Towards Automatic Nanomanipulation: Drift Compensation in Scanning Probe Microscopes

B. Mokaberi and A. A. G. Requicha
Laboratory for Molecular Robotics
University of Southern California
Los Angeles, CA
{mokaberi, requicha}@usc.edu

***Abstract*—Manipulation of nanoparticles with Atomic Force Microscopes (AFMs) has been under development for a decade, and is now well established as a technique for prototyping nanodevices and for other applications. The manipulation process tends to be labor-intensive because a user is needed in the loop to compensate for the numerous uncertainties associated with AFM operation. This paper addresses thermal drift, which is the major cause of errors for AFMs operated in ambient conditions. It is shown that drift can be estimated efficiently by using Kalman filtering techniques. Preliminary results indicate that drift compensation enables manipulation of groups of particles under program control, without human intervention, in ambient air and at room temperature. This is a first step towards fully automatic nanomanipulation, which would permit assembling, from the bottom up, nanostructures much more complex than those being built today with AFMs.**

Keywords-nanomanipulation; nanoassembly; Atomic Force Microscopes; Scanning Probe Microscopes; spatial uncertainty; Kalman filtering; nanorobotics

I. INTRODUCTION

Atom manipulation with Scanning Probe Microscopes (SPMs) was first demonstrated in the early 1990s, and manipulation of particles with sizes on the order of a few nm to the tens of nm soon followed—see [1, 2] and references therein. Today, particles with diameters of ~ 10 nm are manipulated routinely with Atomic Force Microscopes (AFMs) at USC's Laboratory for Molecular Robotics (LMR) and elsewhere. (The AFM is a specific type of SPM that exploits interatomic forces between a sharp tip and a sample.) Nanomanipulation operations are used to prototype nanoscale devices, and to repair or modify nanostructures built by other means.

Manipulation of small nanoparticles (with diameters below ~ 30 nm, say) in ambient conditions, i.e., at room temperature, in air or in a liquid, and without stringent environmental controls, requires extensive user intervention to compensate for the many spatial uncertainties associated with AFMs and their piezoelectric drive mechanisms. Uncertainties are introduced by phenomena that range from non-linearities in the voltage-displacement curves that characterize the piezos, to creep, hysteresis, and thermal drift. The latter is the major cause of spatial uncertainty in our lab, and is due primarily to thermal expansion and contraction of the AFM components. For example, a one degree change in temperature will cause a 50 nm change in the length of a mechanical part that is 5 mm long (assuming a typical expansion coefficient of 10^{-5} /°C).

AFM vendor software usually compensates for piezo non-linearities. Hysteresis effects can be greatly reduced by scanning always in the same direction, and creep effects can be nearly eliminated by waiting a few minutes after each large motion of the scanner (although this is very inefficient). In addition, modern, top-of-the-line instruments are equipped with feedback loops that claim positioning errors below 1 nm in the x, y plane of the sample [3]. Feedback can compensate for non-linearities, hysteresis and creep, but not tip drift—see Section II.B below. Drift tends to increase with time, which implies that complicated assemblies cannot be completed without frequent user interaction. Moreover, most of the AFMs in use today either have no x, y feedback, or their feedback loops have noise levels of several nm, which are too large for manipulation of particles with sizes ~ 10 nm. These machines are normally operated open loop for the small scan sizes (< 1 μm) used in the manipulation of small nanoparticles, to avoid introducing additional noise through the feedback circuitry. With or without x, y feedback, drift compensation remains a crucial issue for successful nanomanipulation.

Typical assemblies of small nanoparticles built by nanomanipulation today consist of ten to twenty particles, and may take an experienced user a whole day to construct. To move towards more complex assemblies requires that the manipulation process become more automated, and this in turn requires compensation of the spatial uncertainties associated with AFMs—especially drift, which is the most pernicious one, as we have argued above.

The remainder of this paper discusses the characteristics of drift, how to estimate it and compensate for it, our implementation of drift compensation, and experimental results.

