Asadi, F., & Richards, A. G. (2015). Ad Hoc Distributed Model Predictive Control of Air Traffic Management. In 16th IFAC Workshop on Control Applications of Optimization CAO’2015: Garmisch-Partenkirchen, Germany, 6–9 October 2015. (pp. 68-73). (IFAC Papers Online; Vol. 48, No. 25). Amsterdam:Elsevier. DOI: 10.1016/j.ifacol.2015.11.061

Ad Hoc Distributed Model Predictive Control of Air Traffic Management

Fatemeh Asadi, Arthur Richards

Aerospace Engineering Department, University of Bristol, BS8 1TR
UK (e-mail: elham.asadi@ Bristol.ac.uk; Arthur.richards@bristol.ac.uk)

Abstract: This paper develops an ad hoc distributed control algorithm for air traffic management. The method is based on model predictive control, in which aircraft use optimization to determine their own flight trajectories. The coordination approach of Self-organized Time Division Multiple Access is used to ensure no two aircraft re-optimize simultaneously, thus ensuring collision avoidance. Unlike existing distributed predictive control, which requires a pre-organized optimizing sequence, this new approach requires no central coordinator.

Keywords: Ad Hoc Distributed Model Predictive Control, Air Traffic Management, Trajectory Design, Space-TDMA and Self-Organized TDMA.

1 INTRODUCTION

Free flight has been proposed as a way to handle ever-increasing air traffic demands and to provide economic benefits to airspace users. One of the primary concepts behind free flight is the transferring of responsibility for separation between aircraft, from air traffic controllers to pilots, which gives the aircraft freedom to select their path and speed in real time (Metzger and Parasuraman 2001). This can be treated as a distributed control problem where each aircraft optimizes its own objective while maintaining safe separation. Here, we have used Distributed Model Predictive Control (DMPC) (Maestre and Negenborn 2014). The main goal of this paper is finding a distributed trajectory optimization strategy for network of aircraft during their flights over the airspace sectors.

Several strategies for DMPC have been presented in the literature which could be categorized by the type of coupling or interactions between constituent subsystems (Scattolini, 2009). For example, dynamically coupled systems (Dong et al., 2001; Dunbar, 2007a), coupling via the cost function (Borrelli and Keviczky 2006; Dunbar 2007b; Dunbar and Murray 2006) and subsystems sharing coupled constraints (Richards and How 2004a, 2004b; Kuwata et al., 2007; Keviczky et al., 2004 & 2006). This paper has focused on the air traffic problem in which systems are dynamically decoupled but have coupled constraints. One distributed control strategy for solving this kind of problems, is using the serial scheme where in each time step just one of the coupled agents optimizes to respect its neighbours' published intensions by freezing their plan and exchanges the new plan to achieve constraint satisfaction. Serial scheme demands some agreements of sequence. Existing researches employ a predetermined sequence of optimizing across the agents which needs centralized coordinator to determine this optimization order (Richards and How 2004a, 2004b; Keviczky et al., 2004a, 2004b; Kuwata, et al., 2006; Trodden, et al., 2006). Therefore, the predetermined sequence is not scalable to the large numbers of agents present in Air Traffic Management (ATM) problems.

We propose "ad hoc" distributed MPC for ATM problem in which aircraft entering and leaving the area dynamically, implement a decentralized approach to sequencing. In the decentralized sequencing, planning collision can happen when two coupled agents determining their sequence of optimization in a group at the same time. This problem is analogous to multiple access to a shared communications channel, in which a collision occurs when two stations transmit at the same time. So, the ideas from communication are adopted here (Keiser, 1989; Pahlavan and Levesque, 2005; Kumar et al., 2004). In particular, for channels using Time Division Multiple Access (TDMA), coordination involves allocation of time slots amongst transmitting agents, analogous to the allocation of slots for optimizing (Rom and Sidi 1989). Self-organizing TMDA (STDMA) which is common for wireless communications sharing a channel, performs this allocation in a distributed fashion without any central coordinator (Gaugel, et al. 2013) and is already used for aviation data link and Mode S (Gustavsson 1996).

