2021 IEEE Conference on Antenna Measurements & Applications (CAMA), 15-17 November 2021, Antibes Juan-les-Pins, France
DOI: 10.1109/CAMA49227.2021.9703640

Reinforcement Learning System of UAV for Antenna Beam Localization

1st Saki Omi
School of Aerospace, Transport and Manufacturing
Cranfield University
Bedford, UK
s.omi@cranfield.ac.uk

2nd Hyo-Sang Shin
School of Aerospace, Transport and Manufacturing
Cranfield University
Bedford, UK
h.shin@cranfield.ac.uk

3rd Antonios Tsourdos
School of Aerospace, Transport and Manufacturing
Cranfield University
Bedford, UK
a.tsourdos@cranfield.ac.uk

4th Joakim Espeland
QuadSAT
Odense, Denmark
je@quadsat.com

5th Andrian Buchi
QuadSAT
Odense, Denmark
ab@quadsat.com

***Abstract*—Along with the growth of satellite communication industry, the demands and benefits to perform satellite terminal antenna evaluation are increasing. UAV based in-situ measurement can increase the efficiency of the measurement procedure. Main beam localization is a necessary procedure to execute the antenna evaluation test. To accelerate the process of finding the antenna beam centre, this paper develop a meta-reinforcement learning based algorithm. The developed algorithm is compared with other methods and it showed the best performance in terms of accuracy, robustness and travelling efficiency not only in the simulated radiation pattern environment but also in the empirically obtained radiation pattern.**

***Index Terms*—antenna measurement, UAV measurement, beam localization, meta-reinforcement learning**

I. Introduction

The satellite communication industry and technology are expanding, and the number of satellites in the orbit is exponentially growing. The satellite in the operation is currently around 8000 and OneWeb and SpaceX are planning to launch more than 40000 satellites in the coming years [1]. As the number of the communication increases, the number of the ground terminal antenna in communication is growing. When the communication environment is busy between the satellite and ground terminal antennas, the issue that emerges more frequently would be interference due to the misalignment or undesired antenna pattern which might not meet the requirement from organizations such as ITU and SOMAP [2], [3]. Therefore, the importance of the satellite terminal antenna measurement is less negligible in this context. The far field range is a function of the frequency and antenna aperture size. In case of the high operation frequency and the large terminal antenna size, the antenna measurement may require long distance to test the antenna in far field range and it sometime goes around a few kilometres. The available test facilities which can afford this long distance are limited and it is generally expensive solutions in terms of time and cost. In addition, antenna under the test (AUT) needs to be shipped to the place where the facility is located. To overcome these challenges and to accelerate the antenna measurement, QuadSAT has developed UAV antenna measurement solution. This solution could allow the measurement procedure much faster, and it is not limited in the location where the facility is since it can be delivered to any site. In this work, one of the developments to extend its function is presented.

From the perspective of the antenna measurement, knowledge about the location of main beam centre is useful. For instance, before starting radiation pattern measurement there is a need to fix the coordinate system around the boresight. Also, when the electronically steerable antenna is evaluated, it might be beneficial to measure the change of the gain of the main beam as its steering angle varies. Hence, it is certain that speeding up the process of beam localization is important.

Machine learning is a promising technique because of the improvement computer technology. One of the benefits of machine learning technique is without knowing a model of the event, it creates the model and apply this approximation function to estimate inexperienced cases. Reinforcement learning (RL) is one of the branches of machine learning technique. In reinforcement learning, the agent interacts with the environment where the agent is and takes the best action in the given state. It has been applied for playing games [4], generating locomotion of robots [5] and controlling a vehicle [6]. In the conventional RL, the environment is fixed and hence the agent learns how to play in this certain environment. However recently RL for more general environment has been researched and this is called meta-reinforcement learning. One of the typical examples which could be similar to beam localization can be found as "MountianCarContinuous-v0" from OpenAI [7]. The similarity between this example and beam localization is that the goal of the agents is reaching a certain point of interest. The limitation of this implementation is that it does not take the measurement noise into account and it lacks the generality since the scenario where the car learns to clime the same shape of the hill from the same starting point. In case of the beam localization discussed in this work, the shape of the radiation pattern and the relative position to the goal which is the beam centre is not fixed. In the situation where the observation provides only partial information regarding

© 2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

the state of the agent, RL algorithm needs to solve Partially Observable Markov Decision Process (POMDP). One of the approaches which could cope with POMDP is introducing recurrent neural network such as Long Short-Term Memory (LSTM) [8] and Gated Recurrent Unit (GRU) [9]. Also, to bring the RL toward more general purpose, meta-reinforcement learning has been proposed [10], [11]. This technique could be related to neuroscience and psychology [12].

