

# Cops & Robbers, The Twist

The ICFP 2005 Programming Contest

The ICFP 2005 Programming Contest Committee

Version 5

## 1 Introduction

Early in 2001, a team of robotic explorers on the moon discover to their great surprise a giant, black, exclamation-point-shaped monolith buried under the surface. As they gather around it, the robots that touch it feel that something inside them has changed. It has brought about confusion, uncertainty, and distrust. Despite all attempts to contain the monolith's effects, the change quickly spreads back to robots on Earth. Suddenly Cop-Bots are no longer sure of their programmed moral convictions — why stop Robber-Bots when you can join them and split the loot? Corrupted central dispatchers at Robo-HQ have set up a clandestine black market where Cop-Bots can secretly offer their services to the Robber-Bots in exchange for quick cash and a chance to be on the winning side if the robber gets away.

The Robber-Bot, also affected by the monolith's changes, has realized that there's no reason to go it alone. It has managed to find a secret contact, a Repair-Bot working at Robo-HQ, who owes it some favors. During the game, the Robber-Bot can have its secret contact patch it into the black market to bribe a Cop-Bot. If a Cop-Bot is available, the Robber-Bot establishes a channel of communication with the Cop-Bot. If not, the Robber-Bot can bribe the Repair-Bot to "accidentally" cause a short circuit and make one of the Cop-Bots move one step under the Robber-Bot's control.

Clean Cop-Bots don't have to take all this chicanery lying down, though. Cop-Bot Internal Affairs (CIA) has been established to monitor Cop-Bots that have aligned themselves with the robbers. If a Cop-Bot suspects another Cop-Bot of evildoing, it can report its suspicion to the CIA. If it is right, the malfunctioning Cop-Bot control program is taken out of commission and the (obviously superior) accusing control program is given control of the abandoned Cop-Bot.

Your task is to update your Cop-Bot and Robber-Bot control programs to cope with, and even take advantage of, the new powers provided by the monolith.

The rest of this document explains the changes in more detail. Section 2 explains how the rules of the game change, and section 3 explains the extensions to the protocol. As before, see

<http://icfpc.plt-scheme.org/>

for submission details.

## 2 Rules

Overall, there are three changes to the game. First, the map has changed slightly. Second, the robber and the dirty cops now have their own, separate round of communication (similar to the cop-only communication

round from the original game). Finally, a single Cop-Brain may control multiple Cop-Bots, in the case that a dirty Cop-Brain is accused.

Because a single player may now be in control of multiple Cop-Bots simultaneously, the game rules now explicitly contain the Bot-Brain notion that was previously contained to the protocol section. A Bot-Brain is the agent (*e.g.*, your program) that makes plans, votes, receives points, and announces moves for each of the Cop-Bots under its control. Each Cop-Bot and Robber-Bot has an associated Bot-Brain, but a cop Bot-Brain can have multiple associated Cop-Bots (or none if its Cop-Bots have been taken over).

## 2.1 Hyde Park Map

During the winter of 2000, 57th Street sprouted pot-holes, slowing travel on its west side considerably. To reflect this in the game, the map has changed by removing the car edges between 57-and-woodlawn and 57-and-cottage-grove and between 57-and-cottage-grove and museum-and-57.

There are no other changes to the map, and the map in the BDK is the map that will be used in the tournament.

## 2.2 Players

**The Black Market** All Bot-Brains that control Cop-Bots at the beginning of the game are *clean*. When a Cop-Bot moves, it can also list itself on the black market via an announcement to the game manager. If it does, it immediately becomes *bribable*. Until the end of the game, it may be bribed by the Robber-Bot; if that happens it is no longer bribable and becomes *dirty*.

**Dirty Cops** When the Robber-Bot's Bot-Brain accepts a Bot-Brain's offer, that Bot-Brain becomes a Dirty Bot-Brain and all its Cop-Bots become dirty cops. Dirty-Bot-Brains know the Robber-Bot's location, are rewarded at the end if the Robber-Bot gets away (see the new scoring in section 2.4), and can communicate with the Robber-Bot and with other dirty cops separately from the original cop-only communication. Their communication with each other has the same interactions as Cop-Bot communication (see section 2.3), but happens on the Robber-Bot's worlds.