The paper is organized as follows. Section 2 begins by presenting the distributed MPC problem. Then, section 3 reviews STDMA method. Section 4 proposes an ad hoc distributed MPC for aircraft conflict avoidance, exploiting ideas from network systems to determine the optimization time of each agent. Finally, preliminary results from numerical simulation using this new algorithm are presented in section 5.

2 DISTRIBUTED MODEL PREDICTIVE CONTROL PROBLEM

Consider the DMPC of a system containing N_v subsystems with decoupled dynamics and coupled constraints. The model predictive control problem for each subsystem is as follows:

$$J_p \; = min_{\{U_t^p\},Q_t^p} \{\textstyle\sum_{k=0}^{N-1} l^p(u_{k,t}^p, x_{k,t}^p) + l_N^p(x_{N,t}^p)\} \qquad (1)$$

Subject to :

$\forall p = 1, \dots, N_v \quad , \; \forall k = 1, \dots, N,$

$$x^{p}_{k+1,t} = f^{p}(u^{p}_{k,t}, x^{p}_{k,t}) \, , \tag{2}$$

$$g^{p,q}(x^{p}_{k,t}, x^{q}_{k,t}) \leq 0 \, , \; , \; \forall q \in C_{p} \tag{3}$$

$$x^{p}_{k,t} \in \mathcal{X}^{p} \, , \tag{4}$$

$$u^{p}_{k,t} \in \mathcal{U}^{p} \, , \tag{5}$$

$$x^{p}_{N,t} \in Q^{p}_{t} \, , \tag{6}$$

$$x^{p}_{0,t} = x^{p}_{t} \, , \tag{7}$$

where N is prediction horizon, C_p is the set of current neighbours of subsystem p, $x^{p}_{k,t}$ is the measured states of subsystem p at step k and l^{p}_{N} is the cost on the terminal state. The decision variables are the control inputs, $u^{p}_{k,t}$, and the terminal invariant set, Q^{p}_{t}, that ensures the safety of the subsystem p beyond the planning horizon respect to its current neighbours.

In sequential DMPC, subsystems who are coupled through their constraints, cannot renew their plan simultaneously. A specific sequence is applied for optimization.

3 STDMA Algorithm

STDMA is a decentralized MAC method in which the network members are responsible for sharing the communication channel. Like TDMA system, time is divided into frames. These frames are further divided into slots, which typically corresponds to one packet duration. Each network member (node) will randomly select a number of free slots within each frame to transmit in (Gaugel, et al. 2013, Bilstrup, et al. 2009). The procedure of slot assignment which is carried out by each node is as follows (as shown in figure 1):

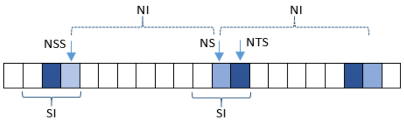


Figure 1. Picking the slots in the STDMA algorithm

- Each node determines its report rate, corresponding to how many slots that needs to be reserved in each frame.
- It will listen to the channel activity during one frame length to find which slots are occupied and what the position is of the node using it.
- Calculate a Nominal Increment (NI) by dividing the number of slots with the report rate.
- Randomly select a Nominal Start Slot (NSS) drawn from the current slot up to NI.
- Determine a Selection Interval (SI) of slots as 20% of NI and put this interval around the NSS.
- Pick the Nominal Transmission Slot (NTS) randomly within the interval SI around NSS. If the randomly chosen NTS is occupied, then the closest free slot within SI is chosen. If all slots within the SI are occupied, the slot used by a node furthest away from oneself will be chosen. The selected slot is the first actual slot to be used for transmission.
- Assign a Nominal Slot (NS) by adding NI to NSS. Then, the interval SI is placed around NS and the procedure of determining the next NTS will start over again. This procedure will be repeated as many times as decided by the report rate.

