

## Industrial Training on VLSI Design

This certificate is awarded to **Anurima Mallick** of Heritage Institute of Technology, Kolkata, D/O Rupam Mallick for successfully completing the training course entitled RTL Design Engineer of 200 Hours duration through online conducted from **16<sup>th</sup> Sept 2021** to **16<sup>th</sup> October 2021**.

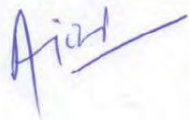


Dr. Ajit Kumar Panda

Center In-Charge

## Industrial Training on VLSI Design

This certificate is awarded to **Arin Saha** of Heritage Institute of Technology, Kolkata, S/O Bablu Saha for successfully completing the training course entitled RTL Design Engineer of 200 Hours duration through online conducted from **16<sup>th</sup> Sept 2021** to **16<sup>th</sup> October 2021**.



Dr. Ajit Kumar Panda

Center In-Charge



souvik basu <souvik.basu@heritageit.edu>

---

## Peer-to-Peer Networking and Applications - Submission Notification to co-author - [EMID:883f1a910f774a4e]

---

Peer-to-Peer Networking and Applications <em@editorialmanager.com>

Fri, Nov 12, 2021 at 9:16 PM

Reply-To: Peer-to-Peer Networking and Applications <genesis.obero@springer.com>

To: Souvik Basu <souvik.basu@heritageit.edu>

Re: "ReliefChain: A Blockchain Leveraged Post Disaster Relief Allocation System over Smartphone-Based DTN"  
Full author list: Nabanita Das; Souvik Basu; Sipra Das Bit

Dear Mr. Souvik Basu,

We have received the submission entitled: "ReliefChain: A Blockchain Leveraged Post Disaster Relief Allocation System over Smartphone-Based DTN" for possible publication in Peer-to-Peer Networking and Applications, and you are listed as one of the co-authors.

The manuscript has been submitted to the journal by Dr. Mrs Nabanita Das who will be able to track the status of the paper through his/her login.

If you have any objections, please contact the editorial office as soon as possible. If we do not hear back from you, we will assume you agree with your co-authorship.

Thank you very much.

With kind regards,

Springer Journals Editorial Office  
Peer-to-Peer Networking and Applications

This letter contains confidential information, is for your own use, and should not be forwarded to third parties.

Recipients of this email are registered users within the Editorial Manager database for this journal. We will keep your information on file to use in the process of submitting, evaluating and publishing a manuscript. For more information on how we use your personal details please see our privacy policy at <https://www.springernature.com/production-privacy-policy>. If you no longer wish to receive messages from this journal or you have questions regarding database management, please contact the Publication Office at the link below.

---

In compliance with data protection regulations, you may request that we remove your personal registration details at any time. (Use the following URL: <https://www.editorialmanager.com/ppna/login.asp?a=r>). Please contact the publication office if you have any questions.



# ReliefChain: A blockchain leveraged post disaster relief allocation system over smartphone-based DTN

Nabanita Das<sup>1</sup> · Souvik Basu<sup>2</sup> · Sipra Das Bit<sup>1</sup>

Received: 18 November 2021 / Accepted: 4 August 2022 / Published online: 3 September 2022  
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

## Abstract

One of the major concerns in any emergency relief operation is appropriate allocation of scarce emergency relief materials to the affected community. Due to several reasons ranging from lack of mechanism to accurately assess demand and utility of relief materials to malicious participation of some of the stakeholders, such allocation may become ad-hoc. Thus, it becomes imperative to have an unchallengeable and globally accessible record of relief requirement vis-à-vis allocation for efficient relief management. Emergency response organizations (e.g. UNICEF) have recommended the adoption of blockchain technology to create such immutable records. However, the usage of blockchain is restricted by the availability of end-to-end internet connection which may not be available in a post-disaster scenario. This paper proposes ReliefChain, a blockchain leveraged post disaster relief allocation system over delay tolerant network that works in such environments. We validate relief requirements to mitigate resource diversion, forecasting the exact demand and enumerating precise utilities of relief items. We design smart contracts for creating new transactions to upload relief requirements and allocations in the blockchain network. The proposed system executes these smart contracts to create an immutable and globally accessible record of relief requirement and allocation. Effectiveness of the proposed system is evaluated through extensive simulation in Ethereum platform. Results substantiate the efficiency of the system over a system using baseline methodologies, in terms of design parameters like shelter specific deficit and average resource deficit while not compromising the blockchain performance in terms of processing time and gas consumption even in presence of malicious forwarders.

**Keywords** Blockchain · Delay-tolerant network (DTN) · Ethereum · Post Disaster Relief Allocation · Smart Contracts

## 1 Introduction

Over the past decade, countries across the world have witnessed a significant number of disasters such as earthquake (Nepal 2015), super cyclone (India 2019) and the more recent COVID-19 pandemic. In the post-disaster scenario, one of the immediate important services is to coordinate the distribution

of emergency relief materials provided by Government, NGO etc. among the victims. However, the effectiveness of this relief distribution depends on the competencies and experience of the logisticians often leading to ad-hoc allocation of scarce emergency relief materials. Such ad-hoc allocation sometimes even questions on allocation transparency.

Further, considering the impact of natural disaster on communication infrastructure, the researchers have suggested the exploitation of delay tolerant network (DTN) for creating an emergency post disaster communication network using the several communication facilities of smartphones like—Wi-Fi Direct or Bluetooth etc. These smartphones are carried by relief volunteers and supply vehicle-drivers, working in the DTN mode [1–6]. However, DTNs have intermittent connectivity, lack of end-to-end path and frequent breakups, as a consequence of which we get poor network performance. One possible solution to improve network performance is to set up fixed relay nodes (e.g., smartphone, laptop having internet connectivity), called DropBoxes, that help in transferring messages.

---

✉ Nabanita Das  
naba.iem@gmail.com

Souvik Basu  
souvik.basu@heritageit.edu

Sipra Das Bit  
sdasbit@yahoo.co.in

<sup>1</sup> Department of Computer Science and Technology, Indian Institute of Engineering Science and Technology, Shibpur, West Bengal, India

<sup>2</sup> Department of Computer Applications, Heritage Institute of Technology, Kolkata, India

To mitigate the problem of ad-hoc allocation, a scientific forecasting model to forecast the demands for post disaster emergency relief materials and enumerating the utilities of those materials is imperative. There are a few works that attempt to forecast the precise demand of relief materials in each emergency relief shelter [5] and enumerate the exact utility of such materials [6] using mathematical modeling.

On the other hand, to address the issues of allocation transparency, resource diversion, etc., it becomes imperative to have a document for providing an unchallengeable and globally accessible record of relief requirement vis-à-vis allocation for efficient relief management. Several emergency response organizations and researchers have proposed [7–14] the adoption of blockchain technology [15, 16] to create such a shared and unchallengeable document of records. However, the usage of blockchain is restricted by the availability of end-to-end internet connection which, as mentioned earlier, may not be available in a post-disaster scenario.

This paper proposes ReliefChain, a blockchain leveraged post disaster relief allocation system over delay tolerant network that works in such environments. The proposed system gathers relief requirements of different shelters at a designated DropBox which in turn validates these relief requirements to mitigate resource diversion and forecasts the exact demand and enumerates precise utilities of these relief items. As we know, smart contracts are a type of Ethereum account that can be used to create and send transactions over the blockchain network, the DropBox executes smart contracts for creating new transactions to log the demands, utilities and allocation details into the blockchain. Logging these details into a blockchain makes them tamper proof, as any change to the record would require multiple validations. The proposed system creates an immutable and globally accessible record of relief requirement vis-à-vis allocation.

The rest of this paper is organized as follows. Section 2 summarizes related works on blockchain based systems related to emergency response. In Sect. 3, we provide preliminaries on blockchain system and Sect. 4 describes the network architecture of the proposed system. We propose the ReliefChain system in Sect. 5. A detailed performance analysis, illustrating the effectiveness of the proposed system, is presented in Sect. 6. We conclude the paper with a direction towards future research in Sect. 7.

## 2 Related work

In this section, we review a few important blockchain based systems related to emergency response.

The study by Panesir in [7] focuses on how we can use blockchain technology to offer relief materials and have

better disaster management. The main focus of this work is to develop a strategy for creating transactions that are safe, and rapid, which helps in reducing corruption, mal-handling of resources and information.

Siemon et al. in [8] propose a strategy for the adoption of blockchain technology in the field of emergency response by applying a Design Science Research (DSR) approach. On one hand, authors show how blockchain technology seems to meet all the desired requirements for trust and information security. On the other hand they illustrate how blockchain technology can be applied for fast and effective disaster response.

Ojetunde et al. in [9] propose a blockchain based mobile payment system for buying essentials over MANETs in a disaster scenario. Here, each customer selects endorsers for validating their transactions, where the digital signature of an endorser is used for all transactions as proof of the validation. The user privacy is protected by using the user nickname (e.g. a Temporary ID) which is used in every transaction. The work also proposes a mechanism which detects double spending before a transaction is completed.

L'Hermitte et al. in [10] studies the sharing economy concept, leveraged by the blockchain technology, can be used for distributing resources in a post disaster scenario. This work develops a framework for commercial organisations to distribute resources in emergency operations.