II. DRIFT CHARACTERIZATION

A. The Problem

Successive AFM scans of a sample without changing any of the scanning parameters will appear as translated versions of

The research reported here was supported in part by the NSF under grants EIA-98-71775 and DMI-02-09678.

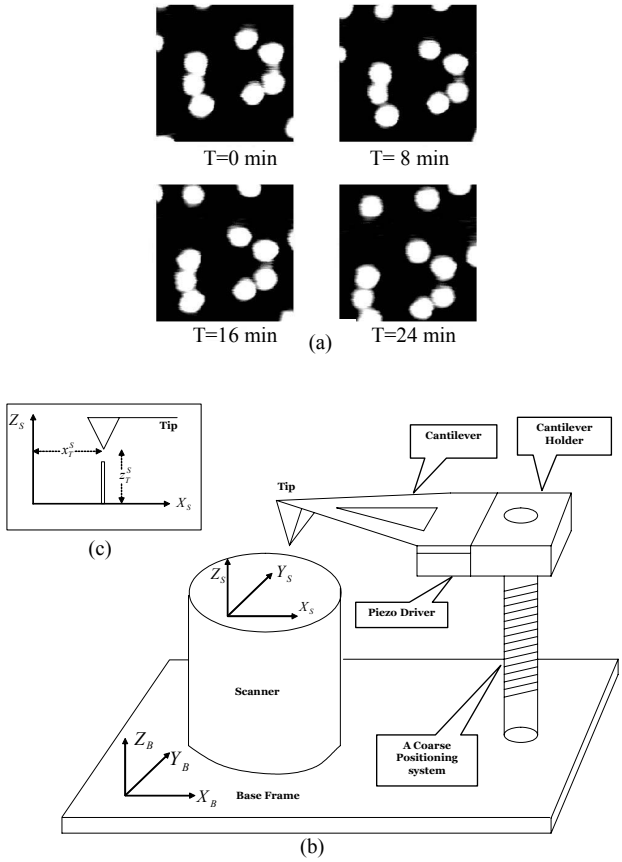


Figure 1. (a) Four images from a 180 nm region of a sample, taken at 8 min intervals. The objects shown are Au nanoparticles with 15 nm diameters. (b) A schematic diagram of an AFM machine. (c) A spatial asperity below the tip.

the sample surface, as shown in Fig. 1.a. This is the physical manifestation of drift in the *x*, *y* plane of the sample. There is also drift in the *z* direction, but it has little impact on nanomanipulation and will be ignored in this paper. (Note that, unlike atoms, particles of the sizes we are discussing in this paper do not diffuse over the surface at room temperature; they are fixed with respect to the surface.) Many experimental observations indicate that the drift is essentially a translation (no rotation is involved) and the drift velocity is approximately constant over periods of several minutes, but changes on a longer time scale. A drift-compensated instrument would produce the same image in each of the panels of Fig. 1.a. The problem addressed in this paper is how to achieve drift compensation in such a way that not only images of the same region are constant in time, but other processes—especially programmed sequences of manipulation operations—also can be carried out as if drift did not exist.

Several authors have reported simple approaches to drift compensation [4, 5, 6, 7, 8]. These approaches assume that the drift velocity is constant and compute it by comparing successive images. The major drawback of these procedures is their failure to adapt when the drift velocity changes. They can be used for correcting images taken over a period of time of approximately constant drift, but cannot support a sequence of nanomanipulation operations, which requires real-time compensation over relatively long durations.

