

VRF Billing

This guide explains how to estimate VRF costs for both the subscription and direct funding methods.

VRF v2 cost calculator

Choose the method you prefer to use the VRF cost calculator:

SubscriptionDirect fundingEthereumBNB ChainPolygon (Matic)AvalancheFantomArbitrumcaution

Your chosen gas price is higher than the selected gas lane, which means that the request will not be fulfilled until the network gas price goes down to the maximum price for the selected gas lane. The estimated cost shown is for informational purposes only.

Understanding transaction costs

SubscriptionDirect fundingFor Chainlink VRF v2 to fulfill your requests, you must maintain a sufficient amount of LINK in your subscription balance. Gas cost calculation includes the following variables:

- Gas price:The current gas price, which fluctuates depending on network conditions.
- Callback gas:The amount of gas used for the callback request that returns your requested random values.
- Verification gas:The amount of gas used to verify randomness onchain.

The gas price depends on current network conditions. The callback gas depends on your callback function, and the number of random values in your request. The cost of each request is final only after the transaction is complete, but you define the limits you are willing to spend for the request with the following variables:

- Gas lane:The maximum gas price you are willing to pay for a request in wei. Define this limit by specifying the appropriatekeyHashin your request. The limits of each gas lane are important for handling gas price spikes when Chainlink VRF bumps the gas price to fulfill your request quickly.
- Callback gas limit:Specifies the maximum amount of gas you are willing to spend on the callback request. Define this limit by specifying thecallbackGasLimitvalue in your request.For Chainlink VRF v2 to fulfill your requests, you must have a sufficient amount of LINK in your consuming contract. Gas cost calculation includes the following variables:
- Gas price:The current gas price, which fluctuates depending on network conditions.
- Callback gas:The amount of gas used for the callback request that returns your requested random values. The callback gas depends on your callback function and the number of random values in your request. Set thecallback gas limitto specify the maximum amount of gas you are willing to spend on the callback request.
- Verification gas:The amount of gas used to verify randomness onchain.
- Wrapper overhead gas:The amount of gas used by the VRF Wrapper contract. See theRequest and Receive Data section for details about the VRF v2 Wrapper contract design.

Because the consuming contract directly pays the LINK for the request, the cost is calculated during the request and not during the callback when the randomness is fulfilled. Test your callback function to learn how to correctly estimate the callback gas limit.

- If the gas limit is underestimated, the callback fails and the consuming contract is still charged for the work done to generate the requested random values.
- If the gas limit is overestimated, the callback function will be executed but your contract is not refunded for the excess gas amount that you paid.

Make sure that your consuming contracts are funded with enough LINK tokens to cover the transaction costs. If the consuming contract doesn't have enough LINK tokens, your request will revert.

Estimate gas costs

SubscriptionDirect fundingYou need to pre-fund your subscription enough to meet theminimum subscription balance in order to have a buffer against gas volatility.

After the request is complete, the final gas cost is recorded based on how much gas is used for the verification and callback. The actual cost of the request is deducted from your subscription balance.

The total gas cost in wei for your request uses the following formula:

(Gas price * (Verification gas + Callback gas)) = total gas costThe total gas cost is converted to LINK using the ETH/LINK data feed. In the unlikely event that the data feed is unavailable, the VRF coordinator uses thefallbackWeiPerUnitLinkvalue for the conversion instead. ThefallbackWeiPerUnitLinkvalue is defined in thecoordinator contract for your selected network.The final gas cost to fulfill randomness is estimated based on how much gas is expected for the verification and callback. The total gas cost in wei uses the following formula:

(Gas price * (Verification gas + Callback gas limit + Wrapper gas overhead)) = total gas costThe total gas cost is converted to LINK using the ETH/LINK data feed. In the unlikely event that the data feed is unavailable, the VRF Wrapper uses thefallbackWeiPerUnitLinkvalue for the conversion instead. ThefallbackWeiPerUnitLinkvalue is defined in theVRF v2 Wrapper contract for your selected network.

The maximum allowedcallbackGasLimitvalue for your requests is defined in theCoordinator contract supported networks page. Because the VRF v2 Wrapper adds a gas overhead, yourcallbackGasLimitmust not exceedmaxGasLimit - wrapperGasOverhead.

Add LINK premium

SubscriptionDirect fundingThe LINK premium is added to the total gas cost. The premium is defined in thecoordinator contract with thefulfillmentFlatFeeLinkPPMTier1parameter in millionths of LINK.