Badarudin et al. in [11], explore how blockchain technologies accelerate speed and availability of post disaster relief materials and services. The federated blockchain model, used in this work, helps in executing monetary dealings safely and transparently by implementing a donation mechanism.

Demir et al. in [12], propose a solution based on blockchain technologies for restoration of emergency services in a post disaster scenario. This work, using blockchain technology, captures the restoration activities where all actors are working and all stakeholders are relying on it. Authors show elaborate details on how blockchain can be used to facilitate transparency in disaster relief operations eventually bringing relief to the burden caused by a disaster.

Chakrabarti et al. in [13] suggest an incentive scheme using blockchain to be used in a DTN enabled smartphone based post disaster communication network. This scheme uses Bitcoins to give incentivizes to nodes that cooperate. The scheme uses special nodes called observer-nodes for accessing the blockchain network on behalf of the forwarder-nodes who do not have Internet connectivity.

Basu et al. in [14], propose a mechanism for integration of blockchain technology with DTNs towards improving disaster management services. With the help of this integration, the post disaster resource needs can be collected from remote shelters, validated, converted to blocks and finally uploaded to a blockchain network. Authors also develop an Application

Program Interface (API) and an Application Binary Interface (ABI) to deploy the blocks containing shelter needs, in the form of smart contracts on the Ethereum network.

Referring the above works, almost all the blockchain based emergency response systems [7–12] require a fully functional network infrastructure to operate at a desired level, which cannot be guaranteed in a post disaster scenario. To the best of our knowledge, only blockchain based systems [13, 14] work in intermittently connected network environment. Finally, none of these above works addresses the issue of transparency in relief management.

The above observations motivate us to propose ReliefChain, the first endeavor, to the best of our knowledge, towards blockchain based emergency relief allocation that works even with impaired network connectivity while ensuring transparency in relief management. The specific contribution of this work is developing a blockchain leveraged post disaster relief allocation system working over a DropBox based DTN by:

- validating relief requirements to—mitigate resource diversion, forecast exact demand for relief items and enumerate precise utilities of relief items
- writing smart contracts for deploying new blocks, consisting demand, utility and allocation for relief items, to the blockchain
- evaluating the proposed system through simulations, in terms of processing time, gas consumption and deficit minimization time using customized Solidity code on the Ethereum platform.

### 3 Blockchain preliminaries

In this section, we present an overview of the blockchain preliminaries.

#### 3.1 Blockchain and its blocks

Blockchain is a technology that helps in recording transactions made with any crypto currency that are maintained across several computers that are linked in a peer-to-peer network. It is a distributed database existing on multiple computers at the same time. A blockchain is constantly growing as new sets of records, or blocks, are added to the chain. Thus, blockchain being a link list kind of data structure maintains the full history of transactions using blocks [17, 18].

Thus, in a blockchain, a particular block is chained to its previous block through the use of a cryptographic hash pointer. A change in a transaction of a block makes it compulsory to recalculate all subsequent blocks, which needs huge computation power. This makes the blockchain immutable, a dominant property of all crypto currencies.

#### 3.2 Transaction

A transaction in a block is a small unit of task. Every transaction has a unique identifier (TX) and has a set of inputs and outputs. Inputs to a transaction include previous transaction identifier (TX') and signature of the user. Outputs from a transaction includes the amount transferred to a recipient and the recipient's public key. Output of a transaction is used as input of the next transaction for verification. Transaction verification is done by peer-to-peer network nodes and only the validated transactions are stored in a block.

#### 3.3 Mining

Mining in blockchain is essentially adding transactions to the existing blockchain ledger of transactions. Mining creates a hash of a block of transactions that is impossible to be forged. Some computers storing the entire blockchain are called miners. When a transaction is made over a blockchain the transaction information is thus put on a block and gets recorded. All the miners solve a cryptographic puzzle through a guess-and-check method in order to find the proper cryptographic hash for the block. Once the miners validate a block, it is added to the chain of blocks and that must be validated by other nodes through a process called consensus.

#### 3.4 Ethereum and smart contracts

Ethereum [18] is computing platform used for writing smart contracts. Ether is the crypto currency used by the Ethereum platform to reward the miners. "Gas" is the internal pricing mechanism, is used to handle spam and allocate resources on the network.

A smart contract is a piece of code that is running on Ethereum to execute credible transactions without any third parties. It is a self-executing contract between network participants, without the need of traditional legal contracts.

### 4 Network architecture

In a post disaster scenario, victims generally take refuge at the nearby safe areas. A control station is setup at nearby resource-rich areas [1–6]. The scenario of volunteers carrying smartphones and supply vehicles communicate may be considered by the system as a DTN based communication network after disaster. The network architecture of our proposed system along with the data flow is illustrated in Fig. 1. The network comprises of four types of nodes, as described below.

- *Shelter-node* – A shelter houses the shelter-node (e.g., a laptop equipped with Wi-Fi Direct). Emergency manager(s) present in a shelter uploads shelter requirements to the shelter-node which, in turn, spreads these information towards the available forwarder-nodes.
- *Forwarder* – Volunteers and supply vehicles who carry smartphones that have facilities like Wi-Fi Direct, are denoted as forwarder-nodes who travel in the disaster affected area, gather shelter requirements from the shelter-nodes, share these messages using the DTN routing protocol PROPHET [19] and finally send them to the control-node.
- *DropBox* – Stationary relay nodes (e.g., smartphone, laptop having LTE/5G Internet connectivity, possessing sufficient storage and computation capacity) that receive and store messages from nodes that pass by it and forward them to the intended receivers. DropBoxes coupled with its local server are deployed at high priority locations, based on the uDBdep scheme [1] to maximize data transfer. DropBoxes are connected to the control station through the Internet. Internet connectivity is essential at the DropBoxes for transmitting aggregated information regarding relief requirements at shelters to the control station. DropBoxes have accounts in the blockchain network to transmit and receive blockchain transactions over Internet.
- *Control-node* – The control station houses the control-node (e.g., a workstation equipped with LTE/5G Internet connection and Wi-Fi Direct) where relief requirements of the distant shelters are gathered and required resources are optimally allocated. It allocates relief materials to the shelters and reverts back the relief allocation figures to the DropBoxes over Internet.

We consider Internet connectivity at both the nodes *DropBoxes* and *Control-node*. If DropBoxes don't have such connectivity, they would not be able to transmit the aggregated information to the control-node, resulting in flooding of the control-node with the huge amount of raw information from the forwarders.

## 5 Proposed ReliefChain system

In this section, we first review the methodologies, in [5] and [6], used as baselines in this work and then present the proposed ReliefChain system.

### 5.1 Baseline methodologies

The ReliefChain system uses three baseline methodologies; one that derives a model for forecasting the demand for different relief materials in a shelter [5], another that enumerates the utility of emergency relief materials in a shelter [6], and finally one that formulates an optimal resource allocation model [6]. Here, we provide a succinct overview of these baselines works.

#### 5.1.1 PCR model for forecasting demand

The work proposed in [5] formulates a PCR model that forecasts demands of post disaster emergency resources, based on influencing situational parameters (ISP) that affect the demand for relief materials. These parameters are different for different categories of relief materials. The demand forecasting model, based on uncorrelated ISPs takes the form as follows:

$$D_{t,rs} = \beta_1 ISP'_{t,1} + \beta_2 ISP'_{t,2} + \dots + \beta_m ISP'_{t,m} + \epsilon_{t,i}, i = 1, 2, \dots, n \tag{1}$$

where demand  $D_{t,rs}$  is the demand for the  $r^{th}$  emergency resource in the  $s^{th}$  shelter and  $ISP'_{t,1}, ISP'_{t,2}, \dots, ISP'_{t,m}$  are the values of the uncorrelated ISPs in that shelter at time point  $t_i, i = 1, 2, \dots, n$ . Here, the  $\beta$  values and  $\epsilon$  are regression coefficients and error respectively that can be estimated from a given dataset.

#### 5.1.2 Utility function for enumerating utility

The instantaneous exigency of an emergency resource is called its utility of a resource. Utility and demand are not same. Demand for a resource is the amount required, while utility of a resource is its importance. The work proposed in [6] derives a utility function that dynamically and quantitatively enumerates the correct utility of each emergency resource at different shelters.