B. Spatial Analysis

Consider a typical AFM, schematically shown in Fig. 1.b. For concreteness we assume that the sample is placed on top of the scanner and that the unloaded cantilever and tip (or their average positions when the AFM is operated in dynamic mode) are fixed in space, except for drift. (This is the most common AFM configuration; other configurations in which the sample is fixed and the cantilever moves can be analyzed by trivial changes to the arguments below.) Consider also a sample which consists of a flat surface with a very sharp asperity on it (a spatial impulse), as shown in Fig. 1.c. (For simplicity we ignore the *y* coordinate in the following discussion and the figure.) In an ideal situation, in a contact mode scan the tip remains at a fixed height touching the flat surface until it encounters the asperity. Then the scanner moves very quickly downward (it contracts) for the tip to remain in contact with the feature, maintaining the same applied force. If we record the height of the tip z_T^S in a coordinate system attached rigidly to the sample (or, equivalently, to the top of the scanner) as a function of the x_T^S coordinate of the tip in the same system, this gives us the true topography of the sample, which is the desired output. (A similar argument holds also for vibratory, or dynamic mode operation.)

However, in a real AFM, when the tip contacts the top of a feature such as the spatial impulse of Fig. 1.c, what we measure is the voltage V_z applied to the vertical piezo motor versus the voltage V_x applied to the horizontal motor. If we ignore (or compensate for) hysteresis and creep, and assume that the AFM is properly calibrated and compensates for non-linearities in the voltage-displacement curve, the voltages can be converted into piezo extensions E_z and E_x. But the *x* position of the scanner (or sample) with respect to the base of the instrument does not equal the piezo extension because there is drift between the sample and the base. Rather, the scanner position is $x_S^B = E_x + dx_S^B$, where dx_S^B is the scanner drift. (We assume the drift is a translation, based on experimental observations, as noted earlier.) In other words, even with no applied voltages, the scanner is moving (drifting) with respect to the base. In addition, the tip is itself drifting with respect to the base. Thus the position of the tip with respect to the base is $x_T^B = dx_T^B$. The position of the tip with respect to the sample, which is the desired topography signal, is the difference between the position of the tip and the position of the sample, both measured with respect to the base:

$$x_T^S = x_T^B - x_S^B = (dx_T^B - dx_S^B) - E_x = dx - E_x \qquad (1)$$

The combined effect of these two drifts, dx, is what we call simply AFM drift. To image the impulse of Fig. 1.c the tip must be on top of the feature, and therefore the tip position with respect to the sample must be constant. Since the drift varies with time, it follows from (1) that the piezo extension and corresponding applied voltage vary with time and so do the images. To stabilize the image and compensate for the drift, it suffices to change the origin of the *x* axis by dx. This can be

done by changing the *offset* values associated with a scan, which is precisely what our system does, as we will show later. It is also clear from (1) that the drift between two instants of time t_1 and t_2 can be measured by subtracting the corresponding piezo extensions for an impulse feature. These can be read directly from the images of the feature taken at times t_1 and t_2. (In practice, the situation is more complicated because the features imaged are not pure impulses, as we will see below.)

Feedback in the horizontal directions x, y is normally used in AFMs to ensure that the scanner is in the correct position with respect to the base. Therefore, x, y feedback can compensate (within the noise level constraints of the system) for non-linearities and scanner-base drift. However, it cannot compensate for tip-base drift.

C. Statistical Properties

The behavior of the drift depends on such factors as temperature, humidity, the construction of the instrument, and thermal expansion coefficients. In our lab, drift velocities tend to vary from 0.01 to 0.1 nm/s. Therefore, for 256x256 pixel images taken at a 1 Hz rate (these are typical values) the drift between two successive images can be as much as 25.6 nm, which is larger than the diameter of the particles we normally manipulate.

Fig 2.a shows a time series of drift displacement values in the x and y directions, which was measured by comparing images of the same feature taken at sampling times 35 seconds apart. The corresponding velocities inferred from the figures are approximately constant for several minutes, and then change in a seemingly random manner. The power spectra of the drift time series are shown in Fig 2.b and exhibit a bandwidth on the order of 0.001 Hz, which corresponds to a time constant of 1,000 seconds or about 16 minutes. The slow-varying character of the drift compared to the typical time required for a manipulation operation, which is at most a few seconds, makes it possible to estimate drift while performing a series of manipulations.

