

Kerbal Space Program - Feedback #7440

What's wrong on maneuver nodes and how to improve it

03/22/2016 05:25 PM - Kasuha

Status:	Needs Clarification	
Severity:	Normal	
Assignee:		
Category:	Gameplay	
Target version:		
Version:	1.0.5	Language: English (US)
Platform:	Windows	Mod Related: No
Expansion:		

Description

I spent some time figuring out how maneuver nodes and maneuver/burn indicators on navball work. I may be wrong about it in details but I am almost certain that there's some truth on it - and if it's not, then it's probably even worse.

The result of my testing is this:

To display the maneuver indicator on navball, the game calculates position of the ship at T+0, calculates its motion vector, subtracts it from absolute motion vector of the maneuver (i.e. sum of actual orbital speed and impulse added by maneuver handles) and displays the resulting difference as dv of the maneuver, and blue maneuver indicator on navball.

At the time preceding the maneuver up to T+0 it's okay. It causes some strange effects on highly elliptical orbits or in very low speed orbits where physics introduce too much noise into the trajectory, but for general use it's probably the best solution.

After T+0 it's wrong, wrong, wrong. It keeps calculating past position of the ship on its current orbit and displaying the velocity difference at that point. But as the orbit gets curved by gravity in long burns, both the dv and the direction get misleading and do not help matching the planned course. Instead, they lead to increasing the difference.

For better understanding what should be done it's important to notice that executing a burn like the one in KSP is pretty much like doing a rendezvous. There's an imaginary ship coasting on the planned orbit and it will be exactly at the maneuver node at T+0. After T+0 it will continue coasting along the planned orbit. To execute the maneuver, the player needs to match position and velocity with that imaginary ship.

To help executing the maneuver precisely, the game should support doing rendezvous with that virtual ship.

That means, after T+0 it should actually deploy a beacon at the maneuver node, and let it coast ("on rails") along the planned orbit. At the very least, the maneuver indicator should display velocity difference between this beacon and the ship, instead of projecting the ship's position and velocity to the past.

And to make things even better, it would be very nice if the game supported a "Maneuver" navball mode, which would be the same as target mode, except it would be relative to this beacon. Target/antitarget could point directly at the beacon, while the maneuver indicator and retrograde point would both show velocity difference. The beacon could even be visible both in map and in normal view, just like a targeted ship.

That would allow players to match the planned maneuver with almost absolute precision (if they choose to) using essentially the same approach as in rendezvous.

History

#1 - 03/25/2016 09:08 AM - Kasuha

After thinking about it more I have an idea that might potentially improve the behavior without introducing new elements like Maneuver mode and the beacon.

The maneuver dv display should still display dv difference between current velocity of the ship, and projected velocity vector where the ship should be. Displaying that value for T+0 if T+0 is in the past gets misleading, if the ship does not correct its speed the dv difference grows and the player should be able to see that.

The major change would be in the thrust vector on navball. The problem with current approach is that it only minimizes velocity difference at T+0. Spatial difference, i.e. position of the ship's projected T+0 position relative to maneuver position is important too (just like with rendezvous).

The idea is to introduce a metric - for starters, just sum of the velocity difference and the position difference (still projected to  $T+0$ ). For given time, ship's position and speed, that would give a number. The direction displayed on navball would be the direction that, if the ship accelerates there, leads to greatest reduction of this number. It's a 2D optimization problem, finding a maximum on the metric function where direction of acceleration is the parameter.

I have not done the math on it yet so I don't know if this can really work or how exactly will it behave. It might end up behaving unintuitively or in a confusing way, especially if players would expect they need to put all that displayed  $dv$  in the indicated direction (which is not true). But it has potential of navigating the player to the right course without introducing new interface options. And it could be used to improve the experience in actual rendezvous, too.

**#2 - 07/17/2016 09:44 AM - TriggerAu**

- Status changed from New to Needs Clarification