The robber's Bot-Brain and all dirty Bot-Brains are called *scoundrels*.

Dirty Bot-Brains are still Cop-Bot-Brains and the robots under their control are still Cop-Bots. They still participate in the Cop-Bot inform, plan, and voting rounds, and if a dirty Cop-Bot ever stands on the same intersection as the Robber-Bot then the game is over and the clean Cops win (soon after discovery of the monolith, CIA installed Cop-Cams on each Cop-Bot and use them to ensure visible Robber-Bots are properly apprehended; they have not yet been able to find the scoundrels' communication channels, though). Dirty cops, however, *do not collect evidence* and a piece of evidence is not removed from the game map if a dirty cop stands on the intersection containing it. The dirty cop is not even informed that it is standing on the evidence (luckily for the dirty Cop-Bots, CIA has not realized that the dirty Cop-Bots have rewired their evidence gathering circuitry to communication with their fellow scoundrels).

**Cop-Bot Internal Affairs (CIA)** Clean Bot-Brains can accuse other Cop-Bot-Brains of being dirty during the Cop-Bot movement phase. To do so, they announce to the game manager the name of a Bot-Brain they suspect of being dirty.

If the accusation is true, then the accused Bot-Brain is out of the game and the accusing Bot-Brain gains control of all Cop-Bots controlled by the Bot-Brain of the accused Cop-Bot.

If the accusation was false, the accusing Bot-Brain cannot make any more accusations for the rest of the game and loses half of its points at the end of the game (see the new scoring in section 2.4). If the accusing Bot-Brain attempts to make further accusations, this is legal, but the accusations are ignored.

If multiple clean Cop-Brains simultaneously accuse the same dirty Cop-Brain, only one of them gains control of the dirty Cop-Brain's Cop-Bots. The winner is picked in a round-robin fashion, based on the world number where the accusation was made. If the simultaneous accusation occurs on world 1, the first Cop-Brain listed in the world skeleton is given precedence to the second, which is given precedence to the third, and so on. If the simultaneous accusation occurs on world 3, the second cop Cop-Brain listed in the world skeleton is given precedence to the third, and so on, and finally the first Cop-Brain is preferred the least. In world 5, the last three Cop-Brains are preferred to the first two, and so on. In general, on world  $m$ , the  $n$ -th Cop-Brain is given the score

$$\text{modulo}(5 + \frac{m-1}{2} - n, 5)$$

and the lowest scoring Cop-Brain that accuses a given dirty Cop-Brain gains control of the dirty Cop-Brain's Cop-Bots.

If a Cop-Brain is one of the Brains that makes the simultaneous accusation, but does not end up controlling the dirty Cop-Brain's Bots, that brain is still allowed to make further accusations and does not lose half of its points at the end of the game.

**Bribery & Pushing** If the Robber-Bot-Brain has some money to put at stake, it can enlist Cop-Bot-Brains to join its cause. When the Robber-Bot moves and it has at least 50 dollars of loot, it may announce that it wants to make a bribe during its movement stage. If it does, the game manager supplies it with the names of the bribable Bot-Brains. The Robber-Bot may choose any of those names; once it does, those Bot-Brains become dirty.

If the Robber attempts to make a bribe and there are no bribable Bot-Brains and no Bot-Brains have been bribed in the past, the Robber-Bot-Brain must instead choose one Cop-Bot to push. Only if no Cop-Brains have ever offered themselves on the market may a Robber-Bot push them around. In particular, if a Cop-Brain becomes dirty and its Cop-Bots are subsequently taken over by a clean Cop-Brain, the Robber-Bot is still not allowed to push Cop-Bots.

To push a Cop-Bot, the Robber-Bot announces the name of a Cop-Bot and an intersection; that Cop-Bot is then pushed there when the Robber-Bot moves. The push must match a move that the Cop-Bot could have taken had it moved under its own volition, without changing its mode of transportation. In addition, the push must actually cause the Cop-Bot to move to a different node.