(total gas cost + LINK premium) = total request costThe total request cost is charged to your subscription balance.A LINK premium is then added to the total gas cost. The premium is divided in two parts:

- Wrapper premium: The premium percentage. You can find the percentage for your network in theSupported networks page.
- Coordinator premium: A flat fee. This premium is defined in thefulfillmentFlatFeeLinkPPMTier1parameter in millionths of LINK. You can find the flat fee of the coordinator for your network in theSupported networks page.

(Coordinator premium + (total gas cost * (1 + Wrapper premium percentage))) = total request cost

Ethereum example

SubscriptionDirect fundingThis is an example calculation of a VRF request on the Ethereum network. The values for other supported networks are available on theSupported Networks page.

Estimate minimum subscription balance

You need to have theminimum subscription balance for your requests to be processed. This provides a buffer in case gas prices go higher when processing the request. The actual cost of the request is usually lower than the minimum subscription balance.

ParameterValueGas lane500 gweiCallback gas limit100000Max verification gas200000LINK premium0.25 LINK1. Calculate the total gas cost, using the maximum possible gas price for the selected gas lane, the estimated maximum verification gas, and the full callback gas limit:

Gas cost calculationTotal gas costGas price x (Verification gas + Callback gas)500 gwei x (200000 + 100000)150000000 gwei (0.15 ETH) 2. Convert the gas cost to LINK using theLINK/ETH feed . For this example, assume the feed returns a conversion value of ≈ 0.004 ETH per 1 LINK.

ETH to LINK cost conversionTotal gas cost (LINK)0.15 ETH / 0.004 ETH/LINK37.5 LINK 3. Add the LINK premium to get the total maximum cost of a request:

Adding LINK premiumMaximum request cost (LINK)Total gas cost (LINK) + LINK premium37.5 LINK + 0.25 LINK37.75 LINK

This example request requires a minimum subscription balance of 37.75 LINK. Check theMax Costin the Subscription Manager to view the minimum subscription balance for all your contracts. When your request is processed, the actual cost of the request is deducted from your subscription balance.

Estimate VRF request cost

This example reflects an estimate of how much a VRF request costs. ChecEtherscan for current gas prices.

ParameterValueActual gas price50 gweiCallback gas used95000Verification gas used115000LINK premium0.25 LINK1. Calculate the total gas cost:

Gas cost calculationTotal gas costGas price x (Verification gas + Callback gas)50 gwei x (115000 + 95000)10500000 gwei (0.0105 ETH) 2. Convert the gas cost to LINK using theLINK/ETH feed . For this example, assume the feed returns a conversion value of ≈ 0.004 ETH per 1 LINK.

ETH to LINK cost conversionTotal gas cost (LINK)0.0105 ETH / 0.004 ETH/LINK2.625 LINK 3. Add the LINK premium to get the total cost of a request:

Adding LINK premiumTotal request cost (LINK)Total gas cost (LINK) + LINK premium2.625 LINK + 0.25 LINK2.875 LINK

```
// VRF variables and calculations// -----// Full script: https://github.com/smartcontractkit/smart-contract-examples/tree/main/vrf-arbitrum-gas-estimation// -----// Estimated upper bound of verification gas for VRF subscription.// To see an estimate with an average amount of verification gas, adjust this to 150000.constmaxVerificationGas=200000// The L1 Calldata size includes:// Arbitrum's static 140 bytes for transaction metadata// VRF V2's static 580 bytes, the size of a fulfillment's calldata abi-encoded in bytes// (from s_fulfillmentTxSizeBytes in VRFV2Wrapper.sol)constVRFCalldataSizeBytes=140+580// For direct funding only. Coordinator gas is verification gasconstwrapperGasOverhead=40000constcoordinatorGasOverhead=90000// VRF use settingsconstcallbackGasLimit=175000// Estimate VRF L1 bufferconstVRF1CostEstimate=L1P.mul(VRFCalldataSizeBytes)constVRF1Buffer=VRF1CostEstimate.div(P)// VRF Subscription gas estimate// L2 gas price (P) * (maxVerificationGas + callbackGasLimit + VRF1Buffer)constVRF1SubscriptionGasSubtotal=BigNumber.from((maxVerificationGas+callbackGasLimit)constVRFSubscriptionGasTotal=VRF1SubscriptionGasSubtotal.add(VRF1Buffer)constVRFVRF Direct funding gas estimate// L2 gas price (P) * (coordinatorGasOverhead + callbackGasLimit + wrapperGasOverhead + VRF1Buffer)constVRF2DirectFundingGasSubtotal=BigNumber.from((coordinatorGasOverhead+wrapperGasOverhead+callbackGasLimit)constVRF2DirectFundingGasTotal=VRF2DirectFundingGasSubtotal.add(VRF1Buffer)
```