**说 明 书 摘 要**

本发明公开了一种基于深度强化学习的车辆跟驰速度控制方法，包括：基于真实驾驶数据集，提取“引导车—自动驾驶车辆—跟驰车”三车模式下的自动驾驶跟驰对；设计马尔可夫奖励过程；基于车辆跟驰安全性及舒适度的要求，设计强化学习奖励函数；构建基于深度强化学习的车辆跟驰速度控制网络框架；对所得到的网络进行训练，得到基于深度强化学习的车辆跟驰速度控制模型；对所得到的网络模型进行测试，验证模型的安全性及舒适度。本发明缓解了传统深度强化学习“试错”与自动驾驶安全性之间的矛盾。

**摘 要 附 图**



**权 利 要 求 书**

1.一种基于深度强化学习的车辆跟驰速度控制方法，其特征在于，包括以下步骤：

步骤1：基于真实数据集，提取“引导车—自动驾驶车辆—跟驰车”三车模式下的自动驾驶跟驰对；

步骤2：设计马尔可夫奖励过程；

步骤3：构建基于深度强化学习的车辆跟驰速度控制网络框架；

步骤4：对经过步骤1，2，3的网络进行训练，得到基于深度强化学习的车辆跟驰速度控制模型；

步骤5：对经过步骤5的网络模型进行测试，验证模型的安全性及舒适度。

2.根据权利要求1所述基于深度强化学习的车辆跟驰速度控制方法，其特征在，所述步骤1中的具体按以下步骤实施：

步骤1.1：基于真实数据集，选择小轿车为研究对象；

步骤1.2：选择引导车，跟驰车等在不改变车道的情况下保持跟车状态；

步骤1.3：选择跟车轨迹片段长度不小于15秒。

3. 根据权利要求1所述基于深度强化学习的车辆跟驰速度控制方法，其特征在于，所述步骤2中的具体按以下步骤实施：

步骤2.1：设计构造马尔可夫奖励过程，采用简化的MDP，由一个四元组组成：，其中代表马尔可夫决策过程，代表状态空间，代表动作空间，代表奖励函数，代表折扣因子；

步骤2.2：设计马尔可夫奖励过程的状态空间，由于三车模式之间的相互作用主要基于运动学过程，因此自动驾驶车辆的跟车过程的状态空间被定义为一个向量，，

其中，代表自动驾驶车辆与引导车之间的速度差，代表自动驾驶车辆与跟驰车之间的速度差，代表自动驾驶车辆与引导车之间的相对距离，代表自动驾驶车辆与跟驰车之间的相对距离；

步骤2.3：设计马尔可夫奖励过程的动作空间，由于自动驾驶车辆纵向速度控制主要基于加速度，因此本方法中动作空间为自动驾驶车辆加速度；

步骤2.4：设计马尔可夫奖励过程的奖励函数。

4. 根据权利要求1所述基于深度强化学习的车辆跟驰速度控制方法，其特征在于，所述步骤2.4中马尔可夫奖励过程的奖励函数的设计，具体步骤如下：

步骤2.4.1：当自动驾驶车辆的速度不高于前车速度，也不低于后车速度时，是最安全的情况，其奖励函数为：



步骤2.4.2：当自动驾驶车辆的速度不高于前车速度，且低于后车速度时，其奖励函数为：



其中，为自动驾驶车辆与后车的停车距离；

步骤2.4.3：当自动驾驶车辆的速度高于前车速度，且不低于后车速度时，其奖励函数为：



其中，为自动驾驶车辆与后车的停车距离；

步骤2.4.4：当自动驾驶车辆的速度高于前车速度低于后车速度，且只有后车与自动驾驶车辆的距离大于停车距离时，其奖励函数为：



步骤2.4.5：当自动驾驶车辆的速度高于前车速度低于后车速度，且只有前车与自动驾驶车辆的距离大于停车距离时，其奖励函数为：



步骤2.4.6：当自动驾驶车辆的速度高于前车速度低于后车速度，且前车与自动驾驶车辆的距离及后车与自动驾驶车辆的距离均大于停车距离时，根据相对距离和相对速度的商来确定防撞优先级；