III. Estimation and Compensation of Drift

A. Drift Measurement

The basic method for measuring drift consists of comparing successive images of features on the sample's surface. We often manipulate spherical nanoparticles, and use a procedure that exploits knowledge of the object's shape. We search for the center of a spherical particle as follows. First we look for the highest point of a single line scan in the x direction. Then we scan along a single line in the y direction and passing through the previously-found high point. We find the highest point of this y-scan, pass an x-line through it and find a new maximum, continuing the process until we reach a desired accuracy. (This process fails if the first line scan misses the particle altogether.)

A general approach, applicable to objects of arbitrary shapes, involves correlating images, by using the three following steps.

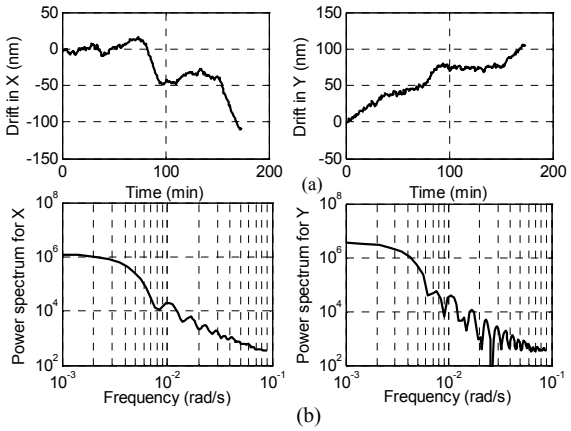


Figure 2. (a) Measurement of drift in x and y directions over 200 minutes, with a sampling time of 35 seconds, (b) corresponding power spectra.

1) Selection of tracking window. We typically use a 64x64 window, which can be scanned in a few seconds and normally contains enough features for successful tracking. The area is selected by maximizing an *interest* operator, defined in terms of the following characteristics.

- Distinctness of features from immediate neighbors.
- Global uniqueness of the features (in the whole image).
- Invariance of features under expected distortions.

All of these criteria can be expressed in terms of metrics defined on the 2-D average squared gradient of the image [9].

2) Coarse computation of translation. The normalized cross-correlation between two images is a good measure of their similarity. It is defined mathematically as:

$$c(dx,dy)=\frac{\sum I_1(x+dx,y+dy)\cdot I_2(x,y)}{\sqrt{\sum I_1^2(x,y)}\cdot\sqrt{\sum I_2^2(x,y)}} \qquad (2)$$

where I_1 and I_2 are the two images, assumed to have zero mean, and the summations are over the discretized values of x and y. If I_2 is a perfect translation of I_1 by (a,b), then the cross-correlation exhibits a peak at $(dx,dy)=(a,b)$, and the peak value is 1. In this case, the rest of the points in $c(dx,dy)$ take values between 0 and 1, depending on how well the two images match at each (dx, dy) translation. In general the match is not perfect, and we use the dx, dy values that correspond to the maximum value of the correlation function as the measured translation, and the peak value as an indication of how well the two images match. The cross-correlation is computed efficiently in the frequency domain, by using the Fast Fourier Transform (FFT) [10].

3) Fine computation of translation. The cross-correlation computation is done with pixel accuracy, which may not be sufficient for succesful nanomanipulation. Sub-pixel accuracies can be obtained by the following procedure [9]. First the cross-correlation method is used to find a coarse value for the translation, (dx_0, dy_0), say. Then $I_2(x, y)$ is expressed as a first-order expansion of I_1:

$$I_2(x,y) = I_1(x + dx_0, y + dy_0) + \frac{\partial I_1}{\partial x} \cdot \Delta dx + \frac{\partial I_1}{\partial y} \cdot \Delta dy + n \qquad (3)$$

where n is a noise term that includes higher-order effects, and Δdx and Δdy are subpixel translations. Next, a best estimate for the deltas is computed by least square estimation over the entire picture. Finally, the refined estimate for the translation is given by $dx = dx_0 + \Delta dx$ and $dy = dy_0 + \Delta dy$.

