non-trivial

Answer: fixed