vice, so A asks B for a recommendation with respect to the car service category, assuming that A trusts B's recommendation within this category. When B receives this request and finds that it doesn't know D either, B forwards A's request to C, which has D's trustworthiness information within the car service category. C sends a reply to A with D's trust value. The path A × B × C × D is the *recommendation path*.

We use the following formula to calculate the trust value from the returned value[1]: $tv\_T = [rtv(1)/4] \times [rtv(2)/4] \times \ldots \times [rtv(i)/4] \times \ldots \times [rtv(n)/4] \times tv(T)$, where $rtv(i)$ is the trust value of the ith recommender in the recommendation path, $tv(T)$ is the trust value of target T returned by the last recommender, and tv_T is the calculated trust value of target T.

When multiple recommendation paths exist between the requester and the target, the target's eventual trust value is the average of the values calculated from different paths.

This model has some weaknesses:

- It doesn't consider false recommendations and assumes that a recommender with a good recommender trust value always makes reliable recommendations, which might not be true.
- It doesn't provide a mechanism for monitoring and reevaluating trust, which is dynamic.

Trust shouldn't be considered a binary concept (that is, either to trust or not to trust). Abdul-Rahman and Hailes quantified trust as a multiple value concept.[1] Many trust-management systems use the same approach.[2,5,9,10] The key challenge then is how to process the trust values to minimize the influence of false recommendations.

We can classify the trust models into two categories:

- *evidence-based model*, in which entities establish a trust relationship based on some evidence, such as keys;[6-8]
- *recommendation-based model*, in which recommendations from intermediaries set up the trust relationship between two strangers.[1]

We can also place the trust-management systems for distributed systems into these two categories. For example, Laurent Eschenauer, Virgil Gligor, and John Baras[11] used an evidence-based approach, while Li Xiong and Ling Liu[2] and Karl Aberer and Zoran Despotovic[3] used a recommendation-based approach.

TRUST MANAGEMENT IN P2P SYSTEMS

P2P systems are distributed systems without centralized control or organization. The peers interact directly and are both consumers and service providers. P2P systems need trust management to ensure cooperation—for example, to reduce free riding.

Table 1. Direct trust value.

Value	Meaning	Explanation
–1	Distrust	Completely untrustworthy
0	Ignorance	Can't decide
1	Minimal	Lowest trust
2	Average	Mean trustworthiness
3	Good	Trusted by major population
4	Complete	Fully trustworthy

Table 2. Recommender trust value.

Value	Meaning	Explanation
–1	Distrust	Completely untrustworthy
0	Ignorance	Can't decide
1	Minimal	The entity itself judges the reliability of recommender's recommendation.
2	Average	
3	Good	
4	Complete	

In P2P systems, peers often must interact with unknown entities whose trustworthiness is also unknown. Centralized schemes or schemes that rely on global knowledge won't work.

Recommendation-based trust management

Xiong and Liu[2] based their distributed trust-management system on feedback or recommendations that help establish trust relationships between unknown or unfamiliar peers. They define a satisfactory interaction as 1 and a complaint as 0. Their trust metric is

$$T(u,t) = \frac{\sum_{v \in P,\, v \neq u} S(u,v,t) \cdot Cr(v,t)}{\sum_{v \in P,\, v \neq u} I(u,v,t)} \quad (1)$$

where

- P is a set of peers in the P2P system;
- u and v are peers in the system, $u, v \in P$;
- $S(u,v,t)$ is the degree of satisfaction that u has with v until the tth transaction;
- $T(u,t)$ is u's trust value evaluated by other peers until the tth transaction;
- $Cr(v,t)$ is the balance factor for filtering feedback from v; and
- $I(u,v,t)$ is the number of interactions that u has with v up to the tth transaction.

So, $T(u,t)$ is the ratio of the cumulative weighted satisfaction that u receives to the total number of interactions that u has within the P2P system.