Observe that both techniques for measuring drift require a coarse estimate of the drift value. Without an approximate position for the spherical particle whose position we want to measure, the search procedure may fail, or produce grossly wrong values, because the first single-line scan may well miss the particle altogether, or, even worse, hit a different particle. The correlation-based technique also may fail if the images in the selected windows are too different. This may produce low correlation values or even spurious peaks that do not correspond to the translation we want to find.

B. Dynamical Model of Drift

We need a dynamical model of the drift to be able to estimate and predict it by using Kalman filtering techniques. Experimental results in our lab indicate that the AFM drifts in both in the x and y directions, but there is a negligible correlation between the two. Therefore we decouple the system and treat the two directions independently.

The drift behavior is very similar to that of a maneuvering target, in which the velocity is approximately constant for a relatively large amount of time, and then changes randomly. A suitable model for such targets was introduced by Singer for radar tracking of manned air vehicles [12]. Singer's model uses an acceleration that is correlated in time. Intuitively, this implies that if a target is accelerating at a time t, it is likely to be still accelerating at a time $t + \tau$, for τ sufficiently small. We model the drift acceleration $a(t)$ by a first-order Markov process governed by the first order differential equation

$$\dot{a}(t) = -\alpha a(t) + w(t) \qquad (4)$$

with a corresponding exponential auto-correlation

$$R(\tau) = E\left[a(t)a(t+\tau)\right] = \sigma_m^2 e^{-\alpha|\tau|} \qquad \alpha > 0 \ . \qquad (5)$$

Here σ_m^2 and $1/\alpha$ are the variance and time constant of acceleration, respectively, and $w(t)$ is white noise with a variance $2\alpha\sigma_m^2$.

The state space formulation for the drift in the x direction is

$$\begin{aligned} \dot{x}(t) &= v(t) \\ \dot{v}(t) &= a(t) \\ \dot{a}(t) &= -\alpha a(t) + w(t) \end{aligned} \qquad (6)$$

where x and v are the drift displacement and velocity, respectively. (Similar equations apply to the y direction in this uncoupled system.) The corresponding discrete-time equations for a sampling period T are

$$\mathbf{x}(k+1) = F\mathbf{x}(k) + \mathbf{u}(k) \qquad (7)$$

where $\mathbf{x}(k) = [x(k)\, v(k)\, a(k)]^T$, $\mathbf{u}(k)$ is a 3x1 process noise, and

$$F = \begin{bmatrix} 1 & T & (\alpha T - 1 + e^{-\alpha T})/\alpha^2 \\ 0 & 1 & (1 - e^{-\alpha T})/\alpha \\ 0 & 0 & e^{-\alpha T} \end{bmatrix}. \qquad (8)$$

In this model, the dynamics of drift can be expressed in terms of three parameters: the variance, or magnitude, of drift acceleration, the time constant, and the sampling interval.

C. Kalman Filter Estimation of Drift

The state of the drift defined in (7) can be estimated recursively, by Kalman filtering. (We consider only the x direction, because the system is decoupled; a similar treatment applies to the y direction.) Given the current state estimate $\hat{\mathbf{x}}(k-1 \mid k-1)$ and the covariance of the error in this estimate $P(k-1 \mid k-1)$, the filter predicts the state and covariance at the next time step by the standard equations

$$\hat{\mathbf{x}}(k \mid k-1) = F\, \hat{\mathbf{x}}(k-1 \mid k-1) \qquad (9)$$

$$P(k \mid k-1) = F\, P(k-1 \mid k-1)\, F^T + Q(k-1) \qquad (10)$$

Here Q is the covariance of the process noise, which can be computed in terms of α and T —see [11, 12] for the actual expressions.

The measurement is modeled in the usual form, as

$$z(k) = H\, \mathbf{x}(k) + v(k) \qquad (11)$$

in which H = [1 0 0] and $v(k)$ is the measurement white noise, independent of the process noise $\mathbf{u}(k)$, and with covariance $R(k) = R_0\delta(k)$.