Fig. 1 Network architecture

We assume utility of a resource as a function of consumption of a resource in a shelter (i.e., the amount of a resource allocated to a shelter), i.e.,  $u_{rs}$  (utility of  $r^{th}$  resource at  $s^{th}$  shelter) is a function of  $x_{rs}$  (total allocated quantity of  $r^{th}$  resource at  $s^{th}$  shelter). Nevertheless, post disaster emergency resources are allocated to shelters at periodic but non-uniform time points. This implies that allocation of a resource in a shelter varies with time, i.e.,  $x_{rs}$  is a function of time  $t$ . Therefore, we can say that  $u_{rs}$  depends on  $x_{rs}$  which again depends on  $t$ , because  $x_{rs}$  can vary only where  $t$  varies. Hence

$$u = f(x) = f(g(t))$$

Therefore, our next task would be examining the nature of the relation between  $x_{rs}$  and  $t$ . Since resources arrive at shelters at non-uniform time points,  $t_1, \dots, t_n$ , it is justifiable to use Lagrange's Interpolation formula to find the dependence of  $x_{rs}$  on  $t$ . Hence,  $x$  can be represented as:

$$\begin{aligned} x_{rs} &= \frac{(t-t_2)(t-t_3)\dots(t-t_n)}{(t_1-t_2)(t_1-t_3)\dots(t_1-t_n)} \\ x_{1rs} &+ \frac{(t-t_1)(t-t_3)\dots(t-t_n)}{(t_2-t_1)(t_2-t_3)\dots(t_2-t_n)} x_{2rs} + \dots \\ \dots &+ \frac{(t-t_1)(t-t_3)\dots(t-t_{n-1})}{(t_n-t_1)(t_n-t_2)\dots(t_n-t_{n-1})} x_{nrs} \end{aligned}$$

$$\text{i.e., } x_{rs} = \sum_{i=1}^n \prod_{\substack{j=1 \\ j \neq i}}^n \frac{t-t_j}{t_i-t_j} x_{irs}$$

Therefore, the function takes the form as follows:

$$\begin{aligned} u_{rs} &= a \left( \sum_{i=1}^n \prod_{\substack{j=1 \\ j \neq i}}^n \frac{t-t_j}{t_i-t_j} x_{irs} \right)^3 \\ &+ b \left( \sum_{i=1}^n \prod_{\substack{j=1 \\ j \neq i}}^n \frac{t-t_j}{t_i-t_j} x_{irs} \right)^2 \\ &+ c \left( \sum_{i=1}^n \prod_{\substack{j=1 \\ j \neq i}}^n \frac{t-t_j}{t_i-t_j} x_{irs} \right) + d \end{aligned} \quad (2)$$

where  $u_{rs}$  and  $x_{rs}$  are the utility and total allocated quantity of the  $r^{th}$  resource at  $s^{th}$  shelter,  $t_i$ 's,  $i = 1, 2, \dots, n$ , are past time points when resources were allocated,  $t$  is the current time point for which utility is to be calculated and (a, b, c, d) are constants to be estimated from a given dataset.

### 5.1.3 Optimal resource allocation model

The formulated optimal resource allocation model, proposed in [6], has two goals –

(a) *Minimizing overall emergency resource deficit*

Deficit of the  $r^{th}$  resource at the  $s^{th}$  shelter can be primarily defined as the ratio of unfulfilled demand and actual demand in that shelter, i.e.,

$$\left( \frac{d_{rs} - x_{rs}}{d_{rs}} \right)$$

However, as demand alone cannot represent the actual exigency of a resource in a shelter; it is the utility that speaks about the extent of the need for that resource. Therefore, to make our resource allocation strategy utility aware, we redefine resource deficit as a weighted ratio of unfulfilled demand and the actual demand, with utility as the weight, i.e.,

$$u_{rs} \left( \frac{d_{rs} - x_{rs}}{d_{rs}} \right)$$

where a higher value of  $u_{rs}$  magnifies the deficit and vice versa. We define the overall resource deficit as below:

$$\sum_{s=1}^S \sum_{r=1}^R u_{rs} \left( \frac{d_{rs} - x_{rs}}{d_{rs}} \right)$$

The above represents the aggregate of resource deficits of all resources and all shelters. The quantity of each resource to be allocated to each shelter in order to minimize the overall resource deficit subject to some constraints. The inputs to the problem are  $S, R, a_r, d_{rs}, u_{rs}$  and the output is a resource allocation matrix. The allocation model is designed as follows:

$$\begin{aligned} \text{Minimize : } & \sum_{s=1}^S \sum_{r=1}^R u_{rs} \left( \frac{d_{rs} - x_{rs}}{d_{rs}} \right) \\ \text{Subject to : } & \sum_{s=1}^S x_{rs} \leq a_r \quad \text{for all } r \text{ and } s \\ & x_{rs} \leq d_{rs} \quad \text{for all } r \text{ and } s \\ & x_{rs} \geq 0 \quad \text{for all } r \text{ and } s \end{aligned} \quad (3)$$

Solving (3) with the constraints yields a resource allocation matrix containing an optimal number ( $x_{rs}$ ) of the  $r^{th}$  resource allocated to the  $s^{th}$  shelter for minimizing the overall resource deficit.

(b) *Minimizing total resource deployment time*

We note that the time required for deploying one unit of  $r^{th}$  resource to  $s^{th}$  shelter from  $w^{th}$  warehouse is  $t_{rsw}$ . Therefore,  $y_{rsw}$  units of  $r^{th}$  resource can be deployed to  $s^{th}$  shelter from  $w^{th}$  warehouse in  $t_{rsw} * y_{rsw}$  time unit. However, in the above definition of deployment time the assumption is to ship one unit of resource at a time, which is unrealistic.



Conventionally, in one consignment several units of relief resources are dispatched. Here, the size of consignment ( $c_r$ ) determines how many number of resource ( $r$ ) can be dispatched in one consignment. Thus, we reformulate the deployment time as shown below:

$$t_{rsw} * \left\lceil \frac{y_{rsw}}{c_r} \right\rceil$$

where  $\left\lceil \frac{y_{rsw}}{c_r} \right\rceil$  is the number (nearest integer  $\geq \frac{y_{rsw}}{c_r}$ ) of consignments required for shipping  $y_{rsw}$  units of the resource.

We can justify this, as the required time for dispatching one consignment holding several units of a resource is same as the time for dispatching one unit of that resource. In the above formulation, the assumption is to ship one consignment at a time. We exclude consignment preparation time and other logistic overheads and therefore, the number of consignments to be shipped at an instance depends only on availability of shipping vehicles at that instance. Total time for resource deployment is defined as

$$\sum_{r=1}^R \sum_{s=1}^S \sum_{w=1}^W t_{rsw} * \left\lceil \frac{y_{rsw}}{c_r} \right\rceil$$

This is the total deployment time of the resources in all the shelters and warehouses. We find the quantity of emergency resources to be allotted to a shelter from different warehouses in order minimize the total deployment time, subject to a few constraints. The inputs to this minimization problem are  $S, R, W, t_{rsw}, x_{rs}$  and  $a_{rw}$  and the output is a resource deployment matrix. The minimization problem is designed as:

$$\begin{aligned} \text{Minimize : } & \sum_{r=1}^R \sum_{s=1}^S \sum_{w=1}^W t_{rsw} * \left\lceil \frac{y_{rsw}}{c_r} \right\rceil \\ \text{Subject to : } & \sum_{w=1}^W y_{rsw} = x_{rs} \quad \text{for all } r \text{ and } s \\ & \sum_{s=1}^S y_{rsw} \leq a_{rw} \quad \text{for all } r \text{ and } s \end{aligned} \tag{4}$$

Solving (4) with the constraints yields a resource deployment matrix containing the optimal number ( $y_{rsw}$ ) of the  $r^{th}$  resource allocated to the  $s^{th}$  shelter from the  $w^{th}$  warehouse for minimizing the total deployment time.

The symbols used here carry the following significance:

---

$S$	:	number of shelters
$R$	:	number of resource types
$a_{rw}$	:	available number of units of resource $r$ at warehouse $w$
$a_r$	:	total available units of $r$
$d_{rs}$	:	demand for $r$ at shelter $s$
$u_{rs}$	:	utility of $r$ at $s$
$t_{rsw}$	:	time required to deploy each unit of $r$ to $s$ from $w$
$c_r$	:	capacity of each consignment of $r$

---

$x_{rs}$	:	number of units of $r$ allocated to $s$
$y_{rsw}$	:	number of units of $r$ allocated to $s$ from $w$

---

$r=1, 2, \dots, R; s=1, 2, \dots, S; \text{ and } w=1, 2, \dots, W$

## 5.2 Our proposal

The proposed ReliefChain system handles post disaster emergency relief material distribution over DTN in an unchallengeable manner. The ISP values generated at remote shelters sent by shelter-nodes to the forwarders, who in turn, transmit them to the nearest DropBox. The DropBoxes validates ISP values, forecast demand and enumerate utility of relief materials. The demands and utilities are then transmitted to the control-node over Internet, where these are used to allocate relief materials to different shelters, using an optimal resource allocation model. The resource allocation matrix is transmitted to the corresponding DropBoxes. The DropBoxes in turn, executes smart contracts, a type of Ethereum account used for creating and sending transactions over the blockchain network, to create new transactions consisting demand, utility and allocated amount of relief materials and transmit them to the blockchain miners. The miners validate the transactions and add them as a new block in the blockchain. The entire ReliefChain system is schematically shown in Fig. 2. The following sub-sections elaborate the tasks performed at the designated nodes of the system.