4 AD HOC DISTRIBUTED CONTROL FOR AIR TRAFFIC MANAGEMENT

In aeronautics, airspaces are the portion of the atmosphere controlled by a country above its territory. These airspaces are divided into smaller "sectors". In this paper, each sector is assumed to have an associated coupled control problem, with aircraft entering and leaving as they progress along their paths. Each aircraft is assumed coupled to others in the same sector and decoupled from all others. Future work could also consider the abolition of sectors in favour of dynamic sets of neighbours, but this is beyond the scope of this paper. Each aircraft has its own objective function which is minimizing its flight time, along with coupled constraint which is avoiding its neighbours.

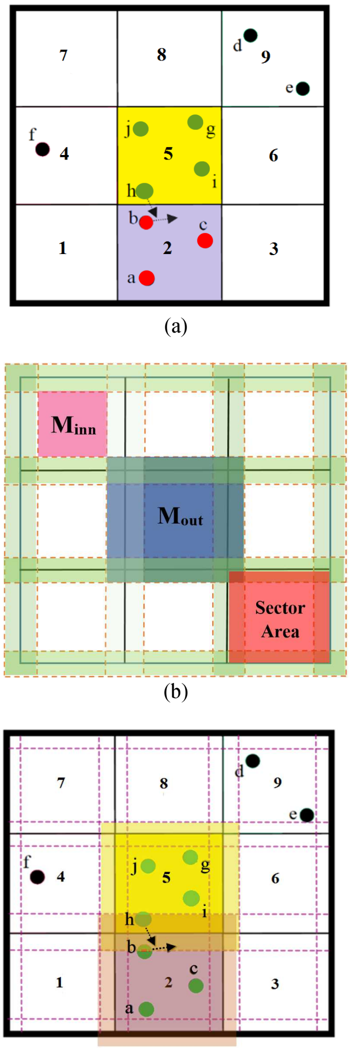
Building on Space-Time Division Multiple Access scheme (Amouris 2001), one virtual frame with infinite time slots is assumed for each airspace sector. Agents should determine their optimization turns before entering into each sector by selecting desirable time slots correspond to the sector's frame. Each agent can re-optimize its plan at its selected time slots. Agents in different sectors can re-plan at the same time. For example, agents *f, g, c* and *e* could re-optimize their trajectories at the same time; but, agents *g* and *i* cannot have simultaneous optimization (figure 3.a).

The terminal invariant set guaranties the safety of each agent regards to its current neighbours. This invariant set is the straight-line extrapolation of the aircraft terminal state which does not intersect any current neighbours' trajectories (Patel and Goulart 2011). However, the neighbours of each aircraft are changing dynamically. This could cause a problem especially when the agents change their sectors. For example, as it is depicted in figure 3.a agent *h* might collide with agent *b* just after entering into sector 2; as it had not considered agent *b* as its neighbour. So, each agent should consider the agents of its next sector as neighbours, to some distance before entering into that sector. For resolving this issue, a safety margin area is constructed around each airspace sector (figure 3.b). Agents who are located in the safety margin area (green area; $M_{margin}= M_{out} \backslash M_{inn}$) and are optimizing their path, should consider all agents in M_{out} of the sectors which are sharing this area, e.g. agent *h* should avoid agents *g, i, j* and *a, b, c* when it is optimizing its path at M_{margin} (figure 3.c). Accordingly, agents *g, i, j, a, b* and *c* should freeze their plan when *h* is planning in M_{margin}.

The next issue is the assigning of suitable time slots to each agent. This task is done by each agent before entering into a new sector. Agents can start the time slot assignment process for the next sector as soon as leaving M_{inn} and arriving to the margin area. In this position, the agent estimates its arrival time into the next sector, t_0, exiting time from that, t_f, arrival time into the next M_{inn}, t_{m1}, and the exiting time from that, t_{m2}, based on its recent optimized trajectory. Depending on the length of the flight over the M_{inn} area, it calculates the Optimization Rate (OR) which determines the number of required time slots for optimization in the next sector (eq. 8).

$$OR = \left\lfloor \frac{t_{m2} - t_{m1}}{k} \right\rfloor + 1 \tag{8}$$

Here, k denotes the demanded optimization period defining how frequently the re-optimization should happen and $\lfloor x \rfloor$ is floor function. Each agent needs at least one time slot in each sector to increase its chance of getting a time slot in upcoming sectors.