For UAV measurement, beam location can be estimated by taking single or multiple raster scans. However, this method is time consuming and if the main beam is outside of the scanned range, it cannot localize the beam from the measured information. Other considered approach is cross-section method. In this method, the signal strength on a line is repeatedly measured and the beam centre is estimated by using the symmetricity of the pattern. This approach has also long flight distance and also dependency on the symmetricity and may not provide accurate estimation. Gradient ascent method could decrease the traveling distance, but it may have a difficulty in parameter tuning. To develop a method which does not require parameter tuning and offers efficiency and enough accuracy, RL learning based approach is developed and discussed in this paper.

II. Background

A. Concept of Reinforcement Learning

The basic concept of RL can be found in the inner loop of Fig. 1. In RL, the agent interacts with the environment and learns to make the optimal decision on action A_t in the given state S_t in the environment. The agent takes observation O_t which contains the information of new state S_{t+1} after taking an action A_t. Also, the agent receives a feedback signal as reward R_t from the environment which is the evaluation of the taken action A_t in the state S_t. One cycle of the experience is normally named as one episode and the end of one episode is determined by termination condition. The purpose of the learning for the agent is to maximize the cumulative reward from the current time step to the end of the episode. The original RL was structured based on the assumption of Markov Decision Processes (MDPs). In MDP environment, the probability distribution of the response from the environment is only depending on the current sates and not on the entire history of the state and action (1).

$$\begin{aligned} P_t\{R_{t+1} = r, S_{t+1} = s' | S_0, A_0, R_1, \ldots, S_t, A_t\} \\ = P_r\{R_{t+1} = r, S_{t+1} = s' | S_t, A_t\} \end{aligned} \quad (1)$$

However, in many scenarios, observation does not directly represent state ($O_t \neq S_t$). In this case, RL needs to learn solving Partially Observable Markov Decision Process (POMDP) and then the agent learns the interpretation between observation and state.

B. Twin Delayed Deep Deterministic Policy Gradient Algorithm

In RL, the agent could include the following components: policy, value function and model. Policy decides the action of the agent and value function is a function to evaluate the state and/or action. Model is the representation of the environment for agent. Action value function is normally described as $Q(s, a)$ which is the evaluation of the current state and action. In the deep RL, this value is approximated by neural network with the parameter θ. When the action has continuous action field, there are infinite number of possible actions. In this case, the policy is also approximated with neural network with the parameter of ϕ in case of deterministic policy [13]. During the training process, ϕ is trained by the deterministic policy gradient and θ is also learned. Twin Delayed Deep Deterministic policy gradient algorithm (TD3) is one of the latest RL algorithm which shows eminent stability in performance [5] and it is implemented in this development.

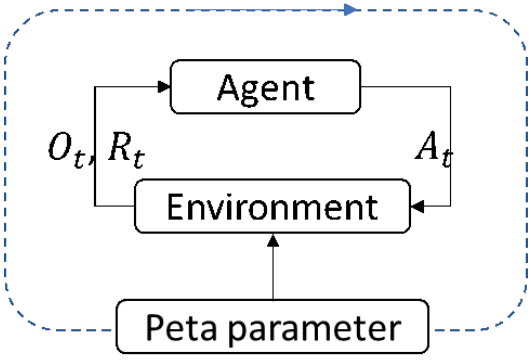


Fig. 1. Meta-reinforcement learning

C. Recurrent Neural Network

In the condition of POMDP, the action value function is $Q(o_t, a_t|\theta) \neq Q(s_t, a_t|\theta)$ due to the partial observability to the true state. Therefore, the agent needs to find the representation of the own state based on the observation. Recurrent Neural Network (RNN) is one of the effective approaches for POMDP condition [14]. GRU and LSTM are the most popular RNN. In addition to output value, RNN generates hidden state h_t which contains the sequential information from the previous time steps to be carried in the next time step.