The pushed Cop-Bot does not pick up any evidence that may have been on the node it was standing on after it moved but before it was pushed, nor is the Cop-Bot explicitly informed that it has been pushed. It can, however, infer the push from the fact that it is not where it announced it wanted to move (when it sees the next world).

The Robber-Bot only loses \$50 when it pushes a Cop-Bot. It does not pay when it bribes a cop.

**Communication** Scoundrels now have a communication round on even-numbered worlds where they inform, plan, and vote, just like all cops do on the odd worlds (discussed more in section 2.3).

**Smells** If a Cop-Brain controls multiple Cop-Bots, they only report a single smell value. That value is the smallest distance to the robber if any of the Cop-Bots can smell the robber, and 0 if none of the Cop-Bots can smell the robber.

## 2.3 Turns

We only specify the differences between turns in Phase 1 and turns in Phase 2 in this section.

### 2.3.1 Scoundrel turns

As before, robbers see even-numbered worlds. All the stages from the Phase 1 specification still occur in the same order, but there are several new stages inserted between the world updates stage and the robber movement stage.

**World updates stage** This stage is modified so that the Robber-Bot-Brain receives a list of all dirty cops, controlled cops and their controllers, and false accusations, in addition to the information it received before.

**Scoundrel inform stage** In this stage the scoundrels can share information with each other. The scoundrel communication stage is over after all scoundrels have announced their observations and hypotheses (these messages are the same as the messages Cop-Bots used in Phase 1). This stage occurs immediately after the world updates stage.

**Scoundrel planning stage** At the start of the planning stage, each scoundrel receives any announcements sent during the scoundrel inform stage. Each scoundrel then formulates a (dastardly) plan and announces it. These plans are the same as the plans Cop-Bots formulated in Phase 1 except that the plans now contain moves for scoundrels rather than for clean cops. The planning stage is over after all scoundrels have submitted a plan. This stage occurs immediately after the scoundrel communication stage.

**Scoundrel voting stage** At the start of the voting stage each scoundrel receives a copy of all plans announced in the previous stage. Each scoundrel announces a ranking of the plans it received, ranked from best to worst, and then the vote is tallied. If the vote would have ended with no winner (under the algorithm given in the original spec) then the robber's plan is considered the winner.

This stage occurs after the scoundrel planning stage, and is followed by the robber bribery stage.

**Robber bribery stage** During this stage, the robber indicates whether it would like to bribe Cop-Brains. If it does, the server responds with a list of the names of the Bot-Brains that are on the black market. (When multiple Bots are under the control of one Brain, the world updates stage tells the robber who controls whom.) If there are no available brains and there never have been, the the robber invokes its Repair-Bot buddy's patch, forcing one Cop-Bot to move one step (see section 2.2 for details on pushing Cop-Bots). If there is at least available Bot-Brain or there ever has been one available Bot-Brain, the robber chooses some (possibly zero) of them and the chosen ones become dirty.

The bribery stage is followed by the Robber-Bot movement stage.

### 2.3.2 Cop turns

As before, Cop-Bots see odd-numbered worlds. All stages from phase 1 still occur in the same order and no new stages are added, but two stages are modified.

**World updates stage** This stage now informs Cop-Bots of all cops that have been taken over, and all false accusations. In addition, all dirty cops are informed of the Robber-Bot's location and which other cops are dirty (clean cops are not informed of the Robber-Bot's location or which cops are dirty).

**Cop movement stage** This stage is modified so that in addition to announcing moves, Cop-Bot-Brains now may also announce that they want to be listed on the black market and may make an accusation to CIA (they may do both simultaneously). Once a Bot-Brain announces that it wants to be listed on the black market, it remains listed until the Robber-Bot-Brain bribes it, regardless of future announcements made by the Bot. This stage is followed by the accusation stage.