(a)

(b)

(c)

Figure 3. (a) Neighbour selection policy without margin (b) Example of M_{inn}, M_{out}, Sector area & margin area (c) Neighbour selection policy after defining margin area

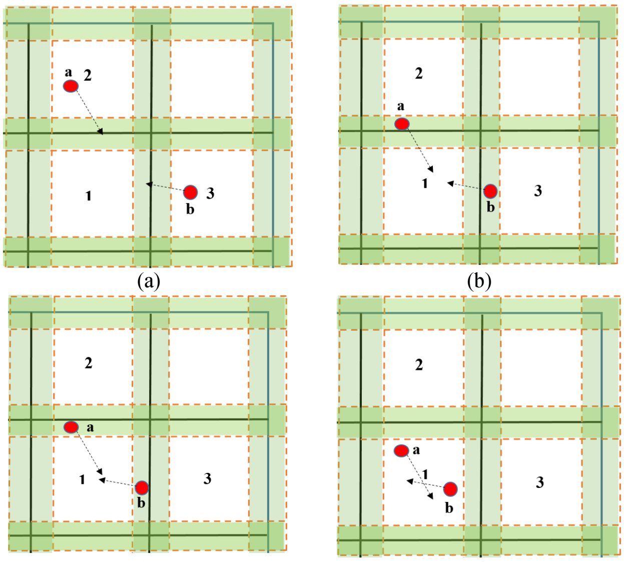
Now, the agent begins the slot assignment process by monitoring the frame of the next sector. Then it attempts to find *OR* free time slots in the time interval [t_{m1} t_{m2}] by running the modified STDMA algorithm (outlined algorithm 1).

In succeeding planning for the next sector, the agent enters into the new sector and by arriving at M_{inn} trajectory optimization at the appointed time slots will be started. When the agent finishes its optimization at the last slot, it is still in M_{inn} area. So, it re-computes the t_{m2} (exit time from M_{inn}) and t_f (exit time from the current sector) by using fresh information. Then, it tries to find a free slot in the margin area at this time interval [t_{m2} t_f]. Since margin area is common between more than one M_{out}, the agent looks for a slot which is free in the frame of all sectors who are sharing this margin area. One likely problem is planning collision which could happen when two or more non-neighbour agents attempt to plan for a same frame simultaneously.

Algorithm 1. Modified STDMA algorithm in ad hoc distributed control

```
1.  Calculate the number of slots at your desired time interval [$t_0$ $t$]:
        $n_s = \left\lfloor \frac{t - t_0}{length\ of\ one\ slot} \right\rfloor$
2.  Calculate the nominal increment: $NI = \frac{n_s}{OR}$
3.  Separate the slots between $N_{t0}$ and $N_t$ (the slot which belongs to $t_0$
    and $t$) into $OR$ parts with length of $NI$.
4.  For $i = 1:OR$ do
      i.   Randomly select a Nominal Slot (NS) drawn from the $i^{th}$
           slot section.
      ii.  If the randomly chosen $NS$ is occupied, then search for
           another free slot in this section (within [$N_{t0} + (i-1) \times NI$
           $N_{t0} + i \times NI - 1$])
      iii. If all slots within [$N_{t0} + (i-1) \times NI$   $N_{t0} + i \times NI - 1$] are
           occupied, skip this section and go to the next slot section.
5.  Update $OR$ based on the number of slots that you have got in this
    process.
```

Imagine two agents *a* and *b* who are in the different sectors (see figure 4.a). Since *a* and *b* are not neighbour, they had their last optimization at the same time. Now, they are monitoring the frames of their current and next sectors (i.e. which slots are occupied). Agent *a* should check the frames of first and second sector and agent *b* should inspect frames of first and third sector to find one free time slot in both frames at desired time interval [t_{m2} t_f]. Since the first frame is common between these two agents, they will have a conflict while making a decision about this frame. Planning collision will happen when these agents want to announce their selected time slots to the other agents. As a result, none of them could gain a time slot. Both agents continue their current plan and after waiting a random time attempt to find a free slot again. If another collision occurs, the random waiting time is increased. This process which is similar to Carrier Sense Multiple Access with Collision Detection (CSMA/CD) protocol (Rom and Sidi 1989) continues until a slot is assigned. However, the time interval, which is explored for finding a free slot depends on the agent's position at the moment of decision. To obtain a free time slot at current time, *t*, different time intervals should be explored in the different positions corresponding to figure 4 (see table 1).