①时，奖励函数为：



②时，奖励函数为：



在本方法中，自动驾驶车辆需要在引导车和跟驰车之间的防撞做出平衡决策，即如何加速和减速以保持三车模式而不发生碰撞。奖励是根据不同的状态和选择的行动来获得的，奖励的减少是根据车辆之间安全性的降低程度来设置的。

5. 根据权利要求1所述基于深度强化学习的车辆跟驰速度控制方法，其特征在于，所述步骤3的具体实施步骤：

步骤3.1：建立经验池Replay Buffer；

步骤3.1.1：建立经验池Replay Buffer，将自动驾驶车辆与环境互动的数据分解为轨迹存储到Replay Buffer中；

步骤3.1.2：随后每次训练从Replay Buffer中随机采样一个minibatch，进行网络训练；

步骤3.2：建立策略网络即Actor网络；

步骤3.2.1：建立当前Actor网络，网络参数为；

步骤3.2.2：建立目标Actor网络，并初始化网络参数；

步骤3.2.3：将从Replay Buffer中随机采样得到的数据minibatch输入网络，并进行目标策略平滑。因为策略网络存在过拟合的情况，所以引入正则项即高斯噪声来平滑目标策略网络，具体实现如下：



其中，为可能的动作范围；

步骤3.3：建立价值网络即Critic网络；

步骤3.3.1：建立两个独立的当前Critic网络，其中两个网络的参数为和；

步骤3.3.2：建立两个独立的目标Critic网络，并初始化网络参数，；

步骤3.3.3：将从Replay Buffer中随机采样得到的数据minibatch输入和网络进行训练，并更新目标。考虑在实际应用中Critic网络会产生高估，因此在更新目标时，选择目标网络中较小的目标来更新，以此来缓解Critic网络带来的高估。下面为更新的目标：



步骤3.3.4：用上式分别与及做均方差，然后作为损失函数，来更新当前Critic网络：



步骤3.4：延迟目标网络更新；

步骤3.4.1：当Critic的估值不准确的时候，Actor就会朝着错误的方向更新。所以在网络更新时设置Actor网络更新速慢于Critic，即当前Critic网络每更新两次，才更新当前Actor网络，目标Actor网络以及目标Critic网络；