D. Meta-Reinforcement Learning

In general, a MDP is determined by a 4-main tuple, $\mathcal{M}_i = [\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}]$ where $\mathcal{S}$ is a set of states, $\mathcal{A}$ is a set of actions, $\mathcal{P}$ is a transition probability function $\mathcal{P}_{ss'} = \mathbb{P}[S_{t+1} = s' | S_t = s]$ and $\mathcal{R}$ is a reward function $\mathcal{R}_s = \mathbb{E}[R_{t+1} | S_t = s]$. The agent learns their policy π_θ for the environment which is governed by $\mathcal{M}_i$. In meta-reinforcement learning, the agent experiences multiple MDPs to perform toward more general purpose. As proposed in [10], [11], during the training procedure, new $\mathcal{M}_i \sim \mathcal{M}$ is sampled. Also, LSTM is implemented to policy which can help tracking the characteristics of the trajectories changed by the outer loop.

III. Method

A. Definition of POMDP Elements

When the position of the agent with respect to the beam centre is unknown, the representation of the observation of the agent cannot rely on it. Alternatively, global coordinate

system, which is common in any environment, can be predefined and the state is described on coordinate system. Then, observation definition would be the position of the agent and the measured signal strength ($o_t = [x, y, P_r]$). However, because the environment is not fixed and various MDPs exist, the interpretation between this state definition and the observation cannot be fixed. Also, the coordinate system could be defined by taking the origin at the initial position of the agent. The drawback of this definition would be when the distance between the initial position of the agent and the goal position is unexpectedly large in implementation. In this case, since the agent will encounter unexperienced observation, it would not be able to make the right decision. Therefore, the observation representation in this work is described as the displacement of the position in 2 dimension and variance of the measured signal strength from the previous time step ($o_t = [\Delta x, \Delta y, \Delta P_r]$). In this way, the state representation and interpretation between the state and the observation is generalized. Then, when the observation and hidden stated from the previous time step are given to the actor, the actor generates the displacement as an action for the next time step to the agent.

$$a = \mu(o_t, h_{t-1}) \tag{2}$$

Also, the reward is defined as

$$r = \Delta P_r - 0.1 \tag{3}$$

where ΔP_r is the variance of the received power from the previous time step in order to encourage the agent to move to position expecting higher gain position. Also, -0.1 in (4) works as a punishment as it receives negative reward every time it steps. This element is to terminate the episode as small steps as possible. Finally, the termination condition is turned to true if the number of the steps reaches to the maximum allowed steps or the agent reaches to the beam centre with certain accuracy.

B. Meta Data to Generate the Environment

In meta-reinforcement learning, various environments of MDPs are generated at outer loop (Fig. 1). The radiation patterns are created based on theoretical models to collect the training data-set. The given meta parameters to change are 3dB beam width, maximum possible signal to noise ratio (SNR), relative position between initial position of the agent and the beam center. By randomly changing these parameters, infinite number of MDPs are produced so that the trained system can be guided toward the goal position in any environments.

C. Network structure

For the sake of POMDP and meta-reinforcement learning, implementing RNN in the neural network could be effective [6], [15]. Several approaches have been developed to implement RNN in TD3 algorithm [14]. To avoid training on full episode, hidden states are directory stored in the replay buffer and used in the training in this work. Fig. 2 shows the comparison between original TD3 neural network and the developed one.