(a) (b)

(c) (d)

Figure 4. Different positions of the agents with communication collision

Table 1. Time intervals which should be investigated in different positions for finding a free time slot

	(Fig 4.a)	(Fig 4.b)	(Fig 4.c)	(Fig 4.d)
agent a	Needs a free slot in frame 1 & 2 at [$t_{m2}(2,a)$ $t_f(2,a)$]	Needs a free slot in frame 1 & 2 at [t $t_f(2,a)$]	Needs a free slot in frame 1 & 2 at [t $t_{m1}(1,a)$]	Needs a free slot in frame 1 at [t $t_f(1,a)$]
agent b	Needs a free slot in frame 1 & 3 at [$t_{m2}(3,b)$ $t_f(3,b)$]	Needs a free slot in frame 1 & 3 at [t $t_f(3,b)$]	Needs a free slot in frame 1 & 3 at [t $t_{m1}(1,b)$]	Needs a free slot in frame 1 at [t $t_f(1,b)$]

If planning collisions prevent an aircraft from obtaining a slot, then it is possible for an aircraft to enter a sector without having obtained slots for coordinating with its new neighbours. In this circumstance, the aircraft will continue to try to get slots until it succeeds. This situation is dangerous, in the sense that it permits collisions, as one agent comes into a sector without adopting its trajectory respect to the new neighbours who are already flying in that sector. An outline of the ad hoc distributed control algorithm is shown in algorithm 2.

Algorithm 2. Ad Hoc Distributed Control Algorithm for One Agent

```
Based on the timing, if it is your optimization turn:

  1. Find your neighbours based on your current position
       i. If you are in $M_{inn}$ area, the neighbours are all the
          agents who are currently located in your $M_{inn}$,
      ii. If you are in margin area, the neighbours are all agents
          located in $M_{out}$ of the sectors which are sharing this
          margin area,

  2. Re-optimize your trajectory
  3. If it is your last optimization in this sector,
       i. Re-calculate your exit time from current $M_{inn}$ and
          current sector ($t_{m2}$ , $t_f$),
      ii. Monitor the frame of your current and next sector and
          find one slot at time interval [$t_{m2}$ , $t_f$] which is free in
          both frames,
     iii. Transmit your intention,
      iv. If communication collision happened, repeat the
          following process until getting a free slot:
            a. wait a random time while continuing your path,
            b. re-calculate your available time,
            c. find one free slot based on your current position
               and available time in appropriate frames,

If it is your slot assignment turn:

  1. If you had a chance of slot assignment for your current
     sector,
       i. Calculate the entrance/exit time into/from $M_{inn}$ of your
          next sector,
      ii. Run STDMA algorithm for [$t_{m1}$ $t_{m2}$] of the next sector
          to find suitable free slots in next sector,

  2. If you have entered into your current sector without
     planning
       i. Calculate the available time before exiting the current
          $M_{inn}$,
      ii. Implement STDMA algorithm for [t $t_{m2}$] current sector
          to find suitable free slots in your current sector.
```

5 Simulation

For simulation of ad hoc distributed control in context of air traffic management, a control area including nine airspace sectors has been considered. At starting time, t_0, four agents are produced in the random start positions at four sides of the control area. Destinations of these agents are placed in the random locations on the opposite side of the start points. The method outlined in section 4 has been implemented in MATLAB. The program is simulating a network of vehicles for 1000 seconds. Every *T* seconds, four new agents which are produced in the random places on different sides of the control area, are added to the network. Each agent generates an initial trajectory for itself, before entering into the control area. It is unlikely that the initially generated trajectory be feasible. Therefore, it is used for specifying the agents' first airspace sector in the control area. When agents have found their first sector, they attempt to get a time slot in the first margin area.