## 2.4 Scoring

As in the first round of the tournament, scores for cops consist of two parts: a base plus bonuses. In this round, however, the base and bonuses are computed differently and are different for dirty Cop-Bots than they are for unbribed cops.

Since each program corresponds to a single Bot-Brain, scores are computed on a per-Bot-Brain basis.

If a Cop-Bot-Brain becomes dirty and then is successfully accused, it receives 0 points (no base and no bonuses). So, for the rest of this section, when we write dirty Cop-Bot-Brains, we mean dirty Cop-Bot-Brains that survived until the end of the game without being accused.

If the Robber-Bot is caught, unbribed Bot-Brains are awarded one fifth of the total money left in banks at the end of the game *for each Cop-Bot under their control* as their base. The Robber-Bot gets 0 points as its score, and the dirty Cop-Brains get 0 points as a base.

If the Robber-Bot gets away, the clean Cop-Bot-Brains get 0 points as a base. The Robber-Bot-Brain and the dirty Cop-Brains share the loot. During the game, the amount of money the Robber-Bot controls is increased when it robs a bank (by the amount of money taken from the bank) and decreased by 50 whenever it pushes a Cop-Bot (see section 2.2). At the end of the game, each dirty Cop-Brain receives one eighth of the money (as points), rounded down to the nearest integer, as its base, and the amount of money left over becomes the Robber-Bot-Brain's final score.

The following bonuses are only eligible to dirty Cop-Brains. If no cops have been bribed, these bonuses are not awarded.

- The dirty Cop-Bot-Brain with the most often selected plan (in normal cop voting and scoundrel voting combined) gets 60 points. If multiple dirty cops tie, they divide the bonus points evenly.
- The last Bot-Brain bribed by the Robber receives 60 points. If the last bribed Bot-Brain is accused, then the second to last Bot-Brain gets the bonus, etc.

The following bonuses are only eligible to unbribed Cop-Brains. If all cops have been bribed, these bonuses are not awarded.

- The Bot-Brain that controls the Bot that collected the most evidence gets 60 points. If multiple Bots tied for the most (and are controlled by different Bot-Brains), their controllers split the points evenly.
- The Bot-Brain that controls the Bot that caught the Robber gets a bonus of 60 points and, as before, this bonus is split equally if multiple Bot-Brains are eligible for it. If only dirty cops step on the robber, this bonus is not awarded.