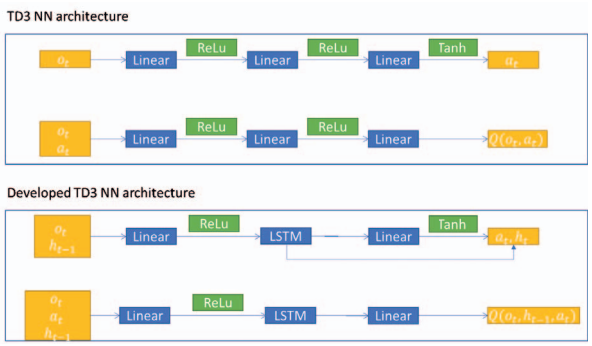


Fig. 2. Neural network structure

D. Assumption

The important assumption to be noted is that this development is focused on the case without local maxima. This is because when the location of the main beam is approximately known and the area of interest is narrow, it is hardly expected that there are local maxima. In the scenario where there is no prior information or where local maxima is considered, additional development could be required.

IV. Experiment

A. Training

During the training, the meta parameters (i.e., 3dB beam width maximum possible SNR, and initial position to the beam centre) are randomly generated in every episode. The range of the parameters can be found in Table I. The antenna patterns are simulated based on TE11 circular source model and perfectly symmetrical. One agent is randomly placed with respect to the centre of the main beam within the range of ϕ, θ in Table I on the spherical coordinate system. The agent makes a measurement at the current location. Based on the observation as it is defined in the previous section, the system is trained to go close to the centre of the main beam. To confirm the performance of RNN, both networks with and without RNN are tested. The termination accuracy is set to 0.05 in x and y direction where $x = \theta \cos\phi$ and $y = \theta \cos\phi$.

TABLE I
META PARAMETER RANGE

	Min	Max
Beamwidth [deg]	3	6
SNR [dB]	40	60
Initial relative position in spherical angle [ϕ,θ]	$[0, 0]$	$[360, 3]$

B. Evaluation

In the evaluation stage, firstly the performance of the system is evaluated in the environment based on Table I with randomly generated theoretical radiation pattern. Secondly, the trained system is tested on the radiation pattern based on the real measurement. The latter radiation pattern has never been used during the training and this pattern is not perfectly symmetrical.

C. Sampling based gradient ascent method

In addition to reinforcement learning based approach, sampling based gradient ascent approach is applied to compare the performance. In this method, the peak of the beam is estimated based on

$$w = w + \alpha \frac{\partial P_r(\psi)}{\partial \psi_i} \tag{4}$$

, where partial derivative in this example is obtained by sampling at $P_r(\psi + \delta\psi_i)$, e.g.,

$$\begin{aligned} \frac{\partial P_r(x)}{\partial x} &= \frac{P_r(x+\delta x) - P_r(x)}{\delta x} \\ \frac{\partial P_r(y)}{\partial y} &= \frac{P_r(y+\delta y) - P_r(y)}{\delta y} \end{aligned} \tag{5}$$

with the parameters of $\delta x = \delta y = d$.

V. Result

Table II shows the averaged result of the trained system (TD3-RNN and TD3 represents the network architecture with and without RNN respectively) and sampling gradient ascent method with the parameter of d after 1000 times trial. In terms of accuracy, trained algorithm showed the best root mean square error (RMSE) value with the best robustness observed from the standard deviation (Std). Also, it can be found that the performance of sampling gradient ascent approach has high dependency on the parameter. Success rate describes the number of the trial which reached the estimation accuracy less than 0.05. Displacement in success is the total displacement distance required to reach the final position from the trials in success. The RL based algorithm showed much more efficient in term of the displacement which represent the traveling distance. Fig. 3 shows one of the example trajectories from the trained algorithm with RNN. To have a further analysis Fig. 4 and Fig. 5 are illustrated. Orange dots show the successful trial and blue dots show unsuccessful ones. "DbyL" is a parameter to control the beam width of the AUT. When this value is larger the beam becomes sharper. From these figures, when the SNR goes smaller and the beam becomes fatter, it becomes more difficult to perform accurately, and this analysis shows compliance with physical interpretation. The success rates of the trained algorithm with and without RNN within SNR $40 - 42$ dB are 0.78 and 0.67 respectively. This could be explained as the effect of introducing RNN in the network. Finally, Table III shows the performance in the empirical radiation pattern. The RL based algorithm remained to perform the best although the trained systems have never experienced the environment. The performance of the gradient ascent algorithm varied noticeably. The algorithm with $d = 0.01$ showed the high success rate which had the worst performance in the theoretical radiation pattern environment.