Different population densities have been examined for performance analysis of the proposed method. Figure 5 provides a snapshot of the control area during the programs run.

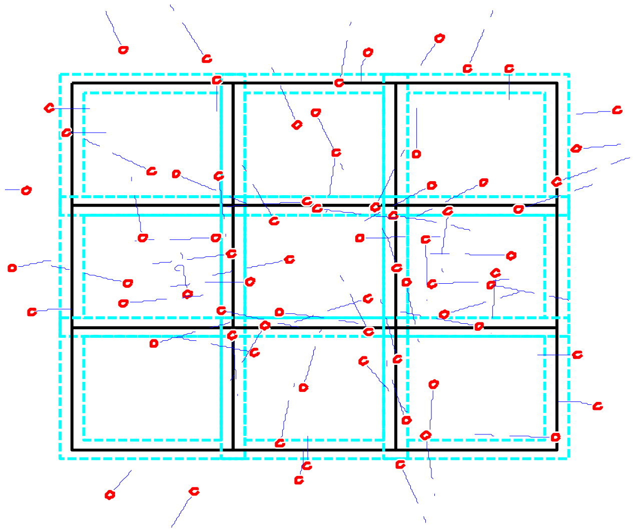


Figure 5. A snapshot of the control area during program run

A key criteria used for evaluation of method's competency is the ratio of travelling time of each agent without having any time slot in the control area to its entire flight time. Figure 6 compares the results for different populations. As it can be seen in this figure, when the network is more populated the competition for getting time slots in all sectors becomes tighter. Accordingly, more agents might travel without having a time slot for re-planning in a sector. For example, when four agents are added to the network every 10 seconds, just 35% of agents always have time slots whiles in the case of adding four agents per 50 seconds almost all agents are successful in getting time slot prior the entering the into a new sector. Thus, limiting the incoming traffic flow is one option for enhancing the chance of agents in getting the time slot.

Another theoretical suggestion for improving this performance metric is increasing the number of slots which could be achieved by decreasing the length of each time slot. So, in the same time interval, more slots will be available. Outcomes of enhancing the resources which are existing time slots have been demonstrated by figure 7. As it was expected, travelling time without having slot has been reduced in all cases. However, the length of each time slot cannot be shorter than the required time for optimization and sharing the new plan with the other agents.

Table 2 summarizes the overall travelling time without having slot per entire travelling time of the whole population in different densities and different time slot lengths.

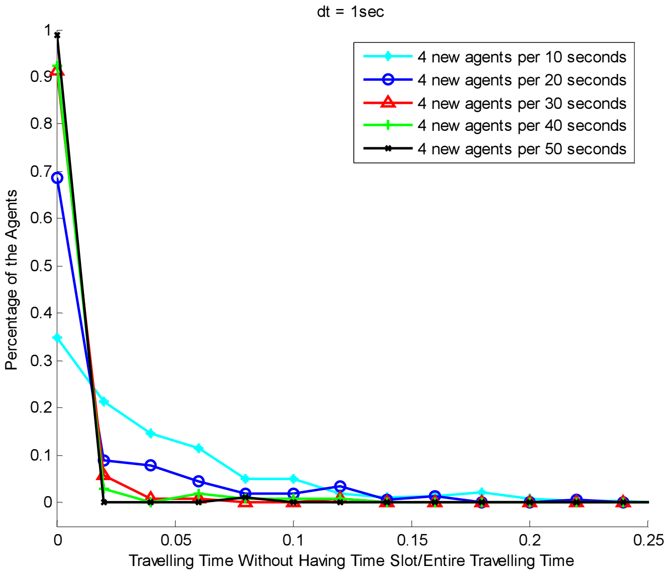


Figure 6.