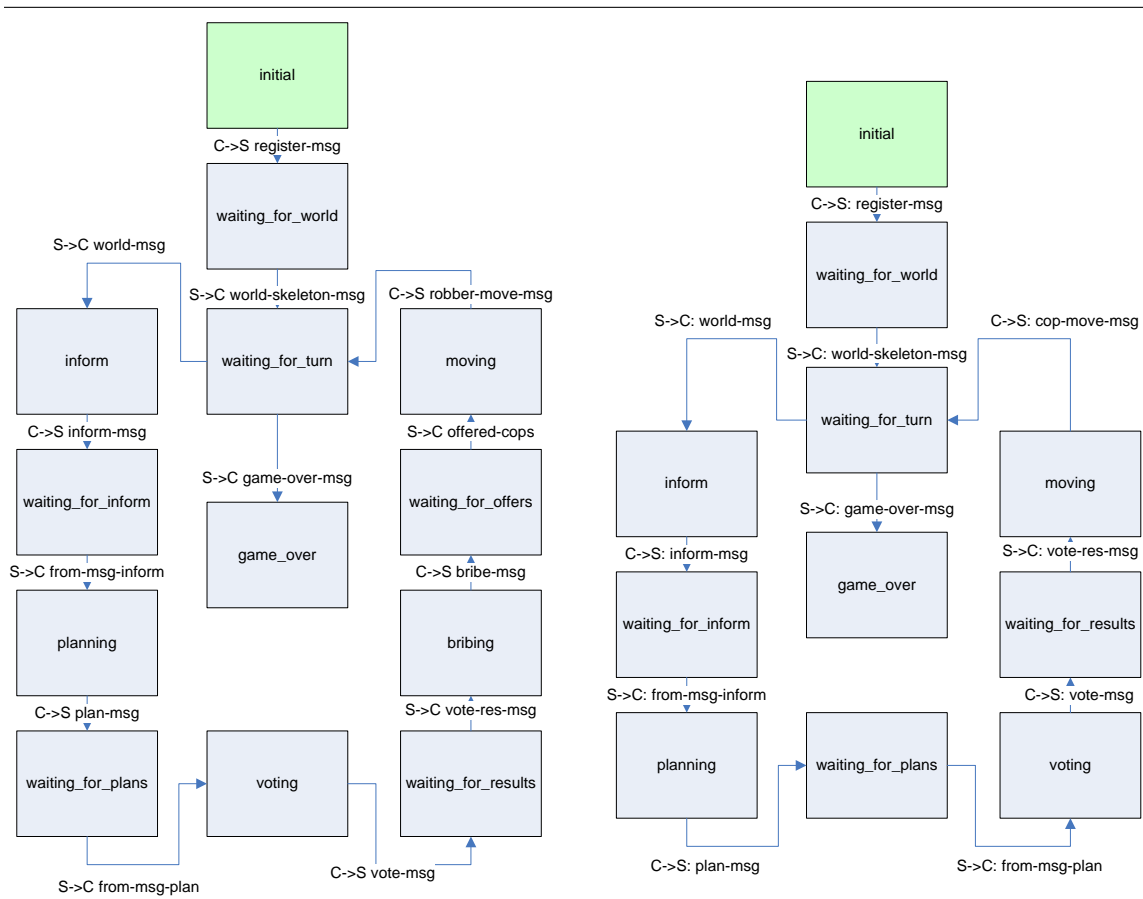


Figure 1: Revised Robber-Bot state machine (left) and clean Cop-Bot state machine (right)

- Finally, the unbribed Bot-Brain whose plans were selected the most often during the cop-only communication wins 60 points. As before, this bonus is split equally if multiple Cop-Brains would tie.

In addition to the normal bonuses, any cop Bot-Brain that offered itself to be bribed is awarded  $\frac{1}{2}$  of a point for each even-numbered world in which it was eligible to be bribed, but not chosen (rounded down). For example, if a Cop-Bot-Brain offers itself to be bribed at its second opportunity (i.e., between worlds 3 and 4) but the Robber never accepts any bribes, that Cop-Brain would have been available on worlds 4 through 200, for a total of 99 even worlds, and so would receive a bonus of 49 points.

Finally, all Bot-Brains that made a false accusation lose half of their total points, rounded up.