VI. Conclusion

The reinforcement learning based antenna beam localization algorithm was proposed and evaluated. The meta-reinforcement learning algorithm with RNN is compared with other methods and showed considerable performance in terms of accuracy, robustness, and efficiency in displacement not only in the simulated environment but also in the empirical radiation pattern environment. This application could be

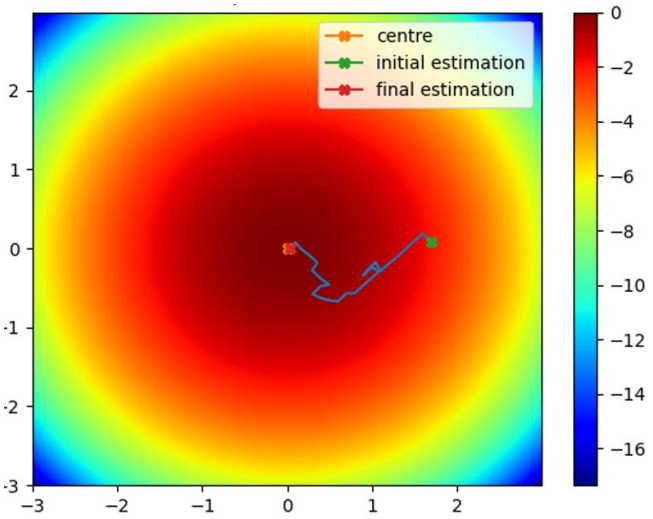


Fig. 3. Trajectory example from TD3-RNN

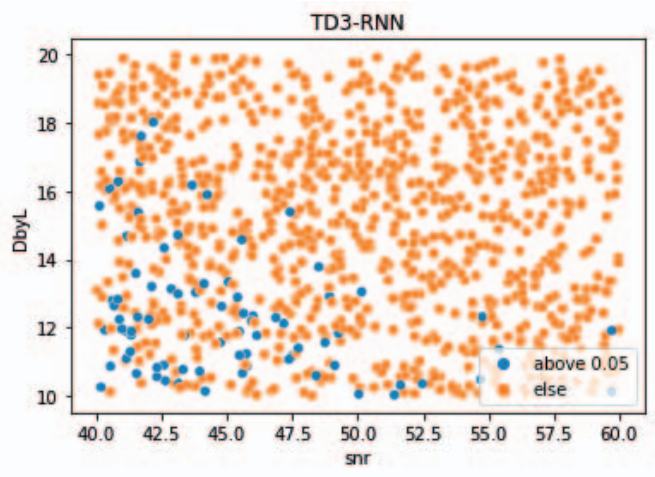


Fig. 4. Performance analysis of TD3-RNN

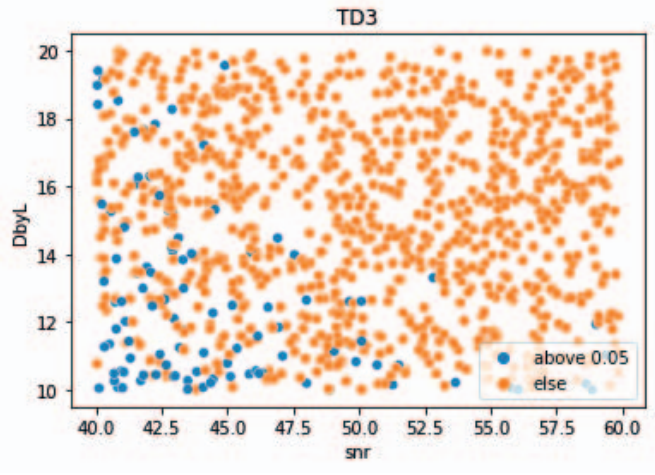


Fig. 5. Performance analysis of TD3

TABLE II
EVALUATION RESULT WITH THEORETICAL RADIATION PATTERN

	RMSE	Std	Success rate	Displacement in success
TD3-RNN	0.056	0.091	0.921	2.812
TD3	0.062	0.101	0.895	2.964
GA d=0.05	0.450	8.027	0.858	7.101
GA d=0.01	281.83	487.80	0.274	9.317

TABLE III
EVALUATION RESULT WITH THEORETICAL RADIATION PATTERN

	RMSE	Std	Success rate	Displacement in success
TD3-RNN	0.059	0.272	0.982	2.817
TD3	0.145	0.541	0.872	2.861
GA d=0.05	0.439	0.670	0.274	4.300
GA d=0.01	0.236	0.810	0.930	3.540

extended to multiple beam localization where local maxima exist.