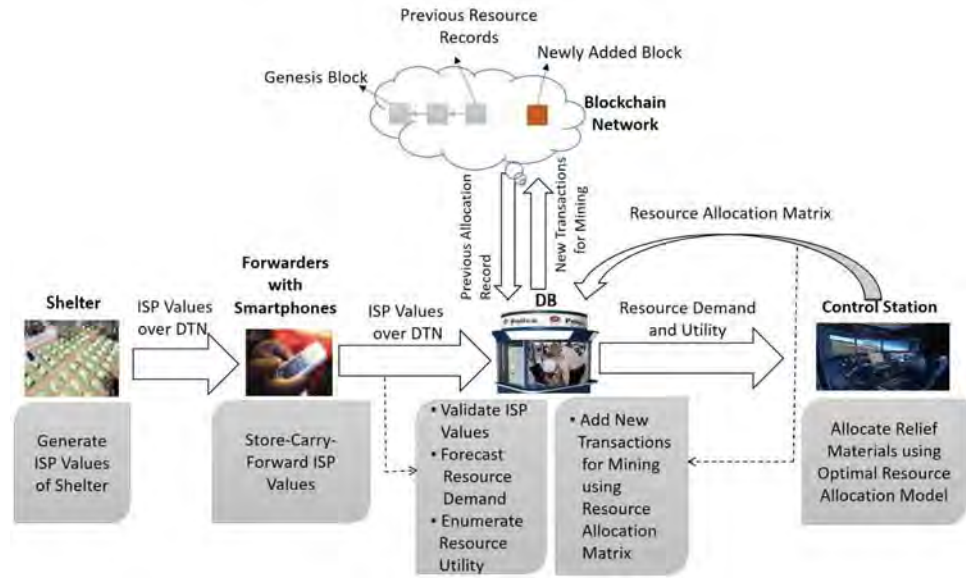
### 5.2.1 Tasks of shelter-nodes and forwarders

The emergency managers, who coordinate relief activities in a shelter, collect the values of the predetermined ISPs pertaining to each relief material, like food, medicine, etc. and upload them to the shelter-node therein. The shelter-node stores these ISP values (Table 1) which are then broadcast to the forwarders, i.e., volunteers and supply vehicles carrying smartphones, either directly using Wi-Fi Direct or in multiple hops indirectly using DTN routing, as shown in the shelter block of Fig. 3. The forwarders store, carry and forward the ISP table to the nearest DropBox, either directly using Wi-Fi Direct or in multiple hops using DTN routing. As shown in the forwarder block of Fig. 3 forwarders transfer the ISP values to the nearest DropBox for further processing.

### 5.2.2 Tasks at dropbox and control-node

At the end of every operational period, which is typically 12 hours in a post disaster scenario, the DropBoxes collate all ISP tables received from all forwarders and starts further

**Fig. 2** Schematic diagram of the proposed system



processing. DropBoxes execute smart contracts, written in Solidity, in the Ethereum platform to (i) validate all received ISP values, (ii) forecast demand for different emergency relief materials for shelters under its purview using the validated ISP values, and (iii) enumerate utility of.

such material for the shelters using previous resource allocation records stored in the blockchain.

This activity of a DropBox is depicted in the “Computation” part of the DropBox block of the flowchart as shown in Fig. 3. All the said activities performed at a DropBox are written as smart contracts to be executed at different phases.

**Validating ISP values** The ISP values at different shelters that are transmitted over a fragile DTN may get corrupted due to network related issues. Moreover, forwarders and/or shelters (i.e., the emergency managers therein) may maliciously broadcast fictitious ISP values with the intentions of resource diversion and embezzlement. If such inaccurate information, received at the DropBox, is used for relief material allocation, the allocation would become irrational. Therefore, it becomes imperative to execute smart contracts to validate the ISP values before these are used to forecast demand for relief materials at the shelters. The

**Table 1** ISP table

Relief Material	ISP Values			
	ISP <sub>1</sub>	ISP <sub>2</sub>	...	ISP <sub>m</sub>
R <sub>1</sub>	ISP <sub>11</sub>	ISP <sub>12</sub>	...	ISP <sub>1m</sub>
R <sub>2</sub>	ISP <sub>21</sub>	ISP <sub>22</sub>	...	ISP <sub>2m</sub>
...	...	...	ISP <sub>ij</sub>	...
R <sub>n</sub>	ISP <sub>n1</sub>	ISP <sub>n2</sub>	...	ISP <sub>nm</sub>

ReliefChain system proposes local and global validation techniques, to validate the received ISP values at the forwarder level and the shelter level respectively. For local and global validation, the DropBox uses statistical mode [20], as a measure of central tendency, to find out the most representative values of the ISPs. Other measures of central tendency like statistical mean and median has their own drawbacks. Mean is affected by very minute or very large values. Arranging the data in ascending or descending order of magnitude for finding median is time-consuming in the case of a large number of observations.

*Local validation:*

For local validation, the DropBox computes the statistical mode [20] of the received ISP values,  $lm_{ISP}$ , for each relief material from all forwarders. The modes are stored in a local mode table, as shown in Table 2.

Then, the DropBox calculates the deviation,  $dev^k_{ISP_{ij}}$ , for the value of the  $j^{th}$  ISP pertaining to  $i^{th}$  relief material provided by the  $k^{th}$  forwarder, using the formula:

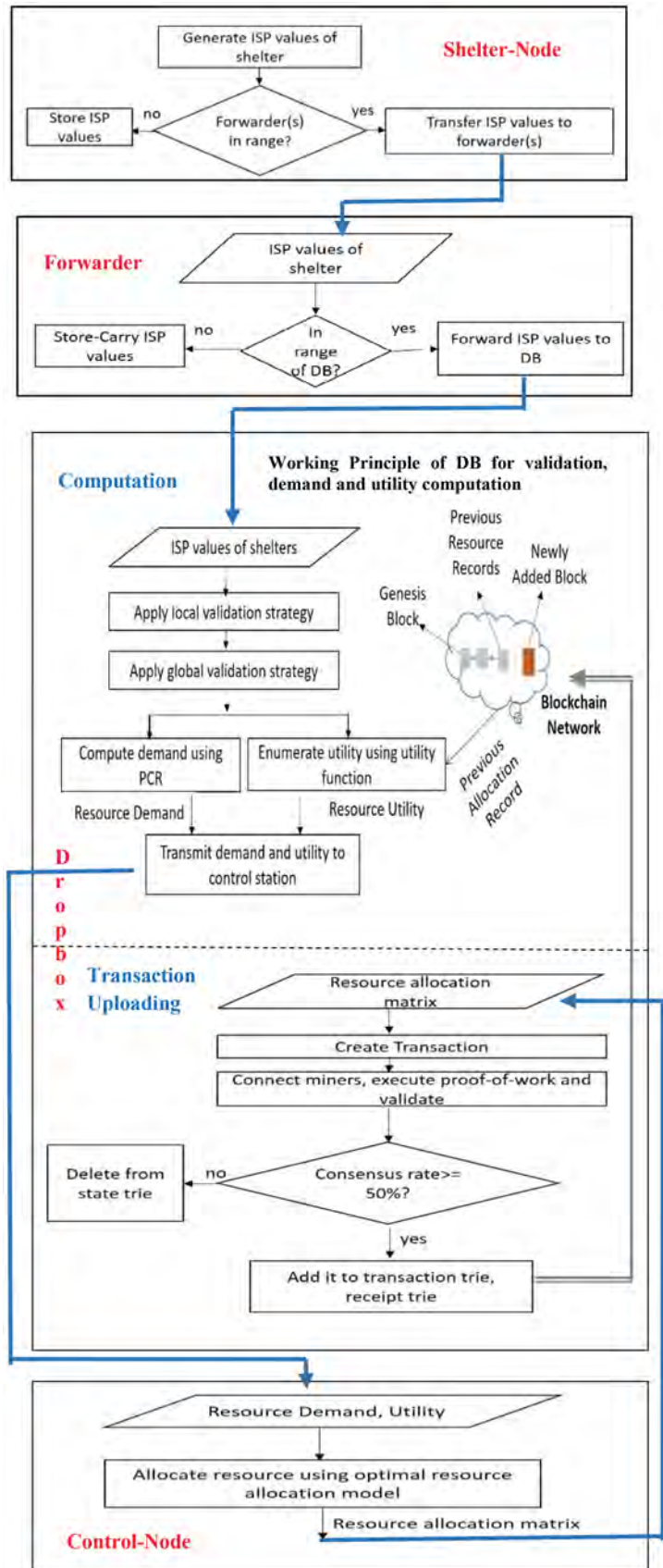
$$dev^k_{ISP_{ij}} = |ISP^k_{ij} - lm_{ISP_{ij}}|$$

where  $ISP^k_{ij}$  is the  $j^{th}$  ISP value pertaining to  $i^{th}$  relief material provided by the  $k^{th}$  forwarder and  $lm_{ISP_{ij}}$  is the mode of all values of the  $j^{th}$  ISP pertaining to  $i^{th}$  relief material. If  $dev^k_{ISP_{ij}} \geq \text{threshold}$  (predefined), the DropBox rejects the ISP value of the corresponding forwarder.

If the average deviation, i.e.  $\frac{1}{m \times n} \sum_{i=1}^n \sum_{j=1}^m dev^k_{ISP_{ij}}$ , for the  $k^{th}$  forwarder exceeds the threshold over a period of time (period of tolerance), the forwarder is ostracized and all future ISP values provided by the forwarder are rejected.