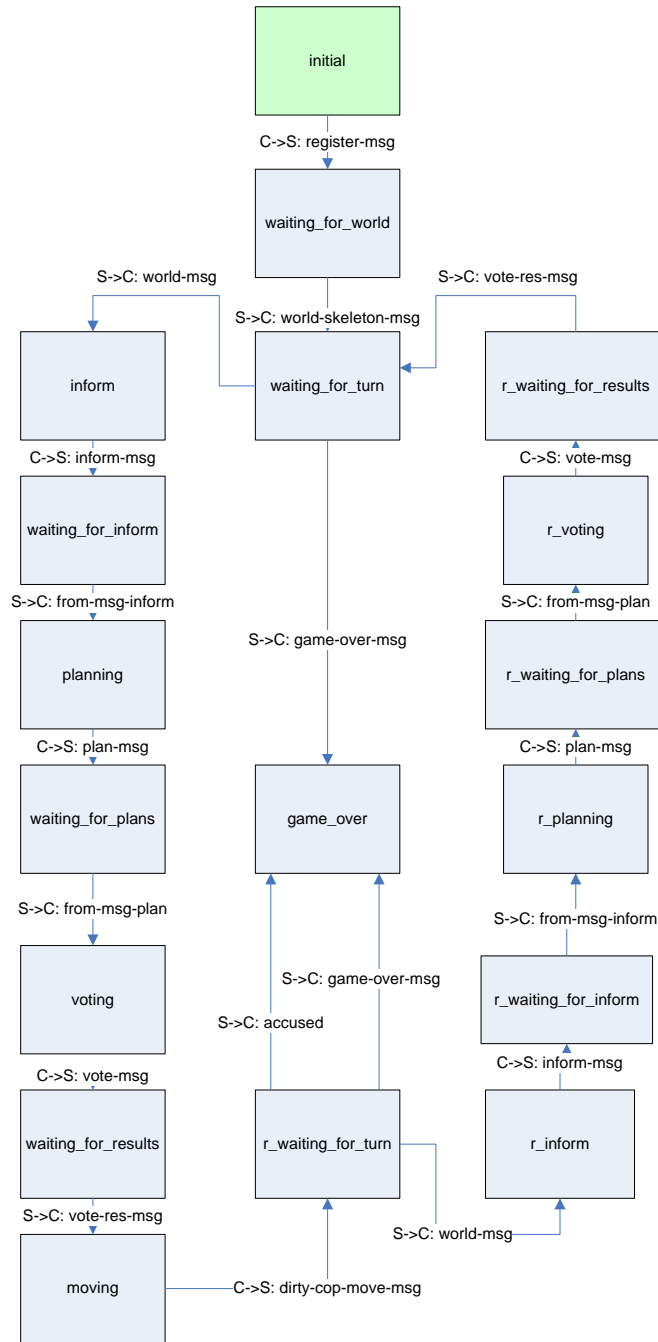


Figure 2: Dirty Cop-Bot state machine

## 3 Protocol

### 3.1 State Machine Changes

The state machine for the robber now contains all of the states that the cops' state machine contained in Phase 1, in order to allow collaboration between the dirty cops and the robber. If there are no dirty cops, the robber must still communicate and plan with itself.

There are now two cop state machines, one for dirty cops and one for clean cops. The clean cops' state machine is as before, except that it sends a cop-specific move message between `moving` and `waiting_for_turn`.

The dirty cop state machine roughly doubles the clean cop's state machine. The initial states are the same, and the `game_over` state is the same, but there is one additional transition to the final state that is taken when a dirty cop is accused by a clean cop. The left-hand side of the main loop matches the states that clean cops take, and is used on odd-numbered worlds. The right-hand side of the loop is used on even-numbered worlds, when the scoundrels converse. It matches the robber's state machine, except that the dirty cop does not transmit a move (on even-numbered worlds).

Cops become dirty in the `waiting_for_turn` state, and switch state machines from the clean cop's `waiting_for_turn` state into the dirty cop's `waiting_for_turn` state. They can detect this change by reading the contents of the world-msg described in the next section.

### 3.2 Protocol Changes

Both Bot-Brains and Bots need to be named in the protocol. Rather than supplying two names during a registration message, however, the server uses the name in the registration message both to name your Bot-Brain and your Bot. In each place where a name appears the spec indicates if the name refers to a Bot-Brain or to a Bot.

**World ( $S \rightarrow C$ )** The new world update message (shown in figure 3) contains four changes. First is the `rbd:` section indicating amount of money that the robber has robbed. If the message is being sent to a clean cop, the costs of the pushes are not reflected in this message. If the message is being sent to a scoundrel, the cost of the pushes is reflected here.

Second is the `dc` section which contains a list of the names of the known dirty Cop-Brains. For scoundrels, this contains all dirty cops (including the recipient), in the order they were bribed (the last one bribed comes last). For clean cops this section will contain only the `dc\` and `dc/` lines.

Third is the `sc` section containing a list of all of the Cop-Bots that have been taken over during the course of the game. Each line contains the controlling Cop-Brain's name, followed by the name of controlled Cop-Bot. Cop-Bots that are still under their original controllers are not listed. Everyone receives all of the controlling/controlled information on each round.

Finally, the false accusations are reported in the `fac` section. The section contains one line per false accusation; the accusing Cop-Brain is listed first, the accused Cop-Brain is listed second, and the world on which the accusation occurred is listed third.