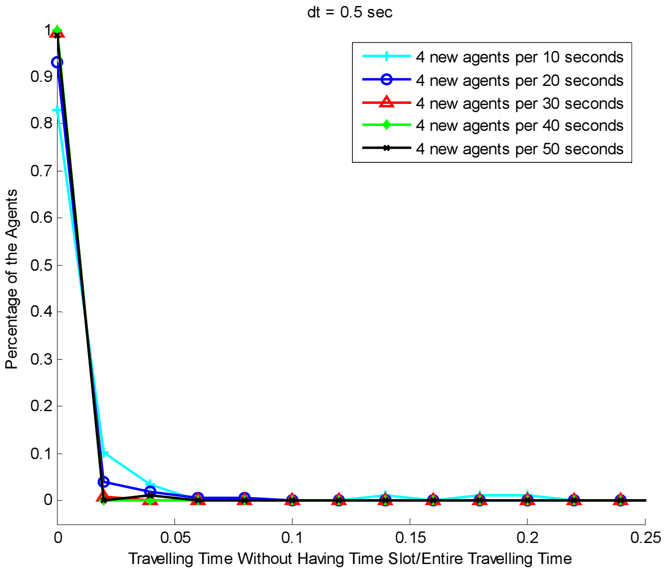


Figure 7.

Table 2. Comparison of overall travelling time without having slot per entire travelling time of the whole population in different densities

	dt = 1	dt = 0.5
4 new agents per 10 sec	% 4.38	% 1.52
4 new agents per 20 sec	% 2.76	% 0.44
4 new agents per 30 sec	% 0.74	% 0.10
4 new agents per 40 sec	% 0.71	% 0.04
4 new agents per 50 sec	% 0.23	% 0.07

6 CONCLUSION

Ad hoc distributed control scheme for air traffic management has been proposed in this paper. Specifying the times of optimization are left to each aircraft. The origin of establishing ad hoc DMPC idea was from network systems. The suggested algorithm is based on Space-TDMA and Self-Organized TDMA. To resolve planning collision, where two agents plan at the same time, the specific procedure based on CSMA/CD is followed.

Applying the proposed algorithm in some air traffic scenarios shows that it works better in low traffic densities. However, there is still some aircraft who might fly without an updated plan for a portion of their flights. These results show that in the most populated situation travelling time without having a slot for 30% of aircraft is more than 5% of their entire flight time. This situation could be improved by decreasing the network population density or/and decreasing length of time slots. As a result, about 90% of the agents always have time slot.

Future work includes the finding an alternative safe solution in the case of travelling without slot. It would be of interest to explore adaptive ad hoc distributed control in which the tuning parameters are investigated by the agents to find the best time slots for acting in the network. Also, slot allocation process could be coupled with trajectory optimization problem to have more achievements by optimizing at the certain times.

7 REFERENCES

Amouris, Konstantinos. 2001. “Space-Time Division Multiple Access (STDMA)* and Coordinated, Power-Aware MACA for Mobile Ad-Hoc Networks.” *In IEEE Symposium on Ad Hoc Wireless Networks (SAWN2001).* San Antonio, Texas.

Bertsimas, Dimitri, and Amedeo Odoni. 1997. *A critical survey of optimization models for tactical and strategic aspects of air traffic flow management.* Technical Report, NASA Ames Research Center.

Bilstrup, Katrin, Elisabeth Uhelmann, Erik G. Strom, and Urban Bilstrup. 2009. “On the Ability of the IEEE 802.11p and STDMA to Provide Predictable Channel Access.” *16th World Congress on Intelligent Transport Systems (ITS).* Stockholm, Sweden.

Borrelli, Franco, and Tamas Keviczky. 2006. “Distributed LQR Design for Dynamically Decoupled Systems.” *Proceedings of the 45th IEEE Conference on Decision & Control.* San Diego, CA, USA.

Chaloulos, Georgios, Peter Hokayem, and John Lygeros. June 2010. “Distributed Hierarchical MPC for Conflict Resolution in Air Traffic Control.” *American Control Conference.* Baltimore, MD, USA.

Dunbar, William B. 2007. “Distributed Receding Horizon Control of Cost Coupled Systems.” *46th IEEE Conference on Decision and Control.* New Orleans, LA, USA.

Dunbar, William B. 2007. “Distributed Receding Horizon Control of Dynamically Coupled Nonlinear Systems.” *IEEE Transactions on Automatic Control* 52: 1249-1263.

