During the handling of complaints (step 3) there can only be valid complaints against a corrupted server. To reconstruct r_i, *SIM* has to reveal the values $f_i(j), f_i'(j)$ for $j \in H_0 \setminus B$. But *SIM* knows all the polynomials $f_i(\cdot), f_i'(\cdot)$ for $i \in H_0 \setminus B$. Therefore, *SIM* has only to broadcast these values, which will always be consistent with the adversary's view.

A more detailed analysis of the distribution can be found in [11]. The computed view, and the induced random variable $SIM(\tilde{A}, G, q, Y)$, has the same probability distribution as $VIEW(\tilde{A}, G, q, Y)$. □

5.3 Unforgeability

In this section, we will show how to reduce the distributed Schnorr signature scheme to the regular Schnorr signature scheme, and visa versa. This implies that the security of the two schemes is identical.

Definition 2 *Let $A_{NormSchnorr}$ be a probabilistic polynomial time adversary who can ask a signer for valid signatures. By $A_{NormSchnorr}(G, q, Y)$ we denote a random variable which specifies the probability of the event that $A_{NormSchnorr}$ queries $(m_1, m_2, ...,)$ to the signer and outputs $(\tilde{m}, \tilde{\sigma}, \tilde{V})$ (on input (G, q, Y)). The probability is taken over all the coin tosses of $A_{NormSchnorr}$ and the signer.*

Definition 3 *Let $A_{DistSchnorr}$ be a probabilistic polynomial time adversary who can corrupt up to $t-1$ players. He also may have $\geq t$ arbitrary signers issue a signature upon his request. By $A_{DistSchnorr}(G, q|Y)$[3] we denote the random variable that has the probability distribution of $A_{DistSchnorr}$ asking for signatures on $(m_1, m_2, ...,)$ (on input (G, q)) and finally computing $(\tilde{m}, \tilde{\sigma}, \tilde{V})$ under the condition that the key generation protocol outputs Y. The probability is taken over all the coin tosses of $A_{DistSchnorr}$ and the signers.*

Theorem 1 *For any adversary $A_{NormSchnorr}$ against $D_{NormSchnorr}$, there exists an adversary $A_{DistSchnorr}$ against $D_{DistSchnorr}$ such that*

$$Pr[A_{DistSchnorr}(G, q|Y) = (m_1, ..., (\tilde{m}, \tilde{\sigma}, \tilde{V}))] = Pr[A_{NormSchnorr}(G, q, Y) = (m_1, ..., (\tilde{m}, \tilde{\sigma}, \tilde{V}))].$$

(Proof) We show how to construct $A_{DistSchnorr}$ given the adversary $A_{NormSchnorr}$. Suppose the key generation protocol of $D_{DistSchnorr}$ generates Y. $A_{DistSchnorr}$ feeds (G, q, Y) and the content of the random tape of $A_{NormSchnorr}$ into $A_{NormSchnorr}$ and starts $A_{NormSchnorr}$. Whenever $A_{NormSchnorr}$ asks for a signature on a message m, $A_{DistSchnorr}$ has some t signers execute the signature issuing protocol for m and returns the signature (σ, V) to $A_{NormSchnorr}$. Thus, $A_{NormSchnorr}$ can perform his chosen message attack. $A_{DistSchnorr}$ outputs $(\tilde{m}, \tilde{\sigma}, \tilde{V})$ if $A_{NormSchnorr}$ outputs $(\tilde{m}, \tilde{\sigma}, \tilde{V})$. □

Theorem 2 *For any adversary $A_{DistSchnorr}$ against $D_{DistSchnorr}$, there exists an adversary $A_{NormSchnorr}$ against $D_{NormSchnorr}$ such that*

$$Pr[A_{NormSchnorr}(G, q, Y) = (m_1, ..., (\tilde{m}, \tilde{\sigma}, \tilde{V}))] = Pr[A_{DistSchnorr}(G, q|Y) = (m_1, ..., (\tilde{m}, \tilde{\sigma}, \tilde{V}))].$$

[3] $A_{DistSchnorr}(G, q|Y)$ is different from $A_{DistSchnorr}(G, q, Y)$. It contains not only the values G, q, Y, but also $A_{DistSchnorr}$'s view from the key generation protocol. For $A_{NormSchnorr}$ this view is empty, while for $A_{DistSchnorr}$ this is not the case (since he can corrupt $t-1$ signers)