Trust-based Service Composition and Binding with Multiple Objective Optimization in Service-Oriented Mobile Ad Hoc Networks

APPENDIX A

In this appendix, we provide details of trust propagation and aggregation in the single-trust baseline protocol design. A node receiving a trust update follows the propagation and aggregation protocol described below to update its (α, β) pair toward the SP. Trust propagation is done through recommendations received from 1-hop neighbors whom the trustee encounters dynamically. A node (trustor) will select n_rec recommenders whom it trusts most to provide trust recommendations of an SP (trustee). A recommender should only pass its direct interaction experience with the trustee node in terms of (α, β) as a recommendation to avoid dependence and looping [25]. Let node i be the trustee, node j be the trustor, and node k be a recommender. Also let (α_{ij}, β_{ij}) be the recommender’s (α, β) toward the trustee, (α_{kj}, β_{kj}) be the recommender’s (α, β) toward the trustee and (α_{jk}, β_{jk}) be the trustor’s (α, β) toward the recommender. Based on belief discounting (see [24] for details), node i will compute its new (α_{ij}^{new}, β_{ij}^{new}) as follows:

\[ α_{ij}^{new} = α_{ij} + \frac{2α_{ik}α_{kj}}{((β_{ik} + 2)(α_{kj} + β_{kj} + 2)) + 2α_{ik}} \quad (14) \]

\[ β_{ij}^{new} = β_{ij} + \frac{2α_{ik}β_{kj}}{((β_{ik} + 2)(α_{kj} + β_{kj} + 2)) + 2α_{ik}} \quad (15) \]

The basic idea is that if node i does not trust k, it will discount the recommendation provided by node k, so α_{ij}^{new}~α_{ij} and β_{ij}^{new}~β_{ij} as if the recommendation from k does not have any effect. This can be derived from (14) and (15). First of all, if node i does not trust node k then α_{ik} ≪ β_{ik}. In case node k is performing a bad-mouthing attack on node j, then α_{kj} ≪ β_{kj}. Applying these two conditions to (14) and (15), one can easily verify α_{ij}^{new}~α_{ij} and β_{ij}^{new}~β_{ij}. In case node k is performing a balloting attack on node j, then α_{k,j} ≫ β_{k,j} and again one can easily verify α_{ij}^{new}~α_{ij} and β_{ij}^{new}~β_{ij}. After trust aggregation, the trustor’s (or node i’s) trust toward the trustee (or node j) is then computed as \( T_{ij} = \frac{α_{ij}^{new}}{α_{ij}^{new} + β_{ij}^{new}} \).

APPENDIX B

In this appendix, we provide implementation details of the ILP solution technique for optimally solving the node-to-service assignment problem with MOO in (9) of the main file for both trust-based and non-trust-based algorithms.

Table IV defines the variables used in the ILP formulation. There is only one decision variable, namely, \( w_{j,k,m} \) to be determined by the ILP, specifying if node j should be assigned to abstract service k of service request m. The ILP will search for an optimal solution of \( w_{j,k,m} \) for all j’s, k’s and m’s to maximize MOO in both trust-based design and non-trust-based design algorithms. The objective function MOO = \( \sum_{m \in T}(ω_Q m D_m + ω_D m D_m + ω_C m C_m) \) as defined by (9) of the main file can be computed as a linear function of \( w_{j,k,m} \) (the only decision variable to be decided by the ILP). The service-to-node assignment MOO problem is formulated as follows:

**Given**: \( T, S_m, N \)

**Calculate**: \( ov_{p,q}, s_j,k, tt_{j,k}, in_{k,m} \)

**Find**: \( w_{j,k,m} \)

**Maximize**: \( \sum_{m \in T}(ω_Q m D_m + ω_D m D_m + ω_C m C_m) \)

**Subject to**: \( \forall (p,q) ov_{p,q} \times (w_{j,k,p} + w_{j,k,q}) \leq 1; \)

\( \sum_{j} w_{j,k,m} = in_{k,m}; w_{j,k,m} \leq to_{j,k,m} \)

**REFERENCES**