*Global validation:*

**Fig. 3** Flowchart for the Relief-Chain system



**Table 2** Local mode table

Relief Material	ISP Values			
	ISP <sub>1</sub>	ISP <sub>2</sub>	...	ISP <sub>m</sub>
R <sub>1</sub>	lm <sub>ISP<sub>11</sub></sub>	lm <sub>ISP<sub>12</sub></sub>	...	lm <sub>ISP<sub>1m</sub></sub>
R <sub>2</sub>	lm <sub>ISP<sub>21</sub></sub>	lm <sub>ISP<sub>22</sub></sub>	...	lm <sub>ISP<sub>2m</sub></sub>
...	...	...	lm <sub>ISP<sub>j</sub></sub>	...
R <sub>n</sub>	lm <sub>ISP<sub>n1</sub></sub>	lm <sub>ISP<sub>n2</sub></sub>	...	lm <sub>ISP<sub>nm</sub></sub>

For global validation, the DropBox computes the statistical mode [20] of the modes,  $gm_{ISP}$ , from all local modes for each relief material in a shelter for the past few days and stores them in a global mode table, as shown in Table 3.

Then, the DropBox calculates the deviation,  $dev^{current}_{ISP_{ij}}$ , for the current value of the  $j^{th}$  ISP pertaining to  $i^{th}$  relief material provided by the  $k^{th}$  forwarder, using the formula:

$$dev^{current}_{ISP_{ij}} = \left| lm_{ISP_{ij}}^{current} - gm_{ISP_{ij}} \right|$$

where  $lm_{ISP_{ij}}^{current}$  is the mode of  $j^{th}$  ISP value pertaining to  $i^{th}$  relief material provided by all forwarders on the current day and  $gm_{ISP_{ij}}$  is the mode of all modes of the  $j^{th}$  ISP pertaining to  $i^{th}$  relief material for the past few days.  $dev^{current}_{ISP_{ij}} \geq$  threshold, the DropBox rejects the current ISP values of the shelter and uses  $gm_{ISP_{ij}}$  as the current value of the  $j^{th}$  ISP pertaining to  $i^{th}$  relief material of that shelter.

If the average deviation, i.e.  $\frac{1}{m \times n} \sum_{i=1}^n \sum_{j=1}^m dev^{current}_{ISP_{ij}}$ , for a shelter exceeds the threshold over a period of time (period of tolerance), the DropBox recommends to depute a fresh set of emergency managers in that shelter.

**Forecasting demand** In this section, we estimate the regression coefficients  $\beta$  and error  $\epsilon$  in the baseline methodology discussed in Sect. 5.1.1 (Eq. 1), a DropBox uses pre-loaded historical dataset on our application domain [21]. The dataset used consists of observed demands for different relief materials and values of ISPs pertaining to different

relief materials at different shelters over a period of time, say the last 30 days.

The demands are the ones calculated by the DropBox itself and the ISP values are the modes of the validated ISP values provided by the forwarders. After selecting the uncorrelated ISPs we formulate a forecast model for predicting demand  $D_{rs}$  for the  $r^{th}$  emergency resource in the  $s^{th}$  shelter. We use multiple linear regression on the selected principal components to derive a predictive model to an available data set on the  $D_{rs}$  and  $(ISP_1, ISP_2, \dots, ISP_m)$  for  $n$  time points. A DropBox periodically executes least square method, for estimating the values of  $\beta$  and  $\epsilon$  and derive a PCR model that can typically take the form as shown below:

$$\hat{D}_{R_1s} = -1152.80 + 1.087ISP_1 + 46.052ISP_2 - 26.47ISP_3 + 3.30ISP_4 - 0.78ISP_5 \tag{5}$$

where  $\hat{D}_{R_1s}$  is the forecast demand for resource  $R_1$  in shelter  $S$ . PCR models similar to Eq. (5) are derived for all relief materials at the shelters.

Thus, once the ISP values are validated, a DropBox proceeds to forecast the demand for different relief materials at the shelters under its purview, putting the validated ISP values in derived PCR models.

**Enumerating utility** In this section, we estimate the constants a, b, c and d in the baseline methodology discussed in Sect. 5.1.2 (Eq. 2), a DropBox uses pre-loaded historical dataset on our application domain [21]. The dataset used consists of resource allocation data for different relief materials at different shelters over a period of time, the last 30 days. Such data is fetched from the blockchain network that stores allocation records for each relief materials at each shelter since day 1 of relief operations in different blocks.

A DropBox periodically executes least square method, for estimating the values of  $a, b, c$  and  $d$  and derive a utility function that can typically take the form as shown below:

$$u_{R_1s} = 0.1086 \left( \sum_{i=1}^n \prod_{\substack{j=1 \\ j \neq i}}^n \frac{t-t_j}{t_i-t_j} x_{irs} \right)^3 - 4.122 \left( \sum_{i=1}^n \prod_{\substack{j=1 \\ j \neq i}}^n \frac{t-t_j}{t_i-t_j} x_{irs} \right)^2 + 39.97 \left( \sum_{i=1}^n \prod_{\substack{j=1 \\ j \neq i}}^n \frac{t-t_j}{t_i-t_j} x_{irs} \right) - 0.0115 \tag{6}$$

**Table 3** Global mode table

Relief Mate-rial	ISP Values			
	ISP <sub>1</sub>	ISP <sub>2</sub>	...	ISP <sub>m</sub>
R <sub>1</sub>	gm <sub>ISP<sub>11</sub></sub>	gm <sub>ISP<sub>12</sub></sub>	...	gm <sub>ISP<sub>1m</sub></sub>
R <sub>2</sub>	gm <sub>ISP<sub>21</sub></sub>	gm <sub>ISP<sub>22</sub></sub>	...	gm <sub>ISP<sub>2m</sub></sub>
...	...	...	gm <sub>ISP<sub>j</sub></sub>	...
R <sub>n</sub>	gm <sub>ISP<sub>n1</sub></sub>	gm <sub>ISP<sub>n2</sub></sub>	...	gm <sub>ISP<sub>nm</sub></sub>

where  $u_{R_1s}$  is the utility for resource  $R_1$  in shelter  $S$ . Utility functions similar to Eq. (6) are derived for all relief materials at the shelters.

Thus, once demands are forecast, a DropBox proceeds to enumerate utility for different relief materials at the shelters under its purview, putting the  $x_{irs}$  values, obtained from the blockchain network, in the utility functions.

### 5.2.3 Allocating relief materials

Once, the control station receives the demand and utility values of all relief materials at different shelters, it initiates optimal resource allocation strategy, as shown in the control-node block of Fig. 3. The objective functions (3) and (4) are solved using integer programming that solves integer models using branch and bound methodology, to generate the resource allocation matrix.

To reduce total deployment time and deficit of resources, the proposed utility based system allocates an optimal amount of resources, having high utility, to the shelters from the most appropriate warehouses. The control station then transmits the resource allocation matrices to the respective DropBoxes.

### 5.2.4 Uploading transactions to blockchain

Once a DropBox receives the resource allocation matrix corresponding to the shelters under its purview, it creates smart contracts for creating new transactions (consisting demand, utility and allocated amount of relief materials for each shelter). The DropBox transmits the new transactions to the blockchain miners, who validate the transactions and add them as a new block in the blockchain.

If the number of approvals for a transaction exceeds 50%, then the transactions are validated. All such transactions, received from all DropBoxes regarding all shelters in a day, are validated by the blockchain network and added as a block. This activity of a DropBox is depicted in the “Transaction Uploading” side of the DropBox block of Fig. 3. On successful uploading of a transaction onto the blockchain the following details, as shown in Fig. 4, are generated. It shows one transaction with transaction id ‘tx’ generates hash (transaction Hash) and is attached with block Number 21. It is deployed from ‘0x51b...0.870’ which is the address from which the smart contract is deployed. It is uploaded to ‘0x8b...doe’ implying the blockchain uploading address. The sample transaction also shows its gas consumption and cumulative consumption. As, only one transaction is shown, both the values are same.

Thus, the smart contracts executed at the DropBoxes with influencing situational parameters, previous allocation records and current allocation matrix builds an immutable and globally accessible record of relief requirement vis-à-vis allocation in the Ethereum blockchain platform.

## 5.3 Our Claim

We claim that the ReliefChain system is immune to malicious behaviour of forwarder nodes and/or shelter nodes through its built-in mechanisms. We establish our claim for each such behaviours as follows:

```
{
  tx:
  '0x6f556bae27968158561547b1aa3c0e13ffbc4425e9040ddf73ee2c22a04208f0',
  receipt: {
    transactionHash:
    '0x6f556bae27968158561547b1aa3c0e13ffbc4425e9040ddf73ee2c22a04208f0',
    transactionIndex: 0,
    blockHash:
    '0x60287644648246891751cd0946c80e767724648ed55b38fdc7e127953ca39ee4',
    blockNumber: 21,
    from: '0x51b0089e75150cf064d476284e2ceddf5d556870',
    to: '0x8b9ed4fd3dee025ab57789cb4f7976a7540c2d0e',
    gasUsed: 46712,
    cumulativeGasUsed: 46712,
  }
}
```

Fig. 4 Sample transaction

#### Case – I: Malicious Forwarder(s)

The forwarder nodes may behave maliciously by transferring false ISP values to DropBoxes. So, DropBoxes upon receiving a ISP value, check whether the deviation (Sect. 5.2.2) of this ISP from the average (local mode  $lm_{ISP_j}$ , in our case) of the ISP values exceeds a predefined threshold. If this exceeds, the DropBox discards it. Thus, it thwarts the attempt of malicious forwarders to inflate the demand for relief materials in a shelter.