Dunbar, William B., and Richard M. Murray. 2006. “Distributed Receding Horizon Control for Multi-Vehicle Formation Stabilization.” *Automatica* 42: 549-558.

Gaugel, T., J. Mittag, H. Hartenstein, and S. Papanastasiou. 2013. “In-depth analysis and evaluation of Self-organizing TDMA.” *Vehicular Networking Conference (VNC), 2013 IEEE.* Boston, MA: IEEE. 79-86.

Gustavsson, N. 1996. “VDL Mode 4/STDMA-a CNS data link.” *Digital Avionics Systems Conference, 15th AIAA/IEEE.* Atlanta, GA: IEEE. 111-116.

Jia, Dong, and Bruce H. Krogh. 2001. “Distributed Model Predictive Control.” *Proceedings of American Control Conference.* Arlington, VA.

Keiser, G.E. 1989. *Local Area Networks.* McGraw-Hill.

Keviczky, Tamas, Fracesco Borrelli, and Gary Balas. 2004b. “Hierarchical Design of Decentralized Receding Horizon Controllers for Decoupled Systems.” *43rd IEEE Conference on Decision and Control.* Atlantis, Paradise Island, Bahamas.

Keviczky, Tamas, Franco Borrelli, and Gary Balas. 2004a. “A Study on Decentralized Receding Horizon Control for Decoupled Systems.” *American Control Conference.* Boston, MA.

Keviczky, Tamas, Franco Borrelli, and Gary Balas. 2006. “Decentralized Receding Horizon Control for Large Scale Dynamically Decoupled Systems.” *Automatica* 42: 2105-2115.

Kumar, Sunil, Vineet Raghavan, and Jing Deng. 2004. “Medium Access Control protocols for ad hoc wireless networks: A survay.” *Ad Hoc Networks*, October.

Kuwata, Yoshiaki, and Jonathan How. 2006. “Decentralized Cooperative Trajectory Optimization for UAVs with Coupling Constrainst.” *45th IEEE Conference on Decision & Control.* San Diego, CA, USA.

Kuwata, Yoshiaki, Arthur Richards, Tom Schouwenaars, and Jonathan How. 2007. “Distributed Robust Receding Horizon Control for Multivehicle Guidance.” *IEEE Transaction on control Systems Technology* 15 (4).

Maestre, José M, and Rudy R Negenborn. 2014. *Distributed Model Predictive Control Made Easy.* Springer Netherlands.

Metzger , Ulla, and Raja Parasuraman. 2001. “The Role of the Air Traffic Controller in Future Air Traffic Management: An Empirical Study of Active Control versus Passive Monitoring.” *HUMAN FACTORS* Vol. 43, No. 4.

Pahlavan, Kaveh, and Allen H. Levesque. 2005. *Wireless Information Networks.* Hoboken, New Jersey: John Wiley & Sons.

Patel, Rushen B., and Paul J. Goulart. 2011. “Trajectory Generation for Aircraft Avoidance Maneuvers Using Online Optimization.” *JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS* Vol. 34, No. 1,.

Richards, Arthur, and Jonathan How. 2004a. “A Decentralized Algorithm for Robust Constrained Model Predictive Control.” *Proceeding of the 2004 American Control Conference.* Boston, Massachusetts.

Richards, Arthur, and Jonathan How. 2004b. “Decentralized Model Predictive Control of Cooperating UAVs.” *43rd IEEE Conference on Decision and Control.* Atlantis, Paradise Island, Bahamas.

Rom, Raphael, and Moshe Sidi. 1989. *Multiple Access Protocols, Performance and Analysis.* Haifa, Israel: Springer-Verlag.

Scattolini, R. 2009. “Architecture for Distributed and Hierarchical Model Predictive Control- a riview.” (Journal of Process Control) 19.

Sridhar, Banavar, Shon Grabbe, and Avijit Mukherjee. 2008. “Modeling and Optimization in Traffic Flow Management.” *Proc. IEEE* 96 (12): 2060-2080.

Trodden, Paul, and Arthur Richards. 2006. “Robust Distributed Model Predictive Control Using Tubes.” Minnesota, USA: American Control Conference.