The drift displacement is measured by the techniques discussed earlier, in section III.A. We use the pre-measurement estimate of the drift $\hat{\mathbf{x}}(k \mid k-1)$ to change the origin of the coordinates when we position the first scan line (for the sphere center search) or the window used for the cross-correlation computation. This change of origin ensures that we don't miss the particle in the first line scan, or that the windows used in the correlation method are not too different. Our measured value is given by

$$z(k) = z(k-1) + dz + \delta \ , \qquad (12)$$

where δ is the distance between the origins of coordinates used to make the two measurements, and is given by

$$\delta = H[\hat{\mathbf{x}}(k \mid k-1) - \hat{\mathbf{x}}(k-1 \mid k-2)] \, . \qquad (13)$$

The Kalman filter computes the Kalman gain and uses the measurement $z(k)$ to update the state estimate $\hat{\mathbf{x}}(k \mid k)$ and the

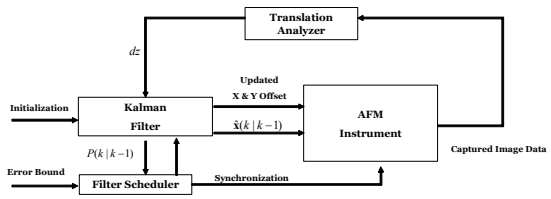


Figure 3. Block diagram of the AFM and the drift compensator.

covariance error estimate $P(k|k)$ by the standard formulas [11].

IV. Implementation and Results

A. System Architecture

The overall system architecture is shown in Fig. 3. The filter maintains current estimates of drift displacement and error covariance. The drift displacement is written onto the offset registers of the controller. This is equivalent to a change of origin. For example, if the x offset is 20 nm and a tip motion to x = 10 nm is requested, the controller applies to the x piezo the voltage required to move the tip by 30 nm. If the drift estimate was perfect, this procedure would ensure that successive requests for scans from x = 0 to x = 500 nm, say, would produce always the same image. The filter also maintains the last acquired image or particle location (depending on which drift measurement technique is used), together with the corresponding offsets and sampling times.

The filter scheduler requests estimate updates either without or with a corresponding measurement. In the first instance, updates for the drift state and the error covariance are computed by using the prediction equations (9) and (10). This computation is done at specified sampling times, typically a few seconds a part, and is very fast. (The sampling interval is a user-modifiable system parameter). A measurement request is more complicated and time consuming, because it implies physical motion of the tip. It generates a job request to the software that controls the AFM. This job may involve an imaging scan or a series of line scans, depending on the measurement technique being used. The current drift estimate is passed to the controller to ensure that the window or line scans are well positioned (see Section III.A). Measurements are scheduled when the error covariance exceeds a user-specified threshold (or at user-specified times).

The image or scan line data that result from a measurement job are passed to the translation analyzer, which compares them with the previous image or particle position and computes the displacement, taking into consideration the changes of origin associated with the offsets. The Kalman filter uses the measured drift and the current estimates to update the state and the covariance, and the process continues.

B. Hardware and Software

The system is implemented on an AutoProbe CP-R AFM (Park Scientific Instruments, now Veeco Instruments). The drift compensation software runs on top of our own Probe Control Software (PCS) for nanomanipulation, which in turn is implemented through the vendor supplied API (Application Programming Interface). (A version of PCS is now commercially available.) The API maintains its own job queue, with no preemptive scheduling. Before any job that involves tip motion is executed, we check for updated offset values, to ensure that we compensate for drift in all imaging and manipulation operations.

C. Selection of Drift Model Parameters

The Kalman filter implementation requires numeric values for the parameters α, σ_m^2, and R_0. In the current implementation we select these values heuristically, based on experimental results. A systematic investigation of the sensitivity of the filter to these parameter values has not been done yet, and we have not yet tried to optimize them.