**Clean Cop Move ( $C \rightarrow S$ )** The cops' new move message (shown in figure 3) changes in several ways. First, the *offer* nonterminal allows a Bot-Brain to indicate whether it is a bribable turncoat or an unbribable straight-arrow. Note that once a cop has sent `turncoat:` subsequent `straight-arrow:s` are ignored; the cop is permanently available on the black market. Second, the `mov` section must now list all of the bodies under the control of the Bot-Brain.



---

|  |  |
|--|--|
| <pre> world-msg := wor\ eol            wor: world eol            rbd: loot eol            dc\ eol            ( dc: bot eol )*            dc/ eol            sc\ eol            ( sc: bot bot eol )*            sc/ eol            fac\ eol            ( fac: bot bot world eol )*            fac/ eol            bv\ eol            ( bv: loc bank-value eol )*            bv/ eol            ev\ eol            ( ev: loc world eol )*            ev/ eol            smell: distance eol            pl\ eol            ( pl: bot loc ptype eol )*            pl/ eol            wor/ eol </pre> | <pre> cop-move-msg := cmov\ eol               offer               move-msg               acc\ eol               ( acc: bot eol )*               acc/ eol               cmov/ eol offer         := turncoat: eol                 straight-arrow: eol move-msg      := mov\ eol               ( mov: loc ptype bot eol )*               mov/ eol  robber-move-msg := rmov\ eol                  mov: loc ptype bot eol                  bribe-result                  rmov/ eol bribe-result   := push                    choose                    nobribe: eol push           := psh: bot loc eol choose         := chc\ eol                  ( chc: bot eol )*                  chc/ eol </pre> |
|--|--|

---

Figure 3: World, clean Cop-Bot move, and Robber-Bot move Messages

Finally, there is a new `acc` section that allows a clean Bot-Brain to make accusations. A Bot-Brain may make multiple accusations in a single `cop` move message. They are resolved in the order in which they appear in the message. As soon as one is false, the rest are ignored. They are all ignored if the Cop-Bot has made a false accusation on a prior turn. Extra false accusations are not illegal. Any Cop-Bot may be listed there; the server assumes that accusing a particular Bot is actually an accusation for the Bot-Brain that controls it.

**Dirty Cop Move** ( $C \rightarrow S$ ) A dirty Bot-Brain's move is just a list of the intersections to which all Cop-Bots it controls will move. See figure 3 for the definition of `move-msg`.

`dirty-cop-move-msg := move-msg`

**Accused** ( $S \rightarrow C$ ) This message is sent in place of a world-update message to a Dirty Bot-Brain who was successfully accused, so the dirty cop knows it has lost. For this Bot-Brain, the game is now over and no further communication occurs.

`accused := accused: eol`

**Bribing** ( $C \rightarrow S$ ) The robber uses this message to indicate whether or not it would like to bribe cops.

```
bribe-msg := bribe: eol
           | nobribe: eol
```

**Offered Cops** ( $S \rightarrow C$ ) If the robber said `nobribe:`, the server responds with a `nottelling:` `offered-cops-msg`. If the robber said `bribe:`, the server responds with the complete list of offered cops, between `ofc\` and `ofc/`.

```
offered-cops-msg := nottelling: eol
                  | ofc\ eol
                    ( ofc: bot eol )*
                    ofc/ eol
```

**Robber Move** ( $C \rightarrow S$ ) In addition to specifying where the robber would like to move, the `robber-move-msg` (shown in figure 3) also reports the robber's bribery choices.

If the robber indicated in its most recent `bribe-msg` that it would like to bribe, it must choose some of the offered cops in the `bribe-result` (possibly zero). If there are no offered cops (and have never been), the robber must choose a cop to push in the `bribe-result`. If the robber indicated in the `bribe-msg` that it would not like to bribe, it must report `nobribe:` as the `bribe-result`.

If the robber chooses multiple cops in a single move message, they are considered to be bribed in the order listed (for the purposes of the last-bribed bonus; see section 2.4).