However, if majority of the forwarders collude to provide false ISP values then the system will compute a false  $lm_{ISP_j}$  value and will not be able to fish out false ISP values. This can be handled as is done in case – II, discussed below.

#### Case – II: Malicious Shelter-Node

The shelter nodes may behave maliciously by broadcasting false ISP values to the forwarders. All forwarders in turn, provide these false ISP values to a DropBox. The local validation scheme will not be able to fish out such false ISP values. This gets handled by the global validation scheme, discussed in Sect. 5.2.2, where a DropBox compares the current local mode,  $lm_{ISP_j}^{current}$ , with a global mode,  $gm_{ISP_j}$ . If the deviation  $dev_{ISP_j}^{current}$  is greater than a threshold, the DropBox discards the current ISP values provided by the shelter. Thus, it thwarts the attempt of malicious shelter nodes to inflate the demand for relief materials in a shelter.

Unlike multiple forwarders colluding, the case of more than one shelter-nodes colluding does not arise. There is no direct advantage for the shelters to collude among themselves, because colluding and diverting relief materials towards itself is an individual shelter-goal and not a community-goal.

### Case – III: Malicious Forwarders and Shelter-Node

If both forwarders and a shelter-node are malicious, the forwarders would further change the false ISP values provided by the shelters and submit these values to the DropBox. In this case also, the global validation scheme is able to detect and discard the false values by comparing the current ISP values with previous ones. Thus no amount of collusion could make the system vulnerable to irrational allocation of relief materials.

## 6 Performance analysis

In this section we evaluate the performance of the proposed ReliefChain system through extensive simulation in a realistic setting.

### 6.1 Simulation setup

The two main objectives of the performance analysis are (i) to observe how far the proposed system succeeds in achieving major design goals and (ii) to observe the overheads incurred for the blockchain activities. We use the LINGO [22] optimization tool for accomplishing the first objective and the Ethereum [23] blockchain platform for the second. Extensive simulation is performed with real disaster data [21], pertaining to the June 2013 cloudburst and flood centred on the North Indian state of Uttarakhand, for setting up the simulation environment. For this experiment, we consider five shelters, three warehouses, and five relief materials. We also assume 50 forwarders working in and around the shelters and opportunistically exchanging situational messages to the DropBox, using DTN routing protocol. The forwarders and/or shelters may act maliciously by changing ISP values. Average results of 20 independent runs of the simulation are used while plotting the simulation graphs.

#### LINGO Setup

We use the LINGO optimization tool to solve the integer programming model, proposed in Sect. 5.1.3, for optimal resource allocation, using the utility and demand matrices generated by the DropBoxes. The availability matrix, shown in Table 4, depicts the availability of different resources at three warehouses. The demand and utility matrices, shown

**Table 5** Demand & Utility Matrix

Shelter	Relief Material	Demand	Utility
S <sub>1</sub> (Uttarkashi)	R <sub>1</sub>	510	9
	R <sub>2</sub>	284	8
	R <sub>3</sub>	250	9
	R <sub>4</sub>	140	7
	R <sub>5</sub>	50	9
S <sub>2</sub> (Pithoragarh)	R <sub>1</sub>	570	7
	R <sub>2</sub>	205	8
	R <sub>3</sub>	350	5
	R <sub>4</sub>	30	7
	R <sub>5</sub>	60	6
S <sub>3</sub> (Chamoli)	R <sub>1</sub>	400	8
	R <sub>2</sub>	286	8
	R <sub>3</sub>	231	7
	R <sub>4</sub>	70	4
	R <sub>5</sub>	80	6
S <sub>4</sub> (Rudraprayag)	R <sub>1</sub>	490	9
	R <sub>2</sub>	271	8
	R <sub>3</sub>	232	9
	R <sub>4</sub>	110	8
	R <sub>5</sub>	55	9
S <sub>5</sub> (Nandaprayag)	R <sub>1</sub>	419	8
	R <sub>2</sub>	281	8
	R <sub>3</sub>	198	9
	R <sub>4</sub>	39	7
	R <sub>5</sub>	66	7

in Table 5, along with the availability matrix are fed to LINGO to solve the objective functions shown in Eqs. (3) and (4) for obtaining the resource allocation matrix, shown in Table 6.

#### Ethereum Setup

We implement and evaluate a prototype of our proposed ReliefChain system using Ethereum [23] as the blockchain platform. Our entire code is approximately 500 lines, consisting of smart contracts written in Solidity. The rationale behind using Ethereum is that it provides a broader range of validation strategies compared to the other blockchain platforms, with the power of smart contracts. We have deployed our implementation on a private Ethereum network consisting of one node, namely the DropBox running on Truffle v4.5. We have used a single machine

**Table 4** Availability Matrix

Warehouse	Availability of different resources				
	R <sub>1</sub> (Wheat)	R <sub>2</sub> (Rice)	R <sub>3</sub> (Tent)	R <sub>4</sub> (Blanket)	R <sub>5</sub> (Medicine)
W <sub>1</sub> (Badrinath)	550	360	150	80	90
W <sub>2</sub> (Haridwar)	450	400	180	90	110
W <sub>3</sub> (Dehradun)	860	240	120	100	80

**Table 6** Resource Allocation Matrix

Shelter	Allocated Amount				
	R1	R2	R3	R4	R5
S1	490	250	150	90	50
S2	300	100	50	30	45
S3	290	130	60	20	65
S4	400	260	100	100	55
S5	380	260	90	30	65

with Intel® Core™ CPU with 2.0 GHz, 8 GB of RAM, and 20 GB of persistent storage to meet the minimum hardware requirement for running Ethereum. Also, as a wallet for message transferring, we have used Geth v1.7.0. The shelter, forwarder and control nodes are also running alongside the Ethereum node.

For writing smart contracts pertaining to local and global validations, as mentioned in Sect. 5.2.2, the decision parameters, i.e., the optimal period of tolerances for forwarders and shelters, need to be computed. For this, we conduct experiments with periods varying from 3 to 15 days. It is observed that for smaller tolerance periods there are lower percentages of corrupted ISP values, which induces accurate relief allocation in the shelter. However, the total amount of ISP values is also considerably less, as shelters and forwarders are excluded from forwarding ISP values only after a few days, leading to insufficient information. On the other hand, for higher tolerance periods, the percentage of corrupted ISP values and total amount of ISP values increase simultaneously. However, with increasing period of tolerance, percentage of corrupted ISP values does not increase significantly, because in this case the corruption of ISP values is due to only DTN issues. From Table 7, we observe that after 9 and 11 days the percentage of corrupted ISP values saturate for forwarders and shelters respectively and hence consider these values as the optimal period of tolerances. Important simulation parameters and their values are provided in Table 8.

### 6.2 Simulation metrics

Performance of the proposed system is measured based on— design metrics and blockchain metrics.

#### Design metrics

1. Shelter Specific Resource Deficit (SRD) [6]: It is the rate at which resource deficits are minimized at each shelter and is defined as.

**Table 7** Optimal period of tolerant selection for forwarder and shelter

Node	Period of Tolerance (days)	% of Corrupted ISP Values
Forwarder	3	26
	5	30
	7	36
	9	39
	11	39
	13	39
	15	40
Shelter	3	25
	5	28
	7	32
	9	37
	11	38
	13	38
	15	38

$$SRD(t) = \frac{1}{\sum_1^R u_{rs}} \left( \sum_1^R u_{rs} * \left( 1 - \frac{\sum_1^W y_{rsw} * f_{rsw}(t)}{d_{rs}} \right) \right)$$

The function  $f_{rsw}(t) = 1$ , if  $t > t_{rsw}$  and  $=0$ , otherwise. Other symbols used here are mentioned in Sect. 5.1.3.

2. Average Resource Deficit (ARD) [6]: It is rate at which overall resource deficits minimized across all shelters and is defined as.

$$ARD(t) = \frac{\sum_{s=1}^S SRD(t)}{S}$$

3. Accuracy Factor (AF) [5]: It measures how accurate the forecast demand with respect to the observed demand and is defined as.

$$AF = 10 * \frac{1}{n} \sum_{i=1}^n \left| \log \frac{\hat{D}_{t,rs}}{D_{t,rs}} \right|$$

**Table 8** Important simulation parameters

Parameter	Value
No. of Shelters	5
No. of Warehouses	3
No. of Relief Materials	5
Period of Tolerance for Forwarders	9 days
Period of Tolerance for Shelters	11 days
Threshold of Deviation for Accepting ISP Values	10%
Initial Gas	3000000 wei
Number of Dropbox	1

where  $\hat{D}_{t_i,rs}$  and  $D_{t_i,rs}$  are the forecast and observed demands, respectively, for resource  $r$  in shelter  $s$  at time points  $t_i$ . The  $n$  time points may be days, weeks, or months. The ideal value of AF is 1. The smaller the value, the more accurate the forecasts.