步骤3.4.2：通过确定性策略梯度来更新当前Actor网络：



步骤3.4.3：对目标网络进行软更新，来稳定目标网络训练，更新方式为：





6. 根据权利要求1所述基于深度强化学习的车辆跟驰速度控制方法，其特征在于，所述步骤4的具体实施步骤：

步骤4.1：利用提取“引导车—自动驾驶车辆—引导车”三车模式跟驰对。并将提取的跟驰对随机抽样80%作为训练集，20%作为测试集；

步骤4.2：设置Adam优化器，学习率设为3e-4，折扣回报率为0.99，Replay Buffer大小为10e6，minibatch 大小为256；

步骤4.3：将提取跟驰对输入网络，对网络进行训练，每200epoch之后计算当前自动驾驶车辆的避碰率及平均奖励；

步骤4.4：训练完成后保存模型；

7. 根据权利要求1所述基于深度强化学习的车辆跟驰速度控制方法，其特征在于，所述步骤5的具体实施步骤：

步骤5.1：加载训练好的模型，读取测试数据集；

步骤5.2：计算自动驾驶车辆在测试数据集的避碰率及平均奖励；

步骤5.3：为了证明模型的驾驶安全，对自动驾驶车辆的碰撞时间TTC进行检验，并与当前人类驾驶员进行对比。其中，TTC计算公式如下：



步骤3.4：为了证明模型的驾驶效率，对自动驾驶车辆的车头时距TH进行检验，并与当前人类驾驶员进行对比。其中，TH计算公式如下：



**说 明 书**

**一种基于深度强化学习的车辆跟驰速度控制方法**

**技术领域**

本方法属于微观交通流自动驾驶车辆跟驰领域，具体涉及一种基于深度强化学习的车辆跟驰速度控制方法。

**背景技术**

近年来我国城镇化建设不断推进，城市居民的出行需求大幅增加。出行量的增加以及机动车数量的增长给城市道路安全、能源问题带来了极大的压力。因此，缓解交通拥堵及提高交通系统的安全性是当前迫切需要解决的问题。面对当前这些复杂交通场景，自动驾驶技术的出现为解决以上问题提供了新的思路。相对于有人驾驶，自动驾驶汽车在减少交通事故、缓解交通压力、更好的适应人群、减少空气污染、减低驾驶人力成本和时间成本等方面更具优势。自动驾驶技术是一个复杂的多学科融合技术，许多学者试图将车辆跟驰模型作为车辆自适应巡航控制策略。对车辆跟驰行为进行研究，并建立车辆跟驰模型，能够对跟驰车辆之间的相互作用做出量化分析，揭示出单一车道的交通流特性，为交通系统的模拟与仿真奠定基础。在相关技术手段成熟的情况下，车辆跟驰技术能够代替驾驶员完成跟驰决策及操作过程，实现车辆的自动跟驰驾驶，从而使得道路中车辆的通行更为高效，并有效避免驾驶员的失当操作以及追尾、碰撞等交通事故的发生。

**发明内容**

本发明的目的是提供一种基于深度强化学习的车辆跟驰速度控制方法，解决理论驱动型车辆跟驰模型存在参数标定复杂，关注问题单一，通用性较低等问题。

本发明提供一种基于深度强化学习的车辆跟驰速度控制方法，包括以下步骤：

步骤1：基于真实数据集，提取“引导车—自动驾驶车辆—跟驰车”三车模式下的自动驾驶跟驰对；

步骤2：设计马尔可夫奖励过程；

步骤3：构建基于深度强化学习的车辆跟驰速度控制网络框架；

步骤4：对经过步骤1，2，3的网络进行训练，得到基于深度强化学习的车辆跟驰速度控制模型；

步骤5：对经过步骤5的网络模型进行测试。

进一步的，所述步骤1的具体实现步骤为：

步骤1.1：基于真实数据集，选择小轿车为研究对象；

步骤1.2：选择引导车，跟驰车等在不改变车道的情况下保持跟车状态；

步骤1.3：选择跟车轨迹片段长度不小于15秒。

进一步的，所述步骤2的具体实现步骤为：

步骤2.1：设计构造马尔可夫奖励过程，采用简化的MDP，由一个四元组组成：，其中代表马尔可夫决策过程，代表状态空间，代表动作空间，代表奖励函数，代表折扣因子；

步骤2.2：设计马尔可夫奖励过程的状态空间，由于三车模式之间的相互作用主要基于运动学过程，因此自动驾驶车辆的跟车过程的状态空间被定义为一个向量，，

其中，代表自动驾驶车辆与引导车之间的速度差，代表自动驾驶车辆与跟驰车之间的速度差，代表自动驾驶车辆与引导车之间的相对距离，代表自动驾驶车辆与跟驰车之间的相对距离；

步骤2.3：设计马尔可夫奖励过程的动作空间，由于自动驾驶车辆纵向速度控制主要基于加速度，因此本方法中动作空间为自动驾驶车辆加速度；

步骤2.4：设计马尔可夫奖励过程的奖励函数。

步骤2.4.1：当自动驾驶车辆的速度不高于前车速度，也不低于后车速度时，是最安全的情况，其奖励函数为：



步骤2.4.2：当自动驾驶车辆的速度不高于前车速度，且低于后车速度时，其奖励函数为：



其中，为自动驾驶车辆与后车的停车距离；

步骤2.4.3：当自动驾驶车辆的速度高于前车速度，且不低于后车速度时，其奖励函数为：



其中，为自动驾驶车辆与后车的停车距离；

步骤2.4.4：当自动驾驶车辆的速度高于前车速度低于后车速度，且只有后车与自动驾驶车辆的距离大于停车距离时，其奖励函数为：



步骤2.4.5：当自动驾驶车辆的速度高于前车速度低于后车速度，且只有前车与自动驾驶车辆的距离大于停车距离时，其奖励函数为：



步骤2.4.6：当自动驾驶车辆的速度高于前车速度低于后车速度，且前车与自动驾驶车辆的距离及后车与自动驾驶车辆的距离均大于停车距离时，根据相对距离和相对速度的商来确定防撞优先级；

①时，奖励函数为：



②时，奖励函数为：



在本方法中，自动驾驶车辆需要在引导车和跟驰车之间的防撞做出平衡决策，即如何加速和减速以保持三车模式而不发生碰撞。奖励是根据不同的状态和选择的行动来获得的，奖励的减少是根据车辆之间安全性的降低程度来设置的。