We measure drift values and compute numerically the corresponding auto-correlation function for the drift acceleration. Experimental results show that, after the instrument has run for a couple of hours and is relatively stable, the drift auto-correlations can be approximated by an exponential $\alpha \sim 2\times10^{-4}$ s^{-1}.

The variance of drift acceleration, σ_m^2, is computed by using a heuristic expression due to Singer [12]

$$\sigma_m^2 = \frac{A_{\max}^2}{3}[1 + 4P_{\max} - P_0] \tag{14}$$

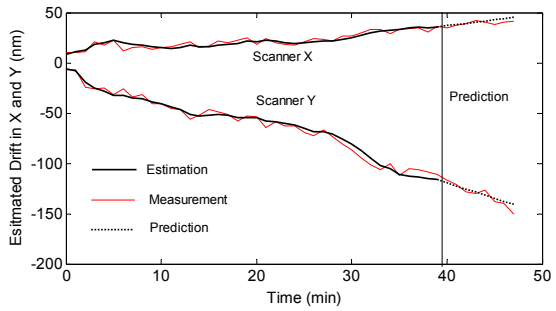
where $A_{\max}$, $P_{\max}$ and P_0 are the maximum possible acceleration for the drift, the probability of being in maximum acceleration, and the probability of being in no acceleration, respectively.

Finally, the variance of the measurement process, R_0, has been set conservatively at 10 nm^2, based on the pixel size normally used in our measurements.

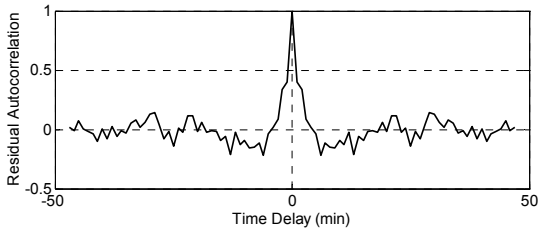
D. Experimental Results

The following results were obtained on the AutoProbe AFM with a sharp Ultralever™ tip in dynamic mode, imaging and manipulating gold nanoparticles with nominal diameters of 15 nm, deposited on a mica surface, covered with poly-L-lysine, in air, at room temperature and humidity.

Fig. 4.a depicts the estimated, measured and predicted (before measurement) values of the drift in the x and y directions. The sampling interval is 60 seconds, and the state is updated using a measurement taken at each sampling time until t = 39 minutes. From then on, we predict the state but do not update it with the measured values. It is apparent from the figure that the filtered estimate smooths the measured values. The predicted values are close to the measurements, even for some 10 minutes after the 39 minute mark, when we use just the predictions, without measurement-based updates. In theory, the measurement residual, defined as $z(k) - H\hat{\mathbf{x}}(k|k)$, is a white noise process. If the observed residual is approximately white, this is a strong indication that the filter and associated



(a)



(b)

Figure 4. (a) Estimation of drift in x and y direction for T=1 min, σ_m^2 =0.03 nm^2/min^4, α =0.012 min^{-1} and R_0=10 nm^2. After time t=39 min the filter does not use measurements for updating the states, and the prediction is used as the instrument offset, (b) normalized autocorrelation for the measurement residual.

models are adequate. The autocorrelation for the residual in a typical experiment is shown in Fig 4.b, and is close to the spike that corresponds to white noise. Thus, even with non-optimized parameters, the filter is performing well.

Initial experiments on automatic sequential manipulation of nanoparticles demonstrate for the first time manipulation over a relatively long duration without human intervention. Figure 5 shows the result of manipulating four 15 nm gold particles at 3.5 minutes intervals. The filter was sampling with a 30 seconds interval, but measurements were done only before pushing the particles. Similar experiments without drift compensation fail.

V. Conclusions and Future Research

Drift is a major cause of spatial uncertainty in AFMs. It causes distortion in AFM images, and it has even more deleterious effects on nanomanipulation, where it is often responsible for outright failure of the desired operations. Drift compensation in today's instruments is done primarily through user interaction.