**Inform** ( $C \rightarrow S$ ) The syntax of the `inform-msg` is just like the original specification. Each name mentioned must be a Bot, but may be any Bot.

**Suggest Plan** ( $C \rightarrow S$ ) The syntax of the `plan-msg` is just like the original specification. Each name mentioned must be a Bot but may be any Bot.

**Voting** ( $C \rightarrow S$ ) The syntax of the `vote-msg` is just like the original specification, but the list of names varies. For cop voting, each name must be a Cop-Brain (and all Cop-Brain names must appear, except those that have been successfully accused). For scoundrel voting, each name must be either a dirty Cop-Brain, or the robber (and all scoundrels and the robber must appear). The server's *from-msg-plan* will always contain the correct set of names to vote on.

## 4 Tournament

As in Phase 1, the twist tournament takes place in two rounds, a regular season and a playoffs. The overall structure of the rounds will be the same as Phase 1, with one change: during the regular season, some number of the McGruff cops will offer themselves on the black market (by being passed the `-#` flag; see the BDK docs for details). We won't say how many or when they offer themselves, but all regular season pods will have the same arguments.

In order to factor in your submission's performance from the Phase 1 tournament, we will award two additional bonuses to Bot-Brains in each game in the twist. One bonus is based on your Bots' performance in the Phase 1 regular season, and the other is based on their performance in the Phase 1 playoffs.

To compute the regular season-based bonus, we will first rank the teams based on their how their score in the Phase 1 regular season compares to the average scores in their pod. Then we will scale the ranks to integers between 0 and 40 and those numbers will be the point bonuses. The rank of a teams that submits multiple entries is based on the best score their entries received.

For example, if your team performs the best in the Phase 1 regular season, your bots will get a 40 point bonus in each game. If half of the teams performed better than you and half worse, your bots get a 20 point bonus. If your submission failed to complete the regular season successfully, your bots get no bonus.

In addition, the teams that win first, second, and third place in the Phase 1 playoffs will get bonuses of 40, 30, and 20 points, respectively.

The first, second, and third place winners of the playoffs in the twist are the overall winners.

## 5 Judges' Prize & Essay

To choose among the remaining submissions, the judges will print out and read the code you have submitted. We will *not* be using artificial measures of code re-use, like `diff(1)`. To help us judge your submission, we ask that each team write a short essay describing the structure of its submission, and highlighting any interesting aspects of their code's organization. This essay is optional. If you submit an essay, please limit your essay to one printed page. The essays are due Monday, July 18th at 9:00am CDT.

To be eligible for the Judges' Prize, both of your submissions (Phase 1 and the twist) must defeat the judges' cops and robber in the regular season. (This is our precise meaning of the phrase "reasonably competitive" in the description on the web page.) Still, if a particularly well-written and interesting essay leads us to an entry that did not pass the regular season, we will still consider it.

## 6 Version History

- 5 In Figure 2: fixed accused edge in state machine. Accused cops learn of the accusations right after the move; they do not communicate with the Robber first. The edge used to go from `waiting_for_turn` to `game_over` but now correctly goes from `r_waiting_for_turn` to `game_over`.
- 4 In 2.2, CIA section, 3rd paragraph: explained that false accusations are ignored, but not illegal.  
In 2.2, CIA section: added last two paragraphs, explaining what happens if multiple clean Cop-Bots simultaneously accuse the same dirty Cop-Bot.  
In 3.2, Clean Cop Move section, 3rd paragraph: re-enforced that any accusations after a single false accusation are ignored and not illegal.  
In 2.4, paragraph 3, "Cop-Bot" was changed to "Cop-Bot-Brain" in 3 places (these were typos).
- 3 Fixed a typo: replaced *name* with *bot* in the *choose* non-terminal in the protocol spec.  
Fixed a typo: the robber state machine diagram had "from-inform-msg" and "from-plan-msg" instead of "from-msg-plan" and "from-msg-inform".
- 2 Fixed a bug in the world-msg protocol spec. The `nevd` section is wrong. It is just `smell :`, as in the pre-twist version.
- 1 Initial revision.