进一步的，所述步骤3的具体实现步骤为：

步骤3.1：建立经验池Replay Buffer；

步骤3.1.1：建立经验池Replay Buffer，将自动驾驶车辆与环境互动的数据分解为轨迹存储到Replay Buffer中；

步骤3.1.2：随后每次训练从Replay Buffer中随机采样一个minibatch，进行网络训练；

步骤3.2：建立策略网络即Actor网络；

步骤3.2.1：建立当前Actor网络，网络参数为；

步骤3.2.2：建立目标Actor网络，并初始化网络参数；

步骤3.2.3：将从Replay Buffer中随机采样得到的数据minibatch输入网络，并进行目标策略平滑。因为策略网络存在过拟合的情况，所以引入正则项即高斯噪声来平滑目标策略网络，具体实现如下：



其中，为可能的动作范围；

步骤3.3：建立价值网络即Critic网络；

步骤3.3.1：建立两个独立的当前Critic网络，其中两个网络的参数为和；

步骤3.3.2：建立两个独立的目标Critic网络，并初始化网络参数，；

步骤3.3.3：将从Replay Buffer中随机采样得到的数据minibatch输入和网络进行训练，并更新目标。考虑在实际应用中Critic网络会产生高估，因此在更新目标时，选择目标网络中较小的目标来更新，以此来缓解Critic网络带来的高估。下面为更新的目标：



步骤3.3.4：用上式分别与及做均方差，然后作为损失函数，来更新当前Critic网络：



步骤3.4：延迟目标网络更新；

步骤3.4.1：当Critic的估值不准确的时候，Actor就会朝着错误的方向更新。所以在网络更新时设置Actor网络更新速慢于Critic，即当前Critic网络每更新两次，才更新当前Actor网络，目标Actor网络以及目标Critic网络；

步骤3.4.2：通过确定性策略梯度来更新当前Actor网络：



步骤3.4.3：对目标网络进行软更新，来稳定目标网络训练，更新方式为：





进一步的，所述步骤4的具体实现步骤为：

步骤4.1：利用提取“引导车—自动驾驶车辆—引导车”三车模式跟驰对。并将提取的跟驰对随机抽样80%作为训练集，20%作为测试集；

步骤4.2：设置Adam优化器，学习率设为3e-4，折扣回报率为0.99，Replay Buffer大小为10e6，minibatch 大小为256；

步骤4.3：将提取跟驰对输入网络，对网络进行训练，每200epoch之后计算当前自动驾驶车辆的避碰率及平均奖励；

步骤4.4：训练完成后保存模型；

进一步的，所述步骤5的具体实现步骤为：

步骤5.1：加载训练好的模型，读取测试数据集；

步骤5.2：计算自动驾驶车辆在测试数据集的避碰率及平均奖励；

步骤5.3：为了证明模型的驾驶安全，对自动驾驶车辆的碰撞时间TTC进行检验，并与当前人类驾驶员进行对比。其中，TTC计算公式如下：



步骤5.4：为了证明模型的驾驶效率，对自动驾驶车辆的车头时距TH进行检验，并与当前人类驾驶员进行对比。其中，TH计算公式如下：



与现有技术相比，本发明构建“引导车—自动驾驶车—跟驰车”三车模式下的基于双Critic神经网络模块的自动驾驶车辆跟驰模型，该模型可以使驾驶员更加关注周围车辆状态变化的同时，减少Critic网络可能出现的高估问题，提高自动驾驶车辆的安全性。并且，根据三车模型下车辆的速度车间距等划分安全等级设计强化学习奖励函数，加速网络收敛，缓解传统强化学习“试错”与自动驾驶安全性之间的矛盾。