This paper presents a Kalman filtering approach to drift estimation. The filter updates the origin of the AFM coordinates at a sampling rate on the order of one sample per minute, scheduling measurements when the covariance of the error exceeds a given threshold. AFM tip motions are always executed in the updated coordinate system and become largely immune to drift. Sequential manipulation of nanoparticles over

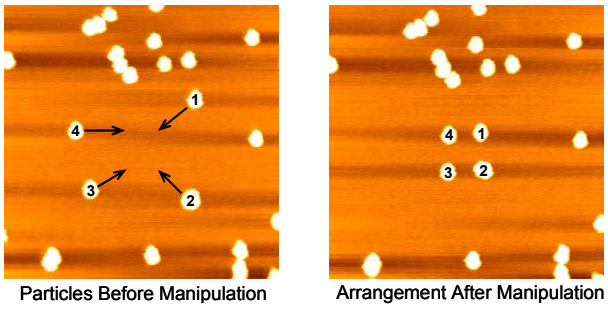


Figure 5. Sequential manipulation of 15nm gold particles using the Kalman filter compensator. Between each manipulation there is a 3.5 min time delay.

a relatively long period of time without user intervention is demonstrated. Without drift compensation such operations almost surely fail. These results are a first step towards the automatic construction by AFM manipulation of nanostructures much more complex and useful than those which can be built today.

Future work in this area includes a systematic approach to the identification of parameter models, and a study of the impact of these parameters on system performance. More accurate drift measurement methods also are needed (e.g., by removing the effect of tip convolution from the sampling images). Compensation for other sources of spatial uncertainty such as creep and hysteresis must be developed. This work will culminate with the demonstration of fully automatic construction of complex nanoassemblies.

References

[1] A. A. G. Requicha, "Nanorobots, NEMS and Nanoassembly", *Proc. IEEE*, Vol. 91, No. 11 , pp. 1922-1933, November 2003.

[2] A. A. G. Requicha, "Nanorobotics", in S. No., Ed., *Handbook of Industrial Robotics*. New York, NY: Wiley, 2nd. ed., pp. 199-210, 1999.

[3] http://www.asylumresearch.com/

[4] V. Y. Yurov and A. N. Klimov, "Scanning tunneling microscope calibration and reconstruction of real image: Drift and slope elimination", *Rev. Sci. Inst.*, Vol. 65, No. 5, pp. 1551-1557, May 1994.

[5] R. Staub, D. Alliata and C. Nicolini, "Drift elimination in the calibration of scanning probe microscopes", *Rev. Sci. Inst,*. Vol. 66, No. 3, pp. 2513-2516, March 1995.

[6] J. T. Woodward and D. K. Schwartz,"Removing drift from scanning probe microscope images of periodic samples", *J. Vac. Sci. Technol. B*, Vol. 16, No. 1, pp. 51-53, January/February 1998.

[7] S. H. Huerth and H. D. Hallen,"Quantitative method of image analysis when drift is present in a scanning probe microscope", *J. Vac. Sci Technol. B*, Vol. 21, No. 2, pp. 714-718, March 2003.

[8] K. J. Ito, Y. Uehara, S. Ushioda and K. Ito,"Servomechanism for locking scanning tunneling microscope tip over surface nanostructures", *Rev. of Sci. Inst.*, Vol. 71, No. 2, pp. 420-423, February 2000

[9] R. Haralick and L. Shapiro, *Computer and Robot Vision*. Reading, MA: Addison-Wesley Inc., Vol. 2, 1993.

[10] A. Oppenheim, and R. Schafer. *Discrete-Time Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall Inc., 1989.

[11] Y. Bar-Shalom and Rong Li X., *Estimation with Application to Tracking and Navigation*. New York, NY: John Wiley & Sons Inc., 2001.

[12] R. A. Singer, "Estimating optimal tracking filter performance for manned maneuvering targets", *IEEE Trans. on Aerospace and Electronic Systems*, Vol. 6, No. 4, pp. 473-483, July 1970.