ACKNOWLEDGMENT

This work is partially funded by QuadSAT. We would like to thank them for their support.

REFERENCES

[1] S. Chakrabarti, "How many satellites are orbiting Earth? — Space." https://www.space.com/how-many-satellites-are-orbiting-earth. (accessed: 10.12.2021).

[2] ITU, "Maximum permissible level of off-axis e.i.r.p. density from very small aperture terminals (vsats)," Tech. Rep. ITU-R S.728-1, International Telecommunication Union, 2002.

[3] G. F. O. R. Type, O. N. The, and M. Mobile, "Global VSAT Forum PERFORMANCE AND TEST GUIDELINES FOR TYPE APPROVAL OF ' COMMS ON THE MOVE ' MOBILE SATELLITE GVF-105 COMMUNICATIONS," pp. 1–30.

[4] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[5] S. Fujimoto, H. Van Hoof, and D. Meger, "Addressing Function Approximation Error in Actor-Critic Methods," *35th International Conference on Machine Learning, ICML 2018*, vol. 4, pp. 2587–2601, 2018.

[6] B. Gaudet and R. Linares, "Adaptive guidance with reinforcement meta learning," *Advances in the Astronautical Sciences*, vol. 168, pp. 4091–4109, 2019.

[7] OpenAI, "MountainCarContinuous-v0." https://www.space.com/how-many-satellites-are-orbiting-earth. (accessed: 10.12.2021).

[8] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, pp. 1735–1780, 11 1997.

[9] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Gated feedback recurrent neural networks," *32nd International Conference on Machine Learning, ICML 2015*, vol. 3, pp. 2067–2075, 2015.

[10] J. X. Wang, Z. Kurth-Nelson, D. Tirumala, H. Soyer, J. Z. Leibo, R. Munos, C. Blundell, D. Kumaran, and M. Botvinick, "Learning to reinforcement learn," 11 2016.

[11] Y. Duan, J. Schulman, X. Chen, P. L. Bartlett, I. Sutskever, and P. Abbeel, "RL$ˆ2$: Fast Reinforcement Learning via Slow Reinforcement Learning," 11 2016.

[12] M. Botvinick, S. Ritter, J. X. Wang, Z. Kurth-Nelson, C. Blundell, and D. Hassabis, "Reinforcement Learning, Fast and Slow," 5 2019.

[13] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," *31st International Conference on Machine Learning, ICML 2014*, vol. 1, pp. 605–619, 2014.

[14] L. Meng, R. Gorbet, and D. Kulić, "Memory-based Deep Reinforcement Learning for POMDP," 2 2021.

[15] J. N. Foerster, G. Farquhar, T. Afouras, N. Nardelli, and S. Whiteson, "Counterfactual multi-agent policy gradients," *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, pp. 2974–2982, 2018.



https://dspace.org/

Reinforcement learning system of UAV for antenna beam localization

Omi, Saki; Shin, Hyo-Sang; Tsourdos, Antonios; Espeland, Joakim; Buchi, Andrian

2022-02-11
Institute of Electrical and Electronics Engineers (IEEE)
https://doi.org/10.1109/CAMA49227.2021.9703640
Omi S, Hyo-Sang S, Tsourdos A, et al., (2022) Reinforcement learning system of UAV for antenna beam localization. In: 2021 IEEE Conference on Antenna Measurements & Applications (CAMA), 15-17 November 2021, Antibes Juan-les-Pins, France, pp. 61-65

Downloaded from a DSpace repository. https://dspace.org/