**附图说明**

图1是本发明一种基于深度强化学习的车辆跟驰速度控制方法所依据的流程图；

图2是三车模式图

图3是本发明一种基于深度强化学习的车辆跟驰速度控制方法的网络框架图；

图4是Replay Buffer经验产生图；

图5是Actor网络模块图；

图6是Critic网络模块图；

图7是模型训练时的模型避碰率。

**具体实施方式**

下面结合附图及实施例对本发明做进一步详细说明。

参见附图1，本发明基于最优速度的车辆跟驰模型及其安全性分析方法包括以下步骤：

步骤1：基于德国真实驾驶数据集HighD，提取“引导车—自动驾驶车辆—跟驰车”三车模式下的自动驾驶跟驰对，三车模式如附图2所示；

步骤1.1：基于德国真实驾驶数据集HighD，选择小轿车为研究对象；

步骤1.2：选择引导车，跟驰车等在不改变车道的情况下保持跟车状态；

步骤1.3：选择跟车轨迹片段长度不小于15秒。

步骤2：设计马尔可夫奖励过程；

步骤2.1：设计构造马尔可夫奖励过程，采用简化的MDP，由一个四元组组成：，其中代表马尔可夫决策过程，代表状态空间，代表动作空间，代表奖励函数，代表折扣因子；

步骤2.2：设计马尔可夫奖励过程的状态空间，由于三车模式之间的相互作用主要基于运动学过程，因此自动驾驶车辆的跟车过程的状态空间被定义为一个向量，，

其中，代表自动驾驶车辆与引导车之间的速度差，代表自动驾驶车辆与跟驰车之间的速度差，代表自动驾驶车辆与引导车之间的相对距离，代表自动驾驶车辆与跟驰车之间的相对距离；

步骤2.3：设计马尔可夫奖励过程的动作空间，由于自动驾驶车辆纵向速度控制主要基于加速度，因此本方法中动作空间为自动驾驶车辆加速度；

步骤2.4：设计马尔可夫奖励过程的奖励函数。

步骤2.4.1：当自动驾驶车辆的速度不高于前车速度，也不低于后车速度时，是最安全的情况，其奖励函数为：



步骤2.4.2：当自动驾驶车辆的速度不高于前车速度，且低于后车速度时，其奖励函数为：



其中，为自动驾驶车辆与后车的停车距离；

步骤2.4.3：当自动驾驶车辆的速度高于前车速度，且不低于后车速度时，其奖励函数为：



其中，为自动驾驶车辆与后车的停车距离；

步骤2.4.4：当自动驾驶车辆的速度高于前车速度低于后车速度，且只有后车与自动驾驶车辆的距离大于停车距离时，其奖励函数为：



步骤2.4.5：当自动驾驶车辆的速度高于前车速度低于后车速度，且只有前车与自动驾驶车辆的距离大于停车距离时，其奖励函数为：



步骤2.4.6：当自动驾驶车辆的速度高于前车速度低于后车速度，且前车与自动驾驶车辆的距离及后车与自动驾驶车辆的距离均大于停车距离时，根据相对距离和相对速度的商来确定防撞优先级；

①时，奖励函数为：



②时，奖励函数为：



在本方法中，自动驾驶车辆需要在引导车和跟驰车之间的防撞做出平衡决策，即如何加速和减速以保持三车模式而不发生碰撞。奖励是根据不同的状态和选择的行动来获得的，奖励的减少是根据车辆之间安全性的降低程度来设置的。

步骤3：构建基于深度强化学习的车辆跟驰速度控制网络框架，如附图3所示；

步骤3.1：建立经验池Replay Buffer，经验产生过程如附图4所示；

步骤3.1.1：建立经验池Replay Buffer，将自动驾驶车辆与环境互动的数据分解为轨迹存储到Replay Buffer中；

步骤3.1.2：随后每次训练从Replay Buffer中随机采样一个minibatch，进行网络训练；

步骤3.2：建立策略网络即Actor网络，如附图5所示；

步骤3.2.1：建立当前Actor网络，网络参数为；

步骤3.2.2：建立目标Actor网络，并初始化网络参数；

步骤3.2.3：将从Replay Buffer中随机采样得到的数据minibatch输入网络，并进行目标策略平滑。因为策略网络存在过拟合的情况，所以引入正则项即高斯噪声来平滑目标策略网络，具体实现如下：



其中，为可能的动作范围；

步骤3.3：建立价值网络即Critic网络，如附图6所示；

步骤3.3.1：建立两个独立的当前Critic网络，其中两个网络的参数为和；

步骤3.3.2：建立两个独立的目标Critic网络，并初始化网络参数，；

步骤3.3.3：将从Replay Buffer中随机采样得到的数据minibatch输入和网络进行训练，并更新目标。考虑在实际应用中Critic网络会产生高估，因此在更新目标时，选择目标网络中较小的目标来更新，以此来缓解Critic网络带来的高估。下面为更新的目标：