#### Blockchain Metrics

4. Processing time ( $t_p$ ) [24]: It is the amount of time a user has to wait, after pressing the “send” transaction button ( $t_1$ ), to see their transaction appear on the blockchain ( $t_2$ ) and is defined as.

$$t_p = t_2 - t_1$$

5. Gas Consumption (G) [25]: It is the gas consumed in processing transactions ( $tx = 1, 2, \dots, n$ ) for the current block and is defined as.

$$G = \sum_{t_x=1}^n GasConsumption_{t_x}$$

### 6.3 Results and discussion

We conduct five sets of experiments with two different perspectives, as mentioned in Sect. 6.2. To observe how far the proposed system succeeds in achieving major design goals, we conduct three sets of experiments and compare the proposed system with the baseline work [5, 6] considering three design parameters SRD, ARD and AF. To observe the overhead introduced by the blockchain platform, we conduct two more sets of experiments to evaluate the impact of varying number of shelters and resources on two blockchain overheads  $t_p$  and G.

#### Achieving major design goals

**Shelter Specific Resource Deficit (SRD):** In this set of experiments, we compute and plot the SRD for each shelter. We observe from Fig. 5 that deficits are reduced at a faster rate in shelters S1 and S4 than in shelters S2,

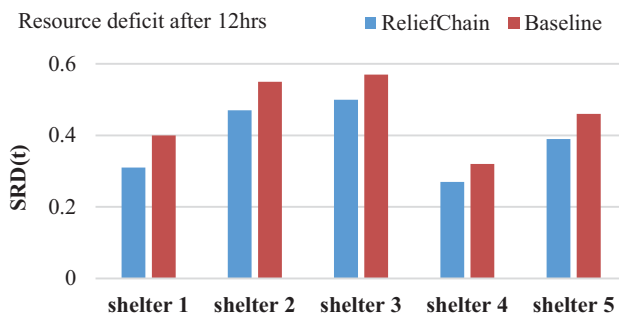


Fig. 5 Resource deficit at different shelters

S3, and S5. This owes to the fact that utilities of resources are higher in S1 and S4. Thus, our proposed ReliefChain system minimizes the deficit of resources in shelters that have higher utility. It is also observed that on an average, the proposed system performs better by reducing resource deficit by 17.2% than the baseline work. This is because our blockchain-based ReliefChain system uses validated and accurate shelter needs while allocating resources using the proposed validation strategy while the baseline one uses invalidated and near-accurate needs that are transmitted over a DTN.

**Average Resource Deficit (ARD):** In the next set of experiments, we study the impact of varying number of malicious forwarders on ARD. We observe from Fig. 6 that ARD increases with the increase in the number of malicious forwarders for both the proposed and baseline work, however ARD is consistently less in case of ReliefChain. The proposed system reduces ARD at a faster rate of 14.54% than the baseline work, even in presence of malicious forwarders. This is because our blockchain-based ReliefChain system uses validated shelter specific information through the proposed validation strategy. It is observed that ARD does not vary significantly when malicious forwarders increase up to 40%. However, this increases substantially when the percentage of malicious forwarders grows to 50%. Thus, ReliefChain can tolerate upto 40% of malicious forwarders to keep ARD at a tolerable limit.

**Accuracy Factor:** In this set of experiments we evaluate at which extent our forecast demand is close to the observed (real) demand. Precisely, we study the impact of validated ISPs (using the proposed validation strategy) over the non-validated ISPs in demand forecast. In Fig. 7 we provide results of forecast demand along with observed demand of shelters for the first 14 weeks for two representative resources R1 and R2. We also compare the performance with the Baseline i.e. PCR based forecast demand [5].

We observe from Fig. 7a that for resource R1, the average AFs are 1.59 and 1.96 for the ReliefChain and baseline respectively. Similarly, from Fig. 7b, for resource R2, these

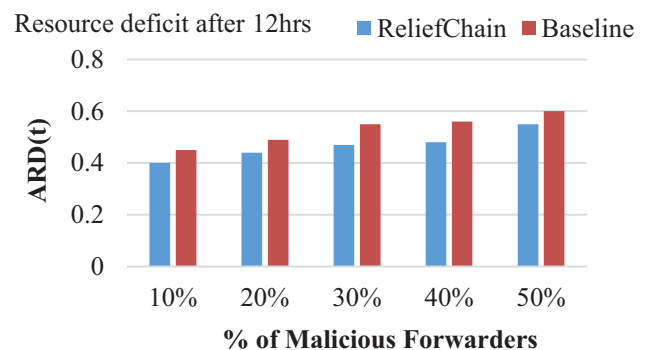
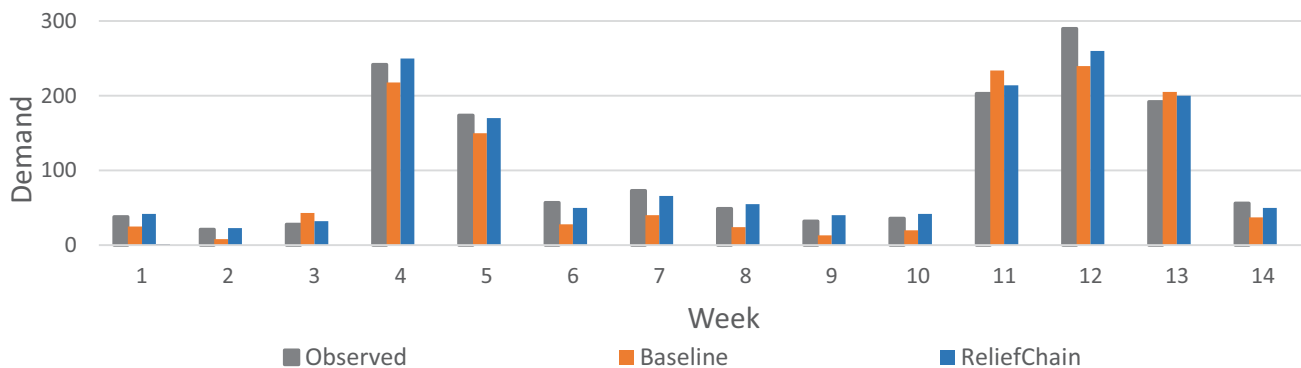
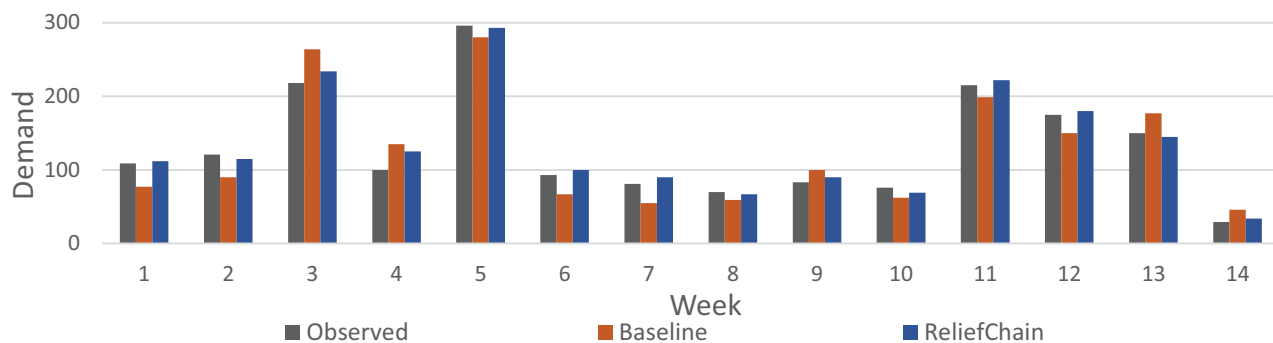


Fig. 6 Impact of malicious forwarders on average resource deficit





(a) For resource R1



(b) For resource R2

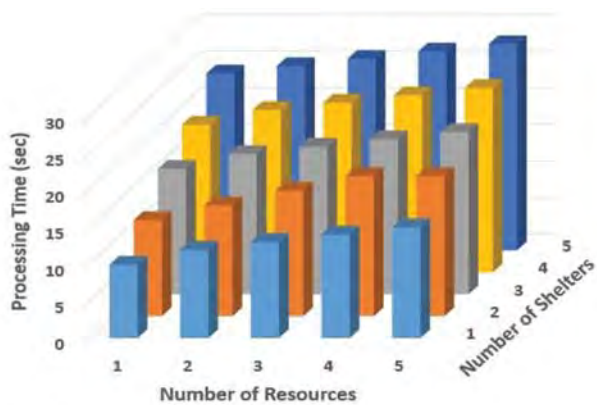
Fig. 7 Forecast demand vs. observed demand for first 14 weeks in shelters

values are 1.67 and 2.02 respectively. Thus, on an average, the proposed ReliefChain reduces AF by 18.09% indicating that the blockchain-based ReliefChain forecast demands are significantly closer to the observed demands. This is because our blockchain-based ReliefChain scheme uses validated

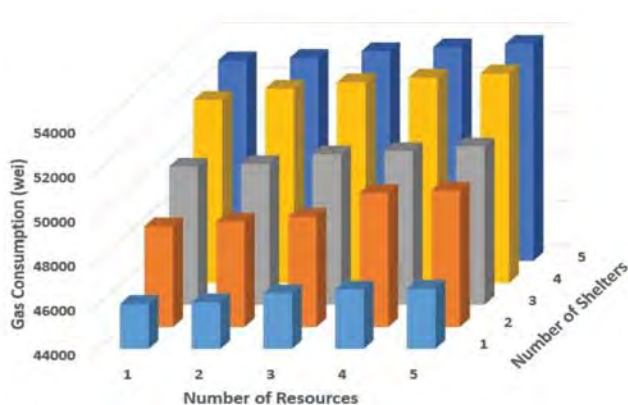
shelter-specific information using our proposed validation strategy.