步骤3.3.4：用上式分别与及做均方差，然后作为损失函数，来更新当前Critic网络：



步骤3.4：延迟目标网络更新；

步骤3.4.1：当Critic的估值不准确的时候，Actor就会朝着错误的方向更新。所以在网络更新时设置Actor网络更新速慢于Critic，即当前Critic网络每更新两次，才更新当前Actor网络，目标Actor网络以及目标Critic网络；

步骤3.4.2：通过确定性策略梯度来更新当前Actor网络：



步骤3.4.3：对目标网络进行软更新，来稳定目标网络训练，更新方式为：





步骤4：对经过步骤1，2，3的网络进行训练，得到基于深度强化学习的车辆跟驰速度控制模型；

步骤4.1：利用提取“引导车—自动驾驶车辆—引导车”三车模式跟驰对。并将提取的跟驰对随机抽样80%作为训练集，20%作为测试集；

步骤4.2：设置Adam优化器，学习率设为3e-4，折扣回报率为0.99，Replay Buffer大小为10e6，minibatch 大小为256；

步骤4.3：将提取跟驰对输入网络，对网络进行训练，每200epoch之后计算当前自动驾驶车辆的避碰率及平均奖励；

步骤4.4：训练完成后保存模型，模型在训练过程中的避碰率如附图7所示；

步骤5：对经过步骤5的网络模型进行测试；

步骤5.1：加载训练好的模型，读取测试数据集；

步骤5.2：计算自动驾驶车辆在测试数据集的避碰率及平均奖励；

步骤5.3：为了证明模型的驾驶安全，对自动驾驶车辆的碰撞时间TTC进行检验，并与当前人类驾驶员进行对比。其中，TTC计算公式如下：



步骤5.4：为了证明模型的驾驶效率，对自动驾驶车辆的车头时距TH进行检验，并与当前人类驾驶员进行对比。其中，TH计算公式如下：



**说 明 书 附 图**



图1



图2



图3



图4



图5



图6

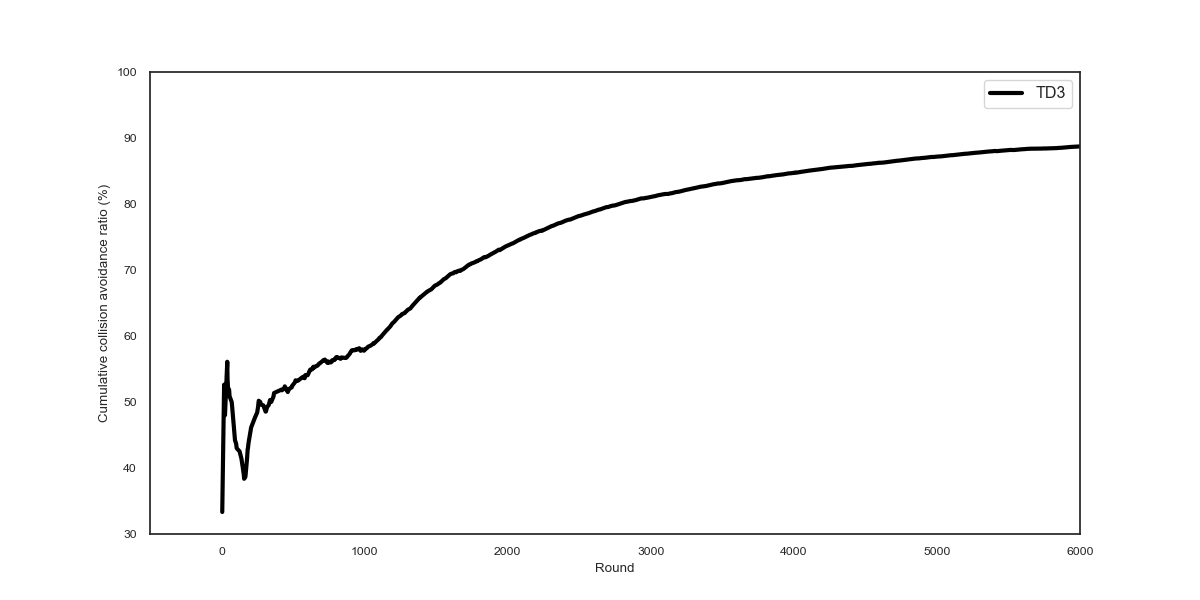


图7