*Overheads introduced by the blockchain platform*

*Processing Time ( $t_p$ ):* In this set of experiments, the impact of varying number of resources and shelters on the processing



(a) Processing time



(b) Gas consumption

Fig. 8 Blockchain overheads

time is measured. Figure 8a depicts that  $t_p$  increases with the increase in the number of shelters due to the higher number of transactions to be performed for global validation of each shelter. For example, with 4 resources,  $t_p$  with 4 shelters is 20.83% and 14.28% higher than the  $t_p$  with 2 and 3 shelters respectively. It is also observed that  $t_p$  does not increase significantly with increasing number of resources, as a slight increase in computation is involved in this case. Only 2% is the average increase in  $t_p$  with increasing number of resources.

**Gas Consumption (G):** In the next set of experiments, we measure the impact of varying number of resources and shelters on gas consumption as shown in Fig. 8b. We observe that gas consumption increases with increasing number of shelters, which incurs higher number of transactions to be performed for global validation. For example, with 4 resources,  $G$  with 4 shelters is 9% and 5% higher than the  $G$  with 2 and 3 shelters respectively. However, the rate of increase in the case of resources is not so significant. Only 3.2% is the average increase in  $G$  with increasing number of resources.

## 7 Conclusion and future work

This paper proposes ReliefChain, a blockchain leveraged post disaster relief allocation system working over a DropBox based DTN. The system creates an immutable and globally accessible record of relief requirement vis-à-vis allocation. Performance analysis justifies that the proposed ReliefChain system performs better than the baseline work by minimizing shelter specific resource deficit and average resource deficit at a faster rate, even in presence of malicious forwarders. In particular, ReliefChain reduces deficit of relief materials by 17.2% faster than the baseline work and tolerates upto 40% of malicious forwarders.

It is also observed that although the processing time and gas consumption for ReliefChain increase with the increasing number of shelters, the rate of increase in case of increasing number of relief materials is not so significant. In particular, the increase in processing time and gas are merely 2% and 3.2% respectively with increasing number of relief materials.

As future work, we aim to alleviate the necessity of Internet even at the DropBoxes by implementing an offline blockchain system that would be able to locally validate transactions without the intervention of the public blockchain system.

## Declarations

**Conflict of interests** The authors declare that they have no conflict of interest.

## References

1. Das N, Basu S, Das Bit S (2020) Efficient dropbox deployment towards improving post disaster information exchange in a smart city. *ACM Trans Spat Algorithms Syst* 6(2):1–18
2. Chakraborty C, Roy S, Basu S (2019) Intention aware misbehavior detection for post-disaster opportunistic communication over peer-to-peer DTN. *Springer Peer Peer Netw Appl* 12(4):705–723
3. Basu S, Roy S (2014) A group-based multilayer encryption scheme for secure dissemination of post-disaster situational data using peer-to-peer delay tolerant network. In: *Proceedings of 3rd International Conference on Advances in Computing, Communications and Informatics*, pp 1566–1572
4. Bhattacharjee S, Kanta S, Modi S, Paul M, DasBit S (2016) Disaster Messenger: An Android based Infrastructure Less Application for Post Disaster Information Exchange. In: *Proceedings of IEEE ANTS*, pp 1–5
5. Basu S, Roy S, DasBit S (2018) A post disaster demand forecasting system using principal component regression analysis and case-based reasoning over smartphone based DTN. *IEEE Trans Eng Manage* 66(2):224–239
6. Basu S, Roy S, Bandyopadhyay S, DasBit S (2018) A utility driven post disaster emergency resource allocation system using DTN. *IEEE Trans Syst Man Cybern Syst* 50(7):2338–2350
7. Panesir M (2018) Blockchain Applications for Disaster Management and National Security. Ph.D. Thesis, University of Buffalo, State University of New York, pp 1–115
8. Siemon C, Rueckel D, Krumay B (2020) Blockchain Technology for Emergency Response. In: *Proceedings of 53rd Hawaii International Conference on System Sciences*, pp 614–623
9. Ojetunde B, Shibata N, Gao J, Ito M (2015) An Endorsement-based Mobile Payment System for a Disaster Area. In: *Proceedings of IEEE 29th International Conference on Advanced Information Networking and Applications*, pp 1–8
10. L'Hermitte C, Nair N (2021) A blockchain-enabled framework for sharing logistics resources in emergency operations. *Wiley* 45(3):527–554
11. Badarudin P, Wan A, Phon-Amnuaisuk S (2020) A Blockchain-based Assistance Digital Model for First Responders and Emergency Volunteers in Disaster Response and Recovery. In: *Proceedings of IEEE 8th International Conference on Information and Communication Technology (ICoICT) 2020*, pp 1–5
12. Demir M, Mashatan A, Turetken O, Ferworn A (2018) Utility Blockchain for Transparent Disaster Recovery. *Proceedings of IEEE Electrical Power and Energy Conference (EPEC) 2018*, pp 1–6
13. Chakrabarti C, Basu S (2019) Blockchain Based Incentive Scheme for Post Disaster Opportunistic Communication over DTN. In: *Proceedings of the 20th ACM International Conference on Distributed Computing and Networking (ICDCN) 2019*, ACM Digital Library, pp 385–388
14. Basu S, Chowdhury S, DasBit S (2021) Using Blockchain in Intermittently Connected Network Environments. *Blockchain Technology and Innovation in Business Process, Smart Innovation, Systems and Technologies*, Springer Nature Singapore Private Ltd 219(3):33–47
15. Paul A, Qu X, Wen Z (2021) Blockchain—a promising solution to internet of things: A comprehensive analysis, opportunities, challenges and future research issues. *Peer Peer Netw Appl* 14:2926–2951
16. Khan S, Loukil F, Ghedira-Guegan C, Benkhelifa E, Bani-Hani A (2021) Blockchain smart contracts: Applications, challenges, and future trends. *Peer Peer Netw Appl* 14:2901–2925

17. Park Y, Sur C, Rhee K (2018) A secure incentive scheme for vehicular delay tolerant networks using cryptocurrency. *Secur Commun Netw* 2018(3):1–13
18. Lee W (2018) Understanding Blockchain: A Beginners Guide to Ethereum Smart Contract Programming. <https://www.codemag.com/Article/1805061/Understanding-Blockchain-A-Beginners-Guide-to-Ethereum-Smart-Contract-Programming>. Accessed 6 May 2022
19. Lindgren A, Doria A, Schelén O (2003) Probabilistic routing in intermittently connected networks. In: *Proceedings of SIGMOBILE CCR 2003*, pp 19–20
20. Nicholas J (2006) Introduction to Descriptive Statistics. <https://www.sydney.edu.au/content/dam/students/documents/mathematics-learning-centre/introduction-to-descriptive-statistics.pdf>. University of Sydney, pp 1–42
21. Uttarakhand Disaster. <http://www.ndtv.com/article/india/uttarakhand-thousands-still-strandedsee-over-view-383781>. Accessed 6 May 2022
22. LINGO—Optimization Modeling Software for Linear, Nonlinear, and Integer Programming, Lindo Systems Inc., [www.lindo.com](http://www.lindo.com). Accessed 6 May 2022
23. Wood G (2014) Ethereum: A secure decentralized generalized transaction ledger. *Ethereum Project Yellow Paper* 151:1–32
24. Michelinet R et al (2018) SpeedyChain: a framework for decoupling data from blockchain for smart cities. In: *Proceedings of 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services 2018, ACM*, pp 145–154
25. Gas usage. <https://www.investopedia.com/terms/g/gas-ethereum.asp>. Accessed 6 May 2022

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.



**Nabanita Das** is currently pursuing PhD in the Department of Computer Science and Technology at Indian Institute of Engineering Science and Technology, Shibpur, India. She has published one research article in a reputed journal *ACM Transactions on Spatial Algorithms and Systems* and three in international conference proceedings. Her research interests include delay tolerant network, mobile and wireless technologies.



**Souvik Basu** is the Head of the Department of Computer Applications at the Heritage Institute of Technology, Kolkata. With over 18 years of teaching experience, his major research interests are in opportunistic and delay-tolerant networks, network & information security, and mobile and wireless technologies for ICT-based social applications.



**Sipra Das Bit** is a Professor of the Department of Computer Science and Technology, Indian Institute of Engineering Science and Technology, Shibpur, India. She also served as a visiting professor in the Department of ICT, AIT, Bangkok in 2017. A recipient of the Career Award for Young Teachers from the All India Council of Technical Education (AICTE), she has more than 30 years of teaching and research experience. Professor Das Bit has published many research papers in reputed journals and refereed International conference proceedings. She also has four books/monographs to her credit. Her current research interests include IoT,

wireless sensor network, delay tolerant network, and mobile computing.